PEDECIBA Informática
Instituto de Computación – Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay

Tesis de Maestría
en Informática

Defectos dentro y fuera del alcance de técnicas de verificación : resultados del análisis de una familia de experimentos

Cecilia Apa

2015

Cecilia Apa
Defectos dentro y fuera del alcance de técnicas de verificación:
Resultados del análisis de una familia de experimentos
ISSN 0797-6410
Tesis de Maestría en Informática
Reporte Técnico RT 15-07
PEDECIBA
Instituto de Computación – Facultad de Ingeniería
Universidad de la República.
Montevideo, Uruguay, 2015

Tesis de Maestría

Defectos dentro y fuera del alcance de Técnicas de Verificación: Resultados del Análisis de una Familia de Experimentos

Cecilia Apa
ceapa@fing.edu.uy

Tutor
Sira Vegas

Director de Estudios
Regina Motz

Montevideo, Uruguay

Julio, 2014

Resumen

En este trabajo se desarrolla y aplica un proceso de análisis estadístico para un conjunto de replicaciones de una familia de experimentos diseñados y ejecutados por la Universidad Politécnica de Madrid (UPM).

La familia de experimentos de UPM fue diseñada con el objetivo de responder a la siguiente pregunta de investigación: *Analizar la aplicación de técnicas de verificación con el propósito de conocer su efectividad a nivel unitario respecto de distintos tipos de defectos, en el contexto de un experimento controlado llevado a cabo por estudiantes universitarios. El conjunto de técnicas a analizar son a nivel unitario y está compuesto por: Lectura por abstracciones sucesivas (reading by stepwise abstraction), Particiones en clases de equivalencia y Criterio de cubrimiento de decisión.*

El objetivo principal de este trabajo es el estudio de técnicas de análisis estadístico para experimentos controlados, aplicados a la serie de experimentos de UPM. La metodología a seguir consta de tres etapas:

- Especificación de un procedimiento de análisis para los experimentos, basado en el estudio de otros análisis a experimentos controlados de la literatura y de las técnicas de análisis estadístico aplicables a este tipo de experimentos.
- Aplicación del procedimiento de análisis generado a cada uno de los experimentos de la serie de UPM.
- Reporte sobre los resultados del análisis, con la subsiguiente interpretación de los mismos.
- Propuestas de mejoras para el diseño del experimento y la evolución de la investigación en general.

Para lograr este objetivo se realiza un estudio de la teoría y fundamentos de la verificación y validación en ingeniería de software, haciendo foco en las técnicas a nivel unitario y en especial las que se utilizan para la familia de experimentos de UPM. Se estudia también la teoría de la Ingeniería de Software Empírica, especialmente en el diseño, ejecución y análisis de experimentos controlados. Además, se participa como replicador responsable en una de las replicaciones que componen el conjunto de replicaciones a analizar.

Índice general

Índice de figuras

Índice de cuadros

Capítulo 1

Introducción

Desde su surgimiento, con el paso de los años los sistemas de software han ido creciendo y evolucionando a gran escala, formando parte y dando sustento a actividades industriales, de negocios, científicas, sociales y de toda índole en todo el mundo. Para numerosas actividades, hoy en día es imposible pensar llevarlas a cabo sin el debido soporte informático. Este fenómeno ha hecho que en su evolución, el software se ha tornado cada vez más grande, complejo y crítico.

La calidad del software construido se ha convertido en un gran foco de atención, debido al rol que tiene en la competitividad del mercado, así como en el impacto negativo que produce un software de mala calidad (grandes pérdidas de dinero y hasta de vidas humanas). Dentro de la Ingeniería de Software (IS en adelante) la disciplina de verificación de software juega un papel fundamental en lo que refiere al aseguramiento de la calidad del software. Existe una gran variedad de técnicas, procesos y metodologías de verificación. Hoy en día no se conoce qué técnica o combinación de técnicas resulta ser más efectiva para la verificación del software.

La realización de experimentos controlados es una forma de conocer la efectividad y costo de técnicas de verificación. El uso de la estadística para un correcto análisis de los experimentos es fundamental para obtener resultados confiables y mejorar el proceso de experimentación. El uso de la estadística en experimentos en IS es un área que aún está muy verde y que necesita madurar para poder dar el debido soporte a los experimentos controlados. El principal foco de este trabajo es la investigación y estudio de técnicas de análisis para experimentos controlados, aplicadas a una serie de experimentos realizados en la Universidad Politécnica de Madrid desde el año 2006.

1.1. Motivación

No es común la utilización de datos recolectados con métodos científicos (y estadísticamente validados) para dar soporte al proceso de elección de las técnicas de verificación a utilizar en la evaluación de un sistema de software. La mayoría de las veces, esta elección se basa en experiencias de otros colegas u organizaciones, en modas, o simplemente en aquellas que resulten más familiares para quienes las vayan a

utilizar. Este tipo de elección es pobre y no asegura de ninguna forma que el resultado sea exitoso. Desafortunadamente, los datos necesarios para dar base a un mejor proceso de elección y que respondan a la pregunta "Cuál es la mejor combinación de técnicas de verificación que puedo utilizar para verificar un programa" hoy en día no se conocen o hay muy pocos.

Existen una gran variedad de técnicas de verificación, las cuales tienen distintos tipos de estrategias y focos (unidades de código, interfaces, el sistema como un todo, entre otras), basadas en distintos aspectos del software (el código, los requisitos, arquitectura del software, riesgos del proyecto, etc.) La efectividad de cada tipo de técnica al día de hoy no se conoce, además de que la efectividad resulta una métrica compleja, ya que depende de múltiples factores y no se ha llegado a un consenso de cómo es que se debe medir, o cuáles medidas serían adecuadas.

A pesar de esto, se han realizado numerosos estudios que intentan dar luz sobre la efectividad de las técnicas, entre éstos: aquellos que usan los experimentos controlados como método de investigación. Un problema general que tienen este tipo de investigaciones es la correcta utilización de las técnicas estadísticas para interpretar los resultados obtenidos y así retroalimentar el cuerpo de conocimiento que se va generando, para que éste impacte luego en las futuras investigaciones de forma apropiada.

En la Universidad Politécnica de Madrid (UPM en adelante) se realizaron una serie de experimentos controlados durante los años 2006 a 2012, los cuales no habían podido ser analizados de forma completa y consistente, en parte por el problema de la correcta utilización de los métodos estadísticos, previamente mencionado.

1.2. Objetivos

El objetivo principal de este trabajo es el estudio de técnicas de análisis estadístico para experimentos controlados, aplicados a la serie de experimentos de UPM. La metodología a seguir consta de tres etapas:

1. **Especificación de un procedimiento de análisis** para los experimentos, basado en el estudio de otros análisis a experimentos controlados de la literatura y de las técnicas de análisis estadístico aplicables a este tipo de experimentos.

2. **Aplicación del procedimiento de análisis** generado a cada uno de los experimentos de la serie de UPM, analizando e interpretando los resultados obtenidos.

3. **Reporte sobre el resultado del análisis y propuestas de mejoras para el diseño del experimento** y la evolución de la investigación en general.

De esta forma se logra no solamente la definición de un proceso de análisis para experimentos controlados, sino también la aplicación práctica del mismo y de cómo los resultados obtenidos ayudan a la mejora de la investigación en su totalidad.

1.3. Organización del Documento

En el capítulo 2 se explican los conceptos básicos de la Ingeniería de Software Empírica (en adelante ISE), con foco en las técnicas de experimentos controlados, terminología y etapas del proceso experimental. En el capítulo 3 se realiza un repaso sobre la teoría y fundamentos de la Verificación y Validación, ya que la familia de experimentos de UPM evalúan técnicas de verificación unitaria.

La familia de experimentos de UPM y el proceso de análisis se describen en el capítulo 4. Primeramente se presenta una revisión de la literatura relacionada con la utilización de experimentos controlados en IS, que fueron la base de la creación de la familia de experimentos de UPM. Luego se detalla el diseño experimental de UPM y la evolución que ha tenido la investigación en las sucesivas replicaciones que se hicieron desde el año 2006 a 2012. Por último se estudian técnicas de análisis estadístico que apliquen al diseño experimental descrito y se especifica un procedimiento de análisis para la serie de experimentos.

En los capítulos 5 y 6 se reportan los resultados de la aplicación del procedimiento de análisis a toda la serie de experimentos. En el capítulo 5 se detalla la aplicación del procedimiento de análisis estadístico al primer experimento de la serie. En el capítulo 6 se reportan los resultados obtenidos del resto de las replicaciones, más resumido y con menor detalle que en el capítulo 5. Finalmente, las conclusiones y trabajos a futuro se presentan en el capítulo 8.

Capítulo 2

Ingeniería de Software Experimental

La Ingeniería de Software Experimental es un área dentro de la Ingeniería de Software que utiliza el método experimental para mejorar y evolucionar el conocimiento existente sobre todo aquello que afecta el desarrollo y mantenimiento del software. La experimentación en IS, como cualquier otro proceso experimental, consiste en un proceso iterativo de generación y refinamiento del conocimiento. Modelos, técnicas, productos o procesos son construídos, se generan hipótesis sobre estos productos y se ponen a prueba, la información aprendida es utilizada para confirmar, refutar o refinar las viejas hipótesis o construir nuevas [BSH86].

La experimentación en IS es un área relativamente nueva comparada con otras áreas de la ciencia como la medicina o la biología. Esta área ha causado un impacto considerable en la comunidad científica y en la industria, teniendo su propia revista internacional (Empirical Software Engineering: An International Journal)[1] desde el año 1996, conferencias internacionales y red de investigación relacionadas a la experimentación como ser ISERN (International Software Engineering Research Network)[2]. Su objetivo es mejorar y optimizar los procesos de construcción y mantenimiento de software utilizando el proceso experimental.

A pesar de que los estudios empíricos en IS ha ido creciendo a lo largo de los años, gran parte del conocimiento que hoy se tiene carece de la validación empírica necesaria. En 1997 Zelkowitz y Wallace [ZW97] realizan encuestas sobre estudios en IS publicados entre 1985 y 1990, los resultados muestran que al menos la mitad de estos artículos tienen un inadecuado nivel de validación empírica, las afirmaciones son derivadas de la intuición y experiencia de los autores o a lo sumo alguna prueba de concepto o caso de estudio. Esta situación ha ido mejorando y en una nueva encuesta realizada entre 2000 y 2005, los resultados muestran que el porcentaje de artículos que incluyen validaciones han ido en aumento, llegando a un 60 %.

La experimentación en IS es necesaria para la evolución del conocimiento, pero a

[1] http://www.springer.com/computer/programming/journal/10664
[2] http://isern.iese.de

la vez es difícil de realizar [BSH86, WRH+12]. El diseño experimental es complejo y el proceso experimental es propenso a errores [BDM+94]. Para asegurarse que los resultados obtenidos en un determinado experimento no sean fruto de la casualidad, los mismos deben poder ser observados mediante la repetición o reproducción del experimento original.

La importancia de la replicación ha sido mencionada por diversos autores [Mil05, BDM+94, SCVJ08], ésta permite dar luz sobre los resultados obtenidos previamente, refutando, confirmando o ampliando los mismos. La actividad de replicación no es una tarea fácil, es necesario comprender el área de investigación sobre la cual se investiga, así como conocer las técnicas, herramientas y teoría experimental. Además, la replicación es una actividad que demanda recursos: materiales, procedimientos, personas, tiempo y otros, por lo cual muchas veces se torna muy costosa de realizar.

Para facilitar esta tarea de replicación, Basili [BSL99] introduce el concepto de familia de experimentos como un framework para la organización de distintos tipos de estudios relacionados. Las familias de experimentos tienen un marco común de investigación, en donde los experimentos no son visualizados de forma aislada sino como parte de un objetivo de alto nivel. Las hipótesis y conclusiones que pueden ser sugeridas por una familia de experimentos amplían y mejoran aquellas que pueden ser generadas por experimentos individuales. Las familias de experimentos fomentan la colaboración entre investigadores y facilitan el trabajo de replicación.

La replicación no es considerada solamente la repetición idéntica del experimento original, además de que en términos prácticos es imposible ejecutar el mismo experimento en idénticas condiciones, al menos ha pasado tiempo, o los sujetos han cambiado, o el lugar de replicación es distinto. Muchas veces las replicaciones se realizan en distintos lugares lo que implica restricciones de contexto diferentes o sujetos diferentes.

La replicación no tiene que ser idéntica para ser de utilidad, también replicaciones con ciertas variaciones son interesantes para la investigación en sí. Juristo y Vegas [JV09, JV11] animan a realizar replicaciones no idénticas, ya que éstas ayudan a entender mejor las relaciones entre variables en el desarrollo de software, donde de hecho es imposible reproducir idénticas condiciones.

Existen varias clasificaciones de replicaciones, dependiendo del punto de vista desde el cual se las tipifica [GJV10, GÍ2]. Dependiendo de si la replicación la realiza el mismo grupo de investigación o no, se clasifican en **internas** y **externas**. Las replicaciones externas permiten asegurar que los resultados observados no son influenciados por el investigador que llevó a cabo la replicación [BDM+96]. Brooks [BDM+94] también afirma que, sin el poder de confirmación de las replicaciones externas, los resultados deberían aceptarse únicamente de forma provisional.

Cuando el contexto experimental varía o bien el diseño experimental cambia moderadamente, las clasificaciones varían entre idénticas, cercanas, diferenciadas o distintas. Debido a la complejidad del diseño experimental y a la cantidad de tipos distintos de cambios que éste puede tener, no está definido de forma exacta cuándo las replicaciones pertenecen a un tipo u otro, pero dan una idea de la similitud con el experimento original y de los objetivos de las mismas.

Otra forma de clasificar las replicaciones es dependiendo del nivel de colaboración de

los investigadores entre el experimento original (o base) y la replicación. Si bien pueden haber distintos grados de colaboración, las replicaciones **dependientes** son aquellas en las cuales participan los mismos investigadores que llevaron a cabo el experimento original, **semi-independientes** si parte de los experimentadores del experimento original participan en la replicación realizada por otros experimentadores, mientras que en las **independientes** los investigadores que participan son distintos a los del experimento original.

Todos los tipos de replicaciones son válidos, necesarios y exploran distintos aspectos que deben ser validados, aportando nueva información al cuerpo de conocimiento. Además de las replicaciones, existen otros tipos de estudio que tienen como objetivo verificar y validar resultados de los experimentos. El **re-análisis** toma datos resultantes de la realización de un experimento y efectúa un nuevo análisis estadístico (utilizando el mismo método de análisis o variándolo en caso de que lo considere necesario), el cual tiene como objetivo confirmar los resultados obtenidos previamente. En la **reproducción** se realiza el experimento desde cero, no reutilizando ningún material, método o artefacto del experimento original. La reproducción sirve para verificar que los resultados no han sido producto del proceso experimental aplicado.

La coordinación de estudios y materiales generados durante los experimentos y replicaciones requiere una organización. Es necesario tener disponible los diseños, materiales y resultados de otros experimentos, así como también la evolución de la investigación a través del tiempo. Esto permite que no se malgasten recursos y que las nuevas actividades se realicen de la forma más eficiente posible. Para esto Solari [Sol11] propuso una estructura de paquete de laboratorio específica para estudios en IS, como forma de organizar los estudios correspondientes a una misma familia de experimentos.

2.1. Enfoques y Estrategias

Para poder realizar estudios empíricos se deben conocer los conceptos, las técnicas y las herramientas normalmente usadas en ISE. Esta sección se basa casi completamente en los libros *Experimentation in Software Engineering: An Introduction* [WRH99], *Basics of Software Engineering Experimentation* [JM01] y *Software Metrics - A Rigorous And Practical Approach* [FP98].

La ISE utiliza métodos y técnicas experimentales como instrumentos para la investigación. La evidencia empírica proporciona un soporte para la evaluación y validación de atributos (p.e. costo, eficiencia, calidad) en varios tipos de elementos de IS (p.e. productos, procesos, técnicas, etc.). Se basa en la experimentación como método para corresponder ideas o teorías con la realidad, la cual refiere a mostrar con hechos las especulaciones, suposiciones y creencias sobre la construcción de software.

El proceso de investigación que se ilustra en la figura 2.1 es un proceso cíclico de aprendizaje. Se compone de tres formas de razonamiento: inducción, deducción y abducción. La inducción parte de la teoría y la observación de fenómenos a partir de los cuales se generan hipótesis, la deducción es la prueba de esa hipótesis mediante algún mecanismo, confirmando o refutando de que la hipótesis ocurre en el mundo real. Los resultados obtenidos generan nuevo conocimiento y nuevas hipótesis son generadas

para poder ser comprobadas.

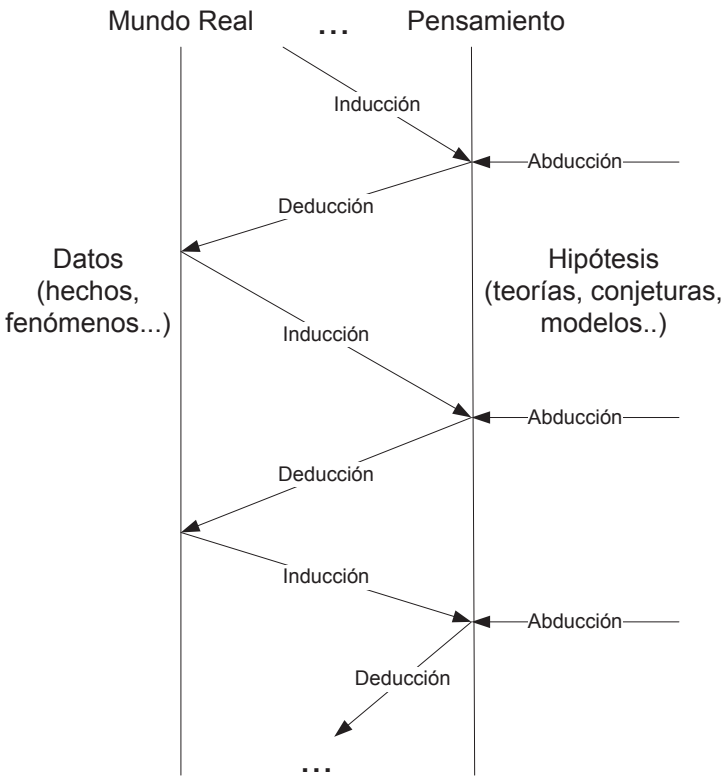

Figura 2.1: Proceso cíclico de investigación

Se pueden distinguir dos enfoques diferentes al realizar una investigación empírica: el enfoque cualitativo y el cuantitativo. El enfoque **cualitativo** se basa en estudiar la naturaleza del objeto y en interpretar un fenómeno a partir de la concepción que las personas tienen del mismo. Los datos que se obtienen de estas investigaciones están principalmente compuestos por texto, gráficas e imágenes, entre otros.

El enfoque **cuantitativo** se corresponde con encontrar una relación numérica entre dos o más grupos. Se basa en cuantificar una relación o comparar variables o alternativas bajo estudio. Los datos que se obtienen en este tipo de estudios son siempre valores numéricos, lo que permite realizar comparaciones y análisis estadístico.

Es posible utilizar los enfoques cualitativos y cuantitativos para investigar el mismo fenómeno, pero cada enfoque responde a diferentes interrogantes. Se puede considerar que estos enfoques son complementarios más que competitivos, ya que el enfoque cualitativo puede ser usado como base para definir la hipótesis que luego puede ser correspondida cuantitativamente con la realidad. Cabe destacar que las investigaciones cuantitativas pueden obtener resultados más justificables y formales que los cualitativos.

Hay 3 tipos principales de técnicas o estrategias para la investigación empírica: las encuestas, los casos de estudio y los experimentos.

Las **encuestas** se utilizan o bien cuando una técnica o herramienta ya ha sido

usada o antes de comenzar a hacerlo. Son estudios retrospectivos de las relaciones y los resultados de una situación. Se puede realizar este tipo de investigación cuando una técnica, o herramienta ya ha sido utilizada o antes de que ésta sea introducida. Las encuestas son realizadas sobre una muestra representativa de la población, y luego los resultados son generalizados al resto de la población. El ámbito donde son más usadas es en ciencias sociales, por ejemplo, para determinar cómo la población va a votar en la siguiente elección.

En la Ingeniería de Software Empírica las encuestas se utilizan de forma similar, se obtiene un conjunto de datos de un evento que ha ocurrido para determinar cómo reacciona la población frente a una técnica, herramienta o método particular, o para determinar relaciones o tendencias. En un estudio es fundamental seleccionar correctamente las variables a estudiar, pues de ellas dependen los resultados que se pueden obtener. Si los resultados no permiten concluir sobre los objetivos del estudio se han elegido mal las variables.

Una de las características más relevantes de las encuestas es que proveen un gran número de variables para estudiar. Esto hace posible construir una variedad de modelos y luego seleccionar el que mejor se ajusta a los propósitos de la investigación, evitando tener que especular cuáles son las variables más relevantes. Dependiendo del diseño de la investigación (cuestionario) las encuestas pueden ser clasificadas como cualitativas o cuantitativas.

Los **casos de estudio** son métodos observacionales, se basan en la observación de una actividad o proyecto durante su curso. Son utilizados para monitorear proyectos, o actividades y para investigar entidades o fenómenos en un período específico.

En un caso de estudio se identifican los factores clave que pueden afectar la salida de una actividad, y se documentan las entradas, las limitaciones, los recursos y las salidas. El nivel de control de la ejecución es menor en los casos de estudio que en los experimentos. Esto se debe principalmente a que en los casos de estudio no se controla, sólo se observa, contrario a lo que ocurre en los experimentos.

Los casos de estudio son muy útiles en IS, se usan en la evaluación industrial de métodos y herramientas. Además, son fáciles de planificar aunque los resultados son difíciles de generalizar y comprender. Los casos de estudio no manipulan las variables, sino que éstas son determinadas por la situación que se está investigando.

Al igual que las encuestas, los casos de estudio pueden ser clasificados como cualitativos o cuantitativos dependiendo de lo que se quiera investigar del proyecto en curso.

Los **experimentos** son generalmente ejecutados en un ambiente controlado. El objetivo en un experimento es manipular una o más variables y controlar el resto. Un experimento es una técnica formal, rigurosa y controlada de llevar a cabo una investigación.

Un aspecto muy importante a la hora de ejecutar los experimentos es el ambiente de ejecución, tanto si el experimento se realiza dentro de un proyecto de desarrollo común o si se crea un ambiente ficticio para su ejecución. Además del ambiente de ejecución , también es importante el tipo de interacción que tienen los sujetos en dicho ambiente.

Tomando en cuenta estas dos variables (ambiente e interacción de los sujetos con éste), los experimentos pueden clasificarse básicamente en 4 tipos [TB03]:

- Experimentos ***in vivo***: involucran a personas (sujetos) en su propio entorno. Por ejemplo: experimentos realizados en organizaciones de desarrollo de software, que involucren una o varias etapas del proceso de desarrollo, con las personas de la propia organización.

- Experimentos ***in vitro***: se ejecutan en un entorno controlado, ya sea un laboratorio o una comunidad controlada. En IS, la mayoría de los experimentos in vitro se realizan en las universidades o en grupos cuidadosamente seleccionados de una organización de desarrollo de software.

- Experimentos ***in virtuo***: implican interacción entre los sujetos y un modelo computarizado de la realidad. En este tipo de experimentos, el comportamiento del entorno con el que interactuan los sujetos se describe como un modelo y es representado por una aplicación de software. En IS, estos estudios se llevan a cabo en universidades y laboratorios de investigación y se caracterizan por pequeños grupos de sujetos manipulando simuladores.

- Experimentos ***in silico***: estos estudios se caracterizan tanto por sujetos del mundo real como los descritos por modelos computarizados. En este caso, el entorno (o contexto) está compuesto por modelos numéricos en los que no se permite la interacción humana. Debido a la necesidad de una gran cantidad de conocimiento, los estudios in silico son todavía muy poco comunes en IS, ya que se limita a las áreas en donde la participación del sujeto no es un tema de estudio experimental o bien la inteligencia artificial puede reemplazar a los sujetos humanos. Por ejemplo: podemos encontrar estudios in silico aplicados a la experimentación en usabilidad de software, como ser la caracterización del rendimiento o performance del software.

En las secciones siguientes se profundiza en los experimentos controlados in vitro como técnica de investigación.

2.2. Experimentos Controlados

Como se mencionó anteriormente, los experimentos son una técnica de investigación en la cual se quiere tener un mejor control del estudio y del entorno en el que éste se lleva a cabo.

Los experimentos son apropiados para investigar distintos aspectos de la IS, como ser: confirmar teorías, explorar relaciones, evaluar la exactitud de los modelos y validar medidas. Tienen un alto costo respecto de las otras técnicas de investigación, pero a cambio ofrecen un control total de la ejecución y en general es posible reproducir la mayoría de las condiciones del mismo para que pueda ser replicado con exactitud.

2.2.1. Terminología

En esta sección se presentan los términos más comunmente usados en el diseño experimental. Se usan dos ejemplos de experimentos a lo largo de esta sección para introducir dichos términos.

En el primer ejemplo se tiene un experimento en el campo de la medicina, mediante el cual se quiere conocer la efectividad de los analgésicos en las personas entre 20 y 40 años de edad, llamado "Efec-Analgésicos".

En el segundo ejemplo, se quiere conocer la efectividad de 5 técnicas de verificación sobre un conjunto de programas, llamado "Efec-Técnicas".

Los objetos sobre los cuales se ejecuta el experimento son llamados **Unidades Experimentales** u objetos experimentales. La unidad experimental en un experimento de IS podría llegar a ser el proyecto de software como un todo o cualquier producto intermedio durante el proceso.

Para *Efec-Analgésicos* se tiene que la unidad experimental es un grupo de personas entre 20 y 40 años de edad, en ese grupo de personas es en donde se observa el efecto de los analgésicos. En el ejemplo de *Efec-Técnicas*, se tiene que la unidad experimental es el conjunto de programas sobre los cuales se aplican las técnicas de verificación.

Aquellas personas que aplican los métodos o técnicas a las unidades experimentales se les llama **Sujetos Experimentales**. A diferencia de otras disciplinas, en la IS los sujetos experimentales tienen un importante efecto en los resultados del experimento, por lo tanto es una variable que debe ser cuidadosamente considerada.

En *Efec-Analgésicos* los sujetos son aquellas personas que administran los analgésicos a ser consumidos por los pacientes (enfermeros por ejemplo). Cómo los enfermeros administran los analgésicos a los pacientes no es algo que se espere vaya a afectar el experimento. La forma en que un enfermero administra un analgésico a un paciente es poco probable que sea diferente a la de otro, y aunque lo fuera, no se espera que afecte los resultados del experimento.

En *Efec-Técnicas* los sujetos pueden ser ingenieros que aplican la técnica en un conjunto particular de programas (unidad experimental). En este caso, los resultados del experimento podrían diferir mucho de acuerdo a la formación y experiencia de los ingenieros, así como también la forma en que las técnicas son aplicadas, incluso el estado de ánimo del verificador podría influir en los resultados.

El resultado de un experimento es llamado **Variable de Respuesta**. Este resultado debe ser cuantitativo. Una variable de respuesta puede ser cualquier característica de un proyecto, fase, producto o recurso que es medida para verificar los efectos de las variaciones que se provocan de una aplicación a otra. En ocasiones, a una variable de respuesta se le llama también variable dependiente.

En *Efec-Analgésicos* la efectividad podría ser medida en el grado de alivio del dolor en un determinado lapso de tiempo, o bien qué tan rápido el analgésico alivia el dolor. En ambos casos, la variable debe ser expresada cuantitativamente. En el primer caso se podría tener una escala, en la cual cada valor signifique un grado de alivio del dolor, en el segundo caso, el lapso de tiempo en que el analgésico es efectivo, se podría medir

en minutos.

Para *Efec-Técnicas* la efectividad podría ser medida de acuerdo a la cantidad de defectos que encuentra la técnica sobre la cantidad de defectos totales del software verificado.

Un **Parámetro o variable de contexto** es cualquier característica que permanezca invariable a lo largo del experimento. Son características que no influyen o que no se desea que influyan en el resultado del experimento o en la variable de respuesta. Los resultados del experimento serán particulares a las condiciones definidas por los parámetros. El conocimiento resultante podrá ser generalizado solamente considerando los parámetros como variables en sucesivos experimentos y estudiando su impacto en las variables de respuesta.

En el ejemplo de *Efec-Analgésicos* se tiene que el rango de edades (entre 20 y 40 años de edad) es un parámetro del experimento, los resultados serán particulares para el rango establecido.

En *Efec-Técnicas* un parámetro posible es el tamaño del software a ser verificado (por ejemplo: que tenga entre 200 y 500 LOCs). Otro parámetro para este experimento podría ser la experiencia de los verificadores, en este caso se podría fijar la experiencia en un determinado nivel.

Cada característica del desarrollo de software a ser estudiada que afecta a las variables de respuesta se denomina **Factor**. Cada factor tiene varias alternativas posibles. Lo que se estudia, es la influencia de las alternativas en los valores de las variables de respuesta. Los factores de un experimento son cualquier característica que es intencionalmente modificada durante el experimento y que afecta su resultado.

El factor en *Efec-Analgésicos* es "los analgésicos", en *Efec-Técnicas* tenemos que el factor es "las técnicas de verificación". Para ambos casos el factor se varía intencionalmente (se varía el tipo de analgésico o tipo de técnica de verificación) para ver cómo afecta en la efectividad.

Los posibles valores de los factores en cada unidad experimental son llamados **Alternativas** o niveles. En algunos casos también se les llama tratamientos.

Las alternativas de *Efec-Analgésicos* son los distintos tipos de analgésicos que se estudian en el experimento (p.e. Aspirina, Zolben, etc). De igual forma, para *Efec-Técnicas* las distintas alternativas son los 5 tipos distintos de técnicas que se estudian.

El intento de ajustar determinadas características de un experimento a un valor constante no es siempre posible. Es inevitable y a veces indeseable tener variaciones de un experimento a otro. Éstas variaciones son conocidas como **Bloqueo de Variables** y dan lugar a un determinado tipo de diseño experimental, llamado *block design*.

Una variable indeseada para *Efec-Analgésicos* podría ser el "umbral del dolor". Si se aplica una alternativa de analgésico a personas con umbral del dolor alto y otra alternativa a personas con umbral del dolor bajo, se tendría una variación indeseada, ya que la efectividad que se mida de los distintos tipos de analgésico va a variar no solamente por el tipo de analgésico administrado sino por el nivel de umbral del dolor del paciente al cual se lo administra.

En el caso de *Efec-Técnicas*, podría resultar que la experiencia de los verificadores resultase una variación indeseada si no se la tiene en cuenta previamente. Una forma de bloquear la experiencia en verificación podría ser dividir a los participantes en dos grupos: uno de verificadores experientes y otro sin experiencia.

Cada ejecución del experimento que se realiza en una unidad experimental es llamada **experimento unitario** o experimento elemental. Lo que significa que cada aplicación de una combinación de alternativas de factores por un sujeto experimental en una unidad experimental es un experimento elemental.

Un experimento elemental es cada terna <analgésico$_i$, enfermero$_j$, paciente$_k$> para el ejemplo de *Efec-Analgésicos*. Para el ejemplo de *Efec-Técnicas* sería la terna <técnica$_i$, verificador$_j$, software$_k$>.

La figura 2.2 ilustra la interacción entre los distintos tipos de componentes de un experimento.

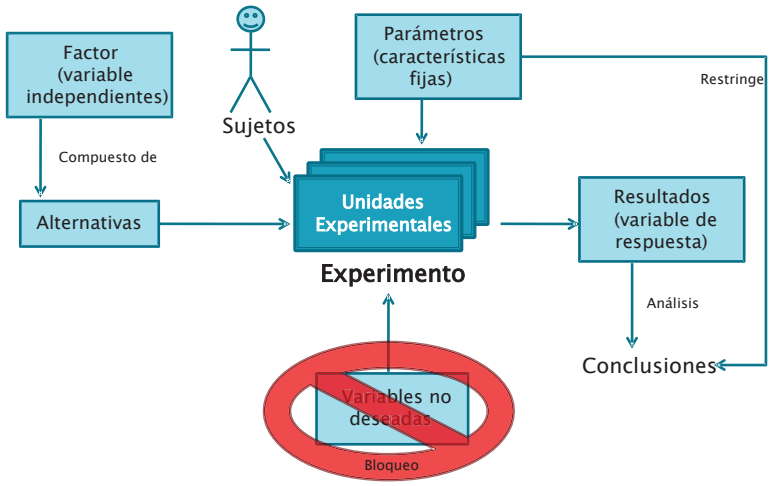

Figura 2.2: Componentes en un experimento de Ingeniería de Software

2.2.2. Principios generales de diseño

Muchos aspectos deben ser tenidos en cuenta cuando se diseña un experimento. Los principios generales de diseño son: aleatoriedad, bloqueo y balanceo. A continuación se describe en qué consiste cada principio.

Aleatoriedad: el principio de aleatoriedad es uno de los principios de diseño más importantes. Todos los métodos de análisis estadístico requieren que las observaciones sean de variables independientes aleatorias. Por consiguiente, tanto las alternativas de los factores como los sujetos tienen que ser elegidos de forma aleatoria, ya que los sujetos tienen un impacto crítico en el valor de las variables de respuesta.

La aleatoriedad que se puede aplicar a un experimento también depende del tipo de diseño que se haya elegido. Por ejemplo, si se tienen dos factores A y B, cada uno con dos posibles alternativas (a1, a2, b1 y b2), las alternativas deben ser combinadas de la

siguiente forma: a1b1, a1b2, a2b1, a2b2, ya que cuando se tienen dos factores se quiere observar el efecto de cada alternativa por separado y de la interacción entre ambas.

Esta combinación de alternativas es especificada por el tipo de diseño experimental que se eligió. Sin embargo, las cuatro combinaciones deben ser asignadas de forma aleatoria a los proyectos y sujetos, y es ahí en donde la aleatoriedad se aplica.

Bloqueo: la técnica de bloqueo se usa cuando se tienen factores que probablemente tengan efectos indeseados en las variables de respuesta y éstos efectos son conocidos y controlables.

Como se mencionaba en el ejemplo de *Efec-Técnicas* en la sección anterior, algunos verificadores podrían tener experiencia en el uso de las técnicas de verificación y otros no. Entonces, para minimizar el efecto de la experiencia, se agrupan a los participantes en dos grupos, uno con verificadores experientes y otro sin experiencia.

Balance: el balance es deseable ya que simplifica y fortalece el análisis estadístico de los datos, aunque no es necesario. Tomando como ejemplo el experimento de *Efec-Analgésicos* nuevamente, sería deseable que la cantidad de personas a las cuales se les administra Zolben sea igual a la cantidad de personas que se les administra Aspirina.

2.2.3. Tipos de Diseño

En el proceso del diseño experimental, primero se debe decidir (basándose en los objetivos del experimento) a qué factores y alternativas estarán sujetas las unidades experimentales y qué parámetros deben ser establecidos. Luego, se debe examinar si existe la posibilidad de que algunos de los parámetros no pueda mantenerse en un valor constante, en ese caso se debe tener en cuenta cualquier variación indeseable. Finalmente, se debe elegir qué variables de respuesta serán medidas y cuáles serán los objetos y sujetos experimentales.

Teniendo establecidos los parámetros, factores, variables de bloqueo y variables de respuesta, se debe elegir el tipo de diseño experimental, en el cual se establece cuántas combinaciones de experimentos unitarios y alternativas deben haber.

Los distintos tipos de diseño experimental dependen del objetivo del experimento, del número de factores, de las alternativas de los factores y de la cantidad de variaciones indeseadas, entre otros.

Los tipos de diseño experimental se dividen en diseños de *un solo factor* y diseños de *múltiples factores*. A continuación se profundiza en los experimentos de un solo factor.

2.2.3.1. Diseño de un solo factor (*One-Factor Design*)

Para experimentos con un solo factor existen distintos tipos de diseños estándar, los principales son: los completamente aleatorios y los aleatorios con comparación por pares.

Los diseños **completamente aleatorios** son los tipos de diseño más simples, en los cuales se intenta comparar dos o más alternativas aplicadas a un determinado número

de unidades experimentales, en donde cada unidad experimental se ve afectada una única vez, y por ende, por una sola alternativa. La asignación de las alternativas a los experimentos debe ser de forma aleatoria para asegurar la validez del análisis de datos.

Tomando como ejemplo *Efec-Técnicas* y suponiendo que el conjunto de programas sobre el cual se quiere conocer la efectividad de las técnicas lo componen diez programas distintos, se tendría que asignar las técnicas y los ingenieros de forma aleatoria a los programas que se vayan a verificar.

Una posible asignación aleatoria sería tener en una bolsa los nombres de todas las técnicas de verificación a aplicar, en donde la primera que se extraiga se aplique al programa P_1, la segunda a P_2 y así hasta el programa P_{10}. Luego de tener las duplas Programa-Técnica, efectuar la misma asignación aleatoria con los participantes: el primer participante extraído se lo asigna la dupla (P_1, T_x), el segundo a la dupla (P_2, T_y), y así sucesivamente.

El análisis estadístico que se puede hacer a este tipo de experimentos varía según si se aplican 2 o más alternativas para el factor.

Los diseños **aleatorios con comparación por pares** tienen como objetivo encontrar cuál es la mejor alternativa respecto de una determinada variable de respuesta. Estos tipos de diseño tienen la particularidad de que las alternativas se aplican al mismo experimento, instanciado en más de una unidad experimental.

Para el experimento de *Efec-Técnicas* no sería una buena decisión que cada ingeniero verificara 2 veces el mismo programa. En la segunda instancia de verificación, el ingeniero posee conocimiento tanto de los defectos del programa como de la tarea de verificar propiamente dicha (aunque sea con una técnica distinta). Por esto, para comparar las dos técnicas, ambas tienen que ser aplicadas por primera vez por ingenieros distintos, pero con similares características (ya que encontrar uno igual es imposible). La alternativa que debe aplicar cada ingeniero al programa debe ser asignada de forma aleatoria y no debe verificar un mismo programa más de una vez.

En este tipo de diseños se bloquean cierto tipo de variables que representan restricciones en la aleatoriedad que se le puede dar. Tomando como ejemplo nuevamente a *Efec-Técnicas*, si un verificador sin experiencia aplica más de una técnica durante el experimento, no sería deseable asignar al azar la técnica que cada verificador aplica en cada verificación.

Existe un efecto de aprendizaje en el cual, luego de que un verificador ejecutó una verificación, éste generó conocimiento sobre la verificación en sí, independientemente de la técnica que haya aplicado, y éste conocimiento influye significativamente en la segunda instancia de verificación que vaya a aplicar. Por tanto, la aleatoriedad en el orden de la asignación de técnicas en este ejemplo no es del todo deseable.

2.2.3.2. Diseños de Medidas Repetidas *(Repeated Measures)*

Para experimentos en IS muchas veces resulta difícil conseguir una cantidad de observaciones suficiente a modo de asegurar la validez estadística de los resultados. La mayoría de los experimentos que se realizan en contextos académicos están enmarcados

en el dictado de cursos, donde el promedio de estudiantes ronda las 30 personas[3]. Además, el costo de realización del experimento aumenta de forma casi proporcional a la cantidad de sujetos, tanto en recursos humanos (supervisión de pruebas, corrección, procesamiento y análisis de los datos) como en materiales e infraestructura (formularios, salones, computadoras, etc.). Esto genera dificultades para realizar experimentos con gran cantidad de sujetos.

En los diseños factoriales, el sujeto aplica un tratamiento por una única vez, teniendo como resultado una única observación. Para un experimento de un solo factor, la muestra de sujetos se debe divir entre la cantidad de alternativas que se quieren probar, teniendo $1/a$ observaciones por cada alternativa, siendo a el número de alternativas. Además, este tipo de diseños está amenazado por la variabilidad de los sujetos entre los grupos para cada tratamiento, en donde puede resultar que el grupo de sujetos que aplica un tratamiento, esté más o menos capacitado/motivado/etc. que otro. En el ejemplo de *Efec-Técnicas*, si se tuvieran 30 sujetos, éstos deberían dividirse entre los 5 tipos de técnica distintos, teniendo un total de 6 observaciones para cada técnica. Para experimentos con pocos sujetos, este tipo de diseño tiene sus desventajas.

Ante la imposibilidad de aumentar el número de sujetos, existe una alternativa de diseño que se puede usar para contrarrestar este efecto: utilizar **diseños de medidas repetidas** [WBM91]. Los diseños con medidas repetidas se basan en que un mismo sujeto aplica varios tratamientos de uno o varios factores, por lo que se tienen varias observaciones de un mismo sujeto para un mismo factor (una por cada alternativa), a este tipo de factores se les llama *factores intra-sujetos*. En el ejemplo, el factor técnica es un factor del tipo intra-sujetos.

2.2.3.3. Diseños Cross-over

Un tipo particular de diseño de medidas repetidas son los llamados diseños cross-over, también conocidos como *factorial cross-over designs* [Kue99]. En este tipo de diseño en particular, se tiene que todas las observaciones realizadas por un mismo sujeto corresponden a la aplicación de todos los tratamientos de cada factor. Este es un tipo especial de diseño de medidas repetidas . Es importante aclarar, que no todos los diseños de medidas repetidas son cross-over, por ejemplo, cuando el sujeto aplica más de una vez el mismo tratamiento (y no la totalidad de los tratamientos para ese factor), o cuando no es factorial completo.

Los diseños cross-over han sido utilizados en experimentos por múltiples disciplinas científicas, por ejemplo: sicología, educación, la ciencia farmacéutica y también en la salud (especialmente la medicina). También han sido muy populares en la experimentación en IS, siendo utilizados desde los comienzos de la IS experimental [BS87] y continúan siendo usados desde entonces [RWM97, ABHL06, PSG12, PUT+01].

Simplifiquemos el ejemplo de *Efec-Técnicas* reduciéndolo a 2 tratamientos: A y B. En el cuadro 2.1 se muestra la aplicación de un diseño cross-over a este ejemplo. La muestra de sujetos se divide en 2 subgrupos, en donde el primer subgrupo ejecuta la secuencia AB, en donde se aplica la técnica A en la primer sesión experimental y la

[3]Puede variar (aunque no mucho) de la institución y de la formación requerida en cada caso.

técnica B en la 2º sesión. El segundo subgrupo ejecuta la secuencia BA, con el orden invertido de aplicación de las técnicas.

Período / **Secuencia**	**Sesión 1**	**Sesión 2**
AB	Técnica A	Técnica B
BA	Técnica B	Técnica A

Cuadro 2.1: Ejemplo de diseño cross-over de 2 tratamientos

La utilización de diseños cross-over trae aparejado la introducción de nuevas variables que influencian a la variable respuesta: el *período* y la *secuencia*. Las secuencias representan el orden en el cual han sido aplicados los tratamientos. Los momentos en los cuales se aplica cada tratamiento se les llama período. En el ejemplo, las secuencias están representadas por las dos combinaciones de aplicación de las técnicas (AB y BA) y los períodos por las sesiones experimentales (Sesión 1 y Sesión 2).

Este tipo de diseño experimental tiene ventajas y desventajas. La ventaja más notoria es la mencionada anteriormente: se requieren menos sujetos que en los diseños completamente aleatorios, obteniéndose mayor cantidad de observaciones y por tanto mayor validez estadística para los resultados obtenidos. Además, permiten eliminar la variación residual debida a las diferencias entre los sujetos. Por ejemplo, en un diseño factorial, la muestra de sujetos se divide entre todos los tratamientos, puede ocurrir que un subgrupo de sujetos asignados a un tratamiento sean más capaces, estén más motivados u otra característica que influya sobre la efectividad de la técnica. Este efecto no se puede separar del efecto de la técnica en sí y por tanto ambos terminan confundidos. En los diseños cross-over, el efecto de un tratamiento sobre el $sujeto_i$ se mide en relación al promedio para todos los tratamientos. Como resultado, los sujetos controlan su variabilidad entre ellos. Como desventaja, resultan diseños muy complejos y han sido criticados por tener 2 grandes debilidades: son complejos de analizar y son sensibles al efecto de arrastre o de *carry-over*.

El efecto de *carry-over* ocurre cuando se administra un tratamiento antes de que haya finalizado el efecto de otro administrado previamente. Esto provoca que los tratamientos administrados con posterioridad aparenten ser más (o menos) efectivos que los administrados previamente. Este efecto puede generar un impacto importante, hasta incluso puede generar la invalidación de los datos. Incluso organizaciones como la *US Food And Drug Administration* (FDA) [CO76], investigadores en IS [KFL03] y otros investigadores de otras disciplinas [Fle89, Fre89, JK89] desaconsejan el uso de los diseños cross-over por causa del efecto de arrastre.

Es necesario comprender de forma adecuada qué puede ocasionar un efecto de carry-over y qué es lo que representa para no realizar análisis inadecuados. En [KFL03] el efecto de carry-over se define como la interacción período*tratamiento. Sin embargo, el carry-over es solamente una de las posibles interacciones entre período y tratamiento. Supongamos que en una de las sesiones de la ejecución del experimento los estudiantes están estresados (por un período de exámenes por ejemplo). Y que ese estrés afecta a una técnica más que a otras. Esa diferencia, si bien se visualiza en la interacción de

período*tratamiento, no es correcto atribuirla al carry-over, ya que la causa es el estrés. En este caso, la interacción del período con el tratamiento es confundida con el efecto de carry-over y también con el efecto de la sesión. Si estos efectos no se analizan por separado es imposible distinguir a qué es atribuible el efecto del carry-over.

Otro de los efectos que muchas veces se confunde con el carry-over es el efecto de aprendizaje por la práctica respecto de la **aplicación de las técnicas** y el efecto de **conocimiento de los programas** y del **cansancio**. Cuando se aplica una técnica de verificación, no solamente se aprende y se mejora sobre la estrategia propia de la técnica, sino sobre la actividad de verificación en sí misma, por tanto las técnicas que sean aplicadas en último lugar podrían resultar más efectivas que las aplicadas en primer lugar. Por otra parte, luego que un programa es verificado se tiene conocimiento sobre ciertos defectos encontrados en esa primera actividad. Si ese programa es verificado por segunda vez, es posible que el sujeto intente encontrar los mismos defectos que encontró previamente, pudiendo parecer más efectivas las técnicas aplicadas con posterioridad. El cansancio también puede afectar (en este caso negativamente) a las técnicas aplicadas en última instancia, si cada aplicación conlleva un tiempo considerable.

El efecto del carry-over debería ser neutralizado en caso de que sea posible. En la medicina por ejemplo, esto es posible utilizando lo que se llama período *washout* (o de lavado). En donde se deja un tiempo considerable entre la aplicación de un tratamiento y otro, a modo de que el efecto causado por la aplicación de un tratamiento desaparezca por completo. Esta técnica en pocos casos se puede utilizar en IS [KFL03]. Cuando un sujeto aprende a utilizar una técnica, no puede des-aprender a utilizarla. Quizá se pueda establecer un período considerable en el cual, al menos olvide cómo aplicarla apropiadamente. Se requiere de experimentos en IS que exploren qué tan viable es la utilización de períodos *washout* en IS. La medicina no poseía este conocimiento antes de que fuera obtenido mediante estudios empíricos.

Un último aspecto que es necesario considerar, es que existen distintos tipos de efectos como resultado de la variación entre las alternativas de los factores en un experimento: los **fijos** y los **aleatorios**. Los efectos fijos son aquellos producidos por factores en donde los niveles son finitos (los de interés para el investigador) y todos ellos están presentes en el experimento. Esto supone que cualquier variación observada se debe al error experimental [MSN08]. En el ejemplo, la Técnica es un factor de efecto fijo.

