UNIVERSIDAD DE LA REPÚBLICA
FACULTAD DE INGENIERÍA

Modelado de Trayectorias Académicas de Estudiantes Universitarios mediante Técnicas de Analítica de Aprendizaje

Por: MARTINEZ BEN, PABLO ANDRES - MONTAÑES SOLERI, OSCAR SEBASTIAN - SERRALTA GASCUE, JUAN MANUEL

Junio 2021
Montevideo, Uruguay

Tutora: Libertad Tansini

AGRADECIMIENTOS

Este proyecto representa el fin de una etapa y el comienzo de otra para nuestras vidas. Es un punto y aparte en la construcción de nuestra carrera profesional.

Este camino es imposible de transitarlo solos y menos aún este proyecto de grado en tiempos de pandemia. Es por eso, que corresponde agradecer: a los integrantes de la Unidad de Enseñanza de la Facultad de Ingeniería, Daniel Alessandrini y Ximena Otegui por el tiempo dedicado como contrapartes del proyecto, siempre dando devoluciones precisas, que aumentaron la calidad de este proyecto. A Daniel Calegari, por sus aportes en la mejora de conceptos de minería de procesos y su visión como Director de carrera. A Adriana Marotta por su contribución en el diseño del Data Warehouse. A nuestra tutora Libertad, por en tiempos de pandemia dedicar tiempo, esfuerzo y paciencia en la elaboración y medición de avance del proyecto.

Imposible olvidarnos de nuestros familiares, parejas, amigos y compañeros que nos acompañaron en este camino de sacrificio, donde mañanas, tardes y noches nos alejamos de ellos. Gracias a todos ellos, finalizamos esta etapa de Ingenieros.

Resumen

La motivación para la realización de este proyecto, surge a raíz de la necesidad de la UEFI (Unidad de Enseñanza de Facultad de Ingeniería), de buscar posibles causas o explicaciones sobre la desvinculación de los estudiantes con sus carreras, en base a los datos disponibles en el sistema de bedelías de la Udelar.

Dentro de este proyecto se desprenden los siguientes resultados: 1.- La automatización de parte del proceso de realización de informes por parte de la UEFI mediante la implementación de una primera versión de un Data Warehouse. 2.- Análisis descriptivo de las variables que pueden incidir en la desvinculación de los estudiantes utilizando nuevas fuentes de datos, que hasta el comienzo de este proyecto no estaban siendo utilizadas (encuesta continua de hogares, escolaridades de los alumnos en la ANEP y geo-referenciación de las direcciones de los estudiantes). 3.- Modelado de las trayectorias y comportamiento de los estudiantes a lo largo de su carrera, utilizando técnicas de machine learning y process minnig, para analizar posibles motivos sociales o curriculares, que podrían brindar alguna explicación de la desvinculación con la carrera.

Keywords – Analíticas del Aprendizaje, Machine Learning, Process Mining, Data Warehouse

Índice general

Capítulo 1

Introducción

En 1903, el dramaturgo uruguayo Florencio Sánchez, escribió una novela teatral llamada “M’hijo el dotor”, la misma contrapone las costumbres e ideas de Julio, un joven recibido de medicina y su padre, en donde en cierta manera y en algunos momentos de la obra, se arrepiente de los comienzos universitarios de su hijo relacionados a la vida en la ciudad. Hoy, estas situaciones son difíciles que se den, donde muchos padres se muestran orgullosos del comienzo de sus hijos en una carrera universitaria. Sin embargo, esta situación no es del todo optimista como se espera, donde muchos estudiantes no logran completar sus estudios terciarios.

Según el Banco Mundial[5], en Latinoamérica, la inscripción universitaria ha sufrido una fuerte expansión, donde los estudiantes considerados más pobres representaron el 45 % del aumento de la matrícula en los últimos años. No obstante, la mitad de las personas entre 25 y 29 años que estaban matriculadas no completaron sus estudios, ya sea por abandono o porque aún continúan estudiando. Uruguay no es ajeno a esta situación en donde la tasa de egreso de carreras universitarias apenas supera los cuarenta puntos porcentuales y si nos focalizamos en ingeniería los guarismos no son mejores, donde el indicador de “Eficiencia de titulación bruta (ETB)”, que se calcula como el total de egresados sobre el total de ingresos, apenas supera el 20 por ciento. [26]

Existen distintas hipótesis entre estudiantes y docentes, que intentan explicar esta condición, particularmente algunos lo atribuyen a una falta de presupuesto que conlleva a grupos iniciales numerosos y otros a una mala preparación en educación media. [1]

La UEFI (Unidad de Enseñanza de Facultad de Ingeniería) tiene dentro de sus cometidos el estudio y la mejora continua de los procesos de enseñanza y de

aprendizaje en la Facultad de Ingeniería (FING). En este marco, esta unidad, presenta periódicamente informes sobre la situación de los estudiantes, los cuales, se hacen de manera no automatizada, por los técnicos que en ella trabajan. Con el fin de optimizar estas tareas, surgió la propuesta de que la UEFI participara como contraparte en el desarrollo de un proyecto de grado para este fin.

1.1. Objetivos

El objetivo general del presente proyecto es desarrollar un modelado de trayectorias académicas de estudiantes universitarios de la FING mediante técnicas de analítica de aprendizaje, disciplina que se describe en el siguiente capítulo.

De este objetivo general se desprenden algunos objetivos específicos, que son:

- Analizar la información con que cuenta la UEFI y otros actores.
- Desarrollar herramientas que permitan a la UEFI obtener indicadores automatizados, para la mejora en la toma de decisiones de los distintos actores de gestión de la FING. Para esto, se buscará desarrollar los distintos reportes mediante la confección de tableros de gestión (dashboards) utilizando técnicas de Data Warehouse.
- Utilizar técnicas de Ciencia de datos [37] para la generación de modelos de trayectorias académicas de los estudiantes, mediante las siguientes técnicas y propósitos:
    - Minería de datos: Generación de modelos para ver cuales son los posibles motivos sociales o de comportamiento de los estudiantes, que podrían brindar alguna explicación de la desvinculación con la carrera.
    - Minería de procesos: Utilización de la herramienta de minería de procesos ProM Tools [20] para analizar los datos presentes en Bedelías con el fin de generar modelos de trayectorias y análisis generales sobre estos. Se busca utilizar esta técnica para identificar aquellas UC que dificulten el avance o egreso de los estudiantes en sus carreras, así como aquellas UC con las cuales los estudiantes egresan o se desvinculan.

1.2. Organización del documento

A continuación se describe la estructura del presente documento:

En el capítulo 2 se brindan conceptos fundamentales y el estado actual de la Analítica del Aprendizaje basada en datos, necesarios para el entendimiento del proyecto. Se incluye una introducción donde se exponen métodos de aprendizaje automático y sus principales características y fundamentos.

Dentro del capítulo 3 se describen los datos disponibles los cuales serán utilizados para los análisis en los subsecuentes capítulos.

En el capítulo 4 detalla el proceso seguido para el diseño e implementación de una primera versión de un Data Warehouse que de soporte a las necesidades de la UEFI.

Siguiendo con el capítulo 5, se desarrollan distintos análisis de los datos mediante métodos clásicos de la información, con los datos disponibles de la UEFI, la ANEP y datos abiertos del estado. También se desarrollan análisis mediante técnicas de minería de procesos.

Finalmente en el último capítulo, se expresan conclusiones generales y recomendaciones que se derivan de los resultados de los capítulos anteriores.

Capítulo 2

Estado del arte

En el presente capítulo, se especifica el significado en el marco de este proyecto de algunos conceptos previos sobre la analítica del aprendizaje, algunas experiencias realizadas y desafíos en el área.

2.1. Analítica del Aprendizaje (AA)

Cada vez los sistemas educativos tienen mayor demanda, donde centenas de miles de alumnos anualmente ingresan con tres o cuatro años a niveles de educación inicial, dando comienzo así a su "vida educativa". Gran cantidad de ellos finalizarán la primaria, una menor cuota la secundaria y solo algunos accederán y finalizarán sus estudios terciarios o universitarios. A esto se suma y de forma casi natural, docentes y alumnos, van registrando información (notas, asistencias, evaluaciones, etc) en sistemas administrativos o en plataformas de corte pedagógicos, registrando así "trazas" educativas del estudiante.[34]

Por otro lado, desde hace algunos años, han existido diferentes iniciativas que aplican tecnologías recientes (Data Mining, IA, Big Data, etc) a diferentes problemas sociales y en donde la educación no es ajena a estas [38].

Siemens [48], fue unos de los primeros autores en unir estas realidades y formar el concepto de *"Analítica del Aprendizaje"*, formalmente él lo define como *"el uso de datos inteligentes, datos producidos por el alumno y modelos de análisis para descubrir información y conexiones sociales, y predecir y asesorar sobre el aprendizaje"*.

2.1.0.1. Minería de datos educativos

Una leve variante a la Analítica del Aprendizaje es la Minería de datos educativos. La minería de datos educativos o EDM por sus siglas en inglés, se preocupa por desarrollar métodos para explorar los tipos únicos de datos que provienen de un contexto educativo, utilizando estos métodos, para comprender mejor a los estudiantes y los entornos en los que aprenden [47].

Los conceptos, los datos, los procesos y los objetivos en Learning Analitycs y Educational Data Mining son bastante similares. Ambos apuntan en trabajar con datos procedentes de entornos educativos y sacar provecho de estos datos en información relevante con el objetivo de mejorar el proceso de aprendizaje o la estructura educativa.

Chatti et al. [31] establecen que sin embargo, las técnicas utilizadas para LA (Learning Analytics) pueden ser bastante diferentes de las utilizadas en EDM. EDM básicamente se enfoca en la aplicación de técnicas típicas de minería de datos (es decir, agrupación, clasificación, y minería de reglas de asociación) para apoyar a los maestros y estudiantes en el análisis del proceso de aprendizaje. Adicionalmente a las técnicas de minería de datos, la Analítica del Aprendizaje también incluye otros métodos, como estadísticas y herramientas de visualización o técnicas de análisis de redes sociales (SNA), y las pone en práctica la mejora de la enseñanza y el aprendizaje.

2.1.1. Un modelo de referencia

En el mismo estudio, Chatti et al. [31] exponen un modelo de referencia para tomar acciones en el proceso de la Analítica del Aprendizaje . En él establecen la AA como 4 incógnitas: ¿Qué se analiza? ¿Quién es el público a analizar? ¿Para qué analizamos los datos recolectados? ¿Cómo se explota la información a partir de los datos?

En esta sección se describe cómo los autores desarrollan cada uno de estos puntos.

2.1.1.1. Datos y entornos ¿Qué se analiza?

Una pregunta interesante que se menciona, es: en AA, ¿de dónde provienen los datos educativos? Como respuesta, los autores describen dos grandes categorías:

- Sistemas informáticos educativos centralizados: como son el caso del sistema de bedelías o el EVA en FING o los datos que se generan en la ANEP y CEIBAL.

- Entornos de aprendizaje distribuido: son una variante que se está volviendo cada vez más importante y popular y un ejemplo son los entornos personales de aprendizaje (PLE por sus siglas en inglés). Que son plataformas enfocadas en la libertad del alumno y su forma de aprender libremente con las TICs, en donde se recolecta información de distinto tipo y formato de distintas fuentes.

La gran dificultad en ambos, es la integración y agregación de datos sin procesar de múltiples fuentes heterogéneas, a menudo disponibles en diferentes formatos, para crear un conjunto de datos educativos útiles que refleje las actividades distribuidas del alumno.

2.1.1.2. Stakeholders ¿Quién es el público a analizar?

Dentro del público objetivo a analizar, se menciona que la aplicación de AA puede orientarse hacia diferentes partes interesadas, incluidos estudiantes, docentes, tutores/mentores, instituciones educativas (administradores y tomadores de decisiones de la facultad), investigadores y diseñadores de sistemas con diferentes perspectivas, objetivos y expectativas. A su vez expresan que los docentes pueden estar interesados en cómo la analítica puede aumentar la efectividad de sus prácticas de enseñanza o ayudarlos a adaptar sus ofertas de enseñanza a las necesidades de los estudiantes. Las instituciones educativas pueden usar herramientas analíticas para apoyar la toma de decisiones, identificar a los estudiantes potenciales "en riesgo" y mejorar el desempeño de la política o gestión educativa.

2.1.1.3. Objetivos ¿Para qué analizamos los datos recolectados?

Dentro de los objetivos que mencionan Chatti et al. [31] en AA se incluyen monitoreo, análisis, predicción, intervención, tutoría, evaluación, retroalimentación, adaptación, personalización, recomendación y reflexión. A continuación se describe un breve resumen de cada uno.

Monitoreo y análisis: en el monitoreo, los objetivos son rastrear las actividades de los estudiantes y generar informes para apoyar la toma de decisiones por distintos actores. Examinar cómo los estudiantes usan un sistema de aprendizaje y analizar los logros de los estudiantes. Además, puede ayudar a los docentes a detectar patrones y tomar decisiones sobre el diseño futuro.

Predicción e intervención: en la predicción, el objetivo es desarrollar un modelo que intente predecir el comportamiento del alumno y el rendimiento futuro, en

función de sus condiciones actuales. Este modelo predictivo se puede utilizar para proporcionar una intervención proactiva para estudiantes que pueden necesitar asistencia adicional. El análisis efectivo y la predicción del rendimiento del alumno, puede apoyar al profesor o la institución educativa en la intervención sugiriendo acciones que deben tomarse para ayudar a los alumnos a mejorar su rendimiento.

Tutoría y mentoría: la tutoría se ocupa principalmente de ayudar a los estudiantes con sus aprendizajes, a menudo en tareas muy específicas del dominio y limitadas al contexto de un curso. Un tutor, por ejemplo, apoya a los alumnos en su orientación e introducción en nuevos módulos de aprendizaje como así también, instrucciones de áreas temáticas específicas dentro de un curso. En contraste, la mentoría va más allá de la tutoría y se centra en apoyar al alumno durante todo el proceso, idealmente durante toda la vida, pero en realidad limitada al tiempo en que tanto el mentor como el alumno forman parte de la misma organización. En los procesos de mentoría, el control reside más bien en los alumnos y el foco está en el proceso de aprendizaje.

Evaluación y retroalimentación: el objetivo es apoyar la (auto) evaluación de mejoras en eficiencia y efectividad del proceso de aprendizaje. La retroalimentación proporciona información interesante generando en base a datos sobre los intereses del usuario y el contexto de aprendizaje.

Adaptación: la adaptación es activada por el docente/sistema de tutoría o el sistema educativo. El objetivo de AA aquí es decirles a los alumnos qué hacer a continuación organizando de manera adaptativa recursos de aprendizaje y actividades educativas de acuerdo con las necesidades del alumno individual.

Personalización y recomendación: en la personalización, AA está altamente centrada en el alumno, enfocándose en cómo ayudar a los estudiantes a decidir sobre su propio aprendizaje y moldear continuamente las plataformas con las cual interactúa para lograr sus objetivos de aprendizaje. También se vuelve crucial examinar algunos mecanismos para ayudar a los estudiantes a hacer frente al problema de sobrecarga de información. Aquí es donde los sistemas de recomendación pueden desempeñar un papel crucial para fomentar el aprendizaje autodirigido. El objetivo de AA en este caso es ayudar a los alumnos a decidir qué hacer a continuación, recomendando explícitamente a los alumnos nodos de conocimiento (es decir, recursos de aprendizaje) y nodos de conocimiento tácito (es decir, personas), en función de sus preferencias y actividades de otros alumnos con preferencias similares.

Reflexión: la analítica puede ser una herramienta valiosa para promover la reflexión.

Los estudiantes y profesores pueden beneficiarse de los datos comparados dentro del mismo curso, entre clases o incluso entre instituciones para sacar conclusiones y (auto) reflexionar sobre la efectividad de su aprendizaje o práctica docente. El objetivo es que los datos recopilados de diferentes entornos alimenten en un modelo personal de aprendizaje permanente, que sería una especie de repositorio donde el alumno puede archivar todas las actividades de aprendizaje a lo largo de su vida.

2.1.1.4. Métodos ¿Cómo se explota información a partir de los datos?

AA aplica diferentes técnicas para detectar patrones interesantes ocultos en conjuntos de datos educativos. En esta sección, mencionamos cuatro técnicas que se describen y han recibido especial atención en AA en la literatura en los últimos tiempos.[29]

Estadísticas: la mayoría de los sistemas de gestión de aprendizaje existentes implementan herramientas de informes para proporcionar estadísticas básicas de la interacción de los estudiantes con el sistema. A menudo generan operaciones estadísticas simples como promedio, media y desviación estándar.

Visualización de información: las estadísticas en forma de informes y tablas de datos no son siempre fáciles de interpretar para los usuarios del sistema educativo. Representando los resultados obtenidos con los métodos de AA en una forma visual fácil de usar podrían facilitar la interpretación y el análisis de los datos educativos. Reconociendo el poder de las representaciones visuales, los informes tradicionales basados en tablas de datos son cada vez más reemplazados por paneles que muestran gráficamente diferentes indicadores de rendimiento.

Minería de datos: La minería de datos, es definida como "el proceso de descubrir patrones o conocimientos útiles de fuentes de datos, por ejemplo, bases de datos, textos, imágenes, la Web". En términos generales, los métodos de minería de datos, que son bastante prominentes en la literatura EDM y generalmente se las clasifica en las siguientes categorías generales: aprendizaje supervisado (o clasificación y predicción), aprendizaje no supervisado (o agrupamiento) y la minería de reglas de asociación.

- La clasificación es el proceso de encontrar una función (o modelo) que describe y distingue clases de datos o conceptos, con el fin de poder utilizar la función para predecir la clase de objetos cuya etiqueta de clase es desconocida. Los métodos de clasificación populares incluyen árboles de decisión, redes neuronales, ingenua clasificación bayesiana, máquinas de vectores de soporte

(SVM) [53] y clasificación de vecinos más cercanos (KNN) [53] para datos discretos, o la regresión que es metodología estadística que a menudo se utiliza para predicción numérica. La clasificación también se denomina aprendizaje supervisado porque los objetos de datos utilizados para el aprendizaje (llamados datos de entrenamiento) están etiquetados con clases predefinidas.

- La agrupación (aprendizaje no supervisado) contrasta con la clasificación (aprendizaje supervisado) en que la etiqueta de clase de cada objeto de entrenamiento no se conoce de antemano. La agrupación es el proceso de organizar los objetos de datos en grupos, de modo que los objetos dentro de un grupo sean "similares" entre sí y "diferentes" a los objetos en otros grupos. La similitud se define comúnmente en términos de qué tan cerca están los objetos en el espacio, en función de una función de distancia. En general, la mayoría de los métodos de agrupamiento se pueden clasificar en las siguientes categorías: métodos de partición y métodos jerárquicos.
- Los métodos basados en la densidad ven los clústeres como regiones densas de objetos en el espacio de datos que son separados por regiones de baja densidad (que representan ruido). DBSCAN y su extensión OPTICS [39] son métodos típicos basados en la densidad.
- La minería de reglas de asociación conduce al descubrimiento de asociaciones y correlaciones interesantes dentro de los datos. Métodos populares para las reglas de asociación minera son el algoritmo Apriori [51] y los árboles de patrones frecuentes (FP-tree) [43] .

Análisis de redes sociales (SNA): a medida que las redes sociales se vuelven importantes para apoyar aprendizaje en red, las herramientas que permiten administrar, visualizar y analizar estas redes han ganando popularidad. Al representar visualmente una red social, se podrían establecer conexiones interesantes para así poder ver y explorar en una forma fácil de usar. SNA es el estudio cuantitativo de las relaciones entre individuos u organizaciones. En SNA, una red social está modelada por un grafo G = (V, E), donde V es el conjunto de nodos (también conocidos como vértices) que representan actores, y E un conjunto de bordes (también conocidos como arcos, enlaces o lazos), que representan un cierto tipo de enlace entre actores.

2.1.2. Desafíos en la Analítica del aprendizaje

Como vimos hasta aquí, todavía la AA es una disciplina emergente, con muchos desafíos por enfrentar, Ferguson [36] establece algunos de ellos que se describen a continuación en esta sección.

Construir fuertes conexiones con las ciencias del aprendizaje: Comprender y optimizar el aprendizaje requiere una buena comprensión de cómo se lleva a cabo el aprendizaje, cómo se puede apoyar y la importancia de factores como identidad, reputación y afecto. A medida que AA emerge de amplios campos de análisis y minería de datos, los investigadores necesitarán construir fuertes conexiones con las ciencias del aprendizaje. Esto tiene el potencial de ser un proceso bidireccional, en donde la AA es una ayuda para formar la base para un buen diseño de aprendizaje, una pedagogía efectiva y un aumento de estudiantes y su conciencia de sí mismos.

Desarrollar métodos de trabajo para optimizar entornos de aprendizaje: Comprender y optimizar los entornos en los que se produce el aprendizaje introduce un segundo desafío. Esto requerirá un cambio hacia conjuntos de datos más desafiantes y combinaciones de conjuntos de datos, incluidos datos móviles, datos biométricos y datos de estado de ánimo. En este orden, para resolver los problemas que enfrentan los alumnos en diferentes entornos, los investigadores necesitarán determinar cuáles son esos problemas y cómo se ve el éxito desde la perspectiva de estudiantes.

Centrarse en las perspectivas de los alumnos: Un enfoque en las perspectivas de los alumnos será esencial para el desarrollo de la analítica. Tal perspectiva tiene el potencial para extender los criterios para el éxito del aprendizaje más allá de las calificaciones y la asistencia, para incluir motivación, confianza, disfrute, satisfacción y cumplimiento de objetivos profesionales. También podría reformar los métodos de calificaciones, alejándose de la evaluación sumativa que mira hacia atrás a lo que los alumnos han logrado, hacia una evaluación formativa que los ayude a desarrollarse. Para lograr esto, se requerirán métodos para informar y visualizar análisis personalizados, que los alumnos pueden entender fácilmente y que estén claramente vinculados con formas de mejorar y optimizar su aprendizaje. En muchos casos, la analítica deberá ser transparente, permitiendo a los alumnos responder con comentarios que puedan ser utilizados para refinar el análisis y permitirles ver cómo se usan sus datos.

Desarrollar y aplicar un conjunto claro de pautas éticas: Enfrentar estos desafíos

requerirá decisiones con respecto a la propiedad y la administración de datos. Los puntos de referencia dentro del campo, no aclaran qué derechos tienen los alumnos en relación con sus datos, o la medida en que tienen la responsabilidad de actuar en base a las recomendaciones proporcionadas por el AA.

Existen algunos concensos entre los investigadores sobre el como obtener el consentimiento informado y continuo para el uso de datos para la AA, donde cada uno puede generar consideraciones adicionales. Algunos autores [44], sugieren tras revisar numerosas propuestas relacionadas con la analítica en general, marcos gubernamentales y directivas regulatorias, una serie de principios donde se busca, proporcionar una línea de base para las instituciones educativas que actualmente utilizan analítica, puedan reflexionar sobre su nivel de cumplimiento y prever posibles mejoras con respecto a la intimidad.

Estos principios son:

- Transparencia: La transparencia puede ser aplicada a prácticamente todas las etapas de la analítica de aprendizaje. En términos generales, las partes interesadas debe tener acceso a la descripción de cómo se lleva a cabo el proceso de análisis. Debe ser informado el tipo de información que se recopila, incluida su forma, como se almacena y se procesa.
- Control de los estudiantes sobre los datos: El control de los estudiantes sobre los datos ahora está presente en la mayoría de las regulaciones de privacidad, pero puede variar significativamente en la forma en que se implementa. Se relaciona con el principio de transparencia en el sentido de que, para para que los estudiantes tengan control sobre los datos que se recopilan, necesitan saber qué se recopilan, cuándo, cómo y cómo se manipulan. Por tanto, todos estos aspectos deben ofrecerse de forma transparente a los estudiantes.
- Derecho de acceso:Los datos recopilados deben estar sujetos a un conjunto de derechos de acceso claramente definidos. Cuando estos derechos no se observan correctamente, las consecuencias para la confianza del usuario podrían ser drásticas. Las instituciones educativas deben prestar especial atención a este principio, la exposición de datos sensibles al público puede tener un impacto profundo en todas las partes interesadas. Esta política puede ser compleja y debe tenerse en cuenta en las primeras etapas de diseño del marco de análisis. La política debe identificar claramente el tipo de operaciones permitidas en los datos y también qué usuarios tienen acceso a qué áreas de la aplicación.

- Responsabilidad y evaluación:La evaluación es un principio que afecta a todos los aspectos en un escenario de análisis de aprendizaje. Cada aspecto debe tener una persona, organismo, departamento o institución identificada como responsable de la funcionamiento de sus componentes relacionados. La evaluación es un principio que se traduce en solidez del proceso general.Por evaluación, nos referimos también a la responsabilidad de la institución de evaluar, revisar y perfeccionar constantemente la recopilación de datos, la seguridad, la transparencia y la rendición de cuentas. Esto es especialmente relevante cuando se considera que las leyes y regulaciones que se aplican al aprendizaje los análisis están cambiando a un ritmo significativamente mayor que en otras áreas.

Existen propuestas de investigadores locales [35], que dan un serie de recomendaciones concretas de como llevar a cabo estos principios, dentro de las que consideramos más importantes se encuentran:

- Consentimiento informado y Reserva: los autores recomiendan la utilización de un mecanismo de recolección automatizada de consentimiento, como por ejemplo una encuesta obligatoria al inicio del curso. También resaltan, que dado el caso que como resultado del procesamiento de los datos, se deriva alguna valoración personal, que afecte de manera significativa al estudiante, este tiene derecho a ser informado sobre el criterio de valoración y el programa utilizado para ello. Si la información recolectada será compartida con otro equipo de investigadores o con otra institución y esto no fue informado a los estudiantes o consultar a la Autoridad de Protección de Datos Nacional.

- Finalidad y Conservación de los datos: Como principio general los autores recomiendan que, no se deben conservar los datos sin anonimizar si no es necesario. También recomiendan que una vez que se extingue la razón original por la que fueron recolectados los datos, estos deben eliminarse. Por lo que las instituciones educativas que optan por aplicar AA de forma sistemática deberán explicitar su política de conservación de datos, decidiendo, por ejemplo, si existen razones que justifiquen mantener en sus bases la información de los egresados.

- Datos sensibles: Los autores aconsejan tomar en cuenta que legislaciones prohíbe el procesamiento de datos personales que revelen origen racial y étnico, preferencias políticas,convicciones religiosas o morales, afiliación sindical e informaciones referentes a la salud o a la vida sexual sin un permiso específico. Por lo que, si se recolecta este tipo de información, se

debe tener cuidado de que permanezca anónima en la base a utilizar.

- Comportamiento de Instituciones educativas: Finalmente los autores generan un marco teórico de como se deben de comportar las instituciones educativas frente a nuevos desafíos, en donde aconsejan: 1) implementar guías de buenas prácticas, 2) generar un comité de ética que interpele y resuelva posibles conflictos, 3) publicar los aspectos relacionados a la AA a desarrollarse fomentando la transparencia y seguridad a todos los involucrados.

Sin dudas la ética y la responsabilidad tienen mucho para desarrollar, no obstante en este capítulos se deja asentado aspectos básicos a tomar en cuenta para el proyecto y su futuro.

Capítulo 3

Datos disponibles

En el marco del proyecto se tuvieron varias reuniones con los integrantes de la UEFI (Unidad de Enseñanza Facultad de Ingeniería), dentro de estas y bajo consentimiento firmado y clausura de confidencialidad, se pudo acceder a una copia de la base de datos del sistema de bedelías, es una base de SQLite, de la cual se encuentran sus tablas y sus respectivas columnas detalladas en este capítulo.

Tabla Activ2: esta tabla contiene todas las actividades del estudiante para una carrera, en formato línea de tiempo ordenados por fecha. Contiene, todas las inscripciones a cursos, exámenes y reválidas, con su correspondiente resultado/nota.

Cuadro 3.0.1: Tabla Activ2.

Columna	**Tipo**	**Descripción**
Cedula	INT	Documento de identidad que identifica al estudiante
Asignatura	VARCHAR	Asignatura correspondiente a la actividad
TipoActividad	VARCHAR	Es un campo que corresponde al tipo de actividad en donde: A=asistencia,E=examen, I=inscripción, C=curso, D=curso caducado, N=curso invalidado por curso posterior
Nota	INT	Nota de la actividad, 0<nota<12 (Nota=20: s/nota)
Fecha	INT	Fecha de la actividad en formato YYYYMMDD

Cuadro 3.0.1: Tabla Activ2.

Columna	**Tipo**	**Descripción**
Curricular	VARCHAR	Formato curricular de la actividad C=curricular, E=extracurricular, I=prueba de ingreso, A=curso de actualización, R=reválida
TipoGenerado	VARCHAR	Tipo de generación del registro: A=automática, C=cambio de plan, N=normal, R=reválida, V=automática a partir de cálculo (solo se utiliza para AnMatII-Planes viejos)
Periodo	INTEGER	Periodo de la actividad: formato YYYYMM
Tipoperiodo	VARCHAR	Tipo de periodo: E=extraordinario,O=ordinario
Dictada	TINYINT	Codigo del Instituto dicta la asignatura
Observacion	VARCHAR	Observaciones correspondientes al registro de la actividad

Tabla Carrera: Es una tabla codiguera, que contiene los datos de las carreras que se imparten en Facultad de Ingeniería.

Cuadro 3.0.2: Tabla Carrera.

Columna	**Tipo**	**Descripción**
Carrera	INT	Identificador numérico de la carrera
NomCarrera	VARCHAR	Nombre de la carrera
Plan	INT	Plan correspondiente, típicamente es el número de año de aprobación
Ciclo	INT	Ciclo de la carrera
NomCiclo	VARCHAR	Nombre del Ciclo
TipoCiclo	VARCHAR	Tipo de ciclo
MinCreditos	VARCHAR	Cantidad mínima de créditos para egresar de la carrera

Tabla Estudiante-Carrera: Al existir la posibilidad de que un estudiante, ingrese o se encuentre en varias carreras en simultáneo, es necesaria esta tabla. En ella es posible encontrar las relaciones estudiante con sus carreras, sus fechas

de ingreso y de egreso.

Cuadro 3.0.3: Tabla Relación Estudiante-Carrera.

Columna	Tipo	Descripción
Cedula	INT	Documento de identidad del estudiante
Carrera	SMALLINT	Identificador de la carrera
Ciclo	SMALLINT	Ciclo de la carrera
Fechaing	INT	Fecha de ingreso a la carrera en formato YYYYMMDD
Fechaegr	INTEGER	Fecha de egreso a la carrera en formato YYYYMMDD, para los casos que el alumno no ha egresado se registra con el string vacío
Generacion	INT	Generación de ingreso a la carrera

Tabla Estudiante: Esta tabla es la correspondiente a la entidad Estudiante, en ella se encuentran datos propios de él/ella: Fecha de nacimiento, sexo, donde realizó la educación media y datos de contacto.

Cuadro 3.0.4: Tabla Estudiante.

Columna	Tipo	Descripción
Cedula	INT	Documento de identidad del estudiante
Nombre	VARCHAR	Nombre del estudiante
LugarVive	INT	Código del lugar de donde vive, no existe otra tabla para referenciar
FechaNac	INT	Fecha de Nacimiento del estudiante
Sexo	VARCHAR	Sexo del estudiante F o M
Anio	INT	Año de ingreso a la facultad, no a la carrera
Inst	INT	Institución proveniente de educación media o universitaria
TipoInstit	INT	Tipo de institución
Direccion	VARCHAR	Dirección del estudiante
Telefono	VARCHAR	Teléfono del estudiante
Mail	VARCHAR	Correo electrónico del estudiante
Celular	VARCHAR	Celular del estudiante

Tabla Asignaturas: Al igual que la tabla carreras, esta tabla es una codiguera

para las asignaturas que se imparten en FING. En ella, es posible identificar la carrera correspondiente, su método de aprobación/exoneración y la cantidad de créditos que otorga en el plan de estudios.

Cuadro 3.0.5: Tabla Asignaturas.

Columna	**Tipo**	**Descripción**
Carrera	SMALLINT	Identificador de la carrera
Ciclo	SMALLINT	Identificador del ciclo de la carrera
Asignatura	VARCHAR	Identificador de la asignatura en la carrera
NombreAsignatura	VARCHAR	Nombre de la asignatura
OrdComAgrup	VARCHAR	Agrupamiento al cual pertenece la asignatura
TipoExo	VARCHAR	Tipo de exoneración de la asignatura, puede ser C= Curso o E = Examen
Creditos	SMALLINT	Créditos que suma la asignatura al salvar la misma
Dictada	VARCHAR	Instituto que dicta la asignatura
Obligatoria	TEXT	Establece si es obligatorio o no la carrera

Con esto, podemos afirmar(a priori) que tenemos información suficiente para visualizar y hacer modelos que representen las trayectorias de los estudiantes. Existen otras fuentes de información que podrían ser de utilidad pero no se tuvo acceso a lo largo del proyecto, por ejemplo: información de la prueba diagnóstica inicial, datos de carácter social (barrio, profesión de los padres, con quien vive, etc), si se inscribió a una beca o si existiera un sistema nacional de educación terciaria si continua sus estudios o no en una institución privada terciaria.

Además de estos datos, hemos tomado datos abiertos, como lo son la encuesta continua de hogares y los mapas de Montevideo. Estos datos, serán utilizados para medir distancias y estratos sociales de los estudiantes de la Facultad.

En coordinación con la UEFI, se ha tomado la decisión de acotar el estudio de este proyecto a los estudiantes de la carrera “Ingeniería en Computación Plan 1997”. Fundamenta esta decisión que es la carrera más numerosa y homogénea, y por otra parte es la carrera de la cual los integrantes del grupo cuentan con mayor conocimiento.

Como datos adicionales se solicitó formalmente a la ANEP los datos correspondientes a las notas de los alumnos que cursaron la carrera “Ingeniería

en Computación", en donde se pudo recabar información de las calificaciones de algunos alumnos, pero sin una completitud adecuada, es decir, solo habían algunas asignaturas registradas o no se encontraban la totalidad de los exámenes rendidos. En consecuencia, no se puede visualizar una completitud de la trayectoria del estudiante, ni una concepción de su rendimiento académico previo.

Capítulo 4

Construcción de un Data Warehouse

A medida que las organizaciones van creciendo en su tamaño, también se incrementa la cantidad de sistemas que utilizan para dar soporte a sus operaciones del día a día, utilizando por ejemplo sistemas de gestión de clientes, control de stock, logística de envíos, entre otros. Cada uno de estos sistemas genera una gran cantidad de información que la almacena en bases de datos de forma tal que estén disponibles para su uso de forma rápida y confiable. Estas bases de datos se diseñan y optimizan para asegurar un alto rendimiento, soportar alta concurrencia de usuarios accediendo y realizando modificaciones y proveyendo mecanismos de recuperación ante fallos. Este tipo de sistemas se los conoce como sistemas de procesamiento transaccional (**OLTP** por su sigla en inglés online transaction processing). Dado que los sistemas OLTP deben admitir cargas de transacciones pesadas, su diseño debe evitar anomalías de actualización y, por lo tanto, las bases de datos orientadas a OLTP están altamente normalizadas.

Por otro lado, es de vital importancia para las organizaciones analizar los datos que poseen con el propósito de optimizar sus procesos y poder tomar decisiones de manera más informada. Pero realizar análisis consultando directamente las bases de datos de los sistemas OLTP es una tarea muy difícil, y a veces imposible. El primer problema que surge es que las bases de datos operacionales están diseñadas y optimizadas para ser usadas por los sistemas, y no para la realización de análisis. Las mismas contienen información muy detallada, no incluyen datos históricos y tienen un desempeño muy pobre al ejecutar consultas complejas que involucran muchas tablas o si se realizan operaciones de agregación sobre grandes volúmenes de datos. Otro problema que surge cuando las organizaciones necesitan analizar el

comportamiento desde un punto de vista integral, donde se deben integrar datos de varias fuentes diferentes. Esta puede ser una tarea difícil de lograr debido a que cada herramienta potencialmente almacena sus datos en lugares y formatos distintos.

A partir de las dificultades anteriores es que nace la necesidad de crear nuevos tipos de sistemas destinados al análisis de los datos por parte de las organizaciones. Estos sistemas se los conoce como sistemas de procesamiento analítico (**OLAP**, por su sigla en inglés online analytical processing). Por lo tanto, la necesidad de un modelo de base de datos diferente para soportar OLAP fue clara y condujo a la noción de Data Warehouse. Los Data Warehouses son bases de datos especialmente diseñadas y optimizadas para dar soporte a la toma de decisiones. Para ello se toman datos de varias bases de datos operativas y otras fuentes de datos y los transforma en nuevas estructuras que se adaptan mejor a la tarea de realizar análisis. Para la implementación de un Data Warehouse es necesario comprender cuales son las fuentes de datos y cuales son los indicadores que se quieren obtener del mismo, y a partir de ello diseñar cómo va a estar estructurados los datos e implementar los procesos de extracción, transformación y carga (ETL) que permitan popular el Data Warehouse a partir de las fuentes de datos.

La FING no es ajena a esta realidad. Por un lado nos encontramos con sistemas que dan apoyo a las distintas áreas de la Facultad, como lo son el sistema de Bedelías, el Entorno Virtual de Aprendizaje (EVA), entre otros, y por el otro está la UEFI que dentro de sus tareas se encuentra la elaboración de informes donde se estudian y analizan diferentes aspectos de la realidad, tomando como base datos provenientes de diversas fuentes. Gran parte del esfuerzo para realizar dichos informes está enfocado en preparar la información para poder ser analizada, debido a que los datos son extraídos de las fuentes y procesados de forma ad-hoc cada vez que se quiere realizar un informe, o repetirlo al año siguiente.

Para mejorar el proceso de análisis dentro de Facultad se propone el diseño y la implementación de una primera versión de un Data Warehouse que aloje los datos relevantes para la UEFI y herramientas que permitan visualizar la información que contiene el mismo. Con la implementación de dichas herramientas se busca no solo facilitar la elaboración de los informes que la UEFI realiza de forma periódica, sino que además a futuro se puedan elaborar nuevos informes con un costo muy bajo y la extensión de los existentes con nuevos indicadores más complejos.