Los efectos aleatorios son provenientes de factores cuya población de niveles es potencialmente infinita y en donde únicamente una muestra al azar está presente en el experimento. Este es el caso del sujeto, en donde el "Tipo de sujeto" como factor, tiene potencialmente infinitos niveles, de los cuales (debido a la muestra escogida) sólo una parte se encuentra presente en el experimento. Como es de esperar, el tipo de sujeto tiene una influencia sobre la efectividad de las técnicas y por lo tanto genera un efecto del tipo aleatorio sobre esta variable de respuesta.

2.3. Proceso Experimental

Como se mencionó anteriormente, los experimentos son una técnica de investigación en la cual se quiere tener un mejor control del estudio y del entorno en el que éste se

lleva a cabo.

Los experimentos son apropiados para investigar distintos aspectos de la IS, como ser: confirmar teorías, explorar relaciones, evaluar la exactitud de los modelos y validar medidas. Tienen un alto costo respecto de las otras técnicas de investigación, pero a cambio ofrecen un control total de la ejecución y son de fácil replicación.

El proceso para llevar a cabo un experimento está formado por varias fases: definición, planificación, operación, análisis e interpretación y presentación.

La primer fase es la de **definición**, en donde se define el experimento en términos del problema, objetivos y metas. La siguiente fase es la **planificación**, en la cual se determina el diseño del experimento. En la fase de **operación** se ejecuta el diseño del experimento, en donde se recolectan los datos que serán analizados posteriormente en la fase de **análisis e interpretación**. En esta última fase, conceptos estadísticos son aplicados para analizar los datos. Por último, se muestran los resultados obtenidos en la fase de **presentación**. En la figura 2.3 se muestra una visión general de todo el proceso.

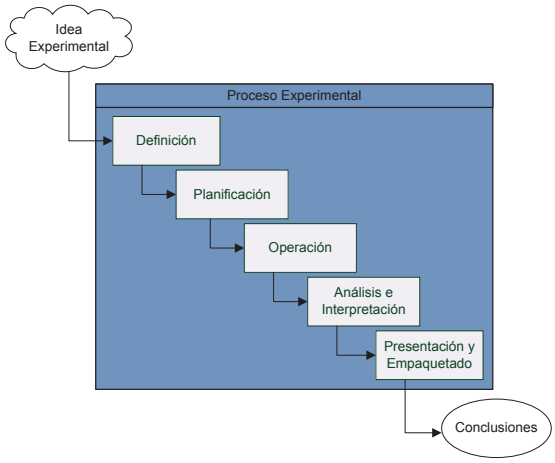

Figura 2.3: Visión general del Proceso Experimental

Si bien las fases de este proceso aparecen de forma secuencial, algunas actividades de las mismas pueden paralelizarse y/o comenzar antes que otras. Un ejemplo es la actividad de empaquetado, que puede comenzar no bien se obtienen los primeros materiales y artefactos. En experimentos de gran escala es necesario comenzar con esta actividad de forma temprana en el proceso de experimentación [TdSM+08]. Cada una de las fases que componen el proceso de experimentación se detallan a continuación.

2.3.1. Definición

En la fase de Definición se determinan las bases del experimento, que se ilustra en la figura 2.4. Para ello se debe definir **el problema que se quiere resolver**, **propósito del experimento** y **los objetivos y metas** del mismo.

Para el planteo del objetivo del experimento se debe definir *el objeto de estudio*, que

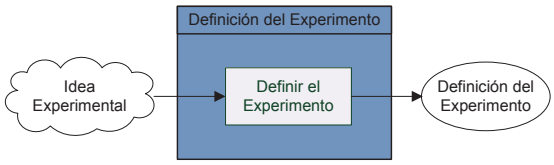

Figura 2.4: Fase de Definición del Experimento

es la entidad que va a ser estudiada en el experimento. Puede ser un producto, proceso, recurso u otro. También se debe establecer el *propósito*: la intención del experimento. Por ejemplo, evaluar diferentes técnicas de verificación.

Se debe definir además el *foco de calidad*, que refiere al efecto primario que esta bajo estudio, ejemplos son la efectividad y el costo de las técnicas de verificación. El propósito y el foco de calidad son las bases para las hipótesis del experimento.

Otro aspecto que debe estar presente es la *perspectiva*, que refiere al punto de vista con que los resultados obtenidos son interpretados. Por ejemplo, los resultados de la comparación de técnicas de verificación pueden verse desde la perspectiva de un experimentador, de un investigador o de un profesional. Un experimentador verá el estudio como una demostración de como una técnica de verificación puede ser evaluada. Un investigador puede ver el estudio como una base empírica para refinar teorías sobre la verificación de software, enfocándose en los datos que apoyan o refutan estas teorías. Un profesional puede ver el estudio como una fuente de información sobre qué técnicas de verificación deberían aplicarse en la práctica.

Junto con los aspectos mencionados se debe definir el *contexto*, que es el ambiente en el que se ejecuta el experimento. En este punto se deben definir los *sujetos* que van a llevar a cabo el experimento y los *artefactos* que son utilizados en la ejecución del mismo. Se puede caracterizar el contexto de un experimento según el número de sujetos y objetos definidos en él: un solo objeto y un solo sujeto, un solo sujeto a través de muchos objetos, un solo objeto a través de un conjunto de sujetos, o un conjunto de sujetos y un conjunto de objetos.

2.3.2. Planificación

La planificación es la fase en la que se define como se va a llevar a cabo el experimento. Esta fase consta de las etapas: selección del contexto, formulación de las hipótesis, elección de las variables, selección de los sujetos, diseño del experimento, instrumentación y evaluación de la validez, que se muestran en la figura 2.5.

La etapa de **selección del contexto** es la etapa inicial de la planificación. En esta etapa se amplía el contexto definido en la etapa de Definición, especificando claramente las características del ambiente donde ejecuta el experimento. Se define si el experimento se va a realizar en un proyecto real (en línea, *on-line*) o en un laboratorio (fuera de línea, *off-line*), características de los sujetos y si el problema es "real" (problema existente en la industria) o "de juguete". También se debe definir si el experimento es válido para un contexto específico o para un dominio general de la Ingeniería de Software.

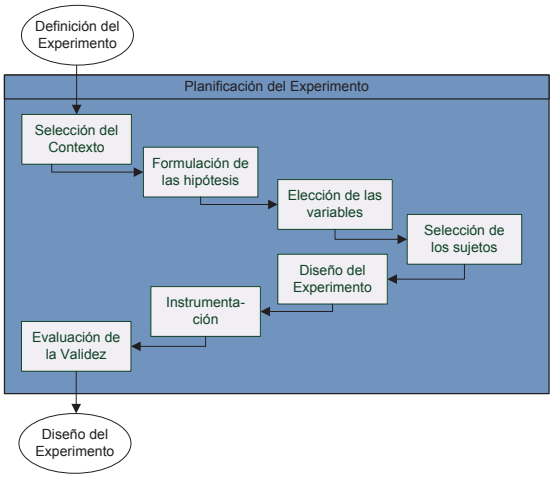

Figura 2.5: Fase de Planificación del Experimento

Una vez que los objetivos están claramente definidos se pueden transformar en una hipótesis formal. La **formulación de las hipótesis** es una fase muy importante dentro de la etapa de planificación, ya que la verificación de la misma es la base para el análisis estadístico. En esta fase se formaliza la definición del experimento en la hipótesis.

Usualmente se definen dos hipótesis, la hipótesis nula y la hipótesis alternativa. La hipótesis nula, denotada H_0, asume que no hay una diferencia significativa entre las alternativas, con respecto a las variables dependientes que se están midiendo. Establece que si hay diferencias entre las observaciones realizadas, éstas son por casualidad, no producto de la alternativa aplicada. Esta hipótesis se asume verdadera hasta que los datos demuestren lo contrario, por lo que el foco del experimento está puesto en rechazarla. Un ejemplo de hipótesis nula es: "No hay diferencia en la cantidad de defectos encontrados por las técnicas de verificación".

En cambio la hipótesis alternativa, denotada H_1, afirma que existe una diferencia significativa entre las alternativas con respecto a las variables dependientes. Establece que las diferencias encontradas son producto de la aplicación de las alternativas. Ésta es la hipótesis a probar, para esto se debe determinar que los datos obtenidos son lo suficientemente convincentes para desechar la hipótesis nula y aceptar la hipótesis alternativa. Un ejemplo de hipótesis alternativa es, si se están comparando dos técnicas de verificación, decir que una encuentra más defectos que la otra. En caso de haber más de una hipótesis alternativa se denotan secuencialmente: H_1, H_2, H_3, ..., H_n.

Una vez definida la hipótesis, se debe identificar qué variables afectan a la/s alternativa/s. Luego de identificadas las variables se debe decidir el control a ejercer sobre las mismas.

La **selección de las variables** dependientes como la de las independientes están relacionadas, por lo que en muchos casos se realizan en simultáneo. Seleccionar estas variables es una tarea muy compleja, que en ocasiones implica conocer muy bien el dominio. Es importante definir las variables independientes y analizar sus características, para así investigar y controlar los efectos que ejercen sobre las variables dependientes.

Se deben identificar las variables independientes que se pueden controlar y las que no. Además, se deben identificar las variables dependientes, mediante las cuales se mide el efecto de las alternativas. Generalmente hay sólo una variable dependiente y se deriva de la hipótesis.

Otro aspecto importante al llevar a cabo un experimento es la **selección de los sujetos**. Para poder generalizar los resultados al resto de la población, la selección debe ser una muestra representativa de la misma. Cuanto más grande es la muestra, menor es el error al generalizar los datos. Existen dos tipos de muestras que se pueden seleccionar: la probabilística, donde se conoce la probabilidad de seleccionar cada sujeto; y la no-probabilística, donde esta probabilidad es desconocida.

Luego de definir el contexto, formalizar las hipótesis, y seleccionar las variables y los sujetos, se debe **diseñar el experimento**. Es muy importante planear y diseñar cuidadosamente el experimento, para que los datos obtenidos puedan ser interpretados mediante la aplicación de métodos de análisis estadísticos.

Para comenzar a diseñar un experimento se debe elegir el diseño adecuado. Se debe planificar y diseñar el conjunto de las combinaciones de alternativas, sujetos y objetos, que conforman los experimentos unitarios. Se describe cómo estos experimentos unitarios deben ser organizados y ejecutados.

La elección del diseño del experimento afecta el análisis estadístico y viceversa, por lo que al elegir el diseño del experimento se debe tener en cuenta qué análisis estadístico es el mejor para rechazar la hipótesis nula y aceptar la alternativa.

Luego de diseñar el experimento y antes de la ejecución es necesario contar con todo lo necesario para la correcta ejecución del mismo. La **instrumentación** involucra, de ser necesario, capacitación a los sujetos, preparación de los artefactos, construcción de guías, descripción de procesos, planillas y herramientas. También implica configurar el hardware, mecanismos de consultas y experiencias piloto, entre otros. La finalidad de esta fase es proveer todo lo necesario para la realización y monitorización del experimento.

2.3.3. Evaluación de la Validez

Una pregunta fundamental antes de pasar a ejecutar el experimento es cuán válidos serían los resultados. Existen cuatro categorías de amenazas a la validez: validez de la conclusión, validez interna, validez del constructo y validez externa.

Las amenazas que afectan la **validez de la conclusión** refieren a las conclusiones estadísticas. Amenazas que afecten la capacidad de determinar si existe una relación entre la alternativa y el resultado, y si las conclusiones obtenidas al respecto son válidas. Ejemplos de estas son la elección de los métodos estadísticos, y la elección del tamaño de la muestra, entre otros.

Las amenazas que influyen en la **validez interna** son aquellas referidas a observar relaciones entre la alternativa y el resultado que sean producto de la casualidad y no del resultado de la aplicación de un factor. Esta "casualidad" es provocada por elementos desconocidos que influyen sobre los resultados sin el conocimiento de los investigadores.

Es decir, la validez interna se basa en asegurar que la alternativa en cuestión produce los resultados observados.

La **validez del constructo** indica cómo una medición se relaciona con otras de acuerdo con la teoría o hipótesis que concierne a los conceptos que se están midiendo. Un ejemplo se puede observar al momento de seleccionar los sujetos en un experimento. Si se utiliza como medida de la experiencia del sujeto el número de cursos que tiene aprobados en la universidad, no se está utilizando una buena medida de la experiencia. En cambio, una buena medida puede ser utilizar la cantidad de años de experiencia en la industria o una combinación de ambas cosas.

La **validez externa** está relacionada con la habilidad para generalizar los resultados. Se ve afectada por el diseño del experimento. Los tres riesgos principales que tiene la validez externa son: tener los participantes equivocados como sujetos, ejecutar el experimento en un ambiente erróneo y realizar el experimento en un momento que afecte los resultados.

2.3.4. Operación Experimental

Luego de diseñar y planificar el experimento, éste debe ser ejecutado para recolectar los datos que se quieren analizar. La operación del experimento consiste en tres etapas: preparación, ejecución y la validación de los datos, que se muestran en la figura 2.6.

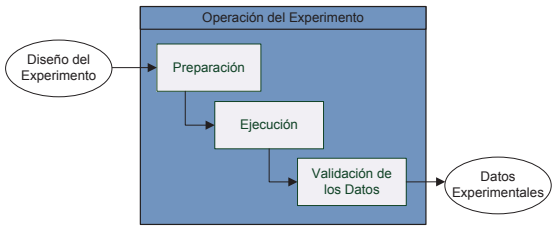

Figura 2.6: Fase de Operación del Experimento

En la etapa de preparación se seleccionan los sujetos y se preparan los artefactos a ser utilizados.

Es muy importante que los sujetos estén motivados y dispuestos a realizar las actividades que les sean asignadas, ya sea que tengan conocimiento o no de su participación en el experimento. Se debe intentar obtener consentimiento por parte de los participantes, que deben estar de acuerdo con los objetivos de la investigación. Los resultados obtenidos pueden volverse inválidos si los sujetos al momento que deciden participar no saben lo que tienen que hacer o tienen un concepto erróneo.

Es importante considerar la sensibilidad de los resultados que se obtienen de los sujetos, por ejemplo: es importante asegurar a los participantes que los resultados obtenidos sobre su rendimiento se mantienen en secreto y no se usarán para perjudicarlos en ningún sentido. Se debe tener en cuenta también los incentivos, ya que ayudan a motivar a los sujetos, pero se corre el riesgo de que participen sólo por el incentivo, lo que puede ser perjudicial para el experimento. En caso de no tener otra alternativa

que no sea engañar a los sujetos, se debe procurar explicar y revelar el engaño lo más temprano posible.

Como se vio en la instrumentación, para que los sujetos comiencen la ejecución es necesario tener prontos todos los instrumentos, formularios, herramientas, guías y otros artefactos que sean necesarios para la ejecución del experimento. Muchas veces se debe preparar un conjunto de instrumentos especial para cada sujeto y otras se utiliza el mismo conjunto de artefactos para todos los sujetos.

Existen muchas formas distintas de ejecutar los experimentos, la duración varía desde días hasta años.

Los datos pueden ser recolectados de las siguientes formas:

- Manualmente mediante el llenado de formularios por parte de los sujetos.
- Manualmente soportado por herramientas.
- Mediante entrevistas.
- Automáticamente por herramientas.

La primera es la forma más común y no requiere mucho esfuerzo por parte del experimentador. Tanto en los formularios como en los métodos soportados por herramientas no es posible identificar inconsistencias o defectos hasta que no se recolecte la información, o hasta que los sujetos los descubran. En las entrevistas, el contacto con los sujetos es mucho mayor permitiendo una mejor comunicación con ellos durante la recolección de datos. Éste método es el que requiere mas esfuerzo por parte del investigador.

Cuando se obtienen los datos, se debe chequear que fueron recolectados correctamente y que son razonables. Algunas fuentes de error son que los sujetos llenen mal sus planillas, o no recolecten los datos seriamente, lo que hace que se descarten datos. Es importante revisar que los sujetos hagan un trabajo serio y responsable y que apliquen las alternativas en el orden correcto, en otras palabras: que el experimento sea ejecutado en la forma en que fue planificado. De lo contrario los resultados podrían ser inválidos.

2.3.5. Análisis e Interpretación

Luego de que finaliza la ejecución del experimento y se cuenta con los datos recolectados, comienza la fase de análisis de los mismos conforme a los objetivos planteados.

Un aspecto importante a considerar en el análisis de los datos es la **escala de medida**. La escala de medida utilizada para recolectar los datos restringe el tipo de cálculos estadísticos que se pueden realizar. de un atributo de una entidad a un valor que de alguna forma lo caracteriza, por lo general un valor numérico, aunque hay otros. Las entidades son objetos que se observan en la realidad, por ejemplo, una técnica de verificación.

El propósito de mapear los atributos en un valor de medida es caracterizar y manipular los atributos formalmente. La medida seleccionada debe ser válida, por tanto,

no debe violar ninguna propiedad necesaria del atributo que mide, y debe ser una caracterización matemática adecuada del atributo.

El mapeo de un atributo a un valor de medida puede realizarse de varias formas. Cada tipo de mapeo posible de un atributo se conoce como escala. Los tipos más comunes de escala son:

- Escala Nominal.- Es la menos poderosa de las escalas. Solo mapea el atributo de la entidad en un nombre o símbolo. El mapeo puede verse como una clasificación de las entidades acorde al atributo. Ejemplos de escala nominal son clasificaciones, etiquetados, entre otras.
- Escala Ordinal.- La escala ordinal categoriza las entidades según un criterio de ordenación. Es más poderosa que la escala nominal. Ejemplos de criterios de ordenación son "mayor que", "mejor que" y "más complejo". Ejemplos de escala nominal son grados, complejidad del software, entre otras.
- Escala de intervalo.- La escala de intervalo se utiliza cuando la diferencia entre dos medidas es significativa. Este tipo de escala ordena los valores de la misma forma que la escala ordinal, pero existe la noción de "distancia relativa" entre dos entidades. Esta escala es más poderosa que la ordinal. Ejemplos de escala de intervalo son la temperatura medida en Celsius o Fahrenheit.
- Escala ratio (cociente de dos números).- Si existe un valor cero significativo que indica ausencia de escala y la división entre dos medidas es significativa, se puede utilizar una escala ratio. Ejemplos de escala ratio son distancia, temperatura medida en Kelvin, etc.

Después de obtener los datos es necesario interpretarlos para llegar a conclusiones válidas. La interpretación se realiza en tres etapas: caracterizar el conjunto de datos usando estadística descriptiva, reducción del conjunto de datos y realización de las pruebas de hipótesis que se ilustran en la figura 2.7.

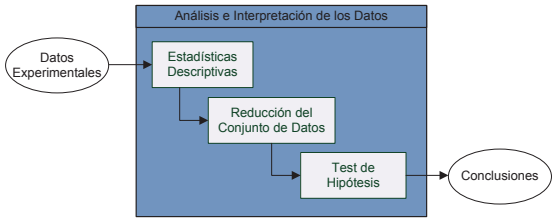

Figura 2.7: Fase de Análisis e Interpretación de los Datos del Experimento

2.3.5.1. Estadística Descriptiva

La **estadística descriptiva** se utiliza antes de la prueba de hipótesis, para entender mejor la naturaleza de los datos y para identificar datos falsos o anormales. Los aspectos principales que se examinan son: la tendencia central, la dispersión y la dependencia.

A continuación se presentan las medidas más comunes de cada uno de estos aspectos. Para ello se asume que existen $x_1 \dots x_n$ muestras.

Las **medidas de tendencia central** indican "el medio" de un conjunto de datos. Entre las medidas más comunes se encuentran: la media aritmética, la mediana y la moda.

La *media aritmética* se conoce como el promedio, y se calcula sumando todas las muestras y dividiendo el total por el número de muestras:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{2.1}$$

La *media*, denotada $\bar{x}$, resume en un valor las características de una variable teniendo en cuenta a todos los casos. Es significativa para las escalas de intervalo y ratio.

La *mediana*, denotada $\tilde{x}$, representa el valor medio de un conjunto de datos, tal que el número de muestras que son mayores que la mediana es el mismo que el número de muestras que son menores que la mediana. Se calcula ordenando las muestras en orden ascendente o descendente, y seleccionando la observación del medio. Este cálculo está bien definido si n es impar. Si n es par, la mediana se define como la media aritmética de los dos valores medios. Esta medida es significativa para las escalas ordinal, de intervalo y ratio (para valores discretos).

La *moda* representa la muestra más común. Se calcula contando el número de muestras para cada valor único y seleccionando el valor con más cantidad. La moda esta bien definida si hay solo un valor más común que los otros. Si este no es el caso, se calcula como la mediana de las muestras más comunes. La moda es significativa para las escalas nominal, ordinal, de intervalo y ratio.

La media aritmética y la mediana son iguales si la distribución de las muestras es simétrica. Si la distribución es simétrica y tiene un único valor máximo, las tres medidas son iguales.

Las medidas de tendencia central no proveen información sobre la dispersión del conjunto de datos. Cuanto mayor es la dispersión, más variables son las muestras, cuanto menor es la dispersión, más homogéneas a la media son las muestras.

Las **medidas de dispersión** miden el nivel de desviación de la tendencia central, o sea, que tan diseminados o concentrados están los datos respecto al valor central. Entre las principales medidas de dispersión están: la varianza, la desviación estándar, el rango y el coeficiente de variación.

La *varianza* (s^2) que presenta una distribución respecto de su media se calcula como la media de las desviaciones de las muestras respecto a la media aritmética. Dado que la suma de las desviaciones es siempre cero, se toman las desviaciones al cuadrado:

$$s^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2 \tag{2.2}$$

Se divide por $n-1$ y no por n, porque una de las medidas es referente al punto

central y queda sin sentido (cero) en el denominador (cuando la distancia es cero). La varianza es significativa para las escalas de intervalo y ratio.

La *desviación estándar*, denotada s, se de

ne como la raíz cuadrada de la varianza:

$$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{2.3}$$

A menudo esta medida se prefiere sobre la varianza porque tiene las mismas dimensiones (unidad de medida) que los valores de las muestras. En cambio, la varianza se mide en unidades cuadráticas. La desviación estándar es significativa para las escalas de intervalo y ratio.

La dispersión también se puede expresar como un porcentaje de la media. Este valor se llama *coeficiente de variación*, y se calcula como:

$$100 \cdot \frac{s}{\bar{x}} \tag{2.4}$$

Esta medida no tiene dimensión y es significativa para la escala ratio. Permite comparar la dispersión o variabilidad de dos o más grupos.

El **rango** de un conjunto de datos es la distancia entre el valor máximo y el mínimo:

$$range = x_{max} - x_{min} \tag{2.5}$$

Es una medida significativa para las escalas de intervalo y ratio. Cuando el conjunto de datos consiste en muestras relacionadas de a pares (xi; yi) de dos variables, X e Y, puede ser interesante examinar la dependencia entre estas variables. Las principales medidas de dependencia son: regresión lineal, covarianza y el coeficiente de correlación lineal.

2.3.5.2. Reducción del Conjunto de Datos

Para las pruebas de hipótesis se utilizan métodos estadísticos. El resultado de aplicar estos métodos depende de la calidad de los datos. Si los datos no representan lo que se cree, las conclusiones que se derivan de los resultados de los métodos son incorrectas. Errores en el conjunto de datos pueden ocurrir por un error sistemático, o por lo que se conoce en estadística con el nombre de outlier. Un outlier es un dato mucho más grande o mucho más chico de lo que se puede esperar observando el resto de los datos.

Las estadísticas descriptivas se ven fuertemente influenciadas por aquellas observaciones que su valor dista significativamente del resto de los valores recolectados. Estas observaciones llevan el nombre de *outliers*.

Los *outliers* influencian las medidas de dispersión, aumentando la variabilidad de lo que se está midiendo. En algunos casos se realiza un análisis acerca de estos valores que

difieren mucho de la media y se decide quitarlos de los datos a analizar porque no son representativos de la población, ya que fueron causados por algún tipo de anomalía: errores de medición, variaciones no deseadas en las características de los sujetos, entre otras.

Quitar *outliers* requiere de un análisis pormenorizado, por quitar outliers se demoró en detectar el agujero de la capa de ozono. [4]

Una vez identificado un outlier se debe identificar su origen para decidir qué hacer con él. Si se debe a un evento raro o extraño que no volverá a ocurrir, el punto puede ser excluido. Si se debe a un evento extraño que puede volver a ocurrir, no es aconsejable excluir el valor del análisis, pues tiene información relevante. Si se debe a una variable que no fue considerada, debería ser considerado para basar los cálculos y modelos también en esta variable.

2.3.5.3. Pruebas de Hipótesis

El objetivo de la **prueba de hipótesis** es ver si es posible rechazar cierta hipótesis nula H_0. Si la hipótesis nula no es rechazada, no se puede decir nada sobre los resultados. En cambio, si es rechazada, se puede declarar que la hipótesis es falsa con una significancia dada (α). Este nivel de significancia también es denominado nivel de riesgo o probabilidad de error, ya que se corre el riesgo de rechazar la hipótesis nula cuando en realidad es verdadera. Este nivel está bajo el control del experimentador.

Para probar H_0 se define una unidad de prueba t y un área crítica C, la cual es parte del área sobre la que varía t. A partir de estas definiciones se formula la prueba de significancia de la siguiente forma:

- Si $t \in C$, rechazar H_0
- Si $t \notin C$, no rechazar H_0

Por ejemplo, un experimentador observa la cantidad de defectos detectados por LOC de una técnica de verificación desconocida bajo determinadas condiciones, y quiere probar que no es la técnica B, de la cual sabe que en las mismas condiciones (programa, verificador, etc.) detecta 1 defecto cada 20 LOC. El experimentador sabe que también pueden haber otras técnicas que detecten 1 defecto cada 20 LOC. A partir de esto se define la hipótesis nula: "H_0: La técnica observada es la B". En este ejemplo, la unidad de prueba t es cada cuantos LOC se detecta un defecto y el área crítica es $C = \{1, 2, 3, \ldots, 19, 21, 22, \ldots\}$. La prueba de significancia es: si $t \leq 19$ o $t \geq 21$, rechazar H_0, de lo contrario no rechazar H_0.

Si se observa que $t = 20$, la hipótesis no puede ser rechazada ni se pueden derivar conclusiones, pues pueden haber otras técnicas que detecten un defecto cada 20 LOC.

[4] En 1985 los científicos británicos anunciaron un agujero en la capa de ozono sobre el polo sur. El reporte fue descartado ya que observaciones más completas, obtenidas por instrumentos satelitales, no mostraban nada inusual. Luego, un análisis más exhaustivo reveló que las lecturas de ozono en el polo sur eran tan bajas que el programa que las analizaba las había suprimido automáticamente como outliers en forma equivocada.

El área crítica, C, puede tener distintas formas, lo más común es que tenga forma de intervalo, por ejemplo: $t \leq a$ o $t \geq b$. Si C consiste en uno de estos intervalos es unilateral. Si consiste de dos intervalos ($t \leq a$, $t \geq b$, donde $a < b$), es bilateral.

Hay varios métodos estadísticos, de aquí en adelante denotados *tests*, que pueden utilizarse para evaluar los resultados de un experimento, más específicamente para determinar si se rechaza la hipótesis nula. Cuando se lleva a cabo un *test* es posible calcular el menor valor de significancia posible (denotado p-valor) con el cual es posible rechazar la hipótesis nula. Se rechaza la hipótesis nula si el p-valor asociado al resultado observado es menor o igual que el nivel de significancia establecido.

Las siguientes son tres probabilidades importantes para la prueba de hipótesis:

- $\alpha = P(\text{cometer el error tipo I}) = P(\text{rechazar } H_0 | H_0 \text{ es verdadera})$. Es la probabilidad de rechazar H_0 cuando es verdadera.
- $\beta = P(\text{cometer el error tipo II}) = P(\text{no rechazar } H_0 | H_0 \text{ es falsa})$. Es la probabilidad de no rechazar H_0 cuando es falsa.
- $\text{Poder} = 1 - \beta = P(\text{rechazar } H_0 | H_0 \text{ es falsa})$. El poder de prueba es la probabilidad de rechazar H_0 cuando es falsa.

El experimentador debería elegir un test con un poder de prueba tan alto como sea posible. Hay varios factores que afectan el poder de un test. Primero, el test en sí mismo puede ser más o menos efectivo. Segundo, la cantidad de muestras: mayor cantidad de muestras equivale a un poder de prueba más alto. Otro aspecto es la selección de una hipótesis alternativa unilateral o bilateral. Una hipótesis unilateral da un poder mayor que una bilateral.

La probabilidad de cometer un error tipo I se puede controlar y reducir. Si la probabilidad es muy pequeña, sólo se rechazará la hipótesis nula si se obtiene evidencia muy contundente en contra de esta hipótesis. La probabilidad máxima de cometer un error tipo I se conoce como la significancia de la prueba (α).

Los valores de uso más común para la significancia de una prueba son 0.01, 0.05 y 0.10. La significancia es en ocasiones presentada como un porcentaje, tal como 1 %, 5 % o 10 %. Esto quiere decir que el experimentador está dispuesto a permitir una probabilidad de 0.01, 0.05, o 0.10 de rechazar la hipótesis nula cuando es cierta, o sea, de cometer un error tipo I.

El valor de la significancia es seleccionado antes de comenzar a hacer el experimento en una de varias formas.

El valor de α puede estar establecido en el área de investigación, por ejemplo: se puede obtener de artículos que se publican en revistas científicas. Otra forma de seleccionarlo es que sencillamente sea impuesto por la persona o compañía para la cual se trabaja. También puede ser seleccionado tomando en cuenta el costo de cometer un error tipo I. Mientras más alto el costo, más pequeña debe ser la probabilidad α de cometer un error tipo I. Tradicionalmente, el valor de α es seteado en 0.05, pero éste podría variar dependiendo del tamaño de la muestra y del tamaño del efecto que se quiere lograr [DKSb06].

Existen dos tipos de tests: paramétricos y no paramétricos. Los **tests paramétricos** están basados en un modelo que involucra una distribución específica. En la mayoría de los casos, se asume que algunos de los parámetros involucrados en un test paramétrico están normalmente distribuidos y presentan homocedasticidad[5]. Los tests paramétricos también requieren que los parámetros puedan ser medidos al menos en una *escala de intervalo*. Si los parámetros no pueden medirse en al menos una escala de intervalo, generalmente no se puede utilizar un test paramétrico. En este caso hay un amplio rango de tests no paramétricos disponible.

Los **tests no paramétricos** no asumen lo mismo respecto a la distribución de los parámetros, son más generales que los paramétricos. Un test no paramétrico se puede utilizar en vez de un test paramétrico, pero el caso inverso no siempre puede darse.

En la elección entre un test paramétrico y un test no paramétrico hay dos aspectos a considerar:

- Aplicabilidad.- Es importante que las suposiciones en cuanto a las distribuciones de parámetros y las que conciernen a las escalas sean realistas.
- Poder.- El poder de los tests paramétricos es generalmente mayor que el de los tests no paramétricos. Por lo tanto, los test paramétricos requieren menos datos (experimentos más pequeños), que los tests no paramétricos, siempre que sean aplicables.

Aunque es un riesgo utilizar tests paramétricos cuando no se cuenta con las condiciones requeridas, en algunos casos vale la pena tomar el riesgo. Algunas simulaciones han mostrado que los tests paramétricos son bastante robustos a las desviaciones de las pre-condiciones (escala de intervalo), mientras las desviaciones no sean demasiado grandes.

En el caso de las pruebas paramétricas, se exige que la distribución de la muestra se aproxime a una normal o que la misma presente homogeneidad de varianzas (depende del tipo de prueba, el supuesto que se aplica). Para poder utilizar aproximación normal se requiere un tamaño mínimo de la muestra, dependiendo del $p(value)$ que se requiera [Spi91]. En el cuadro 2.2 se muestran los tamaños mínimos de muestra para los distintos $p(value)$.

p(value)	**Tamaño mínimo de muestra**
0.05	n = 30
0.04 ó 0.06	n = 50
0.03 ó 0.07	n = 80
0.02 ó 0.08	n = 200
0.01 ó 0.09	n = 600

Cuadro 2.2: Estadísticas descriptivas de la Efectividad

[5]Un conjunto de datos presenta homocedasticidad cuando la varianza del error de la variable endógena se mantiene a lo largo de las observaciones. En otras palabras, la varianza de los errores se mantiene constante.

Algunos de los test paramétricos más usados en experimentos de Ingeniería de Software son:

- ANOVA (*ANalysis Of VAriance*) [WRH99].
- ANOM (*ANalysis Of Means*) [NCC03].

Ambos tests (ANOVA y ANOM), pueden utilizarse para diseños de un solo factor con múltiples alternativas. En ambos test la hipótesis nula refiere a la igualdad de las medias (como es habitual en los test paramétricos):

$H_0 : \bar{x}_1 = \bar{x}_2 = \ldots = \bar{x}_I$

En ANOVA, la variación en la respuesta se divide en la variación entre los diferentes niveles del factor (los diferentes tratamientos) y la variación entre individuos dentro de cada nivel. El objetivo principal del ANOVA es contrastar si existen diferencias entre las diferentes medias de los niveles de las variables (factores).

En el caso de ANOM, este test no solamente responde a la pregunta de si hay o no diferencias entre las alternativas, sino que cuando hay diferencias, también dice cuáles alternativas son mejores y cuáles peores.

Los test no paramétricos más usado son:

- Kruskal Wallis.
- Mann-Whitney.

En el caso de los test no paramétricos, la hipótesis nula refiere a la igualdad de las medianas:

$H_0 : \tilde{x}_1 = \tilde{x}_2 = \ldots = \tilde{x}_I$

Rechazar H_0 significa que existe evidencia estadística como para afirmar de que hay diferencias entre las alternativas. En el caso de que hubiera más de dos alternativas, para conocer cuál es la alternativa que difiere es necesario comparar las alternativas de a dos.

En el caso de Kruskal Wallis, a pesar de no requerir una distribución normal para las muestras, sus resultados se pueden ver afectados por lo que se le llama "heterocedasticidad" de los datos. Cuando una muestra presenta datos heterocedásticos (no presentan homocedasticidad) el test de Kruskal Wallis podría dar un resultado no significativo (no rechazando H_0), aunque haya una diferencia real entre las muestras (debería rechazar H_0).

Para probar la homocedasticidad de los datos se suele utilizar el test de Levene. Las hipótesis del test de Levene son:

- $H_0 : \sigma_1 = \sigma_2 = \ldots = \sigma_k$ donde σ_a es la varianza de la muestra a.
- $H1 : \sigma_i \neq \sigma_j = \ldots = \sigma_k$ para al menos un par de muestras (i, j), donde σ_a es la varianza de la muestra a.

Para poder aplicar ANOVA, y en algunos casos Kruskal-Wallis, es necesario que el test de Levene no sea significativo (no se rechaza H_0), o sea, que las varianzas de las muestras sean similares o iguales. Esto prueba la homocedasticidad de los datos (para pruebas no paramétricas) o la homogeneidad de varianzas (para aquellas paramétricas).

Una vez que se prueba que al menos dos de las k muestras provienen de poblaciones distintas (datos heterocedásticos) se puede aplicar, entre otros, el test de Mann-Whitney para comparar las muestras dos a dos.

Si se presume que una alternativa puede ser mejor o peor que el resto, esto quiere decir que hay un "ordenamiento" entre ellas, lo aconsejable es realizar un test de ordenamiento. Algunos test de ordenamiento son:

- Jonckheere-Terpstra Test. [Mar04]
- Test para alternativas ordenadas L. [Mar04]

Para los test de ordenamiento, las hipótesis que se plantean son las siguientes:

$H_0 : \tilde{x}_1 = \tilde{x}_2 = \ldots = \tilde{x}_I$

$H_1 : \tilde{x}_1 \leq \tilde{x}_2 \leq \ldots \leq \tilde{x}_I$ (con al menos una desigualdad estricta)

Si bien los tipos de análisis para pruebas de hipótesis más utilizados son los mencionados anteriormente, es necesario que el modelo matemático que utiliza la prueba se adecúe al diseño experimental. Para los experimentos con diseños cross-over es necesario que las técnicas de análisis contemplen las diferentes variables que los afectan por ser diseños de medidas repetidas: perído, secuencia y carry-over, además del efecto aleatorio producido por el sujeto. Para este tipo de diseño es más adecuado utilizar el **Análisis de Componentes de la Varianza aplicado a modelos mixtos**, también conocido como **modelos lineales mixtos** (*linear mixed models*).

A diferencia del ANOVA y ANOM, el modelo lineal mixto permite especifiar que haya factores de tratamientos especificados explícitamente como variables de medidas repetidas. El sujeto es especificado como factor aleatorio y la secuencia, período y carry-over son tenidos en cuenta.

El análisis del modelo lineal mixto es una ampliación del **modelo lineal general** [NS11] de manera que los datos puedan presentar variabilidad correlacionada y no constante [MSN08, NS11]. Proporciona, por tanto, la flexibilidad necesaria para modelar no sólo las medias sino también las varianzas y covarianzas de los datos. El Análisis de modelos lineales mixtos es asimismo una herramienta flexible para ajustar otros modelos que puedan ser formulados como modelos lineales mixtos. Dichos modelos incluyen los modelos multinivel, los modelos lineales jerárquicos y los modelos con coeficientes aleatorios.

La literatura sobre los tipos de modelos y los análisis que se pueden realizar para cada uno de ellos es muy extensa y no está dentro del alcance de este trabajo. Sí se describe con detalle el modelo y tipo de análisis que se utiliza para el experimento foco de este trabajo en la sección 4.4.

2.3.6. Presentación y Empaquetado

En la presentación y el empaquetado de un experimento es esencial no olvidar aspectos e información necesaria para que otros puedan replicar o tomar ventaja del experimento y del conocimiento ganado durante su ejecución.

El esquema de reporte de un experimento generalmente cuenta con los siguientes títulos: Introducción, Definición del Problema, Planificación del Experimento, Operación del Experimento, Análisis de Datos, Interpretación de los Resultados, Discusión y Conclusiones, y Apéndice.

En la *Introducción* se realiza una introducción al área y los objetivos de la investigación. En la *Definición del Problema* se describe en mayor profundidad el trasfondo de la investigación, incluyendo las razones para realizarla. En la *Planificación del Experimento* se detalla el contexto del experimento incluyendo las hipótesis, que se derivan de la definición del problema, las variables que se deben medir (tanto independientes como dependientes), la estrategia de medida y análisis de datos, los sujetos que participaran de la investigación y las amenazas a la validez.

En la *Operación del Experimento* se describe como preparar la ejecución del mismo, incluyendo aspectos que permitan facilitar la replicación y descripciones que indiquen cómo se llevaron a cabo las actividades. Debe incluirse la preparación de los sujetos, cómo se recolectaron los datos y cómo se realizó la ejecución.

En el *Análisis de Datos* se describen los cálculos y los modelos de análisis específicos utilizados. Se debe incluir información, como por ejemplo, tamaño de la muestra, niveles de significancia y métodos estadísticos utilizados, para que el lector conozca los prerequisitos para el análisis. En la *Interpretación de los Resultados* se rechaza la hipótesis nula o se concluye que no puede ser rechazada. Aquí se resume cómo utilizar los datos obtenidos en el experimento. La interpretación debe realizarse haciendo referencia a la validez. También se deben describir los factores que puedan tener un impacto sobre los resultados.

Finalmente, en *Discusión y Conclusiones* se presentan las conclusiones y los hallazgos como un resumen de todo el experimento, junto con los resultados, problemas y desviaciones respecto al plan. También se incluyen ideas sobre trabajos a futuro. Los resultados deberían ser comparados con los obtenidos por trabajos anteriores, de manera de identificar similitudes y diferencias. La información que no es vital para la presentación se incluye en el Apéndice. Esto puede ser, por ejemplo, los datos recabados y más información sobre sujetos y objetos. Si la intención es generar un paquete de laboratorio, el material utilizado en el experimento puede ser provisto en el apéndice.

Capítulo 3

Teoría y Fundamentos sobre V&V en IS

Este capítulo profundiza en las técnicas de Verificación y Validación en Ingeniería de Software y los tipos de defectos que se usan en la serie de experimentos. Sobre las técnicas se explica en detalle la teoría y estrategia de aplicación de cada técnica y se dan ejemplos de ejecución de las mismas. Sobre los tipos de defectos se detalla la clasificación utilizada y se explica los motivos de elección de la misma

3.1. Verificación y Validación

La verificación es el conjunto de actividades que busca comprobar que un producto de software cumple correctamente con su especificación.

Tiene como objetivo descubrir defectos y evaluar la calidad de los productos. El software falla cuando no hace lo requerido o hace algo que no debería. La falla es una propiedad de un sistema en ejecución, las mismas se manifiestan debido a que el código del programa contiene un defecto.

Algunas razones por las que el software puede fallar son:

- Las especificaciones no traducen exactamente lo que el cliente quiere.
- Faltas en el diseño.
- Defectos en el código.

La validación es un conjunto de diferentes actividades que aseguran que el software construido se corresponde con los requisitos del cliente.

Verificación ¿Estamos construyendo el producto correctamente?

Validación ¿Estamos construyendo el producto correcto?

La verificación y validación, comúnmente denotada V&V, abarcan una lista de actividades para el aseguramiento de la calidad del software.

El desarrollo de sistemas de software implica una serie de actividades de producción, en las que las posibilidades de que no aparezca una falta humana es casi imposible. Debido a la incapacidad humana de trabajar y comunicarse de forma perfecta, los defectos pueden darse en las diversas etapas del proceso, por lo que el desarrollo de software ha de ir acompañado de actividades que garanticen su calidad.

La prueba de software es un elemento crítico para la garantía de calidad y representa una revisión final de las especificaciones, el diseño y la codificación. Esta requiere que se descarten ideas preconcebidas sobre la corrección del software que se acaba de desarrollar y se supere cualquier conflicto de intereses que aparecen cuando se descubren defectos.

Según Dijkstra: "Las pruebas sólo pueden demostrar la presencia de errores, no su ausencia". Debido a esto, no es posible asegurar que las pruebas de un sistema eliminan la presencia de errores en tiempo de ejecución. Generalmente las pruebas se realizan hasta cumplir con un criterio de aceptación previamente definido [Dij72].

Para comprender mejor los próximos capítulos se incluyen las siguientes definiciones:

- **Prueba**: Es el proceso de ejecutar un programa con el fin de encontrar fallas.
- **Caso de prueba**: Conjunto de datos de entrada, condiciones de ejecución y resultado esperado.
- **Caso de prueba**: Conjunto de instrucciones que se llevará a cabo en el sistema en que se ejecutan los casos de pruebas.

3.1.1. Proceso de Verificación y Validación

En la figura 3.1 se muestra el proceso de V&V. El mismo está compuesto por diferentes tipos de pruebas según la etapa, que se detallan a continuación.

- *Prueba Unitaria*
  Las pruebas unitarias centran el proceso de verificación en la menor unidad de diseño del software. Una unidad puede ser un único método, tanto en código como en diseño.
  En la sección 3.2 referente a pruebas unitarias se detalla en profundidad este tipo de pruebas.
- *Prueba de Integración*
  Las pruebas de integración verifican que los componentes trabajan juntos como un sistema integrado. Son necesarias ya que al integrar cada una de las unidades individuales pueden surgir interacciones no previstas que generan fallas. Normalmente estas pruebas las realiza el equipo de desarrollo, ya que es necesario el conocimiento de las interfaces y de las funciones en general.

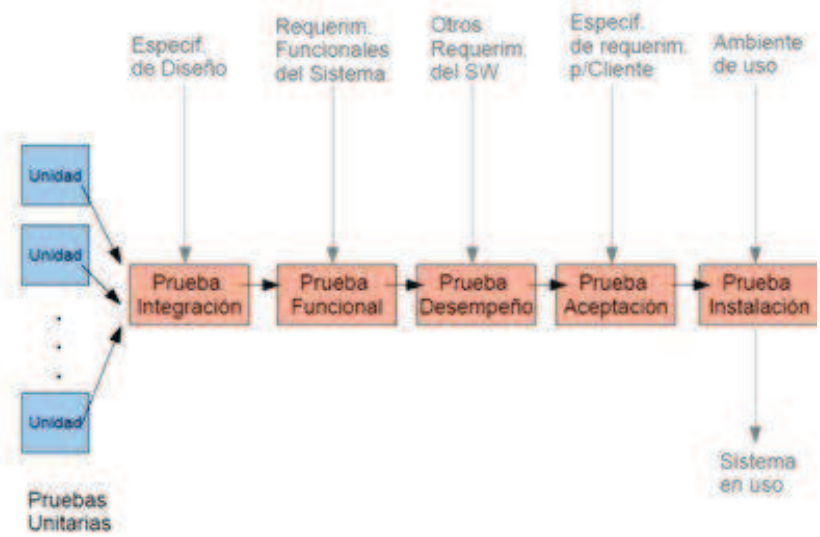

Figura 3.1: Proceso V&V

- *Prueba Funcional*
  Las pruebas funcionales se ejecutan tras la culminación de las pruebas de integración. Las mismas se concentran en las acciones visibles para el usuario y determinan si el sistema integrado cumple las funciones de acuerdo a los requerimientos.
  Un caso de uso es un diagrama que se utiliza para ilustrar los requerimientos de un sistema. En éste se especifica la funcionalidad y el comportamiento de un sistema mediante su interacción con los usuarios y/o otros sistemas. Las pruebas funcionales se realizan a partir de los casos de uso.

- *Prueba de Desempeño*
  Las pruebas de desempeño determinan si el sistema integrado en el ambiente objetivo cumple los requerimientos de: tiempo de respuesta, capacidad de proceso, volumen, etc.

- *Prueba de Aceptación*
  La prueba de aceptación es realizada bajo supervisión del cliente. Sirve para verificar que el sistema cumple con los requerimientos y necesidades del cliente.

- *Prueba de Instalación*
  Las pruebas de instalación son para comprobar que el sistema instalado en el ambiente de trabajo del cliente funciona correctamente.

3.2. Prueba Unitaria

Las pruebas unitarias tienen como principal objetivo asegurar el correcto funcionamiento de una componente individual.

Existen diferentes items que pueden ser probados en las pruebas unitarias:

- Funciones individuales o métodos dentro de un objeto.
- Clases de objetos que tienen varios atributos y métodos.
- Conjunto de componentes formados por varios objetos diferentes o funciones. Estos componentes tienen una interfaz definida que se utiliza para acceder a su funcionalidad.
  Este concepto es presentado por Robert V. Binder como *small class cluster*. Un *small class cluster* incluye varias clases que están fuertemente acopladas y las pruebas de las clases que lo constituyen no es práctico realizarlas de forma aislada. El *head* del *small class cluster* es una única clase que utiliza todas las capacidades de la clases que lo componen y los componentes no se usan fuera del clúster [Bin99].

Se busca lograr que los distintos componentes tengan una mejor calidad y menos defectos para que de esta forma las pruebas de más alto nivel se concentren en aspectos más interesantes y profundos.

Al ejecutar una prueba unitaria se busca detectar los defectos que se introducen al programar. Cuanto antes se detecte un defecto menor será el costo de su resolución.

Es común que las pruebas unitarias sean realizadas por la misma persona que implementa la componente o por el equipo de desarrollo, ya que es importante contar con el conocimiento detallado de la misma.

3.3. Clasificación de Técnicas de Verificación Unitaria

En esta sección se presenta una posible clasificación de las técnicas de verificación unitaria y se detallan las características de estas.

Dentro de la verificación unitaria se distinguen dos grandes niveles: las técnicas de verificación estática y las técnicas de verificación dinámica.

Mientras que las técnicas estáticas están enfocadas en analizar el producto para deducir su correcto funcionamiento, las dinámicas están enfocadas en la realización de pruebas teniendo como objetivo experimentar con el comportamiento de un producto para ver si éste actúa como es esperado.

En la figura 3.2 se muestra una clasificación jerárquica de muchas de las técnicas de verificación unitaria donde se observan estos dos grandes niveles [SLS06].

3.3.0.1. Técnicas de Verificación Unitaria Estática

La verificación unitaria estática consiste en analizar un sistema de software y su especificación en busca de defectos, pero sin ejecutar el código del mismo. Estas técnicas buscan descubrir defectos tempranamente en el proceso del software.

Las técnicas de verificación unitaria estática son realizadas previo a la ejecución de las pruebas unitarias (verificación dinámica) y sirven para verificar el código fuente o

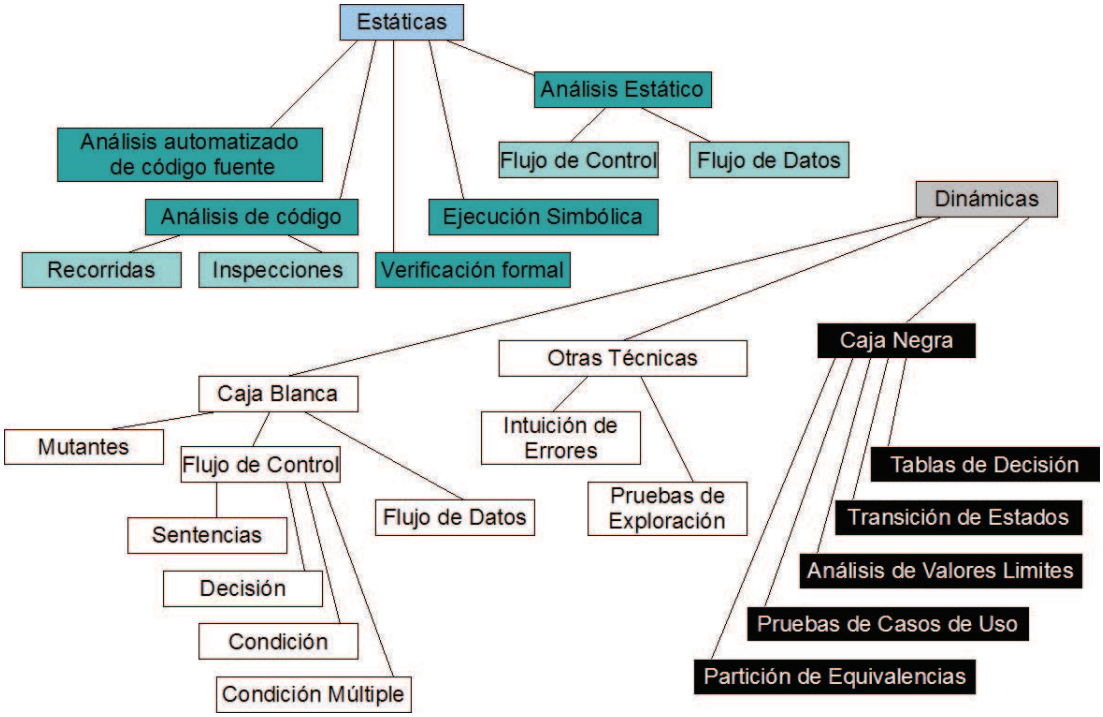

Figura 3.2: Clasificación de técnicas de verificación unitaria

cualquier documento producido como parte del proceso de software, cómo documentos de diseño y requerimientos.

Estas técnicas se dividen en las categorías: análisis de código, análisis automatizado de código fuente, verificación formal, análisis estático y ejecución simbólica.

3.3.0.2. Técnicas de Verificación Unitaria Dinámica

Las técnicas de verificación unitaria dinámica son aquellas que ejecutan el código a probar. Estas pruebas experimentan con el comportamiento del software para ver si el mismo actúa como es esperado.

Existen diferentes enfoques a la hora de probar el código, según la visión que se tome del mismo. Estas técnicas se pueden clasificar en técnicas de caja negra, de caja blanca y otras técnicas.

- ***Técnicas de caja negra:***
  las técnicas de caja negra también son conocidas como técnicas basadas en la especificación, ya que se basan en ésta para definir el conjunto de casos de prueba. Al utilizar el enfoque de caja negra la estructura interna del código y su diseño se toma como si no fuera conocido.

  Estas técnicas son apropiadas para todos los niveles de la verificación dónde exista especificación, desde la verificación unitaria hasta las pruebas de aceptación. El

interés radica en lo que el software debe hacer y no en el código escrito para ello. Es por esto que puede suceder que aplicando esta técnica queden porciones enteras del código sin ejecutar.

Dentro de las técnicas de caja negra se distinguen las familias de: partición en clases de equivalencia, análisis de valores límites, tablas de decisión, transición de estados y basados en casos de uso.

- ***Técnicas de caja blanca:*** las técnicas de caja blanca son también conocidas como técnicas basadas en la estructura, ya que los casos de prueba se determinan basándose principalmente en el código del software que se va a probar.

  Al momento de definir los casos de prueba se tiene en cuenta las características de la implementación. Notar que a diferencia de las técnicas de caja negra, las de caja blanca requieren que el verificador conozca internamente cómo se encuentra implementado el software.

  Estas técnicas se pueden dividir a su vez en tres grandes familias de técnicas (o criterios de cubrimiento): basadas en flujo de control y las basadas en flujo de datos.

  A continuación se detallan algunas de las técnicas más conocidas:

  - **Flujo de control:** las técnicas basadas en el flujo de control consideran el cubrimiento de código, por lo cual, ayudan a que el implementador conozca qué parte del código bajo prueba se está considerando en los casos de prueba y en qué circunstancias. Es por esto que la aplicación de este tipo de técnicas es de gran ayuda para la mejora de las pruebas.

    Ejemplos de este tipo de técnicas son: cubrimiento de sentencias, cubrimiento de decisión y cubrimiento de trayectorias linealmente independientes.
  - **Flujo de datos:** Los métodos basados en el flujo de datos son también conocidos como *Data Flow Testing* (DTF). Son técnicas que observan cómo los distintos valores asociados a variables pueden afectar la ejecución de un programa.

    Los casos de prueba se generan basándose en el conocimiento de las operaciones que se realizan sobre las variables en el código bajo prueba. Se presta atención a cómo se define una variable y cómo se utiliza la misma a lo largo del flujo del control.

- ***Otras técnicas:*** existen otras técnicas que son usadas para complementar las técnicas de caja blanca y caja negra, así como para cuando no existe ninguna especificación o cuando la especificación no es adecuada. Dentro de estas se encuentran por ejemplo las técnicas basadas en la experiencia y el *testing* exploratorio.

  Las técnicas basadas en la experiencia, se basan principalmente en el conocimiento, la habilidad y la experiencia de los verificadores. Por ejemplo cuando el verificador tiene experiencia en sistemas similares es algo relevante ya que podría intuir dónde el sistema puede fallar.

  El *testing* exploratorio ocurre cuando se diseñan y ejecutan las pruebas al mismo tiempo. Esta estrategia es útil para obtener resultados rápidamente y complementa las pruebas planificadas. El éxito de este tipo de pruebas depende de las

habilidades y preparación de quienes las realizan. Quién realiza este tipo de pruebas debe tener, entre otras, las siguientes habilidades: ser curioso, observador, capaz de generar ideas y ver las posibilidades, tener pensamiento crítico y experiencia.

3.4. Tipos de Defectos

De forma simplificada, la efectividad puede calcularse midiendo la cantidad de defectos detectados por la técnica sobre la cantidad de defectos totales. Para este experimento, el número total de defectos es conocido, ya que los mismos se siembran sobre programas ya verificados previamente. Como se mencionó anteriormente, determinados tipos de técnicas son más sensibles a la detección de determinado tipo de defectos, por lo cual el tipo de defectos que contienen los programas influye en el cálculo de la efectividad. Para que esta influencia sea controlada, los defectos se siembran de acuerdo a una clasificación determinada.

Existen muchos tipos de taxonomías (también llamadas clasificaciones) de defectos las cuales son usadas tanto por la academia como por la industria de software. Cada una se basa en distintos aspectos o características de los defectos para clasificarlos, entre las cuales podemos encontrar: fase del proceso de desarrollo en la que se introdujo el defecto, causa del mismo, impacto en el software o en el cliente, tipo de falla que genera, entre otras. Algunas de las más conocidas son:

- **Taxonomía de Hewlett-Packard** [Gra92]: posee una estructura semi-ortogonal y está compuesta por tres atributos a clasificar: origen, tipo y modo. El origen del defecto indica la actividad o proceso en donde se inyecta el mismo, por ejemplo: diseño, código, documentación, entre otros. El tipo de defecto define el área que es responsable el defecto (los valores de éste dependen del origen previamente seleccionado), por ejemplo: requerimientos, funcionalidad, interfaz HW, interfaz SW, definición de datos, entre otros. El modo del defecto refiere a la causa que lleva al defecto, por ejemplo: omiso, erróneo, confuso, cambiado, entre otras.

- **Taxonomía de Kaner, Falk y Nguyen** [KFN99]: taxonomía compuesta por un único atributo a clasificar, que es el tipo de defecto. El conjunto de valores posibles para dicho atributo está formado por 13 grandes categorías, en donde cada categoría contiene un conjunto de subcategorías asociadas. Dicho de otro modo, posee una taxonomía con una estructura jerárquica de dos niveles.

- **Taxonomía de Binder** [Bin99]: se basa en el paradigma de programación orientada a objetos, en donde gran parte de los defectos se deben a problemas en la aplicación de los conceptos de herencia, polimorfismo, secuencia de mensajes y transición de estados. Está compuesta por los atributos “Origen” y “Tipo” y presenta una estructura de árbol ya que los valores posibles para el atributo Tipo dependen del valor elegido para el atributo Origen. Además es jerárquica, pues el atributo tipo posee un conjunto de valores posibles que conforman una jerarquía de dos niveles.

- **IEEE Standard Classification for Software Anomalies** [iee10]: está pensado para organizaciones que quieren implementar o ampliar una taxonomía o para aquellas que deseen expandir y mejorar sus mecanismos de registro y seguimiento de defectos con metodologías ya probadas y analizadas. El standard propone una taxonomía de atributos ortogonales cuya clasificación se realiza en cuatro etapas: Reconocimiento, Investigación, Acción y Disposición. En cada una de las etapas, se registra lo encontrado/investigado/decidido/realizado, se clasifican los atributos correspondientes y se identifica su impacto. Si bien sus atributos son ortogonales, dentro de cada atributo existe una relación jerárquica dentro del conjunto de valores posibles, por lo que la estructura de la taxonomía es ortogonal y jerárquica.
- **Taxonomía ODC propuesta por IBM** [Lyu96]: la *Orthogonal Defect Classification*, es una taxonomía de defectos desarrollada por IBM en 1996. La misma brinda un puente entre las dos metodologías de análisis de defectos existentes previas a 1996: Statistical Defect Models (modelos de análisis cuantitativos) y Root Cause Analysis (modelos de análisis cualitativos), mediante el desarrollo de un sistema de medición basado en semántica. Uno de los objetivos de ODC es brindar al equipo de desarrollo una retroalimentación constante durante el transcurso del proyecto. Además, ODC es utilizada para mejorar el proceso de desarrollo mediante la reducción de la cantidad de defectos con el avance del proyecto. ODC propone dos pasos de recolección de información en el proceso de clasificación para diseño y código llamados Apertura y Clausura. El primer paso se ejecuta cuando un defecto es descubierto y un nuevo reporte de defecto es incorporado al sistema de registro y seguimiento de defectos. El otro paso se ejecuta cuando el defecto es detectado, corregido y el reporte de defecto se encuentra cerrado.
- **Taxonomía de Beizer** [Bei90]: los defectos se clasifican por un número de 4 dígitos y a veces incluye subnumeración usando el punto ".". Además la letra "x" se utiliza de forma de posibilitar la expansión de la taxonomía. El último dígito de cada conjunto es el 9, que se utiliza cuando el defecto no coincide con ninguna de las categorías existentes o porque no existe una descomposición más fina. La taxonomía es puramente jerárquica y se basa principalmente en la causa del defecto.

La elección de la taxonomía de defectos a utilizar en un experimento depende fuertemente de lo que se quiera investigar, en algunos casos podría ser apropiada una taxonomía y en otros casos no. La incorrecta selección de la taxonomía dificulta la interpretación de los resultados, así como la comparación con experimentos similares o replicaciones. Debido a esto, es muy importante el proceso de selección de la taxonomía a utilizar.

Debido a que en nuestro experimento se quiere estudiar un aspecto particular de las técnicas (cómo se comportan respecto de los defectos que están dentro o fuera de su alcance), ninguna de las taxonomías existentes sirve para clasificar nuestros defectos, ya que la clasificación depende de la técnica que se esté usando. Por esto, los investigadores han creado su propia taxonomía de defectos, de acuerdo al ánimo de la investigación.

Para cada tipo de técnica, los investigadores han analizado su estrategia de genera-

ción de casos de prueba. A partir de ella se genera una serie de defectos para los cuales se espera que las técnicas muestren distinta capacidad de detección: alta para algunas y baja o nula para otras.

De acuerdo a este tipo de defectos es que se quiere estudiar el comportamiento de las técnicas, por tanto la clasificación contiene 2 tipos de defectos: (1) aquellos altamente visibles (o detectables) para las técnicas dinámicas (y no visibles por las técnicas estáticas) y (2) aquellos visibles para las técnicas estáticas (y no visibles para las dinámicas). Para simplificar, diremos que un defecto está "dentro" o "fuera" del alcance de una técnica.

Es sabido por los investigadores que la representatividad de estos tipos de defectos no está demostrada, de igual forma se considera válida para estudiar el fenómeno.

Se siembran 6 defectos en total, en donde: 3 defectos que están dentro del alcance de las técnicas del tipo funcional (y fuera de las del tipo estructural) y 3 defectos que están dentro del alcance de las técnicas del tipo estructural (y fuera de las del tipo funcional). Existe la posibilidad de que la distribución de los defectos sembrados no se adecue a la realidad (distribución que hoy en día es desconocida), pero de igual forma, permite establecer conjeturas sobre los tipos de defectos que son detectables por cada técnica.

Capítulo 4

Descripción de la Serie de Experimentos de UPM

Los experimentos de UPM nacen en el año 2001 como una subfamilia de replicaciones diferenciada de la familia de experimentos de Basili y Selby [BS85] [BS87] iniciada en 1982, que luego fue replicada por diversos investigadores tales como Kamsties y Lott [KL95], Wood y Roper et al. [WRBM97] y Shull et. al. [SMB+04]. De estos últimos, se utilizan parte de los paquetes de laboratorio, aunque el material ha sido adaptado en términos de objetivos y contextos.

La familia UPM se divide en dos series de replicaciones. La primera se enmarca en el período que va desde el año 2001 hasta 2005 [JVS+12], en donde el objetivo era analizar la efectividad de 3 técnicas de verificación unitaria: Lectura por abstracciones sucesivas (técnica de revisión), Cobertura de decisión (técnica dinámica y estructural) y Particiones en clases de equivalencia (técnica dinámica y funcional). La efectividad se medía de acuerdo a la capacidad de los casos de prueba generados (aplicando cada técnica) en detectar los defectos sembrados en 3 programas (los mismos usados por Kamsties y Lott con leves modificaciones).

En la primer serie de replicaciones los resultados mostraron que las técnicas estructural y funcional (Cobertura de decisión y Particiones en clases de equivalencia) se comportaban de igual forma, no habiendo diferencias de efectividad entre ellas. Sin embargo y a pesar de los resultados obtenidos, los investigadores intuían que estas técnicas no siempre se comportaban de igual forma, debido a sus diferencias de naturaleza y de estrategias de prueba, en donde una debería ser más sensible a determinado tipo de defectos que la otra no (por ejemplo: la técnica estructural es sensible a detectar código muerto o inalcanzable, mientras que la funcional no).

El diseño de la primer serie de replicaciones no era capaz de diferenciar la sensibilidad particular de las técnicas funcional y estructural. Esto en gran parte se debía a la clase de faltas sembradas en los programas, la cual no era discriminante respecto de la técnica. En base a ésto, los investigadores modificaron las faltas inyectadas, a modo de que las mismas fueran sensibles (o no) a las técnicas, de donde nace la segunda serie de replicaciones de UPM.

La segunda serie de replicaciones es sobre la que trata este trabajo y se enmarca desde el año 2006 hasta la fecha, aunque nosotros utilizaremos las replicaciones realizadas hasta 2012. La principal diferencia es el tratamiento de las faltas. En concreto, las faltas pueden dividirse en dos grandes tipos: faltas que pueden ser detectadas por la estrategia -estructural o funcional- de generación de casos de prueba (en lo que sigue faltas InScope) y, aquellas que (en teoría) no pueden detectarse (faltas OutScope). La diferenciación de los dos tipos de faltas permite estudiar la efectividad de las técnicas de testing de forma más precisa que en experimentos anteriores ya que es posible diferenciar el efecto de la técnica propiamente dicho (esto es, la detección de las faltas InScope) de los efectos positivos que la técnica induce en el tester pero que no pueden adscribirse a la técnica de por sí (esto es, las faltas OutScope).

De cierta forma, el objetivo cambia entre esta serie de replicaciones y la anterior. El tipo de efectividad que se está midiendo es diferente y representa cosas distintas. En esta serie de replicaciones la efectividad se descompone en dos tipos distintos, de acuerdo a los tipos de fallos revelados por los casos de prueba (OutScope e InScope), mientras que en la primer serie de experimentos la efectividad se calcula en base todos los fallos.

A continuación, se organiza el capítulo de la siguiente forma:

- En la sección 4.1 se presentan los trabajos relacionados en el área de investigación que a su vez utilizan procedimientos experimentales como método de investigación
- La sección 4.2 plantea el objetivo de estudio de esta segunda serie de replicaciones de la familia UPM
- El diseño experimental, procedimiento de análisis y reporte de análisis se describen en las secciones 4.3, 4.4 y 4.5.
- Las amenazas a la validez del experimento se presentan en la sección 4.6

4.1. Trabajos relacionados

Dentro de las disciplinas que forman parte de la Ingeniería de Software, la Verificación y Validación (V&V) ha tomado un papel de relevancia cada vez mayor a lo largo de los años. Es una de las actividades principales en lo que refiere al aseguramiento de la calidad del software. Dependiendo del nivel de riesgos asociados a fallas y el nivel de calidad que se quiera asegurar, las actividades de verificación y validación pueden tener asociados costos muy altos, incluso mayores que la propia actividad de desarrollo.

Mejorar las actividades de V&V a modo de lograr un proceso más eficiente ha sido foco de investigación desde numerosos puntos de vista. Dada la gran cantidad y variedad de técnicas, resulta interesante conocer la efectividad de las mismas y de qué variables depende [Ber04], por ejemplo: tipo de sistema a desarrollar, tipos de defectos que el sistema contiene, experiencia de los verificadores, lenguaje de programación y otras tantas características que puedan tener los sistemas y proyectos de software.

La investigación empírica es una forma de conocer la efectividad y costo de las técnicas de verificación y muchos investigadores han optado por utilizar este método para responder a sus preguntas de investigación.

Existe una gran variedad de trabajos de investigación que varían tanto el tipo de estudio experimental (estudios teóricos utilizando técnicas analíticas, casos de estudio, experimentos controlados, replicaciones), como las técnicas o procesos a evaluar, qué variables se quiere medir y comparar (efectividad, costo, eficiencia) y en base a qué se calculan (defectos encontrados, cobertura de código y otros).

Algunos estudios tratan sobre la efectividad de técnicas basadas en las faltas como ser el testing de mutantes (*mutant yesting*). Lee et. al. [LMK04] evalúan la ortogonalidad de operadores mutantes para lenguajes orientados a objetos y su eficiencia en la generación de mutantes en comparación con lenguajes procedurales . Offut [OL94], [OLR+96] estudia el costo computacional de las técnicas mutantes evaluando la mutación débil (*weak mutation*) y la mutación selectiva (*selective mutation*). Wong [WM95] y posteriormente Frankl [FWH97] comparan la efectividad respecto de la detección de faltas entre el testing de mutantes y el criterio de cubrimiento de todos los usos de flujo de datos (*all-uses data flow criteria*).

Las técnicas estáticas, de revisión e inspección de código han sido foco de numerosos estudios. En [BGL+96] Basili estudia la efectividad de las técnicas de revisión basadas en la perspectiva (*Perspective-Based Reading Techniques*) en comparación con las técnicas usuales de revisión que aplican los desarrolladores profesionales de la NASA. Este experimento fue luego replicado por Maldonado [MCS+06] comparando las técnicas de PBR contra el uso de checklist en inspecciones de código. En [BPV95] Basili y Porter realizan un experimento en 1993 que luego replican en 1995 sobre diferentes técnicas de detección de faltas en procesos de inspección aplicadas a la especificación de requisitos (SRS). Biffl [Bif00] también estudia las técnicas de inspección de requisitos a través de un experimento controlado en el que mide la influencia de la técnica de inspecciones sobre la performance individual del sujeto, en base a los defectos que detecta en una especificación de requisitos con un conjunto de defectos sembrados. Myers [Mye78] compara la efectividad de las inspecciones y recorridas de código contra técnicas de testing. Dunsmore y Roper [DRW02] evalúan la efectividad de tres técnicas complementarias de revisión de código: una basada en un checklist con problemas identificados del desarrollo orientado a objetos, otra basada en la construcción sistemática de especificaciones abstractas y la última centrada en la parte dinámica de los casos de uso.

También las técnicas dinámicas han sido evaluadas y comparadas por muchos investigadores. Frankl et. al. [FD00] realizan un caso de estudio comparando tres técnicas de verificación: criterio de decisión (*branch testing*), cubrimiento de todos los usos de flujo de datos (*all-uses data flow criteria*) y testing operacional, midiendo su efectividad en base a las faltas de alto riesgo que las mismas detectan, sobre programas medianos con faltas naturalmente generadas por el desarrollo de los mismos. Hutchins [HFGO94] realiza un estudio experimental que evalúa dos criterios de adecuación de conjuntos de casos de prueba: todas las aristas (*alledges*) y todos los usos (*all uses*) modificada, sobre 130 programas defectuosos derivados de 7 programas de tamaño mediano sembrados con faltas reales. Los investigadores generan los casos de prueba para todos los programas según cada criterio y evalúan la relación entre la detección de faltas y el

cubrimiento.

Vallespir et. al. [VAD+09], [VADR09], [VMBH09], realizan una serie de replicaciones diferenciadas de un experimento controlado que evalúa diferentes técnicas de verificación respecto de su efectividad y costo. En [VAD+09] evalúan 5 técnicas de verificación unitaria (inspecciones de escritorio, particiones en clases de equivalencia combinada con análisis de valores límites, tablas de decisión, caminos linealmente independientes y criterio de cubrimiento de condición múltiple) sobre un programa de pequeñas dimensiones con faltas sembradas, llevado a cabo por estudiantes de grado. Luego en [VADR09] realizan una replicación diferenciada aplicando las mismas técnicas sobre 3 programas de mediano-pequeño tamaño con faltas generadas naturalmente por la construcción del software por parte del desarrollador. Por último en [VMBH09], se replica el experimento cambiando las técnicas de verificación por las técnicas: todos los usos y cubrimiento de sentencias.

4.2. Objetivos del estudio, hipótesis y contexto experimental

De acuerdo al problema planteado previamente, el objetivo de este estudio se resume en:

> **Analizar** la aplicación de técnicas de verificación **con el propósito** de comprender su efectividad a nivel unitario respecto del tipo de faltas que detectan, **en el contexto** de un experimento controlado **llevado a cabo por** estudiantes universitarios.

La **hipótesis nula** en términos generales para el experimento es: *No existe diferencia en la efectividad de las técnicas particiones en clases de equivalencia, cobertura de decisión y lectura por abstracciones sucesivas, con respecto a las faltas que éstas detectan, tanto dentro como fuera de su alcance.*

A continuación se mencionan las técnicas de verificación que se aplican en el experimento, la taxonomía de defectos que se utiliza y los programas a verificar. Además se detalla el método de cálculo de la efectividad y particularidades del contexto experimental.

4.2.1. Técnicas de Verificación

Las técnicas de verificación que se utilizan para este estudio son las siguientes:

- Técnica estática (o de revisión):
  - Lectura por abstracciones sucesivas (reading by stepwise abstraction) [Lin79] (LAS).
- Técnicas dinámicas:
  - Particiones en clases de equivalencia [SLS06] (PCE).

- Criterio de cubrimiento de decisión [SLS06] (CD).

La técnica de partición en clases de equivalencia (PCE) consiste en el estudio de la especificación del sistema para derivar clases de comportamiento equivalente. Se definen subconjuntos de los datos de cada una de las entradas. Cada entrada del sistema define subconjuntos válidos y no válidos de valores. Para cada clase, se deben seleccionar datos concretos. Está técnica constituye una de las formas más básicas de prueba funcional [SLS06].

A partir de la técnica PCE, pueden definirse varias formas de generar los casos de prueba. La forma más simple es la selección de un representante de cada subconjunto, el experimento estudia esta variante. No se pretende analizar los valores límite del sistema, ni hacer un análisis de robustez o sobrecarga. Estas son extensiones válidas de la técnica, pero no son estudiadas en el experimento para permitir la comparación de resultados entre replicaciones.

La técnica de prueba estructural propuesta para estudio, es la de criterio de **cobertura de decisiones**. Este criterio implica que cada decisión debe ejecutarse con una evaluación verdadera y falsa al menos una vez. Las expresiones se consideran globalmente, independientemente del número de predicados o condicionales que haya en las mismas. El sujeto debe hacer un ejercicio de interpretación del programa, determinando las entradas que provocan estas evaluaciones [SLS06].

La técnica de lectura por abstracciones sucesivas es una técnica estática que no requiere la ejecución dinámica del sistema. El método de aplicación de esta técnica comienza con proporcionarle el código fuente del programa a los sujetos. Éstos deben identificar todas las funciones o métodos del programa y escribir las especificaciones de cada una, de acuerdo a lo que la función realiza. Luego agrupan las funciones y sus especificaciones y repiten el proceso hasta haber abstraído la especificación completa de todo el código fuente. Cuando terminan el ciclo de abstracciones se le brinda al sujeto la especificación original del programa. Tomando la epecificación original y la generada a través de las abstracciones, los sujetos deben encontrar las faltas en los programas a partir de las inconsistencias identificadas entre la especificación abstraída y la original [Lin79]

4.2.2. Variables de Respuesta

Existen varias formas de medir la *efectividad* para estudios de este tipo. Medirla únicamente respecto de la totalidad de los defectos podría llevarnos a concluir o deducir características erróneas de las técnicas. Por ejemplo: supongamos que la técnica funcional obtiene un porcentaje de 50 % de efectividad, al igual que la estructural. ¿Podríamos interpretar que ambas técnicas se comportan de igual forma? Observando los tipos de defectos que ambas detectaron se tiene que la mayoría de los defectos que detectó la técnica funcional fueron aquellos que supuestamente no eran visibles para la técnica y los que detectó la técnica estructural fueron únicamente aquellos que estaban dentro de su alcance (eran visibles para la misma). ¿Se puede seguir concluyendo que ambas técnicas se comportan de igual forma? ¿Qué pasaría si se conociera la distribución de defectos de un tipo y otro en los programas? Se podría decir que este método de medir

la efectividad es más fino que medir la efectividad en general y persigue un objetivo más concreto.

Para poder observar la efectividad desde distintos puntos de vista, para este estudio se descompone la medida de efectividad en tres variables de respuesta:

1. Efectividad de la técnica sobre los defectos que se encuentran **dentro de su alcance**: Efectividad *InScope*
2. Efectividad de la técnica sobre los defectos que se encuentran **fuera de su alcance**: Efectividad *OutScope*
3. Efectividad de la técnica sobre la **totalidad de los defectos**: Efectividad *AllFaults*

A continuación se describen cada una de las variables y la forma en la cual se calculan.

- **Efectividad InScope**: Porcentaje de defectos que detecta un sujeto, mediante los Casos de Prueba generados, de los defectos que se encuentran dentro del alcance de la técnica.

  Método de cálculo: El cálculo de la efectividad varía de acuerdo a la técnica que es aplicada, ya que ésta se calcula sobre los defectos observables para esa técnica, los cuales varían de una técnica a otra. En el caso de la técnica estructural, los defectos que se toman en cuenta son aquellos que provocan los fallos F1, F2 y F3. En el caso de la técnica funcional, se toman en cuenta los defectos que provocan los fallos F4, F5 y F6. Para el caso de la revisión de código se toman en cuenta todos los defectos (que provocan los fallos de F1 a F6) ya que todos se encuentran dentro de su alcance. A continuación se describe el cálculo para la efectividad de cada técnica:

  Efectividad InScope para LAS = $(\frac{\sum_{F=1}^{6} Fallos\ detectados}{6})100$

  Efectividad InScope para CD = $(\frac{\sum_{F=1}^{3} Fallos\ detectados}{3})100$

  Efectividad InScope para PCE =$(\frac{\sum_{F=4}^{6} Fallos\ detectados}{3})100$

- **Efectividad OutScope**: Porcentaje de defectos que detecta un sujeto de aquellos que están fuera del alcance (en teoría) de la técnica que aplica.

  Método de cálculo: El cálculo para esta efectividad se realiza únicamente para las técnicas dinámicas (CD y PCE), ya que para la técnica de revisión (LAS) todos los defectos están dentro de su alcance. Los defectos que se toman en cuenta en este caso son aquellos que no deberían haber sido detectados por la técnica (en teoría, están fuera de su alcance). A continuación se describe el cálculo para la efectividad de cada técnica:

  Efectividad OutScope para CD = $(\frac{\sum_{F=4}^{6} Fallos\ detectados}{3})100$

  Efectividad OutScope para PCE =$(\frac{\sum_{F=1}^{3} Fallos\ detectados}{3})100$

- **EfecAllFaults**: Porcentaje de defectos que detecta un sujeto de todos los defectos sembrados.

  Método de cálculo: En este cálculo se toman en cuenta todos los defectos detectados por cada técnica y se calcula sobre todos los defectos sembrados, sin distinguir entre los que están dentro o fuera del alcance de cada técnica. A continuación se describe el cálculo para la efectividad de cada técnica:

  Efectividad AllFaults para LAS = $(\frac{\sum_{F=1}^{6} Fallos\ detectados}{6})100$ (Coincide con el cálculo para la VR EfecInScope)

  Efectividad AllFaults para CD = $(\frac{\sum_{F=1}^{6} Fallos\ detectados}{6})100$

  Efectividad AllFaults para PCE =$(\frac{\sum_{F=1}^{6} Fallos\ detectados}{6})100$

4.2.3. Programas

Para este experimento se utilizan tres programas escritos en lenguaje C que poseen características similares:

- Cmdline: programa del tipo *parser* el cual procesa una línea de entrada y devuelve un resumen de su contenido.
- NameTbl: implementa una estructura de datos y operaciones a partir de una tabla de símbolos.
- Ntree: implementa la estructura de datos y operaciones de un árbol n-ario.

En el cuadro 4.1 se resumen algunas medidas de los programas. Debido a que estos programas son utilizados para la realización de los experimentos, no se publica ni la especificación ni el código fuente de los mismos debido a que podrían ser vistos por estudiantes, invalidando los resultados experimentales.

Métrica	**CmdLine**	**Nametbl**	**Ntree**
Cantidad de líneas de código totales	287	293	238
Cantidad de líneas de código físicas	232	225	199
Cantidad de líneas de código lógicas	144	141	137
Nº de complejidad McCabe VG	46	27	23
Cantidad de líneas de comentarios	23	31	28

Cuadro 4.1: Métricas de los programas

4.2.4. Defectos

Cada uno de los programas contiene seis defectos (faltas) distintos, F1 a F6. En donde tres de ellos (F1, F2 Y F3) están dentro del alcance de la técnica PCE y fuera de CD (tipo 1), y los restantes tres (F4, F5 y F6) están dentro del alcance de CD y

fuera de PCE (tipo 2). La descripción de cada defecto inyectado y la falla que provoca en cada programa tampoco pueden ser publicados (debido al mismo problema que presenta publicar los programas), por lo que presentamos la definición de los tipos de faltas inyectadas:

Faltas por omisión: son las originadas como resultado de olvidar alguna entidad por parte del programador (ausencia de código necesario).

Faltas por comisión: son las que se producen como consecuencia de un segmento de código incorrecto (presencia de código incorrecto).

- *Inicialización*. Una falta de comisión es, por ejemplo, asignar un valor incorrecto a una variable en una función, mientras la falta de inicialización de una variable es una falta de omisión. El experimento tiene faltas de inicialización de comisión (F4) y de omisión (F3).
- *Control*. Una falta de comisión es, por ejemplo, un predicado incorrecto en la condición de una decisión, mientras que si falta el predicado es una falta de omisión. El experimento tiene faltas de control de comisión (F5) y de omisión (F6).
- *Computación*. Una falta de comisión puede ser un operador aritmético incorrecto en la parte derecha de una asignación. El experimento tiene faltas de computación de comisión (F7).
- *Cosmética*: Una falta de comisión es, por ejemplo, una palabra mal escrita en un mensaje de error. En las faltas de omisión falta un mensaje de error. El experimento cubre faltas de comisión (F2) y de omisión (F1).

4.2.5. Características de los Sujetos

El experimento se desarrolla en un contexto académico, en el marco de la asignatura *Evaluación de Sistemas de Información* que se dicta para estudiantes de pre-grado de la carrera de *Ingeniero en Computación* de la Facultad de Informática de la Universidad Politécnica de Madrid.

Todos los sujetos son estudiantes que realizan el experimento como parte de la evaluación final de la asignatura antes mencionada. Esto implica que los estudiantes estén involucrados y comprometidos en la realización de sus tareas, aspecto importante que influye en la validez de los resultados [WRH99].

4.3. Diseño experimental

El diseño experimental de la familia de experimentos UPM es del tipo *cross-over* [Kue99] con un solo factor de estudio, teniendo ciertas variaciones indeseadas de una o más variables que deben ser bloqueadas, siendo además un diseño de bloque [JM01] (por dudas consultar la sección 2.1).

El diseño consta de 1 solo factor de estudio con 3 alternativas:

Factor: Técnica de verificación:

- Particiones en clases de equivalencia (PCE)
- Criterio de cobertura de decisiones (CD)
- Lectura por abstracciones sucesivas (LAS)

En el cuadro 4.2 se visualiza el diseño experimental en términos de cantidad de sesiones que se realizan, programas y técnica que se aplica para cada sesión y orden de las técnicas a aplicar. El experimento se realiza en 3 sesiones, cada una en un día distinto, con separación de una semana como ya se mencionó. En cada sesión se verifica un único programa: en la primera sesión Nametbl, en la segunda Cmdline y en la tercera Ntree. La muestra de sujetos se divide en 6 grupos (las distintas combinaciones de aplicación de cada técnica) en donde por ejemplo, el grupo LAS-CD-PCE, aplica la técnica Stepwise Abstraction en la primer sesión sobre el programa Cmdline, aplica Criterio de Decisión en la segunda sesión sobre el programa Nametbl y en la tercer sesión aplica Particiones en Clases de Equivalencia sobre el programa Ntree.

Programa / Día	**Cmdline**			**Nametbl**			**Ntree**		
Día	**Día 1**			**Día 2**			**Día 3**		
Grupo	**LAS**	**CD**	**PCE**	**LAS**	**CD**	**PCE**	**LAS**	**CD**	**PCE**
LAS-CD-PCE	X	-	-	-	X	-	-	-	X
LAS-PCE-CD	X	-	-	-	-	X	-	X	-
CD-LAS-PCE	-	X	-	X	-	-	-	-	X
CD-PCE-LAS	-	X	-	-	-	X	X	-	-
PCE-LAS-CD	-	-	X	X	-	-	-	X	-
PCE-CD-LAS	-	-	X	-	X	-	X	-	-

Cuadro 4.2: Diseño Experimental

Como se mencionó en la sección 2.2.3.3, la utilización de diseños cross-over trae aparejado la introducción de nuevas variables que influencian a la variable respuesta: el **período**, la **secuencia** y el carry-over. En nuestro experimento se tiene la siguiente correspondencia:

- Secuencia: son los grupos experimentales en los cuales se dividen los sujetos, representados por el orden de aplicación de las técnicas (LAS-CD-PCE, LAS-PCE-CD, CD-LAS-PCE, CD-PCE-LAS, PCE-LAS-CD y PCE-CD-LAS)
- Período: son las sesiones experimentales, en donde los sujetos aplican una técnica en cada una: Sesión 1, Sesión 2 y Sesión 3.

4.3.1. Procedimiento Experimental

El procedimiento experimental consta de 6 sesiones en total. Las primeras 3 sesiones consisten a entrenamiento de los sujetos. Las 3 restantes corresponden a la ejecución del experimento.

Las 3 primeras sesiones de entrenamiento se destinan a enseñar la teoría y estrategia de cada técnica de verificación a aplicar en el experimento. Se dedica una sesión de 4 horas para cada técnica: (1) Particiones en Clases de Equivalencia, (2) Cubrimiento de Decisión y (3) Lectura de código por Abstracciones Sucesivas. Cada sesión incluye la explicación de la teoría de cada técnica y una práctica de la estrategia de aplicación. Con estas tres sesiones se tiene un total de 12 horas de entrenamiento de los sujetos.

Las 3 últimas sesiones corresponden a la ejecución del experimento, que se realizan con 1 semana de separación entre sesión y sesión. En cada una de ellas los participantes aplican una única técnica de verificación sobre un solo programa.

Las sesiones de ejecución del experimento corresponden a las instancias evaluatorias de la asignatura, por tanto se realizan en salones de clases, vigilados por un docente o replicador responsable en donde los estudiantes no pueden interactuar entre sí (ya que la evaluación es individual) ni consultar ningún tipo de material.

Las tres evaluaciones tienen igual peso en la calificación del estudiante y son las que determinan la aprobación o no de la asignatura. En el proceso de evaluación se tienen en cuenta dos aspectos: la correcta aplicación de la técnica y el correcto seguimiento del procedimiento de aplicación de las técnicas. De esta forma, se intenta asegurar que los estudiantes se apeguen al proceso lo máximo posible, evitando variabilidades en las formas de aplicación y asegurándose que no se está aplicando otro tipo de técnica como complemento.

El procedimiento de aplicación de la técnica varía según la técnica que se aplica, ya que cada una tiene estrategias distintas por naturaleza. En los cuadros 4.3, 4.4 y 4.5 se presentan los procedimientos para las técnicas PCE, CD y LAS respectivamente.

Luego de realizada la fase operativa del experimento (ejecución de las sesiones de verificación), se debe realizar una corrección de lo entregado por cada estudiante (sujeto) en cada sesión del experimento, con un posterior procesamiento de los datos obtenidos. De la corrección se obtiene: faltas detectadas por el sujeto y nivel de apego al proceso para la aplicación de cada técnica. El procesamiento posterior debe realizar el cálculo de cantidad de faltas detectadas *InScope*, *OutScope* y *AllFaults* para luego calcular la efectividad obtenida para cada una de las variables de respuesta.

En la evaluación del estudiante, se tienen en consideración 2 aspectos: los fallos detectados por el estudiante y el correcto seguimiento del procedimiento de aplicación de la técnica en cuestión. Esto es, la correcta entrega de todos los formularios completos que componen la evaluación.

La obtención de los datos experimentales a través de los formularios se compone de 2 etapas, en la primera se realiza un **análisis de los fallos y casos de prueba** detectados y generados por el sujeto respectivamente, en la segunda etapa, se realiza el **cálculo de la efectividad**.

Procedimiento para la técnica Particiones en Clases de Equivalencia		
Paso	Actividades	Descripción
1	Generación de Casos de Prueba	1. Se le brinda al sujeto los formularios correspondientes a: ▪ Instrucciones para el sujeto. ▪ Especificación del Programa. ▪ Registro de clases de equivalencia. ▪ Registro de casos de prueba (datos de prueba y salidas esperadas). 2. El sujeto sigue las instrucciones generando las clases de equivalencia y los casos de prueba para el programa a verificar. 3. El sujeto entrega los formularios completos correspondientes a las clases de equivalencia y casos de prueba, los cuales se marcan por el instructor para que no pueda generar más casos de prueba.
2	Ejecución	1. Se le brinda al sujeto el programa ejecutable. 2. El sujeto ejecuta los casos de prueba diseñados en el paso anterior y registra las salidas observadas de cada caso de prueba en el formulario de casos de prueba.
3	Identificación de fallos	1. Se el brinda al sujeto el formulario de registro de fallos. 2. El sujeto registra los fallos encontrados en la ejecución de los casos de prueba en el formulario de registro de fallos, en base a las salidas incorrectas de la ejecución de los casos de prueba.
4	Finalización	1. El sujeto entrega todos los formularios completos al instructor. 2. El instructor da por finalizada la sesión del experimento para ese sujeto.

Cuadro 4.3: Procedimiento para la técnica Particiones en Clases de Equivalencia

Los fallos reportados por el sujeto se clasifican en *observables por la técnica* y *observados por el sujeto*. Por ejemplo, si un sujeto aplica una técnica dinámica para generar un caso de prueba que descubre un defecto el cual genera un fallo, este fallo es observable por la técnica. Sin embargo, el sujeto podría no ver (observar) el fallo, y por tanto no reportarlo. En las técnicas estáticas, los defectos observables coinciden con los observados. La efectividad de las técnicas refieren a los defectos observables.

Ahora bien, el sujeto podría reportar un fallo que no fue detectado por la generación

Procedimiento para la técnica de Cubrimiento de Decisión		
Paso	Actividades	Descripción
1	Generación de Casos de Prueba	1. Se le brinda al sujeto los formularios correspondientes a: ▪ Instrucciones para el sujeto. ▪ Código fuente del programa. ▪ Registro de casos de prueba parcial (solamente datos de prueba). 2. El sujeto sigue las instrucciones generando los datos de prueba para lograr obtener un cubrimiento lo más cercano posible al 100 % de las decisiones del código. 3. El sujeto entrega el formulario completo correspondiente a los datos de prueba, el cual se marca por el instructor para que no pueda generar más datos de prueba.
2	Ejecución	1. El instructor le brinda la especificación del programa, el formulario de salidas (esperadas y obtenidas) y el programa ejecutable. 2. El sujeto completa los casos de prueba con las salidas esperadas de acuerdo a la especificación del programa. 3. El sujeto ejecuta los casos de prueba diseñados y registra las salidas observadas de cada caso de prueba en el formulario de salidas.
3	Identificación de fallos	1. Se el brinda al sujeto el formulario de registro de fallos. 2. El sujeto registra los fallos encontrados en la ejecución de los casos de prueba en el formulario de registro de fallos, en base a las salidas incorrectas de la ejecución de los casos de prueba.
4	Finalización	1. El sujeto entrega todos los formularios completos al instructor. 2. El instructor da por finalizada la sesión del experimento para ese sujeto.

Cuadro 4.4: Procedimiento para la técnica de Cubrimiento de Decisión

de los casos de prueba, por tanto, ese fallo es observado por el sujeto, se toma en cuenta para la evaluación en sí, pero no se contabiliza como defecto detectado por la técnica en cuestión, ya que el fallo no fue revelado por un caso de prueba generado por la técnica. También, puede suceder el caso contrario, el sujeto genera y ejecuta un caso de prueba

Procedimiento para la técnica de Lectura por Abstracciones sucesivas		
Paso	Actividades	Descripción
1	Generación de Abstracciones	1. Se le brinda al sujeto los formularios correspondientes a: ▪ Instrucciones para el sujeto. ▪ Código fuente del programa. ▪ Registro de abstracciones. 2. El sujeto sigue las instrucciones generando las abstracciones que considere necesarias para cubrir todas las funciones del programa y la especificación del mismo. 3. El sujeto entrega el formulario completo correspondiente a las abstracciones y la especificación generada, el cual se marca por el instructor para que no pueda ser cambiado.
2	Búsqueda de Inconsistencias	1. El instructor le brinda la especificación del programa y el formulario de inconsistencias. 2. El sujeto identifica y registra las inconsistencias comparando la especificación brindada con la generada en el paso anterior, rellenando el formulario de inconsistencias.
3	Finalización	1. El sujeto entrega todos los formularios completos al instructor. 2. El instructor da por finalizada la sesión del experimento para ese sujeto.

Cuadro 4.5: Procedimiento para la técnica de Lectura por Abstracciones sucesivas

para detectar un defecto, el fallo aparece, pero el sujeto "no lo vé". En este caso, si bien el sujeto no reporta ese fallo como encontrado, éste se contabiliza como detectado por la técnica.

Por tanto, se realiza un doble chequeo entre fallos detectados y casos de prueba, verificando cuáles de los fallos reportados fueron producto de la aplicación de la técnica y qué casos de prueba detectaron fallos observables por la técnica que el sujeto obvió.

4.3.2. Variaciones del Diseño Experimental

El diseño experimental detallado anteriormente es el considerado "base" para la realización de una replicación del experimento. Este diseño puede sufrir modificaciones en distintas replicaciones ya sea porque el contexto experimental lo restringe o porque se quiera estudiar otro tipo de efectos.

Es necesario tener cuidado de que las variaciones introducidas en el diseño base no impacten en que las replicaciones no puedan ser comparables entre sí, por ejemplo: cambiar todos los niveles de los factores a analizar, cambiar la variable de respuesta o la forma de medirla, entre otros.

En el cuadro 4.6 se detallan las variaciones que sufrió el diseño base en las distintas replicaciones que van de 2006 a 2012.

Replicación	**Cambios**	**Observaciones**
2006	Ninguno	El orden de ejecución de los programas es: Cmdline-Ntree-Nametbl
2007	Ninguno	El orden de ejecución de los programas es: Ntree-Nametbl-Cmdline
2008	Ninguno	El orden de ejecución de los programas es: Ntree-Cmdline-Nametbl
2011	Se tienen 2 sesiones con 2 tratamientos del factor técnica (CD y PCE) y 2 tratamientos del factor programa (nametbl y ntree)	El orden de ejecución de los programas es: Ntree-Nametbl
2012	idem 2011	El orden de ejecución de los programas es: Nametbl-Ntree. El sitio de replicación es ESPEL.

Cuadro 4.6: Variaciones del Diseño Base en las sucesivas replicaciones del experimento

Para las replicaciones 2006, 2007 y 2008 no hubieron cambios significativos, el único cambio realizado fue el orden en el cual se organizaron los programas en las sesiones. Esto no implica un cambio en el diseño experimental como tal, ni tampoco en el procedimiento experimental. Estas tres replicaciones podrían caalogarse como "cercanas".

En el caso de las replicaciones 2011 y 2012 sí hubieron cambios de diseño experimental: se realizan 2 sesiones, en donde cada sujeto aplica una única técnica sobre un solo programa en cada una. El diseño se reduce en 1 tratamiento para el factor técnica (se quita LAS) y en 1 tratamiento para el factor programa (se quita cmdline).

La replicación de 2012 tuvo además otros cambios no tan referentes al diseño propiamente dicho, pero que podrían impactar en la interpretación y comparación de los resultados. Estos cambios son:

- Cambio de sitio experimental
- Cambio de tipo de sujetos
- Cambio en el procedimiento experimental

El sitio experimental de la replicación de 2012 se da en la Escuela Politécnica del Ejército Sede Latacunga (ESPEL), Ecuador, en el marco de un curso de Evaluación de

Sistemas Software. El curso se dictó para estudiantes de máster a diferencia del resto de las replicaciones en donde se estudian estudiantes de grado. En el procedimiento experimental, se cambia el tiempo entre sesiones, pasando de ser 1 semana (en el resto de las replicaciones) a 1 día de diferencia.

4.4. Procedimiento de Análisis

Elegir la técnica de análisis estadístico que se va a utilizar para el análisis de los datos es un aspecto clave de la realización de experimentos y replicaciones. Dado que no existe un correspondencia clara ni exacta de qué técnica utilizar dado un diseño experimental, es necesario estudiar las técnicas de análisis disponibles en pro de escoger la que mejor se adapte al diseño experimental que se tiene. Una buena adaptación de la técnica implica que ésta toma en cuenta todas las variables que pueden afectar el experimento y las caracteriza adecuadamente. Una incorrecta elección de la técnica puede provocar conclusiones y deducciones erróneas, haciendo fracasar el experimento.