En este capítulo se describe el proceso de construcción del Data Warehouse, comenzando con la sección 4.1 donde se analizan los informes que periódicamente

realiza la UEFI, identificando las preguntas que se quieren responder y los indicadores que se miden en cada una de ellas. Luego en la sección 4.2 se detallan nuevos indicadores que surgieron luego de múltiples reuniones con la UEFI, que son de interés y complementan a los informes anteriormente analizados. En la sección 4.3 con la primera etapa de la construcción del Data Warehouse, que consiste en el diseño de las tablas que van a dar soporte al mismo. Seguidamente en la sección 4.4 se procede a la implementación del proceso de extracción, transformación y carga utilizado para popular el Data Warehouse. Y por último en la sección 4.5 se describen los dashboards que permiten la visualización de los datos dentro del Data Warehouse.

Todas las definiciones volcadas en este capítulo sobre conceptos relacionados a Data Warehouse son extraídas del libro *Data warehouse systems: design and implementation* de los autores Alejandro Vaisman y Esteban Zimanyi Vaisman [49]

4.1. Análisis de informes

A continuación revisaremos cada uno de los informes que realiza la UEFI periódicamente extrayendo las preguntas que se plantean y los indicadores que se miden en las mismas.

4.1.1. Informe de ingresos

En este informe se analiza la cantidad de estudiantes que ingresan cada año a la Facultad de Ingeniería según varios aspectos:

- Sexo.
- Edad de ingreso.
- Institución de procedencia (liceo público, liceo privado, UTU).
- Procedencia geográfica (Montevideo, Interior, Exterior).
- Carrera a la que se inscribieron.

Se consideran ingresantes en un determinado año a todos aquellos estudiantes que registraron inscripción por primera vez a la Facultad de Ingeniería durante ese año.

Todos estos datos se presentan en formato tabla como se muestra en la figura 4.1.1 junto con el porcentaje que representan del total de la generación.

Figura 4.1.1: Tabla extraída del informe de ingresos con la cantidad y el porcentaje de ingresos por el rango de edad y generación

- Edad al ingreso

Año	17 /18 años	19 / 20 años	21 / 23 años	24 /26 años	> 26 años
2015	928 (60%)	273 (18%)	146 (9%)	87 (6%)	117 (7%)
2016	942 (61%)	273 (17%)	141 (9%)	95 (6%)	106 (7%)
2017	904 (63%)	206 (14%)	121 (8%)	85 (6%)	129 (9%)

.

4.1.2. Informe de egresos

En este informe se analizan la cantidad de egresos por año y carrera como se ve en la figura 4.1.2. También se analiza la Eficiencia de Titulación Real (ETR) que se calcula como:

$$ETR = \frac{Egresos\ Totales}{Egresos\ Totales + Estudiantes\ Activos}$$

Figura 4.1.2: Tablas extraídas del informe de egresos que muestra la cantidad de egresados por carrera y generación y la Eficiencia de Titulación Real.

Carrera	97	98	99	00	01	02	03	04	05	06	07	08	09	10	11	12	Total
Ing. Producción																1	1
Ing. Mecánica	33	26	21	20	21	23	39	28	30	29	32	42	23	17	5	1	390
Ing. Naval	1	2		1		1	1	1		1							8
Ing. Civil	72	58	54	43	55	50	54	43	28	33	28	25	15	13	10		581
Ing. Eléctrica	55	46	42	68	64	59	62	60	28	47	28	22	25	10	10	1	627
Agrimensura	5	7	3	9	6	8	2	3		3	1	2		2			51
Ing. Química (plan 2000)		1	2	47	31	35	32	48	55	46	25	31	29	17	7	1	407
Ing. Alimentos (plan 2003)			3	6	15	16	98	29	28	27	80	11	17	12	2		344
Ing. en Computación	78	105	107	85	96	99	72	87	53	64		56	37	37	8	5	989
Total por gen.	244	245	232	279	288	291	360	299	222	250	194	189	146	108	42	9	3398

Gen.	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	Total
ETR	0,89	0,83	0,81	0,79	0,75	0,72	0,70	0,65	0,52	0,52	0,40	0,33	0,25	0,19	0,07	0,01	**0,46**

4.1.3. Informe de puntos críticos

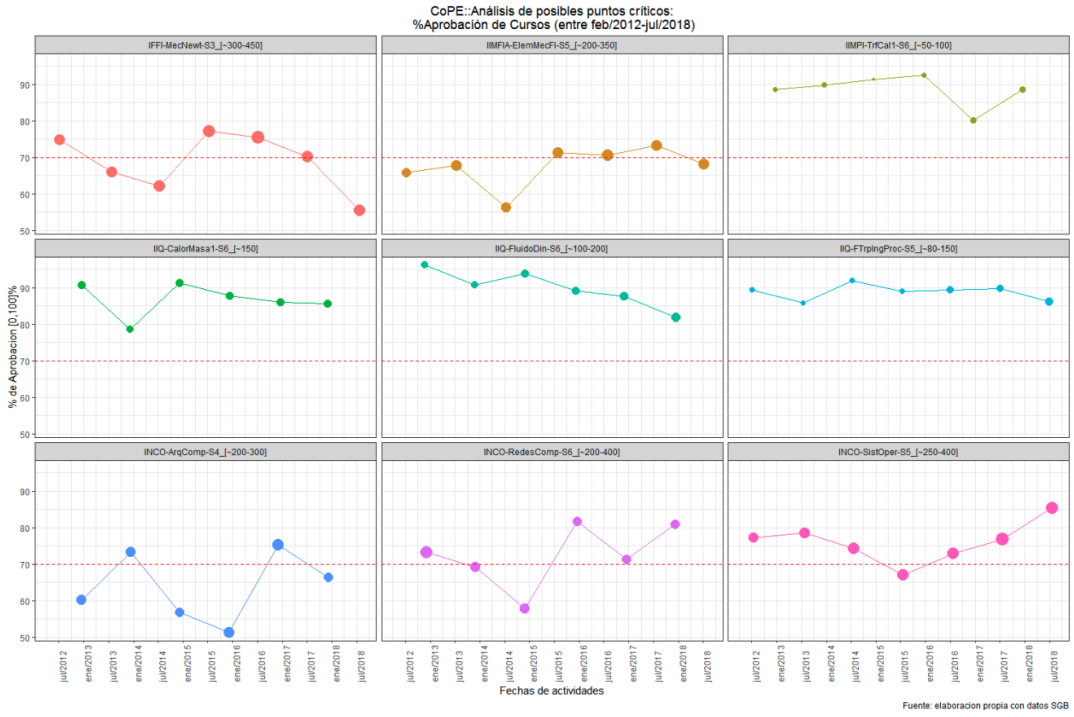
En dicho informe interesa analizar si un conjunto de Unidades Curriculares (UC) son un punto crítico. Se establecieron como criterios para identificar a una UC como punto crítico que en más de la mitad de las ediciones en un periodo de tiempo suceda alguna de las siguientes condiciones:

- La aprobación del curso sea menor o igual al 70 % (incluye la ganancia del derecho a examen y exoneración).
- La aprobación del examen sea menor o igual al 30 %.

En el primer caso se identifica a la UC como posible punto crítico para aprobación de cursada y en el segundo, se identifica a la UC como posible punto crítico para aprobación de examen.

Los resultados con una gráfica por UC en un periodo de tiempo, como se muestra en la figura

Figura 4.1.3: Gráfico extraído del informe de puntos críticos donde se muestra el porcentaje de aprobación para ediciones de cursos de distintas UC.



4.1.4. Informe de duración de carreras

Para este informe se estudia para cada generación y cada carrera el tiempo medio que le tomó a los estudiantes de esa carrera llegar a la mitad de los créditos y terminarla en cada generación. Los datos se muestran en una tabla como en la figura 4.1.4

Figura 4.1.4: Tabla extraída del informe de duración de carreras que muestra la cantidad de años en promedio.

Carrera	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007
Ing. Mecánica	10,1	9,3	9,5	7,6	9,4	9,5	9,4	8,9	8,1	8,4	8,3
Ing. Naval	15,0	9,0		10,0		10,0	10,0	6,0		8,0	
Ing. Civil	9,1	9,7	10,0	9,2	9,3	8,7	8,9	8,6	7,8	7,9	8,2
Ing. Eléctrica	8,6	8,8	9,1	8,9	8,9	8,8	8,3	8,7	8,9	7,6	8,0
Agrimensura	9,1	8,2	9,2	10,3	8,7	9,5	12,7	10,6		7,3	7,0
Ing. Química		11,0	9,7	9,4	8,9	9,8	10,0	9,6	9,0	8,4	7,9
Ing. Alimentos			12,5	10,7	9,3	9,3	8,8	9,4	8,7	8,4	7,9
Ing. Computación	10,9	10,4	10,3	9,8	9,7	10,0	9,4	9,3	8,8	8,3	8,2

4.1.5. Informe de Indicadores de Seguimiento del Plan de Estudios

En dicho informe se presentan un conjunto de indicadores con el fin de dar una visión integral de una carrera, tomando como base para el análisis a los estudiantes, en sus distintas etapas: cuando están activos, egresan o se desvinculan de la carrera. Los estudiantes considerados corresponden a un rango de generaciones.

Para los estudiantes activos el primer indicador a tener en cuenta es la **Distribución de Estudiantes Activos (DEA)**. Es el porcentaje de alumnos activos en cada carrera respecto al total de alumnos activos de todas las carreras de grado [1] discriminado por generación:

$$DEA = \frac{(activos\ carrera)_t}{(activos\ de\ todas\ las\ carreras)_t}; t: \ generación$$

El mismo se visualiza mediante una tabla como muestra la figura 4.1.5 donde se puede ver la cantidad de activos de la carrera seleccionada, la cantidad de activos del resto de las carreras y el valor del DEA.

[1] Las carreras de grado consideradas son: 21-0 Ing. en Sist. de la Comunicacion , 22-2 Ing. de Producción, 22-3 Ing. Industrial Mecánica, 22-4 Ing. Naval, 22-5 Ing. Civíl, 22-8 Ing. Eléctrica, 42-0 Agrimensura, 53-0 Ing. Química, 56-0 Ing. de Alimentos, 72-0 Ing. en Computación

Figura 4.1.5: Tabla extraída del informe de indicadores que muestra los estudiantes activos y DEA.

Gen	Activos.Carrera	Activos.Total	DEA
2008	103	255	40.4
2009	127	283	44.9
2010	149	359	41.5
2011	167	409	40.8
2012	219	545	40.2
2013	275	732	37.6
2014	310	824	37.6
2015	332	975	34.1
2016	411	1214	33.9
2017	623	1717	36.3
2018	594	1575	37.7

El segundo indicador para los estudiantes activos es el **avance por franja de créditos**, el cual muestra cómo los estudiantes activos de la carrera se dividen dentro de las franjas de créditos. Todas las carreras analizadas necesitan un total de 450 créditos para ser completadas, y para medir el avance se toman franjas de créditos equiespaciadas cada 45 créditos, a excepción de la franja inicial que solo abarca al cero y la franja de egresados (que técnicamente no son estudiantes activos pero se agregan para un panorama más completo). Para cada franja se obtiene el porcentaje de estudiantes que la componen. Este indicador está compuesto por tres gráficas:

- Avance por franja de créditos dentro de la carrera seleccionada como en la figura 4.1.6.
- Avance comparativo por franja de créditos de la carrera seleccionada con el resto de las carreras como en la figura 4.1.7.
- Tabla con la cantidad de estudiantes de cada generación que se encuentran en dicha franja, como en la figura 4.1.8.

Figura 4.1.6: Gráfico extraído del informe de indicadores con la cantidad de estudiantes activos por franja de la carrera seleccionada.

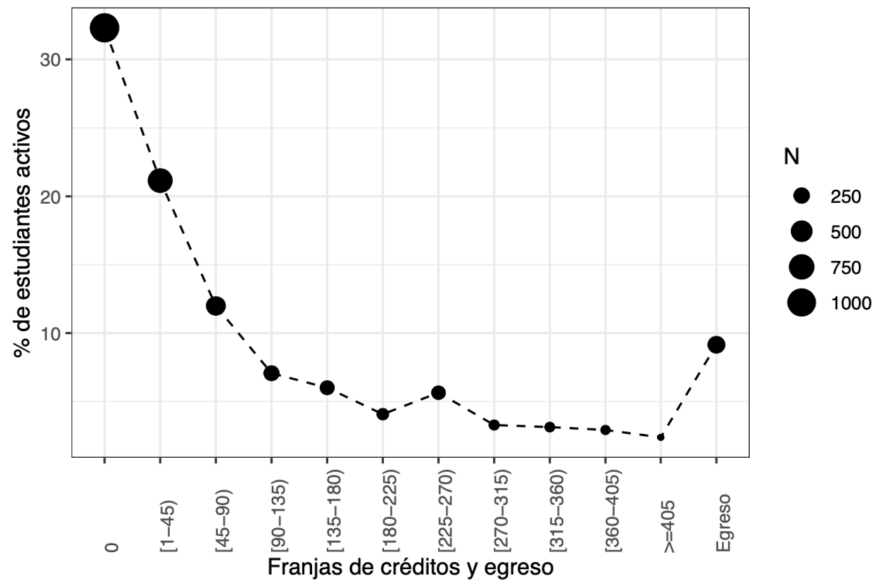


Figura 4.1.7: Gráfico extraído del informe de indicadores con el porcentaje de estudiantes en cada franja de la carrera seleccionada en comparación a las otras carreras.

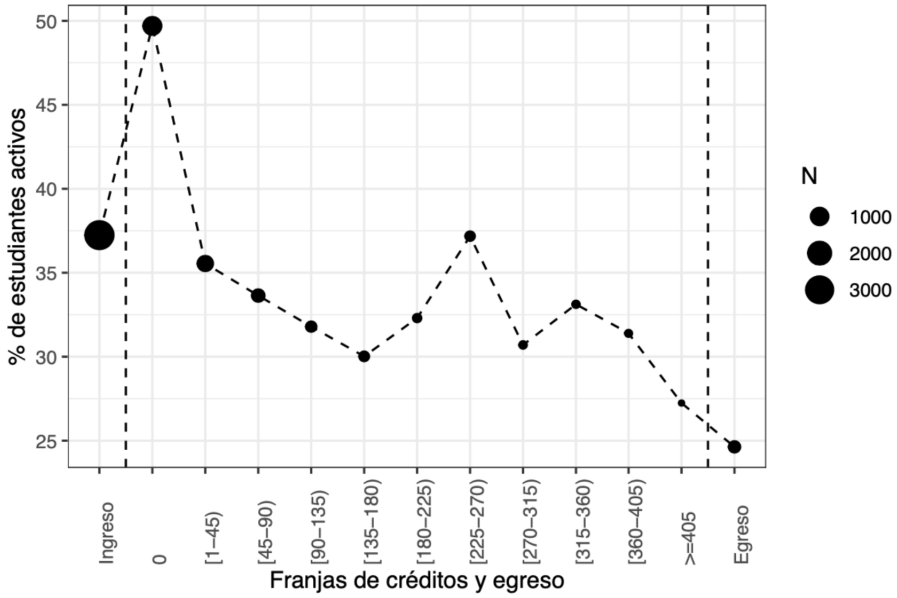


Figura 4.1.8: Tabla extraída del informe de indicadores con la cantidad de estudiantes activos por franja para cada generación.

	0	[1,45)	[45,90)	[90,135)	[135,180)	[180,225)	[225,270)	[270,315)	[315,360)	[360,405)	>=405
2008	8	14	9	11	11	5	14	6	8	8	9
2009	12	21	11	7	10	11	19	9	5	14	8
2010	10	22	14	7	9	11	22	9	12	12	21
2011	10	21	20	15	18	13	21	16	14	11	8
2012	33	38	24	20	13	9	19	16	14	24	9
2013	31	38	34	30	30	21	17	24	19	18	13
2014	48	56	45	24	29	21	32	13	23	8	11
2015	69	88	53	37	26	22	17	12	7	1	0
2016	136	93	60	45	26	21	24	4	2	0	0
2017	345	155	58	35	27	1	2	0	0	0	0
2018	367	154	69	3	0	0	0	0	0	1	0
Total	1069	700	397	234	199	135	187	109	104	97	79

Para los estudiantes egresados el primer indicador es la **Tasa Terminal de la Carrera (TTC)** que se calcula como:

$$TTC = \frac{egresados_t}{inscriptos_t}; t : \textit{generación}$$

El TTC es visualizado por generación en un formato tabla como lo muestra la figura 4.1.9 junto a la cantidad de ingresos y egresos.

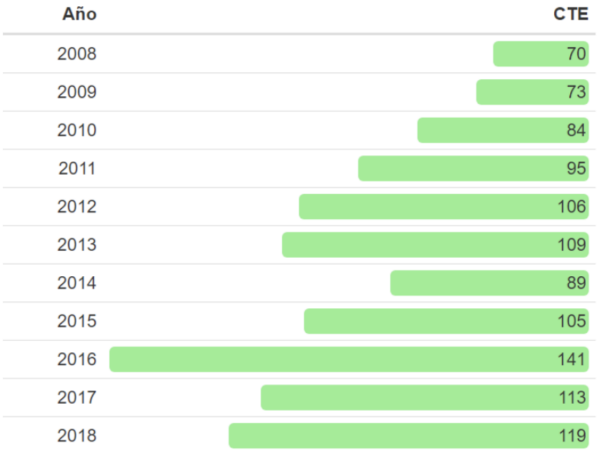
Figura 4.1.9: Tabla extraída del informe de indicadores donde se visualiza el TTC.

Generación	Egresos	Inscriptos	%TTC
2008	83	550	15.09
2009	65	527	12.33
2010	75	648	11.57
2011	44	633	6.95
2012	23	585	3.93
2013	9	603	1.49
2014	4	590	0.68
2015	0	562	0.00
2016	0	573	0.00
2017	0	623	0.00
2018	0	594	0.00

El segundo indicador para los estudiantes egresados es el de la **Cantidad de títulos expedidos por año (CTE)**, y se visualiza como se muestra en la figura

4.1.10

Figura 4.1.10: Tabla extraída del informe de indicadores donde se visualiza el CTE.



Año	CTE
2008	70
2009	73
2010	84
2011	95
2012	106
2013	109
2014	89
2015	105
2016	141
2017	113
2018	119

Otros dos indicadores para los egresados son **Tasa Bruta de Eficiencia Terminal de la Carrera (TBrETC)** y **Tasa neta de Eficiencia Terminal de la Carrera (TNeETC)** que se calculan como:

$$TBrETC = \frac{egresados_{t_e}}{inscriptos_{t_e - Dc+1}} \quad TNeETC = \frac{egresados_{t_e}, inscriptos_{t_e - Dc+1}}{inscriptos_{t_e - Dc+1}}$$

$$Dc : duración\ teórica\ de\ la\ carrera, t_e : año\ de\ egreso$$

Los indicadores TBrETC y TNeETC se visualizan en formato tabla por generación como en la figura 4.1.11.

Figura 4.1.11: Tabla extraída del informe de indicadores donde se visualizan los indicadores de TBrETC y TNeETC.

Año egreso	TBrETC	%TBrETC	TNeETC	%TNeETC
2008	0.150	15.02	0.0193	1.93
2009	0.141	14.12	0.0251	2.51
2010	0.160	16.03	0.0267	2.67
2011	0.211	21.06	0.0200	2.00
2012	0.221	22.08	0.0333	3.33
2013	0.215	21.46	0.0315	3.15
2014	0.162	16.18	0.0145	1.45
2015	0.199	19.92	0.0209	2.09
2016	0.218	21.76	0.0386	3.86
2017	0.179	17.85	0.0221	2.21
2018	0.203	20.34	0.0239	2.39

El último de los indicadores para los egresados es el indicador de **Coeficiente de la eficiencia terminal de la carrera (CETC)**, el cual mide la eficiencia de la carrera mediante la proporción del tiempo utilizado para la culminación de la carrera y el tiempo teórico previsto por el plan de estudios. Se calcula de la siguiente forma:

$$CETC = \frac{mediana\ de\ la\ duraciAcuteon\ de\ la\ carrera_{t_e}}{Dc}$$
$$Dc : duración\ teórica\ de\ la\ carrera, t_e : año\ de\ egreso$$

El indicador CTEC se puede visualizar por medio de una tabla como en la figura 4.1.12 donde para cada generación se puede ver la mediana de tiempo que le lleva terminar la carrera y el indicador CETC.

Figura 4.1.12: Tabla extraída del informe de indicadores donde se visualiza el indicador CETC.

Año egreso	MedTiempEgr	CETC
2008	8.45	1.69
2009	8.20	1.64
2010	8.90	1.78
2011	9.30	1.86
2012	8.90	1.78
2013	8.80	1.76
2014	8.70	1.74
2015	8.40	1.68
2016	8.50	1.70
2017	8.90	1.78
2018	8.70	1.74

Por último para los estudiantes que se desvincularon de la carrera se calcula la **Tasa de Abandono por generación (TA)** la cual mide la relación de estudiantes que se desvincularon en relación a los activos y se calcula como:

$$TA = \frac{inactivos\ carrera_t}{activos\ carrera_t}; t: \ generación$$

La TA se visualiza en una tabla con una fila por generación utilizando barras para facilitar la visualización como en la figura 4.1.13.

Figura 4.1.13: Tabla extraída del informe de indicadores donde se visualiza el indicador TA.

Gen	TA
2008	3.53
2009	2.64
2010	2.85
2011	2.53
2012	1.57
2013	1.16
2014	0.89
2015	0.69
2016	0.39

4.2. Nuevos indicadores

De las reuniones con la UEFI donde se analizaron los informes anteriormente descritos, también surgieron nuevos indicadores de interés que se detallarán a continuación.

4.2.1. Distribución de estudiantes por unidad curricular

Para este indicador interesa saber el resultado de los estudiantes en cada edición de un curso de una unidad curricular. Este análisis es una versión más detallada del elaborado en el informe de puntos críticos. Se toman en cuenta las distintas características de la persona (definidas en el informe de ingresos) y se dividen a los estudiantes en las siguientes categorías:

- **Reprobaron:** Los que no llegaron al puntaje mínimo para poder aprobar el curso o llegaron a la cantidad máxima de posibilidades para dar el examen.
- **Vencidos:** Los que aprobaron el curso pero no llegaron a salvar la UC antes de que se venciera el plazo.
- **Cursando:** Los que aprobaron el curso pero todavía no aprobaron la UC.
- **Exonerados:** Aprobaron la UC llegando al puntaje necesario para exonerar.
- **Salvaron el examen:** Aprobaron la UC mediante la aprobación del examen.

- **Recursaron:** Aprobaron el curso, y antes de que se les venza el plazo se anotaron nuevamente a la UC.

Al igual que en el Informe de Puntos Críticos la visualización de este indicador es a través de un gráfico por unidad curricular, donde se visualiza la cantidad de alumnos en cada grupo para las distintas ediciones de los cursos en un periodo de tiempo.

4.2.2. Tiempo que lleva salvar una unidad curricular

En este caso interesa saber el tiempo promedio que lleva salvar una Unidad Curricular, que se calcula desde la primera vez que el estudiante cursa la UC hasta que la salva. A cada tipo de actividad (curso o examen) se le asigna una fecha de inicio y una fecha de fin, entonces el tiempo que lleva salvar una UC va desde la fecha inicial del primer registro del estudiante hasta la fecha final del registro que acredita que salvó la UC.

Para la visualización de este indicador se tendrá una gráfica por unidad curricular, donde para cada edición de la misma se mostrarán la mediana del tiempo que tardan los estudiantes en salvar la unidad curricular.

Con el repaso por los principales informes que realiza la UEFI, donde se enfatizan los indicadores que se analizan en cada uno de ellos, y los nuevos indicadores surgidos a partir de las reuniones, se tiene la suficiente información para realizar el diseño del Data Warehouse.

4.3. Diseño del Data Warehouse

Como se mencionó al inicio de este capítulo, los Data Warehouses no son más ni menos que bases de datos especialmente diseñadas para ser utilizadas con fines analíticos. Por ende, en la etapa de diseño de un Data Warehouse es donde se aplican una serie de pasos y técnicas para que la base de datos tenga el formato óptimo para satisfacer las necesidades analíticas. Los Data Warehouse y en general los sistemas OLAP se basan en el modelo multidimensional, que visualiza los datos en un espacio n-dimensional, generalmente llamado cubo de datos o hipercubo. Un cubo de datos se define por dimensiones y hechos. Un **hecho** representa el foco del análisis (por ejemplo, el análisis de las ventas en las tiendas) y normalmente incluye atributos llamados medidas. Las **medidas** suelen ser valores numéricos que permiten una evaluación cuantitativa de varios aspectos de una organización. Por ejemplo, medidas como la cantidad o el número de ventas pueden ayudar a analizar

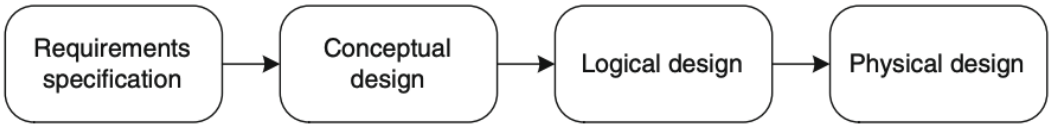
las actividades de ventas en varias tiendas. Las **dimensiones** se utilizan para ver las medidas desde varias perspectivas. Por ejemplo, una dimensión de tiempo se puede utilizar para analizar los cambios en las ventas durante varios períodos de tiempo, mientras que una dimensión de ubicación se puede utilizar para analizar las ventas de acuerdo con la distribución geográfica de las tiendas. Los usuarios pueden combinar varias perspectivas de análisis (es decir, dimensiones) de acuerdo con sus necesidades. Por ejemplo, un usuario puede requerir información sobre las ventas de accesorios de computadora (la dimensión del producto) en julio de 2012 (la dimensión del tiempo) en todas las ubicaciones de las tiendas (la dimensión de la tienda). Las dimensiones suelen incluir atributos que forman **jerarquías**, que permiten a los usuarios explorar medidas en varios niveles de detalle. Ejemplos de jerarquías son mes-trimestre-año en la dimensión de tiempo y ciudad-estado-país en la dimensión de ubicación.

Desde un punto de vista metodológico, los Data Warehouses se diseñan de forma análoga a las bases de datos operativas, siguiendo el proceso de cuatro pasos, como lo muestra la figura 4.3.1, que consiste en:

- **Especificación de requerimientos:** En esta etapa se recopilan las necesidades de los usuarios en varios niveles de la organización, determinando cuales son las consultas y análisis que requieren hacer. En esta etapa se determina que información es la que debe estar disponible y como debe de estar organizada, guiando al diseñador a descubrir los elementos esenciales del esquema multidimensional, como los hechos y sus dimensiones asociadas. La especificación de requerimientos puede hacerse tanto a partir del análisis de las necesidades de los usuarios, de las fuentes de entrada o de un esquema híbrido.
- **Diseño conceptual:** Esta etapa tiene como objetivo construir una representación abstracta de la base de datos para que sea entendible por el usuario a partir de los requerimientos, sin entrar en detalles de cómo va a ser implementado. En esta etapa se construye un modelo conceptual que permita ver los aspectos relevantes del análisis multidimensional, identificándose las dimensiones con sus jerarquías, y los hechos con sus medidas. Para las medidas se deben representar como se comportan ante la agregación de las mismas cuando se navegan a través de las jerarquías. discriminando entre aquellas que se pueden agregar en cualquier jerarquía (aditivas), aquellas que solo se pueden agregar en determinadas jerarquías (semi-aditivas) o las que no se pueden agregar (no aditivas).

- **Diseño lógico:** En esta etapa se realiza la traducción del modelo conceptual a un modelo lógico. En el modelo lógico se representa como van a estar almacenados los cubos multidimensionales, teniendo en cuenta tipo de base de datos que se va a utilizar. Además en esta etapa se tienen en cuenta requisitos no funcionales que no fueron representados en el modelo conceptual. Las formas más comunes para el almacenamiento de cubos son las base de datos relacionales (ROLAP por su sigla en inglés de Relational OLAP) y en base de datos con estructuras de datos especializadas para el almacenamiento de cubos (MOLAP por su sigla en inglés Multidimensional OLAP).
- **Diseño físico:** En esta etapa de diseño se toman en cuenta particularidades de la base de datos utilizada con el fin de realizar optimizaciones para garantizar un buen rendimiento. Estas optimizaciones pueden ser la utilización de índices, vistas materializadas o particionamiento de las tablas.

Figura 4.3.1: Etapas de diseño de un Data Warehouse. Imagen extraída del libro [49].



Para diseñar el Data Warehouse tomamos como entrada los informes elaborados por la UEFI, los nuevos indicadores que surgieron a raíz de las reuniones mantenidas y los datos que se encontraban dentro de la base de datos de Bedelías. Con el análisis anterior se realizan las cuatro etapas de diseño anteriormente descritas y de las mismas surgen las dimensiones y los hechos que se listan a continuación junto con su modelo conceptual utilizando la especificación *CMDM* [30]. El detalle del proceso completo de diseño se encuentra en el apéndice A Diseño completo del Data Warehouse, listando a continuación las dimensiones y los cubos resultantes de esta etapa.

Las dimensiones que surgen de la etapa de diseño son la de Estudiantes, Unidades Curriculares, Carreras, Generaciones, Año Egresos, Franja de créditos, Tiempo cursos y Tiempo exámenes. El modelo conceptual para todas las dimensiones se encuentra en la figura 4.3.2.

La dimensión Estudiantes contiene la información para analizar las medidas por los distintos atributos del estudiante, donde cada una de ellas conforma una jerarquía:

- **Sexo:** Permite visualizar las medidas según el sexo de los estudiantes: masculino o femenino.
- **Rango edad de ingreso:** Posibilita ver las medidas de acuerdo al rango de edad con el que los estudiantes ingresaron a la facultad: 17 a 18, 19 a 20, 21 a 23, 24 a 26, 27 o más.
- **Procedencia geográfica:** Permite visualizar las medidas según si los estudiantes provienen de Montevideo, interior o el exterior.
- **Tipo de institución de procedencia:** Posibilita ver las medidas de acuerdo en que tipo de institución estudiaron los estudiantes: liceos públicos, privados, UTU u otros.

Las dimensión Tiempo Cursos contiene la información para analizar las medidas agrupando el tiempo de diversas formas, cada una de ellas formando las siguientes jerarquías:

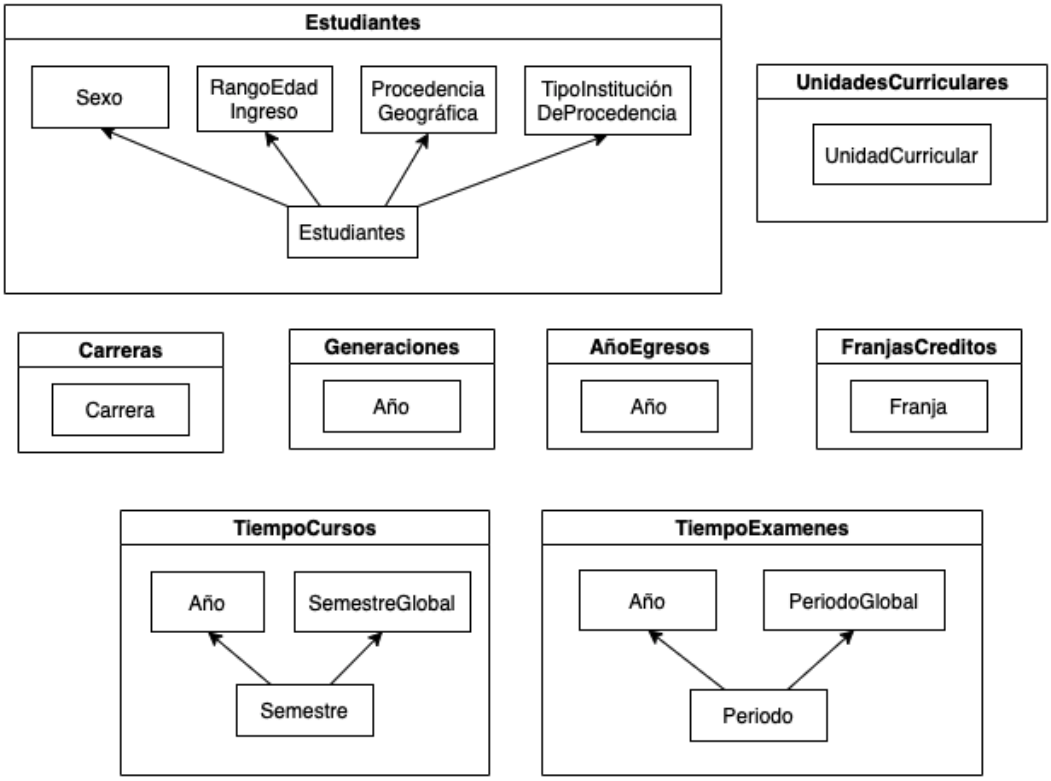
- **Semestre:** Permite visualizar las medidas según el semestre. Por ejemplo: Primer semestre del 2020
- **Año:** Posibilita ver las medidas agrupadas por año
- **Semestre global:** Permite visualizar las medidas según el semestre sin considerar los años, por ejemplo todos los primeros semestres o todos los segundos semestres.

Las dimensión Tiempo Exámenes contiene las siguientes jerarquías:

- **Periodo:** Permite visualizar las medidas según el periodo. Por ejemplo: Febrero del 2020
- **Año:** Posibilita ver las medidas agrupadas por año
- **Periodo global:** Permite visualizar las medidas según el periodo sin considerar los años, por ejemplo todos los Febrero.

Las dimensiones Carreras, Unidades Curriculares, Generaciones, Año Egresos y Franjas Créditos tienen una sola jerarquía y permite visualizar las medidas agrupándolas por carrera, unidad curricular, generación, año egreso y franja de créditos respectivamente.

Figura 4.3.2: Modelo conceptual de las dimensiones del Data Warehouse.



Los cubos que surgen de la etapa de diseño son:

- **Ingresos:** está compuesto por las dimensiones de Estudiantes, Generaciones y Carreras y permite visualizar la medida de la cantidad de estudiantes que se inscribieron.
- **Egresos:** está compuesto por las dimensiones de Estudiantes, Generaciones, Años egresos y Carreras y permite visualizar la medida de la cantidad estudiantes que egresaron.
- **Activos:** está compuesto por las dimensiones de Estudiantes, Generaciones y Carreras y permite visualizar la medida de la cantidad de estudiantes activos.
- **Puntos críticos cursos:** está compuesto por las dimensiones de Unidades Curriculares y Tiempo cursos y permite visualizar la medida del porcentaje de aprobación de la UC.
- **Puntos críticos exámenes:** está compuesto por las dimensiones de Unidades Curriculares y Tiempo Exámenes y permite visualizar la medida

del porcentaje de aprobación del examen.

- **Tiempo duración carrera:** está compuesto por las dimensiones de Estudiantes, Generaciones y Carreras y permite visualizar las medidas del tiempo que se tarda en llegar a la mitad y al total de los créditos de la carrera.
- **Distribución estudiantes en franjas de créditos:** está compuesto por las dimensiones de Estudiantes, Generaciones, Carreras y Franjas créditos y permite visualizar la medida de la cantidad de estudiantes que pertenecen a cada franja.
- **Distribución estudiantes en UC:** está compuesto por las dimensiones de Estudiantes, Generaciones, Carreras, Unidades Curriculares y Tiempo cursos y permite visualizar las medidas del porcentaje de estudiantes que exoneraron, salvaron en examen, están cursando, reprobaron, recursaron y se les venció el plazo para salvarla.
- **Tiempo en salvar UC:** está compuesto por las dimensiones de Estudiantes, Generaciones, Carreras, Unidades Curriculares y Tiempo cursos y permite visualizar la medida de la cantidad de meses promedio que se tarda en salvar la UC.

Una vez terminada la etapa de diseño donde se definió cómo se va a almacenar la información se le da paso a la etapa de implementación de los procesos de extracción, transformación y carga para que los datos lleguen a las tablas correspondientes.

4.4. Proceso de ETL

El proceso de extracción, transformación y carga (ETL por su correspondiente sigla en inglés: Extract, Transform y Load) de un Data Warehouse tiene como finalidad la carga y, a futuro, la actualización de los datos que son almacenados dentro del Data Warehouse. Durante este proceso, los datos se toman de diversas fuentes de origen, se depuran para corregir y filtrar información de mala calidad, se les aplican transformaciones para enriquecerlos y adaptarlos al formato esperado, y por último son guardados dentro del Data Warehouse. Además se prevé mecanismos para la correcta actualización de los datos ante futuras ejecuciones de dicho proceso.

Las herramientas seleccionadas para este proceso son SQLite [21], Postgres [19] y Pentaho Data Integration [18], también conocido como Kettle. SQLite y Postgres

son motores de base de datos relacionales, donde el primero es muy liviano pero con funcionalidades muy básicas y el segundo es más completo y avanzado. SQLite se usa para acceder a la base de datos brindada por Bedelías y Postgres para dar soporte al Data Warehouse. Por otro lado, Kettle es una herramienta que permite la manipulación de datos y la creación de procesos de ETL a través de una interfaz gráfica. Esta herramienta nos permite crear dos tipos de archivos, los archivos de tipo transformaciones (.ktr) y los archivos de tipo Job (.kjb). Las transformaciones se utilizan para describir los flujos de datos para ETL, como leer desde una fuente, transformar datos y cargarlos en una ubicación de destino. Los jobs son los encargados de orquestar el proceso de ETL ejecutando una serie de pasos en un orden en específico, donde los pasos pueden ser transformaciones u otros jobs.

Una vez instaladas las herramientas y levantado el servidor de Postgres, hay que crear una base de datos con el nombre `pgrado`. Si se desea colocar otro nombre a la base de datos hay que cambiar la configuración de conexión de las herramientas que lo utilicen. Este es el único paso que se debe hacer de forma manual, y por única vez luego de instalar Postgres.

En el repositorio de archivos fuente del proyecto [24] se encuentran todos los archivos necesarios para el funcionamiento del Data Warehouse. Particularmente para el proceso de ETL se usan los archivos que se encuentran dentro de la carpeta llamada `etl`, que a su vez están divididos en dos subcarpetas llamadas `inputs` y `kettle_files`, como se observa en la figura 4.4.1. La carpeta `inputs` contiene los archivos que el proceso de ETL precisa como entrada y la carpeta `kettle_files` contiene los archivos de las transformaciones y el job que implementan el proceso de ETL.