La técnica se tiene que adaptar al modelo estadístico matemático que mejor modela las variables del diseño experimental. Un modelo estadístico es una expresión simbólica en forma de igualdad o ecuación que se emplea en todos los diseños experimentales para indicar los diferentes factores que modifican la variable de respuesta. En nuestro caso, el modelo estadístico para nuestro experimento es un *modelo de efectos mixtos*, en donde se encuentran variaciones causadas por factores fijos y aleatorios. La ecuación matemática para el modelo de nuestro experimento es la siguiente [Kue99]:

$$Y_{ijk} = \mu + \pi_k + \tau_i + \phi_u + \lambda_v + \xi_{ij} + \varepsilon_{ijk}$$
$$(k = 1 \ldots \text{p},\ i = 1 \ldots \text{g},\ j = 1 \ldots \text{n},\ u = 1 \ldots \text{r})$$

- μ: media general
- τ_i: efecto de la i-ésima secuencia de tratamiento
- ξ_{ij}: es el efecto aleatorio del h-ésimo sujeto de la i-ésima secuencia de tratamiento
- π_k: efecto del k-ésimo período
- ε_{ijk}: error aleatorio del sujeto en el período k
- ϕ_u: efecto directo del tratamiento administrado en el período k para la i-ésima secuencia de grupo
- λ_v: effecto de carry-over para el $v$ésimo tratamiento en el $k-1$-ésimo período para la respuesta en el siguiente k-ésimo período

μ, π_k, τ_i, ϕ_u y λ_v son efectos fijos, ξ_{ij} y ε_{ijk} son efectos aleatorios e independientes con media cero y varianzas σ^2_ξ y σ^2_ε, respectivamente.

Para analizar nuestro experimento, se siguen las indicaciones presentadas en [Sen02] y [Fre89] para el análisis de modelos mixtos, en donde se sugiere:

- Utilizar el **Análisis de Componentes de la Varianza** aplicado a modelos mixtos, también conocido como **modelo lineal mixto** (*linear mixed model*)
- No introducir en el modelo a evaluar el efecto del carry-over

Freeman y otros [Fre89] realizan estudios sobre numerosos experimentos con diseños de medidas repetidas y cross-over, en donde observaron que una mínima cantidad de esos experimentos (en el orden del 10 %) presentaban efectos significativos de carry-over, valores que pueden ser atribuídos a errores de tipo I y no a la real existencia del carry-over. Por tanto, se sugiere obviar del modelo a analizar el efecto del carry-over, teniendo en cuenta que los hallazgos pueden estar condicionados al supuesto de que la influencia del carry-over no ha distorsionado seriamente los resultados.

4.4.1. Aplicación de la técnica de análisis estadístico al diseño experimental

De acuerdo al análisis realizado previamente, se escoje la técnica de análisis de modelo lineal mixto (también conocida como *Análisis de Componentes de la Varianza* aplicado a modelos mixtos) que se aplica utilizando el software *Statistical Package for the Social Sciences* (SPSS)[1].

Littell, Pendergast y Natarajan [LPN00], citados por Vallejo en [VAB+10], explican que cuando se utilizan modelos mixtos es necesario modelar correctamente los dos aspectos fundamentales de los datos. Por un lado, los efectos fijos usados para describir el promedio de las respuestas en función del tiempo. Y, por otro lado, los efectos aleatorios usados para describir la variación entre medidas repetidas dentro de los sujetos.

Dentro de los modelos mixtos existen distintas formas de modelar estos efectos, la selección del modelo que mejor se ajuste resulta central para interpretar correctamente los datos. Para ésto se utilizan los llamados *Criterios de Información* que mediante el análisis de los datos determina un coeficiente el cual, cuanto más pequeño, indica un mejor ajuste. [VAB+10]

Existen numerosos criterios que se pueden utilizar para seleccionar el modelo que mejor se ajuste, qué criterio seleccionar es una cuestión que hoy en día no se ha dilucidado del todo. Según una serie de experimentos realizada por [VAB+10] y estudios realizados por [LPN00] se concluye que el Criterio de Información de Akaike (AIC) [Aka74] resulta bastante adecuado para el estudio de medidas repetidas y modelos mixtos multinivel como es nuestro caso.

Para seleccionar el modelo que mejor se ajuste a nuestro experimento se realiza un ranking de puntuación de cada modelo evaluado en cada replicación del experimento, seleccionando aquel mejor rankeado. Inicialmente se ordena de menor a mayor los resultados para cada modelo en cada réplica y luego tomando los valores de todas las réplicas se vuelve a rankear y de ahí se selecciona el mejor rankeado.

En el cuadro 4.7 se describen los modelos evaluados de los cuales se obtiene el AIC para seleccionar el que mejor se ajuste. Entre las distintas opciones que ofrece el

[1] http://www-01.ibm.com/software/es/analytics/spss/products/statistics/

SPSS para evaluar los modelos lineales mixtos, se seleccionaron aquellas que resultan de interés para el diseño de nuestro experimento:

1. **Considerar al sujeto como factor aleatorio:** es de interés ver si al considerar el sujeto como factor aleatorio, ésto conduce a un modelo mejor ajustado. Por tanto se evalúan los modelos con y sin esta consideración.

2. **Tipos de covarianza:** Los efectos aleatorios modelan la estructura de las covarianzas de la variable dependiente. El SPSS ofrece 16 tipos distintos de covarianzas para aplicar al modelo, de éstos 16 seleccionamos aquellos que se recomiendan en los tutoriales de SPSS: *Identidad Escalada*, *Sin Estructura* y *Diagonal*.

3. **Factor Medidas Repetidas:** Para este diseño experimental, se puede considerar que existen medidas repetidas tanto para el factor Técnica (los sujetos aplican todos los niveles del factor técnica) como para el factor Programa/Sesión (los sujetos verifican todos los programas). Sin embargo, no se puede considerar a ambos factores como de medidas repetidas ya que ello significa que el sujeto aplica todas las combinaciones del producto cartesiano de los niveles de cada factor. Entonces los modelos evaluados son aquellos que consideran al factor técnica como de medidas repetidas, ya que es el factor que nos interesa.

Modelo	Factor sujeto aleatorio	Covarianza			Factor Medidas Repetidas	
		Identidad Escalada	Sin Estructura	Diagonal	Técnica	Programa
Modelo 1						
Modelo 2	X					
Modelo 3		X			X	
Modelo 4	X	X			X	
Modelo 5			X		X	
Modelo 6	X		X		X	
Modelo 7				X	X	
Modelo 8	X			X	X	
Modelo 9		X				X
Modelo 10	X	X				X
Modelo 11			X			X
Modelo 12	X		X			X
Modelo 13				X		X
Modelo 14	X			X		X

Cuadro 4.7: Modelos Mixtos Evaluados

Finalmente los modelos evaluados son la combinación de todas las alternativas de las características seleccionadas que se detallaron previamente, lo que da un total de 14 modelos a evaluar. La lectura de los modelos del cuadro 4.7 se realiza de la siguiente forma: en cada casillero se marca con una "X" si se selecciona esa opción o no para el modelo. Por ejemplo: en el modelo 1 no se considera el sujeto como factor aleatorio, no se considera ningún factor como de medidas repetidas y por tanto no se selecciona el tipo de covarianza. En el modelo 6 se considera el sujeto como factor aleatorio, se determina que la técnica es un factor de medidas repetidas y se selecciona una covarianza sin estructura.

En el cuadro 4.8 se detalla el AIC obtenido para cada uno y el ranking realizado para cada replicación. Las replicaciones que se muestran para el ranking son las de 2006,

2007 y 2008, para las replicaciones de 2011 y 2012 se realiza el mismo procedimiento obteniendo iguales primeras posiciones en el ranking (no se muestran los resultados para simplificar y no ser repetitivo)

VALORES																			PUNTAJE TOTAL
Modelo	Año 2006						Año 2007						Año 2008						
	ObsCP		ObsCPComp		Obs		ObsCP		ObsCPComp		Obs		ObsCP		ObsCPComp		Obs		
	AIC	Rank	AIC	Rank	AIC	Rank	AIC	Rank	AIC	Rank	AIC	Rank	AIC	Rank	AIC	Rank	AIC	Rank	
Modelo 1	1211,8	3	771,7	1	1143,8	6	1282,2	9	823,4	5	1203,3	8	1190,2	5	739,4	8	1096,1	3	48
Modelo 2	1212,5	11	773,7	5	1144,9	9	1280,1	5	824,3	8	1203,2	5	1191,8	10	739,3	5	1096,3	6	64
Modelo 3	1211,8	12	771,7	2	1143,8	7	1282,2	10	823,4	6	1203,3	9	1190,2	6	739,4	9	1096,1	4	65
Modelo 4	1212,5	5	773,7	6	1144,9	10	1280,1	6	824,3	9	1203,2	6	1191,8	11	739,3	6	1096,3	7	66
Modelo 5	1220,0	8	775,4	9	1147,5	13	1277,4	3	820,3	4	1201,9	3	1188,6	3	740,3	11	1101,1	12	66
Modelo 6	1222,0	9	777,4	12	1149,5	14	1279,4	4	794,8	1	1176,3	1	1190,6	8	742,3	14	1103,1	14	77
Modelo 7	1215,5	10	773,4	4	1142,5	3	1276,9	2	819,6	2	1202,8	4	1183,7	1	740,4	13	1098,7	9	48
Modelo 8	*1216,2*	*1*	*775,4*	*10*	*1143,5*	*4*	*1274,9*	*1*	*820,3*	*3*	*1200,7*	*2*	*1185,4*	*2*	*740,3*	*12*	*1098,8*	*10*	*45*

Cuadro 4.8: Valores del Criterio de Informaciónd de Akaike y Ranking para los Modelos Evaluados

Cada modelo se evalúa 9 veces en total, una vez para cada variable de respuesta (3) de cada replicación (3). Para cada combinación de VR-Replicación se genera un ranking de acuerdo al AIC obtenido para cada modelo, estando en primer lugar el AIC más bajo (aquel que mejor se ajusta). El puntaje total del modelo se calcula sumando los rankings obtenidos en cada combinación VR-Replicación. Finalmente, se selecciona aquel modelo que obtenga menor puntaje en la sumatoria.

El modelo seleccionado es el Modelo 8, que se destaca en el cuadro 4.8. Este modelo es el que se usará para el análisis estadístico de todas las replicaciones.

Para poder aplicar la técnica de Análisis de componentes de la varianza para modelos mixtos, es necesario chequear que los datos cumplen con los supuestos requeridos por este tipo de análisis (analizar la validez del modelo). Los supuestos para este tipo de análisis son:

1. Que el término residual tenga una media de cero y una varianza constante finita y que no tenga correlación con respecto a los parámetros del modelo de cualquier efecto aleatorio.

2. Se asume que los términos residuales de diferentes observaciones no están correlacionados.

Para evaluar si los datos a analizar cumplen con el supuesto requerido para la aplicación de este tipo de análisis, se debe realizar una prueba de normalidad sobre los residuos resultantes. Si los residuos presentan una distribución similar a la normal, es posible la aplicación de la técnica de análisis para modelos mixtos.

En caso de no cumplir con la normalidad de los residuos[2], es posible realizar transformaciones sobre los datos a modo de conseguir cumplir con el supuesto requerido. Las transformaciones de datos son muy usadas para éste y otro tipo de análisis que requieren normalidad de los datos en algún sentido [Kue99]. En caso de no cumplirse la

[2]Se requiere un nivel de significancia menor a 0,001 para rechazar la hipótesis nula de normalidad

normalidad de los residuos para alguno de los análisis a realizar, se prueba transformar los datos utilizando alguna de las transformaciones presentadas en el cuadro 4.9

Transformación	**Descripción**
y^2	Pontencia de 2
$\sqrt{y}$	Raiz cuadrada
$\ln(y)$	Logaritmo eneperiano
$\ln(y+\frac{1}{2})$	Logaritmo eneperiano de $y+\frac{1}{2}$
$\log_{10}(y)$	Logaritmo en base 10
$\log_{10}(y+\frac{1}{2})$	Logaritmo en base 10 de $y+\frac{1}{2}$
$\frac{1}{y}$	1 sobre y
$\frac{1}{\sqrt{y}}$	1 sobre raiz cuadrada de y
$\exp y$	Exponencial

Cuadro 4.9: Transformaciones de datos utilizadas en los análisis

El procedimiento de transformación de datos se realiza únicamente para aquellas variables con las cuales no se obtiene normalidad de residuos, este procedimiento consiste en:

1. Transformar la variable respuesta con cada una de las transformaciones presentadas en el cuadro 4.9
2. Analizar los datos con cada variable transformada
3. Verificar la normalidad de los residuos resultantes de cada análisis
4. Elegir aquella transformación que obtenga un mejor nivel de significancia para la normalidad de los residuos
5. Utilizar los resultados del análisis realizado con la mejor transformación obtenida

En caso de que ninguna transformación logre obtener residuos con distribución normal, de igual forma se utiliza la técnica de Análisis de componentes de la varianza para modelos mixtos, ya que este tipo de técnicas son robustas (poco sensibles) a la falta de normalidad [Ito80]. Aún así, los datos resultado de estos análisis son tratados con cierto cuidado, ya que presentan una amenaza a la validez agregada.

4.5. Reporte de Análisis

En base a lo sugerido en las guías de Jedlitschka [JP05] para reporte de experimentos, cada reporte de análisis detalla:

- **Estadísticas Descriptivas**: número de observaciones, media y desviación estándar, que se representan con gráficos de caja y bigotes (*bloxplots*).

- **Reducción del conjunto de datos**: en caso de que se decida omitir algún dato como resultado de un error o outlier.
- **Pruebas de Hipótesis**:
  - **Validez del modelo**: se presenta una evaluación de la validez del modelo respecto de los supuestos que requiere el procedimiento de análisis para *Modelos Mixtos*
  - **Estadísticas inferenciales**: estadístico F para estudiar la influencia de cada factor, gráficos de perfil y pruebas de comparaciones múltiples para estudiar el efecto de los niveles de cada factor sobre la variable de respuesta. Todas las pruebas son realizadas con un nivel de significancia de 0.05 %.
- **Resultados**: Por último se presenta un análisis de los resultados obtenidos y qué se puede inferir y qué no a partir de éstos.

4.6. Amenazas a la Validez

Es importante analizar las amenazas a la validez a las que podrían estar sujetos los resultados del experimento, a modo de realizar una correcta interpretación de los mismos y entender sus limitaciones.

Como se mencionó en el capítulo 2, hay cuatro tipos de amenazas que deben ser consideradas en el diseño y procedimiento experimental, así como también en el análisis de los datos: validez de la conclusión, validez del constructo, validez externa y validez interna.

Respecto de las amenazas que afectan la **validez de la conclusión** que refieren a las conclusiones estadísticas, si bien los métodos de análisis estadístico elegidos son los adecuados de acuerdo a las relaciones que se quieren analizar (y no se detectaron problemas en este sentido), el tamaño de la muestra podría llegar a ser un problema. Es sabido que cuanto más grande sea la muestra, mayor poder de prueba se tiene y más probabilidades de acierto se tienen al generalizar los resultados.

Si bien el tamaño de muestra mínimo teórico establecido es de 30 y en algunos experimentos no se llega a ese número (de ahí el utilizar un diseño cross-over para aumentar la cantidad de las muestras), tampoco está demostrado que con números menores los resultados no puedan ser utilizados. Además, el método de análisis para modelos mixtos utilizado, deja claro en sus premisas que logra un buen comportamiento con muestras pequeñas.

No se han detectado amenazas a la **validez del constructo**, que indica cómo una medición se relaciona con otras de acuerdo con la teoría o hipótesis que concierne a los conceptos que se están midiendo. La única variable que se mide y compara a lo largo del experimento es la efectividad, de tres formas diferentes: InScope, OutScope y AllFaults. Dichas mediciones se realizan de la misma forma en todas las replicaciones, tanto en la recolección de datos, como en el cálculo de la efectividad que se describió en la sección 4.2.2.

A continuación se enumeran algunas de las amenazas que creemos podrían influir en la **validez interna** (aquellas referidas a observar relaciones entre la alternativa y el resultado que sean producto de la casualidad y no del resultado de la aplicación de un factor) y cómo éstas han sido tratadas:

- **Copia**: El experimento se ejecuta en un laboratorio donde los sujetos son controlados por un docente. De esta forma se minimiza la posibilidad de que los estudiantes se copien unos a otros.
- *Capacidad*: La asignación al azar y el diseño cruzado (*cross-over*) minimiza el problema de que no todos los sujetos que tienen las mismas capacidades.
- **Aprendizaje de la técnica**: Ya que los sujetos aplican cada tratamiento una sola vez, no hay manera de que puedan aplicar de mejor forma una técnica en particular, como consecuencia del aprendizaje por repetición.
- **Aprendizaje sobre el programa**: Este efecto se evita haciendo que cada sujeto aplique una técnica diferente a un programa diferente. Los sujetos utilizan cada programa una única vez, a modo de no tener oportunidad de aprender sobre él.
- **El aprendizaje por la práctica**: Hemos incluido el factor de la sesión (como forma de estudiar el período) para estudiar si hay alguna mejora en la efectividad debida a la repetición de la verificación por parte de los sujetos.
- **El aburrimiento**: La calificación de este ejercicio debe motivar a los estudiantes a hacer su mejor trabajo.
- **Entusiasmo**: Los sujetos podrían preferir una técnica sobre las demás, que los pueda llevar a aplicarla de mejor forma que al resto. Se espera que el sistema de calificación ayude a superar este problema, ya que todas las técnicas contribuyen por igual a la calificación.
- **Formalización inconsciente**: Ninguna de las técnicas es más formal que el resto. Es poco probable que aprenda la formalización de una técnica y este conocimiento lo aplique en la siguiente.
- **Efecto de arrastre de una técnica a otra**. Los sujetos pueden "arrastrar" conocimiento u otro tipo de beneficio de la aplicación de una técnica a otra en forma sucesiva, para complementar la estrategia de la técnica que viene después. Por ejemplo, entre LAS y CD (los sujetos podrían utilizar las ventajas de la lectura de código para entender el programa y realizar de forma más efectiva las pruebas de cubrimiento de decisión). Esta amenaza es estudiada por la introducción del factor grupo (que representa la secuencia en el experimento), lo que asegura que si hay algún efecto de arrastre, sea detectado.
- **Factor Programa/Sesión con efectos combinados**: este problema surge por tener varias sesiones de verificación y un único programa asignado a cada una de ellas. Las dos formas de minimizar este aspecto serían, por un lado tener una única sesión con la consecuente pérdida de muestras (que es el objetivo) o verificar todos los programas en cada sesión, pero que trae consigo el riesgo de copia entre

los estudiantes, lo que invalidaría los resultados. Se opta entonces por minimizar esta amenaza analizando varias replicaciones separando ambos puntos de vista. Si bien es cierto que en una única replicación estos efectos son difíciles de separar, tomando en cuenta varias replicaciones que varían el orden de correspondencia de los programas resulta más claro.

- **Factor Grupo no homogéneo**: Comparar los resultados de este factor a través de las replicaciones no fue directo debido a que algunas presentaban diferencias de diseño. En la replicación de 2008 no se tuvo en cuenta balancear a los sujetos en todas las combinaciones posibles, con lo cual se conformaron 4 grupos y no 6, y que las replicaciones de 2011 y 2012 se conformaron grupos distintos (por no tener un nivel del factor técnica y una sesión menos). Esta amenaza se intentó minimizar separando el análisis en 2 subgrupos de replicaciones. Aún así, la conclusión a la que se llegó fue la misma para ambos subgrupos, en relación a que no se aprecian efectos de carry-over relacionado al orden de las técnicas, ya sean 2 o 3 sesiones las analizadas.
- **Apego al proceso**. Los sujetos podrían no cumplir con el procedimiento establecido para la aplicación de la técnica. Para evitar que esta amenaza, los sujetos son evaluados no sólo por la cantidad de fallos detectados, sino también por qué tan bien se han apegado al proceso de aplicación de la técnica de aplicación. Aún así, esta amenaza se minimiza, pero no se neutraliza.

La **validez externa** está relacionada con la habilidad para generalizar los resultados y se encuentra muy relacionada con el contexto, los objetos experimentales y el tipo de sujetos que ejecutan el experimento. Se enumeran a continuación las amenazas detectadas en este sentido, cómo se trataron y de que forma limitan los resultados obtenidos.

- **Contexto académico y no industrial**: el experimento se ejecuta en aulas de clase de la Universidad, bajo la modalidad de prueba evaluatoria para los sujetos. Cambios en el ambiente de ejecución del experimento podrían variar los resultados, por ejemplo: ambiente laboral.
- **Los sujetos son estudiantes de grado y posgrado**: los resultados han sido obtenidos por estudiantes sin experiencia en la industria ni en las técnicas de verificación a aplicar, teniendo únicamente una base teórica del proceso de aplicación de las técnicas, brindado por el docente de la asignatura. Los resultados no son generalizables de forma directa a profesionales u otro tipo de poblaciones.
- **Programas utilizados**
  - *Escritos en C*: los programas que se utilizan para el experimento están escritos en lenguaje C debido a que los sujetos están familiarizados con este lenguaje debido a que es utilizado en otras asignaturas que forman parte de la currícula de los estudiantes. Variar el lenguaje de programación podría generar variaciones en los resultados, así como también la relación que genera el lenguaje con el que está escrito el programa y la experiencia del tester en dicho lenguaje.

  - *De tamaño pequeño*: Los programas que se utilizan son pequeños, en el orden de las 200 líneas de código. Si bien es cierto que los sistemas de la industria de software contienen millones de líneas de código, la investigación se focaliza en la verificación unitaria, por lo que entendemos que el tamaño de los programas utilizados es acorde a los objetivos del experimento.
  - *Específicos*: utilizamos tres programas específicos. Los experimentos deben ser realizados con otros programas para poder entender otro tipo de poblaciones de programas, para las cuales los resultados se puedan aplicar y generalizar.

- **Faltas inyectadas**: tanto la cantidad como el tipo de faltas que contienen los programas han sido generadas por los investigadores y no son producto de la actividad natural del programador al construir los programas, por lo cual podrían no ser representativas de las faltas reales en el software. En nuestros programas tenemos una distribución del 50 % de faltas que son del tipo InScope y otro 50 % OutScope. Si quisiéramos mirar la efectividad "en general" de una técnica, sería necesario conocer la distribución real de este tipo de faltas dentro de los programas para conocer qué tan efectiva podría llegar a ser una técnica sobre éste.

- **Aplicación de la Técnica**: existen muchas formas distintas de aplicar una técnica de verificación a un programa, la cual podría estar combinada con el uso de herramientas o de otras técnicas. Los resultados de este experimento están condicionados a un procedimiento específico de aplicación de cada técnica que se les indicaba realizar a los sujetos. Los resultados podrían estar condicionados a estos procedimientos y variar si se usasen otros. Por ejemplo, el uso de herramientas de cobertura de código, herramientas de generación de casos de prueba, entre otras.

Capítulo 5

Análisis del Experimento Original (base)

En este capítulo se presenta el análisis estadístico del primer experimento de la serie (año 2006), el cual llamaremos *base*. Para este primer análisis se detalla la aplicación del proceso de análisis descrito en el capítulo anterior, a modo de ejemplificar y explicar la aplicación del mismo. El resto de las replicaciones serán analizadas de igual forma (de acuerdo al procedimiento descrito y la técnica que aplique), pero solamente se presentarán los resultados.

Como se mencionó anteriormente, el análisis completo de una replicación implica el análisis de cada una de las variables de respuesta para las cuales se quiere conocer su efectividad: InScope (efectividad sobre los defectos que están dentro del alcance de la técnica), OutScope (efectividad sobre los defectos que se encuentran fuera del alcance de la técnica) y AllFaults (efectividad sobre todos los defectos sembrados).

Este capítulo se organiza de la siguiente forma: en la sección 5.1 se detallan los resultados del análisis de la variable de respuesta InScope, en la sección 5.2 los resultados de la variable OutScope y en las sección 5.3 los resultados de la variable AllFaults. La sección 5.4 se discute sobre los resultados globales de la replicación en relación a todas las variables de respuesta.

5.1. Variable de Respuesta: InScope

La variable de respuesta (VR) *InScope* corresponde a la cantidad de defectos que detectan los casos de prueba (generados por el sujeto aplicando la técnica de verificación), de aquellos tipificados como visibles para la técnica. El método de cálculo se explicita en la sección 4.2.2. El análisis para esta VR se divide en 3 partes, analizando el factor Técnica, Programa/Sesión y Grupo respectivamente.

5.1.1. Estadísticas Descriptivas

En el cuadro 5.1 se presentan medidas de cantidad de observaciones, media y desviación estándar de los datos (en valor absoluto).

Factor	Nivel	# de Obs.	Media	Desv. Estándar
Técnica	LAS	46	53,62	23,02
	CD	46	66,67	28,11
	PCE	46	79,71	27,65
Programa/Sesión	cmdline/S1	46	56,52	31,33
	nametbl/S3	46	68,84	24,25
	ntree/S2	46	74,64	26,23
Grupo	LAS-CD-PCE	24	71,53	29,27
	LAS-PCE-CD	24	59,03	32,96
	CD-LAS-PCE	27	75,31	25,89
	CD-PCE-LAS	27	57,41	26,69
	PCE-LAS-CD	18	55,55	24,26
	PCE-CD-LAS	18	82,41	18,50

Cuadro 5.1: Estadísticas Descriptivas - VR: InScope - Replicación 2006

Con respecto al factor Técnica, observando la figura 5.1 se ve una tendencia en donde LAS resulta ser la menos efectiva (53,62 % de efectividad en promedio), seguida de CD (66,67 %) y por último PCE (79,71 %). En términos de variabilidad (desviación estándar) se observa que LAS sería la más estable (23,02 %), seguida de PCE (27,65 %) y por último CD (28,11 %). Resta ver en los resultados de las pruebas de hipótesis si la tendencia observada resulta ser significativa o no.

Respecto del factor Programa/Sesión, la figura 5.2 muestra que la tendencia que se observa es: ntree/S2 con mayor efectividad (74,64 %), seguido de nametbl/S3 (68,94 %) y por último cmdline/S1 (56,52 %). En la figura se han puesto los programas ordenados respecto de a qué sesión del experimento corresponden (sesión 1 = cmdline, sesión 2 = ntree y sesión 3 = nametbl). Algo interesante a observar en este gráfico es que, suponiendo que hubiera un efecto de aprendizaje por la práctica, el programa en que se obtiene mayor efectividad no resulta ser el correspondiente a la sesión 3 (nametbl) sino que la efectividad en promedio baja respecto a la sesión 2 (ntree), en donde se obtiene el pico de efectividad.

Resulta difícil diferenciar qué efecto es el que está influenciando la efectividad, mirando solamente una replicación, ya que ambos efectos actúan sobre la misma. Debido a ésto, este fenómeno será contrastado nuevamente cuando se presenten los resultados de los análisis de todas las replicaciones, intentando diferenciar un efecto de otro, o buscando patrones de comportamiento.

Respecto del factor Grupo, el orden observado (de mayor a menor efectividad) en el gráfico de líneas que se muestra en la figura 5.3 es el siguiente: PCE-CD-LAS, CD-LAS-PCE, LAS-CD-PCE, LAS-PCE-CD, CD-PCE-LAS, PCE-LAS-CD. De dicho orden no se logra determinar ninguna tendencia que pueda ser explicada por el orden de ejecución

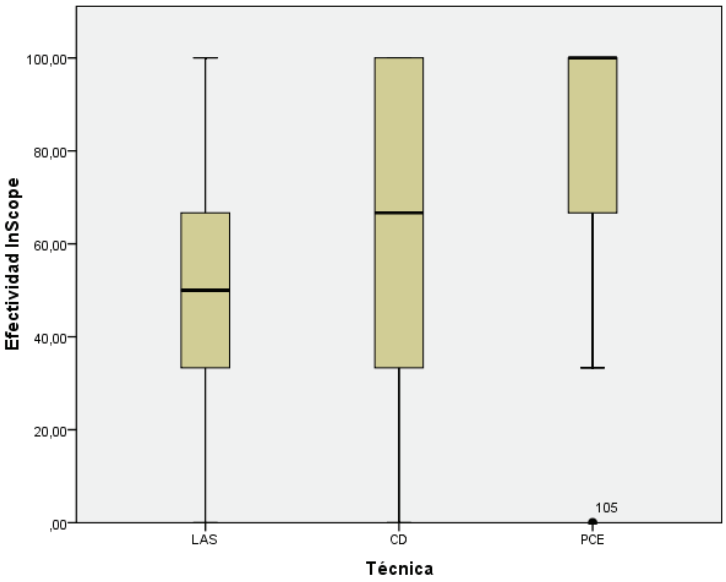

Figura 5.1: Gráfico de estadísticos descriptivos para el factor Técnica - VR: InScope - Replicación 2006

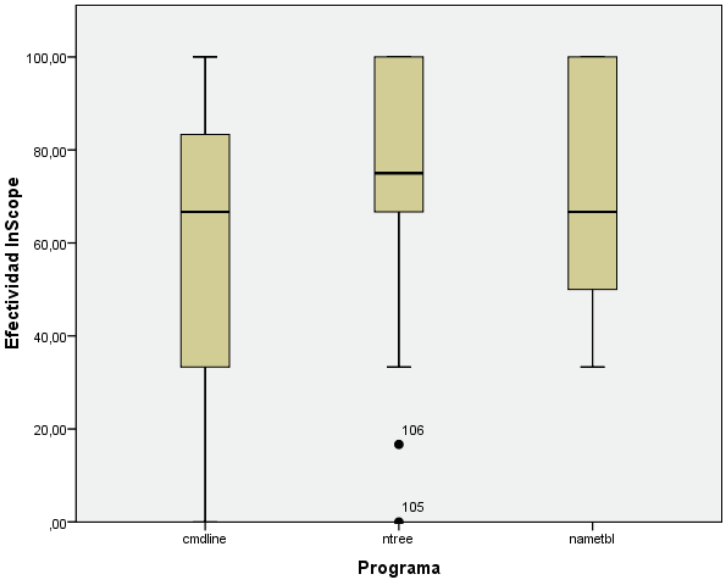

Figura 5.2: Gráfico de estadísticos descriptivos para el factor Programa/Sesión - VR: InScope - Replicación 2006

de las técnicas, por ejemplo: el grupo con mayor efectividad (PCE-CD-LAS) tiene como primer técnica a PCE y a su vez el grupo con menor efectividad también (PCE-LAS-CD), tampoco resultan ser los más efectivos aquellos grupos que tienen como primeras técnicas las dinámicas (PCE y CD) ya que los dos grupos con menos efectividad están

precedidos por ésta. Observando el orden de a 2 técnicas tampoco se logra determinar ningún tipo de patrón. En resumen, no parecería haber ninguna causa que explique las relaciones de orden observadas.

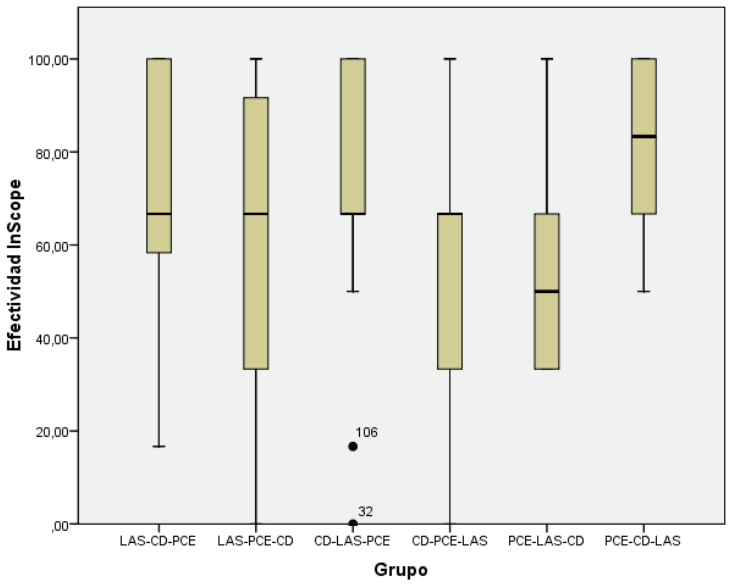

Figura 5.3: Gráfico de estadísticos descriptivos para el factor Grupo - VR: InScope - Replicación 2006

5.1.2. Reducción del Conjunto de Datos

La reducción de datos refiere a un análisis de los datos observados, en donde se determina si existen una o varias observaciones que deberían quitarse del conjunto de datos a analizar por presentar características que distorsionen el análisis posterior. El análisis debe determinar qué observaciones del conjunto de observaciones atípicas (outliers) debe quitarse del conjunto de datos, a modo de evitar deducciones incorrectas.

Como se mencionó anteriormente, no todos los outliers deben ser quitados del conjunto de datos, ya que no siempre son causados por malas mediciones o eventos extraños. En esos casos, estas observaciones contienen información importante que debe ser tenida en cuenta.

Para el caso de la replicación base, se identificaron 4 outliers y su análisis determinó que su origen es confiable ya que no se detectó ninguna anomalía que refiera al proceso de medición, eventos atípicos o variables indeseadas. Por tanto, no se quita ninguna observación y se toma el conjunto de datos original para el análisis.

5.1.3. Pruebas de Hipótesis

En esta sección se estudian los efectos principales de cada factor, utilizando el *Análisis de Componentes de la Varianza* aplicado a modelos mixtos. Se presentan los resultados obtenidos para cada uno de los efectos fijos (Técnica, Programa/Sesión y Grupo) y las comparaciones por pares dentro de cada factor.

En primer lugar, se chequea la **validez del modelo** a través de una prueba de normalidad para los residuos resultantes de la aplicación del análisis. En el cuadro 5.2 se observan los resultados de las pruebas de normalidad de Kolmogorov-Smirnov y de Shapiro-Wilk, en donde para ambas pruebas se acepta la hipótesis de normalidad con una significancia mayor a 0,05 (0,069 para K-S y 0,111 para S-W).

Pruebas de normalidad

	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
Residuos	,073	138	,069	,984	138	,111

a. Corrección de la significación de Lilliefors

Cuadro 5.2: Prueba de normalidad para efectos residuales - VR: InScope - Replicación 2006

La normalidad de los residuos también se puede apreciar en los gráficos de tipo histograma y comparando los valores residuales observados con la normal esperada. Estos gráficos pueden visualizarse en la figura 5.4. En el gráfico de histograma (izquierda) se aprecia que la distribución se parece a una campana de gauss, que representa una distribución normal. En el gráfico de los valores observados, se aprecia que los valores se ajustan a la normal esperada.

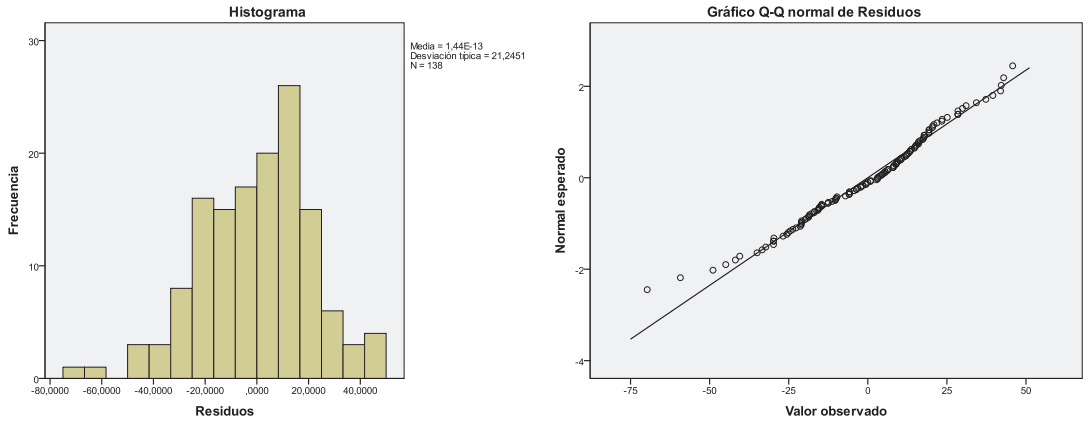

Figura 5.4: Gráficos de histograma y valores observados para los residuos - VR: InScope - Replicación 2006

Habiendo validado el modelo, se analizan los resultados de las pruebas de hipótesis para los efectos fijos. En el cuadro 5.3 se presentan los resultados de las pruebas de

efectos fijos para cada factor. De las significancias obervadas, se deduce que se rechaza la hipótesis nula (de que las medias son iguales) para todos los factores: Técnica, Programa/Sesión y Grupo, ya que las significancias obtenidas (columna Sig.) en las pruebas de efectos fijos no superan el 0,05 %: Técnica (0,000), Programa/Sesión (0,003) y Grupo (0,010).

Origen	Numerador df	Denominador df	Valor F	Sig.
Intersección	1	40,244	852,256	,000
Grupo	5	39,964	3,491	,010
Programa	2	87,853	6,158	,003
Tecnica	2	61,138	14,062	,000

Variable dependiente: Efectividad InScope.

Cuadro 5.3: Prueba de hipótesis para efectos fijos - VR: InScope - Replicación 2006

A continuación se comparan los niveles dentro de cada factor (comparaciones por parejas) para analizar qué combinación de niveles son aquellos que tienen diferencias significativas.

5.1.3.1. Comparaciones múltiples: Factor Técnica

En el cuadro 5.4 se presentan las estimaciones de media, error típico e intervalo de confianza para el factor Técnica.

Técnica	Media	Error típico	gl	Intervalo de confianza 95%	
				Límite inferior	Límite superior
LAS	54,037	3,442	40,673	47,085	60,989
CD	67,676	3,691	44,420	60,239	75,113
PCE	78,932	3,639	42,168	71,589	86,275

Variable dependiente: Efectividad InScope.

Cuadro 5.4: Estimaciones - Factor Técnica - VR: InScope - Replicación 2006

Las pruebas de comparaciones múltiples toman de a pares los niveles de cada factor (el producto cartesiano) y analizan si la diferencia que existe entre cada par es significativa o no. En el caso del factor técnica, de las tres comparaciones que se realizan, solamente 2 de ellas resultan ser significativas: LAS vs. CD Y LAS vs. PCE (cuadro 5.5). Existe evidencia estadística para afirmar que LAS (con 54,037 % de media estimada) es menos efectiva que CD (67,676 %) y que PCE (78,932 %), entre PCE y CD no existe diferencia significativa, esto es: $LSA < (CD = PCE)$. Esta observación es

interesante ya que entre las técnicas dinámicas no existe diferencia, pero sí entre las dinámicas y la estática, ambas más efectivas que esta última.

(I) Técnica	(J) Técnica	Diferencia entre las medias (I-J)	Error típico	gl	Sig.[c]	Intervalo de confianza al 95% para la diferencia[c]	
						Límite inferior	Límite superior
LAS	CD	-13,639*	4,748	60,873	,017	-25,329	-1,950
	PCE	-24,896*	4,715	63,315	,000	-36,491	-13,301
CD	LAS	13,639*	4,748	60,873	,017	1,950	25,329
	PCE	-11,256	4,911	60,872	,076	-23,347	,834
PCE	LAS	24,896*	4,715	63,315	,000	13,301	36,491
	CD	11,256	4,911	60,872	,076	-,834	23,347

Basado en las medias marginales estimadas

La diferencia entre las medias es significativa al nivel ,05.

Variable dependiente: Efectividad InScope.

Corrección por comparaciones múltiples: Bonferroni.

Cuadro 5.5: Comparaciones por parejas - Factor Técnica - VR: InScope - Replicación 2006

5.1.3.2. Comparaciones múltiples: Factor Programa/Sesión

Para este factor, no solamente hay que tener en cuenta la influencia del programa como tal, ya que cada programa corresponde a su vez a una sesión del experimento. Por tanto, ambos efectos se encuentran "mezclados". Como el orden de los programas varía de una replicación a otra, intentaremos diferenciar un efecto de otro en un análisis más global que se realiza en el capítulo 6.

En el cuadro 5.6 se presentan las estimaciones de media, error típico e intervalo de confianza para el factor Programa/Sesión.

Programa	Media	Error típico	gl	Intervalo de confianza 95%	
				Límite inferior	Límite superior
cmdline	57,750	3,597	122,878	50,631	64,869
nametbl	68,555	3,586	122,854	61,457	75,654
ntree	74,339	3,586	122,854	67,240	81,438

Variable dependiente: Efectividad InScope.

Cuadro 5.6: Estimaciones - Factor Programa/Sesión - VR: InScope - Replicación 2006

Los resultados de las pruebas de comparaciones múltiples (cuadro 5.7) confirman solamente diferencias significativas entre cmdline/S1 (57,750 %) y ntree/S2 (74,339 %). Entre cmdline/S1 (57,750 %) y nametbl/S3 así como entre nametbl/S3 y ntree/S2 las

diferencias resultan no significativas, esto es $cmdline/S1 = nametbl/S3$, $nametbl/S3 = ntree/S2$ y $cmdline/S1 < ntree/S2$.

(I) Programa	(J) Programa	Diferencia entre las medias (I-J)	Error típico	gl	Sig.[c]	Intervalo de confianza al 95% para la diferencia[c]	
						Límite inferior	Límite superior
cmdline	nametbl	-10,805	4,795	87,772	,080	-22,507	,897
	ntree	-16,589*	4,795	87,772	,003	-28,291	-4,887
nametbl	cmdline	10,805	4,795	87,772	,080	-,897	22,507
	ntree	-5,784	4,771	87,569	,686	-17,428	5,861
ntree	cmdline	16,589*	4,795	87,772	,003	4,887	28,291
	nametbl	5,784	4,771	87,569	,686	-5,861	17,428

Basado en las medias marginales estimadas

La diferencia entre las medias es significativa al nivel ,05.

Variable dependiente: Efectividad InScope.

Corrección por comparaciones múltiples: Bonferroni.

Cuadro 5.7: Comparaciones por parejas - Factor Programa/Sesión - VR: InScope - Replicación 2006

5.1.3.3. Comparaciones múltiples: Factor Grupo

En el cuadro 5.8 se presentan las estimaciones de media, error típico e intervalo de confianza para el factor Grupo.

Grupo	Media	Error típico	gl	Intervalo de confianza 95%	
				Límite inferior	Límite superior
LAS-CD-PCE	71,253	5,409	40,038	60,321	82,186
LAS-PCE-CD	59,382	5,409	40,038	48,449	70,314
CD-LAS-PCE	74,988	5,100	40,055	64,680	85,296
CD-PCE-LAS	57,571	5,100	40,055	47,263	67,879
PCE-LAS-CD	55,967	6,245	39,997	43,345	68,589
PCE-CD-LAS	82,129	6,245	39,997	69,506	94,751

Variable dependiente: Efectividad InScope.

Cuadro 5.8: Estimaciones - Factor Grupo - VR: InScope - Replicación 2006

De acuerdo a las comparaciones múltiples que se presentan en el cuadro 5.9, ninguna resultó ser significativa. Esto contradice un poco el resultado obtenido en las pruebas de efectos fijos (cuadro 5.3, que se presentó anteriormente), en donde el Grupo aparece significativo con un valor de 0,010. Esto algunas veces sucede porque el tipo de prueba que se realiza para efectos fijos resulta ser más (o menos) estricta que la utilizada para las comparaciones por pares. En este caso, las pruebas de efectos fijos utilizan la

prueba de Tukey y en las comparaciones múltiples se utiliza la prueba de Bonferroni. Esto sucede por una limitación de la herramienta, en la cual no es posible escojer la prueba de Bonferroni para evaluar los efectos fijos, que es más estricta que Tukey.

Sería deseable utilizar otra herramienta para corroborar los resultados (como podría ser R[1]) en la cual se pudiera escoger qué tipos de pruebas se desea utilizar para la evaluación de cada tipo de efecto. Sin embargo, debido al esfuerzo que requiere tanto el aprendizaje como la ejecución de las pruebas, queda fuera del alcance de este trabajo.

(I) Grupo	(J) Grupo	Diferencia entre las medias (I-J)	Error típico	gl	Sig.[b]	Intervalo de confianza al 95% para la diferencia[b]	
						Límite inferior	Límite superior
LAS-CD-PCE	LAS-PCE-CD	11,871	7,646	39,875	1,000	-12,002	35,745
	CD-LAS-PCE	-3,735	7,434	40,038	1,000	-26,942	19,472
	CD-PCE-LAS	13,682	7,434	40,008	1,000	-9,524	36,888
	PCE-LAS-CD	15,286	8,261	39,982	1,000	-10,505	41,077
	PCE-CD-LAS	-10,875	8,261	39,944	1,000	-36,666	14,915
LAS-PCE-CD	LAS-CD-PCE	-11,871	7,646	39,875	1,000	-35,745	12,002
	CD-LAS-PCE	-15,606	7,434	40,008	,632	-38,813	7,600
	CD-PCE-LAS	1,811	7,434	40,038	1,000	-21,397	25,018
	PCE-LAS-CD	3,415	8,261	39,944	1,000	-22,375	29,205
	PCE-CD-LAS	-22,747	8,261	39,982	,132	-48,538	3,044
CD-LAS-PCE	LAS-CD-PCE	3,735	7,434	40,038	1,000	-19,472	26,942
	LAS-PCE-CD	15,606	7,434	40,008	,632	-7,600	38,813
	CD-PCE-LAS	17,417	7,211	39,964	,306	-5,096	39,930
	PCE-LAS-CD	19,021	8,059	39,874	,349	-6,144	44,186
	PCE-CD-LAS	-7,141	8,062	39,987	1,000	-32,310	18,029
CD-PCE-LAS	LAS-CD-PCE	-13,682	7,434	40,008	1,000	-36,888	9,524
	LAS-PCE-CD	-1,811	7,434	40,038	1,000	-25,018	21,397
	CD-LAS-PCE	-17,417	7,211	39,964	,306	-39,930	5,096
	PCE-LAS-CD	1,604	8,062	39,987	1,000	-23,566	26,774
	PCE-CD-LAS	-24,557	8,059	39,874	,061	-49,722	,607
PCE-LAS-CD	LAS-CD-PCE	-15,286	8,261	39,982	1,000	-41,077	10,505
	LAS-PCE-CD	-3,415	8,261	39,944	1,000	-29,205	22,375
	CD-LAS-PCE	-19,021	8,059	39,874	,349	-44,186	6,144
	CD-PCE-LAS	-1,604	8,062	39,987	1,000	-26,774	23,566
	PCE-CD-LAS	-26,162	8,832	39,990	,077	-53,734	1,410
PCE-CD-LAS	LAS-CD-PCE	10,875	8,261	39,944	1,000	-14,915	36,666
	LAS-PCE-CD	22,747	8,261	39,982	,132	-3,044	48,538
	CD-LAS-PCE	7,141	8,062	39,987	1,000	-18,029	32,310
	CD-PCE-LAS	24,557	8,059	39,874	,061	-,607	49,722
	PCE-LAS-CD	26,162	8,832	39,990	,077	-1,410	53,734

Basado en las medias marginales estimadas

Variable dependiente: Efectividad InScope.

Corrección por comparaciones múltiples: Bonferroni.

Cuadro 5.9: Comparaciones por parejas - Factor Grupo - VR: InScope - Replicación 2006

[1]http://www.r-project.org/

5.2. Variable de Respuesta: OutScope

Esta VR corresponde a la cantidad de defectos que detecta el sujeto, de aquellos que no son visibles para la técnica que está aplicando (o que están fuera de su alcance) y se corresponde con la variable *OutScope* que se describió en la sección 4.2.2.

Para este análisis, se deja fuera a la técnica de revisión (LAS) ya que en teoría, para ésta técnica, todos los defectos inyectados son visibles (o detectables). Por tanto, carece de sentido este tipo de análisis.

5.2.1. Estadísticas Descriptivas

En el cuadro 5.10 se presentan medidas de cantidad de observaciones, media y desviación estándar de los datos.