Figura 4.4.1: Archivos y carpetas utilizados por el proceso de ETL.

Dentro de la carpeta `inputs` el archivo `db_bedelias.db` es el archivo que contiene la base de datos de Bedelías y el archivo `db_datawarehouse_setup.sql` contiene el script para la inicialización de los schemas y las tablas de la base de datos Postgres. Dentro de la carpeta `kettle_files` se halla el archivo `DatawarehouseJob.kjb` que es el encargado de orquestar el proceso de ETL llamando a cada una de las transformaciones, las cuales son los restantes archivos dentro de la carpeta.

La base de datos Postgres, luego de aplicar el script de inicialización, contiene dos schemas: `public` y `datawarehouse`, donde el primer schema contiene tablas que son usadas tanto por el proceso de ETL como los análisis que se realizan en los siguientes capítulos, y el segundo schema contiene solo las tablas de dimensión y de hechos del Data Warehouse.

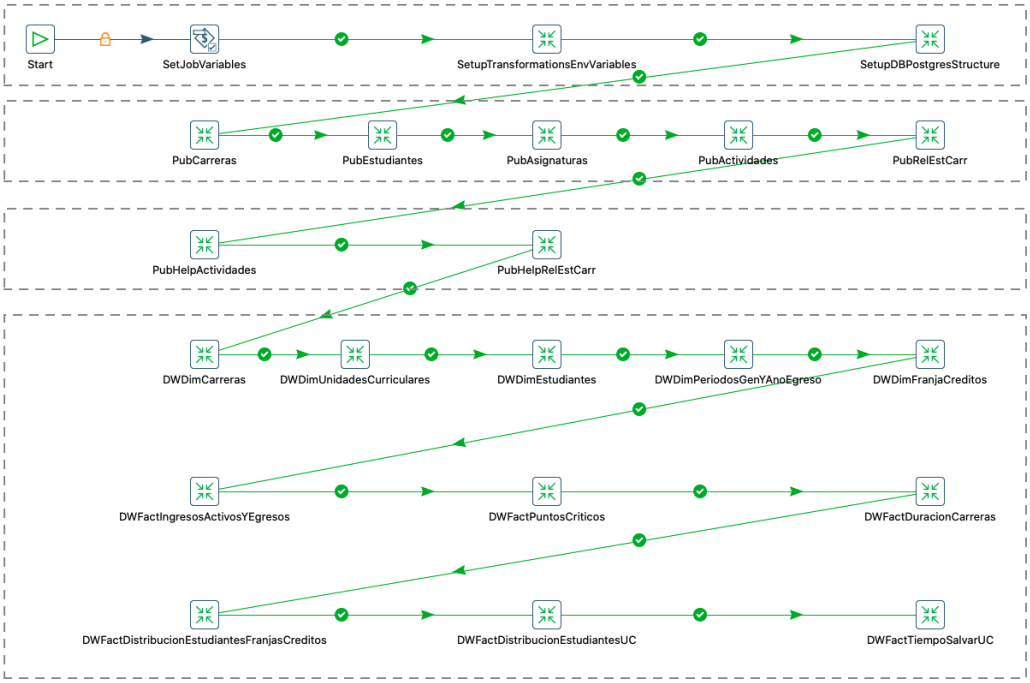
Si se analiza el contenido del archivo DatawarehouseJob.kjb se puede ver, como muestra la imagen la figura 4.4.2, que la implementación del proceso de ETL está dividido en múltiples pasos, y los mismos se pueden catalogar en cinco módulos:

- Inicialización, donde se encuentra el paso que da inicio al job, los pasos donde se inicializan variables necesarias para la ejecución de las transformaciones posteriores y por último se ejecuta el script de inicialización de la base de

datos.

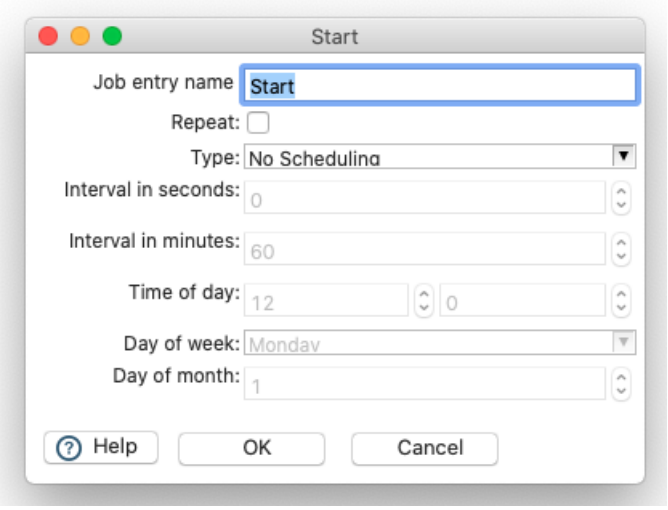
- La extracción de los datos de la base de datos provista por Bedelías y guardados en tablas en Postgres casi con el mismo formato. Las únicas transformaciones que se hacen son el pasaje de las columnas que representan fechas y están en un formato numérico a un formato de tipo fecha.
- El llenado de tablas intermedias donde se almacenan datos precalculados que son usados en múltiples pasos posteriores.
- El llenado de las tablas de las dimensiones.
- El llenado de las tablas de hechos.

Figura 4.4.2: Implementación del job DatawarehouseJob.kjb.



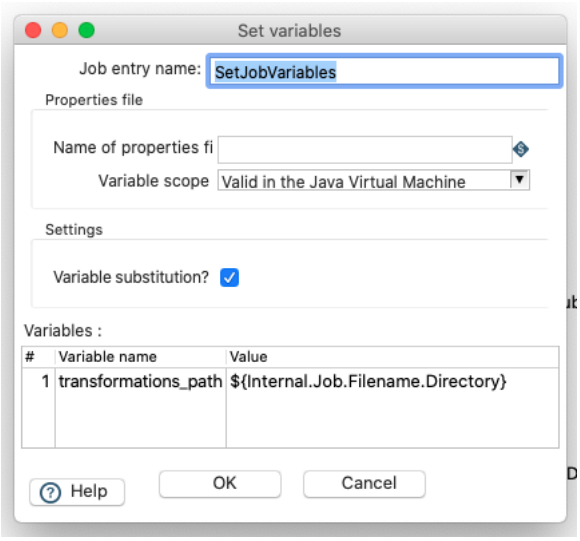
El paso **Start** (figura 4.4.3) es el inicial y permite configurar cada cuanto se quiere ejecutar el Job, que en nuestro caso la ejecución es manual.

Figura 4.4.3: Paso inicial del job.



El paso **SetJobVariables** establece la variable de entorno `transformations_path` que indica la ruta en la que están ubicadas las transformaciones. El valor por defecto toma la misma ruta en la que está ubicado el Job.

Figura 4.4.4: Paso SetJobVariables.



El paso de **SetupTransformationsEnvVariables** ejecuta la transformación `SetupTransformationsEnvVariables.ktr` y setea las variables las siguientes variables de entorno que se muestran en el cuadro 4.4.1.

Cuadro 4.4.1: Tabla con las variables de entorno que se inicializan en la transformación SetupTransformationsEnvVariables.ktr.

Variable	Descripción	Valor
inputs_path	Es la ruta donde se encuentran los archivos de entrada. Este valor es relativo a la ruta que está alojado el archivo de la la transformación.	../inputs/db_bedelias.db
db_fing_sqlite_url	Es la url que se necesita configurar para acceder a la base de datos de Bedelías.	jdbc:sqlite:${inputs_path} /db_bedelias.db
postgres_db_host	IP o dominio donde se encuentra alojada la base de datos Postgres.	localhost
postgres_db_port	Puerto donde se expone la base de datos Postgres.	pgrado (modificar si se la crea con otro nombre)
postgres_db_name	Nombre de la base de datos. Postgres	pgrado (modificar si se la crea con otro nombre).
postgres_db_user	Usuario para conectarse a la base de datos Postgres.	postgres
postgres_db_pass	Contraseña para conectarse a la base de datos Postgres.	postgres
min_activity_date _to_be_active	La fecha mínima de la última actividad de un estudiante para ser considerado activo.	01/04/2017

El paso **SetupDBPostgresStructure** ejecuta la transformación `SetupDBPostgresStructure.ktr` (figura 4.4.5) y es la encargada de ejecutar el script de inicialización de la base de datos Postgres, ubicado en `$inputs_path/db_datawarehouse_setup.sql`, siendo `$inputs_path` la variable definida anteriormente.

Figura 4.4.5: Transformación SetupDBPostgresStructure.

El paso **PubCarreras** ejecuta la transformación `PubCarrerasETL.ktr` (figura 4.4.6) y es la encargada de pasar los datos de la tabla `Asignaturas` de la base de datos de Bedelías a la tabla `public.asignaturas` de la base de datos Postgres.

Figura 4.4.6: Transformación PubCarrerasETL.

El paso **PubEstudiantes** ejecuta la transformación `PubEstudiantesETL.ktr` (figura 4.4.7) y es la encargada de pasar los datos de la tabla `Estudiantes` de la base de datos de Bedelías a la tabla `public.estudiantes` de la base de datos Postgres.

Figura 4.4.7: Transformación PubEstudiantesETL.

El paso **PubAsignaturas** ejecuta la transformación `PubAsignaturasETL.ktr` (figura 4.4.8) y es la encargada de pasar los datos de la tabla `asigcarr` de la base de datos de Bedelías a la tabla `public.asignaturas` de la base de datos Postgres.

Figura 4.4.8: Transformación PubAsignaturasETL.

El paso **PubActividadesETL** ejecuta la transformación `PubActividadesETL.ktr` (figura 4.4.9) y es la encargada de pasar los datos de la tabla `Activ2` de la base de datos de Bedelías a la tabla `public.actividades` de la base de datos Postgres.

Figura 4.4.9: Transformación PubActividadesETL.

El paso **PubRelEstCarr** ejecuta la transformación `PubRelEstCarrETL.ktr` y es la encargada de pasar los datos de la tabla `EstudCarr` de la base de datos de Bedelías a la tabla `public.rel_est_carr` de la base de datos Postgres.

Figura 4.4.10: Transformación PubRelEstCarrETL.

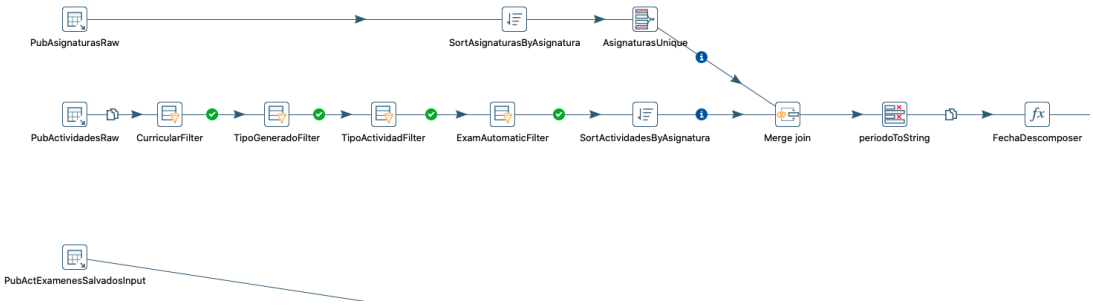
El paso **PubHelpActividades** ejecuta la transformación `PubHelpActividadesETL.ktr` y es la encargada de cargar la tabla `public.help_actividades` de la base de datos Postgres. Esta transformación es la encargada de extraer, filtrar y calcular información extra sobre las actividades de los estudiantes que se encuentran en la tabla `public.actividades`, con el fin de facilitar el trabajo en múltiples transformaciones posteriores. La extracción y filtrado se puede ver en la figura 4.4.11 y el cálculo de los campos en la figura 4.4.12. Las actividades que son filtradas son aquellas que no cumplen con las siguientes condiciones:

- Actividades no curriculares, es decir aquellas actividades cuyo valor de la columna `curricular` sea distinto de `''C''`.
- Actividades que no se registren de forma automática o normal, es decir aquellas actividades cuyo valor de la columna `tipogenerado` sea distinto de `''A''` o `''N''`.
- Actividades que no correspondan a cursos o exámenes, es decir aquellas actividades cuyo valor de la columna `tipoactividad` sea distinto de "C",

"N", "D" o "E"

- Actividades que se generen a partir de la exoneración de un curso, es decir aquellas actividades cuyo valor de la columna `tipoactividad` sea igual a `''E''` y que el valor de la columna `tipogenerado` sea igual a `''A''`.

Figura 4.4.11: Extracción y filtrado realizado en la transformación PubHelpActividadesETL.



El primero de los campos calculados es el que identifica en qué periodo se realizó la actividad, y se almacena en la columna `idperiodo` de la tabla `public.help_actividades`. Los periodos de los cursos se identifican por el año y el semestre en que se realizó (par o impar), y el periodo de los exámenes se determinan por el año y el mes. Este campo es de interés en aquellas transformaciones que populan tablas de hechos donde se utilicen las dimensiones de **TiempoCursos** y **TiempoExamenes**. Si bien la tabla `public.actividades` tiene una columna llamada periodo, el mismo solo tiene valores para los exámenes, y de allí es tomado para calcular el identificador del período para los mismos. Para los cursos el cálculo se realiza a partir del valor de la columna fecha, donde se extrae el año y el mes y se realiza el siguiente cálculo:

- Si el mes está entre Enero y Mayo se considera que esa actividad corresponde a un curso realizado en el segundo semestre del año anterior al extraído de la fecha.
- Si el mes está entre Junio y Octubre se considera que esa actividad corresponde a un curso realizado en el primer semestre correspondiente al año extraído de la fecha.
- Si el mes es Noviembre o Diciembre se considera que esa actividad corresponde a un curso realizado en el segundo semestre correspondiente al año extraído de la fecha.

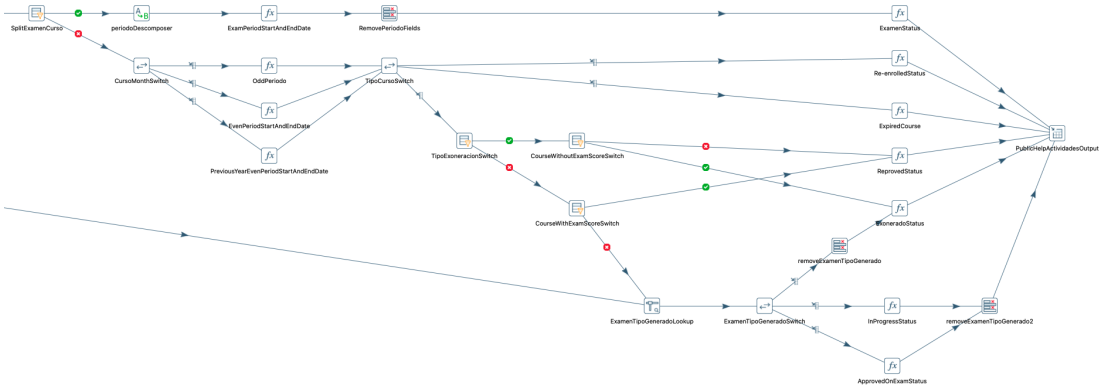
Otros dos campos que se calculan en esta transformación son la fecha de inicio y

la fecha de fin de las actividades, y se almacenan en las columnas `startdate` y `enddate` de la tabla `public.help_actividades`. Estos campos permiten calcular períodos de tiempo como por ejemplo el tiempo que se tarda a la mitad de la carrera o el tiempo que se tarda en salvar una unidad curricular. Teniendo el fin de estos campos en mente para los cursos se toman las fechas de inicio y fin con una distancia entre ellas de seis meses y para los exámenes dicha diferencia es de un mes. Para los periodos de cursos en semestre impar la fecha de inicio corresponde al primero de Enero y termina el primero de Julio del año correspondiente, y para los cursos de semestre par la fecha de inicio corresponde al primero de Julio del año correspondiente y termina el primero de Enero del año próximo. Para los exámenes la fecha de inicio es calculada con el primer día del mes y la fecha de fin es calculada como el primer día del mes siguiente, siempre tomando el mes y año correspondiente al periodo.

El último campo calculado en esta transformación es el correspondiente al estado de esa actividad, y se guarda en la columna `status` de la tabla `public.help_actividades`. Este campo es usado tanto para saber la distribución de los estudiantes en una edición de un curso en una unidad curricular así como para saber cuánto tiempo demora un estudiante en salvar una unidad curricular (para este último también se usa la fecha de fin calculada anteriormente). El cálculo de este campo sigue la siguiente lógica:

- Para los exámenes, si la nota es mayor o igual a 3 el valor es `approved` y sino el valor es `reproved`.
- Para los cursos, si el campo `tipoActividad` es igual a `‘‘N’’` o `‘‘D’’` el valor asignado a `reenrolled` o `expired` respectivamente.
- Para los cursos, si el campo `tipoActividad` es igual a `‘‘C’’` si la nota es menor a 3 se asigna el valor `reproved`. En cambio si la nota es mayor o igual a 3 se toma en cuenta si la unidad curricular tiene o no examen (se toma del campo `TipoExo` de la tabla `public.asignaturas`), donde si la asignatura no tiene examen se le asigna el valor `exonerated` y si tiene examen se busca el resultado de algún examen con fecha posterior del estudiante para la unidad curricular y se le asigna los valores `approved_on_exam`, `exonerated` o `in_progress`.

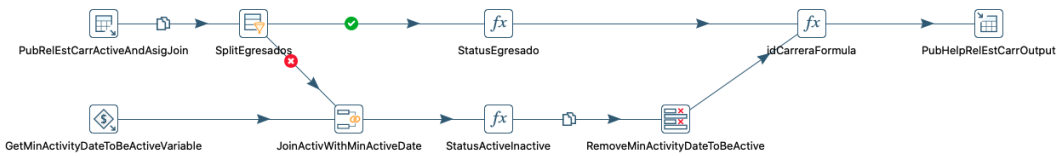
Figura 4.4.12: Cálculo de información realizado en la transformación PubHelpActividadesETL.



El paso **PubHelpRelEstCarr** ejecuta la transformación `PubHelpRelEstCarrETL.ktr` (figura 4.4.13) y es la encargada de popular la tabla `public.help_rel_est_carr` de la base de datos Postgres. Esta transformación calcula información extra sobre la relación entre los estudiantes y las carreras a las que están inscriptos, tomando como entrada los registros de la tabla `public.rel_est_carr`. Para cada registro calcula el estado del estudiante en la carrera con la siguiente lógica:

- Si el estudiante está recibido, se le asigna el valor `graduate`.
- Si no, se obtiene la fecha de la última actividad del estudiante y es comparada con la fecha cargada en la variable `min_activity_date_to_be_active`, donde si es menor se le asigna el estado `inactive` y si es mayor o igual se le asigna el valor `active`.

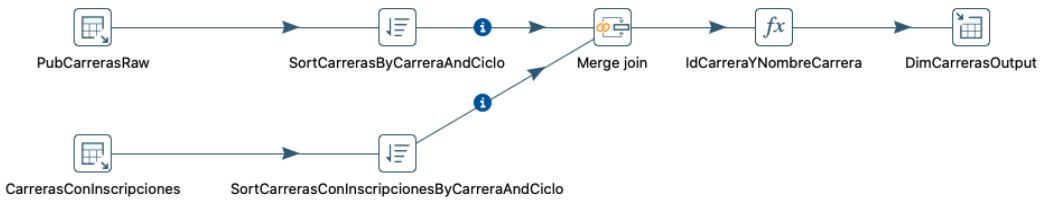
Figura 4.4.13: Transformación PubHelpRelEstCarrETL.



El paso `DWDimCarreras` ejecuta la transformación `DWDimCarrerasETL.ktr` (figura 4.4.14) y es la encargada de cargar la tabla de dimensión de las carreras `datawarehouse.dim_carreras`. Los datos son extraídos de la tabla `public.carreras` y se filtran aquellas carreras que no tienen inscripciones desde

1980 tomando como datos de entrada la relación entre los estudiantes y las carreras de la tabla `public.rel_est_carr`.

Figura 4.4.14: Transformación DWDimCarrerasETL.

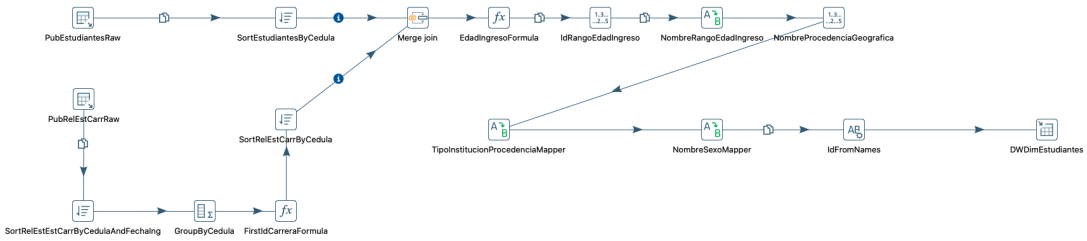


El paso **DWDimUnidadesCurriculares** ejecuta la transformación `DWDimUnidadesCurricularesETL.ktr` (figura 4.4.15) y es la encargada de popular la tabla de dimensión de las unidades curriculares `datawarehouse.dim_unidades_curricualres`. Los datos son extraídos de la tabla `public.asignaturas`.

Figura 4.4.15: Transformación DWDimUnidadesCurricularesETL.

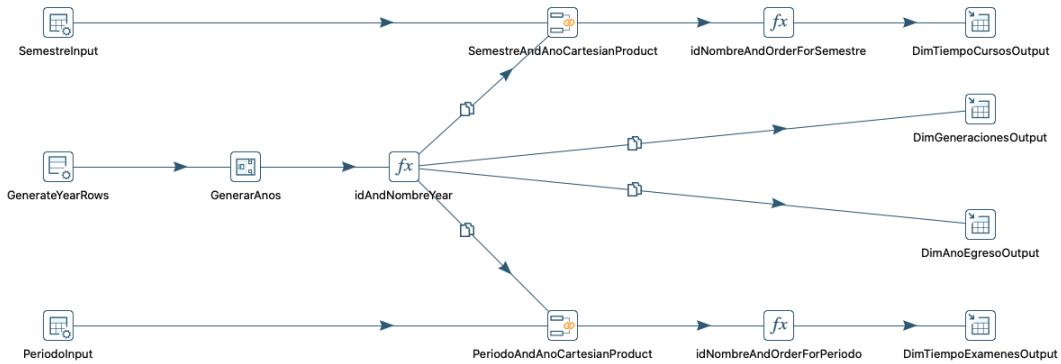
El paso **DWDimEstudiantes** ejecuta la transformación `DWDimEstudiantesETL.ktr` (figura 4.4.16) y es la encargada de cargar la tabla de dimensión de los estudiantes `datawarehouse.estudiantes`. Los datos son extraídos de la tabla `public.estudiantes` para obtener los datos de los estudiantes y de `public.rel_est_carr` para obtener la información de la primera carrera a la que se inscribió el estudiante. Con los datos obtenidos se calcula el rango de edad en que se encontraba el estudiante al ingresar a la facultad y los demás campos de la dimensión.

Figura 4.4.16: Transformación DWDimEstudiantesETL.



El paso **DWDimPeriodosGenYAnoEgreso** ejecuta la transformación `DWDimPeriodosGenYAnoEgresoETL.ktr` (figura 4.4.17) y es la encargada de llenar las tablas de dimensión referentes a tiempo como lo son `datawarehouse.dim_tiempo_cursos`, `datawarehouse.dim_tiempo_examenes`, `datawarehouse.dim_generaciones` y `datawarehouse.dim_ano_egreso`. Los datos de entrada son generados dentro de la misma transformación, donde para los años se genera una secuencia de números que van desde 1950 a 2050 y para los semestres y los periodo se genera una tabla con el nombre y el identificador de los mismos. Luego se hace el producto cartesiano de los años con cada una de las tablas generadas obteniendo así los registros de cada semestre y cada periodo correspondiente a cada uno de los años.

Figura 4.4.17: Transformación DWDimPeriodosGenYAnoEgresoETL.

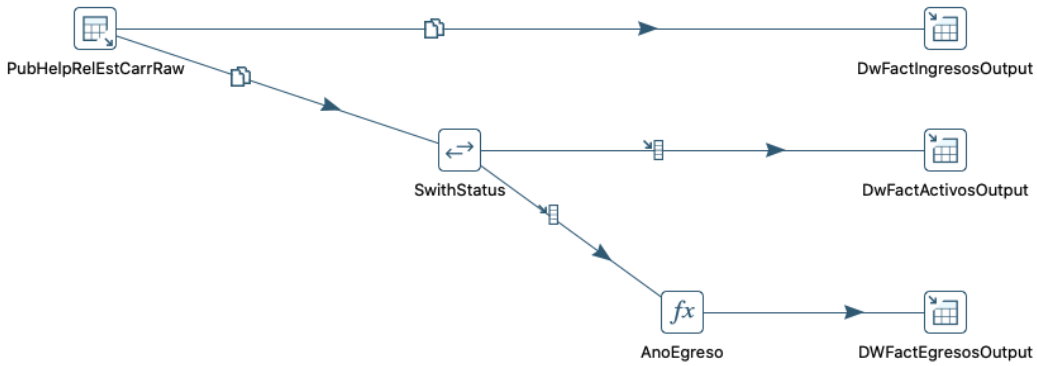


El paso **DWDimFranjasCreditos** ejecuta la transformación `DWDimFranjasCreditosETL.ktr` (figura 4.4.18) y es la encargada de popular la tabla de dimensión de las franja de crédito `datawarehouse.dim_franja_creditos`. Los datos de entrada son generados dentro de la misma transformación con una tabla que contiene un registro por cada una de las franjas de créditos requeridas.

Figura 4.4.18: Transformación DWDimFranjasCreditosETL.

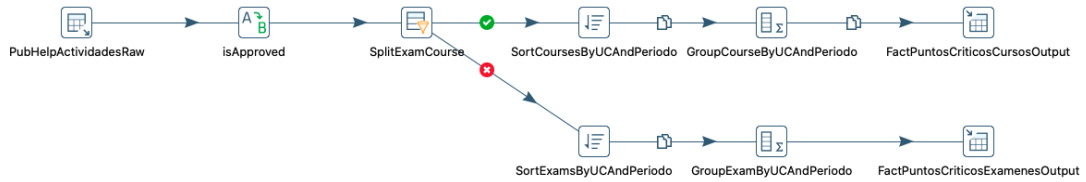
El paso **DWFactIngresosActivosYEgresos** ejecuta la transformación `DWFactIngresosActivosYEgresosETL.ktr` (figura 4.4.19) y es la encargada de llenar las tablas de `datawarehouse.fact_ingresos`, `public.fact_activos` y `datawarehouse.fact_egresos` a partir de la tabla `public.help_rel_est_carr`. Para el caso de los ingresos el traspaso es total, y para los activos y egresados se traspasan aquellos registros cuyo valor del campo `status` es active o `graduate` respectivamente.

Figura 4.4.19: Transformación DWFactIngresosActivosYEgresosETL.



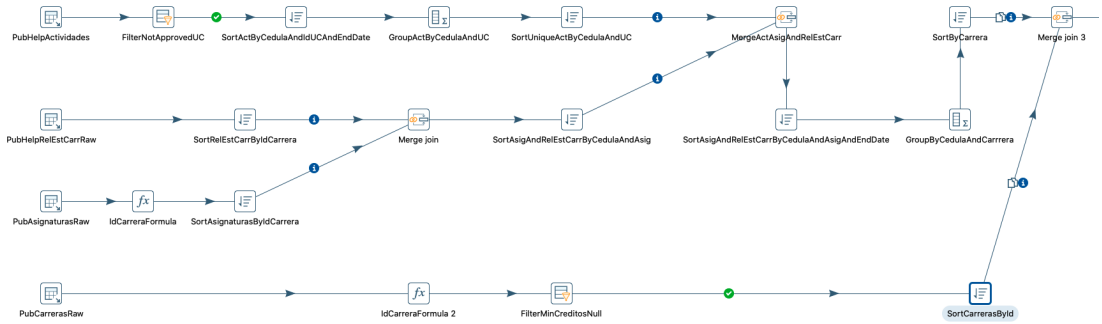
El paso **DWFactPuntosCriticos** ejecuta la transformación `DWFactPuntosCriticosETL.ktr` (figura 4.4.20) y es la encargada de llenar las tablas `datawarehouse.fact_puntos_criticos_cursos` y `public.fact_puntos_criticos_examenes`. Esta transformación calcula la cantidad de personas que se inscribieron y salvaron un curso o examen de una unidad curricular. Esta transformación toma como entrada los registros de la tabla `public.help_actividades` y a partir del campo `status` calcula si el estudiante aprobó dicha actividad.

Figura 4.4.20: Transformación DWFactPuntosCriticosETL.



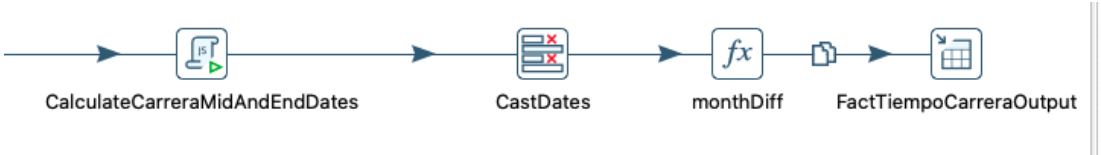
El paso **DWFactDuracionCarreras** ejecuta la transformación `DWFactDuracionCarrerasETL.ktr` y es la encargada de cargar la tabla `datawarehouse.fact_tiempo_carrera`. En esta transformación se calcula la cantidad de meses que pasaron desde que el alumno entró a la facultad hasta que el estudiante alcanzó la mitad y el fin de cada carrera a la que se inscribió. Se dice que un estudiante llega a la mitad o al final de la carrera cuando obtiene la mitad o al total de los créditos que la carrera requiere para ser terminada (dicho valor es obtenido del campo `mincreditos` de la tabla `public.carreras`). Se toma como fecha de inicio el primero de Enero del año correspondiente a la generación del estudiante, dado que si tomamos el valor del campo `fechaingreso` de la tabla `public.help_rel_est_carr` los cálculos serían incorrectos para aquellos casos donde los estudiantes empiezan en una carrera y luego se cambian a otra y dichas carreras comparten unidades curriculares. Como primer paso (figura 4.4.21) se deben obtener información sobre las unidades curriculares salvadas por los estudiantes en cada carrera en la que esté inscripto, en particular cuántos créditos aporta y la fecha en que fue salvada cada unidad curricular. Para ello se toma como datos de entrada las actividades de los estudiantes (tabla `public.help_actividades`), las carreras a las que está inscripto (tabla `public.help_rel_est_carr`) y la información de a qué carrera pertenece y la cantidad de créditos que otorga cada unidad curricular (tabla `public.asignaturas`) y se combinan para obtener la información deseada. Cabe destacar que si el estudiante salva una unidad curricular que pertenece a más de una carrera en la que está inscripto, se genera un registro por cada una de las carreras correspondientes.

Figura 4.4.21: Primera parte de la transformación DWFactDuracionCarrerasETL.



Con la información antes calculada (figura 4.4.22) y la información de cuántos créditos son necesarios para terminar cada carrera (tabla `public.carreras`), se obtienen las fechas en que el alumno alcanzó la mitad y el total de los créditos de la carrera. Esto se logra agrupando los registros calculados anteriormente por estudiante y carrera en la que está inscripto, ordenándolos por fecha y sumando los créditos obtenidos hasta obtener la actividad con la cual se alcanzó la mitad de los créditos requeridos para la carrera, para así obtener la fecha de fin de dicha actividad. El mismo procedimiento es utilizado para obtener la fecha de fin de la actividad con la cual el alumno alcanzó la totalidad de los créditos. Por último con las fechas calculadas anteriormente se calcula la cantidad de meses de diferencia con la fecha en la que el alumno entró a la facultad.

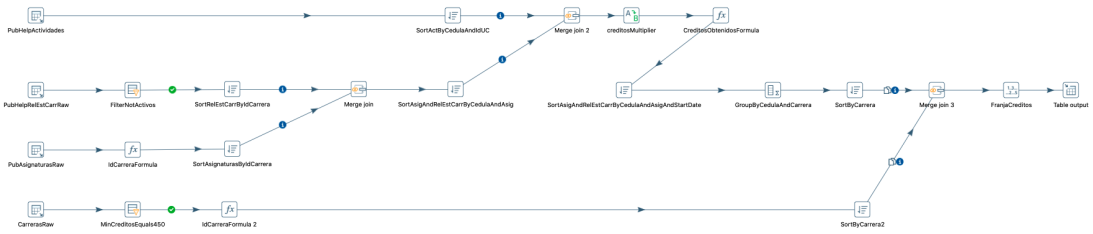
Figura 4.4.22: Segunda parte de la transformación DWFactDuracionCarrerasETL.



El paso **DWFactDistribucionEstudiantesFranjasCreditos** ejecuta la transformación `DWFactDistribucionEstudiantesFranjasCreditosETL.ktr` (figura 4.4.23) y es la encargada de popular la tabla `datawarehouse.fact_distribucion_estudiantes_franjas_creditos`. En esta transformación se calcula en qué franja de créditos se encuentra cada estudiante activo de las carreras que necesitan 450 créditos para ser completadas. Como primer paso se obtiene la suma total de créditos obtenidos por cada estudiante en cada carrera en la que está inscripto. Para ello se toma como datos de

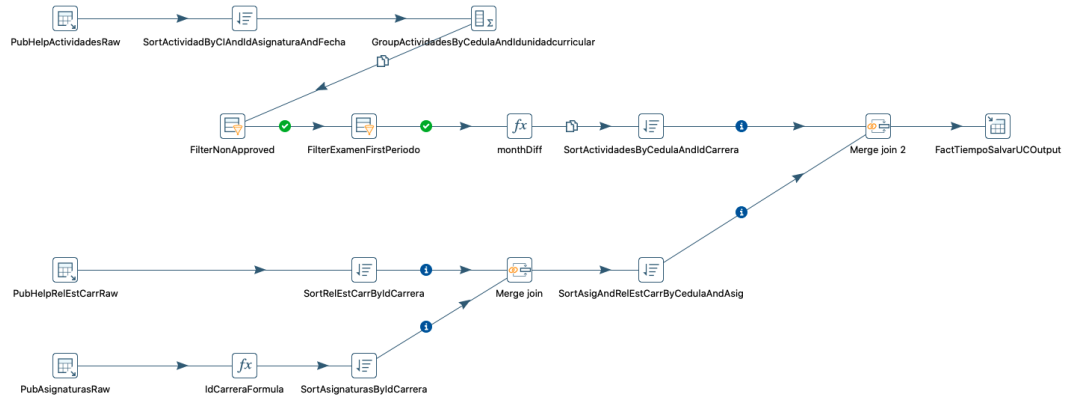
entrada las actividades de los estudiantes (tabla `public.help_actividades`), las carreras en las que el estudiante está activo (tabla `public.help_rel_est_carr` filtrando aquellos registros que el valor en el campo `status` es distinto de `active`) y la información de a qué carrera pertenece y la cantidad de créditos que otorga cada unidad curricular (tabla `public.asignaturas`) y se combinan, y luego se suman la cantidad de créditos obtenidos en cada actividad que el estudiante aprobó. Luego de calcular el total de créditos obtenidos por estudiante y carrera, se le asigna la franja de créditos correspondiente dependiendo del valor recién obtenido.

Figura 4.4.23: Transformación DWFactDistribucionEstudiantesFranjasCreditosETL.



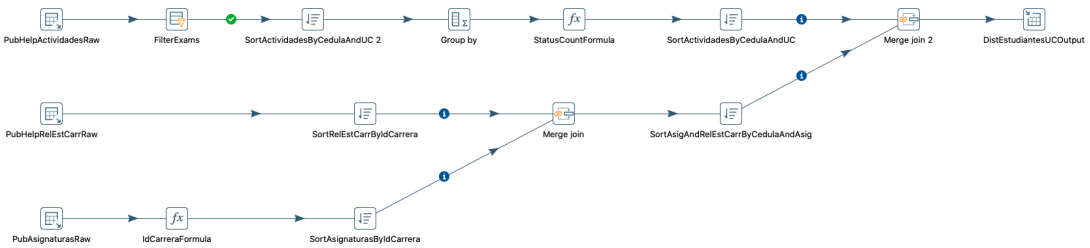
El paso **DWFactTiempoSalvarUC** ejecuta la transformación `DWFactTiempoSalvarUCETL.ktr` (figura 4.4.24) y es la encargada de cargar la tabla `datawarehouse.fact_distribucion_estudiantes_uc`. En esta transformación se calcula el estado de cada estudiante que se inscribió a un curso de una unidad curricular. Para ello se toma como datos de entrada las actividades de los estudiantes correspondiente a la participación en cursos (tabla `public.help_actividades` filtrando aquellos registros que el campo `tipoactividad` es igual a `‘‘E’’`), las carreras en las que el estudiante está inscripto (tabla `public.help_rel_est_carr`) y la información de a qué carrera pertenece cada unidad curricular (tabla `public.asignaturas`).

Figura 4.4.24: Transformación DWFactTiempoSalvarUCETL.



El paso **DWFactDistribucionEstudiantesUC** ejecuta la transformación `DWFactDistribucionEstudiantesUCETL.ktr` (figura 4.4.25) y es la encargada de popular la tabla `datawarehouse.fact_tiempo_salvar_uc`. En esta transformación se calcula la cantidad de meses que pasa desde que el estudiante cursa por primera vez una unidad curricular hasta que finalmente la aprueba. Se toman como datos de entrada las actividades de los estudiantes (tabla `public.help_actividades`), donde se agrupan las actividades que pertenecen a la misma unidad curricular para cada estudiante y se calcula la diferencia de meses entre la fecha de inicio de la primera actividad y la fecha fin de la última actividad. También se toman las carreras en las que el estudiante está inscripto (tabla `public.help_rel_est_carr`) y la información de a qué carrera pertenece cada unidad curricular (tabla `public.asignaturas`) y se agrupan a los registros de las actividades de los estudiantes.

Figura 4.4.25: Transformación DWFactDistribucionEstudiantesUCETL.