Factor	Nivel	# de Obs.	Media	Desv. Estándar
Técnica	CD	46	28,26	27,19
	PCE	46	14,49	19,44
Programa/Sesión	cmdline/S1	30	15,55	20,96
	nametbl/S3	31	29,03	29,49
	ntree/S2	31	19,35	20,68
Grupo	LAS-CD-PCE	16	14,58	17,08
	LAS-PCE-CD	16	35,42	25,73
	CD-LAS-PCE	18	14,81	20,52
	CD-PCE-LAS	18	20,37	25,92
	PCE-LAS-CD	12	36,11	30,01
	PCE-CD-LAS	12	8,33	15,07

Cuadro 5.10: Estadísticas Descriptivas - VR:OutScope - Replicación 2006

Comparando los estadísticos descriptivos de la variable OutScope con los de InScope (cuadro 5.1), se puede apreciar que en promedio, todos los niveles de todos los factores presentan medidas de efectividad más pequeños. Esto es de esperarse ya que se supone que las técnicas van a comportarse de forma más efectiva con aquellos defectos que están dentro de su alcance a con aquellos que están fuera.

Con respecto al factor Técnica, en la figura 5.5 se ve una tendencia en donde CD (28,26 %) parecería ser más efectiva que PCE (14,49 %). Esta tendencia es inversa a la observada en la variable InScope, en donde CD presentaba menor efectividad que PCE en promedio, a pesar de que esta diferencia resultó no significativa en las pruebas de hipótesis.

Respecto del factor Programa/Sesión, la tendencia que se observa en la figura 5.6 es: nametbl/S3 con mayor efectividad (29,03 %), seguido de ntree/S2 (19,35 %) y por último cmdline/S1 (15,55 %). De acuerdo al gráfico de la figura 5.6 se aprecia que para esta VR, si tomamos los programas como las sesiones, la efectividad aumenta de una sesión a otra, esto podría estar indicando un efecto de aprendizaje por la práctica, así

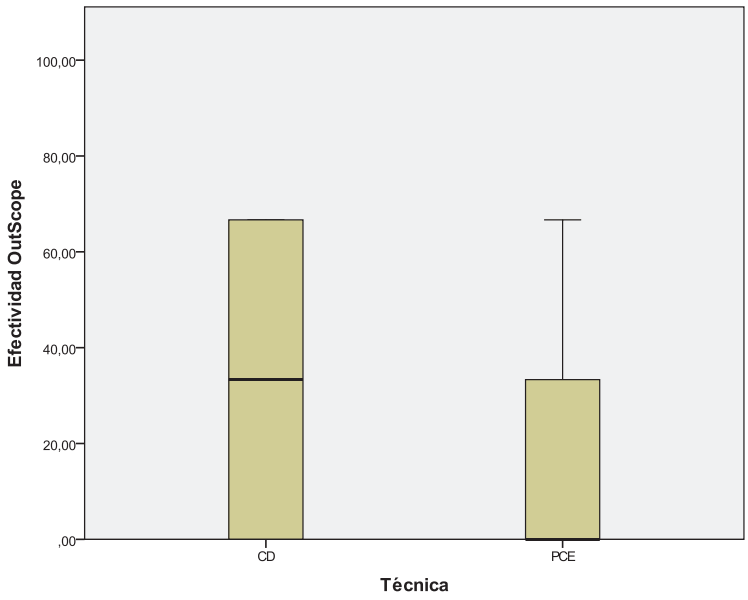

Figura 5.5: Gráfico de estadísticos descriptivos para el factor Técnica - VR: OutScope - Replicación 2006

como también descartando un efecto de cansancio (o que la influencia de uno es mayor que la del otro). En cualquier caso, hay que ver si el resto de los análisis de las otras replicaciones coinciden o no con estos resultados.

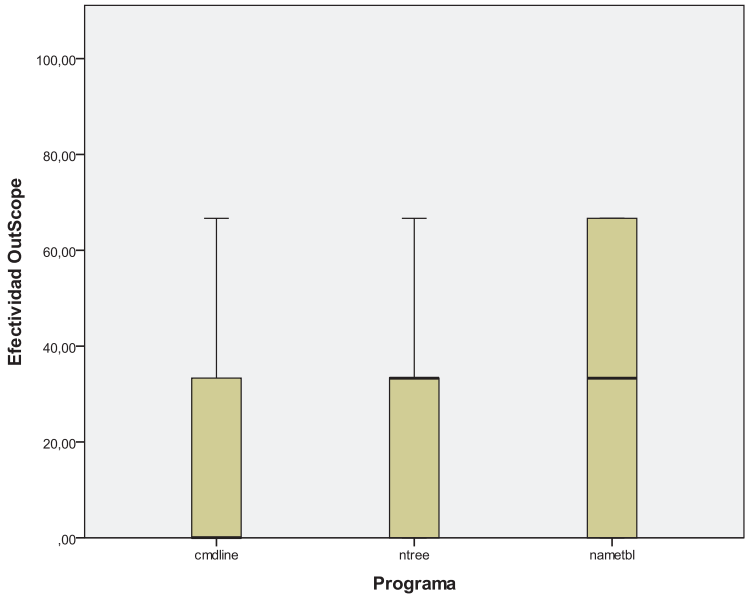

Figura 5.6: Gráfico de estadísticos descriptivos para el factor Programa/Sesión - VR: OutScope - Replicación 2006

Respecto del grupo, el orden observado (figura 5.7) es el siguiente: PCE-LAS-CD, LAS-PCE-CD, CD-PCE-LAS, CD-LAS-PCE, LAS-CD-PCE , PCE-CD-LAS. De dicho orden no se logra observar ninguna tendencia ni patrón.

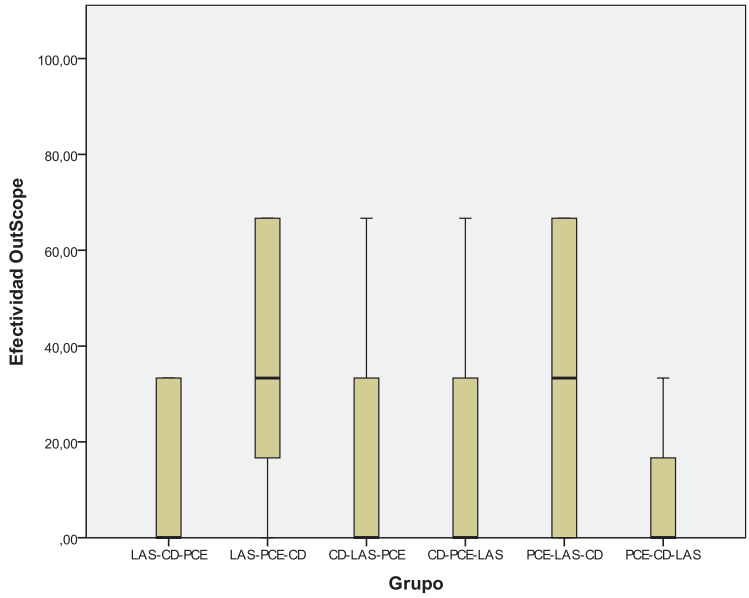

Figura 5.7: Gráfico de estadísticos descriptivos para el factor Grupo - VR: OutScope - Replicación 2006

5.2.2. Pruebas de Hipótesis

Análogamente a la sección 5.1.3 en donde se estudian las pruebas de hipótesis para la VR InScope, lo primero que se realiza es la validación del modelo a través del chequeo de la normalidad en los valores residuales.

Las pruebas de normalidad de Kolmogorov-Smirnov y de Shapiro-Wilk que se muestran en el cuadro 5.11 indican que se rechaza la hipótesis de normalidad para los residuos, obteniéndose valores de significancia menores a 0,05 (0,000 para K-S y S-W). Los gráficos de los residuos que se presentan en la figura 5.8 también presentan la falta de normalidad, no pudiéndose apreciar una campana en el gráfico de la izquierda, y no ajustándose las observaciones a la estimación normal en el gráfico de la derecha.

Para obtener un modelo que se ajuste mejor al tipo de análisis se aplica un conjunto de 9 transformaciones a la variable respuesta, chequeando si con alguna de éstas se consigue distribución normal para los residuos, como se describió en la sección 4.4.1. En el cuadro 5.12 se muestra la significancia obtenida en la prueba de normalidad de los residuos, para cada transformación realizada. Debido a que ninguna transformación logra obtener normalidad de residuos (todas las significancias de la prueba de normalidad son inferiores a 0,01), se opta por tomar los resultados del análisis de la variable respuesta original (sin transformar). Los resultados de estas pruebas de hipótesis contienen una amenaza a la validez agregada.

Pruebas de normalidad

	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
Residuos	,197	92	,000	,922	92	,000

a. Corrección de la significación de Lilliefors

Cuadro 5.11: Prueba de normalidad para efectos residuales - VR: OutScope - Replicación 2006

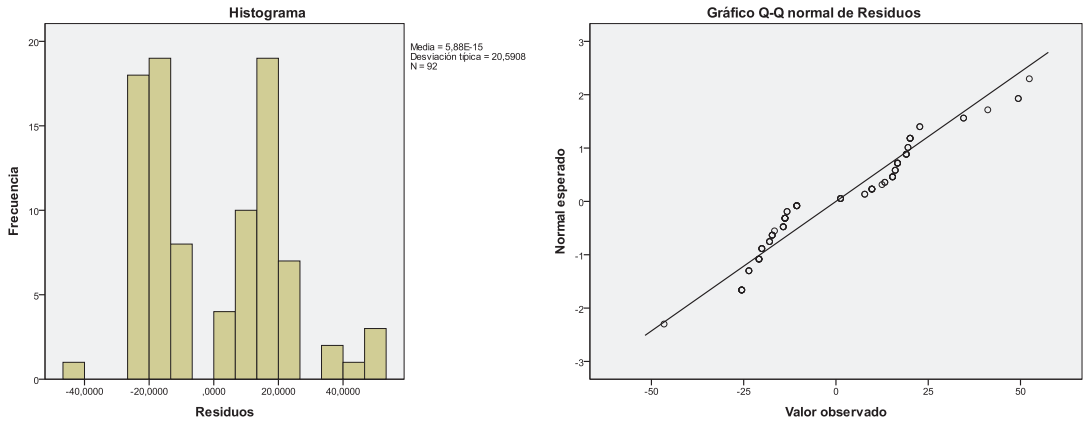

Figura 5.8: Gráficos de histograma y valores observados para los residuos - VR: OutScope - Replicación 2006

Transformación	**Kolmogorov-Smirnov**	**Shapiro-Wilk**
y^2	0,005	0,008
$\sqrt{y}$	0,000	0,000
$\ln(y)$	0,000	0,002
$\ln(y+\frac{1}{2})$	0,000	0,000
$\log_{10}(y)$	0,000	0,000
$\log_{10}(y+\frac{1}{2})$	0,000	0,000
$\frac{1}{y}$	0,000	0,002
$\frac{1}{\sqrt{y}}$	0,000	0,002
$\exp y$	0,000	0,000

Cuadro 5.12: Resultados de pruebas de Normalidad para las transformaciones realizadas - VR: OutScope - Replicación 2006

En relación a los efectos fijos, los resultados que se presentan en el cuadro 5.13 muestran que el único factor que supera el nivel de significancia del 0,05 % es el Programa/Sesión, por lo tanto se rechaza la hipótesis nula solamente para la Técnica y el Grupo.

A continuación se presentan los resultados de las comparaciones múltiples única-

Origen	Numerador df	Denominador df	Valor F	Sig.
Intersección	1	81,791	89,877	,000
Grupo	5	81,024	3,318	,009
Programa	2	81,778	2,608	,080
Tecnica	1	81,778	11,108	,001

Variable dependiente: Efectividad Out Scope.

Cuadro 5.13: Prueba de hipótesis para efectos fijos - VR:OutScope - Replicación 2006

mente para aquellos factores que presentaron diferencias significativas.

5.2.2.1. Comparaciones múltiples: Factor Técnica

En el cuadro 5.14 se presentan las estimaciones para el factor Técnica.

Técnica	Media	Error típico	gl	Intervalo de confianza 95%	
				Límite inferior	Límite superior
CD	29,205	3,390	34,627	22,321	36,090
PCE	14,026	3,046	34,316	7,838	20,214

Variable dependiente: Efectividad Out Scope.

Cuadro 5.14: Estimaciones - Factor Técnica - VR: OutScope - Replicación 2006

De la única comparación que se realiza (por considerar únicamente 2 niveles) para el factor técnica (cuadro 5.15), existe evidencia estadística para afirmar que PCE (14,026 %) es menos efectiva que CD (29,205 %). Dados que los niveles de eficiencia son bajos, no se puede considerar a ninguna técnica como "efectiva".

5.2.2.2. Comparaciones múltiples: Factor Grupo

En el cuadro 5.16 se presentan las estimaciones generadas para el factor Grupo.

De las comparaciones por pares realizadas (cuadro 5.17), ninguna resultó ser significativa, a diferencia de las pruebas de efectos fijos. Ocurre el mismo caso que la VR InScope (sección 5.1.3.3), en donde las pruebas de efectos fijos resultan menos sensibles que las de comparaciones múltiples.

[a]

(I) Técnica	(J) Técnica	Diferencia entre las medias (I-J)	Error típico	gl	Sig.[c]	Intervalo de confianza al 95% para la diferencia[c]	
						Límite inferior	Límite superior
CD	PCE	15,180*	4,555	81,778	,001	6,119	24,240
PCE	CD	-15,180*	4,555	81,778	,001	-24,240	-6,119

Basado en las medias marginales estimadas

La diferencia entre las medias es significativa al nivel .05.

Variable dependiente: Efectividad Out Scope.

Corrección por comparaciones múltiples: Bonferroni.

Cuadro 5.15: Comparaciones por parejas - Factor Técnica - VR: OutScope - Replicación 2006

Grupo	Media	Error típico	gl	Intervalo de confianza 95%	
				Límite inferior	Límite superior
LAS-CD-PCE	10,704	5,681	82,889	-,595	22,002
LAS-PCE-CD	31,481	5,681	82,889	20,183	42,780
CD-LAS-PCE	13,900	5,391	82,927	3,177	24,623
CD-PCE-LAS	25,157	5,391	82,927	14,434	35,880
PCE-LAS-CD	34,890	6,477	82,945	22,008	47,772
PCE-CD-LAS	13,561	6,477	82,945	,679	26,443

Variable dependiente: Efectividad Out Scope.

Cuadro 5.16: Estimaciones - Factor Grupo - VR: OutScope - Replicación 2006

5.3. Variable de Respuesta: AllFaults

Para el cálculo de esta VR se toma la cantidad de defectos que detecta el sujeto, de aquellos que son visibles para la técnica que está aplicando más aquellos no visibles, en definitiva, para todos los defectos sembrados. Se corresponde con la variable *AllFaults* que se describe en la sección 4.2.2.

5.3.1. Estadísticas Descriptivas

En el cuadro 5.18 se presentan medidas de cantidad de observaciones, media y desviación estándar de los datos.

Con respecto al factor técnica, la figura 5.9 muestra una tendencia en donde PCE resulta ser la menos efectiva (47,10 % de efectividad en promedio), seguida de CD (47,46 %) y por último PCE (53,62 %). En términos de variabilidad (desviación estándar) se observa que CD sería la más estable (16,84 %), seguida de PCE (17,68 %) y por último LAS (23,02 %). En términos generales, los resultados muestran similaridad y estabilidad entre las técnicas.

(I) Grupo	(J) Grupo	Diferencia entre las medias (I-J)	Error típico	gl	Sig.[b]	Intervalo de confianza al 95% para la diferencia[b]	
						Límite inferior	Límite superior
LAS-CD-PCE	LAS-PCE-CD	-20,778	7,607	82,854	,116	-43,772	2,217
	CD-LAS-PCE	-3,196	7,909	69,162	1,000	-27,243	20,851
	CD-PCE-LAS	-14,454	8,055	82,995	1,000	-38,800	9,893
	PCE-LAS-CD	-24,186	8,825	82,975	,113	-50,859	2,486
	PCE-CD-LAS	-2,858	8,931	78,157	1,000	-29,901	24,185
LAS-PCE-CD	LAS-CD-PCE	20,778	7,607	82,854	,116	-2,217	43,772
	CD-LAS-PCE	17,582	8,055	82,995	,478	-6,765	41,928
	CD-PCE-LAS	6,324	7,909	69,162	1,000	-17,723	30,371
	PCE-LAS-CD	-3,409	8,931	78,157	1,000	-30,452	23,634
	PCE-CD-LAS	17,920	8,825	82,975	,682	-8,753	44,593
CD-LAS-PCE	LAS-CD-PCE	3,196	7,909	69,162	1,000	-20,851	27,243
	LAS-PCE-CD	-17,582	8,055	82,995	,478	-41,928	6,765
	CD-PCE-LAS	-11,258	7,947	75,105	1,000	-35,352	12,836
	PCE-LAS-CD	-20,991	8,017	82,909	,158	-45,224	3,242
	PCE-CD-LAS	,338	8,605	82,989	1,000	-25,671	26,348
CD-PCE-LAS	LAS-CD-PCE	14,454	8,055	82,995	1,000	-9,893	38,800
	LAS-PCE-CD	-6,324	7,909	69,162	1,000	-30,371	17,723
	CD-LAS-PCE	11,258	7,947	75,105	1,000	-12,836	35,352
	PCE-LAS-CD	-9,733	8,605	82,989	1,000	-35,742	16,277
	PCE-CD-LAS	11,596	8,017	82,909	1,000	-12,637	35,829
PCE-LAS-CD	LAS-CD-PCE	24,186	8,825	82,975	,113	-2,486	50,859
	LAS-PCE-CD	3,409	8,931	78,157	1,000	-23,634	30,452
	CD-LAS-PCE	20,991	8,017	82,909	,158	-3,242	45,224
	CD-PCE-LAS	9,733	8,605	82,989	1,000	-16,277	35,742
	PCE-CD-LAS	21,329	9,190	73,789	,346	-6,550	49,208
PCE-CD-LAS	LAS-CD-PCE	2,858	8,931	78,157	1,000	-24,185	29,901
	LAS-PCE-CD	-17,920	8,825	82,975	,682	-44,593	8,753
	CD-LAS-PCE	-,338	8,605	82,989	1,000	-26,348	25,671
	CD-PCE-LAS	-11,596	8,017	82,909	1,000	-35,829	12,637
	PCE-LAS-CD	-21,329	9,190	73,789	,346	-49,208	6,550

Basado en las medias marginales estimadas

Variable dependiente: Efectividad Out Scope.

Corrección por comparaciones múltiples: Bonferroni.

Cuadro 5.17: Comparaciones por parejas - Factor Grupo - VR: OutScope - Replicación 2006

Respecto del factor Programa/Sesión, la tendencia que se observa en la figura 5.10 es: ntree/S2 con mayor efectividad (74,64 %), seguido de nametbl/S3 (68,94 %) y por último cmdline/S1 (56,52 %).

En el gráfico de la figura 5.10 se han puesto los programas ordenados respecto de a qué sesión del experimento corresponden (sesión 1 = cmdline, sesión 2 = ntree y sesión 3 = nametbl). Algo interesante a observar en este gráfico es que, suponiendo que hubiera un efecto de aprendizaje por la práctica, el programa con mayor efectividad no es nametbl (correspondiente a la sesión 3) sino que la efectividad baja (en promedio) respecto a la sesión 2 (ntree), aunque muy poco. Tampoco explica un efecto de cansancio, ya que la efectividad aumenta en la Sesión 3 respecto de la Sesión 2.

Factor	Nivel	# de Obs.	Media	Desv. Estándar
Técnica	LAS	46	53,62	23,02
	CD	46	47,46	16,84
	PCE	46	47,10	17,68
Programa/Sesión	cmdline/S1	46	40,94	21,29
	nametbl/S3	46	53,26	13,89
	ntree/S2	46	53,98	19,93
Grupo	LAS-CD-PCE	24	47,92	19,85
	LAS-PCE-CD	24	48,61	23,53
	CD-LAS-PCE	27	53,09	17,93
	CD-PCE-LAS	27	43,83	19,69
	PCE-LAS-CD	18	50,00	21,39
	PCE-CD-LAS	18	54,63	11,15

Cuadro 5.18: Estadísticas Descriptivas - VR: AllFaults - Replicación 2006

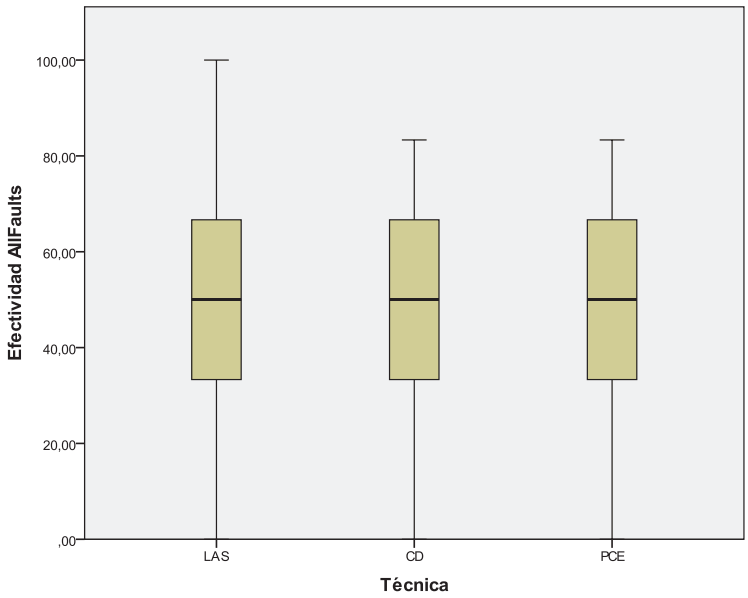

Figura 5.9: Gráfico de estadísticos descriptivos para el factor Técnica - VR: AllFaults - Replicación 2006

Respecto del grupo, en la figura 5.11 se observa el siguiente orden: PCE-CD-LAS, CD-LAS-PCE, LAS-CD-PCE, LAS-PCE-CD, CD-PCE-LAS, PCE-LAS-CD. De dicho orden no se logra observar ninguna tendencia que se explique debido al orden de ejecución de las técnicas, por ejemplo: el grupo con mayor efectividad (PCE-CD-LAS) tiene como primer técnica a PCE y a su vez el grupo con menor efectividad también (PCE-LAS-CD), tampoco resultan ser los más efectivos aquellos grupos que tienen como primeras técnicas las dinámicas (PCE y CD) ya que los dos grupos con menos efectividad están precedidos por ésta, tampoco observando el orden de a 2 técnicas se logra obtener ningún tipo de patrón. En resumen, no resulta claro determinar ningún

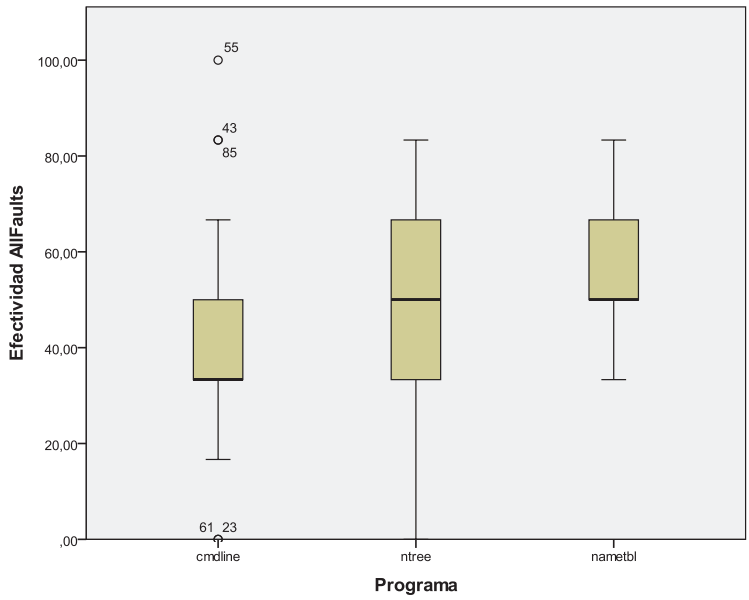

Figura 5.10: Gráfico de estadísticos descriptivos para el factor Programa/Sesión - VR: AllFaults - Replicación 2006

patrón que sustente el orden observado.

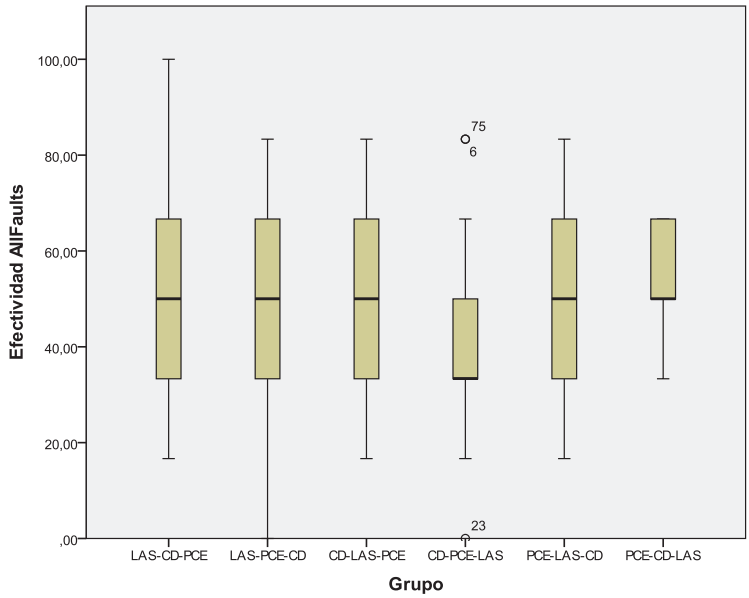

Figura 5.11: Gráfico de estadísticos descriptivos para el factor Grupo - VR: AllFaults - Replicación 2006

5.3.2. Pruebas de Hipótesis

Análogamente a las 2 variables de respuesta anteriores, se realizan pruebas de normalidad sobre los residuos para validar el modelo. Los resultados que se muestran en el cuadro 5.19 indican que no se rechaza la hipótesis de normalidad para los residuos, obteniéndose valores de significancia mayores a 0,01 (0,200 y 0,838 para K-S y S-W respectivamente). En los gráficos de los residuos que se presentan en la figura 5.12 también se puede observar el ajuste a la distribución normal.

Pruebas de normalidad

	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
Residuos	,037	138	,200*	,994	138	,838

a.Este es un límite inferior de la significación verdadera.

Corrección de la significación de Lilliefors

Cuadro 5.19: Prueba de normalidad para efectos residuales - VR: AllFaults - Replicación 2006

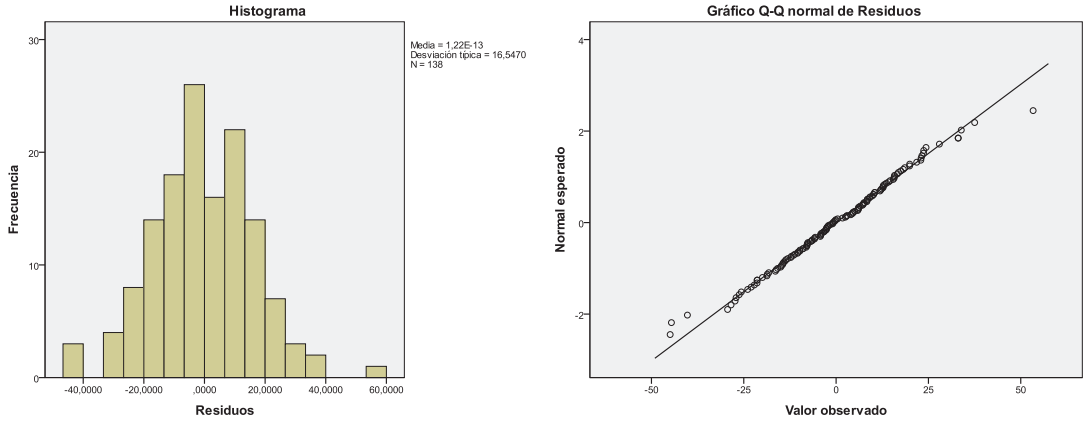

Figura 5.12: Gráficos de histograma y valores observados para los residuos - VR: AllFaults - Replicación 2006

Habiendo validado el modelo, se procede a efectuar el resto de las pruebas de hipótesis. En el cuadro 5.20 se presentan los resultados de las pruebas para los factores de efectos fijos.

El único factor que presenta diferencias significativas entre sus niveles es el factor Programa/Sesión con una significancia de 0,000. Para el resto de los factores no existe evidencia estadística como para afirmar una diferencia significativa. Por tanto, se asume igualdad de comportamiento entre los niveles del factor Técnica y del factor Grupo.

Origen	Numerador df	Denominador df	Valor F	Sig.
Intersección	1	42,753	823,101	,000
Grupo	5	40,853	,857	,518
Programa	2	85,969	8,502	,000
Tecnica	2	65,603	1,887	,160

Variable dependiente: Efectividad AllFaults.

Cuadro 5.20: Prueba de hipótesis para efectos fijos - VR: AllFaults - Replicación 2006

5.3.2.1. Comparaciones múltiples: Factor Programa/Sesión

En el cuadro 5.21 se presentan las estimaciones generadas para el factor Programa/Sesión.

Programa	Media	Error típico	gl	Intervalo de confianza 95%	
				Límite inferior	Límite superior
cmdline	41,034	2,725	122,292	35,639	46,429
nametbl	54,088	2,701	124,689	48,743	59,433
ntree	53,817	2,701	124,689	48,471	59,162

Variable dependiente: Efectividad AllFaults.

Cuadro 5.21: Estimaciones - Factor Programa/Sesión - VR: AllFaults - Replicación 2006

Para las comparaciones por pares realizadas, el cuadro 5.22 muestra que existe evidencia estadística para afirmar que la efectividad sobre cmdline/S1 (41,034 %) es menor que en nametbl/S3 (54,088 %) y ntree/S2 (53,817 %). Sin embargo, para la comparación de nametbl/S3 con ntree/S2 no se ven diferencias significativas.

5.4. Discusión del Análisis de la Replicación 2006

De acuerdo a los resultados de los análisis previamente presentados, se realiza una discusión englobando a todas las variables de respuesta, infiriendo posibles causas que luego serán contrastadas con los análisis de las replicaciones posteriores. Es importante recordar, que los resultados obtenidos con la variable OutScope fueron obtenidos sin validar el modelo completamente (no se obtuvo normalidad de residuos) por lo que deben ser tratador como resultados con una amenaza a la validez agregada.

Respecto del factor técnica, los resultados obtenidos para cada VR reflejan que las técnicas se comportan de forma bien distinta respecto de si los defectos están dentro o

(I) Programa	(J) Programa	Diferencia entre las medias (I-J)	Error típico	gl	Sig.[c]	Intervalo de confianza al 95% para la diferencia[c]	
						Límite inferior	Límite superior
cmdline	nametbl	-13,054*	3,614	86,136	,002	-21,878	-4,230
	ntree	-12,782*	3,614	86,136	,002	-21,607	-3,958
nametbl	cmdline	13,054*	3,614	86,136	,002	4,230	21,878
	ntree	,271	3,602	85,309	1,000	-8,525	9,068
ntree	cmdline	12,782*	3,614	86,136	,002	3,958	21,607
	nametbl	-,271	3,602	85,309	1,000	-9,068	8,525

Basado en las medias marginales estimadas

La diferencia entre las medias es significativa al nivel ,05.

Variable dependiente: Efectividad AllFaults.

Corrección por comparaciones múltiples: Bonferroni.

Cuadro 5.22: Comparaciones por parejas - Factor Programa/Sesión - VR: AllFaults - Replicación 2006

fuera de su alcance. El cuadro 5.23 muestra que ninguna tendencia ni efecto significativo es similar entre una VR y otra.

VR	**Efectos significativos**
InScope	LAS < CD LAS < PCE CD = PCE
OutScope	PCE < CD
AllFaults	PCE = CD = LAS

Cuadro 5.23: Factor Técnica - Todas las VR - Replicación 2006

Otra observación interesante es que para la VR OutScope se confirma el efecto significativo de que $PCE < CD$, sin embargo, la tendencia para la VR InScope es inversa, a pesar de no ser una diferencia significativa.

En el caso del factor Programa/Sesión, el cuadro 5.24 muestra que para las VR InScope y AllFaults, la tendencia resulta ser similar y la diferencia significativa de $cmdline/S1 < ntree/S2$ se comparte. Una posible causa de este efecto podrían ser que el programa cmdline resultase más complejo de probar (o que los otros dos programas fueran más fáciles o intuitivos).

El hecho de que cmdline fuera el programa correspondiente a la primer sesión, también podría ser una causa de la menor efectividad observada. Al ser los otros programas correspondientes a sesiones posteriores, podría haberse generado un efecto de aprendizaje por la práctica.

En cualquiera de los dos casos, es necesario ver si estos efectos y tendencias se repiten en otras replicaciones, sobre todo aquellas que tienen distinto orden de correspondencia programa-sesión.

VR	**Efectos significativos**
InScope	cmdline/S1 < ntree/S2 cmdline/S1 = nametbl/S3 nametbl/S3 = ntree/S2
OutScope	cmdline/S1 = nametbl/S3 = ntree/S2
AllFaults	cmdline/S1 < nametbl/S3 cmdline/S1 < ntree/S2 nametbl/S3 = ntree/S2

Cuadro 5.24: Factor Programa - Todas las VR - Replicación 2006

Para el caso del factor grupo, como se había observado previamente, el cuadro 5.25 muestra tendencias completamente distintas para cada variable, de donde no se puede deducir patrón alguno. Además de esto, para todas las pruebas de comparaciones por parejas hechas para las tres variables, ninguna resultó ser significativa. Esto lleva a suponer que el grupo no tiene influencia sobre la efectividad en ninguna de las variables.

VR	**Efectos significativos**
InScope	PCE-LAS-CD = CD-PCE-LAS = LAS-PCE-CD = LAS-CD-PCE = CD-LAS-PCE = PCE-CD-LAS
OutScope	PCE-LAS-CD = CD-PCE-LAS = LAS-PCE-CD = LAS-CD-PCE = CD-LAS-PCE = PCE-CD-LAS
AllFaults	PCE-LAS-CD = CD-PCE-LAS = LAS-PCE-CD = LAS-CD-PCE = CD-LAS-PCE = PCE-CD-LAS

Cuadro 5.25: Factor Grupo - Todas las VR - Replicación 2006

Capítulo 6

Análisis de las Replicaciones

En este capítulo se presenta un resumen del análisis de todas las replicaciones que se realizaron para el experimento, siguiendo el mismo procedimiento de análisis que se utilizó para el experimento base.

En el cuadro 6.1 se resumen las particularidades de cada replicación, como ser: sitio en que se lleva a cabo la ejecución, cantidad de sesiones que se realizan y las técnicas y programas que se utilizan por sesión. Por ejemplo: para las replicaciones de 2006 a 2008 se tiene un diseño Cross-over, por lo cual los sujetos aplican más de un nivel de factor por sesión, que en estos casos son los 3 niveles del factor técnica sobre 1 programa por cada sesión (completando los 3 programas en las 3 sesiones).

Replicación	**Sitio**	**# de Sesiones**	**Técnicas y Programas por Sesión**
2006	UPM	3	3 Técnicas 1 Programa
2007	UPM	3	3 Técnicas 1 Programa
2008	UPM	3	3 Técnicas 1 Programa
2011	UPM	2	2 Técnicas 1 Programa
2012	ESPEL	2	2 Técnicas 1 Programa

Cuadro 6.1: Descripción de cada replicación

Debido a que los diseños no son exactamente los mismos para todas las replicaciones, se introduce una amenaza a la validez en lo que refiere a las comparaciones que se realicen de los resultados y a las conclusiones que se obtengan a partir de ellos. De igual forma, creemos que las diferencias no son tan grandes como para que la comparación no se pueda realizar, aunque con ciertos cuidados.

Otro aspecto importante a considerar al momento de comparar las replicaciones es la correspondencia de los programas a verificar en cada sesión para cada replicación, esto

se muestra en el cuadro 6.2. Para las replicaciones de 2011 y 2012 notar que solamente se tienen 2 sesiones y 2 programas.

Replicación	Sesión 1	Sesión 2	Sesión 3
2006	cmdline	ntree	nametbl
2007	ntree	nametbl	cmdline
2008	ntree	cmdline	nametbl
2011	ntree	nametbl	-
2012	nametbl	ntree	-

Cuadro 6.2: Correspondencia de Programas a Sesiones en cada Replicación

Como se puede apreciar en el cuadro 6.2, la mayoría de las replicaciones presentan un orden distinto en el uso de cada programa en cada sesión. Debido a que el efecto del programa y el de la sesión están confundidos y/o combinados, en las comparaciones que se realizan entre los niveles de estos factores no es posible diferenciar si el resultado de la comparación se debe al efecto del programa, de la sesión, o de ambos factores combinados. Esta interacción programa-sesión, si bien teóricamente puede afectar, no parece razonable que lo haga, ya que no hemos podido identificar ninguna razón por la cual un programa pueda aumentar/disminuir la efectividad de una sesión en particular, o viceversa.

Un aspecto importante a considerar al interpretar los análisis es que la replicación de 2008 no tuvo un balanceo completo, respecto de los grupos conformados por el orden de aplicación de las técnicas. Si bien los sujetos utilizaron las 3 técnicas (una distinta en cada sesión), de los 6 grupos que se pueden conformar combinando las distintas técnicas en distintos órdenes, se tiene que para esta replicación se conformaron los grupos: LAS-CD-PCE, LAS-PCE-CD, CD-PCE-LAS y PCE-LAS-CD, faltando conformar los grupos de CD-LAS-PCE y PCE-CD-LAS. Esto fue debido a un error en la etapa previa de la ejecución del experimento, al momento de asignar los sujetos a los grupos.

Si bien es importante tener un balanceo completo respecto del factor grupo, ya que el mismo influye en los efectos de carry-over, creemos que la falta de esos 2 grupos que representan 2 de las 6 combinaciones de técnicas no impide realizar dicho análisis y comparar los resultados con las otras replicaciones. La técnica de análisis que se utiliza (Análisis de Componentes de la Varianza aplicado a modelos mixtos) es robusta respecto de modelos no equilibrados, como el caso de la replicación de 2008.

A continuación se presentan los resultados obtenidos de los análisis agrupados por cada variable de respuesta analizada y luego una discusión comparando todos los resultados.

6.1. Variable de Respuesta: InScope

En esta sección se realiza el análisis estadístico respecto de la variable InScope. En primer lugar se presentan las estadísticas descriptivas para cada factor, seguidas de las pruebas de hipótesis.

6.1.1. Estadísticas Descriptivas - InScope

Las estadísticas descriptivas para el factor técnica respecto de la variable InScope se muestran en el cuadro 6.3, los gráficos boxplot respecto de las medias se muestran en la figura 6.1. Se puede ver que la técnica LAS presenta más estabilidad respecto de la efectividad que el resto de las técnicas ya que su mayor desviación estándar se registra para 2006 en un 23,017 % y tanto CD como LAS superan esa variabilidad para todas las replicaciones. La menor efectividad presentada para CD y PCE resulta ser la replicación de 2012 con 31,883 % y 44,928 % de efectividad respectivamente. Los valores más bajos registrados corresponden a la replicación de ESPEL para todas las técnicas.

Técnica	Replicación	# de Obs.	Media	Desv. Estándar
LAS	2006	46	53,624	23,017
	2007	48	52,778	21,008
	2008	46	40,579	16,350
CD	2006	46	66,667	28,110
	2007	48	46,528	30,552
	2008	42	61,905	27,123
	2011	22	75,759	23,417
	2012	23	31,883	25,581
PCE	2006	46	79,710	27,648
	2007	48	62,502	24,433
	2008	46	63,044	31,607
	2011	22	77,273	29,790
	2012	23	44,928	29,490

Cuadro 6.3: Estadísticas Descriptivas - VR: InScope - Factor Técnica

En el cuadro 6.4 se muestran los estadísticos descriptivos del factor Programa/Sesión, agrupados desde el punto de vista del programa. En la figura 6.2 se presentan los gráficos boxplot para las medias. Se observa una muy baja efectividad para el programa ntree en la replicación de ESPEL 2012. Entre las replicaciones de 2006 a 2008 se aprecia más baja efectividad para el programa cmdline.

El análisis del factor Grupo resulta un poco más complicado que el de los otros factores, ya que los distintos niveles de este factor varían de acuerdo al diseño del experimento y cómo se implementa. Esto resulta en que no todos los resultados son comparables unos con otros en las distintas replicaciones, ya que algunas tienen diseños distintos o lo implementan de distinta forma. En el cuadro 6.5 se muestran los niveles del factor Grupo que corresponden a cada replicación.

En las replicaciones de 2006 y 2007 se tienen 6 niveles distintos del factor Grupo, los cuales se conforman mediante las distintas combinaciones de los tres niveles del factor técnica, de acuerdo al orden en que cada una fue aplicada en cada sesión.

En la replicación de 2008, si bien también fueron utilizados los tres niveles del factor técnica, se conformaron solamente 4 grupos de orden de aplicación de técnicas. Esto fue

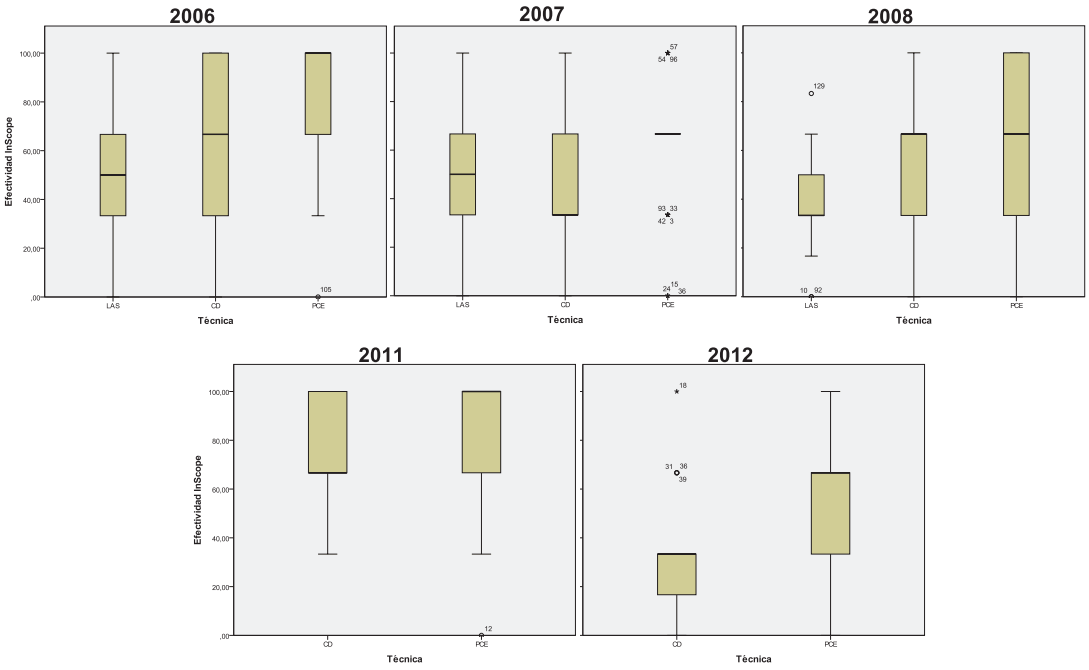

Figura 6.1: Gráficos de caja y bigote - VR: InScope - Factor Técnica

Programa/Sesión	Replicación	# de Obs.	Media	Desv. Estándar
cmdline	2006	46	56,522	31,325
	2007	48	51,390	27,470
	2008	45	45,184	20,910
nametbl	2006	46	68,841	24,245
	2007	48	60,418	22,447
	2008	46	59,783	28,017
	2011	22	80,304	24,472
	2012	23	47,827	26,260
ntree	2006	46	74,638	26,232
	2007	48	50,000	27,932
	2008	43	60,077	31,105
	2011	22	72,728	28,427
	2012	23	28,984	27,163

Cuadro 6.4: Estadísticas Descriptivas - VR: InScope - Factor Programa

una omisión en la asignación de los sujetos a los grupos, en la cual no se tuvo en cuenta el balanceo de la cantidad de sujetos en todos los grupos posibles. En las replicaciones de 2011 y 2012 se tienen únicamente 2 niveles para el factor grupo, ya que se tienen únicamente 2 sesiones utilizando las técnicas CD y PCE.

Para comparar los resultados, generaremos 2 subgrupos de análisis. El primer subgrupo (que llamaremos Subgrupo A), comprende las replicaciones de 2006, 2007 y 2008.

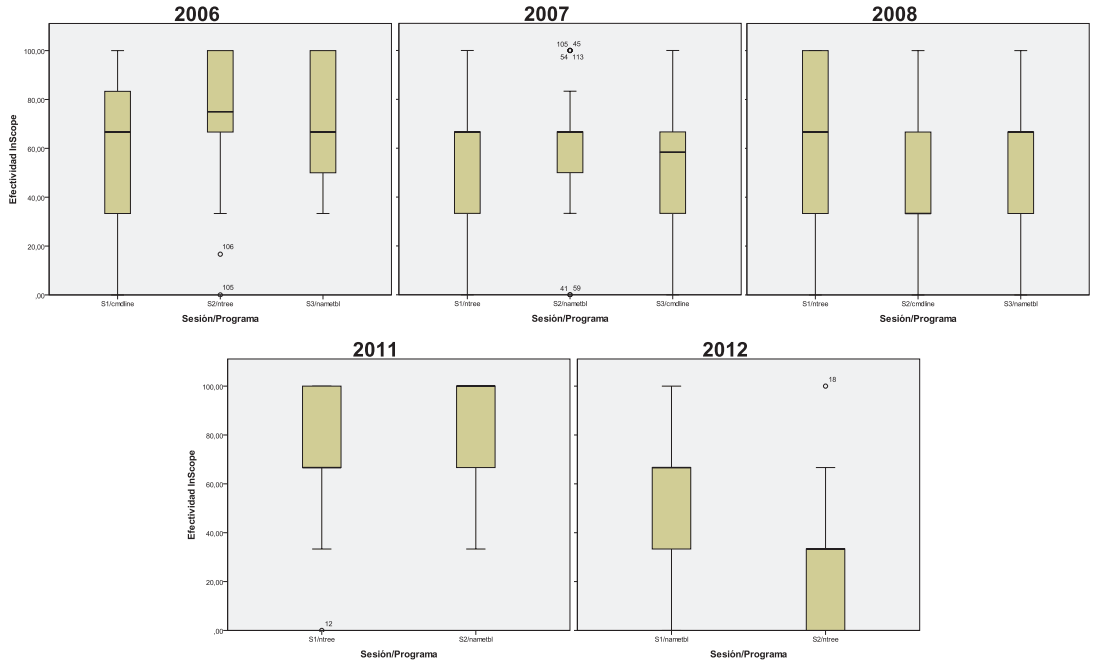

Figura 6.2: Gráficos de caja y bigote - VR: InScope - Factor Programa

Replicaciones	**Niveles del factor Grupo**
2006	LAS-CD-PCE
2007	LAS-PCE-CD
	CD-LAS-PCE
	CD-PCE-LAS
	PCE-LAS-CD
	PCE-CD-LAS
2008	LAS-CD-PCE
	LAS-PCE-CD
	CD-PCE-LAS
	PCE-LAS-CD
2011	CD-PCE
2012	PCE-CD

Cuadro 6.5: Niveles del factor Grupo para cada replicación

A pesar de que en la replicación de 2008 falten niveles del factor grupo, resulta más adecuado incorporarla a este subgrupo y comparar con los niveles y comparaciones en los cuales se coincide con el resto de las replicaciones. El segundo subgrupo (Subgrupo B), está conformado por las replicaciones de 2011 y 2012.