El proceso de carga fue diseñado para que soporte futuras actualizaciones de forma que la información quede consistente. Dicho proceso no maneja histórico, es decir que no se mantiene un historial del estado en que se encontraban las tablas en cargas anteriores.

Ante una nueva actualización del Data Warehouse lo que se debe realizar es sustituir el archivo de la base de datos SQLite de Bedelías y actualizar la fecha de la variable de entorno `min_activity_date_to_be_active` con la fecha correspondiente, que debería ser dos años antes de la fecha que fue obtenida la copia de la base en cuestión.

Con el Data Warehouse ya cargado con toda la información, lo que queda es poder visualizar dichos datos. En la siguiente sección se describe el proceso para desarrollar herramientas que permitan visualizarlos

4.5. Visualización

Se llama visualización de datos a los procesos de convertir información como texto, números o símbolos en un formato gráfico, con el propósito de transmitir al usuario información relevante de forma rápida e intuitiva. Dependiendo de la naturaleza de los datos y la relación entre ellos, se debe elegir el formato más adecuado para poder sacarle el mayor provecho. El valor estético de una visualización no es lo más importante, sino la claridad del mensaje que transmite.

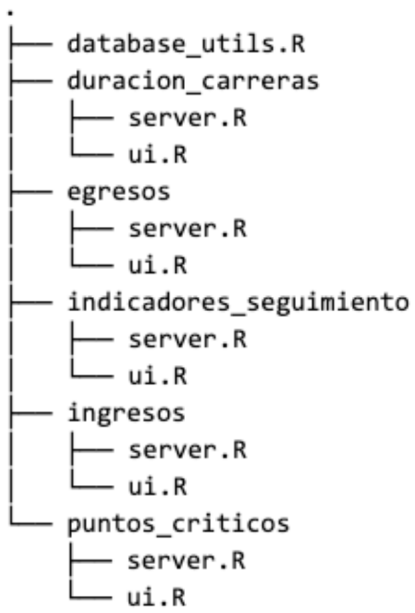
Existen múltiples herramientas que permiten crear visualizaciones a partir de datos, también conocidas como herramientas de Business Intelligence, que van desde las más básicas que solo permiten crear gráficos estáticos independientes, hasta aquellas que permiten crear tableros con múltiples elementos visuales donde el usuario puede interactuar y realizar consultas personalizadas. Otro aspecto que las diferencia es el que existen herramientas existen que son gratuitas como Pentaho BI [8] (que además es Open Source) o de pago como Microsoft Power BI [7] o Tableau [13].

Dado la gran curva de aprendizaje que requiere Pentaho BI, el costo de las licencias de las alternativas y el gran conocimiento previo de los integrantes de la UEFI del lenguaje y entorno de R, en este proyecto nos decantamos por construir una herramienta de visualización propia basada en el lenguaje R utilizando paquetes de R tales como Shiny [12], ggplot [4], DBI [2], entre otras. Shiny es un paquete de R que permite crear aplicaciones web interactivas directamente desde R, sin necesidad de poseer conocimientos de desarrollo Web. Shiny combina las potentes funcionalidades de R para manipular, analizar y representar datos, con facilidades para crear componentes web que permiten a los usuarios ver e interactuar con los componentes visuales propios de Shiny o gráficos generados con ggplot u otros paquetes. DBI es el paquete de R que nos permite conectarnos a la base de

datos del Data Warehouse. Para elaborar y ejecutar las aplicaciones Shiny antes mencionadas utilizaremos RStudio [10].

Al igual que para el proceso de ETL, en el repositorio de archivos fuentes del proyecto [24] se encuentran los archivos necesarios para poder visualizar los datos almacenados en el Data Warehouse, específicamente dentro de la carpeta `visualizacion` y siguen la siguiente estructura que se muestra en la figura 4.5.1.

Figura 4.5.1: Estructura de carpetas y archivos utilizados para la visualización.



A fin de simplificar y modularizar el trabajo se divide la visualización en múltiples módulos, tomando como hilo conductor los informes realizados por la UEFI. Los módulos son carpetas que dentro tienen una aplicación de Shiny, es decir un archivo `ui.R` y un archivo `server.R`. Los módulos de `ingresos`, `egresos`, `duracion_carreras` e `indicadores_seguimiento` contienen las aplicaciones de Shiny con los gráficos y las tablas observadas en los informes de Ingresos, Egresos, Duración de carreras e Indicadores de seguimientos respectivamente. Dentro del módulo `puntos_criticos` se encuentra la aplicación Shiny que permite visualizar los gráficos del informe de Puntos Críticos y también las gráficas de los indicadores que surgieron a partir de las reuniones con los integrantes de la UEFI, de cómo se distribuyen los estudiantes en las ediciones de los cursos y el tiempo que lleva salvar una unidad curricular. Por último, pero no menos importante, en la raíz de la carpeta `visualizacion` se encuentra el archivo `database_utils.R`. El mismo

contiene utilidades que se emplean en todos los módulos, como la creación de la conexión a la base de datos y las funciones que ayudan a armar dinámicamente las consultas sql realizadas desde los módulos.

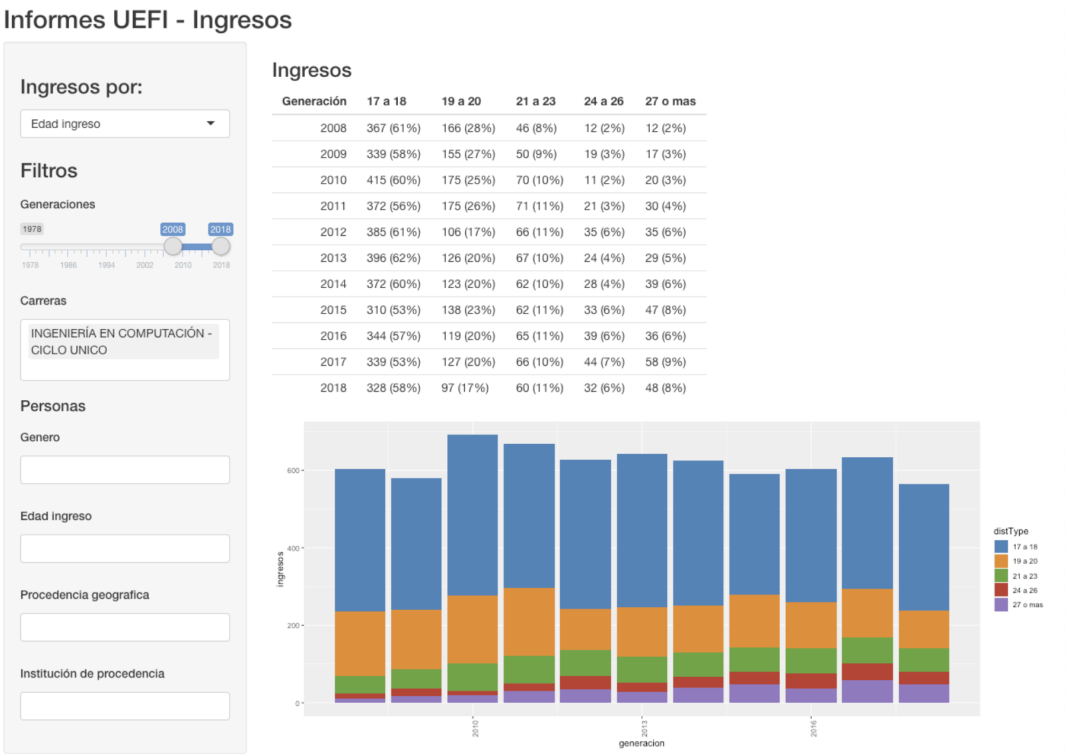
Si bien cada módulo está enfocado en una temática distinta, todos comparten cierta estructura en común. Visualmente están compuestos por dos grandes zonas, la zona donde se le permite configurar al usuario múltiples variables de configuración en una barra lateral izquierda (como se muestra en la figura 4.5.2) y una zona de visualización a la derecha de la misma donde están las tablas y los gráficos que permiten visualizar los datos. Los parámetros de configuración y los gráficos dependen fuertemente de la naturaleza de los datos que se muestran en el informe, pero como regla general los parámetros de configuración están muy ligados a los datos que se encuentran dentro de las tablas de dimensiones como por ejemplo los estudiantes, pudiendo filtrar por las distintas jerarquías de la dimensión.

Figura 4.5.2: Ejemplo de barra lateral presente en todos los módulos que permite configurar la visualización.

El módulo de ingresos (figura 4.5.3) permite visualizar las tablas con los ingresos por generación agrupado por los mismos criterios que el informe de Ingresos, es decir por género, rango de edad de ingreso, procedencia geográfica, institución de procedencia y carrera a la cual ingresó. La visualización se puede cambiar con el selector que se encuentra en la esquina superior izquierda. A su vez se pueden filtrar los ingresos seleccionando una o más carreras por medio del filtro

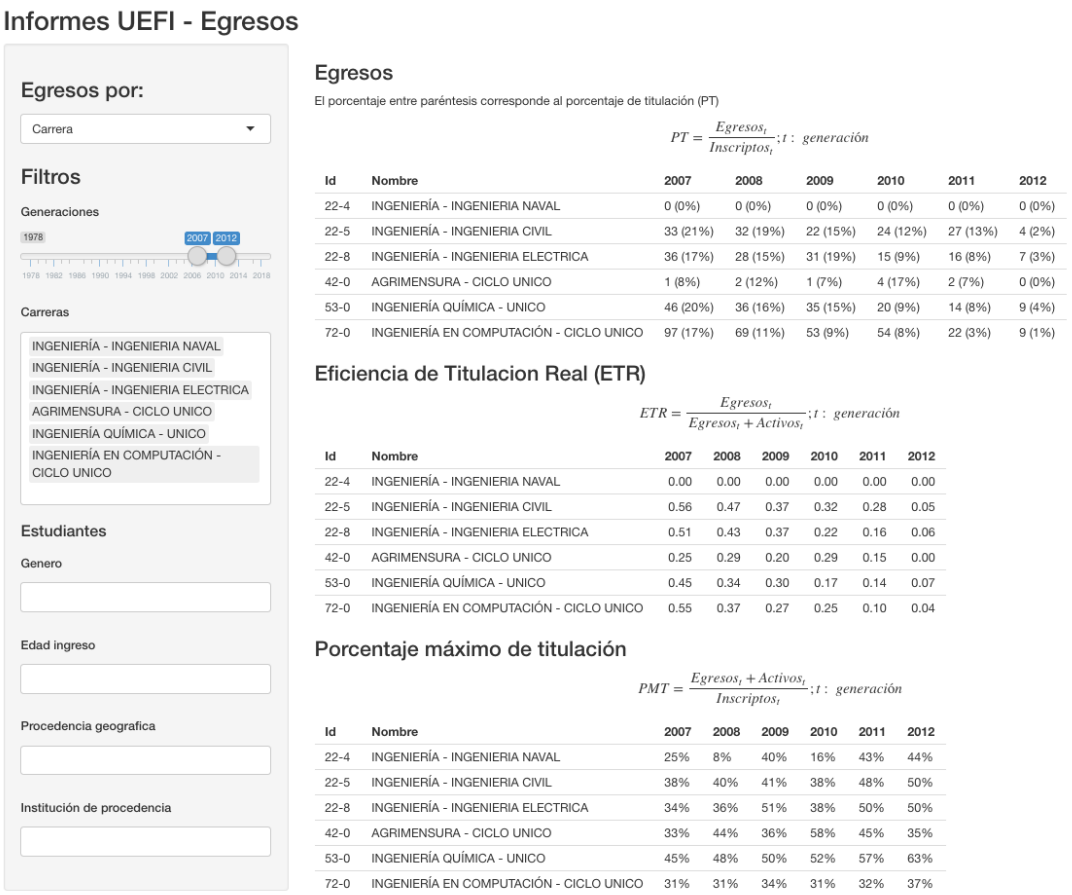
de Carreras, seleccionar cuales generaciones se quieren analizar y filtrar por las distintas características de los estudiantes. Además de las tablas que originalmente se encuentran en el informe, se agregó un gráfico de barras apiladas con la misma información que contiene la tabla, pero en un formato más fácil de visualizar.

Figura 4.5.3: Visualización del módulo de ingresos.



Para el módulo de egresos (figura 4.5.4) se puede ver la tabla de la cantidad de egresos y la Eficiencia de Titulación Real por carrera para cada generación como en el informe de Egresos. También se agrega en la tabla de egresos el porcentaje de titulación, que corresponde al total de egresos sobre el total de ingresos de dicha generación. Por otro lado también se agrega el porcentaje de titulación máximo, que corresponde a la máxima cantidad de estudiantes recibidos que podría haber si todos los que continúan activos para dicha carrera se terminaran recibiendo. Todas las tablas antes mencionadas no solo permiten agrupar los estudiantes por carreras, sino que se pueden agrupar y filtrar por mismos criterios que en el módulo de ingresos.

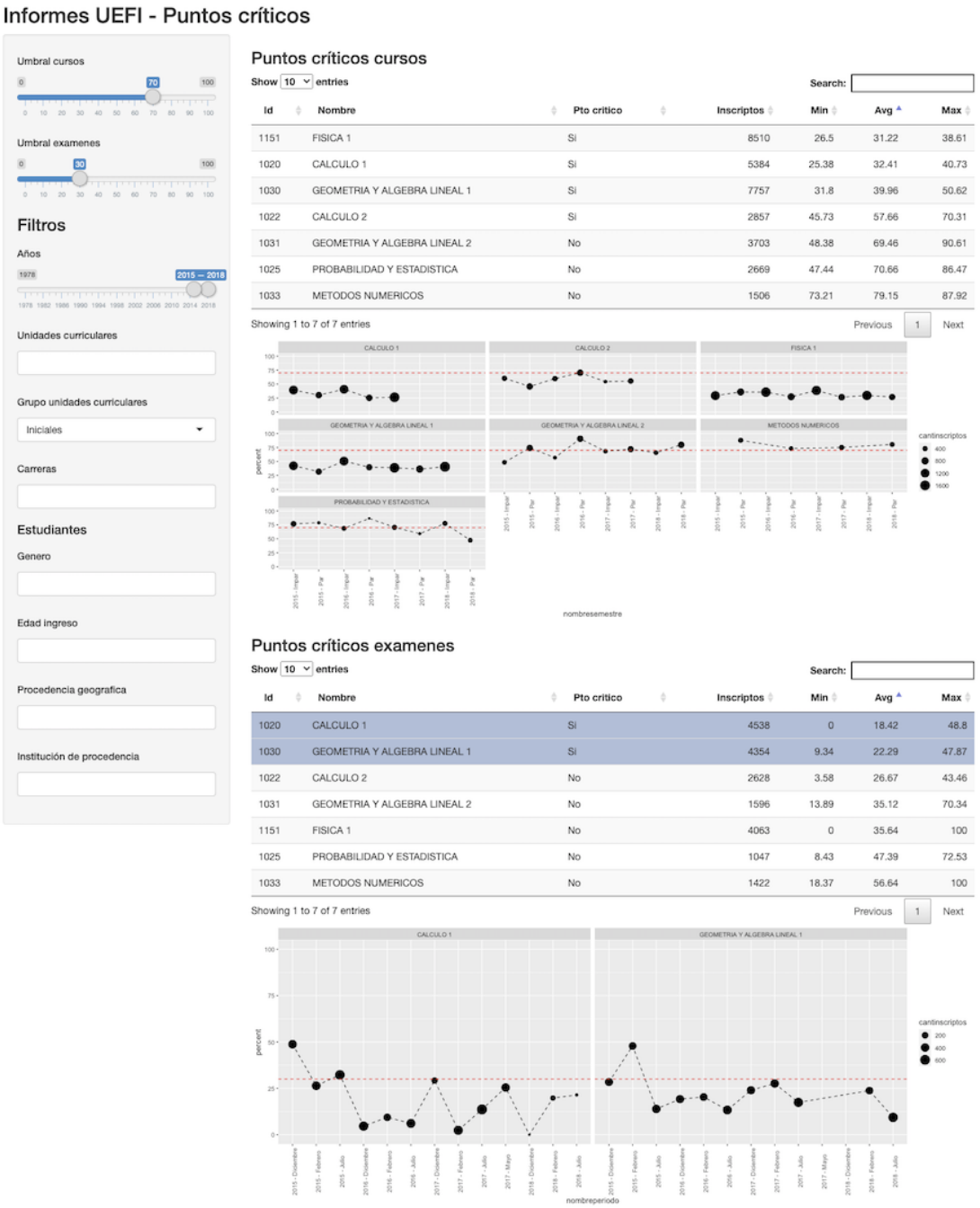
Figura 4.5.4: Visualización del módulo de egresos.



Pasando al módulo de puntos críticos se visualizan gráficos de las distintas actividades (cursos o exámenes) de las unidades curriculares y además se pueden visualizar los gráficos de la distribución de los estudiantes en las unidades curriculares según su estado y el tiempo que se demora en salvar las unidades curriculares. Los gráficos de puntos críticos están conformados de la misma forma que en el informe de Puntos Críticos (figura 4.5.5), es decir un gráfico de puntos que representa el porcentaje de aprobación por unidad curricular en cada periodo. A estos gráficos los acompañan tablas interactivas que permiten visualizar para cada unidad curricular cual es el porcentaje mínimo, promedio y máximo de aprobación, saber si es considerado como punto crítico y conocer la cantidad total de inscriptos. La interacción con dichas tablas se puede hacer mediante la búsqueda en un campo de texto libre, el ordenamiento por cualquiera de los campos antes mencionados y la navegación hacia las siguientes páginas de la tabla. Cabe destacar que estas tablas ayudan a la visualización de las gráficas, porque al realizar cualquier acción sobre la tabla va a impactar sobre las gráficas que se muestran. Otra acción que

se puede realizar sobre las tablas es seleccionar una o más de sus filas, lo que hará que se muestran las gráficas correspondientes a las unidades curriculares de las filas seleccionadas. Los parámetros de configuración para la parte de puntos críticos son la selección del umbral por el cual es considerado un curso o un examen como punto crítico, los años a analizar y la selección de qué unidades curriculares mostrar.

Figura 4.5.5: Primera parte de la visualización del módulo de puntos críticos.

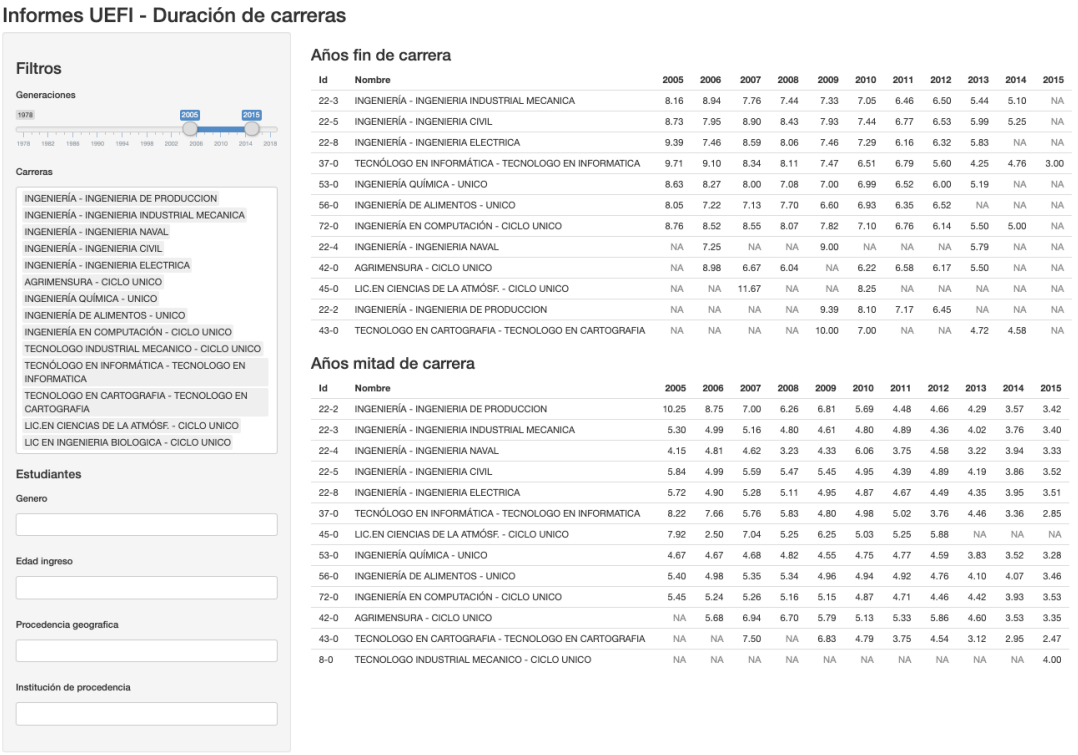


La segunda parte del módulo de puntos críticos (figura 4.5.6) está compuesta por las gráficas de barras apiladas que muestra la distribución de los estudiantes en cada edición de un curso de una unidad curricular, y el gráfico de cajas (o también conocido como de bigotes) con la distribución estadística del tiempo que lleva a los estudiantes salvar cada unidad curricular. Al igual que las gráficas de puntos críticos, estas gráficas vienen acompañadas de tablas interactivas que permiten controlar los gráficos que se muestran.

Figura 4.5.6: Segunda parte de la visualización del módulo de puntos críticos.

En el módulo de duración de carreras (figura 4.5.7) se visualizan las tablas con el promedio de tiempo que se demora en alcanzar la mitad y el fin de la carrera al igual que el informe de Duración de carreras. Los parámetros por los cuales se configura esta visualización son seleccionando el rango de generaciones y las carreras que se desean analizar. Además se puede filtrar por las distintas características de los estudiantes.

Figura 4.5.7: Visualización del módulo de duración de carreras.



Años fin de carrera

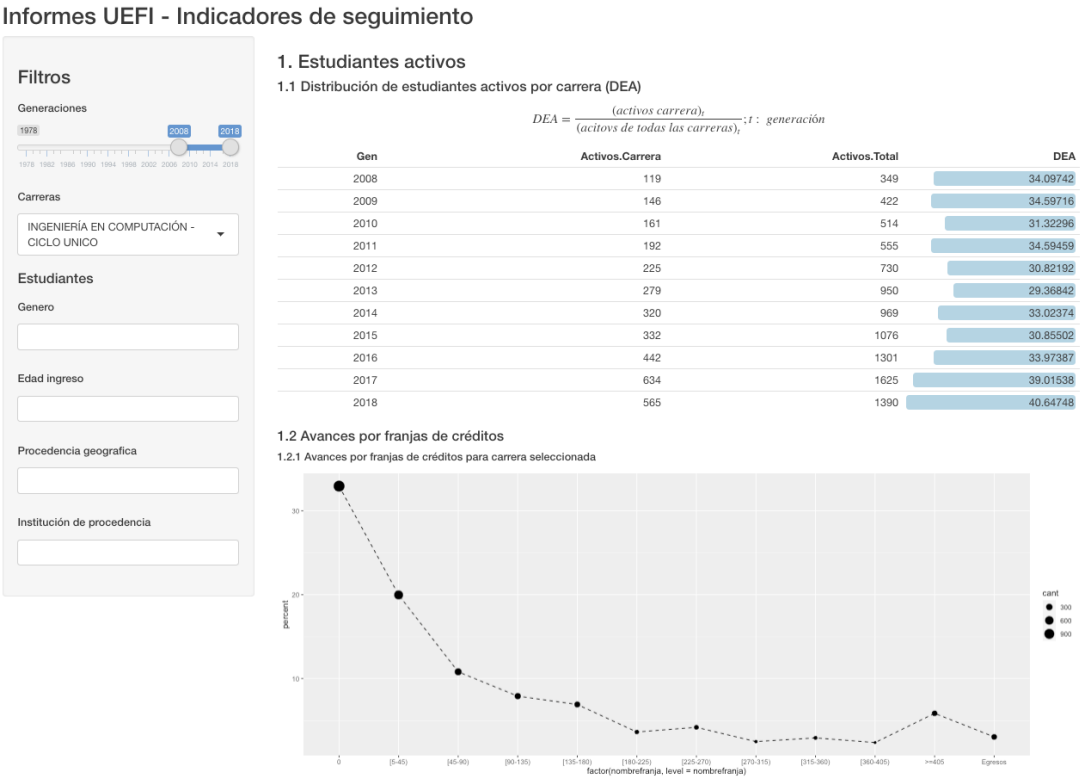
Id	Nombre	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015
22-3	INGENIERÍA - INGENIERIA INDUSTRIAL MECANICA	8.16	8.94	7.76	7.44	7.33	7.05	6.46	6.50	5.44	5.10	NA
22-5	INGENIERÍA - INGENIERIA CIVIL	8.73	7.95	8.90	8.43	7.93	7.44	6.77	6.53	5.99	5.25	NA
22-8	INGENIERÍA - INGENIERIA ELECTRICA	9.39	7.46	8.59	8.06	7.46	7.29	6.16	6.32	5.83	NA	NA
37-0	TECNÓLOGO EN INFORMÁTICA - TECNOLOGO EN INFORMATICA	9.71	9.10	8.34	8.11	7.47	6.51	6.79	5.60	4.25	4.76	3.00
53-0	INGENIERÍA QUÍMICA - UNICO	8.63	8.27	8.00	7.08	7.00	6.99	6.52	6.00	5.19	NA	NA
56-0	INGENIERÍA DE ALIMENTOS - UNICO	8.05	7.22	7.13	7.70	6.60	6.93	6.35	6.52	NA	NA	NA
72-0	INGENIERÍA EN COMPUTACIÓN - CICLO UNICO	8.76	8.52	8.55	8.07	7.82	7.10	6.76	6.14	5.50	5.00	NA
22-4	INGENIERÍA - INGENIERIA NAVAL	NA	7.25	NA	NA	9.00	NA	NA	NA	5.79	NA	NA
42-0	AGRIMENSURA - CICLO UNICO	NA	8.98	6.67	6.04	NA	6.22	6.58	6.17	5.50	NA	NA
45-0	LIC.EN CIENCIAS DE LA ATMÓSF. - CICLO UNICO	NA	NA	11.67	NA	NA	8.25	NA	NA	NA	NA	NA
22-2	INGENIERÍA - INGENIERIA DE PRODUCCION	NA	NA	NA	NA	9.39	8.10	7.17	6.45	NA	NA	NA
43-0	TECNOLOGO EN CARTOGRAFIA - TECNOLOGO EN CARTOGRAFIA	NA	NA	NA	NA	10.00	7.00	NA	NA	4.72	4.58	NA

Años mitad de carrera

Id	Nombre	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015
22-2	INGENIERÍA - INGENIERIA DE PRODUCCION	10.25	8.75	7.00	6.26	6.81	5.69	4.48	4.66	4.29	3.57	3.42
22-3	INGENIERÍA - INGENIERIA INDUSTRIAL MECANICA	5.30	4.99	5.16	4.80	4.61	4.80	4.89	4.36	4.02	3.76	3.40
22-4	INGENIERÍA - INGENIERIA NAVAL	4.15	4.81	4.62	3.23	4.33	6.06	3.75	4.58	3.22	3.94	3.33
22-5	INGENIERÍA - INGENIERIA CIVIL	5.84	4.99	5.59	5.47	5.45	4.95	4.39	4.89	4.19	3.86	3.52
22-8	INGENIERÍA - INGENIERIA ELECTRICA	5.72	4.90	5.28	5.11	4.95	4.87	4.67	4.49	4.35	3.95	3.51
37-0	TECNÓLOGO EN INFORMÁTICA - TECNOLOGO EN INFORMATICA	8.22	7.66	5.76	5.83	4.80	4.98	5.02	3.76	4.46	3.36	2.85
45-0	LIC.EN CIENCIAS DE LA ATMÓSF. - CICLO UNICO	7.92	2.50	7.04	5.25	6.25	5.03	5.25	5.88	NA	NA	NA
53-0	INGENIERÍA QUÍMICA - UNICO	4.67	4.67	4.68	4.82	4.55	4.75	4.77	4.59	3.83	3.52	3.28
56-0	INGENIERÍA DE ALIMENTOS - UNICO	5.40	4.98	5.35	5.34	4.96	4.94	4.92	4.76	4.10	4.07	3.46
72-0	INGENIERÍA EN COMPUTACIÓN - CICLO UNICO	5.45	5.24	5.26	5.16	5.15	4.87	4.71	4.46	4.42	3.93	3.53
42-0	AGRIMENSURA - CICLO UNICO	NA	5.68	6.94	6.70	5.79	5.13	5.33	5.86	4.60	3.53	3.35
43-0	TECNOLOGO EN CARTOGRAFIA - TECNOLOGO EN CARTOGRAFIA	NA	NA	7.50	NA	6.83	4.79	3.75	4.54	3.12	2.95	2.47
8-0	TECNOLOGO INDUSTRIAL MECANICO - CICLO UNICO	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	4.00

Por último el módulo de indicadores de seguimiento (figura 4.5.8) contiene todas las tablas y gráficos que se encuentran en el informe de Indicadores de Seguimiento. Los parámetros por los cuales se configura esta visualización son seleccionando el rango de generaciones y la carrera que se desea analizar. Además se puede filtrar por las distintas características de los estudiantes.

Figura 4.5.8: Visualización del módulo de indicadores de seguimiento.



Capítulo 5

Análisis y aplicaciones

Como parte del trabajo con la UEFI, se tiene como objetivo analizar la información de que dispone para hacer diagnósticos de los estudiantes, docentes y cursos para tomar decisiones. Junto con la exploración del tipo de informes que se requieren periódicamente, se evaluaron otras técnicas de análisis que permitan a la UEFI, en un futuro, ofrecer informes más complejos. Con este fin, se busca encontrar unidades curriculares (UC) que sean de gran dificultad de aprobación para los estudiantes, en qué UC se desvinculan y posibles patrones que expliquen indicios de la desvinculación con la carrera.

En el presente capítulo se aplicarán distintas técnicas para el análisis de la información que disponemos. Para todos los estudios a continuación, tomamos como fecha valor para la desvinculación como el 1ro de abril del 2019, es decir: para calcular la desvinculación, se tomará la distancia de tiempo entre la fecha del último curso y el 1ero de Abril del 2019. Esto se debe a que se nos dispuso de una base estática y a medida que avanzaba el proyecto se podían encontrar diferencias con estudios iniciales.

Para facilitar el manejo de fechas y operaciones relacionadas con ubicación geográfica se tomó la decisión de migrar la base de datos de bedelías de SQLite [21] a Postgres [19] con su extensión de Posgis[9]. Estas migraciones fueron realizadas con el software de Data Integration de Pentaho “Kettle” [18].

5.1. Definiciones conceptuales y análisis de los datos

En esta sección, en primera instancia, el foco del análisis se encuentra en las características de la población que ingresa a la carrera Ingeniería en Computación, como puede ser la cantidad y evolución de estudiantes que ingresan, la proporción por género en la carrera y en qué estado se encuentran los estudiantes.

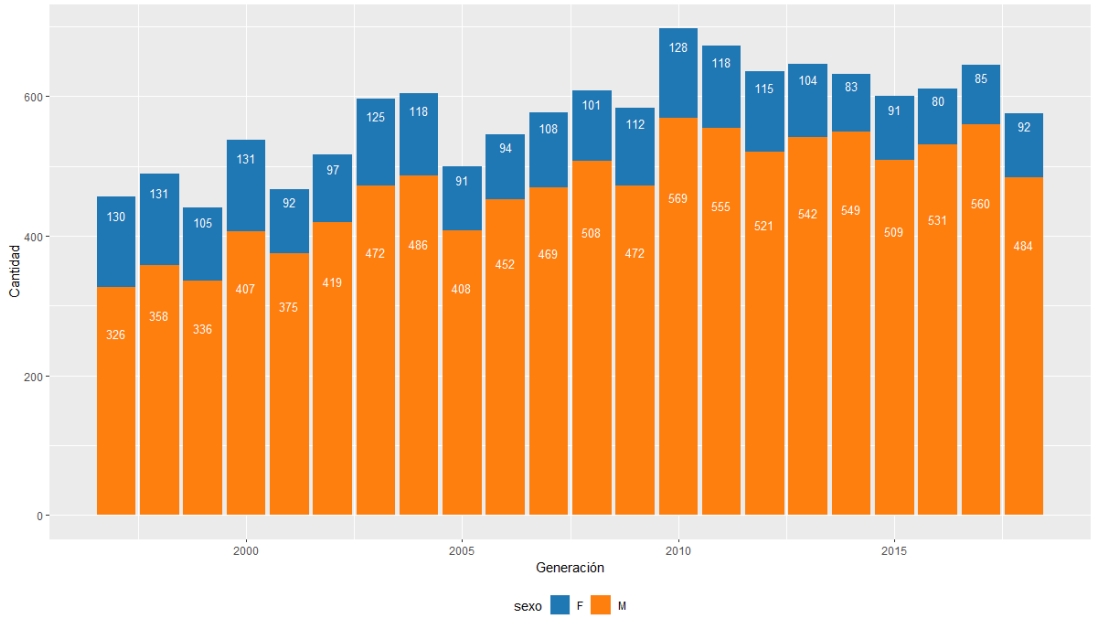
Con el fin de tener un conocimiento previo de la carrera, se presenta en esta sección un análisis detallado mediante métodos clásicos descriptivos. Esto tiene un componente importante dentro del proyecto, dado que nos permite realizar un acercamiento a los datos brindados y conocer sus distribuciones y el estado de los estudiantes según generación y los tiempos de egreso y desvinculación. No obstante, existen algunas restricciones como dijimos en el capítulo 3, en lo que respecta a las fuentes de información, que no nos permiten hacer un análisis completo sobre el perfil social de los estudiantes.

Para comenzar, es interesante ver cuántos estudiantes ingresan a la carrera y de qué género son. En la figura 5.1.1 se puede ver que desde 1997 hasta la fecha, los ingresos han tenido un comportamiento sostenido con tendencia al alza. El máximo número de ingresos se dio en el año 2010, donde ingresaron 715 estudiantes. También en el mismo gráfico es posible analizar que la cantidad de estudiantes del género femenino se mantiene prácticamente constante en el tiempo, a pesar de algunas iniciativas en los últimos años para aumentar la proporción.

Para cada generación es posible categorizar 3 estados: uno es trivial, un estudiante que se encuentra recibido, es decir ha llegado a al menos 450 créditos, cumplido con los créditos por unidades curriculares básicas y aprobado las asignaturas obligatorias. Luego existen los estudiantes que se encuentran en un estado "desvinculado" en donde en los últimos dos años no han tenido inscripciones a cursos o exámenes, es decir ninguna actividad académica en la carrera. Finalmente, es posible identificar un tercer estado como el complemento de los dos anteriores, que son aquellos estudiantes que aún se encuentran cursando la carrera.

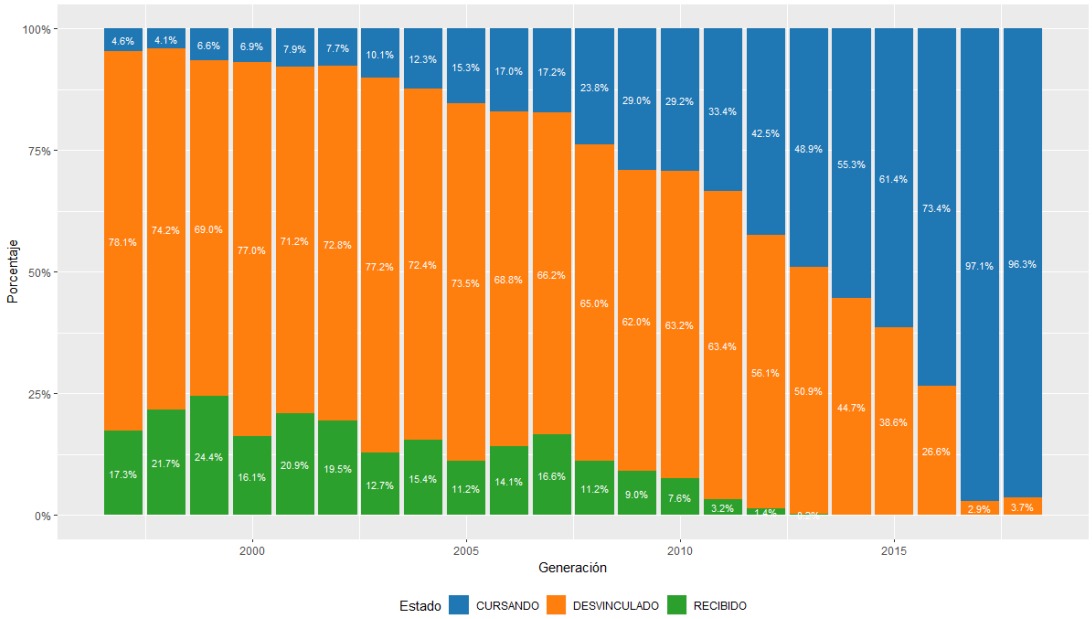
Comenzando por los recibidos, es interesante conocer el porcentaje de egreso para cada generación. En la figura 5.1.2 podemos ver que, desde 1997, el egreso no ha llegado al 30 %. Algo que además impresiona, es que en las generaciones 2015 y 2016, el 26 % y 38 % de los estudiantes ya se han desvinculado de sus actividades académicas respectivamente. Inmediatamente, esto trae aparejadas preguntas como: ¿Cuánto demoran en recibirse los estudiantes? o, los que abandonan, ¿en qué etapa

Figura 5.1.1: Cantidad de estudiantes según sexo que realizaron una inscripción a la carrera ingeniería en computación. plan 97, periodo 1997-2019.



lo hacen?. En la tabla 5.1.1 se puede visualizar cuánto demoran los estudiantes en recibirse, en donde la media oscila entre los 7 y los 9 años dependiendo la generación. Además, existen casos atípicos en los que algunos estudiantes terminan la carrera en 4 años (un año menos que la currícula sugerida) y en el otro extremo algunos estudiantes que han tardado 16 o más años en recibirse, más de una década respecto a lo estimado curricularmente.

Figura 5.1.2: Porcentaje de estudiantes según estado por generación de ingreso a la carrera ingeniería en computación plan 97, periodo 1997-2019 .

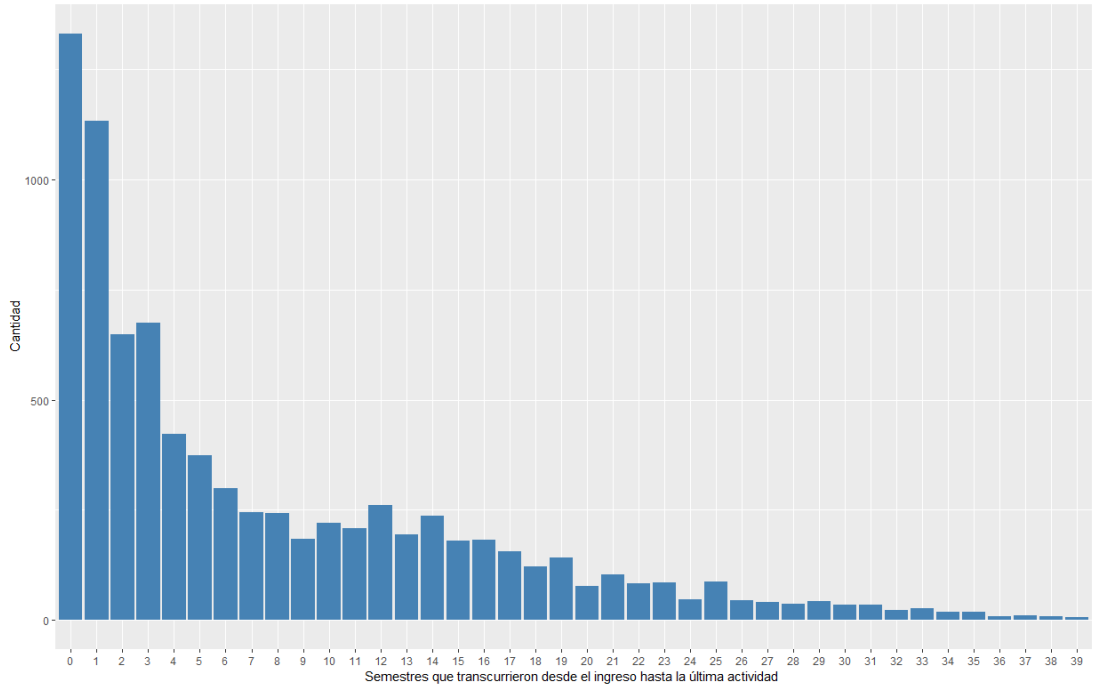


Cuadro 5.1.1: Cantidad de estudiantes que se reciben de acuerdo a su generación y el tiempo en que demoraron en terminar la carrera ingeniería en computación plan 97, periodo 1997-2019 .

Gen	4	5	6	7	8	9	10	11	12	13	14	15	16+	Total gen.
1997		9	8	11	9	9	6	4	6	3	5	4	5	79
1998		6	21	17	10	9	6	7	5	13	4	3	5	106
1999		6	18	16	13	13	9	7	8	3	3	3	9	108
2000	2	10	9	5	14	11	13	10	2	4	1	2	4	87
2001	1	6	15	15	16	13	8	5	9	4	2	4		98
2002		4	10	19	14	10	16	15	2	5	6			101
2003	1	5	14	14	12	9	4	6	6	3	3			77
2004		4	16	18	11	15	11	8	8	3				94
2005		1	9	14	14	6	6	6	1					57
2006		6	17	16	8	9	11	11						78
2007		10	16	25	16	16	14							97
2008		14	8	20	19	8								69
2009		6	13	21	13									53
2010		13	25	16										54
2011		9	13											22
2012	1	8												9
2013	1													1

Poniendo el foco en los que se desvinculan de la carrera, llama la atención que cerca de 2500 estudiantes de todas las generaciones (ver figura 5.1.3), lo cual representa casi el 30 % de los estudiantes que alguna vez cursaron la carrera, optaron por no continuar a un año de su inscripción. Sin embargo, también existen casos que no finalizan después de casi 20 años cursando, posiblemente con muchas intermitencias.

Figura 5.1.3: Cantidad de estudiantes que abandonaron según su tiempo de actividad en semestres, para la carrera ingeniería en computación plan 97, periodo 1997-2019 .



5.2. Factores de incidencia para la desvinculación

Si bien es importante conocer como se comportan las generaciones y la carrera en sí, así como pasa en el resto de la educación formal, interesa saber el porqué de la desvinculación de los estudiantes, más aún en los porcentajes que se mostraron en la sección anterior. En la presente sección recorreremos algunos factores de riesgo o mejor nombrados, de incidencia en la desvinculación universitaria.

5.2.1. Extraedad

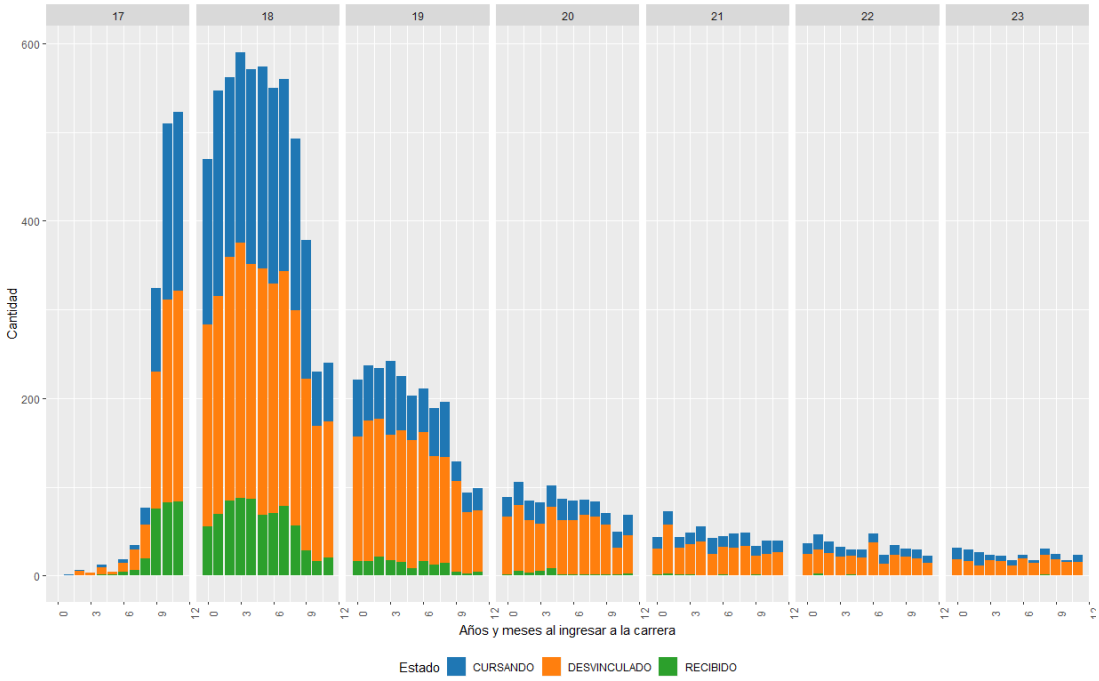
Algunos estudios [28], mencionan que los estudiantes con extra-edad (no corresponde su edad cronológica con el nivel que cursan) son más propensos

a desvincularse del sistema educativo y especialmente en la educación media. De por sí, es atrayente para este proyecto verificar si esto también ocurre en la Facultad de Ingeniería.

En Uruguay, los estudiantes pueden ingresar a primero de primaria si tienen cumplidos los 6 años al 30 de abril del año lectivo, esta fecha se traslada a todo el sistema educativo. En nuestro caso, tomaremos como fecha de corte el 1ro de Marzo de cada generación y observaremos con qué edad ingresan a la carrera terciaria en estudio. Es decir, que un estudiante se encuentra en una situación de extraedad, si al ingresar a su primer día de clases cuenta con más de 18 años y 9 meses desde su nacimiento.

Para este enfoque, visualizando la figura 5.2.1, se ve que no solo los estudiantes con mayor edad cronológica (los mayores) tienen menor probabilidad de recibirse, sino que aquellos que son más jóvenes (17 años), en proporción, se reciben en mayor cantidad. Sin dudas es algo para seguir profundizando si se tienen datos de las trayectorias de los estudiantes de todo el sistema educativo uruguayo.

Figura 5.2.1: Cantidad de estudiantes de acuerdo a su edad cronológica al ingreso por estado para la carrera ingeniería en computación plan 97, periodo 1997-2019 .



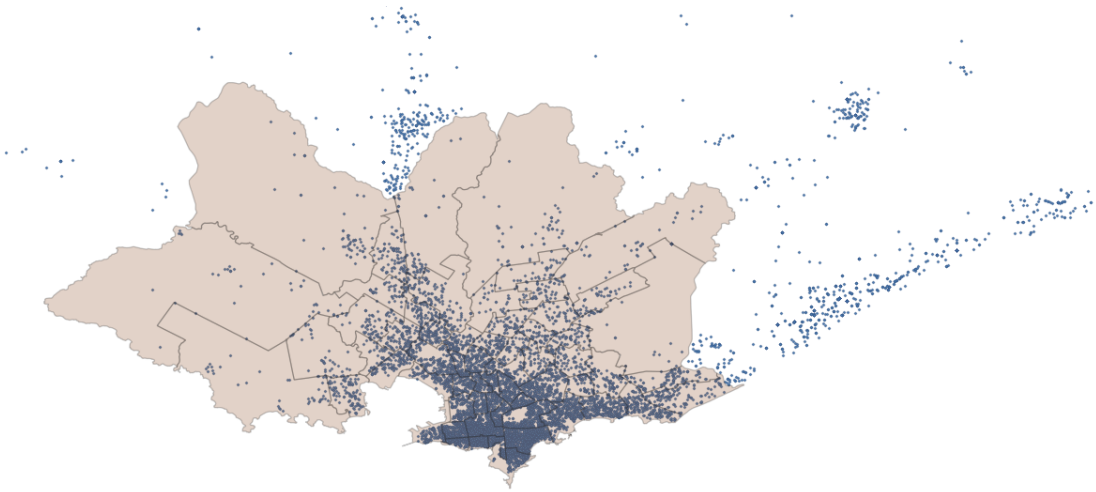
5.2.2. El estrato social del estudiante

Una hipótesis o creencia dentro del grupo al comenzar el proyecto, era que la distancia desde el domicilio del estudiante a la facultad de ingeniería puede tener una incidencia en el desempeño o ser un posible predictor a la desvinculación de los estudiantes. Dentro de la base de datos que nos facilitó la UEFI, se encontraba el campo de dirección del estudiante al momento de la inscripción, este campo, mediante la utilización de la API de Google Maps [3], puede llegar a transformarse en un par de coordenadas geográficas. Previo a impactar sobre la interfaz que dispone Google, fue necesario depurar todas las direcciones, eliminando así anotaciones auxiliares que realiza el funcionario de bedelías en el momento de la inscripción, como son el apartamento, celulares y similares.

Mediante un script en Python [15], fue posible georeferenciar 10018 direcciones, 1993 fueron catalogadas como no confiables y 545 no fueron encontradas.

Posteriormente, con el fin de representar en un mapa los puntos, se descargó desde el sitio del INE la capa de barrios de Montevideo y se representó con QGIS [16] la capa de los domicilios de los estudiantes. Allí se puede apreciar que los estudiantes que concurren a Ing. en Computación se nuclean en la región centro-sur de Montevideo.

Figura 5.2.2: Mapa de Montevideo y los domicilios de los estudiantes (ingeniería en computación plan 97, periodo 1997-2019) geo-referenciados .

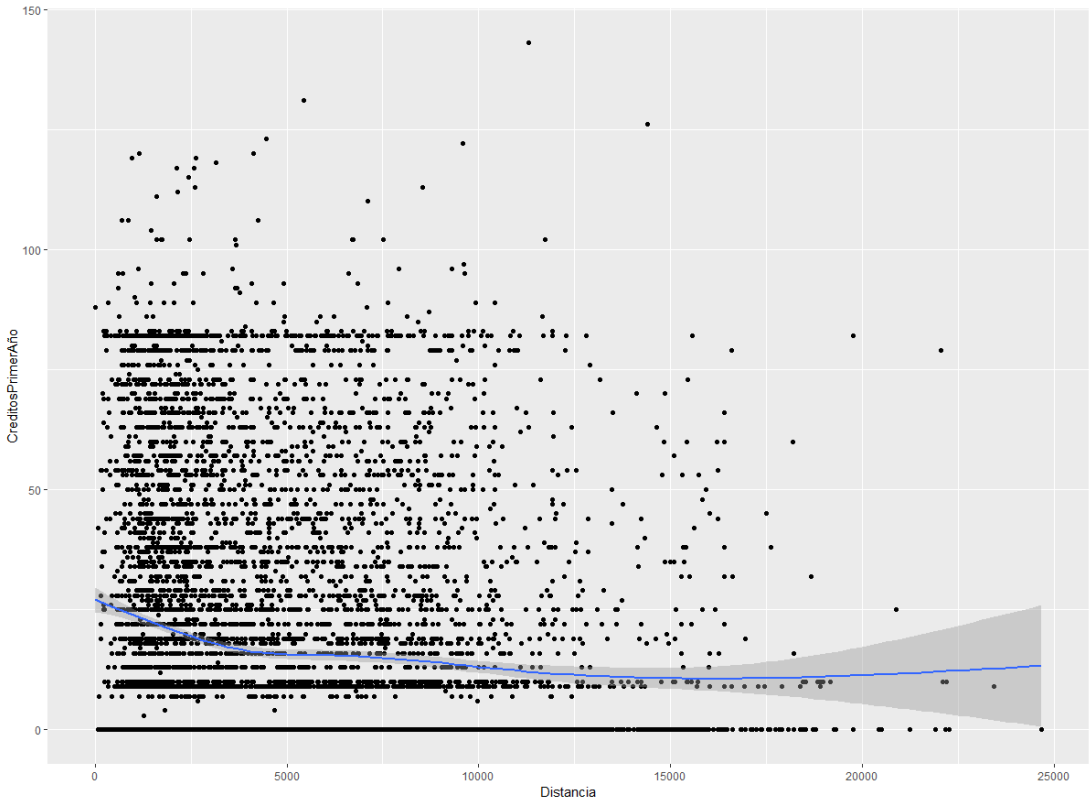


También se puede ver en el mapa que existen estudiantes que vienen desde una distancia muy grande, peor aún si tomáramos el mapa de todo Uruguay. Esto es porque algunos de ellos son del interior del país y, al momento de la inscripción, declaran su domicilio de origen por no tener un domicilio fijo más cercano. Es por

ello que para este estudio se descartaron direcciones superiores a 25 kms desde la Facultad de Ingeniería

Luego, mediante las funciones que dispone la extensión GIS de Postgres, calculamos la distancia geodésica (que es el camino más corto entre dos puntos en una superficie curva, como la Tierra). Una vez obtenida, analizamos si la distancia donde vive el estudiantes tienen alguna relación con su desempeño. Para esto, vamos a definir este indicador como la sumatoria de créditos en un plazo menor al año del ingreso de estudiante a la carrera, dado que existen estudiantes cuya trayectoria educativa tiene una larga duración y es altamente probable que se muden de domicilio.

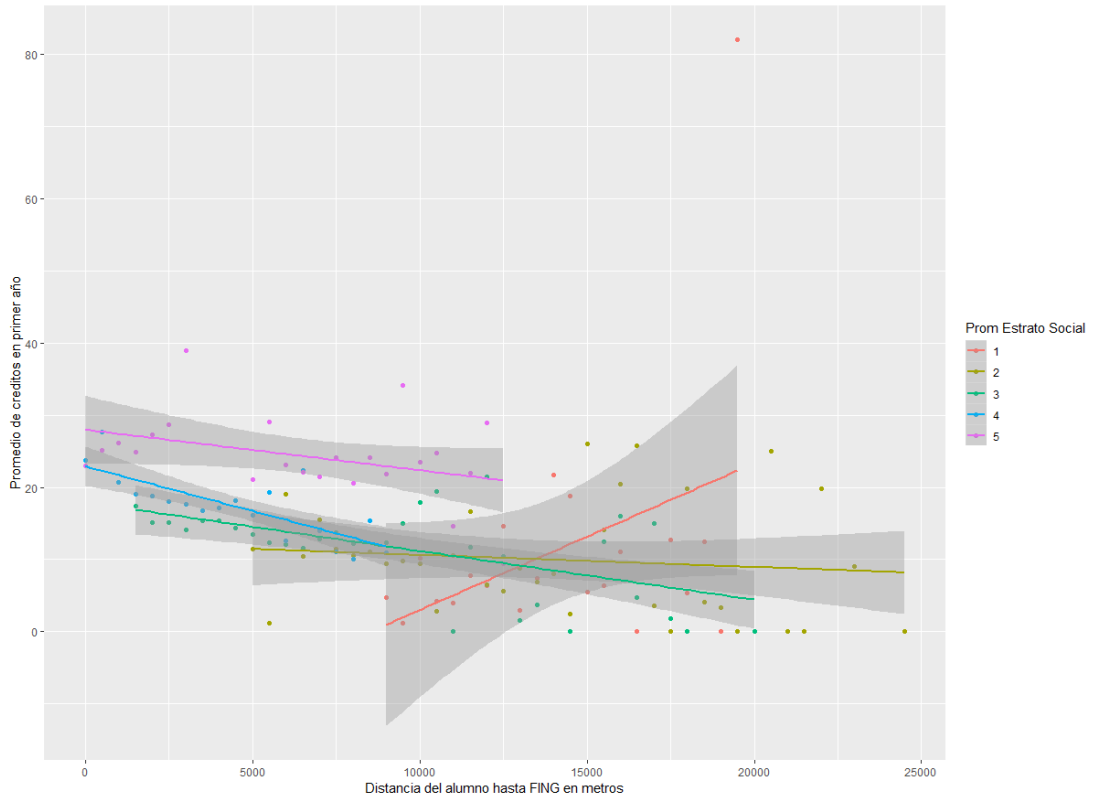
Figura 5.2.3: Créditos obtenidos en los primer año según la distancia (cada 500 metros) del estudiante (ingeniería en computación plan 97, periodo 1997-2019) y la facultad(FING) y la media por distancia.



Visualizando la figura 5.2.3, al parecer existiría alguna relación en los primeros 5000 metros. Sin embargo, Montevideo es una ciudad segmentada geográficamente a nivel socioeconómico [46] y considerando que Facultad de Ingeniería se encuentra en un lugar de alto poder adquisitivo, puede existir un sesgo en la dimensión distancia y su relación con los créditos en el primer año. Para analizar este efecto recurrimos a la encuesta continua de hogares del INE e hicimos un promedio del

estrato social del barrio donde se encuentra el domicilio del estudiante y, gracias a la segmentación que mencionamos anteriormente, analizamos nuevamente la distancia en relación a los créditos, pero agregando también la dimensión del promedio del estrato social del barrio.

Figura 5.2.4: Media de créditos obtenidos en los primer año (ingeniería en computación plan 97, periodo 1997-2019) según la distancia (cada 500 metros) del estudiante por media del estrato social del INE.

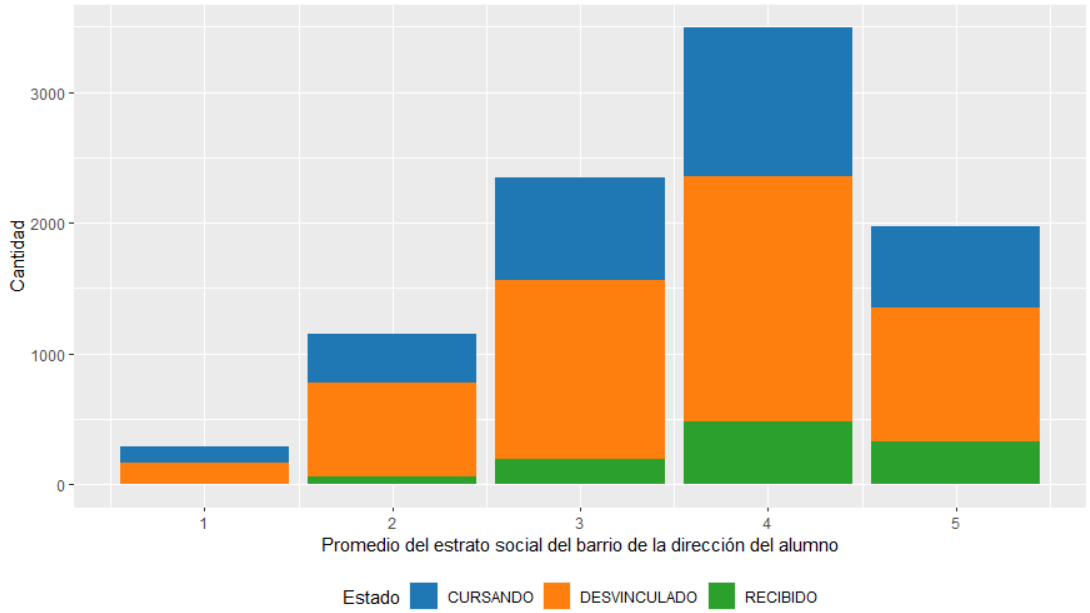


Una vez procesados (Ver Fig. 5.2.4), es posible visualizar que la distancia al centro de estudios no juega un rol protagónico en los primeros años del desempeño estudiantil de los estudiantes, sino más bien el mismo está más relacionado al estrato social del hogar y en este caso al del barrio, en donde el quintil más alto ronda entre los 20 y 30 créditos en el primer año, mientras que en el quintil más bajo se puede apreciar un gran error, esto invita a pensar sobre la poca cantidad de estudiantes de estrato social bajo que acceden y en qué condiciones.

Adicionalmente, es posible ver cuantos estudiantes se reciben en función del estrato social adjudicado a estos. Como se visualiza en la figura 5.2.5, en proporción aquellos estudiantes que provienen de mayores estratos sociales se reciben en mayor proporción que los de estratos bajos. Si bien, es algo que no sorprende,

es un posible factor de incidencia en la desvinculación para armar un modelo de alerta temprana de desvinculación.

Figura 5.2.5: Cantidad de estudiantes inscriptos (ingeniería en computación plan 97, periodo 1997-2019) según estado y estrato social.

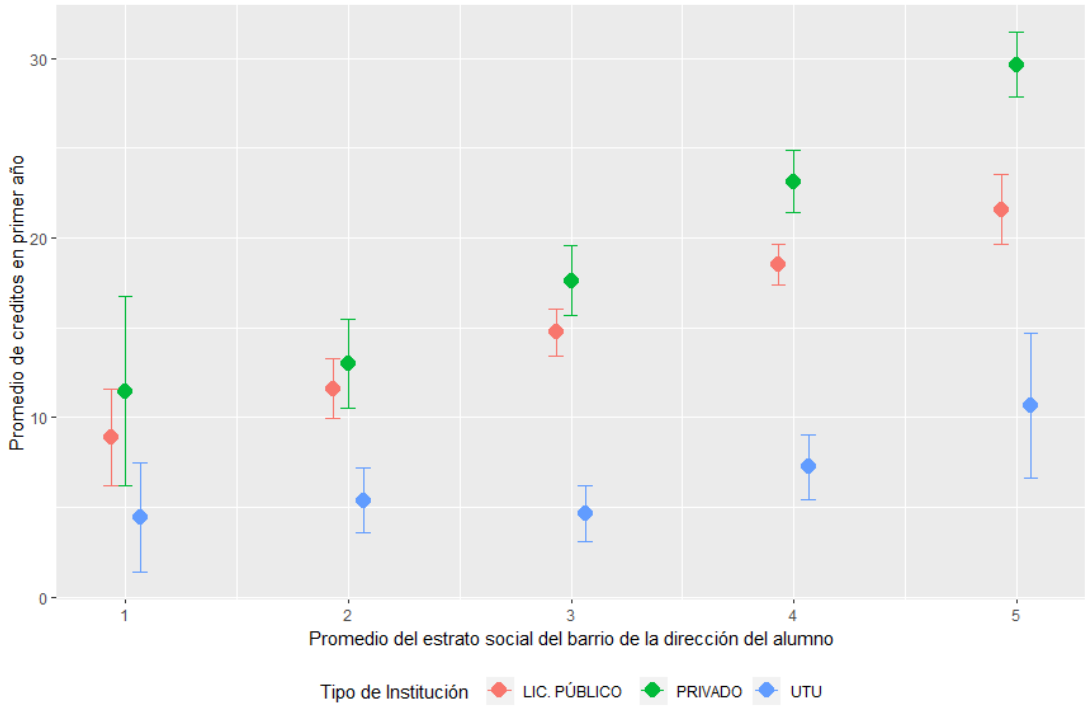


Es cierto que este indicador tiene sus deficiencias, dado que al redondear el promedio del estrato social, existen estudiantes que cambian de estrato social, por ejemplo un estudiante del quintil más alto, si pertenece a un barrio con alta varianza, probablemente va a considerado en el quintil 4. No obstante, las proporciones no cambiarían demasiado y es una aceptable aproximación hacia un indicador a realizar dentro de la UEFI.

5.2.3. Trayectorias educativas previas y universitarias iniciales

La educación no se debe de ver como una fotografía, estática, sin antecedentes, sino como un proceso o una secuencia de sucesos que afectan, positiva o negativamente en mayor o menor medida a los sujetos que la transcurren. Es por ello, que en este apartado se intenta conceptualizar algunas relaciones de la actividad previa del estudiante con su vida universitaria.

Figura 5.2.6: Créditos obtenidos en los primer año según promedio de estrato del estudiante (ingeniería en computación plan 97, periodo 1997-2019) y tipo de institución.



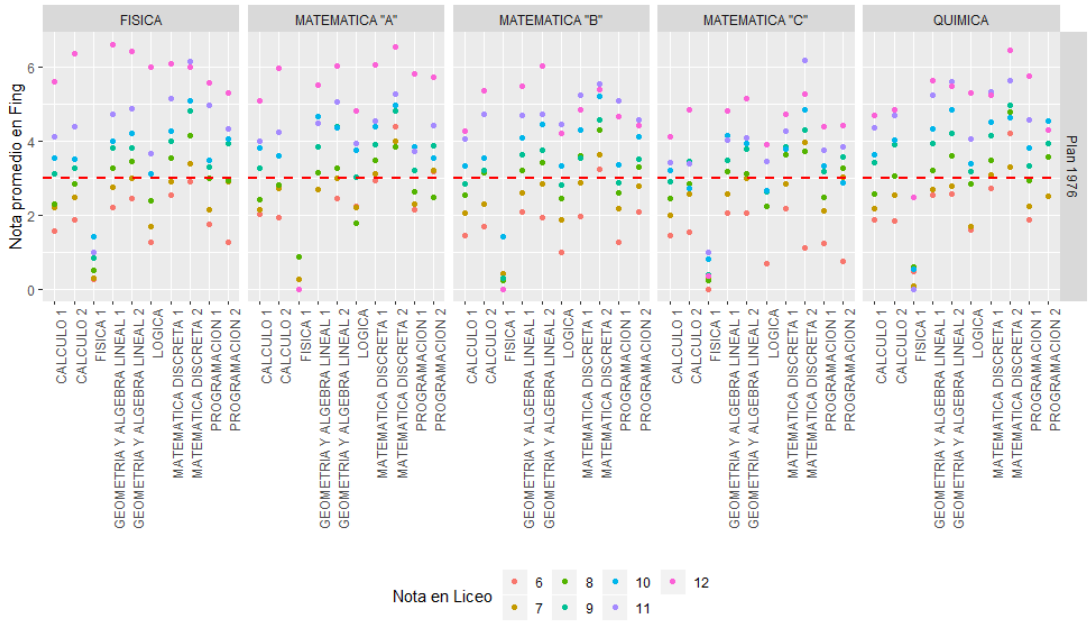
En función de esto, para un primer acercamiento que haremos, necesitamos saber como les va a los estudiantes en función de la institución educativa de la cual provienen. Como se puede apreciar en la figura 5.2.6 existe una importante incidencia en el tipo de institución de la cual proviene el estudiante.

Para aquellos estudiantes que tuvieron una educación secundaria en una institución privada tienen sensiblemente mejores rendimientos que aquellos que lo hicieron en una institución pública y dentro de estos se destacan aquellos que fueron a un liceo por encima de aquellos que concurrieron a una escuela técnica. Cabe destacar que esta situación no solo se da en todos los estratos sociales que calculamos, sino que a mayor nivel socioeconómico del barrio, mayor es la diferencia en el rendimiento estudiantil en el primer año universitario entre las categorías de institución privada, liceo público y UTU. Este no es un fenómeno nuevo, sino que ya se empieza a evidenciar en tercero de educación media, como se muestra en el informe de pruebas Aristas del Ineed [6].

Otra de las intrigas que se tenía como equipo al comienzo del proyecto, eran si los logros mínimos de aprobación en educación media (aprobar con 6) eran suficientes para el ingreso a la carrera o si existía alguna relación entre la nota de aprobación

de sexto y los resultados de aprobación en los primeros dos años de Facultad.

Figura 5.2.7: Nota promedio en las primeras asignaturas de la carrera (ingeniería en computación plan 97, periodo 1997-2019), según nota de aprobación en las UC del liceo del plan 76 de los datos brindados por ANEP.



Según se muestra en la figura 5.2.7, en donde se graficó el promedio de nota de los estudiantes en facultad según el promedio de nota que obtuvieron en UC liceales para el plan 76, no solo es fácil de visualizar que existe una relación directa entre las notas de facultad y liceo sino además que aquellos estudiantes que no obtuvieron notas por encima de 8 o 9 en promedio no lograron la suficiencia en la mayoría de las asignaturas de la carrera.

5.3. Aplicación de minería de datos

Como hablamos en el capítulo 2, la analítica del aprendizaje, está altamente relacionada con la minería de datos en relación a los datos del estudiante, el centro educativo o los docentes. En donde mediante la aplicación de estos métodos y algoritmos, es posible emitir alertas tempranas o modelar las trayectorias (uno de los objetivos de este proyecto) para acompañar los aprendizajes de los estudiantes.

En función de los datos que contamos, descritos en el capítulo 3, nos planteamos el siguiente objetivo: analizar si mediante técnicas de aprendizaje automático, es posible predecir la desvinculación de los estudiantes en la carrera ingeniería

en computación. Para ello, realizamos un juego de datos con algunas de las variables que se describieron en la sección anterior y verificar si pueden explicar el fenómeno de la desvinculación, en consecuencia se tomaron las siguientes variables "predictoras":

- Tipo de institución proveniente.
- Sexo del estudiante.
- Estrato social del estudiante, asignado a este en función de la dirección declarada al momento de la inscripción.
- Edad al ingreso en meses al primero de marzo del año de la generación.
- Cantidad de créditos que obtuvo ese estudiante en el primer semestre de su ingreso a la facultad.

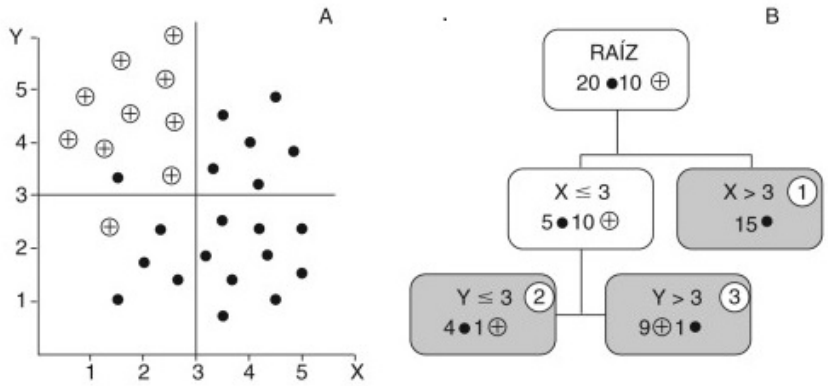
Como variable a predecir utilizaremos el estado del estudiante, en donde solo utilizaremos los estados de "Desvinculado" y "Recibido". Motiva esta decisión que los que se encuentran en estado "Cursando" son una porción muy pequeña y puede verse como un estado intermedio a los otros dos.

Dado que la mayoría de nuestras variables son numéricas y categóricas solo algunos modelos de minería de datos son aplicables, para este proyecto probaremos cuatro de ellos, estos son:

CART(Classification And Regression Trees) [40]

Los árboles de clasificación y regresión son dos métodos de aprendizaje automático para construir modelos de predicción a partir de datos. Los modelos se obtienen mediante particiones recursivas del espacio de datos y ajustando un modelo de predicción dentro de cada partición. Cada partición se puede representar gráficamente como un árbol de decisión. Cada hoja representa una etiqueta de clase, mientras que las ramas representan las conjunciones de características que conducen a esas etiquetas de clase. Tiene como ventajas que es muy fácil de visualizar y entender su ejecución sin embargo es un algoritmo muy propenso a sobreajuste.

Figura 5.3.1: Ejemplo de una ejecución de ML por CART.

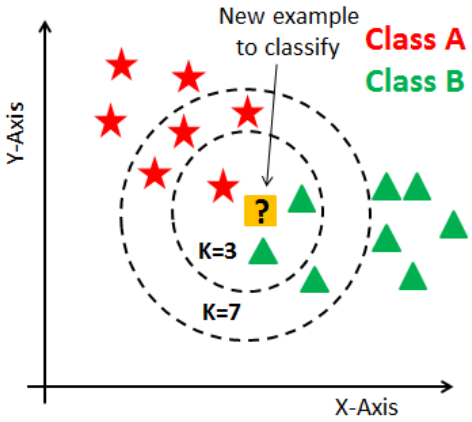


kNN(k-nearest neighbors)

El algoritmo de vecinos más cercanos a k (k-NN) es un método para clasificación y regresión. En ambos casos, la entrada consiste en los k ejemplos de entrenamiento más cercanos en el espacio de características. El resultado depende de si k-NN se usa para clasificación o regresión:

- En la clasificación k-NN. Un objeto se clasifica por un voto de pluralidad de sus vecinos, y el objeto se asigna a la clase más común entre sus k vecinos más cercanos (k es un número entero positivo, generalmente pequeño). Si k=1, entonces el objeto simplemente se asigna a la clase de ese vecino más cercano.
- En la regresión k-NN, la salida es el valor de la propiedad del objeto. Este valor es el promedio de los valores de k vecinos más cercanos.

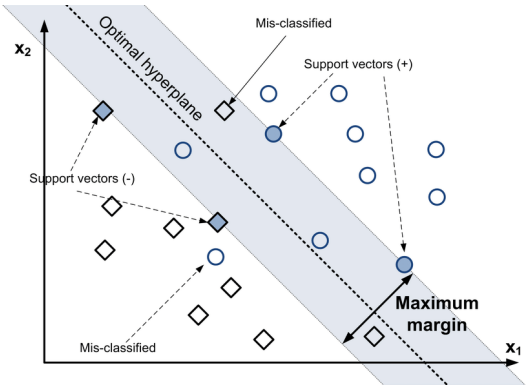
Figura 5.3.2: Ejemplo de una ejecución de ML por K-NN para K=3 y K=7.



SVM(Support Vector Machines)[33]

Un algoritmo SVM también es un método de clasificación, el cual se basa en la separación de clases mediante el trazado de vectores o formalmente hiperplanos. Se logra una buena separación mediante el hiper-plano que tiene la mayor distancia a dos puntos de dos clases distintas, ya que en general cuanto mayor es el margen, menor es el error de generalización del clasificador.

Figura 5.3.3: Ejemplo de una ejecución de ML por SVM.



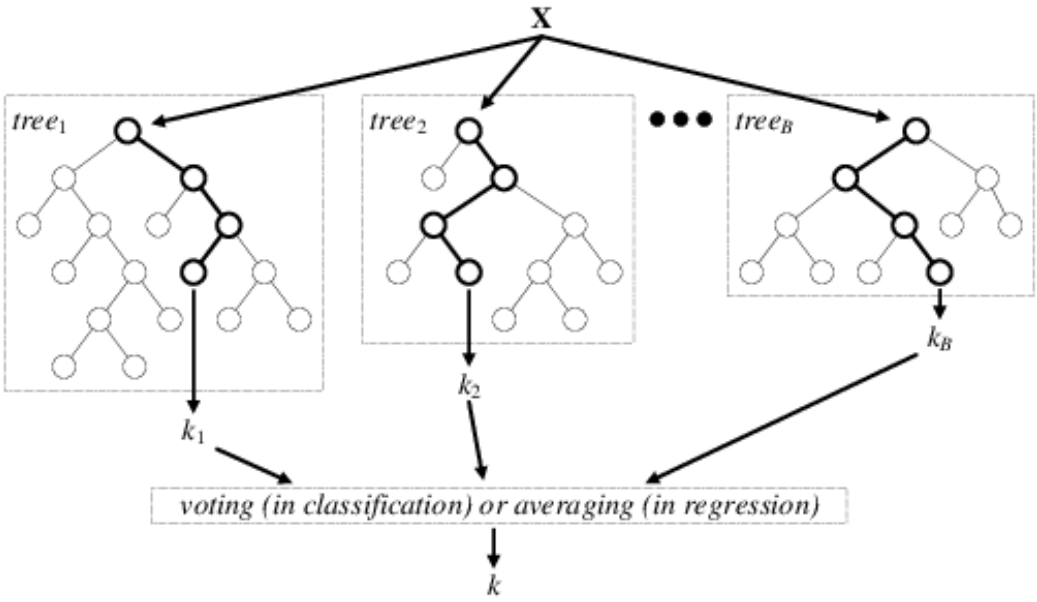
Random Forest[27]

Al igual que en CART, Random Forest utliza árboles para la clasificación, sin embargo no utiliza uno sino muchos, de ahí viene el término Forest (Bosque en inglés). Para clasificar un nuevo objeto a partir de un vector de entrada, RF ejecuta el vector de entrada sobre cada uno de los árboles en el bosque. Cada árbol da una clasificación y realiza un "voto" para esa clase. Luego, el bosque elige la clasificación que tiene más votos (sobre todos los árboles en el bosque).

Cada árbol se crece de la siguiente manera:

- Si el número de casos en el conjunto de entrenamiento es N, se hace un muestreo de N casos al azar (con reemplazo) de los datos originales. Esta muestra será el conjunto de entrenamiento para hacer crecer el árbol.
- Si hay M variables de entrada, se especifica un número m << M tal que en cada nodo, m variables son seleccionadas al azar de las M, luego se utiliza la mejor partición para dividir el nodo. El valor de m se mantiene constante durante el crecimiento del bosque.
- Cada árbol crece en la mayor medida posible. No hay poda.