En los cuadros 6.6 y 6.7 se presentan los estadísticos descriptivos para el factor grupo para el subgrupo de replicaciones A (2006, 2007 y 2008) y B (2011 y 2012) respectivamente, en la figura 6.3 se presentan los graficos de boxplot. Nuevamente se registra baja efectividad (sobre todo en el grupo CD-PCE) para la replicación de ESPEL

2012.

Grupo	Replicación	# de Obs.	Media	Desv. Estándar
LAS-CD-PCE	2006	24	71,529	29,273
	2007	24	59,029	26,458
	2008	44	61,742	27,512
LAS-PCE-CD	2006	24	59,028	32,962
	2007	24	57,639	29,068
	2008	–	–	—
CD-LAS-PCE	2006	27	75,310	25,889
	2007	24	52,084	22,690
	2008	X	X	X
CD-PCE-LAS	2006	27	57,407	26,689
	2007	24	47,917	26,610
	2008	42	56,746	24,984
PCE-LAS-CD	2006	18	55,554	24,255
	2007	24	57,640	27,794
	2008	45	47,407	29,293
PCE-CD-LAS	2006	18	82,409	18,498
	2007	24	49,306	25,292
	2008	X	X	X

Cuadro 6.6: Estadísticas Descriptivas - VR: InScope - Factor Gruppo (A)

Grupo	Replicación	# de Obs.	Media	Desv. Estándar
CD-PCE	2011	24	84,723	21,933
	2012	24	27,777	23,399
PCE-CD	2011	20	66,668	28,614
	2012	20	50,001	28,639

Cuadro 6.7: Estadísticas Descriptivas - VR: InScope - Factor Gruppo (B)

6.1.2. Reducción del Conjunto de Datos - InScope

El análisis de los outliers identificados para la variable InScope determinó que todos tienen origen confiable y por tanto no deben quitarse del conjunto de datos para su posterior análisis.

6.1.3. Pruebas de Hipótesis - InScope

Se realiza un chequeo de la validez del modelo para todas las replicaciones, a través de pruebas de normalidad para los residuos. Los resultados para estas pruebas pueden

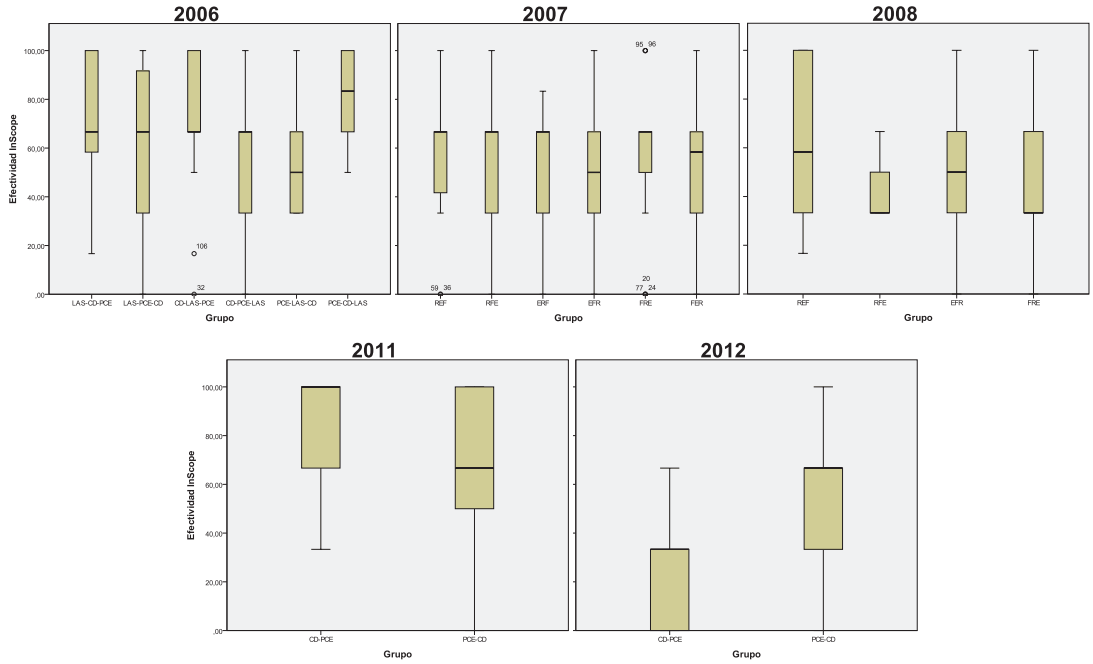

Figura 6.3: Gráficos de caja y bigote - VR: InScope - Factor Grupo

visualizarse en el cuadro 6.8. Los resultados muestran que se validan los modelos para todas las replicaciones a excepción de la replicación de 2008 y 2012.

Replicación	Shapiro-Wilk	Kolmogorov-Smirnov
2006	0,069	0,111
2007	0,200	0,088
2008	0,001	0,000
2011	0,013	0,037
2012	0,005	0,006

Cuadro 6.8: Resultados de pruebas de Normalidad para residuos - VR: InScope

Se realizan transformaciones para las VR de las replicaciones de 2008 y 2012, intentando buscar una mejor significación de normalidad para los residuos. Como se indicó en la sección 4.4.1 se realizan 9 transformaciones para cada VR. En el cuadro 6.9 se muestran los valores de significancia obtenidos para cada una.

Debido que no se obtiene mejor significancia con ninguna transformación para ninguna replicación, se decide realizar los análisis con la variable respuesta original. Se tendrá especial cuidado al momento de analizar e interpretar los datos con los resultados de estas replicaciones ya que contienen una amenaza a la validez agregada.

En el cuadro 6.10 se muestran los niveles de significancia de los tests estadísticos realizados a todas las replicaciones estudiadas en referencia a los factores fijos: Técnica, Programa/Sesión y Grupo. Con respecto al factor Técnica, el cuadro muestra que se

Transformación	2008		2012	
	K-S	S-W	K-S	S-W
y^2	0,000	0,000	0,000	0,000
$\sqrt{y}$	0,000	0,000	0,000	0,002
$\ln(y)$	0,000	0,001	0,000	0,001
$\ln(y+\frac{1}{2})$	0,000	0,000	0,000	0,000
$\log_{10}(y)$	0,000	0,000	0,000	0,000
$\log_{10}(y+\frac{1}{2})$	0,000	0,000	0,000	0,000
$\frac{1}{y}$	0,000	0,000	0,001	0,003
$\frac{1}{\sqrt{y}}$	0,000	0,000	0,000	0,002
$\exp(y)$	0,000	0,000	0,000	0,000

Cuadro 6.9: Resultados de pruebas de Normalidad para las transformaciones realizadas - VR: InScope

han podido detectar diferencias significativas para las replicaciones de 2006, 2007, 2008 y 2012, siendo la única excepción la replicación de 2011. Con respecto al factor Programa/Sesión, han habido diferencias significativas en las replicaciones de 2006, 2007 y 2012, no así para 2008 y 2011. Con respecto al factor Grupo, han habido diferencias significativas en las replicaciones de 2006, 2011 y 2012, no así para las replicaciones de 2007 y 2008.

Factores Fijos	Replicaciones				
	2006	2007	2008	2011	2012
Técnica	0,000	0,009	0,000	0,915	0,041
Programa-Sesión	0,003	0,018	0,094	0,337	0,006
Grupo	0,010	0,562	0,357	0,025	0,008

Cuadro 6.10: Niveles de Significancia de Análisis estadístico para todas las replicaciones - Variable de Respuesta: InScope

Si bien es relevante conocer qué factores tuvieron efectos significativos en cada replicación, más relevante aún es conocer las comparaciones por pares de los niveles dentro de cada factor e identificar cuáles interacciones son las que resultan significativas y las diferencias de medias de efectividad entre ellas. A continuación analizaremos cada factor que mostró diferencias significativas en al menos una replicación.

6.1.3.1. Pruebas de Hipótesis: VR: InScope - Factor Técnica

Las medias marginales estimadas para cada técnica se presentan en el cuadro 6.11, los resultados de significancia y diferencia de medias entre los distintos niveles del factor técnica se presentan en el cuadro 6.12. Para las replicaciones de 2011 y 2012 solamente se utilizan las técnicas CD y PCE, por tanto no hay datos para las comparaciones con respecto a LAS.

Técnica	2006	2007	2008	2011	2012
LAS	54,037	52,778	38,118	-	-
CD	67,676	46,528	59,953	75,279	31,943
PCE	78,932	62,502	60,809	76,112	45,834

Cuadro 6.11: Medias Marginales Estimadas del Factor Técnica - Variable de Respuesta: InScope

Técnicas	Valores	Replicaciones				
		2006	2007	2008	2011	2012
LAS vs. CD	Sig.	0,017	0,616	0,000	-	-
	Dif. Med.	-13,639	6,250	-21,835	-	-
LAS vs. PCE	Sig.	0,000	0,058	0,000	-	-
	Dif. Med.	-24,896	-9,724	-22,691	-	-
CD vs. PCE	Sig.	0,076	0,012	1,000	0,915	0,041
	Dif. Med.	-11,256	-15,974	-0,856	-0,833	-13,891

Cuadro 6.12: Comparaciones por pares del Factor técnica - Valores de significancia y diferencia entre medias - Variable de Respuesta: InScope

Para la comparación entre las técnicas de Lecturas por Abstracciones Sucesivas y Criterio de Decisión (LAS vs. CD) las pruebas estadísticas realizadas muestran que existen diferencias significativas entre la efectividad de ambas técnicas para las replicaciones de 2006 y 2008 (valores de Sig. de 0,017 y 0,000, respectivamente), no así para la replicación de 2007 (Sig. 0,616).

A pesar de que en la mayoría de las replicaciones se encuentran diferencias significativas para LAS vs. CD, las tendencias no resultan ser las mismas. Para las replicaciones de 2006 y 2008 CD resulta más efectiva que LAS, siendo 13,639 % y 21,835 % más efectiva respectivamente, valores que se muestran en el cuadro 6.12. Sin embargo, para la replicación de 2007 (aunque esta no resulte significativa) la tendencia es opuesta y LAS resulta más efectiva que CD, en 6,250 %.

Tomando en cuenta las replicaciones significativas, parecería que CD presenta mayor efectividad que LAS en relación a las faltas que están dentro del alcance de la técnica. Esto se puede observar en el gráfico de líneas que se muestra en la figura 6.4.

Para la comparación entre las técnicas de Lecturas por Abstracciones Sucesivas y Particiones en Clases de equivalencia (LAS vs. PCE) existen diferencias significativas para las replicaciones de 2006 y 2008 y no significativa para la de 2007, siendo ésta una diferencia muy pequeña por la cual se descarta significancia (Sig. 0.058). Para estas dos técnicas, todas las replicaciones muestran la misma tendencia en las comparaciones por pares, siendo PCE más efectiva que LAS en un promedio de 19,104 %, tomando las diferencias entre medias presentadas en el cuadro 6.12. Estos resultados parecen ser concluyentes ya que a pesar de que no todas las replicaciones muestren diferencias significativas, aún así conservan la misma tendencia, siendo PCE más efectiva que LAS, para las faltas que están dentro del alcance de cada técnica.

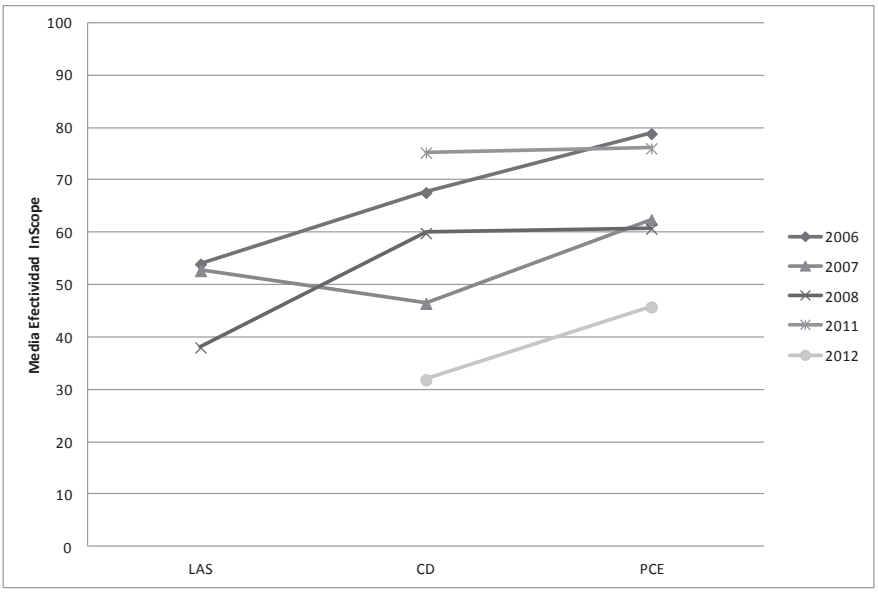

Figura 6.4: Gráfico de líneas para el factor Técnica - VR: InScope

Entre las técnicas de Criterio de Decisión y Particiones en Clases de equivalencia (CD vs. PCE), se presentan diferencias significativas para las replicaciones de 2007 y 2012. La tendencia es la misma tanto para las replicaciones con diferencias significativas como para las que no, siendo PCE más efectiva que CD en un 8,562 % en promedio. Los resultados parecen ser concluyentes también para esta comparación, en donde PCE resulta más efectiva que CD en lo que refiere a las faltas que están dentro del alcance de cada técnica.

6.1.3.2. Pruebas de Hipótesis: VR: InScope - Factor Programa/Sesión

Es importante recordar que el programa y la sesión están confundidos, por lo que una diferencia significativa no es claramente adjudicable al efecto de uno u otro factor. Por eso es importante ver los resultados de las pruebas estadísticas tanto desde el punto de vista de los programas como de la sesión.

En el cuadro 6.13 se presentan las medias estimadas y en el cuadro 6.14 se presentan los resultados de significancia y diferencia de medias, vistos desde el punto de vista del programa. Para las replicaciones de 2011 y 2012 se tienen solamente 2 sesiones con 2 programas cada una (nametble y ntree). En el cuadro 6.13 se presentan las medias marginales estimadas.

Desde el punto de vista del programa, los resultados de las comparaciones por pares para cmdline y nametable dan significativos solamente en la replicación de 2007. Aún así, la tendencia es la misma para todas las replicaciones, en donde la efectividad en el programa cmdline es inferior a la efectividad en nametable. Esta tendencia se puede observar en el gráfico de líneas que se presenta en la figura 6.5.

Para la comparación entre cmdline y ntree, la única replicación que resulta sig-

Programa	2006	2007	2008	2011	2012
cmdline	57,750	49,215	46,950	-	-
nametbl	68,555	61,458	54,588	79,445	48,612
ntree	74,339	51,135	57,341	71,945	29,165

Cuadro 6.13: Medias Estimadas del Factor Programa - Variable de Respuesta: InScope

Programas	Valores	Replicaciones				
		2006	2007	2008	2011	2012
cmdline vs. nametbl	Sig.	0,080	0,025	0,356	-	-
	Dif. Med.	-10,805	-12,243	-7,638	-	-
cmdline vs. ntree	Sig.	0,003	1,000	0,111	-	-
	Dif. Med.	-16,589	-1,920	-10,391	-	-
nametbl vs. ntree	Sig.	0,686	0,076	1,000	0,337	0,006
	Dif. Med.	-5,784	10,323	-2,753	7,500	19,446

Cuadro 6.14: Comparaciones por pares del Factor programa - Valores de significancia y diferencia entre medias - Variable de Respuesta: InScope

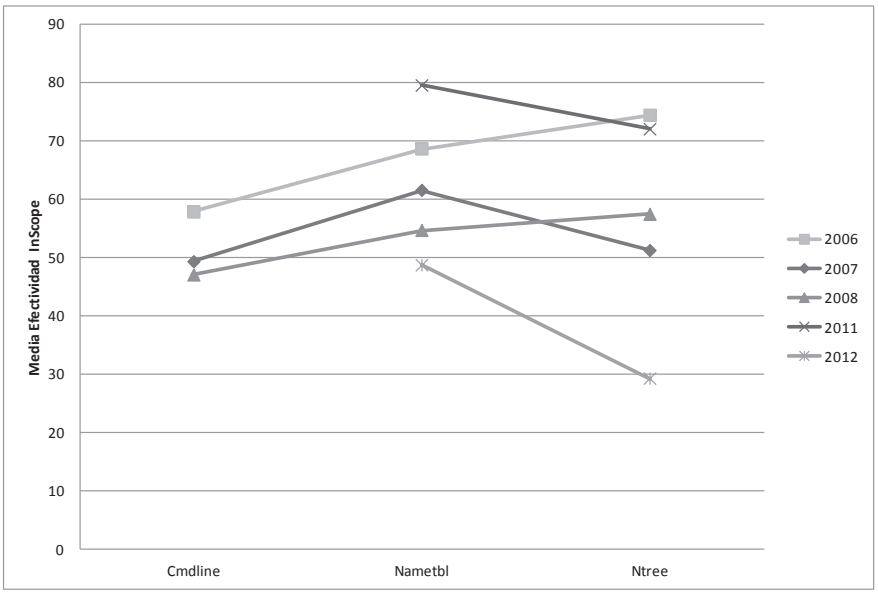

Figura 6.5: Gráfico de líneas para el factor Programa - VR: InScope

nificativa es la de 2006, aún así la tendencia nuevamente se mantiene para todas las replicaciones, en donde la efectividad sobre cmdline es menor que sobre el programa ntree.

Por último, para la comparación entre nametable y ntree, la única replicación que resulta significativa es la de 2012, siendo mayor la efectividad en nametable que en ntree. Para esta última comparación las tendencias no se mantienen a lo largo de todas las replicaciones, en donde nametable es más efectivo que ntree para las replicaciones de

2007, 2011 y 2012 con una diferencia de 10,323 %, 7,500 % y 19,446 % respectivamente. En contrapartida, ntree resulta más efectivo que nametable en las replicaciones de 2006 y 2008 con una diferencia de 5,784 % y 2,753 % respectivamente.

Cambiando el enfoque hacia la sesión, en el cuadro 6.17 se presentan los valores de estimación de medias y en el cuadro 6.16 se muestran los resultados de significancia. Si bien son los mismos resultados, no se pueden agrupar en la misma tabla que los resultados del programa, ya que la correspondencia de programa-sesión varía de una replicación a otra. Dicha correspondencia se presentó anteriormente en el cuadro 6.2.

Sesión	**2006**	**2007**	**2008**	**2011**	**2012**
Sesión 1	57,750	51,135	57,341	71,945	48,612
Sesión 2	74,339	61,458	46,950	79,445	29,165
Sesión 3	68,555	49,215	54,588	-	-

Cuadro 6.15: Medias Estimadas del Factor Sesión - Variable de Respuesta: InScope

Sesiones	**Valores**	**Replicaciones**				
		2006	**2007**	**2008**	**2011**	**2012**
S1 vs. S2	Sig.	0,003	0,076	0,111	0,337	0,006
	Dif. Med.	-16,589	-10,323	10,391	-7,500	19,446
S1 vs. S3	Sig.	0,080	1,000	1,000	-	-
	Dif. Med.	-10,805	1,920	2,753	-	-
S2 vs. S3	Sig.	0,686	0,025	0,356	-	-
	Dif. Med.	5,784	-12,243	-7,638	-	-

Cuadro 6.16: Comparaciones por pares del Factor Sesión - Valores de significancia y diferencia entre medias - Variable de Respuesta: InScope

Hubo diferencias significativas en 2006 y 2012 en lo que refiere a la comparación entre S1 y S2. Las tendencias no se mantienen para estas dos replicaciones (resultan opuestas), ni para el resto que no resultaron significativas. En 2006, 2007 y 2011, S2 resulta más efectiva que S1 en un 16,589 %, 10,323 % y 7,500 % respectivamente. En 2007 y 2012 S1 resulta más efectiva que S2 en un 10,391 % y 19,446 % respectivamente. Dichas tendencias se pueden visualizar en el gráfico de líneas que se presenta en la figura 6.6.

Ninguna comparación resultó significativa entre las sesiones S1 y S3 y las tendencias no se mantienen. En 2006 S3 resulta más efectiva que S1 y en 2007 y 2008, S1 resulta más efectiva que S3, aunque con muy poca diferencia de medias (1,920 % y 2,753 % respectivamente).

Entre S2 y S3 solamente la replicación de 2007 resulta significativa en donde S3 resulta más significativa que S2 en un 12,243 %. Sin embargo la tendencia tampoco es siempre la misma para todas las replicaciones, en donde S2 resulta más efectiva que S3 en la replicación de 2006.

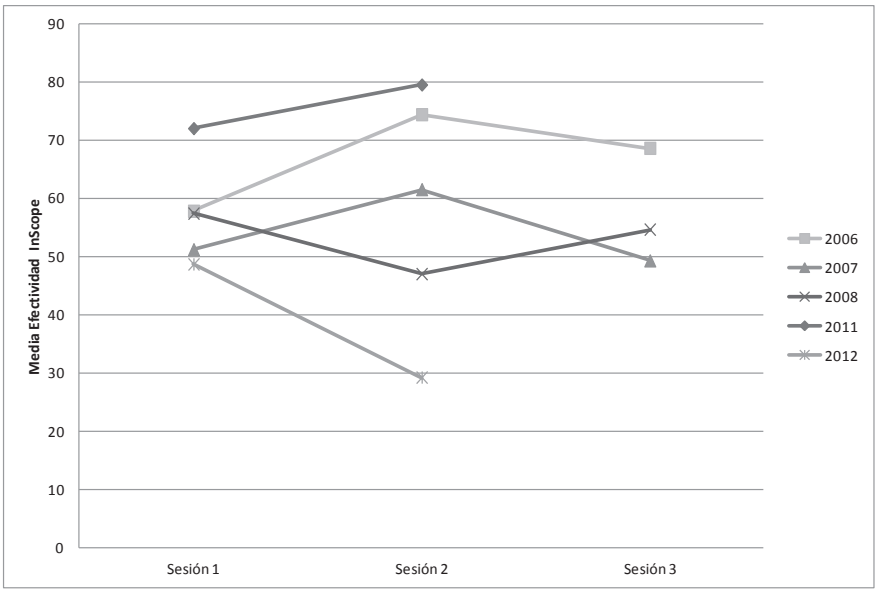

Figura 6.6: Gráfico de líneas para el factor Sesión - VR: InScope

El hecho de que la tendencia de las comparaciones entre las sesiones no se mantenga para la mayoría de las replicaciones podría estar descartando un posible efecto de carry-over entre una sesión y la siguiente. Este fenómeno descartaría un posible efecto de aprendizaje por la práctica. Sin embargo, también hay comparaciones entre sesiones que denotan un descenso de la efectividad entre una sesión y la sesión siguiente que resultan significativas. Esto último descartaría un posible efecto de cansancio.

Sin embargo, desde el punto de vista del programa, las tendencias coinciden en su gran mayoría, esto es fácilmente observable comparando los gráficos de líneas que aparecen en las figuras 6.5 y 6.6. Esto parecería indicar que la variable que más influye en este factor Programa/Sesión que se encuentra combinado, es el efecto del programa y no el de la sesión.

6.1.3.3. Pruebas de Hipótesis: VR: InScope - Factor Grupo

De acuerdo a los distintos subgrupos de replicaciones definidos anteriormente, primeramente se realiza un análisis del Subgrupo A de replicaciones (2006, 2007 y 2008) y luego del Subgrupo B (2011 y 2012).

En el cuadro 6.17 se muestran las medias estimadas y en el cuadro 6.18 se presentan las comparaciones por pares para las replicaciones que conforman el **Subgrupo A**.

De todas las comparaciones por pares realizadas para las replicaciones de 2006, 2007 y 2008, ninguna resulta significativa y las tendencias no coinciden en casi ninguna comparación, teniendo gran variabilidad entre ellas. En la figura 6.7 se muestra el gráfico de líneas para las replicaciones 2006, 2007 y 2008, en donde no se logra visulmbrar ningún patrón de variabilidad que sea constante a lo largo de las replicaciones.

Los resultados parecerían bastante concluyentes en que el factor grupo no tiene

Grupo	2006	2007	2008
LAS-CD-PCE	71,253	60,933	57,828
LAS-PCE-CD	59,382	56,175	45,188
CD-LAS-PCE	74,988	54,622	-
CD-PCE-LAS	57,571	46,354	58,230
PCE-LAS-CD	55,967	55,700	50,593
PCE-CD-LAS	82,129	49,831	-

Cuadro 6.17: Medias Estimadas del Factor Grupo - Variable de Respuesta: InScope

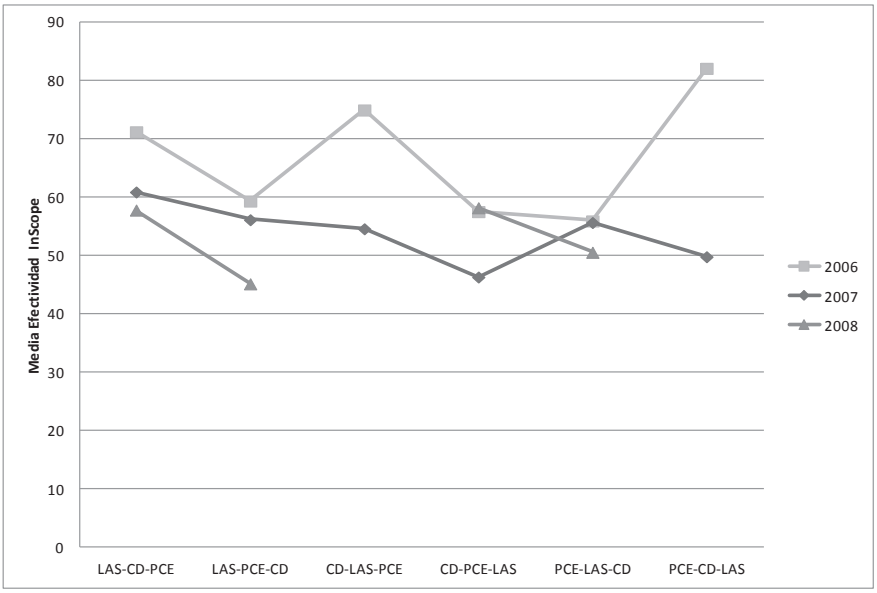

Figura 6.7: Gráfico de líneas para el factor Grupo - Subgrupo A - VR: InScope

influencia sobre la efectividad de las técnicas para las faltas que están dentro de su alcance, en lo que respecta al grupo A de replicaciones.

En los cuadros 6.19 y 6.20 se presentan las medias estimadas y las comparaciones por pares para las replicaciones que conforman el **Subgrupo B**, respectivamente.

Las comparaciones por pares del factor grupo resultan ser significativas tanto para la replicación de 2011 como de 2012. Sin embargo, las tendencias resultan opuestas entre ellas, esto se puede visualizar en la figura 6.8. Esto también parecería indicar que el grupo no tiene una influencia clara en los resultados.

Tomando en cuenta ambos análisis (subgrupo A y B), los resultados parecieran ser concluyentes en que el factor grupo no influencia la efectividad de las técnicas para las faltas que están dentro de su alcance. Los resultados no son consistentes a lo largo de las replicaciones, tanto para las del subgrupo A como las del subgrupo B, teniendo muy pocas replicaciones con resultados significantes, y aquellas que sí lo tienen, no conservan consistencia en las tendencias.

Grupos	Valores	Replicaciones		
		2006	2007	2008
LAS-CD-PCE vs. LAS-PCE-CD	Sig.	1,000	1,000	1,000
	Dif. Med.	11,871	4,757	12,640
LAS-CD-PCE vs. CD-LAS-PCE	Sig.	1,000	1,000	-
	Dif. Med.	-3,735	6,310	-
LAS-CD-PCE vs. CD-PCE-LAS	Sig.	1,000	1,000	1,000
	Dif. Med.	13,682	14,578	-0,402
LAS-CD-PCE vs. PCE-LAS-CD	Sig.	1,000	1,000	1,000
	Dif. Med.	15,286	5,233	7,236
LAS-CD-PCE vs. PCE-CD-LAS	Sig.	1,000	1,000	-
	Dif. Med.	-10,875	11,101	-
LAS-PCE-CD vs. CD-LAS-PCE	Sig.	0,632	1,000	-
	Dif. Med.	-15,606	1,553	-
LAS-PCE-CD vs. CD-PCE-LAS	Sig.	1,000	1,000	1,000
	Dif. Med.	1,811	9,821	-13,042
LAS-PCE-CD vs. PCE-LAS-CD	Sig.	1,000	1,000	1,000
	Dif. Med.	3,415	0,476	-5,405
LAS-PCE-CD vs. PCE-CD-LAS	Sig.	0,132	1,000	-
	Dif. Med.	-22,747	6,344	-
CD-LAS-PCE vs. CD-PCE-LAS	Sig.	0,306	1,000	-
	Dif. Med.	17,417	8,268	-
CD-LAS-PCE vs. PCE-LAS-CD	Sig.	0,349	1,000	-
	Dif. Med.	19,021	-1,077	-
CD-LAS-PCE vs. PCE-CD-LAS	Sig.	1,000	1,000	-
	Dif. Med.	-7,141	4,791	-
CD-PCE-LAS vs. PCE-LAS-CD	Sig.	1,000	1,000	0,931
	Dif. Med.	1,604	-9,345	7,637
CD-PCE-LAS vs. PCE-CD-LAS	Sig.	0,061	1,000	-
	Dif. Med.	-24,557	-3,477	-
PCE-LAS-CD vs. PCE-CD-LAS	Sig.	0,077	1,000	-
	Dif. Med.	-26,162	5,868	-

Cuadro 6.18: Comparaciones por pares del Factor Grupo (A) - Valores de significancia y diferencia entre medias - Variable de Respuesta: InScope

Grupo	2011	2012
CD-PCE	84,723	27,777
PCE-CD-	66,668	50,000

Cuadro 6.19: Medias Estimadas del Factor Grupo B - Variable de Respuesta: InScope

Grupos	Valores	Replicaciones	
		2011	2012
CD-PCE vs. PCE-CD	Sig.	0,025	0,008
	Dif. Med.	18,055	-22,224

Cuadro 6.20: Comparaciones por pares del Factor Grupo(B) - Valores de significancia y diferencia entre medias - Variable de Respuesta: InScope

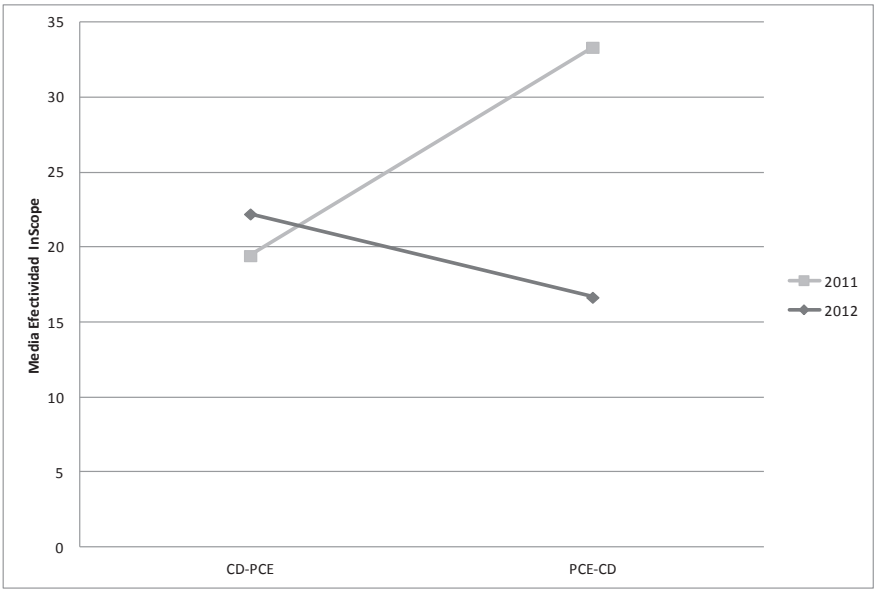

Figura 6.8: Gráfico de líneas para el factor Grupo - Subgrupo B - VR: InScope

6.2. Variable de Respuesta: OutScope

En esta sección se realiza el análisis estadístico respecto de la variable OutScope.

6.2.1. Estadísticas Descriptivas - OutScope

En el cuadro 6.21 se presentan los estadísticos descriptivos para cada replicación y cada nivel del factor Técnica, en la figura 6.9 se muestran los gráficos boxplot para las medidas de media. Se observa un notorio descenso de la efectividad en cada técnica respecto de la variable InScope que se presentó en el cuadro 6.3. Lo cual de cierta forma resulta lógico, ya que se espera que la técnica sea más efectiva para las faltas que están dentro de su alcance que para las que están fuera. Se observa además que en CD resulta siempre más efectiva que PCE de forma muy notoria. Observando la desviación estándar, PCE presenta un poco menos de variabilidad que CD.

De los estadísticos descriptivos para el factor programa que se presentan en 6.22, se observa baja efectividad para el programa cmdline, aspecto también osbervado para la variable InScope. Los gráficos de boxplot se muestran en la figura 6.10.

Técnica	Replicación	# de Obs.	Media	Desv. Estándar
CD	2006	46	28,261	27,188
	2007	48	35,417	32,549
	2008	42	23,809	26,835
	2011	22	34,848	33,298
	2012	23	27,536	32,802
PCE	2006	46	14,492	19,439
	2007	48	15,277	19,397
	2008	46	15,941	20,774
	2011	22	16,666	22,420
	2012	23	11,593	16,231

Cuadro 6.21: Estadísticas Descriptivas - VR: OutScope - Factor Técnica

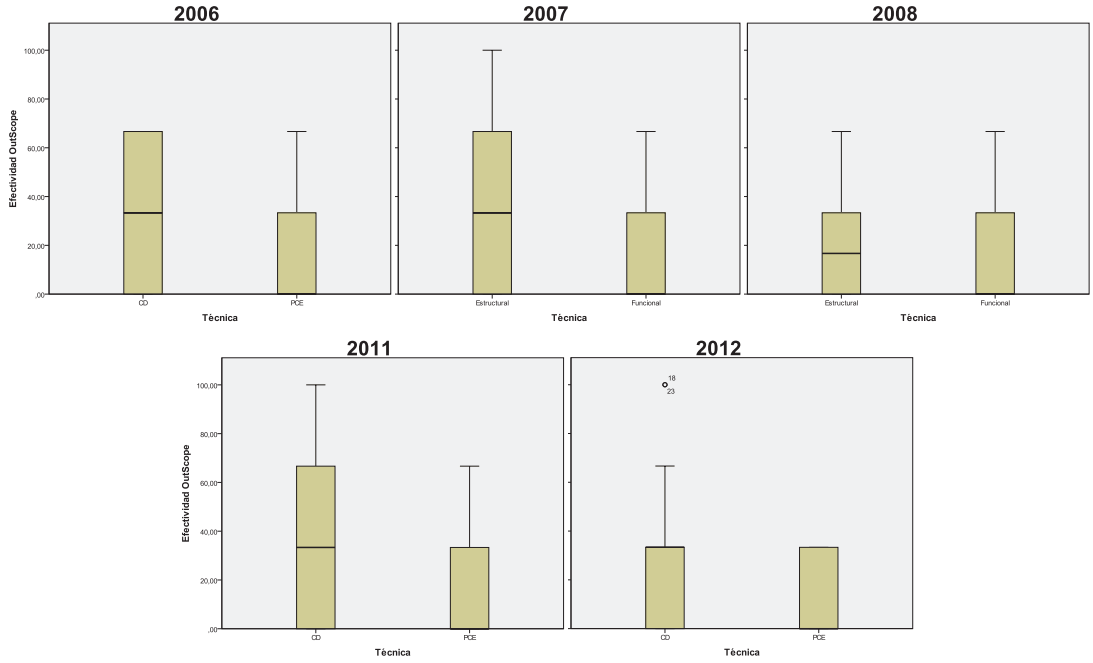

Figura 6.9: Gráficos de de caja y bigote - VR: OutScope - Factor Técnica

Los estadísticos descriptivos para el factor Grupo se presentan en los cuadros 6.23 para las replicaciones 2006, 2007 y 2008, y en el cuadro 6.24 para las replicaciones de 2011 y 2012, en la figura 6.11 se muestran los gráficos de boxplot. No se observan diferencias notorias entre los niveles de los grupos, esto también se debe a que existe gran variabilidad de efectividad entre replicaciones dentro del mismo nivel (p.e. el nivel LAS-CD-PCE presenta efectivades de 14,582 %, 41,667 % y 5,747 % para las replicaciones de 2006, 2007 y 2008 respectivamente).

Programa	Replicación	# de Obs.	Media	Desv. Estándar
cmdline	2006	30	15,555	20,960
	2007	32	14,583	22,301
	2008	30	8,888	14,991
nametbl	2006	31	29,033	29,493
	2007	32	37,500	36,663
	2008	31	24,731	27,174
	2011	22	28,787	31,363
	2012	23	20,289	26,090
ntree	2006	31	19,354	20,681
	2007	32	23,957	19,370
	2008	27	25,925	25,036
	2011	22	22,727	27,959
	2012	23	18,840	28,116

Cuadro 6.22: Estadísticas Descriptivas - VR: OutScope - Factor Programa

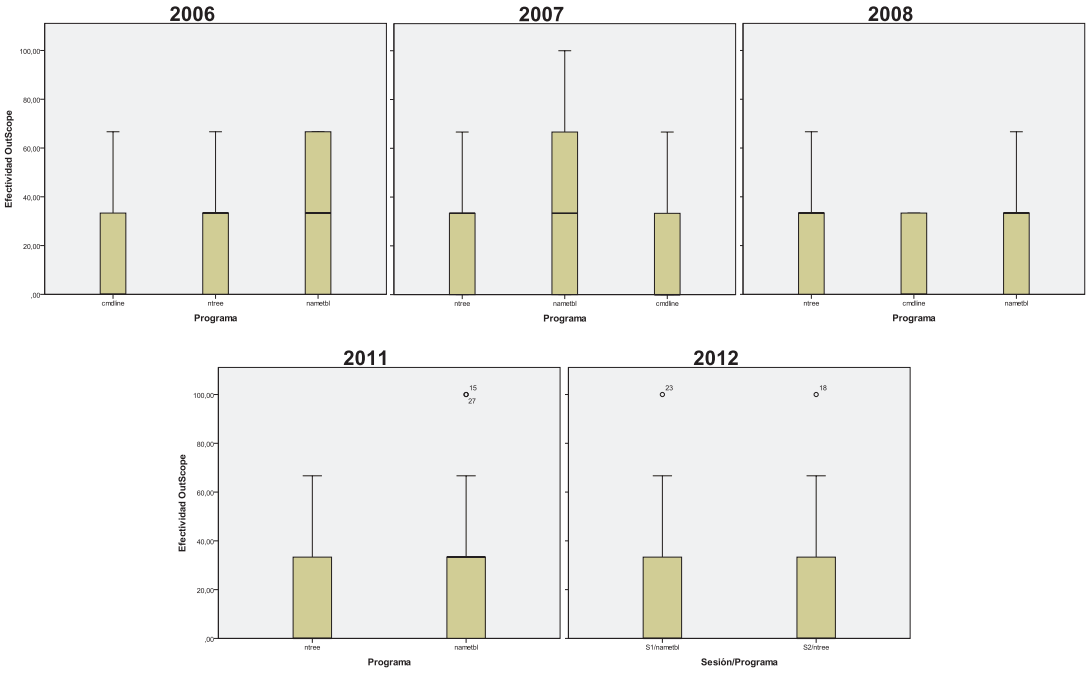

Figura 6.10: Gráficos de de caja y bigote - VR: OutScope - Factor Programa

6.2.2. Reducción del Conjunto de Datos - OutScope

El análisis de los outliers identificados para la variable OutScope determinó que todos tienen origen confiable y por tanto no deben quitarse del conjunto de datos para su posterior análisis.

Grupo	Replicación	# de Obs.	Media	Desv. Estándar
LAS-CD-PCE	2006	16	14,582	17,077
	2007	16	41,667	35,487
	2008	29	5,747	12,813
LAS-PCE-CD	2006	16	35,416	25,732
	2007	16	10,416	20,069
	2008	–	–	–
CD-LAS-PCE	2006	18	14,814	20,523
	2007	16	16,666	21,081
	2008	X	X	X
CD-PCE-LAS	2006	18	20,37	25,919
	2007	16	22,916	23,472
	2008	27	16,049	21,424
PCE-LAS-CD	2006	12	36,112	30,013
	2007	16	18,749	20,971
	2008	30	34,444	25,498
PCE-CD-LAS	2006	12	8,333	15,074
	2007	16	41,666	33,334
	2008	X	X	X

Cuadro 6.23: Estadísticas Descriptivas - VR: OutScope - Factor Grupo (A)

Grupo	Replicación	# de Obs.	Media	Desv. Estándar
CD-PCE	2011	24	19,444	23,910
	2012	24	22,221	25,379
PCE-CD	2011	20	33,333	34,200
	2012	22	16,666	28,638

Cuadro 6.24: Estadísticas Descriptivas - VR: OutScope - Factor Grupo (B)

6.2.3. Pruebas de Hipótesis - OutScope

En el cuadro 6.25 se presentan los resultados de las pruebas de normalidad para los residuos de todas las replicaciones, a modo de chequear la validez del modelo. Los resultados muestran que no se validan los modelos para las replicaciones de 2006, 2008, 2011 y 2012.

Se realizan transformaciones para las VR de las replicaciones de 2006, 2008, 2011 y 2012, intentando buscar una mejor significación de normalidad para los residuos. Como se indicó en la sección 4.4.1 se realizan 9 transformaciones para cada VR. En el cuadro 6.26 se muestran los valores de significancia obtenidos para cada una.

En la replicación de 2006 no se obtiene mejor significancia con ninguna transformación y por lo tanto se realiza el análisis con la variable de respuesta original. Sin embargo, para la replicación de 2008 se obtiene mejor significancia con la transformación $\sqrt{y}$ (0,039), para la replicación de 2011 se obtiene mejor significancia con las

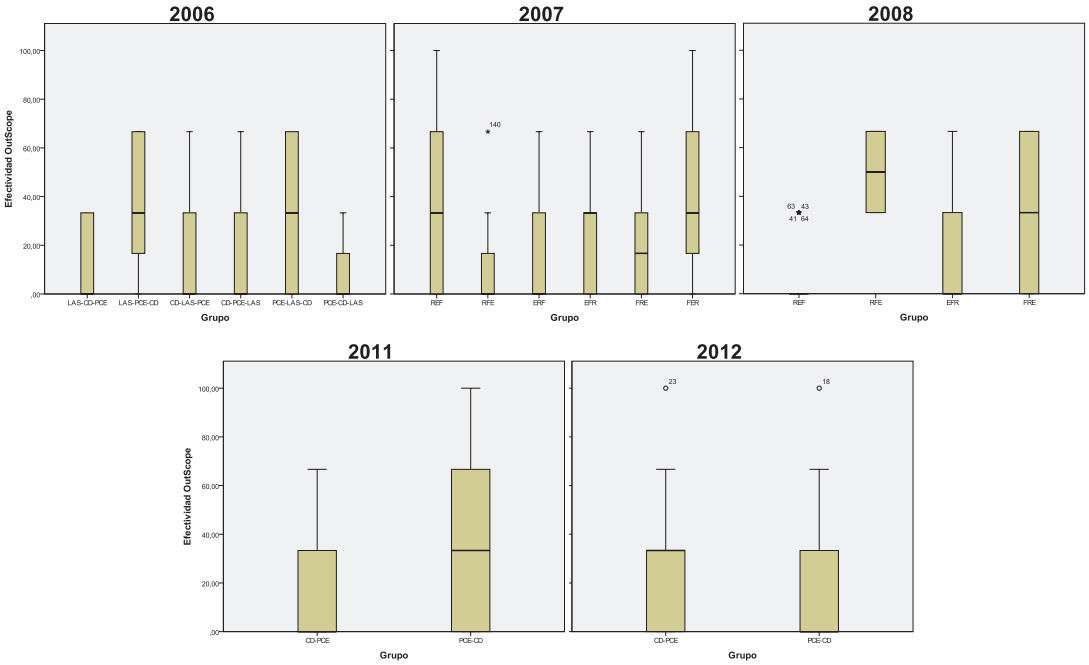

Figura 6.11: Gráficos de de caja y bigote - VR: OutScope - Factor Grupo

Replicación	Shapiro-Wilk	Kolmogorov-Smirnov
2006	0,000	0,000
2007	0,198	0,190
2008	0,000	0,009
2011	0,000	0,000
2012	0,000	0,000

Cuadro 6.25: Resultados de pruebas de Normalidad para residuos - VR: OutScope

transformaciones $\ln(y)$ y $\log_{10}(y)$ (0,013 para ambas) y en la replicación de 2012 se obtiene mejor significancia con la transformación $\sqrt{y}$. Por tanto, para las replicaciones 2008 y 2012 se realizan los análisis con la transformación $\sqrt{y}$, y para la replicación de 2011 con la trasnformación $\ln(y)$.

Para los análisis en los que se usan variables transformadas, se presentan los valores de significancia obtenidos con la transformación, pero los resultados de estimaciones de medias, diferencias de medias y otros se realizan aplicando la función anti-transformada (y^2 para el caso de $\sqrt{y}$ y $\exp(y)$ para el caso de $\ln(y)$.

Los niveles de significancia obtenidos en relación a los efectos fijos para la variable de respuesta OutScope se presentan en el cuadro 6.27, en relación a todos los factores estudiados. Para el factor Técnica se han encontrado diferencias significativas para las replicaciones de 2006, 2007 y 2012, no así para las replicaciones de 2008 y 2011. Para el factor Programa/Sesión únicamente se han encontrado diferencias significativas para la replicación de 2007. Por último, para el factor Grupo se han encontrado diferencias

Transformación	2006		2008		2011		2012	
	K-S	S-W	K-S	S-W	K-S	S-W	K-S	S-W
y^2	0,005	0,008	0,000	0,000	0,000	0,000	0,000	0,000
$\sqrt{y}$	0,000	0,000	0,000	0,039	0,000	0,002	0,012	0,014
$\ln(y)$	0,000	0,002	0,002	0,001	0,006	0,013	0,000	0,001
$\ln(y+\frac{1}{2})$	0,000	0,000	0,000	0,022	0,000	0,000	0,000	0,003
$\log_{10}(y)$	0,000	0,000	0,002	0,001	0,006	0,013	0,000	0,000
$\log_{10}(y+\frac{1}{2})$	0,000	0,000	0,000	0,022	0,000	0,000	0,000	0,003
$\frac{1}{y}$	0,000	0,002	0,002	0,001	0,002	0,003	0,000	0,000
$\frac{1}{\sqrt{y}}$	0,000	0,002	0,002	0,001	0,003	0,006	0,000	0,000
$\exp(y)$	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000

Cuadro 6.26: Resultados de pruebas de Normalidad para las transformaciones realizadas - VR: OutScope

significativas en las replicaciones de 2006, 2007 y 2008, no así para 2011 y 2012.

Factores Fijos	Replicaciones				
	2006	2007	2008	2011	2012
Técnica	0,001	0,000	0,125	0,155	0,004
Programa-Sesión	0,080	0,007	0,256	0,427	0,610
Grupo	0,009	0,006	0,001	0,155	0,680

Cuadro 6.27: Niveles de Significancia de Análisis estadístico para todas las replicaciones - Variable de Respuesta: OutScope

A continuación se presentan las comparaciones por pares realizadas para cada factor.

6.2.3.1. Pruebas de Hipótesis: VR: OutScope - Factor Técnica

Cabe recordar que para el análisis de la variable de respuesta OutScope, no se incluye la técnica de Lectura por Abstracciones Sucesivas (LAS), ya que al ser una técnica de revisión, en teoría todos los defectos resultan visibles a la misma y no se tienen defectos fuera de su alcance. Por tanto, las comparaciones por pares toman en cuenta únicamente a las técnicas de Criterio de Decisión (CD) y Particiones en Clases de Equivalencia (PCE). En el cuadro 6.28 se presentan las estimaciones de medias y en el cuadro 6.29 se presentan los resultados de significancia y diferencia de medias del factor técnica para la variable OutScope.