Figura 5.3.4: Ejemplo de una ejecución de ML por Random Forest [50].



Explicados los métodos que utilizamos. Para la ejecución de estos algoritmos usamos R[11], el cual es un software estadístico gratuito y su paquete Caret[14], que ya cuenta con varias implementaciones de algoritmos de minería de datos.

Para la ejecución de los algoritmos mencionados, se dividió la nómina (n=5644) de estudiantes en 2 partes, a una razón 75 % para entrenar el modelo y un 25 % para validarlo, posteriormente se configuró la validación cruzada [32] a un valor 10. Finalmente se estipuló como medida de efectividad del modelo la Accuracy [45].

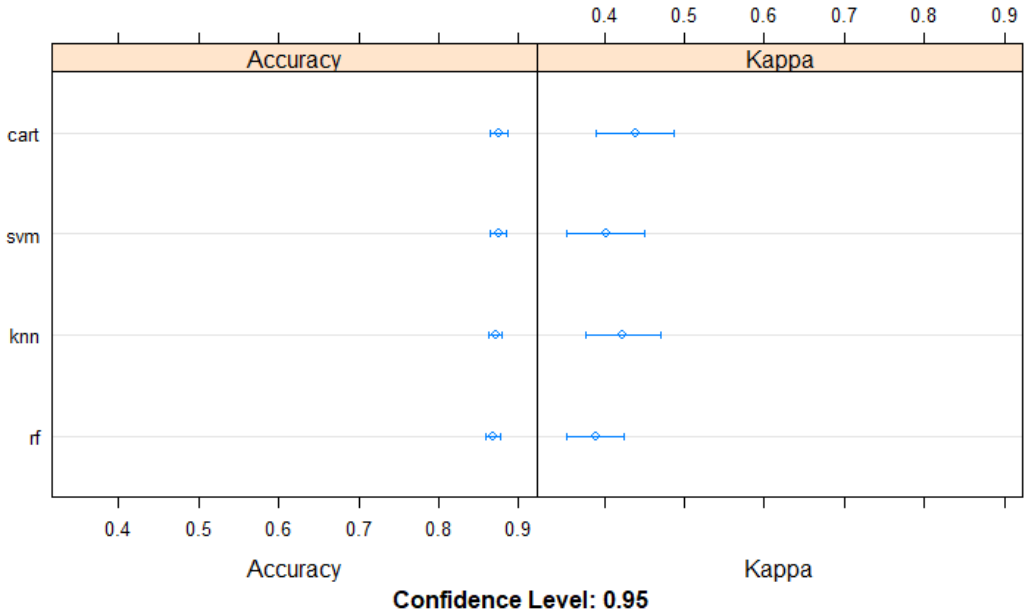
Ejecutados los 4 algoritmos, se procede a visualizar los resultados, lo cuales arrojan los que se muestran en la figura 5.3.5.

Figura 5.3.5: Resultados del modelo ejecutado

```
Models: cart, knn, svm, rf
Number of resamples: 10

Accuracy
          Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
cart 0.8537736 0.8660776 0.8770686 0.8753050 0.8837746 0.8959811    0
knn  0.8537736 0.8653277 0.8665892 0.8710530 0.8776596 0.8936170    0
svm  0.8514151 0.8684375 0.8724910 0.8748272 0.8812057 0.9007092    0
rf   0.8466981 0.8606023 0.8689504 0.8679825 0.8735225 0.8888889    0

Kappa
          Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
cart 0.3337051 0.3865319 0.4445897 0.4387379 0.5035109 0.5199122    0
knn  0.3407126 0.3816782 0.4055012 0.4237152 0.4531661 0.5402590    0
svm  0.3248390 0.3540762 0.3832402 0.4019090 0.4458152 0.5110903    0
rf   0.3171457 0.3559200 0.3797888 0.3893372 0.4248013 0.4746452    0
```

En donde los mismos se pueden ver gráficamente como se muestran en la figura 5.3.8.

Figura 5.3.6: Gráfica de métricas Accuracy y Kappa para los algoritmos ejecutados



Como se puede apreciar en la figura se muestran 2 datos: uno es el Accuracy y el otro kappa [35], tomaremos Ramdom Forest, la razón la veremos más adelante.

Una vez entrenado el modelo, es interesante ver como se comporta con un juego de datos independiente, para ello tomaremos la nómina correspondiente al 25 %

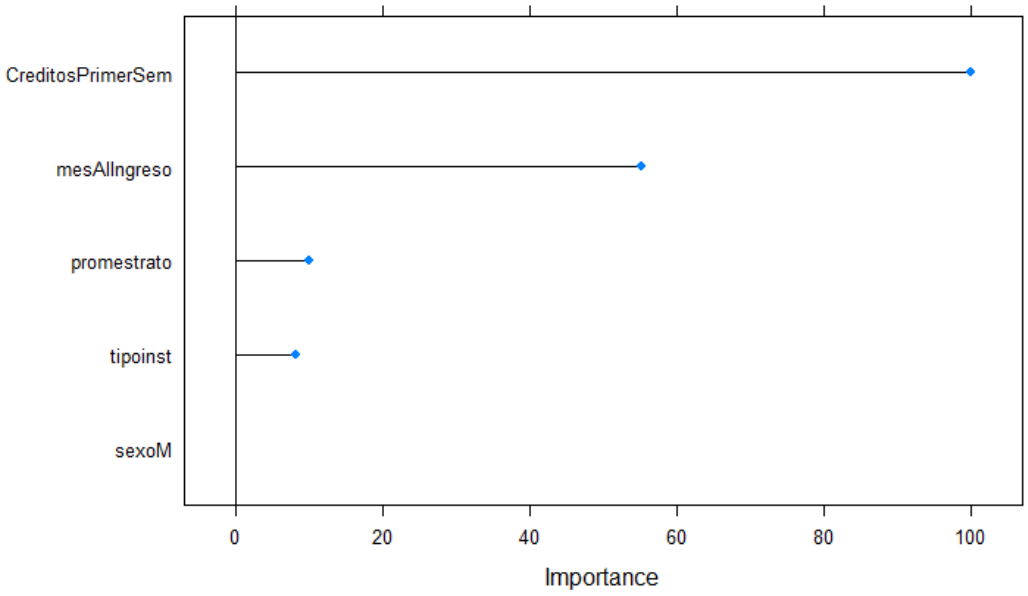
que habíamos reservado para validar el modelo. Ejecutada la validación se pueden observar en la figura 5.3.7.

Figura 5.3.7: Matriz de confusión para el modelo generado N=5644

```
Confusion Matrix and Statistics


predictions    DESVINCULADO RECIBIDO
  DESVINCULADO         1260       51
  RECIBIDO               21       79

               Accuracy : 0.949
                 95% CI : (0.9362, 0.9599)
    No Information Rate : 0.9079
    P-Value [Acc > NIR] : 5.347e-09

                  Kappa : 0.6597

 Mcnemar's Test P-Value : 0.0006316

              Precision : 0.9611
                 Recall : 0.9836
                     F1 : 0.9722
             Prevalence : 0.9079
         Detection Rate : 0.8930
   Detection Prevalence : 0.9291
      Balanced Accuracy : 0.7956

       'Positive' Class : DESVINCULADO
```

En donde podemos observar una alta precisión en predecir quienes se desvincularán de la carrera (1260 de 1311 predicciones), mientras que en un menor porcentaje se pueden predecir los recibidos (79 en 100). El motivo porque elegimos Ramdom Forest es que en la implementación que estamos usando es posible medir la importancia de las variables en el modelo según se muestra en la figura 5.3.8,

Figura 5.3.8: Gráfico de importancia de variables en el modelo



En el eje Y podemos ver cada variables, y en el X la importancia donde la misma se calcula como: "Para cada árbol, se registra la precisión de la predicción en la porción out-of-bag de los datos. Luego, se hace lo mismo después de permutar cada variable predictiva. La diferencia entre las dos precisiones se promedia sobre todos los árboles y se normaliza mediante el error estándar. Para la regresión, el error cuadrático medio se calcula en los datos out-of-bag para cada árbol, y luego se calcula de la misma manera después de permutar una variable. El error estándar promedia y normaliza las diferencias. Si el error estándar es igual a 0 para una variable, la división no se realiza"

Como vimos hasta ahora, mediante el uso de dos técnicas distintas hemos podido hacer un primer diagnóstico, encontrar posibles causas y hacer predicciones respecto a la desvinculaciones en la carrera, sin dudas, obteniendo más información del estudiante (parciales, laborales, familiares, etc.) la precisión aumentaría y en consecuencia esto permitiría mejor asistencia en la toma de decisiones.

5.4. Aplicación de minería de procesos

El objetivo de la siguiente sección es la utilización de procesos de minería de datos, específicamente minería de procesos, con el objetivo de modelar las trayectorias de los estudiantes de la carrera de Ingeniería en Computación para el plan 97

utilizando los datos provistos por la UEFI.

Es necesario definir en primer lugar a que nos referimos con "trayectoria de un estudiante", específicamente en el contexto de minería de procesos. Definimos como trayectoria de un estudiante como la secuencia de diferentes actividades (unidades curriculares, tareas, parciales, exámenes, etc.) que realiza un estudiante desde que inicia su vida estudiantil hasta que la completa, tanto egresando como desvinculándose de la misma. Enfocados específicamente en la minería de procesos, solo consideramos relevantes aquellas actividades asociadas a la aprobación de unidades curriculares dentro de la trayectoria ya que es la trayectoria final que deseamos modelar.

Para este análisis se utiliza la herramienta de minería de procesos ProM Tools [20], la cual permite la aplicación de diferentes algoritmos a un conjunto de datos provistos.

En las siguientes secciones se presenta la herramienta de minería de procesos ProM Tools, como fue recolectada la información para ser utilizada en la herramienta, la aplicación de minería de procesos en si, análisis de los resultados obtenidos y conclusiones generales sobre los resultados y la herramienta en si.

5.4.1. ProM Tools

ProM Tools es un framework que proporciona un entorno versátil y extensible para minería de procesos. Cuenta con plug-ins para extraer diferentes tipos de modelos de logs de eventos, por ejemplo, la construcción de un proceso y modelos organizativos. Además, admite la conversión y el análisis de modelos.

Usando técnicas de verificación de conformidad, los modelos también se pueden comparar con la realidad y aquellos existentes se pueden mejorar con información adicional, por ejemplo, detectando cuellos de botella dentro de un proceso en particular.

Es independiente del sistema operativo, ya que se encuentra implementado en Java y, al ser open source, cuenta con una gran comunidad y una vasta disponibilidad de plug-ins para diferentes tareas (conexiones a bases de datos, diferentes algoritmos de minería de datos, etc).

Como se dijo anteriormente, el objetivo de la aplicación de minería de procesos es lograr obtener el modelo de trayectoria que caracteriza a los estudiantes de Facultad de Ingeniería en la carrera Ingeniería en Computación para el plan 97.

Con este fin es necesario primero realizar un proceso de ETL para obtener la información necesaria a la cual aplicarle el proceso de minería en si.

5.4.2. Extracción, Transformación y Carga (ETL)

Esta es la primer etapa para poder aplicar métodos de minería de datos. Es necesario obtener la información con la cual se va a trabajar, realizar las transformaciones necesarias, limpieza e unificación para su posterior carga en el sistema a ser utilizado, en nuestro caso ProM Tools. Cabe aclarar que este proceso de ETL fue realizado de forma particular para el análisis en ProM Tools.

La información con la que se trabaja es el registro de Bedelías provista por la UEFI y en las secciones siguientes se describe de qué tablas proviene, el proceso de transformación y carga.

5.4.2.1. Extracción:

La base de datos provista por la UEFI cuenta con el conjunto de tablas descripta en el Capitulo 3, para este análisis se utilizaran las tablas **actividades**, **estudiante**, **rel_est_carr** y **asignatura** las cuales corresponden con las tablas de la base de datos original a **Activ2** [3.0.1], **Estudiantes** [3.0.4], **estudCarr** [3.0.3] y **asigCarr** [3.0.5] respectivamente.

Con este conjunto de tablas es posible determinar la escolaridad de un estudiante dado y cargar los datos necesarios para la aplicación de minería de procesos.

5.4.2.2. Transformación:

Como se encuentran actualmente estructuradas las tablas no es posible obtener de forma simple las escolaridades de los estudiantes. Por ello, es necesario realizar un conjunto de transformaciones y limpieza de datos para llegar a información de utilidad para nuestro análisis.

El objetivo de esta sección es obtener la información curada con el formato descripto en la tabla 5.4.1.

Cuadro 5.4.1: Información necesaria para aplicación de minería de procesos.

Columna	Tipo	Descripción
cedula	INT	Cédula del estudiante
unidad_curricular	VARCHAR	Nombre de la UC
fecha_inicio	DATE	Fecha de primer inscripción

Cuadro 5.4.1: Información necesaria para aplicación de minería de procesos.

Columna	**Tipo**	**Descripción**
fecha_fin	DATE	Fecha de aprobación
estado	VARCHAR	Estado actual del estudiante (DESVINCULADO, CURSANDO, RECIBIDO)

Esta tabla describe las diferentes asignaturas aprobadas por los estudiantes, es decir describe la escolaridad de los mismos.

Un punto a tener en cuenta es que para los siguientes análisis la fecha de inscripción de las UC no será utilizada. Esto se debe a que se encuentra una gran cantidad de registros cuya fecha de inscripción y aprobación son idénticas. Estas particularidades se detectaron en varios escenarios (UC exoneradas, UC que requieren examen obligatorio, UC que no requieren examen obligatorio, etc) y podrían ser causante de desviaciones en los modelos, por lo que solo será considerada la fecha de aprobación. Con esta fecha podemos obtener un estimado general de los tiempos de aprobación para las diferentes UC.

Por último, como parte de la transformación de datos, es necesario además realizar análisis y limpieza de estos, en este caso se aplicaron las siguientes acciones:

Eliminación de trayectorias aprobadas por revalida: Para nuestro análisis no consideramos de relevancia las UC que fueron aprobadas por reválidas, ya que son las que se encuentran en menor proporción dentro del total, lo cual agregaría casos aislados en nuestro análisis, generando alteraciones en los resultados.

Eliminación de UC optativas: Las UC optativas son necesarias para completar los créditos requeridos para egresar, sin embargo, comparado con el total de las asignaturas de la currícula obligatoria, son un porcentaje menor y esto podría generar alteraciones en el análisis y modelo generado, por lo que se decidió excluirlas.

Determinar UC a considerar: La trayectoria sugerida para la carrera de Ingeniería en Computación provista por Bedelías[22] fue utilizada como punto de partida para seleccionar qué UC serán consideradas para el análisis. En la tabla 5.4.2 podremos encontrarlas.

Cuadro 5.4.2: UC según la trayectoria sugerida por Bedelías.

Unidad curricular (UC)
Cálculo 1
Geometría y Álgebra Lineal 1
Física 1
Calculo 2
Geometría y Álgebra Lineal 2
Programación 1
Matemática Discreta 1
Probabilidad y Estadística
Matemática Discreta 2
Lógica
Programación 2
Ciencia, Tecnología y Sociedad
Arquitectura de Computadoras
Programación 3
Economía
Métodos Numéricos
Introducción a la Investigación Operativa
Sistemas Operativos
Programación 4
Teoría de Lenguajes
Fundamentos de Bases de Datos
Taller de Programación
Redes de Computadoras
Introducción a la Administración para Ingenieros
Introducción a la Ingeniería de Software
Programación Funcional
Programación Lógica
Proyecto de Ingeniería de Software
Proyecto de Grado

Unificación de UC: Las UC no son estáticas en el tiempo, se fueron versionando, cambiando de nombre, código o contenido. Se pueden encontrar distintas versiones para una misma asignatura. Cálculo 1 es un ejemplo, en donde a partir del año 2017 paso a llamarse **Calculo Dif. e Integral en una Variable**, además de cambiar parte de su contenido.

Para nuestro análisis estos cambios dentro de las UC no son relevantes, por lo que se optó por unificar y agrupar los diferentes códigos de UC que están relacionadas. A modo de ejemplo, algunas de las unificaciones que fueron realizadas se pueden encontrar en la tabla 5.4.3.

Cuadro 5.4.3: Unificación de UC.

Código	**UC**	**Grupo**
1020	Cálculo 1	1
1070	Cálculo 1	
1061	Cálculo Dif. e Integral en una variable	
1030	Geometría y Álgebra lineal 1	2
1071	Geometría y Álgebra lineal 1	
1021	Álgebra	
1151	Física 1	3
1171	Física 1	
1120	Física General 1	
1121	Física General 2	

Por ejemplo, en la tabla anterior podemos observar como las UC cuyos códigos son 1020, 1070 y 1061 se unifican en el grupo 1 bajo el concepto de **Cálculo 1**. El resto de las unificaciones de UC se pueden encontrar dentro del anexo B Minería de procesos.

Luego de la unificación de UC y basándonos en la trayectoria sugerida [5.4.2], contamos con un total de 24 UC para el análisis las cuales se encuentran en la tabla 5.4.4.

Cuadro 5.4.4: UC utilizadas para el análisis.

Unidad curricular
Cálculo 1
Geometría y Álgebra Lineal 1
Física 1
Calculo 2
Geometría y Álgebra Lineal 2
Programación 1
Matemática Discreta 1
Probabilidad y Estadística
Matemática Discreta 2
Lógica

Cuadro 5.4.4: UC utilizadas para el análisis.

Unidad curricular
Programación 2
Arquitectura de Computadoras
Programación 3
Métodos Numéricos
Introducción a la Investigación Operativa
Sistemas Operativos
Programación 4
Teoría de Lenguajes
Fundamentos de Bases de Datos
Taller de Programación
Redes de Computadoras
Introducción a la Ingeniería de Software
Proyecto de Ingeniería de Software
Proyecto de Grado

5.4.2.3. Carga:

La tabla de logs (o escolaridades) descripta en 5.4.1 es obtenida mediante una vista generada a partir de las tablas mencionadas anteriormente, la consulta completa puede ser vista en el anexo B Minería de procesos.

Luego, es posible extraer los datos de la misma en formato CSV [17], cargarlos en ProM y, posteriormente, procesarlos. Para poder aplicar los diferentes algoritmos de minería de procesos provistos por la herramienta es necesario convertir los datos crudos (CSV) a XES[23], que es el formato requerido por ProM. Este formato permite la representación de eventos en formato basado en XML. Para realizar esta conversión de CSV a XES es posible utilizar un plug-in ya provisto en la herramienta donde es necesario contar con un conjunto mínimo de datos. La información necesaria se puede ver en la tabla 5.4.5 y se puede asociar directamente a la información provista por el log (tabla 5.4.1).

Cuadro 5.4.5: XES.

Columna	**Descripción**
Identificador del caso	Se mapea con la cédula del estudiante
Identificador del evento	Se mapea con la UC
Fecha de completitud	Se mapea con la fecha de aprobación a la UC

El proceso de carga de datos en ProM se realiza de forma muy simple, solamente es necesario abrir la herramienta, importar el CSV a procesar y luego ejecutar el plug-in correspondiente para la conversión.

Figura 5.4.1: Proceso de conversión de CSV a XES en ProM.

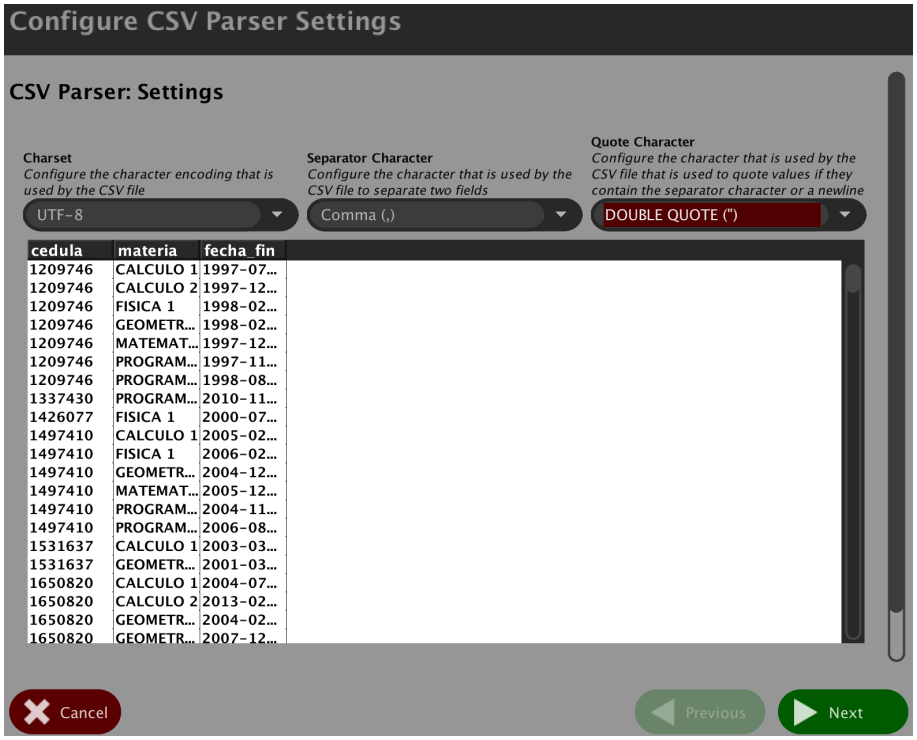
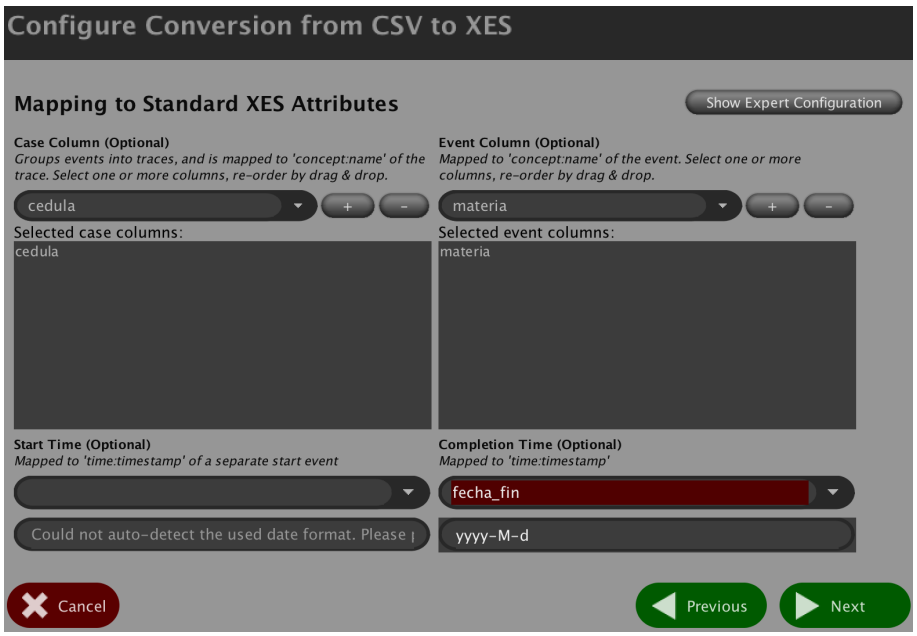


Figura 5.4.2: Proceso de conversión de CSV a XES en ProM.



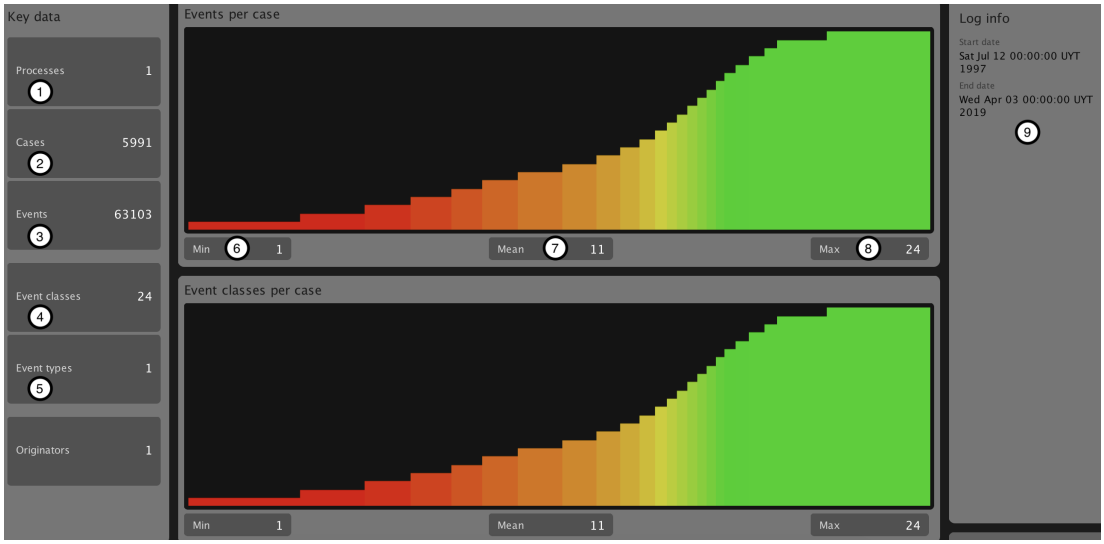
En la figura 5.4.1 y 5.4.2 podemos observar el matcheo directo entre las diferentes columnas del archivo CSV a convertir y las propiedades del archivo XES descriptas en la tabla 5.4.5.

5.4.2.4. Análisis del log en ProM

Luego de importar y convertir los datos al formato deseado, es posible obtener información relevante dentro del archivo XES generado. En la figura 5.4.3 podemos encontrar:

- **(1) Cantidad de procesos**: En este caso solo contamos con uno, la trayectoria de los estudiantes.
- **(2) Cantidad de casos**: Se corresponde con la cantidad de estudiantes totales dentro del log provisto.
- **(3) Eventos**: Corresponde a la cantidad de aprobaciones para las diferentes UC por parte de los estudiantes.
- **(4) Cantidad de clases de eventos**: Contamos con 24 eventos diferentes ya que contamos con 24 UC.
- **(5) Tipos de eventos**: Para el análisis consideramos solo la aprobación por ello contamos con un solo tipo de eventos (complete).
- **(6) Cantidad mínima de eventos por caso**: Corresponde al mínimo de eventos de aprobación que se registran por estudiante.
- **(7) Cantidad máxima de eventos por caso**: Corresponde al máximo de eventos de aprobación que se registran por estudiante.
- **(8) Cantidad promedio de eventos por caso**: Corresponde al promedio de eventos aprobación que se registran por estudiante.
- **(9) Fecha del primer y ultimo registro detectado**: Indica el rango de fechas en el cual el log tiene registros.

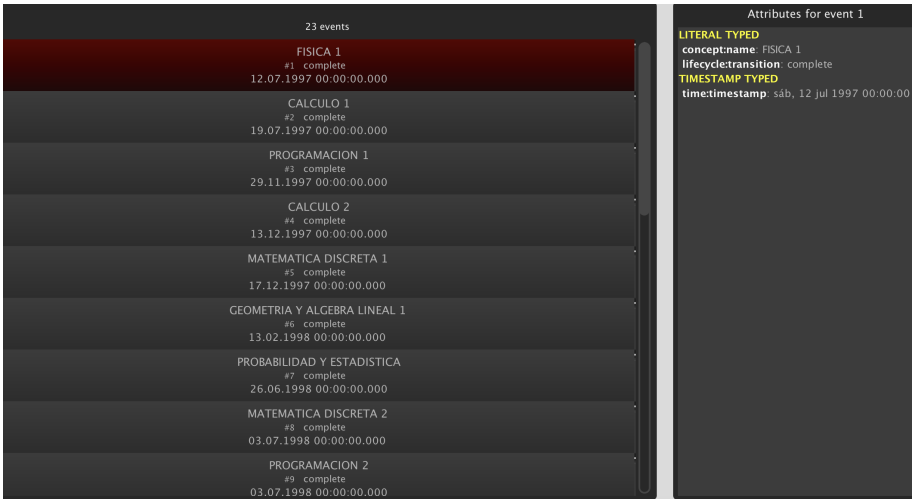
Figura 5.4.3: Información provista en el archivo XES.



Otra información que obtenemos del log mediante ProM es el detalle de los diferentes estudiantes. Es posible inspeccionar como es el transcurso de un estudiante por la carrera, lo que implica conocer a que UC se inscribió, en que fecha aprobó la misma, y qué tan frecuente es esa UC en el log. En la figura 5.4.4 podemos observar la siguiente información:

- Listado de los eventos registrados para el estudiante, en este caso las aprobaciones de la misma.
- Nombre de la UC.
- Tipo de evento, en nuestro caso corresponde a la aprobación del mismo.
- Fecha del evento.

Figura 5.4.4: Detalle para un estudiante.



También es posible explorar el orden de los eventos registrados para cada estudiante, conociendo además su frecuencia. En la figura 5.4.5 podemos observar dicha información.

Figura 5.4.5: Detalle para un estudiante.

Para este caso particular podemos ver que el estudiante registra 21 eventos. Para el caso de la UC **Programación 1**, podemos observar que la misma fue aprobada en la fecha 22/07/1999. Además, es posible conocer la frecuencia de los eventos, representada mediante un rango de colores. El color verde indica mayor frecuencia de aparición de la UC en el log y el rojo, lo contrario.

También contamos con una sección de sumario donde es posible observar información agrupada y condensada. Allí podemos encontrar el total de casos y eventos correspondientes al total de estudiantes y aprobaciones respectivamente. En la figura 5.4.6 podemos observar para las diferentes UC su cantidad de ocurrencias absolutas / relativas. A modo de ejemplo podemos observar que la UC **Geometría y Álgebra Lineal 1** es la UC con mas apariciones dentro del log.

Figura 5.4.6: Sumario de UC.

Event Name

Event classes defined by Event Name
All events

Total number of classes: **24**

Class	Occurrences (absolute)	Occurrences (relative)
GEOMETRIA Y ALGEBRA LINEAL 1	5034	7,977%
PROGRAMACION 1	4547	7,206%
CALCULO 1	4452	7,055%
MATEMATICA DISCRETA 1	3998	6,336%
GEOMETRIA Y ALGEBRA LINEAL 2	3665	5,808%
FISICA 1	3642	5,772%
CALCULO 2	3363	5,329%
PROGRAMACION 2	3122	4,947%
PROBABILIDAD Y ESTADISTICA	2763	4,379%
MATEMATICA DISCRETA 2	2607	4,131%
PROGRAMACION 3	2467	3,909%
LOGICA	2397	3,799%

Luego, dentro del mismo sumario, podemos encontrar información referente a la primer y última aprobación. En la figura 5.4.7 podemos observar que cerca del **46 %** de los estudiantes comienzan sus estudios aprobando la UC **Geometría Y Álgebra Lineal 1**.

Figura 5.4.7: Sumario de primeras aprobaciones.

Start events

Total number of classes: **10**

Class	Occurrences (absolute)	Occurrences (relative)
GEOMETRIA Y ALGEBRA LINEAL 1	2728	45,535%
PROGRAMACION 1	970	16,191%
FISICA 1	872	14,555%
CALCULO 1	865	14,438%
MATEMATICA DISCRETA 1	400	6,677%
GEOMETRIA Y ALGEBRA LINEAL 2	104	1,736%
CALCULO 2	44	0,734%
PROGRAMACION 2	6	0,1%
LOGICA	1	0,017%
INT. A LA INVESTIGACION DE OPERACIONES	1	0,017%

Por otra parte, es posible también visualizar cuales fueron las últimas aprobaciones para los diferentes estudiantes. Dentro de la figura 5.4.8 podemos observar que la UC **Proyecto De Grado** ocupa el primer puesto de última UC aprobada con cerca de un **14 %**, seguida de **Programación 1** con cerca de **11 %**, lo cual indicaría que un **11 %** de los estudiantes siguen cursando la carrera, o en el peor de los escenarios, se desvincularon de la misma.

Figura 5.4.8: Sumario de ultimas aprobaciones.

End events

Total number of classes: **24**

Class	Occurrences (absolute)	Occurrences (relative)
PROYECTO DE GRADO	817	13,637%
PROGRAMACION 1	645	10,766%
GEOMETRIA Y ALGEBRA LINEAL 1	468	7,812%
FISICA 1	426	7,111%
CALCULO 1	406	6,777%
MATEMATICA DISCRETA 1	392	6,543%
REDES DE COMPUTADORAS	376	6,276%
METODOS NUMERICOS	298	4,974%
PROGRAMACION 2	293	4,891%
CALCULO 2	239	3,989%
PROYECTO DE INGENIERIA DE SOFTWARE	187	3,121%
GEOMETRIA Y ALGEBRA LINEAL 2	182	3,038%
PROBABILIDAD Y ESTADISTICA	162	2,704%

5.4.3. Minería de procesos en ProM

Como se dijo anteriormente, ProM Tools es una herramienta extensible que cuenta con una vasta comunidad de desarrolladores, quienes proveen una gran cantidad de plug-ins, diferentes algoritmos de minería y análisis de datos.

En primer instancia obtenemos un modelo de dependencias para el log provisto que será generado utilizando el plug-in Interactive Data-Aware Heuristic Miner [41], lo que permitirá la aplicación de diferentes algoritmos para la generación de modelos, además de poder presentar el modelo obtenido utilizando diferentes representaciones de procesos (Dependency nets, Petri nets, Causal nets, etc). Para nuestro análisis generaremos un grafo de dependencias utilizando el Minero Heurístico Flexible [52] como algoritmo de descubrimiento. Este grafo de dependencias será utilizado para obtener una primera idea del modelo subyacente en el log.

Luego, con el objetivo de generar un modelo de trayectoria para los estudiantes, se analizaron diferentes algoritmos, cuyo objetivo es explorar y descubrir el modelo subyacente dentro del log provisto. En nuestro caso de estudio, se optó por la utilización un algoritmo inductivo [25], para esto se dispone del plug-in "Mine Petri net with Inductive Miner" el cual lo implementa, permite tomar el log a analizar y producir una red Petri como resultado. Esta decisión se basó en los resultados presentes en el paper de "Discovering learning processes using Inductive Miner: A case study with Learning Management Systems (LMSs)" [25], ya que dicho algoritmo presentó los mejores resultados si es comparado con otros algoritmos tradicionales de minería de procesos. Una consideración a tener en cuenta es que para los diferentes análisis fue utilizada la configuración por defecto de los algoritmos, salvo que se indique lo contrario en algún análisis.

Por ultimo, para realizar análisis sobre el modelo obtenido, se utilizarán los siguientes plug-ins sobre el mismo: Replay a log on Petri Net for Conformance/Performance analysis, Measure precision/generalization.

Replay a log on Petri Net for Conformance/Performance analysis: Este plug-in tiene como objetivo obtener métricas generales sobre el modelo, tanto de conformidad como de performance del mismo. Para el caso de análisis de conformidad tenemos las siguientes métricas a analizar:

- **Fitness**: Que porcentaje de las trazas dentro del log son reconocidas por el modelo generado.
- **Desviaciones**: Conocer en que conjunto de actividades se presentan desviaciones entre el log y el modelo, es decir, donde se pueden observar únicamente movimientos presentes en el log o en el modelo.

En cuanto a las métricas de performance la información relevante es:

- **Throughput promedio**: Duración promedio de las diferentes trazas.
- **Detectar cuellos de botella**: Poder determinar que UC requieren mas tiempo para ser aprobadas.

Measure precision/generalization: Dado un modelo, el análisis de conformidad del mismo y el log este plug-in permite determinar las siguientes métricas:

- **Precisión**: Muestra la proporción del comportamiento representado por el modelo que no se ve en el log.
- **Generalización**: Evalúa la medida en que el modelo podrá reproducir el comportamiento futuro del proceso y puede verse como una medida de confianza en la precisión.

5.4.4. Resultados

En esta sección encontraremos los resultados obtenidos luego de aplicar el proceso de minería sobre los datos provistos por la UEFI. El análisis será realizado sobre tres grupos de estudiantes diferentes:

- Estudiantes recibidos.
- Estudiantes desvinculados.
- Estudiantes que se encuentran cursando pero están avanzados en la carrera.

Esta distinción fue realizada para generar modelos específicos dentro de cada categoría, y así evitar que las características de cada caso afectaran a los demás. El fundamento principal de dicha distinción esta dada por la gran cantidad de estudiantes desvinculados (cerca del 62 % de los estudiantes considerados), estos estudiantes si son considerados con los demás pueden generar alteraciones al modelo final debido a su escasa cantidad de UC aprobadas.

Además, para cada caso se hicieron las siguientes consideraciones:

- Del conjunto de estudiantes desvinculados de la carrera solamente fueron considerados aquellos que no superaban los 100 créditos aprobados, ya que corresponde a cerca del 84 % de ellos.
- Siguiendo con los estudiantes que egresaron de la carrera, solo serán considerados aquellos que además hayan aprobado las 24 UC a analizar.
- Por último, del total de estudiantes que se encuentran cursando se tomó como condición de "encontrarse avanzados en la carrera", es decir, que contaran con al menos tres cuartos (18) de las UC aprobadas dentro de las UC consideradas.

Luego de estas consideraciones contamos con los datos de 3477 estudiantes, los cuales generan en total 33396 eventos dentro del log general, el cual se extiende desde el año 1997 hasta el 2019 (momento en que nos fueron provistos los datos). En las siguientes secciones se analiza cada uno de los escenarios antes descriptos.

5.4.4.1. Estudiantes recibidos

Para este análisis contamos con los datos de 684 estudiantes (aproximadamente 20 % del total), los cuales generan un total de 16416 eventos (aproximadamente 49 % del total). Como se dijo anteriormente, solo se consideraran los estudiantes recibidos que contaban con la aprobación de las 24 UC propuestas para el análisis, ya que se encontraron casos donde el estudiante se encontraba recibido pero no cumplía esta condición. Estos casos incompletos fueron descartados para el análisis ya que eran la minoría y podían generar alteraciones en el modelo final.

Alguna de las causas que se lograron identificar para la falta de UC para ciertos estudiantes son:

- No se encuentra el registro en la base de datos provista.
- El caso de la UC **Álgebra** la cual se dividió en **Geometría y Álgebra Lineal 1 y 2**.

Análisis del log:

Al realizar un análisis simple del log mediante las herramientas provistas en ProM, podemos observar que cerca del 45 % de las ocurrencias Geometría y Álgebra Lineal 1 corresponden a la primer UC aprobada por los estudiantes (figura 5.4.9). Para el caso de última aprobación, cerca del 67 % de los estudiantes culmina su carrera aprobando el Proyecto de Grado. Como UC interesantes a mencionar tenemos a Redes de Computadoras y Métodos Numéricos, que ocupan ocupan el 2do y 3er lugar como últimas UC aprobadas al recibirse (figura 5.4.10).

Figura 5.4.9: Primeras aprobaciones para estudiantes egresados.

Start events

Total number of classes: **4**

Class	Occurrences (absolute)	Occurrences (relative)
GEOMETRIA Y ALGEBRA LINEAL 1	444	48,525%
FISICA 1	390	42,623%
CALCULO 1	78	8,525%
PROGRAMACION 1	3	0,328%

Figura 5.4.10: Últimas aprobaciones para estudiantes egresados.

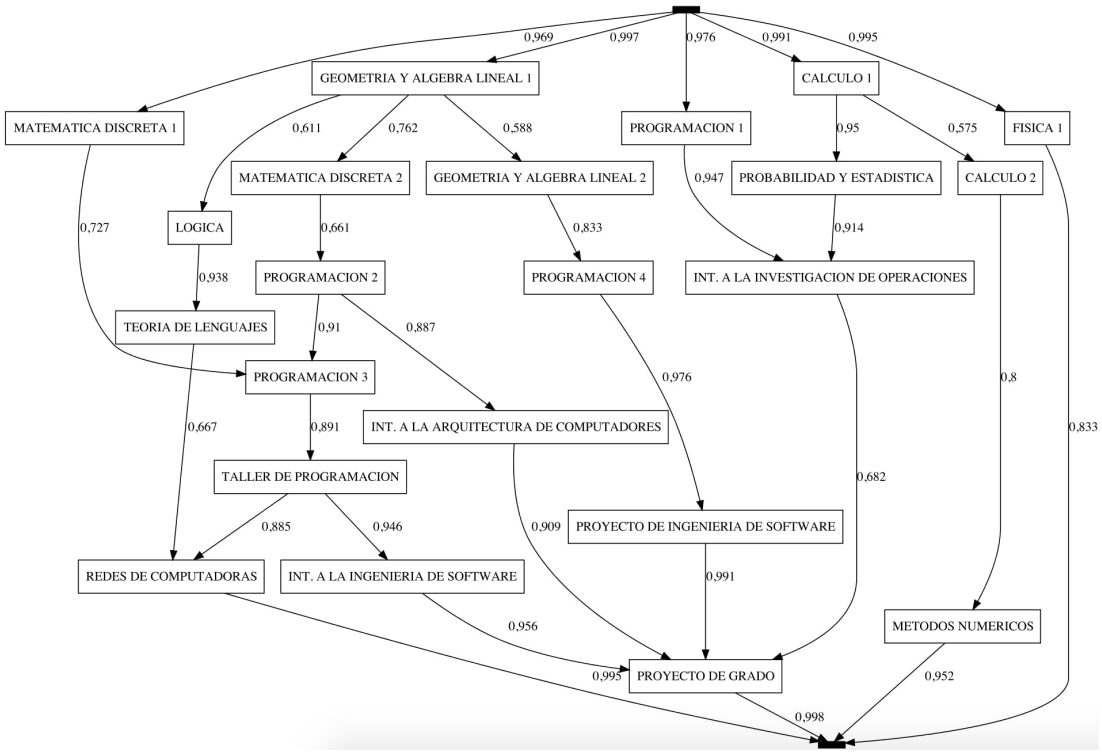
End events

Total number of classes: **7**

Class	Occurrences (absolute)	Occurrences (relative)
PROYECTO DE GRADO	649	70,929%
REDES DE COMPUTADORAS	223	24,372%
METODOS NUMERICOS	31	3,388%
FISICA 1	6	0,656%
SISTEMAS OPERATIVOS	3	0,328%
INT. A LA INVESTIGACION DE OPERACIONES	2	0,219%
INT. A LA INGENIERIA DE SOFTWARE	1	0,109%

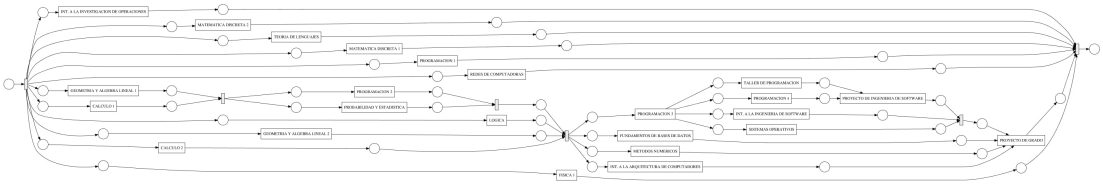
Modelo generado: Primeramente presentamos un modelo utilizando un grafo de dependencias, este tipo de grafos permite conocer las relaciones de dependencia entre las diferentes UC y obtener una idea general del modelo. En la figura 5.4.11 podemos observar el grafo obtenido a partir del log, los rectángulos representan las UC y los valores en los arcos indican la confianza de la dependencia señalada (a mayor valor mayor confianza sobre la relación de dependencia/precedencia entre las diferentes UC).

Figura 5.4.11: Grafo de dependencia para estudiantes recibidos.



Continuando con el análisis, se genera ahora un modelo basado en Redes Petri [42] con el objetivo de detectar información relevante dentro del proceso. Como se dijo anteriormente, la información que analizaremos será: fitness, throughput promedio, detección de cuellos de botella, precisión y generalización del modelo. En la figura 5.4.12 podemos observar la Red Petri generada luego de ejecutar el plug-in "Mine Petri net with Inductive Miner" sobre el log.

Figura 5.4.12: Petri net para estudiantes recibidos.

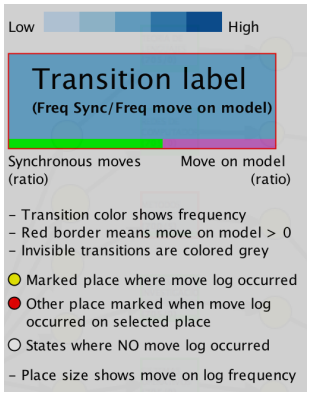


Por último, se procede a medir las diferentes métricas para el modelo generado, para ello se ejecutan los plug-ins "Replay a log on Petri Net for Performance/Conformance analysis" y "Measure precision/generalization".

Para poder comprender el resultado del mismo es necesario entender la forma en que se representan los resultados. Para ello tenemos la figura 5.4.13, donde en la leyenda podemos observar que:

- Los eventos (UC) se representan por rectángulos cuyo color va de celeste a azul el cual representa la frecuencia de aparición del evento.
- Un reborde rojo sobre el evento indica una desalineación entre el modelo y el log.
- Una barra debajo del evento indica el ratio entre los movimientos síncronos y los movimientos solo presentes en el modelo (verde y violeta respectivamente).

Figura 5.4.13: Leyenda sobre resultado de conformidad del modelo.



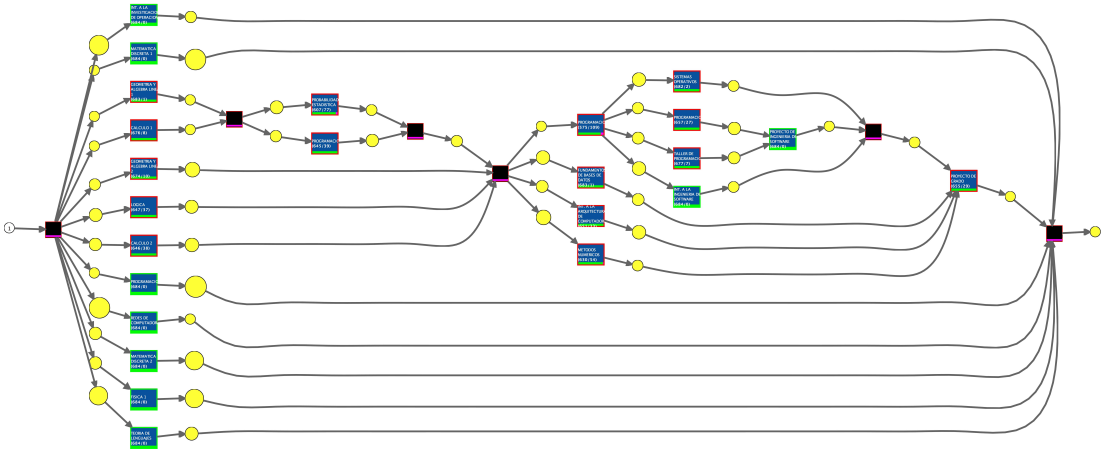
En la figura 5.4.14 podemos observar el resultado de conformidad del modelo para el log provisto. De este análisis se desprende que el modelo permite reproducir un 97 % de las trazas, donde 375 trazas de las 648 analizadas se alinean perfectamente al modelo obtenido. También podemos observar que 9 de las 24 UC se encuentran alineadas entre el modelo y el log y las restantes 15 presentan desviaciones ya que encontramos transiciones entre UC que solo son detectadas por el modelo pero no presentes en el log. Estas UC con desviaciones son:

- Cálculo 1
- Cálculo 2
- Geometría y Álgebra Lineal 1
- Geometría y Álgebra Lineal 2
- Lógica
- Probabilidad y Estadística
- Matemática Discreta 2
- Métodos Numéricos

- Programación 3
- Programación 4
- Introducción a la Arquitectura de Computadores
- Sistemas Operativos
- Taller de Programación
- Fundamentos de Bases de Datos
- Proyecto de Grado

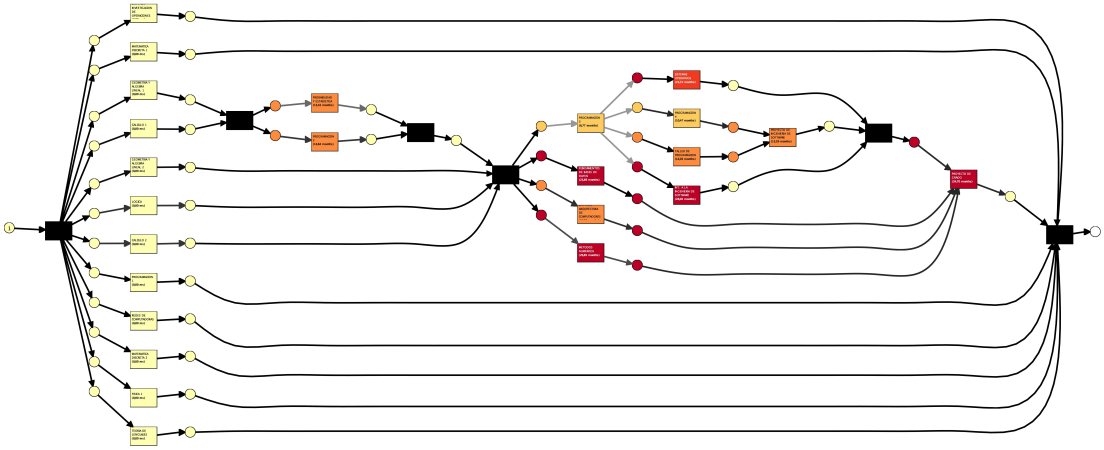
Para estos casos la desviación entre el modelo y el log no supera el 5 % de las trazas.

Figura 5.4.14: Análisis de conformidad del log sobre el modelo generado.



Luego se realiza un análisis de performance, el cual se observa en la figura 5.4.15. Aquí podemos determinar que: en lo que respecta al tiempo de ejecución de una traza (throughput), en promedio los estudiantes registran 8 años para aprobar la carrera, contando con mínimos de 4.5 años y máximos de casi 20 años.

Figura 5.4.15: Análisis de performance del log sobre el modelo generado.



Al realizar un análisis de los tiempos de espera promedios dentro del modelo, detectamos que existen 12 UC con un alto tiempo de espera. En la tabla 5.4.6 podemos obtener un detalle de las mismas.

Cuadro 5.4.6: Tiempos promedio de aprobación.

UC	**Tiempo promedio de aprobación (años)**
Métodos Numéricos	2.5
Introducción a la Ingeniería de Software	2.3
Proyecto de Grado	2.1
Fundamentos de Bases de Datos	1.9
Sistemas Operativos	1.9
Proyecto de Ingeniería de Software	1.3
Programación 2	1.2
Probabilidad y Estadística	1.1
Arquitectura de Computadores	1.0
Taller de Programación	1.0
Programación 4	0.8
Programación 3	0.6

Del análisis del tiempo de espera por UC podemos obtener información relevante respecto a cuellos de botella dentro del modelo, aquí podemos observar que las

UC Sistemas Operativos, Fundamentos de Bases de Datos, Proyecto de Grado, Introducción a la Ingeniería de Software y Métodos Numéricos presentan tiempos de aprobación superior a cuatro veces lo esperado si consideramos que deberían ser aprobadas en un semestre.

Es interesante observar también que cuanto mas avanza un estudiante en su carrera mas tiempo es necesario para la aprobación de las mismas, esto puede verse en la figura 5.4.15.

Por último, se obtienen algunas métricas respecto al modelo generado. En la tabla 5.4.7 podemos observar las métricas de precisión y generalización del modelo, las cuales se pueden obtener gracias al modelo generado, el análisis de conformidad del mismo y el log inicial.

Cuadro 5.4.7: Precisión y generalización del modelo generado.

Métrica	**Valor)**
Precesión	0.30126
Generalización	0.8949

Considerando la precisión obtenida podemos observar que, a simple vista, no se encuentra cercana a 1, igualmente este no es un valor a buscar en este tipo de construcción de modelos, ya que una precisión elevada indica que el modelo sobre ajusta a los datos y solamente será observable lo presente en el log, restringiendo el modelo a comportamientos no observados en él. El valor de generalización del modelo nos indica qué tan adaptable es a reconocer nuevos comportamientos no presentes dentro del log, dado su valor de 0.89 podemos concluir que el modelo es capaz de reconocer en gran medida nuevos comportamientos. Si unimos ambas medidas con el objetivo de obtener una conclusión final sobre el modelo, podemos decir que el mismo no se encuentra sobre ajustado a los datos utilizados para generarlos y además es capaz de soportar nuevo comportamiento no presente en el log brindándole gran flexibilidad ante nuevas trazas.

En cuanto a los tiempos de aprobación promedio, es posible ver que las UC Arquitectura de Computadores, Sistemas Operativos y Redes de Computadoras conllevan mayor tiempo promedio de aprobación, este es un dato no menor ya que estas UC son consideradas por los estudiantes como un grupo de gran dificultad y con estos números se corroboraría dicha teoría. Luego aparecen los casos de Física 1 y Métodos Numéricos, también con altos tiempos de aprobación, lo que puede deberse a que las mismas no son previa de UC en cierto punto de la carrera entonces los estudiantes pueden inscribirse una vez, no llegar a aprobarlas y luego

rendirlas mas adelante.

5.4.4.2. Estudiantes desvinculados

Para este análisis contamos con los datos de 2158 estudiantes (aproximadamente 62 % del total) los cuales generan un total de 8266 eventos (aproximadamente 25 % del total).

Análisis del log: En la figura 5.4.16 podemos observar que nuevamente la UC Geometría y Álgebra Lineal 1 es la primera aprobada con más de un 41 % de los casos.

Figura 5.4.16: Primeras UC aprobadas para estudiantes desvinculados.

End events

Total number of classes: **19**

Class	Occurrences (absolute)	Occurrences (relative)
GEOMETRIA Y ALGEBRA LINEAL 1	357	17,032%
PROGRAMACION 1	340	16,221%
FISICA 1	274	13,073%
MATEMATICA DISCRETA 1	249	11,88%
CALCULO 1	228	10,878%
PROGRAMACION 2	131	6,25%
GEOMETRIA Y ALGEBRA LINEAL 2	116	5,534%
METODOS NUMERICOS	96	4,58%
CALCULO 2	91	4,342%
PROBABILIDAD Y ESTADISTICA	63	3,006%
INT. A LA INVESTIGACION DE OPERACIONES	60	2,863%
MATEMATICA DISCRETA 2	42	2,004%
LOGICA	28	1,336%
PROGRAMACION 3	12	0,573%
TALLER DE PROGRAMACION	3	0,143%
PROGRAMACION 4	3	0,143%
INT. A LA ARQUITECTURA DE COMPUTADORES	1	0,048%
TEORIA DE LENGUAJES	1	0,048%
FUNDAMENTOS DE BASES DE DATOS	1	0,048%

Si continuamos analizando el log, si sumamos las ocurrencias de UC de primer año (Cálculo 1, Geometría y Álgebra Lineal 1, Física 1, Programación 1 y Matemática Discreta 1) podemos ver que cerca del 69 % de los estudiantes tienen como última UC aprobada alguna de ellas, por lo que se puede inferir que estos no avanzan mas allá del primer año. En la figura 5.4.17 podemos ver dicho comportamiento.

Figura 5.4.17: Últimas UC aprobadas para estudiantes desvinculados.

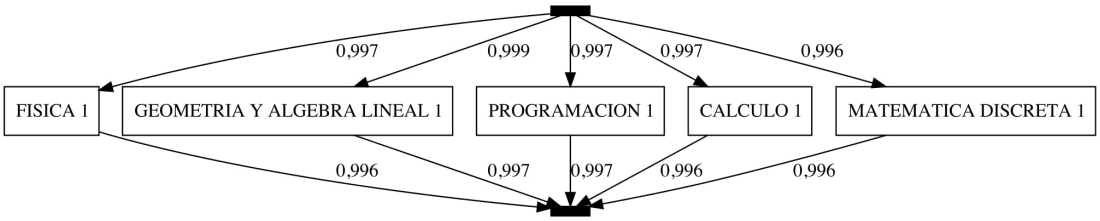
End events

Total number of classes: **19**

Class	Occurrences (absolute)	Occurrences (relative)
GEOMETRIA Y ALGEBRA LINEAL 1	357	17,032%
PROGRAMACION 1	340	16,221%
FISICA 1	274	13,073%
MATEMATICA DISCRETA 1	249	11,88%
CALCULO 1	228	10,878%
PROGRAMACION 2	131	6,25%
GEOMETRIA Y ALGEBRA LINEAL 2	116	5,534%
METODOS NUMERICOS	96	4,58%
CALCULO 2	91	4,342%
PROBABILIDAD Y ESTADISTICA	63	3,006%
INT. A LA INVESTIGACION DE OPERACIONES	60	2,863%
MATEMATICA DISCRETA 2	42	2,004%
LOGICA	28	1,336%
PROGRAMACION 3	12	0,573%
TALLER DE PROGRAMACION	3	0,143%
PROGRAMACION 4	3	0,143%
INT. A LA ARQUITECTURA DE COMPUTADORES	1	0,048%
TEORIA DE LENGUAJES	1	0,048%
FUNDAMENTOS DE BASES DE DATOS	1	0,048%

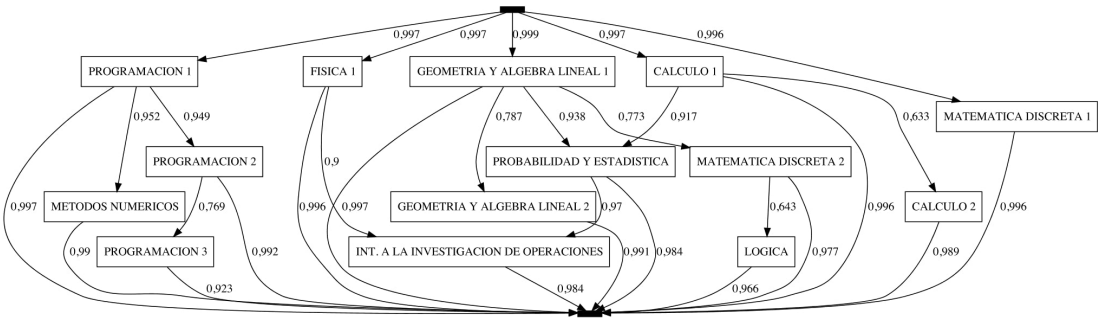
Modelo generado: Como se realizó en el caso anterior, primero se genera un modelo de dependencias para tener una visión general del modelo en si. Para esto haremos uso nuevamente del plug-in "Mine Petri net with Inductive Miner" sobre el log. Como se puede observar en la figura 5.4.18, vemos que para los parámetros por defecto del algoritmo solo se presentan las UC correspondientes al primer año de la carrera, lo cual es un reflejo de lo que se observó anteriormente al hacer un análisis del log en sí; los estudiantes se desvinculan antes de terminar primer año.

Figura 5.4.18: Grafo de dependencia entre UC para estudiantes desvinculados.



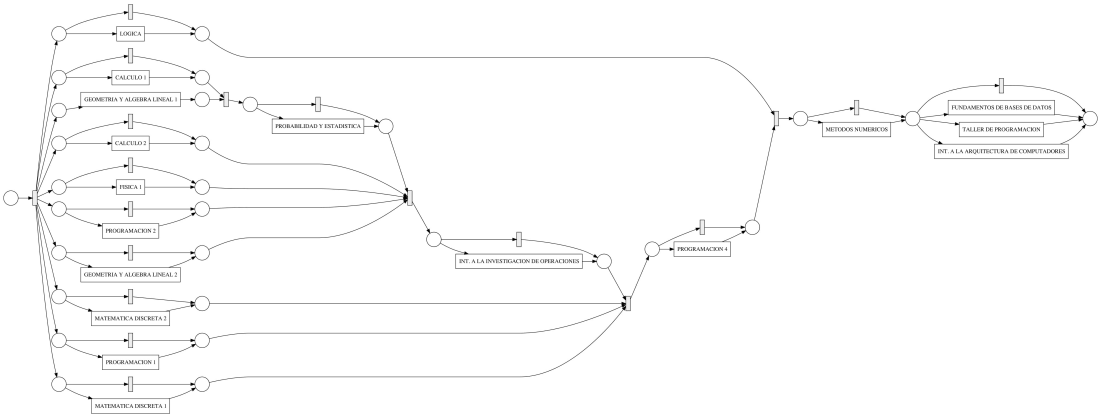
Un dato interesante a tener en cuenta es que si reducimos la frecuencia mínima necesaria para que una UC sea considerada por el algoritmo, es decir, sea visible en el modelo de dependencias, podemos observar en la figura 5.4.19 que los estudiantes que mas avanzan antes de desvincularse de la carrera completan desde Programación 1 hasta Programación 3. Igualmente cabe destacar que para la aparición de las mismas fue necesario reducir la frecuencia mínima de aparición a 0 para que considere todos los flujos directos dentro del log.

Figura 5.4.19: Grafo de dependencia entre UC para estudiantes desvinculados, frecuencia mínima requerida 0.



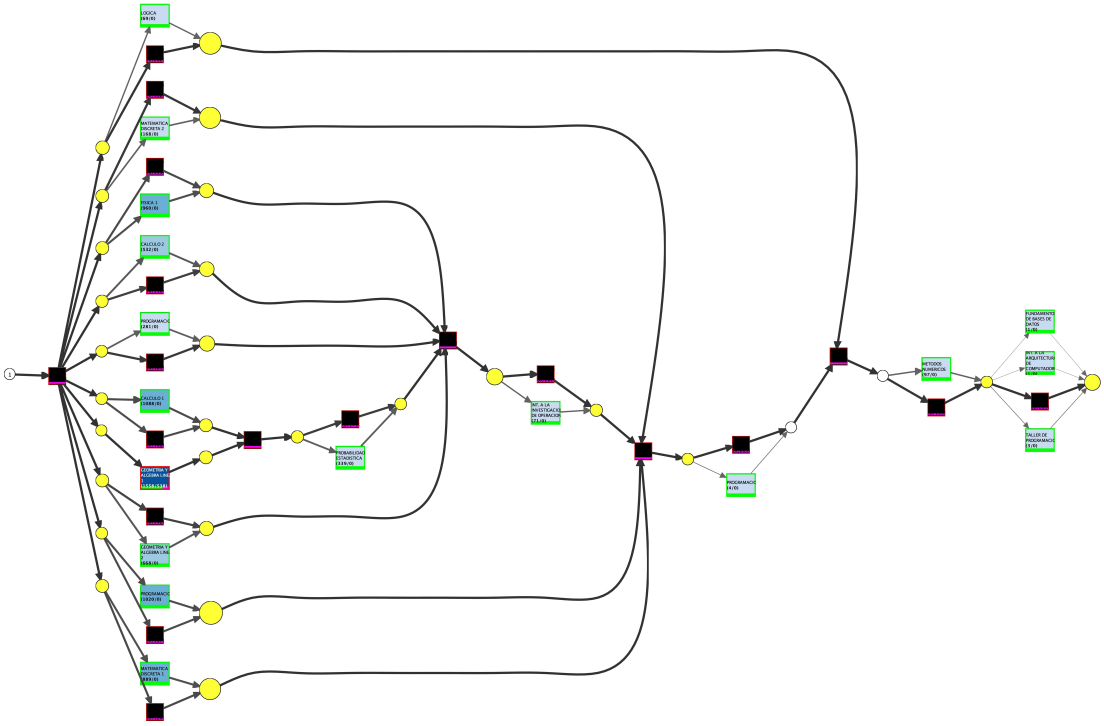
En la figura 5.4.20 podemos observar la red Petri generada luego de utilizar el minero inductivo sobre el log provisto, sobre este modelo procederemos a realizar análisis de conformidad y performance.

Figura 5.4.20: Red Petri para estudiantes desvinculados.



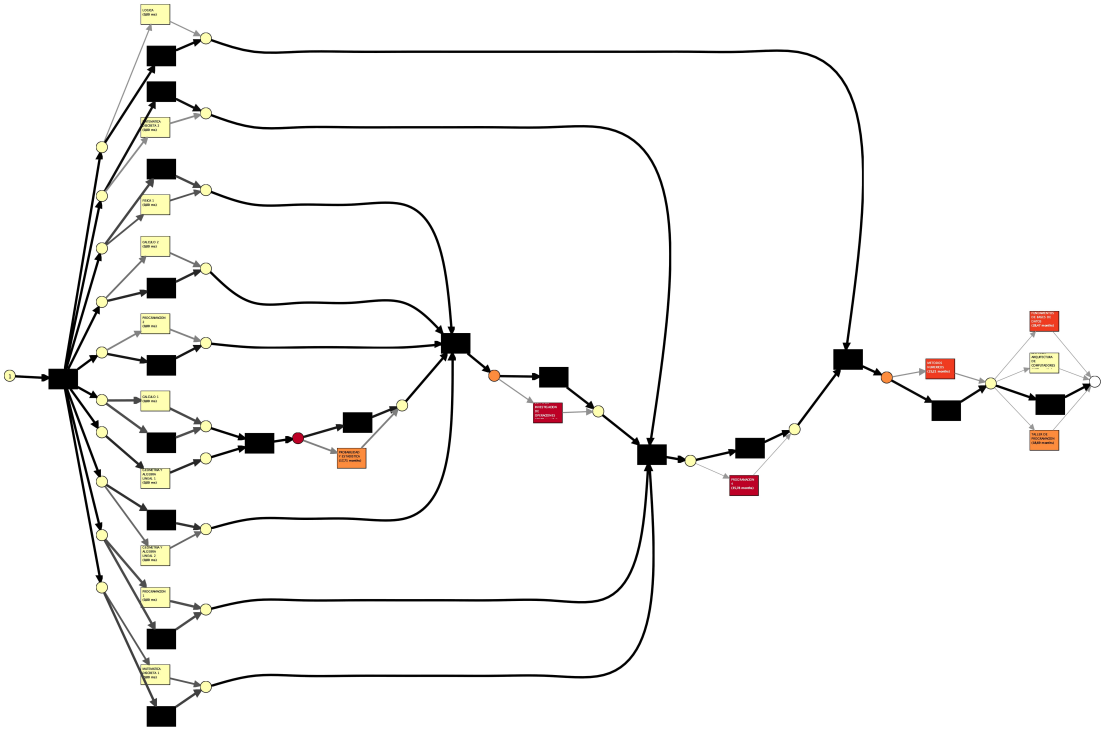
En cuanto a la conformidad del modelo sobre el log, luego de ejecutar el plug-in "Replay a log on Petri Net for Conformance/Performance" se observa que el mismo puede modelar 97 % de las trazas presentes en el log. En la figura 5.4.21 podemos observar que las UC Cálculo 1, Geometría y Álgebra Lineal 1, Programación 1 y Matemática Discreta 1 son las UC con mayor frecuencia de aparición (lo cual se condice con lo presentado en la figura 5.4.16 y 5.4.17). Luego, dentro de este mismo análisis podemos observar que solamente Geometría y Álgebra Lineal 1 y Matemática Discreta 1 presentan desviaciones en el modelo, en este caso contamos con un 33 % de desviación y menos del 0.1 % respectivamente.

Figura 5.4.21: Análisis de conformidad del log sobre el modelo generado.



En lo que respecta a la performance del modelo, el cual se puede ver en la figura 5.4.22, como información general tenemos que en promedio los estudiantes se desvinculan de la carrera en 20,16 meses, o lo que es lo mismo un poco mas de año y medio. En cuanto a UC con elevado tiempo de aprobación, que son Probabilidad y Estadística, Introducción a la Investigación de Operaciones, Programación 4, Métodos Numéricos, Fundamentos de Bases de Datos y Taller de Programación, en la tabla 5.4.8 se puede observar un detalle de los tiempos para cada caso. Igualmente cabe destacar que en este caso para las UC observadas se cuenta con pocos estudiantes que llegan a cursarlas.

Figura 5.4.22: Análisis de performance del log sobre el modelo generado.



Cuadro 5.4.8: Tiempos promedio de aprobación.

UC	**Tiempo promedio de aprobación (años)**
Probabilidad y Estadística	1.5
Taller de Programación	1.6
Métodos Numéricos	2
Fundamentos de Bases de Datos	2.4
Programación 4	2.9
Introducción a la Investigación de Operaciones	3

En este caso podemos observar nuevamente que a mayor avance de la carrera mayor es el tiempo requerido para la aprobación de las diferentes UC, en todos los casos se requiere por lo menos más de tres veces el tiempo mínimo de aprobación (aproximadamente 6 meses).

Por último, en la tabla 5.4.9, podemos observar las métricas de precisión y generalización del modelo. Para este caso contamos con valores superiores

comparados con el caso de estudiantes recibidos (Tabla 5.4.7), igualmente es posible concluir que este modelo no sobre ajusta al log con el cual se generó (precisión media) y el mismo es lo suficientemente general, lo que le permite reconocer nuevas trazas no presentes en el log (alta generalización).

Cuadro 5.4.9: Precisión y generalización del modelo generado.

Métrica	**Valor)**
Precesión	0.41671
Generalización	0.99775

5.4.4.3. Estudiantes avanzados en la carrera

Análisis del log: Para este caso contamos con los datos de 635 estudiantes (aproximadamente 18 % del total) que generan un total de 13718 eventos dentro del log (aproximadamente 26 % del total). En la figura 5.4.23 podemos observar que Geometría y Álgebra Lineal 1 es la UC que primero aprueban los estudiantes con casi un 55 % de los mismos.

Figura 5.4.23: Primeras UC aprobadas para estudiantes avanzados en la carrera.

Start events

Total number of classes: **5**

Class	Occurrences (absolute)	Occurrences (relative)
GEOMETRIA Y ALGEBRA LINEAL 1	349	54,961%
CALCULO 1	97	15,276%
PROGRAMACION 1	86	13,543%
FISICA 1	67	10,551%
MATEMATICA DISCRETA 1	36	5,669%

Dentro de la figura 5.4.24 podemos observar la distribución de las últimas UC aprobadas por parte de los estudiantes. Un dato interesante que se observa es que cerca del 21 % de los mismos aprueban como última UC el Proyecto de Grado, es decir que tienen aprobada la UC que se supone debería ser la última en la carrera según la currícula, pero no se encuentran egresados de la misma. Podemos destacar que para los estudiantes egresados como para los estudiantes avanzados las UC Proyecto de Grado y Redes de Computadoras se encuentran como la última y penúltima UC aprobadas respectivamente.

Figura 5.4.24: Últimas UC aprobadas para estudiantes avanzados en la carrera.

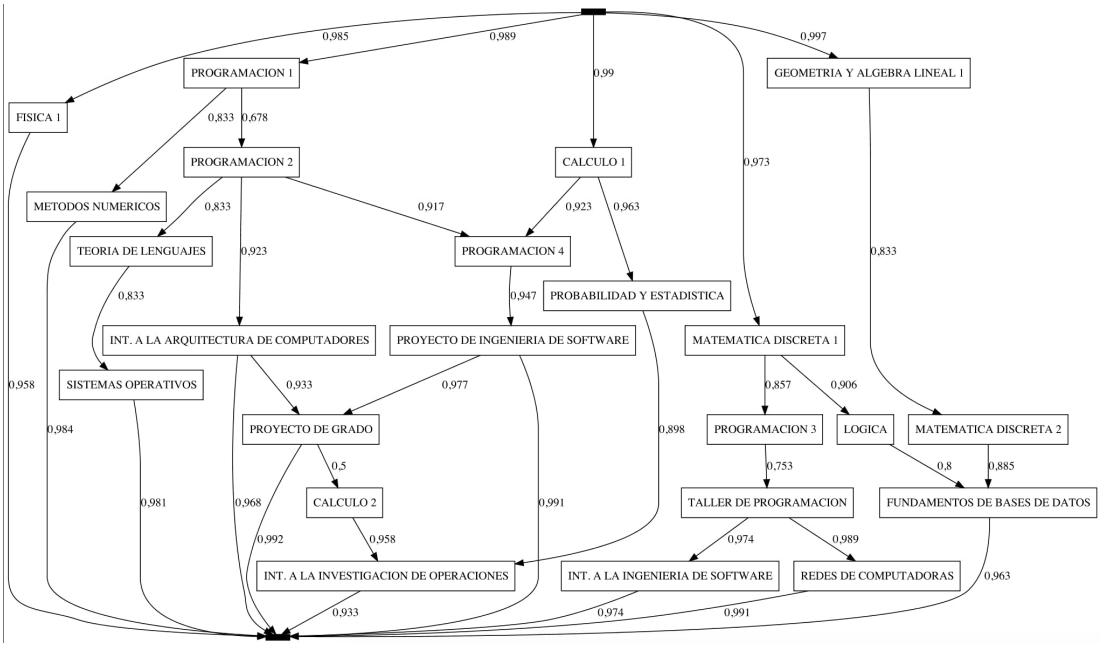
End events

Total number of classes: **15**

Class	Occurrences (absolute)	Occurrences (relative)
PROYECTO DE GRADO	132	20,787%
REDES DE COMPUTADORAS	116	18,268%
PROYECTO DE INGENIERIA DE SOFTWARE	111	17,48%
METODOS NUMERICOS	63	9,921%
SISTEMAS OPERATIVOS	53	8,346%
INT. A LA INGENIERIA DE SOFTWARE	38	5,984%
INT. A LA ARQUITECTURA DE COMPUTADORES	30	4,724%
FUNDAMENTOS DE BASES DE DATOS	26	4,094%
FISICA 1	23	3,622%
INT. A LA INVESTIGACION DE OPERACIONES	14	2,205%
CALCULO 2	8	1,26%
PROBABILIDAD Y ESTADISTICA	7	1,102%
TEORIA DE LENGUAJES	7	1,102%
TALLER DE PROGRAMACION	6	0,945%
PROGRAMACION 4	1	0,157%

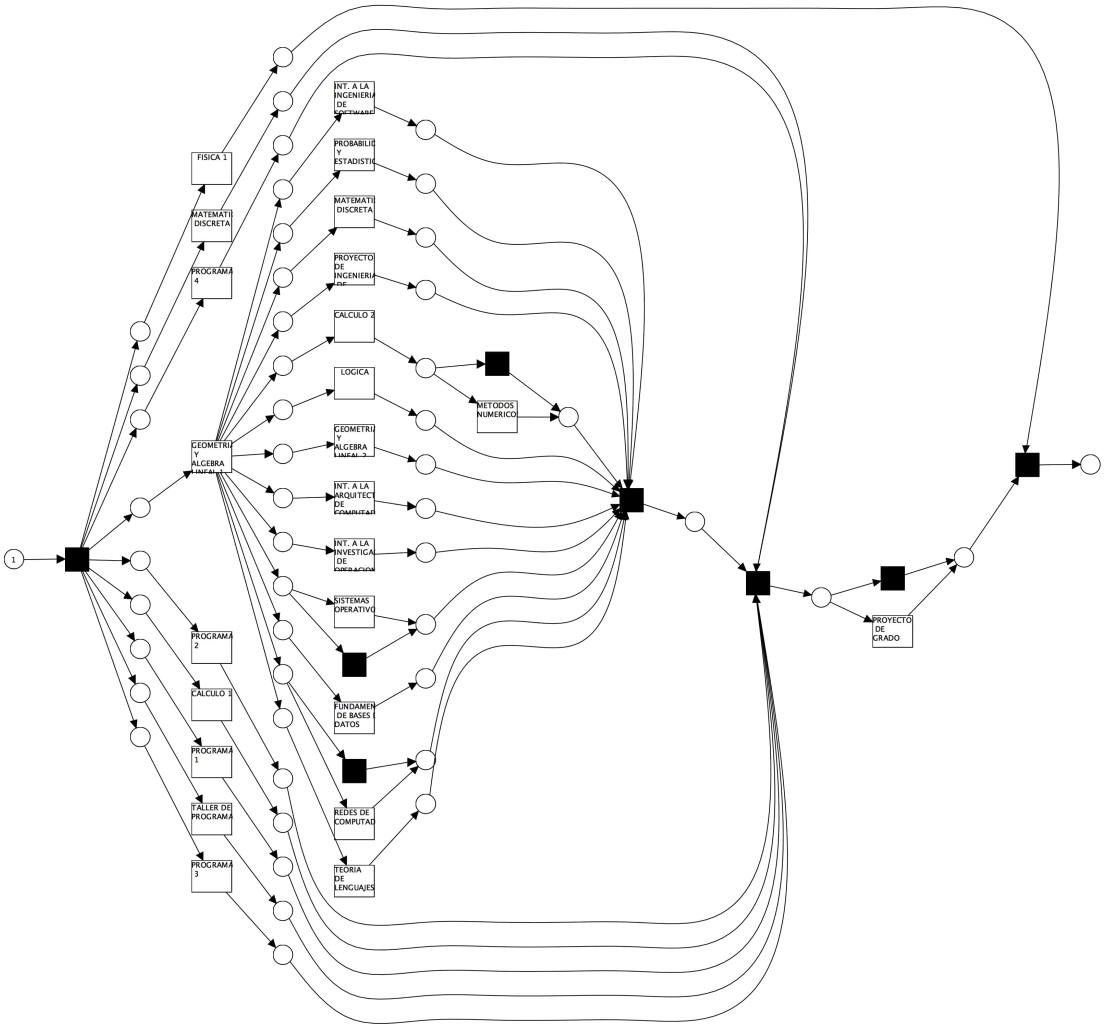
Modelo generado: En la figura 5.4.25 podemos observar el grafo de dependencias para este caso. En el mismo no se cuenta con la UC Geometría y Álgebra Lineal 2 debido a que para los parámetros utilizados no se cuenta con suficiente dependencia directa entre ninguna UC y ésta por lo cual no es visible. Con el objetivo de entender mejor este comportamiento dentro del grafo, se optó por ajustar el parámetro correspondiente al mínimo valor de dependencia necesaria para que una UC sea considerada por el algoritmo. Fue necesario configurar este valor en 0 para que dicha UC fuera visible dentro del grafo y se observó que la UC Geometría y Álgebra Lineal 1 es su única predecesora (con un valor de confianza por debajo de 0.6) y su única sucesora es Métodos Numéricos. Como último paso se procedió a verificar de qué UC es previa Geometría y Álgebra Lineal 2, donde se identificó que lo es solamente de Métodos Numéricos e Int. a la Investigación de Operaciones donde solo se observo una dependencia directa entre Geometría y Álgebra Lineal e Int. a la Investigación de Operaciones.