La única comparación que se tiene entonces es entre CD y PCE, para la cual se encuentran diferencias significativas para las replicaciones de 2006, 2007 y 2012. Para todas las replicaciones se mantiene la tendencia, en donde CD resulta más efectiva que PCE en un 13,98 % de efectividad en promedio. Esto se puede observar en el gráfico de líneas de la figura 6.12. Los resultados parecen ser concluyentes en torno a que la técnica CD es más efectiva que PCE para las faltas que están fuera de su alcance.

Técnica	2006	2007	2008	2011	2012
CD	29,089	35,416	18,131	49,590	22,700
PCE	14,119	15,277	12,230	39,638	3,960

Cuadro 6.28: Medias Estimadas del Factor Tecnica - Variable de Respuesta: OutScope

Técnicas	Valores	Replicaciones				
		2006	2007	2008	2011	2012
CD vs. PCE	Sig.	0,001	0,000	0,125	0,155	0,004
	Dif. Med.	15,180	20,140	5,900	9,952	18,740

Cuadro 6.29: Comparaciones por pares del Factor técnica - Valores de significancia y diferencia entre medias - Variable de Respuesta: OutScope

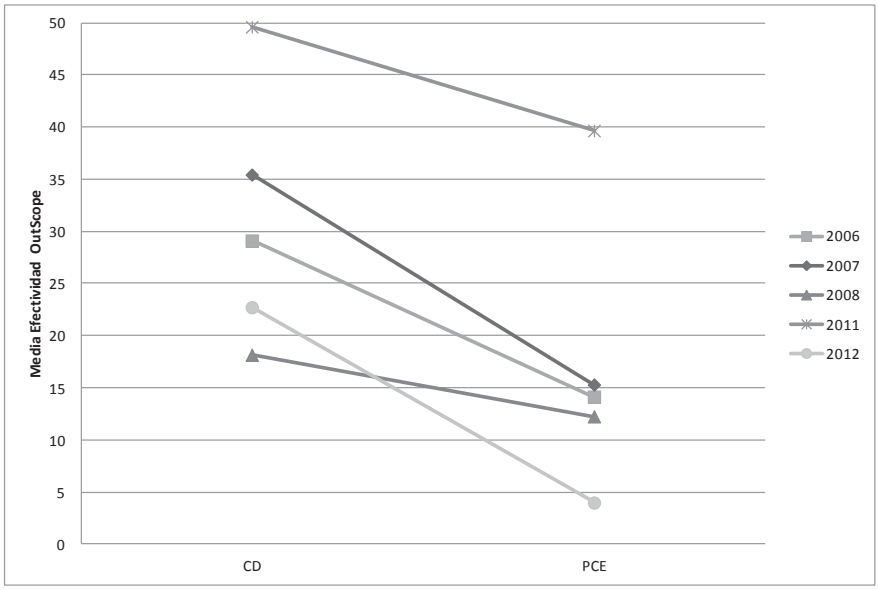

Figura 6.12: Gráfico de líneas para el factor Técnica - VR: OutScope

6.2.3.2. Pruebas de Hipótesis: VR: OutScope - Factor Programa/Sesión

Al igual que para la variable InScope, se analizan los resultados de este factor desde el punto de vista del programa por un lado, y de la sesión por otro. En el cuadro 6.31 se presentan los resultados de significancia y diferencia de medias vistos desde el punto de vista del programa.

Para la comparación entre cmdline y nametbl se obtienen resultados significativos para la replicación de 2007, no así para las replicaciones de 2006 y 2008. La tendencia es igual para todas las replicaciones, en donde la efectividad sobre nametbl resulta mayor que en cmdline en un 14,770 % en promedio.

Para la comparación entre cmdline y ntree no se obtienen diferencias significativas

Técnica	2006	2007	2008	2011	2012
cmdline	14,406	15,624	10,827	-	-
nametbl	29,122	36,459	19,588	41,698	9,932
ntree	21,319	23,957	15,339	47,139	12,981

Cuadro 6.30: Medias Estimadas del Factor Programa - Variable de Respuesta: OutScope

Programas	Valores	Replicaciones				
		2006	2007	2008	2011	2012
cmdline vs. nametbl	Sig.	0,076	0,005	0,302	-	-
	Dif. Med.	-14,716	-20,834	-8,761	-	-
cmdline vs. ntree	Sig.	0,863	0,568	1,000	-	-
	Dif. Med.	-6,913	-8,332	-4,512	-	-
nametbl vs. ntree	Sig.	0,655	0,155	1,000	0,427	0,610
	Dif. Med.	7,803	12,502	4,249	-5,441	-3,049

Cuadro 6.31: Comparaciones por pares del Factor programa - Valores de significancia y diferencia entre medias - Variable de Respuesta: OutScope

para ninguna replicación. Sin embargo, la efectividad en ntree resulta ser mayor que en cmdline para todas las replicaciones en un 6,586 % en promedio.

Finalmente, para la comparación entre nametble y ntree tampoco se obtuvieron resultados significativos en ninguna replicación y las diferencias de medias varían en tendencia. En la figura 6.13 se pueden visualizar claramente las tendencias mencionadas.

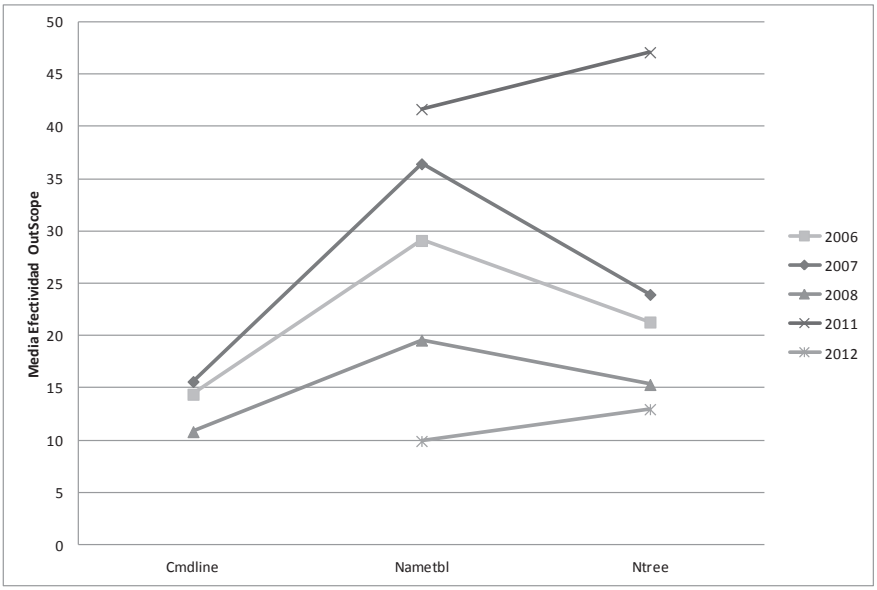

Figura 6.13: Gráfico de líneas para el factor Programa - VR: OutScope

Desde el punto de vista de la sesión, en el cuadro 6.32 se presentan las medias estimadas y en el cuadro 6.33 se muestran los resultados de significancia para todas las replicaciones.

Sesión	2006	2007	2008	2011	2012
Sesión 1	14,406	23,957	15,339	47,139	9,932
Sesión 2	21,319	36,459	10,827	41,698	12,981
Sesión 3	29,122	15,624	19,588	-	-

Cuadro 6.32: Medias Estimadas del Factor Sesión - Variable de Respuesta: OutScope

Sesiones	Valores	Replicaciones				
		2006	2007	2008	2011	2012
S1 vs. S2	Sig.	0,863	0,155	1,000	0,427	0,610
	Dif. Med.	-6,913	-12,502	4,512	5,441	-3,049
S1 vs. S3	Sig.	0,076	0,568	1,000	-	-
	Dif. Med.	-14,716	8,332	-4,249	-	-
S2 vs. S3	Sig.	0,655	0,005	0,302	-	-
	Dif. Med.	-7,803	20,834	-8,761	-	-

Cuadro 6.33: Comparaciones por pares del Factor sesión - Valores de significancia y diferencia entre medias - Variable de Respuesta: OutScope

De todas las comparaciones realizadas, la única comparación que resulta significativa es la de S2 vs. S3, en donde S2 resulta ser más efectiva que S3 en un 20,834 % (porcentaje bastante alto). Esta tendencia no se comparte por el resto de las replicaciones, en donde S3 resulta más efectiva que S2. Esto se puede visualizar en el gráfico de la figura 6.14.

6.2.3.3. Pruebas de Hipótesis: VR: OutScope - Factor Grupo

Al igual que para la variable de respuesta InScope, se dividen las replicaciones en dos subgrupos de replicaciones (los mismos que para InScope) que comparten los mismos tipos de grupos, realizando un análisis por separado en cada uno de ellos. En el cuadro 6.34 se presentan las estimaciones de medias, en el cuadro 6.35 se muestran los resultados de las comparaciones por pares del **Subgrupo A** de replicaciones y en la figura 6.15 se muestran los gráficos de línea.

Hubieron únicamente dos comparaciones que mostraron resutados significativos: LAS-CD-PCE vs. LAS-PCE-CD y LAS-CD-PCE vs. La comparación de LAS-CD-PCE vs. LAS-PCE-CD solamente presenta resultados significativos para la replicación de 2007 con una significancia de 0,006. Sin embargo, la tendencia que marca esta replicación, en donde la efectividad del grupo LAS-CD-PCE es mayor que la del grupo LAS-PCE-CD, no se mantiene en las otras dos replicaciones, en donde la efectividad de LAS-CD-PCE es menor que la de LAS-PCE-CD.

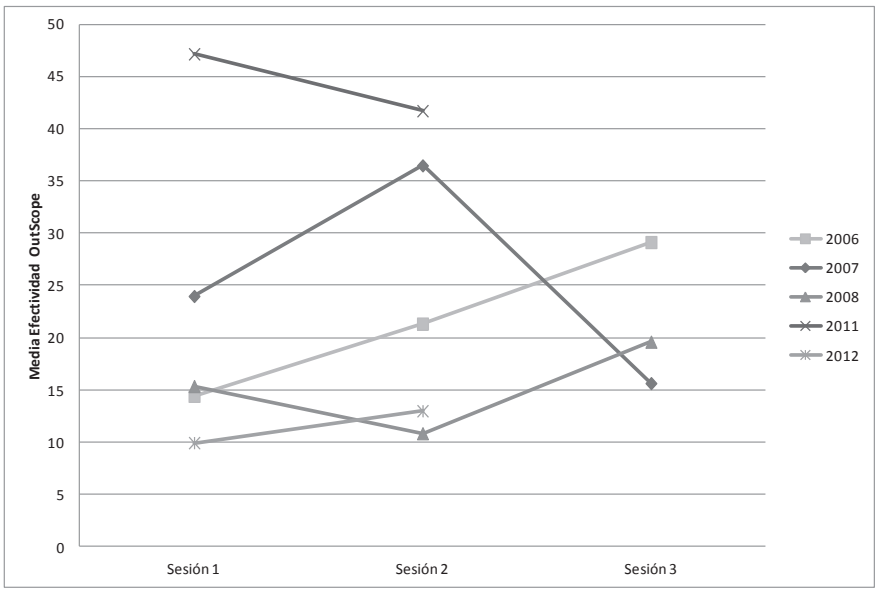

Figura 6.14: Gráfico de líneas para el factor Sesión - VR: OutScope

Grupo	**2006**	**2007**	**2008**
LAS-CD-PCE	10,704	40,820	0,984
LAS-PCE-CD	31,481	6,845	48,705
CD-LAS-PCE	13,900	22,979	-
CD-PCE-LAS	25,157	18,811	8,302
PCE-LAS-CD	34,890	27,786	21,698
PCE-CD-LAS	13,561	34,837	-

Cuadro 6.34: Medias Estimadas del Factor Grupo A - Variable de Respuesta: OutScope

Para la comparación de LAS-CD-PCE vs. PCE-LAS-CD también se tienen resultados significativos para una sola replicación (2008) con 0,002 de significancia en donde LAS-CD-PCE <PCE-LAS-CD. Sin embargo, esta tendencia tampoco se mantiene a lo largo de todas las replicaciones de este subgrupo, siendo que en 2007 la tendencia es opuesta siendo LAS-CD-PCE >PCE-LAS-CD.

En el cuadro 6.37 se presentan las comparaciones por pares para las replicaciones que conforman el **Subgrupo B** y en la figura 6.16 el gráfico de líneas.

No hubieron resultados significativos para este subgrupo de replicaciones en lo que refiere al factor grupo. Además, las tendencias resultan opuestas en ambas replicaciones.

Tomando en cuenta los resultados obtenidos en los dos subgrupos de replicaciones, se deduce que el factor grupo parecería no tener influencia en la efectividad que refiere a las faltas que están fuera del alcance de las técnicas, ya que pocas comparaciones han dado resultados significativos y los mismos no se mantienen a lo largo de todas las replicaciones.

Grupos	Valores	Replicaciones		
		2006	**2007**	**2008**
LAS-CD-PCE vs. LAS-PCE-CD	Sig.	0,116	0,006	0,124
	Dif. Med.	-20,778	33,975	-42,721
LAS-CD-PCE vs. CD-LAS-PCE	Sig.	1,000	0,633	-
	Dif. Med.	-3,196	17,842	-
LAS-CD-PCE vs. CD-PCE-LAS	Sig.	1,000	0,306	0,318
	Dif. Med.	-14,454	22,009	-7,318
LAS-CD-PCE vs. PCE-LAS-CD	Sig.	0,113	1,000	0,002
	Dif. Med.	-24,186	13,034	-20,714
LAS-CD-PCE vs. PCE-CD-LAS	Sig.	1,000	1,000	-
	Dif. Med.	-2,858	5,983	-
LAS-PCE-CD vs. CD-LAS-PCE	Sig.	0,478	1,000	-
	Dif. Med.	17,582	-16,134	-
LAS-PCE-CD vs. CD-PCE-LAS	Sig.	1,000	1,000	0,659
	Dif. Med.	6,324	-11,966	40,403
LAS-PCE-CD vs. PCE-LAS-CD	Sig.	1,000	0,465	1,000
	Dif. Med.	-3,409	-20,941	27,007
LAS-PCE-CD vs. PCE-CD-LAS	Sig.	0,682	0,055	-
	Dif. Med.	17,920	-27,992	-
CD-LAS-PCE vs. CD-PCE-LAS	Sig.	1,000	1,000	-
	Dif. Med.	-11,258	4,167	-
CD-LAS-PCE vs. PCE-LAS-CD	Sig.	0,158	1,000	-
	Dif. Med.	-20,991	-4,808	-
CD-LAS-PCE vs. PCE-CD-LAS	Sig.	1,000	1,000	-
	Dif. Med.	0,338	-11,859	-
CD-PCE-LAS vs. PCE-LAS-CD	Sig.	1,000	1,000	0,429
	Dif. Med.	-9,733	-8,975	-13,396
CD-PCE-LAS vs. PCE-CD-LAS	Sig.	1,000	1,000	-
	Dif. Med.	11,596	-16,026	-
PCE-LAS-CD vs. PCE-CD-LAS	Sig.	0,346	1,000	-
	Dif. Med.	21,329	-7,051	-

Cuadro 6.35: Comparaciones por pares del Factor Grupo(A) - Valores de significancia y diferencia entre medias - Variable de Respuesta: OutScope

Grupo	2011	2012
CD-PCE	39,638	12,724
PCE-CD	49,590	10,160

Cuadro 6.36: Medias Estimadas del Factor Grupo B - Variable de Respuesta: OutScope

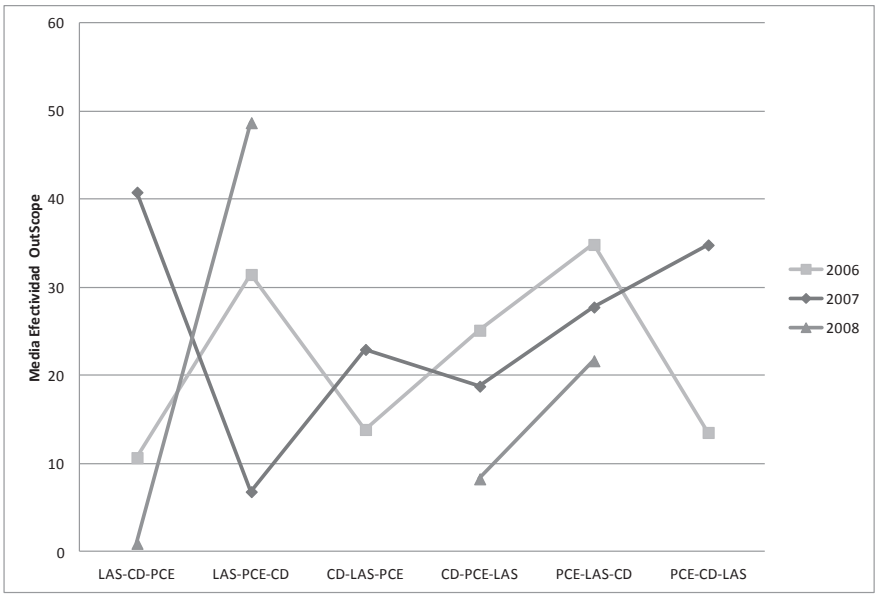

Figura 6.15: Gráfico de líneas para el factor Grupo - Subgrupo A - VR: OutScope

Grupos	**Valores**	**Replicaciones**	
		2011	**2012**
CD-PCE vs. PCE-CD	Sig.	0,155	0,680
	Dif. Med.	-9,952	2,564

Cuadro 6.37: Comparaciones por pares del Factor Grupo (B) - Valores de significancia y diferencia entre medias - Variable de Respuesta: OutScope

6.3. Variable de Respuesta: AllFaults

En esta última sección se presenta el análisis de la variable de respuesta AllFaults.

6.3.1. Estadísticas Descriptivas - AllFaults

En el cuadro 6.38 se presentan las estadísticas descriptivas para el factor Técnica, en la figura 6.17 los gráficos de boxplot. Para las replicaciones 2006 a 2008 que utilizan LAS, ésta última resulta en general más efectiva que CD y PCE. Entre CD y PCE, todas las replicaciones registran mayor efectividad para CD que para PCE. Observando la desviación estándar, PCE parecería ser la más estable respecto de las otras dos técnicas. En ESPEL 2012 se registran bajas medidas de efectividad para las técnicas CD y PCE.

En el cuadro 6.39 se presentan las estadísticas descriptivas para el factor Programa/Sesión, en la figura 6.18 se presentan los gráficos de boxplot. Al igual que para el resto de las variables de respuesta, se visualiza menos efectividad para el programa cmdline que para el resto de los programas. La replicación de ESPEL 2012 presenta

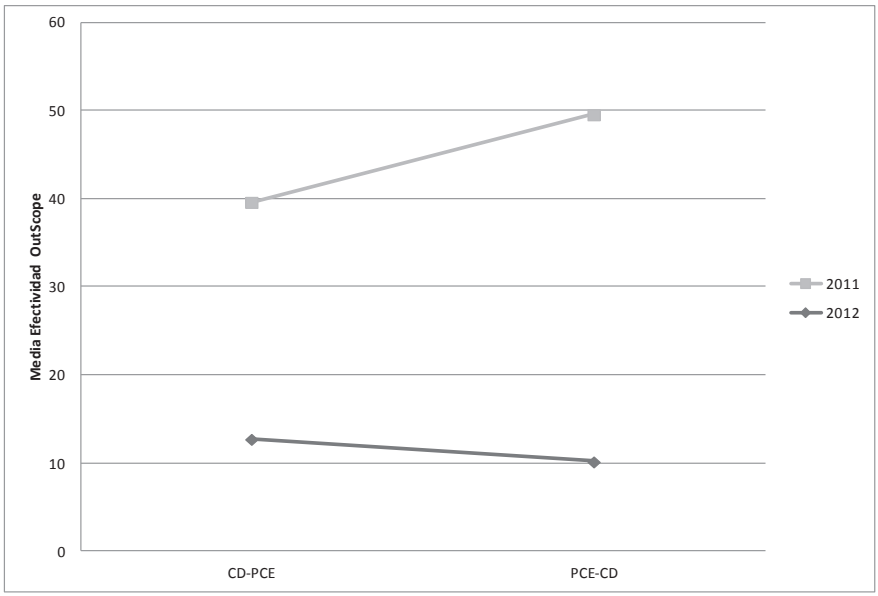

Figura 6.16: Gráfico de líneas para el factor Grupo - Subgrupo B- VR: OutScope

Técnica	Replicación	# de Obs.	Media	Desv. Estándar
LAS	2006	46	53,624	23,017
	2007	48	52,778	21,008
	2008	46	40,579	16,350
CD	2006	46	47,464	16,841
	2007	48	40,972	23,559
	2008	42	42,857	17,709
	2011	22	55,303	20,821
	2012	23	29,710	25,104
PCE	2006	46	47,101	17,676
	2007	48	38,888	15,503
	2008	46	39,493	19,360
	2011	22	46,970	17,547
	2012	23	28,260	17,720

Cuadro 6.38: Estadísticas Descriptivas - VR: AllFaults - Factor Técnica

muy baja efectividad para los programas nametbl y ntree.

En los cuadros 6.40 y 6.41 se presentan los estadísticos descriptivos para el Subgrupo A (2006 a 2008) y Subgrupo B (2011 y 2012) de replicaciones respectivamente, en la figura 6.19 se presentan los gráficos de boxplot. Al igual que en el resto de las replicaciones, se observa una baja efectividad en la replicación de ESPEL 2012.

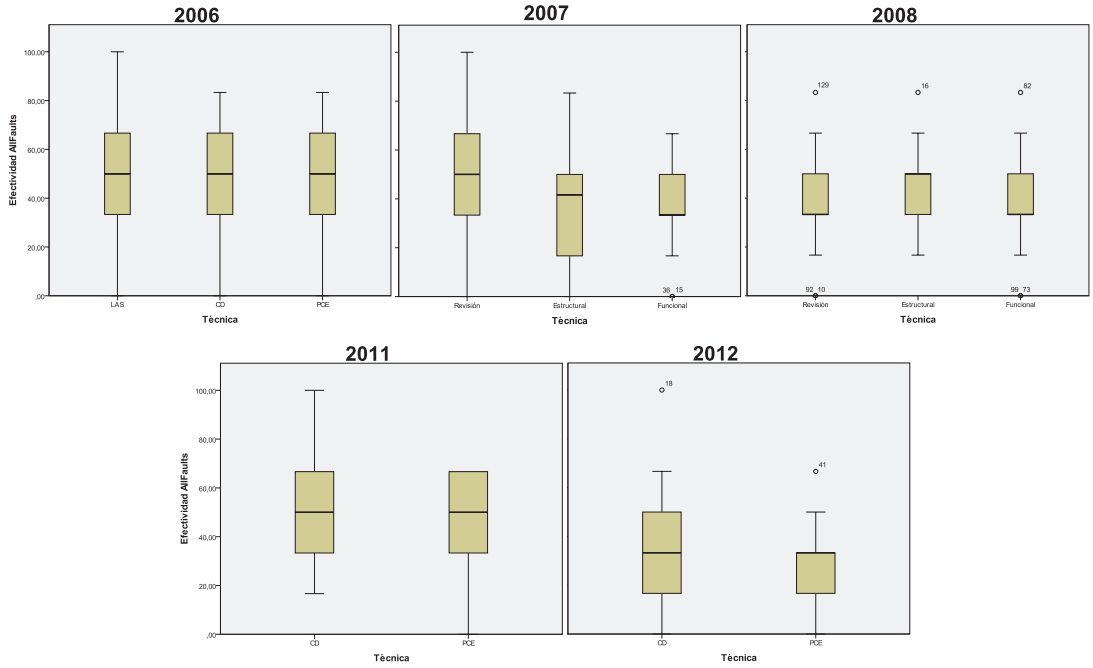

Figura 6.17: Gráficos de de caja y bigote - VR: AllFaults - Factor Técnica

Programa/Sesión	Replicación	# de Obs.	Media	Desv. Estándar
cmdline	2006	46	40,942	21,288
	2007	48	37,153	18,591
	2008	45	31,481	13,863
nametbl	2006	46	53,261	13,889
	2007	48	53,125	19,647
	2008	46	45,290	15,973
	2011	22	54,546	16,413
	2012	23	34,057	17,025
ntree	2006	46	53,985	19,934
	2007	48	42,361	22,003
	2008	43	46,124	19,531
	2011	22	47,727	22,001
	2012	23	23,914	24,528

Cuadro 6.39: Estadísticas Descriptivas - VR: AllFaults - Factor Programa

6.3.2. Reducción del Conjunto de Datos - AllFaults

El análisis de los outliers identificados para la variable AllFaults determinó que todos tienen origen confiable y por tanto no deben quitarse del conjunto de datos para su posterior análisis.

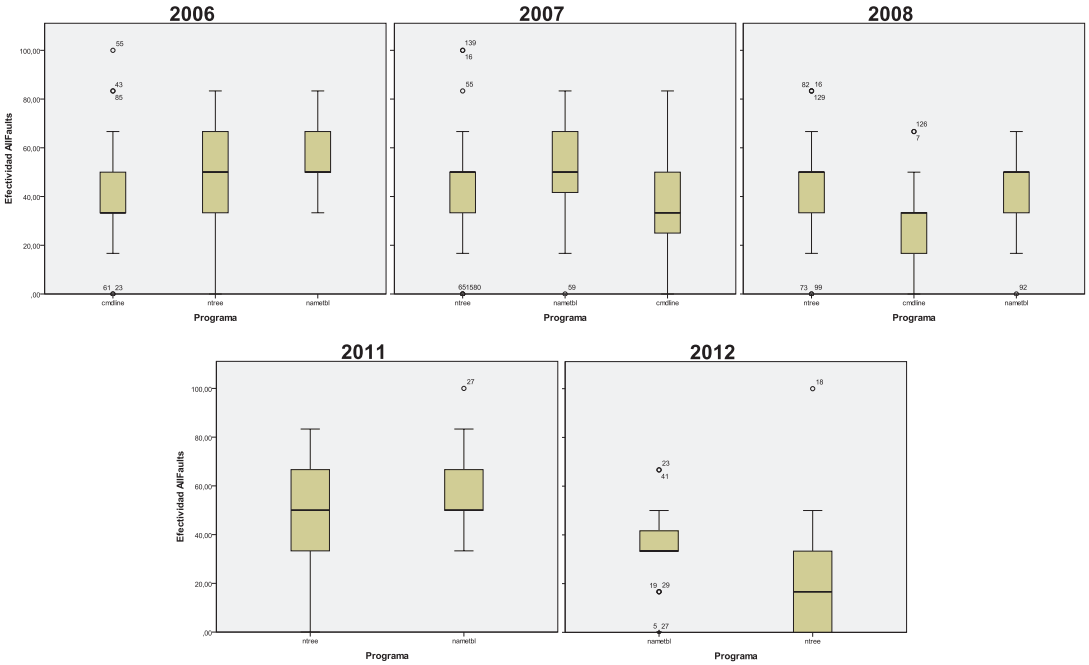

Figura 6.18: Gráficos de de caja y bigote - VR: AllFaults - Factor Programa

Grupo	Replicación	# de Obs.	Media	Desv. Estándar
LAS-CD-PCE	2006	24	47,917	19,851
	2007	24	53,472	24,067
	2008	44	40,151	15,798
LAS-PCE-CD	2006	24	48,611	23,529
	2007	24	40,973	23,041
	2008	–	–	–
CD-LAS-PCE	2006	27	53,086	17,926
	2007	24	42,361	21,411
	2008	X	X	X
CD-PCE-LAS	2006	27	43,826	19,693
	2007	24	37,499	15,735
	2008	42	41,269	16,560
PCE-LAS-CD	2006	18	49,999	21,389
	2007	24	45,140	18,702
	2008	45	41,111	20,902
PCE-CD-LAS	2006	18	54,631	11,154
	2007	24	45,833	20,996
	2008	X	X	X

Cuadro 6.40: Estadísticas Descriptivas - VR: AllFaults - Factor Grupo (A)

Grupo	Replicación	# de Obs.	Media	Desv. Estándar
CD-PCE	2011	24	52,083	16,531
	2012	24	25,000	20,264
PCE-CD	2011	20	50,000	22,942
	2012	22	33,332	22,420

Cuadro 6.41: Estadísticas Descriptivas - VR: AllFaults - Factor Grupo (B)

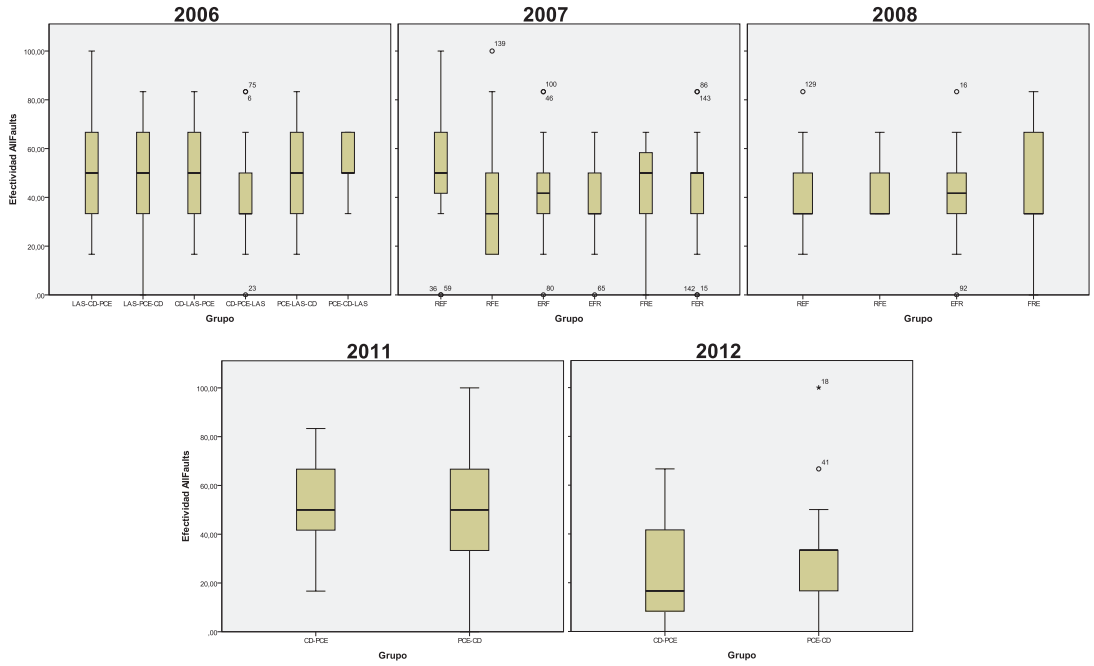

Figura 6.19: Gráficos de de caja y bigote - VR: AllFaults - Factor Grupo

6.3.3. Pruebas de Hipótesis - AllFaults

Análogamente al resto de las variables de respuesta, se realiza un chequeo de la validez del modelo mediante la realización de pruebas de normalidad para los residuos. Los resultados se muestran en el cuadro 6.42, en los que se puede observar que se validan los modelos para todas las replicaciones a excepción de la replicación de 2012.

Replicación	Shapiro-Wilk	Kolmogorov-Smirnov
2006	0,200	0,838
2007	0,200	0,234
2008	0,028	0,118
2011	0,200	0,954
2012	0,008	0,000

Cuadro 6.42: Resultados de pruebas de Normalidad para residuos - VR: AllFaults

Se realizan transformaciones para la replicación de 2012, intentando buscar una mejor significación de normalidad para los residuos. En el cuadro 6.43 se muestran los valores de significancia obtenidos para cada una.

Transformación	**2012**	
	K-S	**S-W**
y^2	0,000	0,000
$\sqrt{y}$	0,005	0,082
$\ln(y)$	0,200	0,237
$\ln(y+\frac{1}{2})$	0,000	0,000
$\log_{10}(y)$	,200	0,237
$\log_{10}(y+\frac{1}{2})$	0,000	0,000
$\frac{1}{y}$	0,053	0,034
$\frac{1}{\sqrt{y}}$	0,200	0,159
$\exp(y)$	0,000	0,000

Cuadro 6.43: Resultados de pruebas de Normalidad para las transformaciones realizadas - VR: AllFaults

Ya que varias transformaciones presentaron normalidad de residuos, se elije una de las que tiene mejor valor de significancia, para este caso $\ln(y)$. Al igual que para la variable OutScope, en aquellos análisis que se usen variables transformadas se presentarán los valores de significancia obtenidos con la transformación, pero los resultados de estimaciones de medias, diferencias de medias y otros se realizarán aplicando la función anti-transformada, en este caso $\exp(y)$.

En el cuadro 6.44 se muestran los niveles de significancia obtenidos en los análisis para la variable de respuesta AllFaults, para todos los factores estudiados.

Factores Fijos	**Replicaciones**				
	2006	**2007**	**2008**	**2011**	**2012**
Técnica	0,160	0,000	0,551	0,041	0,749
Programa-Sesión	0,000	0,000	0,000	0,080	0,262
Grupo	0,518	0,200	0,850	0,774	0,312

Cuadro 6.44: Niveles de Significancia de Análisis estadístico para todas las replicaciones - Variable de Respuesta: AllFaults

Para el factor Técnica se han encontrado diferencias significativas para la replicación de 2011, no así para las replicaciones de 2006, 2007, 2008 y 2012. Para el factor Programa/Sesión se han encontrado diferencias significativas para las replicaciones de 2006, 2007 y 2008 y no significativas las de 2011 y 2012. Por último, para el factor Grupo no se han encontrado diferencias significativas en ninguna replicación.

En las secciones siguientes se presentan las comparaciones por pares realizadas para el factor Técnica y Programa/Sesión, ya que el Grupo no mostró diferencias significativas en ninguna replicación.

6.3.3.1. Pruebas de Hipótesis: VR: AllFaults - Factor Técnica

En los cuadros 6.45 y 6.46 se presentan las medias estimadas y los resultados de significancia y diferencia de medias del factor técnica para la variable AllFaults, en la figura 6.20 se presenta el gráfico de líneas correspondiente.

Técnica	2006	2007	2008	2011	2012
LAS	54,061	52,778	41,675	-	-
CD	48,462	40,972	44,280	55,555	32,439
PCE	46,415	38,888	40,929	46,528	30,856

Cuadro 6.45: Medias Estimadas del Factor Programa - Variable de Respuesta: AllFaults

Técnicas	Valores	Replicaciones				
		2006	2007	2008	2011	2012
LAS vs. CD	Sig.	0,430	0,011	1,000	-	-
	Dif. Med.	5,598	11,806	-2,605	-	-
LAS vs. PCE	Sig.	0,180	0,000	1,000	-	-
	Dif. Med.	7,645	13,809	0,746	-	-
CD vs. PCE	Sig.	1,000	1,000	0,947	9,027	0,749
	Dif. Med.	2,047	2,084	3,351	0,041	1,583

Cuadro 6.46: Comparaciones por pares del Factor técnica - Valores de significancia y diferencia entre medias - Variable de Respuesta: AllFaults

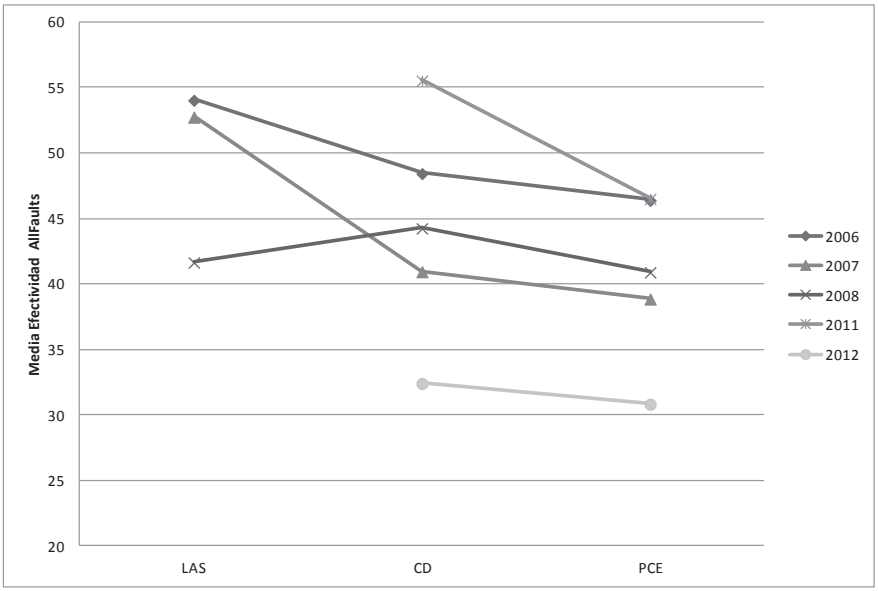

Figura 6.20: Gráfico de líneas para el factor Técnica - VR: AllFaults

Para la comparación entre LAS y CD se obtienen diferencias significativas para la replicación de 2007, en donde LAS resulta más efectiva que CD, al igual que en la replicación de 2006 (aunque esta última no resultó significativa). La única replicación que difiere en tendencia es la de 2008, aunque la diferencia de medias entre LAS y CD resulta bastante pequeña (2,65 %) y no significativa.

Entre las técnicas de LAS y PCE, los resultados muestran diferencias significativas para la replicación de 2007, en donde la tendencia indica que LAS es más efectiva que PCE. Esta tendencia se mantiene también para el resto de las replicaciones que no resultaron significativas.

Por último, para la comparación entre CD y PCE no se encuentran diferencias significativas en ninguna de las replicaciones. Las diferencias entre medias son pequeñas y con igual tendencia para todas las replicaciones.

6.3.3.2. Pruebas de Hipótesis: VR: AllFaults - Factor Programa

En los cuadros 6.47 y 6.48 se muestran los resultados de medias estimadas y de significancia de las comparaciones por pares para el factor Programa/Sesión, desde el punto de vista del programa, en la figura 6.21 se presenta el gráfico de líneas correspondiente.

Técnica	**2006**	**2007**	**2008**	**2011**	**2012**
cmdline	41,034	36,811	32,561	-	-
nametbl	54,088	52,670	46,122	54,861	34,591
ntree	53,817	43,157	48,202	47,222	28,937

Cuadro 6.47: Medias Estimadas del Factor Programa - Variable de Respuesta: AllFaults

Programas	**Valores**	**Replicaciones**				
		2006	**2007**	**2008**	**2011**	**2012**
cmdline vs. nametbl	Sig.	0,002	0,000	0,000	-	-
	Dif. Med.	-13,054	-15,859	-13,561	-	-
cmdline vs. ntree	Sig.	0,002	0,201	0,000	-	-
	Dif. Med.	-12,782	-6,346	-15,641	-	-
nametbl vs. ntree	Sig.	1,000	0,020	1,000	0,080	0,262
	Dif. Med.	0,271	9,513	-2,080	7,639	5,655

Cuadro 6.48: Comparaciones por pares del Factor programa - Valores de significancia y diferencia entre medias - Variable de Respuesta: AllFaults

Para la comparación entre cmdline y nametbl se encontraron diferencias significativas para las replicaciones de 2006, 2007 y 2008 (que comprenden todas las replicaciones en que se utilizó el programa cmdline), la tendencia es la misma para todas las replicaciones, en donde la efectividad de las técnicas para cmdline resulta inferior a la efectividad en nametbl en un 14,158 % en promedio.

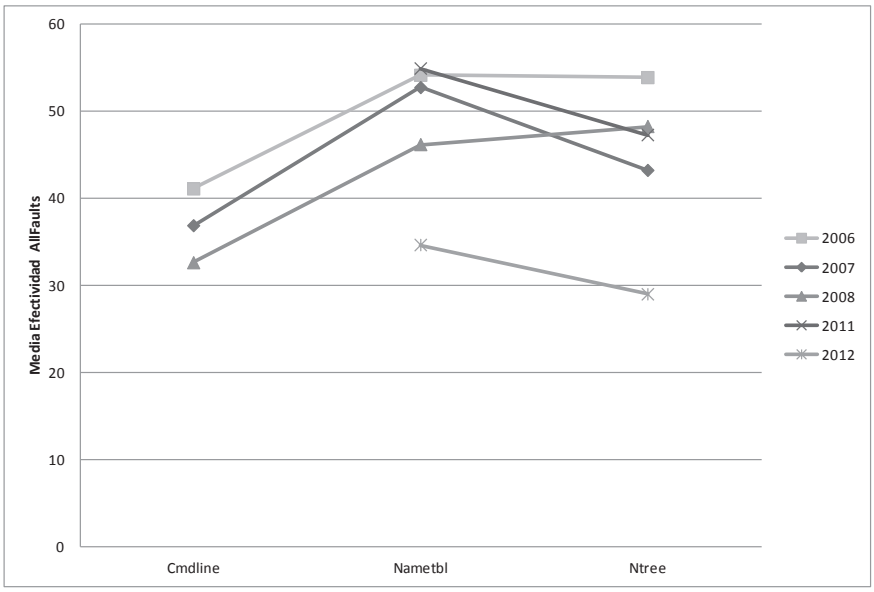

Figura 6.21: Gráfico de líneas para el factor Programa - VR: AllFaults

Entre cmdline y ntree los resultados muestran diferencias significativas para las replicaciones de 2006 y 2008, en donde la tendencia muestra mayor efectividad sobre el programa ntree que sobre cmdline. A pesar de que la replicación de 2007 no resulta significativa, la tendencia se mantiene. La efectividad sobre ntree resulta un 11,59 % mayor que en cmdline en promedio.

Por último, para la comparación entre nametbl y ntree, la única replicación que muestra diferencias significativas es la de 2007, siendo las técnicas más efectivas sobre el programa nametbl que sobre ntree en un 9,513 %. Para las demás replicaciones, la mayoría conserva la misma tendencia a excepción de la 2008, en donde la efectividad en ntree resulta mayor que en nametbl en un 2,080 %.

Los resultados parecen ser bastante concluyentes en lo que refiere a las comparaciones con cmdline, en donde la efectividad de las técnicas resulta inferior que sobre el resto de los programas. Entre nametbl y ntree, a pesar de que la tendencia se mantiene en la mayoría de las replicaciones, la diferencia no resulta ser tan pronunciada.

Cambiando el enfoque hacia la sesión, en el cuadro 6.49 se muestran las estimaciones de medias y en el cuadro 6.50 se presentan los resultados de significancia de las comparaciones por pares para el factor Programa/Sesión, desde el punto de vista de la sesión. En la figura 6.22 se presenta el gráfico de líneas correspondiente.

Sesión	**2006**	**2007**	**2008**	**2011**	**2012**
Sesión 1	41,034	43,157	48,202	47,222	34,591
Sesión 2	53,817	52,670	32,561	54,861	28,937
Sesión 3	54,088	36,811	46,122	-	-

Cuadro 6.49: Medias Estimadas del Factor Sesión - Variable de Respuesta: AllFauls

Sesiones	Valores	Replicaciones				
		2006	**2007**	**2008**	**2011**	**2012**
S1 vs. S2	Sig.	0,002	0,020	0,000	0,080	0,262
	Dif. Med.	-12,782	-9,513	15,641	-7,639	5,655
S1 vs. S3	Sig.	0,002	0,201	1,000	-	-
	Dif. Med.	-13,054	6,346	2,080	-	-
S2 vs. S3	Sig.	1,000	0,000	0,000	-	-
	Dif. Med.	-0,271	15,859	-13,561	-	-

Cuadro 6.50: Comparaciones por pares del Factor sesión - Valores de significancia y diferencia entre medias - Variable de Respuesta: AllFaults

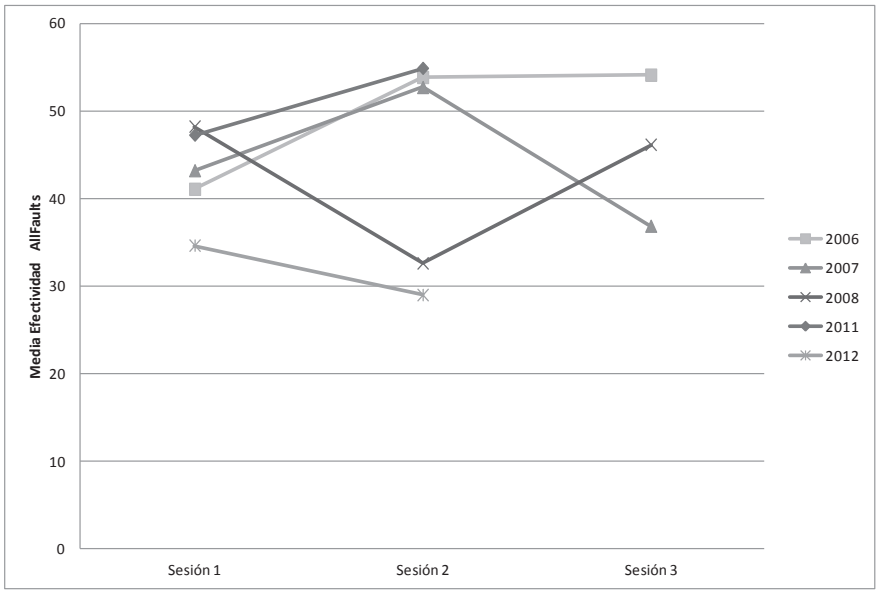

Figura 6.22: Gráfico de líneas para el factor Sesión - VR: AllFaults

Las replicaciones de 2006 y 2008 son las únicas que muestran resultados significativos para las comparaciones entre S1 y S2. Las tendencias entre ellas son opuestas (en 2006 la efectividad de S1 es menor que en S2 y en 2008 al revés). De igual forma la tendencia varía de una replicación a otra en aquellas replicaciones que no resultaron significativas.

Entre S1 y S3, resulta significativa solamente la replicación de 2006, mostrando mayor efectividad en S3. La tendencia no se mantiene en las replicaciones que no resultan significativas, siendo S1 más efectiva que S3.

Por último, para las comparaciones entre S2 y S3, las replicaciones de 2007 y 2008 muestran diferencias significativas con tendencias opuestas: en 2007 S2 resulta más efectiva que S3 y en 2008 S3 resulta más efectiva que en S2. En 2006 la diferencia resulta mínima.

Tomando en cuenta ambos puntos de vista (sesión y programa) los resultados parecen ser concluyentes en que la variable que más influye en la efectividad de las técnicas

es el programa y no la sesión. En las comparaciones entre programas los resultados se mantienen a lo largo de las replicaciones, cosa que no sucede con las sesiones que varían en gran medida de una replicación a otra.

Capítulo 7

Discusión de los Resultados del Análisis

En este capítulo se realiza un análisis más crítico de los resultados presentados en las secciones anteriores. El objetivo es comparar los resultados obtenidos entre las distintas variables de respuesta y además inferir posibles relaciones causales en el comportamiento de los factores. Las amenazas a la validez son un aspecto importante a mencionar sobre los resultados obtenidos.

Para el análisis e interpretación de los resultados de los **factores fijos**, se calcula una relación entre los niveles de cada factor, en base a los resultados de las comparaciones por pares de todas las replicaciones que se presentaron anteriormente en este capítulo. Dicha relación denota si la efectividad resulta mayor o menor entre dos niveles del factor y se conforma de acuerdo a las diferencias significativas y las tendencias observadas en el conjunto de todos los resultados. La definición de dicha relación está hecha por la autora, pero bien podría ser otra en caso de querer ser más estricto/laxo.

El cálculo de la relación entre niveles se realiza de la siguiente forma:

- $Nivel_1 > Nivel_2$ si se cumple que: al menos existe 1/3 de replicaciones con diferencias significativas y al menos 1/3 de las replicaciones restantes conservan la misma tendencia ($Nivel_1 > Nivel_2$)
- $Nivel_1 \geq Nivel_2$ si se cumple que: al menos existe 1/3 de replicaciones significativas con tendencia $Nivel_1 \geq Nivel_2$, pero existe como máximo 1/3 de replicaciones significativas con tendencia opuesta
- $Nivel_1 = Nivel_2$ si se cumple al menos una de las siguientes condiciones:
    1. No hay ninguna replicación con diferencia significativa (conserven o no igual tendencia)
    2. No hay ninguna replicación significativa y hay replicaciones con tendencia opuesta
    3. Existen replicaciones significativas, pero entre ellas la tendencia no se mantiene

El cálculo se realiza tomando en cuenta las replicaciones con las cuales se logró validar el modelo, ya sea con la variable respuesta original o con una transformación. Los modelos no validados si bien no forman parte de la definición de la relación, se observará si éstos mantienen los mismos resultados.

Si bien este tipo de discusión se puede organizar de múltiples formas según el punto de vista en el cual se quiera hacer foco, lo organizaremos por factores siguiendo el estilo de los análisis, pero intentando abordar múltiples puntos de vista. En la sección 7.1 se presenta la discusión e interpretación de resultados del factor Técnica, en la sección 7.2 la del factor Programa/Sesión y en la sección 7.3 la del factor Grupo.

7.1. Discusión de Resultados: Factor Técnica

En el cuadro 7.1 se encuentra un resumen del análisis de todas las variables de respuesta para el factor **técnica**: el resultado del cálculo de relación observada, cantidad de replicaciones significativas sobre el total de replicaciones, promedio de diferencia de medias de todas las replicaciones y la tendencia observada.

Variable Respuesta	**Relación Obs.**	**Rep. Sig.**	**Dif. Medias promedio**	**Tendencia**
InScope	LAS<CD	1/2	3,695 %	Tendencias iguales en rep. significativas.
	LAS<PCE	1/2	17,310 %	Tendencias iguales para todas las rep.
	CD<PCE	1/3	9,354 %	Tendencias iguales para todas las rep.
OutScope	CD>PCE	2/4	13,683 %	Tendencias iguales para todas las rep.
AllFaults	LAS>CD	1/3	4,933 %	Tendencias iguales en rep. significativas.
	LAS>PCE	1/3	7,400 %	Tendencias iguales para todas las rep.
	CD=PCE	0/5	1,821 %	Tendencias iguales para todas las rep.

Cuadro 7.1: Relaciones y tendencias Observadas - Factor Técnica

Para la VR InScope existe una relación clara pero no muy fuerte entre LAS y CD ya que de las 2 replicaciones que se tomaron para el cálculo (2006 y 2007) solo una resultó significativa (2006). De todas formas, la replicación de 2008 confirma la relación establecida (siendo significativa), a pesar de no haberse tenido en cuenta para el cálculo por no haberse podido validar el modelo. Para las comparaciones entre LAS vs. PCE y CD vs. PCE la relación resulta ser más clara, teniendo la misma tendencia para todas las replicaciones. La replicación de 2008 confirma la relación obtenida para LAS

y PCE (resultando significativa), siendo neutra para la relación de CD y PCE, ya que no resultó significativa.

En términos de implicancia práctica y de lo que se está midiendo respecto de esta variable de respuesta, los resultados parecen indicar que en relación a los defectos que están dentro del alcance de la técnica, PCE (que es una técnica basada en la especificación del programa) resulta más efectiva que LAS (técnica de revisión de código) y CD (técnica basada en la estructura del código), lo cual podría indicar que las técnicas basadas en la especificación y que no tienen acceso al código (aspecto en común entre CD y LAS) resultan más efectivas para los defectos que están dentro del alcance de las mismas). El porcentaje de diferencia resulta bastante alto, siendo que en un programa que contiene 100 defectos, PCE estaría encontrando 19 defectos más que LAS y 8 más que CD. En términos de lo que se espera de la calidad de un sistema y de la efectividad de las técnicas de verificación unitaria, resulta un porcentaje bastante alto.

En relación a la variable OutScope, la relación observada entre CD y PCE resulta contraria a lo observado en InScope. La relación es clara ya que todas las replicaciones conservan la misma tendencia y 2 de 4 resultaron significativas. La diferencia entre medias llega a más de un 13 % de diferencia, en donde CD es más efectiva que PCE. La replicación de 2006 que había quedado fuera del cálculo por no tener modelo válido también confirma esta relación, resultando significativa. Es interesante ver que la relación sea tan clara y opuesta a la VR InScope.

Una posible explicación para que CD resulte más efectiva que PCE es que al momento de ejecutar una técnica con acceso al código, el tester a su vez la combine con una revisión del mismo. Cuando se aplica una técnica estructural en donde se tiene acceso al código, el propio proceso de examinar el código buscando las combinaciones de entrada para las decisiones, podría dejar a la vista del tester otros defectos que en teoría estarían fuera del alcance de la técnica. Esto no es posible para la técnica de PCE ya que la misma no tiene acceso al código.

Para la VR AllFaults las relaciones resultan claras para las comparaciones con LAS, la cual resulta más efectiva que CD y PC en un 5 % y 7 % respectivamente. Entre CD y PCE no hay replicaciones significativas y las diferencias de medias registradas resultan menores. De estos resultados se deduce que, tomando en cuenta todos los defectos (los que están dentro del alcance de la técnica y los que no) la técnica estática de revisión resulta más efectiva que las dos técnicas dinámicas.

A diferencia de InScope, en donde PCE resultaba más efectiva que LAS, en AllFaults LAS resulta más efectiva que PCE. Esto podría estar denotando nuevamente una muy disminuida efectividad de PCE en relación a los defectos que están fuera de su alcance, ya que cuando éstos se toman en cuenta, la efectividad de PCE disminuye notoriamente.

En el cuadro 7.2 se presenta el promedio de las medias marginales estimadas de todas las replicaciones, para cada técnica y variable de respuesta. La diferencia más notoria se produce entre CD y PCE para la VR OutScope, teniendo CD el doble de efectividad en promedio que PCE.

Las técnicas CD y PCE presentan mayor efectividad para InScope que para OutScope, esto estaba previsto y de cierta forma comprueba que los defectos sembrados en InScope tienen mayor probabilidad de ser detectados que los defectos OutScope. Esta

Técnica	InScope	OutScope	AllFaults
LAS	48,311	-	49,505
CD	56,276	30,985	44,342
PCE	64,838	17,045	40,723

Cuadro 7.2: Promedio de medias marginales estimadas - Factor Técnica

capacidad es más notoria en PCE (47 % más efectiva) que en CD (25 % más efectiva). Aún así, el porcentaje de detección de defectos fuera del alcance de las técnicas no es despreciable (en el orden del 40 %), cuál es la causa de este alto porcentaje es un tema a investigar, que podría deberse a múltiples factores: la estrategia de la técnica en sí, que el sujeto complementa la estrategoa de la técnica, los defectos inyectados y otros.

7.2. Discusión de Resultados: Factor Programa/Sesión

En el cuadro 7.3 se muestran las relaciones observadas entre los distintos niveles de este factor, vistas desde el punto de vista del **programa**.

VR	Relación Obs.	Rep. Sig.	Dif. Medias promedio	Tendencia
InScope	cmdl<nmtbl	1/2	11,524	Tendencias iguales para todas las rep.
	cmdl<ntr	1/2	9,255	Tendencias iguales para todas las rep.
	nmtbl=ntr	0/3	2,269	Tendencias difieren entre replicaciones
OutScope	cmdl<nmtbl	1/2	14,798	Tendencias iguales para todas las rep.
	cmdl=ntr	0/2	6,422	Tendencias iguales para todas las replicaciones
	nmtbl=ntr	0/4	2,065	Tendencias iguales para todas las replicaciones
AllFaults	cmdl<nmtbl	3/3	14,158	Tendencias iguales para todas las replicaciones
	cmdl<ntr	2/3	11,590	Tendencias iguales para todas las rep.
	nmtbl=ntr	1/5	4,200	Tendencia difiere en 1 rep.

Cuadro 7.3: Relaciones y tendencias Observadas - Factor Programa/Sesión

En relación al programa cmdline, la efectividad sobre éste resulta menos efectiva que para nametbl para todas las variables de respuesta, con una relación clara y fuerte en donde esta tendencia se mantiene para todas las replicaciones y en todas las variables de respuesta (incluso para la replicación de 2008-InScope y para las replicaciones 2006-OutScope y 2012-OutScope que no es tenida en cuenta para el cálculo), observándose diferencias significativas en todas ellas. La efectividad sobre el programa cmdline resulta 11,524 %, 14,798 % y 14,158 % inferior a nametbl para InScope, OutScope y AllFaults respectivamente. Esto parecería indicar que no importa sobre qué tipo de defectos se mide la efectividad, ésta siempre resulta menor en cmdline que en nametbl.

El caso de la relación entre cmdline y ntree es muy parecido al anterior, solamente

que en la variable de respuesta OutScope la relación no es tan clara ya que no hay ninguna replicación que haya arrojado diferencias significativas. Sin embargo, la tendencia se mantiene para todas las replicaciones, siendo la efectividad más alta en ntree que en cmdline.

La relación entre nametbl y ntree resulta distinta, en donde para InScope y OutScope no hay diferencia de efectividad y para AllFaults parecería haber una pequeña diferencia en donde la efectividad es mayor en nametbl que en ntree, pero hay una sola replicación que lo confirma con diferencias significativas. Esto parecería indicar que la efectividad de las técnicas no varía entre nametbl y ntree para ninguna variable de respuesta.

En el cuadro 7.4 se presenta el promedio de las medias marginales estimadas de todas las replicaciones, para cada programa y variable de respuesta. Se observa la misma relación entre programas para todas las variables de respuesta: cmdline es el que presenta menor efectividad, seguido de nametbl y ntree, siendo más notoria la baja efectividad sobre cmdline. De acuerdo experiencia de los investigadores en la ejecución del experimento y la corrección de las pruebas de los estudiantes, se deduce que dicha diferencia se le atribuye a cmdline por tener una lógica más compleja y en consecuencia resultar más difícil de verificar por los sujetos que los otros dos programas. Este fenómeno ya se había percibido en las primeras replicaciones y debido a ésto se tomó la decisión de quitar cmdline de los programas a verificar para 2011 y 2012.

Programa	**InScope**	**OutScope**	**AllFaults**
cmdline	51,305	13,619	36,802
nametbl	62,532	27,360	48,466
ntree	56,785	24,147	44,267

Cuadro 7.4: Promedio de medias marginales estimadas - Factor Programa

Desde el punto de vista de la **sesión**, el resumen de los análisis y las relaciones observadas entre las diferentes sesiones se presentan en el cuadro 7.5.

Desde el punto de vista de la sesión (a diferencia del programa) no se encuentra ninguna relación clara entre las comparaciones de las sesiones, siendo la amplia mayoría de ellas iguales en términos de efectividad.

En el cuadro 7.6 se presenta el promedio de las medias marginales estimadas de todas las replicaciones, para cada sesión y variable de respuesta. Las diferencias observadas entre sesiones dentro de la misma variable de respuesta son pequeñas, en el orden de 1 o 2 puntos porcentuales.

Analizando el punto de vista de la sesión, queda más claro aún que de la combinación Programa-Sesión que se tiene para este factor, claramente es el efecto del programa el que genera diferencias significativas y no así la sesión. Resulta interesante destacar que este aspecto no podría haberse observado de no ser por la variación de la correspondencia de los programas a las sesiones en las distintas replicaciones. Si el mismo programa correspondiese siempre a la misma sesión en todas las replicaciones, no se podría haber diferenciado un efecto de otro.

VR	Relación Obs.	Rep. Sig.	Dif. Medias promedio	Tendencia
InScope	S1=S2	1/3	11,470	Tendencias difieren entre replicaciones
	S1=S3	0/2	4,443	Tendencias difieren entre replicaciones
	S2=S3	1/2	3,230	Tendencias difieren entre replicaciones
OutScope	S1=S2	0/4	1,013	Tendencias difieren entre replicaciones
	S1=S3	0/2	3,192	Tendencias difieren entre replicaciones
	S2=S3	1/2	6,515	Tendencias difieren entre replicaciones
AllFaults	S1=S2	2/5	1,727	Tendencias difieren entre replicaciones
	S1=S3	1/3	1,543	Tendencias difieren entre replicaciones
	S2=S3	2/3	0,676	Tendencias difieren entre replicaciones

Cuadro 7.5: Relaciones y tendencias Observadas - Factor Sesión

Programa	InScope	OutScope	AllFaults
Sesión 1	57,357	22,155	42,841
Sesión 2	58,271	24,657	44,569
Sesión 3	57,453	21,445	45,674

Cuadro 7.6: Promedio de medias marginales estimadas - Factor Sesión

7.3. Discusión de Resultados: Factor Grupo

Para el análisis del factor **grupo**, se dividen las replicaciones en los subgrupos A y B de acuerdo a los tipos de grupos que comparten como en las secciones anteriores de este capítulo. En el cuadro 7.7 se presentan los resultados de las relaciones observadas para el **subgrupo A**. El cuadro muestra de forma resumida todas las relaciones entre los distintos niveles del factor grupo, agrupando en una misma fila todas aquellas comparaciones con igual relación, a modo de simplificar la visualización (ya que son 15 comparaciones por cada VR).

Para la variable InScope, todas las comparaciones entre niveles del factor grupo resultan en una relación de igualdad, en donde las tendencias difieren y ninguna replicación resulta significativa. Esto mismo se repite para la variable AllFaults.

En OutScope la única comparación que no resulta en igualdad es la de LAS-PCE-CD y PCE-LAS-CD, en donde el segundo grupo resulta más efectivo que el primero. Sin embargo esta relación no es tan clara ya que de las 2 replicaciones que se toman para el cálculo, una de ellas muestra una tendencia opuesta. Además, esta relación no se observa en los resultados de las otras variables de respuesta, no encontrando motivo por el cual la efectividad en un grupo varíe de una VR a otra. Por tanto se considera este caso como un error de tipo I (un falso positivo).

En el cuadro 7.8, se presenta el resumen de los análisis del factor grupo para el **sugrupo B** de replicaciones. Debido a que no se pudo tomar la replicación de 2012 para el cálculo de la relación InScope (ya que no se pudo validar el modelo) únicamente

VR	Relación Obs.	Rep. Sig.	Dif. Medias promedio	Tendencia
InScope	$G_i = G_j$ $(i = 1 \cdots 6, j = 1 \cdots 5, i \neq j)$	0/2	-	Tendencias difieren entre replicaciones
OutScope	$G_1 = G_2$	1/2	6,598	Tendencias difieren entre replicaciones
	$G_2 \leq G_5$	1/2	5,576	Tendencias difieren entre replicaciones
	$G_i = G_j$ $(i = 1 \cdots 6, j = 1 \cdots 5, i \neq j, (i,j) \neq (1,2), (2,5))$	0/3	-	Tendencias difieren entre replicaciones
AllFaults	$G_i = G_j$ $(i = 1 \cdots 6, j = 1 \cdots 5, i \neq j)$	0/3	-	Tendencias difieren entre replicaciones

Cuadro 7.7: Relaciones y tendencias Observadas - Factor Grupo - Subgrupo A G_1 =LAS-CD-PCE, G_2 =LAS-PCE-CD, G_3 =CD-LAS-PCE, G_4 =CD-PCE-LAS, G_5 =PCE-LAS-CD, G_6 =PCE-CD-LAS

quedó la replicación de 2011 para este subgrupo. La relación obtenida en donde CD-PCE resulta más efectivo que PCE-CD no se comparte por la replicación de 2012 y debido a que los otras variables de respuesta no muestran diferencias entre los grupos, se asume un error de tipo I (falso positivo) y se asume una relación de igualdad para los grupos.

VR	Relación Obs.	Rep. Sig.	Dif. Medias promedio	Tendencia
InScope	$G_1 > G_2$	1/1	18,055	Una sola replicación
OutScope	$G_1 = G_2$	0/2	4,167	Tendencias difieren entre replicaciones
AllFaults	$G_1 = G_2$	0/2	6,249	Tendencias difieren entre replicaciones

Cuadro 7.8: Relaciones y tendencias Observadas - Factor Grupo - Subgrupo B G_1 =CD-PCE, G_2 =PCE-CD

En el cuadro 7.9 se presenta el promedio de las medias marginales estimadas de todas las replicaciones, para cada programa y variable de respuesta. Si bien en algunos casos las diferencias entre grupos pueden rondar los 10 puntos porcentuales, éstas diferencias no se mantienen en proporción a lo largo de las diferentes variables de respuesta.

El grupo representa el orden en el cual los sujetos aplican las técnicas en las sesiones y los resultados muestran que no hay diferencia de efectividad entre los grupos de forma clara y contundente. Esto indica que se descarta un posible efecto causado por el orden de aplicación de las técnicas sobre la efectividad de las mismas. Sumado a que los resultados del factor Programa/Sesión vistos desde el punto de vista de la sesión no influencian en la efectividad de las técnicas, estaríamos cerca de poder descartar un posible efecto de carry-over entre las sesiones, descartando un posible efecto de

Programa	**InScope**	**OutScope**	**AllFaults**
LAS-CD-PCE	63,338	17,503	47,080
LAS-PCE-CD	53,582	29,010	44,542
CD-LAS-PCE	64,805	18,440	48,360
CD-PCE-LAS	54,052	17,423	41,704
PCE-LAS-CD	54,086	28,125	45,598
PCE-CD-LAS	65,980	24,199	49,420
CD-PCE	56,250	26,181	40,531
PCE-CD	58,334	29,875	42,270

Cuadro 7.9: Promedio de medias marginales estimadas - Factor Grupo

aprendizaje, efecto de cansancio o efecto del orden de aplicación de las técnicas.

Es importante observar que en caso de haber existido una única replicación, no habría sido posible separar el efecto del programa del de la sesión. Es comparando el conjunto de varias replicaciones lo que nos permite observar los efectos por separado y poder concluir acerca ellos de forma independiente.

El efecto de carry-over en los experimentos se genera debido al diseño del mismo, cuando un sujeto aplica más de una vez un tratamiento o nivel de un factor. En el diseño de esta familia de experimentos, se intentó minimizar este efecto creando subgrupos de sujetos los cuales variaban el orden de aplicación de las técnicas. En base a los resultados obtenidos, se podría decir que el diseño fue exitoso con respecto a la minimización del efecto de carry-over. De igual forma, siempre existe la posibilidad de que este efecto sea anulado por otro no detectado.

Capítulo 8

Conclusiones y Trabajos a Futuro

Como parte de este trabajo se realiza un estudio y análisis de la familia de experimentos de UPM (en particular el diseño experimental para las replicaciones que se realizaron desde el año 2006 a 2012), en el cual se participa como replicador responsable en la replicación de UPM 2011, llevando a cabo la preparación y ejecución de la misma, con sus actividades posteriores de recolección de datos.

Tomando como base la familia de experimentos UPM (en particular el diseño experimental para las replicaciones que se realizaron desde el año 2006 a 2012), se genera un procedimiento de análisis estadístico para este diseño experimental, acompañado de una guía para el reporte de los resultados obtenidos.

El procedimiento sugerido para el análisis del experimento de UPM es utilizando la técnica de análisis del *modelo lineal mixto*, el cual se adapta al modelo matemático asociado al diseño experimental de UPM. La utilización de este procedimiento y el reporte de los resultados se detalla y ejemplifica con la replicación del año 2006. A partir de este trabajo se redacta el artículo "Effectiveness for Detecting Faults Within and Outside the Scope of Testing Techniques: A Controlled Experiment" el cual fue enviado al 7^o *International Symposium on Empirical Software Engineering and Measurment* y actualmente se encuentra pendiente de aprobación.

La replicación de 2012 fue llevada a cabo en la Universidad de ESPEL, Ecuador, llevada a cabo por los doctorandos Fonseca y Espinosa. Luego de que se obtuvieran los datos brutos resultantes de la replicación, colaboramos en la realización del análisis estadístico a estos datos e interpretación de los resultados, los cuales fueron publicados comparándolos con los resultados de la replicación de 2006 en el artículo "Effectiveness for Detecting Faults Within and Outside the Scope of Testing Techniques: An Independent Replication" el cual fue aceptado y será publicado en el *Special Issue on Experimental Replications* de la revista internacional *Empirical Software Engineering (EMSE)* [ADEF13].

El estudio de los diseños de medidas repetidas cross-over y los tipos de análisis estadísticos que se pueden aplicar a este tipo de diseño forman parte también de otro artículo en el cual se está colaborando. Del cual aún no está definido lugar de publicación.

Los resultados del análisis realizado al conjunto de replicaciones muestran los siguientes hallazgos: Respecto de las técnicas:

- PCE resulta más efectiva que CD y LAS para las faltas InScope, mientras que no existen diferencias entre CD y LAS.
- CD resulta más efectiva que PCE para las faltas que están fuera del alcance de las técnicas.
- LAS resulta más efectiva que CD y PCE tomando en cuenta los dos tipos de faltas (InScope y OutScope)

Se presume que las diferencias observadas para PCE en InScope pueden deberse a que las técnicas basadas en especificación sean más efectivas al encontrar las faltas que están dentro de su alcance. Este aspecto sería interesante confirmarlo realizando otros experimentos variando la técnica de PCE por otra del mismo tipo.

Respecto del Programa/Sesión:

- La efectividad de las técnicas sobre cmdline es menor que sobre nambtbl y ntree para las faltas InScope, mientras que no hay diferencia entre nametbl y ntree.
- La efectividad sobre cmdline resulta menor que sobre nametbl y levemente menor que en ntree para las faltas OutScope.
- La efectividad sobre cmdline es menor que sobre nambtbl y ntree tomando en cuenta todas las faltas, mientras que no hay diferencia entre nametbl y ntree.
- La efectividad es la misma para todas las sesiones en todas las variables de respuesta.

Los resultados muestran que la sesión no tiene influencia sobre la efectividad siendo el factor programa el que sí influye. Es notoria la baja efectividad sobre el programa cmdline y se presume que dicha diferencia es resultado de una complejidad mayor del programa cmdline en comparación con los otros programas, tornándose más dificil de verificar por los sujetos.

Respecto del Grupo:

- La efectividad es la misma para todos los grupos en todas las variables de respuesta.

Todos estos resultados están condicionados y limitados al contexto y diseño experimental, los cuales limitan la generalización de los resultados a otros contextos, como ser:

Contexto Académico Los sujetos son estudiantes y no profesionales de la industria, que realizan las actividades de verificación en el marco de la aprobación de una asignatura que forma parte de su currícula y no como una actividad laboral.

Programas no reales Los programas son generados para el uso del experimento, escritos en lenguaje C y conteniendo faltas inyectadas de acuerdo a si están dentro o fuera del alcance de cada técnica. Cada falta tiene la peculiaridad de que si está dentro del alcance de la técnica CD se encuentra fuera del alcance de la técnica PCE.

Procedimiento de aplicación específico para cada técnica Se indica un método específico para la aplicación de las técnicas en el experimento, a modo de asegurar una correcta aplicación de la misma de acuerdo a sus premisas y una adecuada documentación de los fallos encontrados para luego generar los datos brutos.

La tarea de análisis para la elección de la técnica de análisis no es una tarea sencilla, más aún cuando los modelos a analizar son complejos. Se requiere de cierto conocimiento en el área de la estadística aplicada a la experimentación, la cual es muy amplia y maneja conceptos y términos que muchas veces son desconocidos por los investigadores en IS.

8.1. Lecciones Aprendidas

Las dificultades por las que se atraviesa en las actividades de ejecución y de análisis de los datos de cada replicación de los experimentos deja numerosas lecciones que son necesarias tomar en cuenta para futuros experimentos que permitan hacer evolucionar la línea de investigación. Muchas veces la falta de documentación de algunos aspectos del experimento o de las replicaciones llevan a cometer los mismos errores en las replicaciones futuras. A continuación mencionaremos las lecciones aprendidas más importantes tanto de la ejecución de la replicación de UPM 2011, como del análisis de resultados de todas las replicaciones

Hacer variaciones pequeñas en el diseño base del experimento para una replicación puede ocasionar que parte de los resultados no sean comparables con el resto de las replicaciones. Este es el caso de las replicaciones de 2011 y 2012 en donde se quita uno de los niveles del factor técnica (LAS) y uno de los niveles del factor Programa/Sesión (cmdline). Estos cambios afectaron en cascada no solamente a las comparaciones por pares de cada factor, sino también a que los niveles del factor Grupo fueran completamente distintos y muy difíciles de comparar con el resto de las replicaciones.

El diseño experimental del conjunto de replicaciones 2006-2012 es una evolución del diseño de las replicaciones anteriores de esta familia de experimentos (2000-2005) en donde se tenía un diseño factorial completo, en donde cada sujeto aplicaba una única vez cada nivel de cada factor. La evolución a un diseo cross-over fue impulsada por la obtención de una mayor cantidad de muestras y con ellas lograr aplicar técnicas estadísticas paramétricas que obtuvieran resultados más confiables. Con la introducción del cross-over se introdujo el problema de la combinación de efectos que se tiene del Programa con la Sesión, factor que se introdujo con el cambio de diseño y que no se visualizó de forma completa hasta no comenzar con los análisis estadísticos.

Lograr separar el efecto del programa del de la sesión fue posible debido a que se contaban con numerosas replicaciones que variaban la correspondencia programa-sesión, esto no habría sido posible de realizar analizando una única replicación. Aquí tenemos dos aspectos importantes a considerar: por un lado el beneficio extra de contar

con varias replicaciones cuyo análisis en conjunto ayuda a diferenciar los efecto de los factores programa y sesión, y por otro lado, la importancia de variar la correspondencia entre ambos factores para lograr diferenciar los efectos. Éste último aspecto no se tuvo en cuenta al momento del diseño del experimento y se dio de forma casual, que si bien se dio una buena variación de correspondencia, no se logró un cubrimiento completo de todas las posibles combinaciones programa-sesión, que sería lo ideal en este sentido.

Los diseños complejos son enemigos de los resultados claros y de interpretaciones no ambiguas. Dentro de casi cualquier área de investigación las realidades son complejas y fruto de múltiples efectos de variables que interactúan conjuntamente y de forma inter-dependiente sobre los fenómenos que se quieren estudiar. Querer reproducir la realidad fielmente en un ambiente controlado es súmamente difícil y se torna imposible su estudio, debido a las múltiples interacciones que deben ser estudiadas, analizadas e interpretadas.

Debido al problema de obtener gran cantidad de muestras para los experimentos y lo complejas que son muchas veces las relaciones, tener un experimento con más de 2 o 3 variables, cada una con 2 o 3 niveles ya se torna muy complejo de analizar. Tal es el ejemplo del factor Grupo para las replicaciones 2006-2008, que conformado por 6 alternativas (que a su vez son un efecto cascada de los 3 niveles del factor técnica distribuido entre las sesiones) generan 15 comparaciones por pares que resultan muy complejas de interpretar, más aún si se las combinaran con los niveles de los demás factores. Si los resultados en un solo factor no son claros y de fácil interpretación, menos aún podrá decirse de las interacciones que se generan con el resto de los factores.

Cambios en el tipo de diseño experimental muchas veces conllevan un cambio en el tipo de análisis que se puede realizar sobre los datos. El hecho de considerar el sujeto como factor aleatorio dentro del diseño experimental trae aparejado el efectuar un tipo de análisis que contemple los efectos aleatorios. Debido a esto se escoje el Análisis de Componentes de la Varianza aplicado a modelos mixtos como técnica de análisis estadístico.

Personalmente, creo que es mejor realizar experimentos con 1 o 2 factores, con 2 o 3 niveles en cada uno de ellos y no mucho más. Es preferible lograr interpretar primero de forma aislada el comportamiento de un factor y de los niveles que lo componen, para luego entender cómo éste interacciona con otros. Ésto debe realizarse de forma progresiva y sobre resultados que sean claros (para esto probablemente se requieran varias replicaciones). Incluso, los cambios inevitables de contexto de una replicación a otra introducen cambios y variaciones que deben ser tomados en cuenta al momento del análisis.

Cuando se introducen muchas variaciones a la vez, es difícil interpretar correctamente cuándo un cambio en la respuesta observada es resultado de cuál variación, o combinación de variaciones introducidas. Esto muchas veces trae controversia al momento de diseñar un experimento con pocos cambios o al realizar una replicación idéntica, ya que pareciera que el aporte generado fuera mínimo o nulo, o incluso que el investigador no se dio cuenta de aspectos interesantes a estudiar. Sin embargo, los resultados que se pueden obtener de este tipo de replicaciones es mucho más claro y ayuda a fortalecer o refutar resultados de estudios anteriores.

8.2. Trabajos a Futuro

Por un lado mencionaremos los trabajos a futuro relacionados con la familia de experimentos que fue objeto de este estudio. Planteando posibles cambios a realizar en el diseño experimental y otros aspectos. Por otro lado, mencionaremos otras líneas de investigación paralelas que complementarían y aportarían en la evolución de la línea de investigación en general

- **Cambios o sugerencias de mejora al diseño experimental de la familia de experimentos UPM**

  Si bien en las dos últimas replicaciones no se utilizó el programa cmdline, es claro que debería discontinuarse su uso al haberse detectado una complejidad y resistencia mayor por parte de los sujetos a este último.

  Cambiar las técnicas de PCE y CD por técnicas de su mismo tipo (basadas en la especificación y basadas en la estructura del programa) de similar complejidad. Esto ayudaría a confirmar si las diferencias encontradas para InScope y OutScope son compartidas por técnicas del mismo tipo o solamente se manifiestan para estas dos técnicas en particular. Creo recomendable realizar un cambio a la vez y ver el resultado en al menos una replicación identificando correctamente si las variaciones se deben al cambio introducido.

- **Investigaciones complementarias**

  Una de las amenazas a la validez mencionadas para este experimento es qué tan representativas de las faltas reales son las faltas sembradas en los programas y además, qué distribución real presentarían éstas en los mismos. Si la técnica A es 30 % más efectiva que la técnica B en las faltas InScope, pero se tiene que la distribución de faltas InScope para la técnica A es del 1 % del total de faltas sobre el programa y la distribución de las faltas InScope de la técnica B es de un 10 % en los programas... pregunta: ¿qué técnica resultaría más efectiva?

  De nada vale saber que una técnica es más efectiva sobre las faltas que están dentro de su alcance, cuando no sabemos qué cantidad o distribución de ese tipo de faltas existe en los programas. Un estudio paralelo que podría resultar de gran aporte, sería el análisis de una posible correspondencia entre las faltas InScope o OutScope a un conjunto de características, que nos permitan determinar un tipo de falta a nivel general (o generar un mapeo a tipos de faltas ya conocidos), de la cual se conozca (o sea posible conocer) su distribución en los programas.

  La distribución de faltas en un sistema depende de múltiples factores, como por ejemplo: tipo de sistema, metodologías y procesos de desarrollo y mantenimiento para el mismo, entre otras. Generalmente las organizaciones conocen la distribución de defectos en sus sistemas por datos históricos referentes al seguimiento y control de incidentes de los mismos, hay veces que utilizan taxonomías o categorizaciones de defectos conocidas y otras veces generan su propia categoría de defectos.

  Mapear las categorías de InScope y OutScope de cada técnica de interés a un conjunto de características o tipos de defectos de los cuales podría conocerse su

distribución sería un gran aporte en lo que refiere a la implicancia práctica de los resultados obtenidos acerca de la efectividad InScope y OutScope de cada técnica.

Los resultados del experimento podrían estar condicionados por la experiencia, motivación, formación y otras características de los sujetos. Resultaría interesante realizar otros experimentos en donde se explore sobre la influencia o no de estas variables y daría un marco para definir qué tan generalizables a otros contextos son los resultados obtenidos en experimentos que utilizan estudiantes como sujetos.

Bibliografía

[ABHL06] E Arisholm, L C Briand, S E Hove, and Y Labiche. The impact of UML documentation on software maintenance: an experimental evaluation. *Software Engineering, IEEE Transactions on*, 32(6):365–381, 2006. 2.2.3.3

[ADEF13] Cecilia Apa, Oscar Dieste, EdisonG. Espinosa G., and EfraínR. Fonseca C. Effectiveness for detecting faults within and outside the scope of testing techniques: an independent replication. *Empirical Software Engineering*, pages 1–40, 2013. 8

[Aka74] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, December 1974. 4.4.1

[BDM+94] A Brooks, J Daly, J Miller, M Roper, and M Wood. Replication's Role in Experimental Computer Science, 1994. 2

[BDM+96] A. Brooks, J. Daly, J. Miller, M. Roper, and M. Wood. Replication of experimental results in software engineering. Technical report, International Software Enginnering Research Network (ISERN) Technical Report, 1996. 2

[Bei90] Boris Beizer. *Software Testing Techniques (2Nd Ed.)*. Van Nostrand Reinhold Co., New York, NY, USA, 1990. 3.4

[Ber04] Antonia Bertolino. Guide to the Knowledge area of Software Testing. In *Guide to the Software Engineering Body of Knowledge (SWEBOK)*. IEEE Computer Society, 3Âª edition, 2004. 4.1

[BGL+96] Victor R Basili, Scott Green, Oliver Laitenberger, Filippo Lanubile, Forrest Shull, Sivert Sï¿½rumgï¿½rd, and Marvin V Zelkowitz. The Empirical Investigation of Perspective-Based Reading, 1996. 4.1

[Bif00] S Biffl. Analysis of the impact of reading technique and inspector capability on individual inspection performance. In *Software Engineering Conference, 2000. APSEC 2000. Proceedings. Seventh Asia-Pacific*, pages 136–145, 2000. 4.1

[Bin99] Robert V. Binder. *Testing Object-Oriented Systems: Models, Patterns, and Tools*. Addison-Wesley Professional, 1999. 3.2, 3.4

[BPV95] V R Basili, A A Porter, and Jr. Votta L.G. Comparing detection methods for software requirements inspections: a replicated experiment. *Software Engineering, IEEE Transactions on*, 21(6):563–575, 1995. 4.1

[BS85] Victor R Basili and Richard W Selby. Comparing the Effectiveness of Software Testing Strategies. Technical Report TR-1501, Departament of Computer Science, University of Maryland, College Park, MD, USA, May 1985. 4

[BS87] V R Basili and R W Selby. Comparing the Effectiveness of Software Testing Strategies. *Software Engineering, IEEE Transactions on*, SE-13(12):1278–1296, 1987. 2.2.3.3, 4

[BSH86] V R Basili, R W Selby, and D H Hutchens. Experimentation in software engineering. *IEEE Trans. Softw. Eng.*, 12(7):733–743, 1986. 2

[BSL99] Victor R. Basili, Forrest Shull, and Filippo Lanubile. Building Knowledge through Families of Experiments. *IEEE Transactions on Software Engineering*, 25(4):456, 1999. 2

[CO76] J. Cornfield and R. T. O'Neill. Minutes of the Food and Drug Administration. In *Biostatics and Epidemiology Advisory Committee Meeting*, 1976. 2.2.3.3

[Dij72] Edsger W. Dijkstra. The humble programmer. *Communications of the ACM*, 15(10):859–866, October 1972. 3.1

[DKSb06] Tore Dybå, Vigdis By Kampenes, and Dag I K Sjø berg. A systematic review of statistical power in software engineering experiments. *Information and Software Technology*, 48(8):745–755, 2006. 2.3.5.3

[DRW02] A Dunsmore, M Roper, and M Wood. Further investigations into the development and evaluation of reading techniques for object-oriented code inspection. In *Software Engineering, 2002. ICSE 2002. Proceedings of the 24rd International Conference on*, pages 47–57, 2002. 4.1

[FD00] Phyllis G Frankl and Yuetang Deng. Comparison of delivered reliability of branch, data flow and operational testing: A case study. *SIGSOFT Softw. Eng. Notes*, 25(5):124–134, 2000. 4.1

[Fle89] Joseph L Fleiss. A critique of recent research on the two-treatment crossover design. *Controlled Clinical Trials*, 10(3):237–243, 1989. 2.2.3.3

[FP98] Norman E. Fenton and Shari Lawrence Pfleeger, editors. *Software Metrics: A Rigorous and Practical Approach, Revised [Paperback]*. Course Technology; 2 edition, 2 edition, 1998. 2.1

[Fre89] P R Freeman. The performance of the two-stage analysis of two-treatment, two-period crossover trials. *Statistics in Medicine*, 8(12):1421–1432, 1989. 2.2.3.3, 4.4

[FWH97] Phyllis G Frankl, Stewart N Weiss, and Cang Hu. All-uses vs mutation testing: An experimental comparison of effectiveness. *Journal of Systems and Software*, 38(3):235–253, 1997. 4.1

[GÍ2] Omar Gómez. *Tipología de replicaciones para la síntesis de experimentos en ingeniería del software*. PhD thesis, Facultad de Informática - Universidad Politécnica de Madrid, 2012. 2

[GJV10] Omar S Gómez, Natalia Juristo, and Sira Vegas. Replications types in experimental disciplines. *Empirical Software Engineering and Measurement*, 38(23):9772–9782, 2010. 2

[Gra92] Robert B Grady. *Practical Software Metrics for Project Management and Process Improvement*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1992. 3.4

[HFGO94] Monica Hutchins, Herb Foster, Tarak Goradia, and Thomas Ostrand. Experiments of the effectiveness of dataflow- and controlflow-based test adequacy criteria. In *Proceedings of the 16th international conference on Software engineering*, ICSE '94, pages 191–200, Los Alamitos, CA, USA, 1994. IEEE Computer Society Press. 4.1

[iee10] IEEE Standard Classification for Software Anomalies. *IEEE Std 1044-2009 (Revision of IEEE Std 1044-1993)*, pages 1–23, 2010. 3.4

[Ito80] P K Ito. 7 Robustness of ANOVA and MANOVA test procedures. In P R Krishnaiah, editor, *Analysis of Variance*, volume 1 of *Handbook of Statistics*, pages 199–236. Elsevier, 1980. 4.4.1

[JK89] Byron Jones and Michael G. Kenward. *Design and Analysis of Crossover Trials*. Chapman and Hall, 1989. 2.2.3.3

[JM01] Natalia Juristo and Ana M. Moreno. *Basics of Software Engineering Experimentation*. Springer, 2001. 2.1, 4.3

[JP05] a. Jedlitschka and D. Pfahl. Reporting guidelines for controlled experiments in software engineering. *2005 International Symposium on Empirical Software Engineering, 2005.*, pages 92–101, 2005. 4.5

[JV09] Natalia Juristo and Sira Vegas. Using differences among replications of software engineering experiments to gain knowledge. *Empirical Software Engineering and Measurement*, pages 356–366, 2009. 2

[JV11] Natalia Juristo and Sira Vegas. The role of non-exact replications in software engineering experiments. *Empirical Software Engineering*, 16(3):295–324, 2011. 2

[JVS+12] N Juristo, S Vegas, M Solari, S Abrahao, and I Ramos. Comparing the Effectiveness of Equivalence Partitioning, Branch Testing and Code Reading by Stepwise Abstraction Applied by Subjects. In *Software Testing, Verification and Validation (ICST), 2012 IEEE Fifth International Conference on*, pages 330–339, 2012. 4

[KFL03] Barbara Kitchenham, John Fry, and Stephen Linkman. The Case Against Cross-Over Designs in Software Engineering. In *Proceedings of the Eleventh Annual International Workshop on Software Technology and Engineering Practice*, STEP '03, pages 65–67, Washington, DC, USA, 2003. IEEE Computer Society. 2.2.3.3

[KFN99] Cem Kaner, Jack L Falk, and Hung Quoc Nguyen. *Testing Computer Software, Second Edition.* John Wiley & Sons, Inc., New York, NY, USA, 2nd edition, 1999. 3.4

[KL95] Erik Kamsties and Christopher M. Lott. An Empirical Evaluation of Three Defect-Detection Techniques, 1995. 4

[Kue99] Robert O. Kuehl. *Design of Experiments: Statistical Principles of Research Design and Analysis.* Duxbury Press, 1999. 2.2.3.3, 4.3, 4.4, 4.4.1

[Lin79] Richard C. Linger. *Structured Programming: Theory and Practice (The Systems programming series).* Addison-Wesley, 1979. 4.2.1

[LMK04] Hyo-Jeong Lee, Yu-Seong Ma, and Yong-Rae Kwon. Empirical evaluation of orthogonality of class mutation operators. In *Software Engineering Conference, 2004. 11th Asia-Pacific*, pages 512–518, 2004. 4.1

[LPN00] R C Littell, J Pendergast, and R Natarajan. Modelling covariance structure in the analysis of repeated measures data. *Statistics in medicine*, 19(13):1793–819, July 2000. 4.4.1

[Lyu96] Michael R Lyu, editor. *Handbook of Software Reliability Engineering.* McGraw-Hill, Inc., Hightstown, NJ, USA, 1996. 3.4

[Mar04] E J Martínez. Notas del curso de posgrado Maestría en Estadística Matemática. Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires, 2004. 2.3.5.3

[MCS+06] JoséC. Maldonado, Jeffrey Carver, Forrest Shull, Sandra Fabbri, Emerson Dória, Luciana Martimiano, Manoel Mendonça, and Victor Basili. Perspective-Based Reading: A Replicated Experiment Focused on Individual Reviewer Effectiveness. *Empirical Software Engineering*, 11(1):119–142, 2006. 4.1

[Mil05] James Miller. Replicating software engineering experiments: a poisoned chalice or the Holy Grail. *Inf. Softw. Technol.*, 47(4):233–244, 2005. 2

[MSN08] Charles E. McCulloch, Shayle R. Searle, and John M. Neuhaus. *Generalized, Linear, and Mixed Models (Wiley Series in Probability and Statistics).* Wiley-Interscience, 2008. 2.2.3.3, 2.3.5.3

[Mye78] Glenford J Myers. A controlled experiment in program testing and code walkthroughs/inspections. *Commun. ACM*, 21(9):760–768, 1978. 4.1

[NCC03] Peter Nelson, Marie Coffin, and Karen Copeland. *Introductory statistics for engineering experimentation.* Elsevier Science, California, 2003. 2.3.5.3

[NS11] Marija Norusis and Inc. SPSS. *IBM SPSS Statistics 19 Advanced Statistical Procedures Companion*. Addison Wesley, 1 edition, 2011. 2.3.5.3

[OL94] A J Offutt and S D Lee. An empirical evaluation of weak mutation. *Software Engineering, IEEE Transactions on*, 20(5):337–344, 1994. 4.1

[OLR+96] A Jefferson Offutt, Ammei Lee, Gregg Rothermel, Roland H Untch, and Christian Zapf. An experimental determination of sufficient mutant operators. *ACM Trans. Softw. Eng. Methodol.*, 5(2):99–118, 1996. 4.1

[PSG12] Victor Pankratius, F Schmidt, and G Garreton. Combining functional and imperative programming for multicore software: An empirical study evaluating Scala and Java. In *Software Engineering (ICSE), 2012 34th International Conference on*, pages 123–133, 2012. 2.2.3.3

[PUT+01] L Prechelt, B Unger, W F Tichy, P Brossler, and L G Votta. A controlled experiment in maintenance: comparing design patterns to simpler solutions. *Software Engineering, IEEE Transactions on*, 27(12):1134–1144, 2001. 2.2.3.3

[RWM97] Marc Roper, Murray Wood, and James Miller. An empirical evaluation of defect detection techniques. *Information and Software Technology*, 39(11):763–775, 1997. 2.2.3.3

[SCVJ08] Forrest J Shull, Jeffrey C Carver, Sira Vegas, and Natalia Juristo. The role of replications in Empirical Software Engineering. *Empirical Softw. Engg.*, 13(2):211–218, 2008. 2

[Sen02] Stephen S. Senn. *Cross-over Trials in Clinical Research*. John Wiley & Sons, 2 edition, 2002. 4.4

[SLS06] Andreas Spillner, Tilo Linz, and Hans Schaefer. *Software Testing Foundations: A Study Guide for the Certified Tester Exam*. Rocky Nook, 1 edition, May 2006. 3.3, 4.2.1

[SMB+04] Forrest Shull, Manoel G. Mendoncça, Victor Basili, Jeffrey Carver, José C. Maldonado, Sandra Fabbri, Guilherme Horta Travassos, and Maria Cristina Ferreira. Knowledge-Sharing Issues in Experimental Software Engineering. *Empirical Software Engineering*, 9(1/2):111–137, March 2004. 4

[Sol11] Martin Solari. *Propuesta de Paquete de Laboratorio para Experimentos en Ingeniería de Software*. PhD thesis, Facultad de Informática - Universidad Politécnica de Madrid, 2011. 2

[Spi91] M R Spiegel. *Estadística - 2da Edición*. Mc.Graw-Hill, Madrid, 1991. 2.3.5.3

[TB03] Guilherme Horta Travassos and Márcio Barros. Contributions of In Virtuo and In Silico Experimentes for the Future of Empirical Studies in Software Engineering. In *2nd Workshop in Workshop Series on Empirical Software Engineering The Future of Empirical Studies in Software Engineering*, 2003. 2.1

[TdSM+08] Guilherme H. Travassos, Paulo Sérgio Medeiros dos Santos, Paula Gomes Mian, Arilo Cláudio Dias Neto, and Jorge Biolchini. An Environment to Support Large Scale Experimentation in Software Engineering. In *13th IEEE International Conference on Engineering of Complex Computer Systems (iceccs 2008)*, pages 193–202. IEEE, March 2008. 2.3

[VAB+10] Guillermo Vallejo Seco, Jaime Arnau Gras, Roser Bono Cabré, Paula Fernández Garcia, and Ellián Tuero Herrero. Selección de modelos anidados para datos longitudinales usando criterios de información y la estrategia de ajuste condicional. *Psicothema*, 22(2):323–333, 2010. 4.4.1

[VAD+09] Diego Vallespir, Cecilia Apa, Stephanie De León, Rosana Robaina, and Juliana Herbert. Effectiveness of Five Verification Techniques. *Text*, 2009. 4.1

[VADR09] Diego Vallespir, Cecilia Apa, Stephanie De León, and Rosana Robaina. Efectiveness and cost of verification techniques: Preliminary conclusions on five techniques. In *n Proceedings of the XXVIII International Conference of the Chilean Computer Society*. IEEE Computer Society, 2009. 4.1

[VMBH09] Diego Vallespir, Silvana Moreno, Carmen Bogado, and Juliana Herbert. Towards a Framework to Compare Formal Experiments that Evaluate Testing Techniques. In *Proceedings of the X Mexican International Conference in Computer Science*, 2009. 4.1

[WBM91] Benjamin J Winer, Donald R Brown, and Kenneth M Michels. *Statistical Principles In Experimental Design*. McGraw-Hill Humanities/Social Sciences/Languages, 1991. 2.2.3.2

[WM95] W.Eric Wong and AdityaP. Mathur. Fault detection effectiveness of mutation and data flow testing. *Software Quality Journal*, 4(1):69–83, 1995. 4.1

[WRBM97] Murray Wood, Marc Roper, Andrew Brooks, and James Miller. Comparing and combining software defect detection techniques: A replicated empirical study. In Mehdi Jazayeri and Helmut Schauer, editors, *Software Engineering - ESEC/FSE'97*, volume 1301 of *Lecture Notes in Computer Science*, pages 262–277. Springer Berlin / Heidelberg, 1997. 4

[WRH99] Claes Wohlin, Per Runeson, and Martin Höst. *Experimentation in Software Engineering: An Introduction (International Series in Software Engineering)*. Kluwer Academic Publishers, Norwell, MA, USA, 1999. 2.1, 2.3.5.3, 4.2.5

[WRH+12] Claes Wohlin, Per Runeson, Martin Hst, Magnus C Ohlsson, Bjrn Regnell, and Anders Wessln. *Experimentation in Software Engineering*. Springer Publishing Company, Incorporated, 2012. 2

[ZW97] Marvin V Zelkowitz and Dolores Wallace. Experimental validation in software engineering. *Information and Software Technology*, 39(11):735–743, 1997. 2