Figura 5.4.25: Grafo de dependencias para estudiantes avanzados en la carrera.



En la figura 5.4.26 se observa la red Petri luego de ejecutar el plug-in sobre el log. Luego, en la figura 5.4.27, se muestran los resultados de conformidad de la red sobre el log provisto. Aquí podemos ver que este modelo reproduce un 97 % de la traza, donde 12 de las 24 UC presentan desviaciones entre el log y el modelo. En este caso el promedio de desviación para estas UC es menos del 7 % en las trazas analizadas.

Figura 5.4.26: Red Petri para estudiantes avanzados en la carrera.



En cuanto a la información de performance sobre el modelo, en la figura 5.4.28 podemos observar el resultado de ejecutar el log sobre el modelo obtenido. Aquí podemos ver que en promedio los estudiantes llevan cursando hasta el momento 7.5 años, es decir 2.5 años mas que el que establece la carrera.

Figura 5.4.27: Análisis de conformidad para estudiantes avanzados en la carrera.

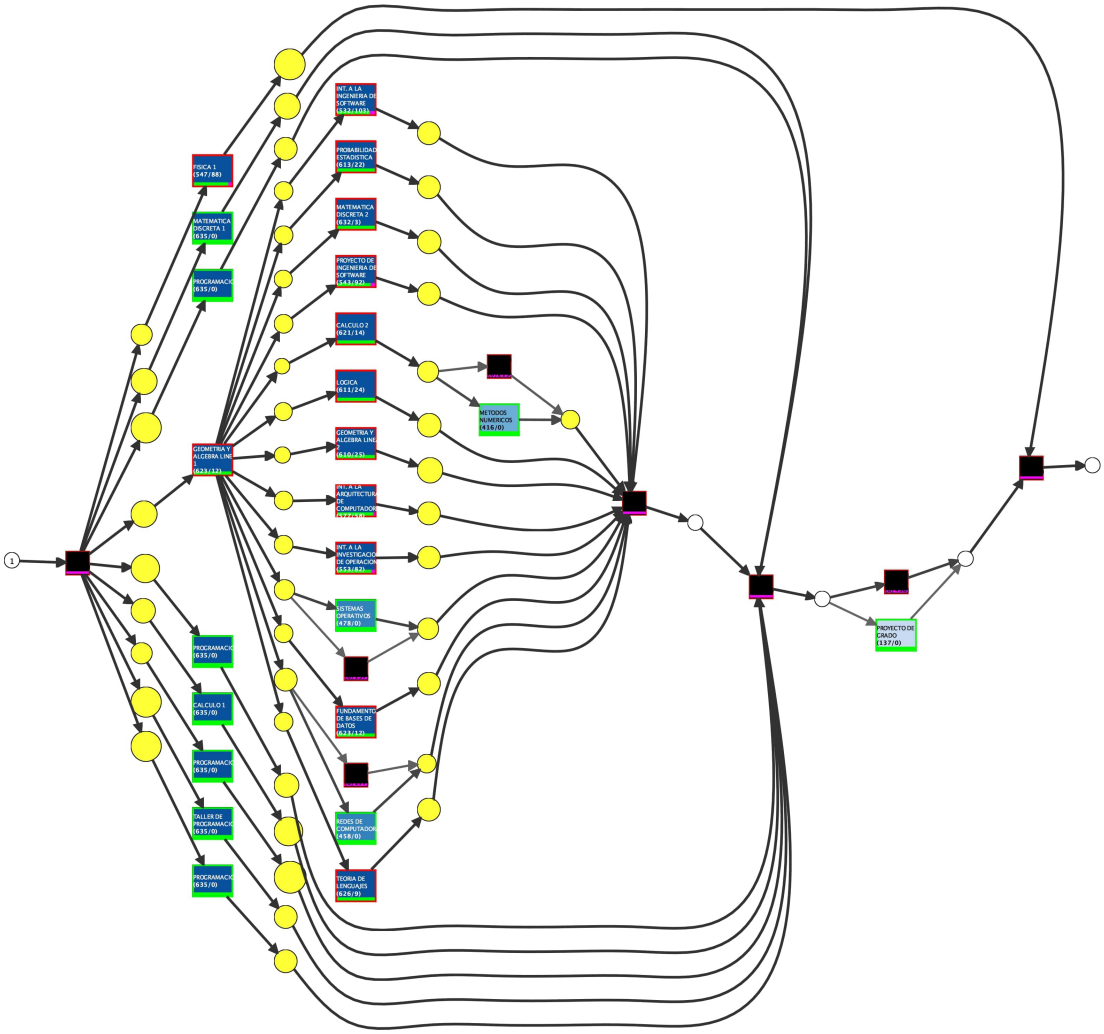
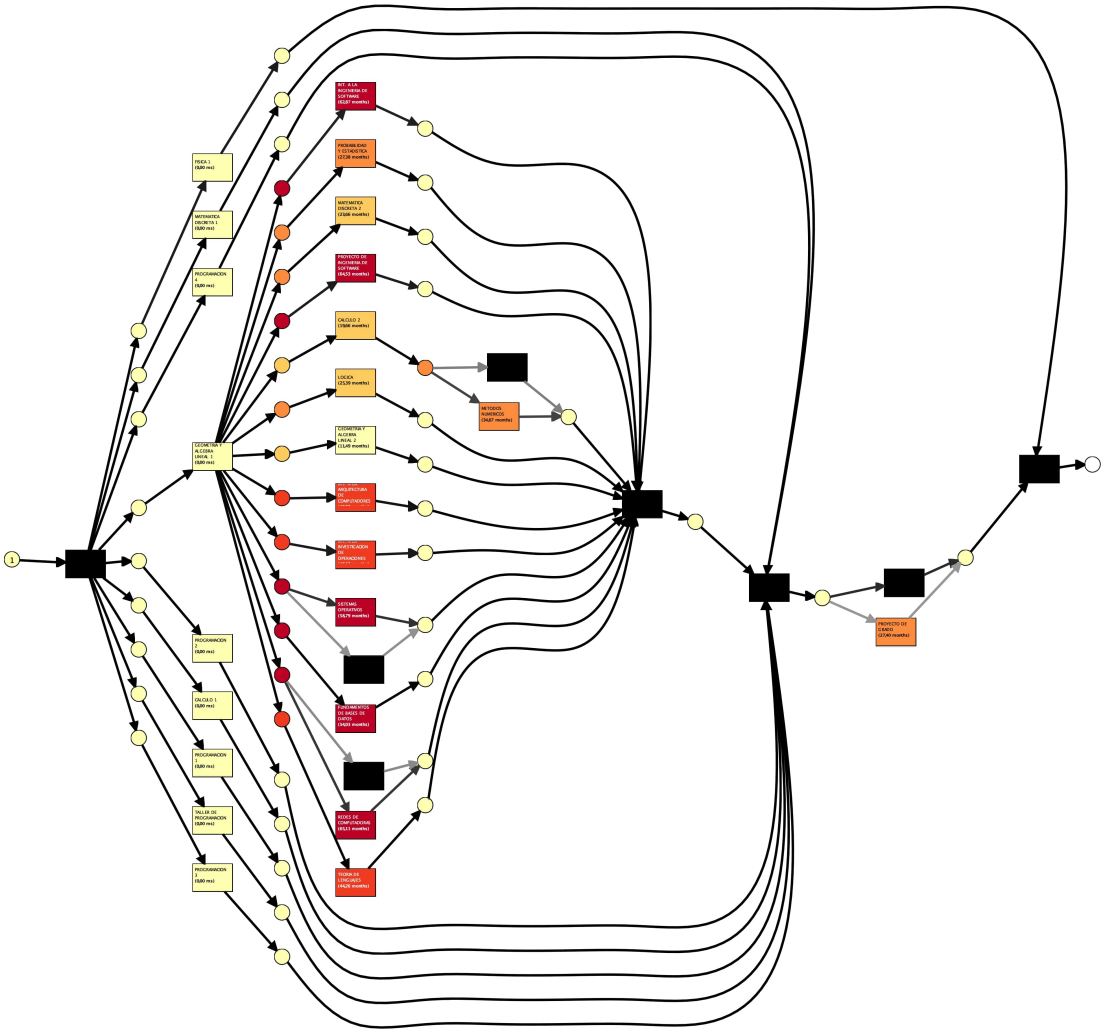


Figura 5.4.28: Análisis de performance para estudiantes avanzados en la carrera.



Indagando un poco mas en profundidad podemos observar 14 UC con altos tiempo de aprobación. En la tabla 5.4.10 podemos ver un detalle de las mismas.

Cuadro 5.4.10: Tiempos promedio de aprobación.

UC	**Tiempo promedio de aprobación (años)**
Cálculo 2	1.6
Matemática Discreta 2	1.9
Lógica	2.1
Probabilidad y Estadística	2.3
Proyecto de Grado	2.3
Métodos Numéricos	2.9

Cuadro 5.4.10: Tiempos promedio de aprobación.

UC	**Tiempo promedio de aprobación (años)**
Teoría de Lenguajes	3.7
Int. a la Investigación de Operaciones	3.8
Int. a la Arquitectura de Computadores	3.9
Fundamentos de Bases de Datos	4.5
Sistemas Operativos	4.9
Int. a la Ingeniería de Software	5.2
Proyecto de Ingeniería de Software	5.4
Redes De computadores	5.4

Aquí nuevamente observamos que a mayor avance dentro de la carrera el tiempo de aprobación de las diferentes UC es mayor, pero en este caso tenemos UC que requieren, en promedio, mas años que el establecido para la carrera en su totalidad, este es el caso de Int. a la Ingeniería de Software, Proyecto de Ingeniería de Software y Redes De computadores.

Por ultimo, en la tabla 5.4.11 podemos observar las métricas de precisión y generalización del modelo. Este caso es el que cuenta con las métricas menos relevantes de los tres, igualmente vemos que el modelo se adapta adecuadamente al log provisto sin sobre ajustar al mismo, permitiendo reconocer nuevo comportamiento no explorado por el log.

Cuadro 5.4.11: Precisión y generalización del modelo generado.

Métrica	**Valor)**
Precesión	0.18380
Generalización	0.65019

5.4.4.4. Conclusiones generales del modelado de trayectorias con herramientas de minería de procesos

La utilización de la herramienta de minería de procesos ProM permite realizar un análisis general de los datos con los que se cuenta, la generación de diferentes modelos y realizar diferentes análisis sobre los mismos. En nuestro caso particular

permitió obtener una idea general de cómo se comportan los estudiantes durante su transcurso por la carrera, identificando cuellos de botella con los cuales avalar la teoría de UC "difíciles" (este es el caso de las UC Sistemas Operativos, Redes de Computadoras y Arquitectura de Computadores) y los casos de UC que no bloquean el avance, por lo tanto, son dejadas para más adelante (Física 1 y Métodos Numéricos). Si consideramos el caso de los estudiantes desvinculados es posible corroborar que, tanto utilizando métodos estadísticos o utilizando ProM, la gran mayoría de los estudiantes abandonan la carrera sin siquiera completar el primer año y los que mas avanzan (sin egresar) optan por cursar UC relacionadas a las diferentes programaciones, llegando a completar las cuatro UC.

En cuanto a los modelos generados podemos concluir que los mismos describen de forma adecuada la realidad (por encima del 97 % de fitness de las trazas del log), no sobre ajustan al log provisto y permiten reconocer otras trayectorias no presentes (una precisión media y alta generalización de los modelos) sin llegar a el caso de modelos de estrella (los cuales se adaptan a todos los logs pero no son correctos descriptores de la realidad).

Cómo idea general de como se comportan los estudiantes en la carrera, esta primera aproximación sirve como puntapié inicial para futuros análisis tanto de anomalías en las trayectorias como en los tiempos promedio y de allí poder determinar posibles accionables sobre las UC claves que terminan extendiendo más de lo estipulado la carrera de un estudiante.

En cuanto al trabajo a futuro es posible:

- Realizar un análisis mas profundo de la base de datos provista con el objetivo de entender las diferentes heurísticas que hay respecto a aprobaciones automáticas, revalidas y demás para poder incorporar esta información.
- No solo considerar un conjunto reducido de UC sino la totalidad de las mismas.
- Es posible realizar el mismo análisis para diferentes carreras para poder detectar anomalías.
- ProM cuenta con una amplia variedad de plug-ins destinados para diferentes tipos de análisis sobre los datos, donde se podría profundizar más la investigación de los mismos para obtener otro tipo de resultados.
- Contar con mas información mas allá de la provista por Bedelías e incorporarla a ProM para determinar otro tipo de comportamiento ligado

a otras variables (sexo del estudiante, edad, información laboral, ingresos, información tanto de primaria o secundaria), ya que es posible determinar comportamiento condicional a diferentes variables.

- Realizar este mismo análisis segmentado a semestres o años, para determinar si particularmente en alguno de ellos se están cursando las UC sugeridas o si hay comportamientos diferentes.

Capítulo 6

Conclusiones y trabajo a futuro

El desarrollo del proyecto, implementado en tres grandes segmentos, permitió alcanzar los objetivos propuestos:

- Automatización y generación de dashboards con los informes generados por la UEFI.
- Análisis cuantitativo y cualitativo de los datos de las fuentes proporcionadas.
- Utilización de los datos para la generación de modelos de Machine Learning y Process Mining.

Como primer punto se diseñó e implementó una primera aproximación de un Data Warehouse que aloja los datos relevantes para la UEFI y herramientas que permitan visualizar la información que contiene el mismo. Dichas herramientas quedan disponibles para la UEFI, con el objetivo de facilitar la elaboración de los informes que realizan de forma periódica, y además a futuro ayuden a la elaboración de nuevos informes con un costo muy bajo y la extensión de los existentes con nuevos indicadores más complejos.

Se formuló una descripción de la población universitaria de la FING, en donde se caracterizaron las variables de sexo, extraedad, el estrato social calculado, el tipo de institución previa al comienzo universitario y los desempeños académicos en 6to año de liceo y el primer semestre al ingresar a la carrera.

Finalmente, se realizaron esfuerzos para intentar predecir el abandono de los estudiantes de la Carrera de Ingeniería en Computación con los resultados arrojados por el análisis anterior, para esto se utilizaron técnicas de data mining con las que se pudo generar modelos con altos niveles de precisión, en donde se explica que los

factores sociales y los primeros resultados en la vida universitaria del estudiante, son determinantes para su posterior continuidad.

También se realizaron ejercicios para identificar mediante minería de procesos cómo se comportan las trayectorias de los estudiantes, cuáles son las unidades curriculares donde abandonan, cuales son las asignaturas cuello de botella, entre otros. De este análisis fue posible además contrastar dichos resultados con resultados anteriores (análisis previos vía métodos descriptivos e informes de la UEFI) llegando a similares resultados (materias cuello de botella y verificar desvinculación antes del primer año de carrera).

Asimismo fue posible generar modelos de procesos asociados a los diferentes casos analizados (estudiantes egresados, desvinculados y actualmente cursando) y realizar diferentes análisis sobre los mismos para compararlos con la realidad, llegando a modelos que representan con gran nivel de precisión los datos provistos.

Como recomendaciones, este grupo de trabajo sugiere:

- Iterar el Data Warehouse con el fin de seguir sumando nuevos informes, además de realizar las adecuaciones necesarias para disponibilizarlo en una infraestructura productiva, con el fin de que pueda ser accedido desde distintos dispositivos.
- Ahondar e iterar sobre la utilización de minería de procesos con el objetivo de estandarizar la obtención de información para luego ser procesada y obtener información relevante de forma simple y rápida, sumando otros elementos que en nuestro análisis no fueron considerados (extender hacia otras carreras, exámenes, unidades curriculares reprobadas, etc) para refinar los resultados.
- Comenzar a generar una base de datos transversal en toda la FING, donde se registre un histórico de las pruebas diagnósticas iniciales, las actividades realizadas en EVA, como así también los resultados de parciales y otras evaluaciones.
- Continuar con el uso de técnicas de Data Science para comprender de mejor manera las causas de los resultados académicos en los estudiantes.
- Establecer acuerdos y mecanismos entre la ANEP y la FING para el intercambio de información oportuna. Este intercambio, debe de poder acompañar el diagnóstico y la atención a la singularidad de los aprendizajes.
- Evolucionar la herramienta confeccionada en este proyecto y disponibilizarla

a los directores de carrera en estado productivo.

- Generar dispositivos que permitan acompañar las trayectorias educativas de los estudiantes.
- Realizar las adaptaciones necesarias en los productos emanados de este proyecto para que puedan consumir información del nuevo sistema de bedelías de la UDELAR (SGAE).

Bibliografía

[1] 14 (2018-2020) educación media. facultad de ingeniería, aprendizajes y equidad.pdf. URL https://www.fing.edu.uy/sites/default/files/claustro_citaciones/2019/distribuido/36753/14%20%282018-2020%29%20Educaci%C3%B3n%20Media.%20Facultad%20de%20Ingenier%C3%ADa%2C%20aprendizajes%20y%20equidad.pdf.

[2] Databases using r - dbi. URL https://db.rstudio.com/dbi/. (Accessed on 03/06/2021).

[3] Google geocoding API. URL https://developers.google.com/maps/documentation/geocoding/intro.

[4] Create elegant data visualisations using the grammar of graphics • ggplot2. URL https://ggplot2.tidyverse.org/index.html. (Accessed on 03/06/2021).

[5] Graduarse: solo la mitad lo logra en américa latina. URL https://www.bancomundial.org/es/news/feature/2017/05/17/graduating-only-half-of-latin-american-students-manage-to-do-so.

[6] Ineed - aristas 2018. informe de resultados de tercero de educación media. URL https://www.ineed.edu.uy/aristas-2018-informe-de-resultados-de-tercero-de-educacion-media.html.

[7] Visualización de datos | microsoft power bi. URL https://powerbi.microsoft.com/es-es/. (Accessed on 03/06/2021).

[8] Pentaho BI Server. URL https://wiki.pentaho.com/display/ServerDoc2x. (Accessed on 03/06/2021).

[9] Postgis — documentation. URL https://postgis.net/documentation/.

[10] Rstudio | open source & professional software for data science teams - rstudio. URL https://rstudio.com/. (Accessed on 03/06/2021).

[11] R: What is r? URL https://www.r-project.org/about.html.

[12] Shiny. URL https://shiny.rstudio.com/. (Accessed on 03/06/2021).

[13] Business intelligence and analytics software. URL https://www.tableau.com. (Accessed on 03/06/2021).

[14] The caret package. URL http://topepo.github.io/caret/index.html.

[15] Python programming language. URL https://docs.python.org/3/.

[16] QGIS: Software libre para sistemas de información geográfica. URL https://qgis.org/es/docs/index.html.

[17] Csv, comma separated values file. https://tools.ietf.org/html/rfc4180#section-2, 2020.

[18] Data integration - kettle. https://community.hitachivantara.com/s/article/data-integration-kettle, 2020.

[19] PostgreSQL, 2020. URL https://www.postgresql.org/docs/11/index.html.

[20] Prom tools home page. http://www.promtools.org/doku.php?id=start, 2020.

[21] SQLite: C-language library that implements a database engine, 2020. URL https://www.sqlite.org/index.html.

[22] Trayectoria sugerida para la carrera en ingeniería en computación, plan 97. https://www.fing.edu.uy/carreras/grado/computacion/implementacion/archivos/TrayectoriaSugerida.pdf, 2020.

[23] Xes, extensible event stream. http://xes-standard.org/, 2020.

[24] Recursos fuente, 2021. URL https://drive.google.com/drive/folders/1-jHXLrxBa6POv18ZX6wMDII2OmpQ6b_X?usp=sharing.

[25] R. C. Alejandro Bogarín1 and C. Romero. Discovering learning processes using inductive miner: A case study with learning management systems (lmss). 2018.

[26] N. Baumgart and J. Johnstone. Desempeño estudiantil en fing.

[27] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[28] M. Bucheli and C. Casacuberta. Asistencia a instituciones educativas y actividad laboral de los adolescentes en uruguay, 1986-2008. *La desafiliación en la Educación Media y Superior en Uruguay. Conceptos, estudios y políticas. Montevideo: UDELAR-CSIC*, 2010.

[29] J. Campbell, P. DeBlois, and D. Oblinger. Academic analytics: A new tool for a new era. *EDUCAUSE Review*, 42, 01 2007.

[30] F. Carpani. CMDM: Un modelo conceptual para la especificación de bases de datos multidimensionales. Master's thesis, InCo - Pedeciba, Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay, October 2000.

[31] M. A. Chatti, A. L. Dyckhoff, U. Schroeder, and H. Thüs. A reference model for learning analytics. *International Journal of Technology Enhanced Learning*, 4(5-6):318–331, 2013.

[32] R. Christensen. Thoughts on prediction and cross-validation. *Department of Mathematics and Statistics University of New Mexico*, 2015.

[33] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3): 273–297, 1995.

[34] B. I. de Desarrollo. Del papel a la nube: Cómo guiar la transformación digital de los sistemas de información y gestión educativa (siged). *Recuperado de: https://publications. iadb. org/es/del-papel-la-nube-como-guiar-la-transformacion-digital-de-los-sistemas-de-informacion-y-gestion*, 2019.

[35] P. Díaz Charquero, M. Jackson, and R. Motz. Learning analytics y protección de datos personales. recomendaciones. page 981, 10 2015. doi: 10.5753/cbie.wcbie.2015.981.

[36] R. Ferguson. Learning analytics: drivers, developments and challenges. *International Journal of Technology Enhanced Learning*, 4(5/6):304–317, 2012.

[37] J. García, J. Molina, A. Berlanga, M. Patricio, A. Bustamante, and W. Padilla. Ciencia de datos. *Técnicas Analíticas y Aprendizaje Estadístico. Bogotá, Colombia. Publicaciones Altaria, SL*, 2018.

[38] I. Jara and J. Ochoa. Usos y efectos de la inteligencia artificial en educación. *Sector Social división educación. Documento para discusión número IDB-DP-00-776. BID. doi: http://dx. doi. org/10.18235/000238 0*, 2020.

[39] H. K. Kanagala and V. V. Jaya Rama Krishnaiah. A comparative study of k-means, dbscan and optics. In *2016 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1–6, 2016. doi: 10.1109/ICCCI.2016.7479923.

[40] W.-Y. Loh. Classification and regression trees. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, 1:14–23, 2011.

[41] F. Mannhardt, M. de Leoni, and H. A. Reijers. Heuristic mining revamped: An interactive, data-aware, and conformance-aware miner. In *BPM (Demos)*, 2017.

[42] T. Murata. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580, 1989. doi: 10.1109/5.24143.

[43] M. Narvekar and S. F. Syed. An optimized algorithm for association rule mining using fp tree. *Procedia Computer Science*, 45:101–110, 2015.

[44] A. Pardo and G. Siemens. Ethical and privacy principles for learning analytics. *British Journal of Educational Technology*, 45(3):438–450, 2014.

[45] D. M. Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. 2011.

[46] A. Retamoso and R. Kaztman. Segregación espacial, empleo y pobreza en montevideo. *Revista de la CEPAL*, 2005.

[47] C. Romero, S. Ventura, M. Pechenizkiy, and R. S. Baker. *Handbook of educational data mining*. CRC press, 2010.

[48] P. Siemens, George y Long. Penetrating the fog: Analytics in learning and education. *EDUCAUSE review*, 46(5):30, 2011.

[49] E. Vaisman, Alejandro y Zimanyi. *Data warehouse systems: design and implementation*. Springer-Verlag, 2014.

[50] A. Verikas, E. Vaiciukynas, A. Gelzinis, J. Parker, and M. C. Olsson. Electromyographic patterns during golf swing: Activation sequence profiling and prediction of shot effectiveness. *Sensors*, 16:592, 04 2016. doi: 10.3390/s16040592.

[51] A. Wasilewska. Apriori algorithm. *Lecture Notes, http://www. cs. sunysb. edu/˜ cse634/lecture_ notes/07apriori. pdf, accessed*, 10, 2007.

[52] A. J. M. M. Weijters and J. T. S. Ribeiro. Flexible heuristics miner (fhm). *2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*, pages 310–317, 2011.

[53] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J.

McLachlan, A. Ng, B. Liu, S. Y. Philip, et al. Top 10 algorithms in data mining. *Knowledge and information systems*, 14(1):1–37, 2008.

Apéndice A

Diseño completo del Data Warehouse

En este apéndice se describe el proceso completo seguido para el diseño del Data Warehouse. Como se mencionó anteriormente esta etapa se puede dividir en cuatro partes: Especificación de requerimientos, Diseño conceptual, Diseño lógico y Diseño físico, donde abordaremos cada uno en una sección.

A1. Especificación de requerimientos

A partir de los informes presentados en la sección 4.1 y de los nuevos indicadores de la sección 4.2 se presentan en esta sección los requerimientos funcionales y no funcionales necesarios para satisfacer las distintas necesidades y que darán forma al Data Warehouse.

Requerimiento 1: Ingresos

Origen: Informe ingresos.

Se requiere analizar la información de los ingresos de los estudiante a la Facultad por generación, carrera y distintas características de los estudiantes. Particularmente se requiere saber la cantidad y el porcentaje sobre el total al agrupar los datos según los distintos criterios.

Esta información debe ser analizada por generación, carrera y los estudiantes.

Los estudiantes deben poder agruparse por sexo, edad ingreso, tipo de institución de procedencia, procedencia geográfica o carrera a la que entraron.

Se toma como generación de ingreso del estudiante como el año que se inscribió por primera vez a una carrera de la facultad.

Requerimiento 2: Egresos

Origen: Informe egresos, egresos por carrera e Informe de Indicadores de Seguimiento del Plan Estudios, cantidad de títulos expedidos por año.

Se requiere analizar la información de los egresos de los estudiante a la Facultad por generación, carrera y distintas características de los estudiantes. Particularmente se requiere saber la cantidad de egresos al agrupar los datos según los distintos criterios y la cantidad de egresados por año.

Esta información debe ser analizada por generación, carrera y los estudiantes.

Los criterios de agrupación de los estudiantes son los mismos que en el requerimiento 1.

Requerimiento 3 : Estudiantes activos

Origen: Informe egresos, eficiencia de titulación real.

Se requiere analizar la información de los estudiantes activos por generación, carrera y distintas características de los estudiantes. Particularmente se requiere saber la cantidad de activos al agrupar los datos según los distintos criterios.

Esta información debe ser analizada por generación, carrera y los estudiantes.

Los criterios de agrupación de los estudiantes son los mismos que en el requerimiento 1.

Al cruzar estos datos con los de los egresados se va a poder calcular la eficiencia de titulación real.

Requerimiento 4: Puntos críticos

Origen: Informe de puntos críticos.

Se requiere analizar el resultado de las actividades que se realizan para las Unidades Curriculares a lo largo del tiempo, donde las actividades se dividen en Cursos y Exámenes. Para ambos tipos de actividades interesa saber el porcentaje de aprobación de la actividad según los distintos criterios.

Esta información debe ser analizada por unidad curricular y tiempo.

El tiempo en las actividades de tipo curso se identifican por el semestre y el año en que fue dictada, y se requieren agrupar por año o por el semestre global (todos los primeros semestres o todos los segundos semestres).

El tiempo en las actividades de tipo examen se identifican por el mes y el año en que fue rendida, y se requieren agrupar por año o por el mes global (por ejemplo todos los Febrero).

Requerimiento 5: Duración carreras

Origen: Informe duración de carreras.

Se requiere analizar la información de la cantidad tiempo que le lleva a los estudiantes llegar a la mitad y al total de los créditos que exige la carrera. Para ambos casos interesa saber el promedio.

Esta información debe ser analizada por generación, carrera y los estudiantes.

Los criterios de agrupación de los estudiantes son los mismos que en el requerimiento 1.

Requerimiento 6: Avance por franja de créditos

Origen: Informe de Indicadores de Seguimiento del Plan Estudios, avance por franja de créditos.

Se quiere analizar información acerca del avance de los estudiantes en cada carrera, donde las carreras tengan 450 créditos. Interesa visualizar el total de estudiantes activos que hay en cada franja de créditos.

Los créditos se dividen en franjas equiespaciadas cada 45 créditos, a excepción de la franja inicial que solo abarca al cero.

Esta información debe ser analizada por generación, carrera y los estudiantes.

Los criterios de agrupación de los estudiantes son los mismos que en el requerimiento 1.

Requerimiento 7: Distribución estudiantes por unidad curricular

Origen: Distribución de estudiantes por unidad curricular.

Se requiere analizar información acerca del resultado de las ediciones de las unidades curriculares. Particularmente interesa en qué proporción se distribuyen los estudiantes según si reprobaron, se les venció el curso, están cursando, salvaron el examen, exoneraron o recursaron en una edición posterior. Este indicador se quiere ver como el porcentaje de estudiantes en cada categoría respecto al total de inscriptos en esa edición de la unidad curricular al agrupar por los distintos criterios.

El indicador antes mencionado se requiere visualizar por unidad curricular, tiempo de ediciones de cursos, carrera, generación y estudiantes.

Los criterios de agrupación del tiempo de ediciones de cursos son los mismos que en el requerimiento 4.

Los criterios de agrupación de los estudiantes son los mismos que en el requerimiento 1.

Requerimiento 8: Tiempo que lleva salvar una unidad curricular

Origen: Tiempo que lleva salvar una unidad curricular.

Se requiere analizar información acerca del tiempo que tardan los estudiantes en aprobar las unidades curriculares. Particularmente se quiere saber el tiempo promedio al agrupar por los distintos criterios.

El indicador antes mencionado se requiere visualizar por unidad curricular, tiempo de ediciones de cursos (la primera vez que el estudiante cursó la unidad curricular), carrera y estudiantes.

Requerimiento no funcional 1

En el informe de ingresos solo se toman como ingreso a aquellos estudiantes que hayan ingresado anteriormente a ninguna otra carrera. Por lo tanto, un requerimiento es tener la posibilidad de filtrar aquellos alumnos que su primer carrera no es la que se va a analizar.

Requerimiento no funcional 2

Para que la visualización de los datos sea más fácil e intuitiva se requiere poder ordenar los parámetros por los cuales se miden los indicadores en los casos donde el orden no se desprende implícitamente.

Con la especificación de requerimientos finalizada se procede a realizar la siguiente etapa del diseño, que consiste en el diseño conceptual.

A2. Diseño conceptual

La fase de Diseño Conceptual tiene como objetivo construir una representación abstracta de la base de datos para que sea entendible por el usuario a partir de los requerimientos, sin entrar en detalles de como va a ser implementado. En esta etapa se construye un modelo conceptual que permita ver los aspectos relevantes del análisis multidimensional, identificándose las dimensiones con sus jerarquías, y los hechos con sus medidas [49]. Para ello se va a utilizar el modelo conceptual *CMDM* [30], el cual permite modelar gráficamente base de datos multidimensionales y un lenguaje de restricciones de integridad que permite dar una descripción precisa de las relaciones entre los datos.

A continuación se presentarán los diagramas de las dimensiones y las relaciones dimensionales, junto con sus respectivas tablas de aditividad para cada requerimiento.

Requerimiento 1: Ingresos

Figura A2.1: Dimensiones de Estudiantes, Carreras y Generaciones.

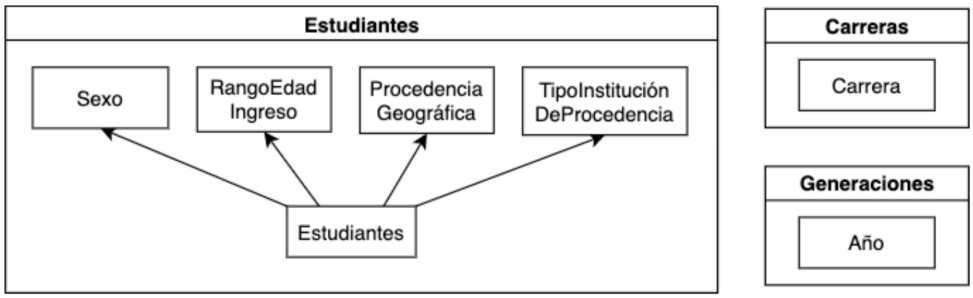
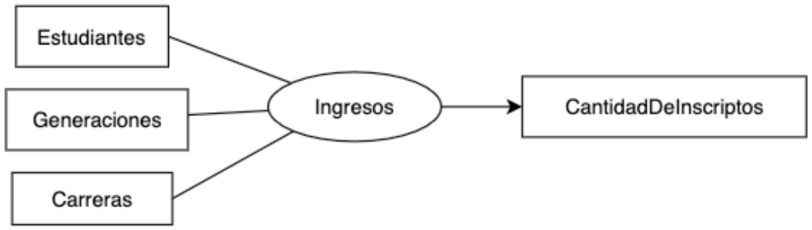


Figura A2.2: Relación dimensional Ingresos.

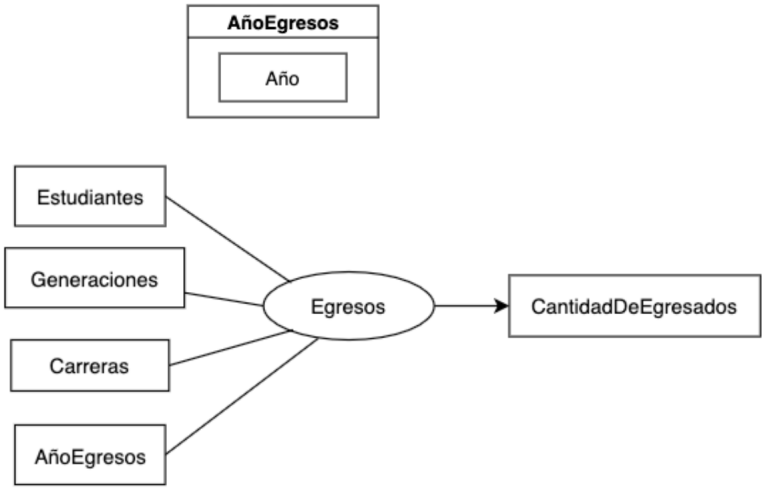


Cuadro A2.1: Tabla de aditividad de la relación dimensional Ingresos.

		CantidadDeInscriptos
Personas	Persona ->Sexo	Sum
	Sexo ->ALL	Sum
	Persona ->RangoEdadIngreso	Sum
	RangoEdadIngreso ->ALL	Sum
	Persona ->ProcedenciaGeográfica	Sum
	ProcedenciaGeográfica ->ALL	Sum
	Persona ->TipoInstituciónDeProcedencia	Sum
	TipoInstituciónDeProcedencia ->ALL	Sum
Generaciones	Año ->ALL	Sum
Carreras	Carrera ->ALL	Sum

Requerimiento 2: Egresos

Figura A2.3: Dimensión AñoEgresos y relación dimensional Egresos.

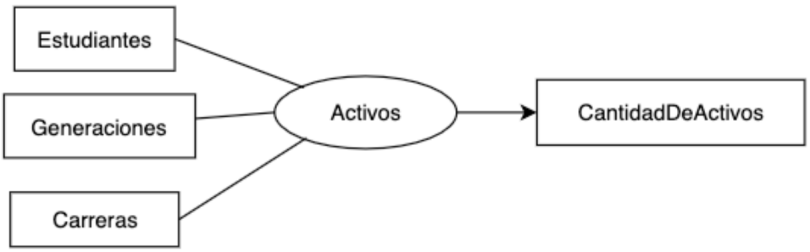


Cuadro A2.2: Tabla de aditividad de la relación dimensional Egresos.

		CantidadDeEgresados
Personas	Persona ->Sexo	Sum
	Sexo ->ALL	Sum
	Persona ->RangoEdadIngreso	Sum
	RangoEdadIngreso ->ALL	Sum
	Persona ->ProcedenciaGeográfica	Sum
	ProcedenciaGeográfica ->ALL	Sum
	Persona ->TipoInstituciónDeProcedencia	Sum
	TipoInstituciónDeProcedencia ->ALL	Sum
Generaciones	Año ->ALL	Sum
Carreras	Carrera ->ALL	Sum
AñoEgresos	Año ->ALL	Sum

Requerimiento 3 : Estudiantes activos

Figura A2.4: Relación dimensional Activos.



Cuadro A2.3: Tabla de aditividad de la relación dimensional Activos.

		CantidadDeActivos
Personas	Persona ->Sexo	Sum
	Sexo ->ALL	Sum
	Persona ->RangoEdadIngreso	Sum
	RangoEdadIngreso ->ALL	Sum
	Persona ->ProcedenciaGeográfica	Sum
	ProcedenciaGeográfica ->ALL	Sum
	Persona ->TipoInstituciónDeProcedencia	Sum
	TipoInstituciónDeProcedencia ->ALL	Sum
Generaciones	Año ->ALL	Sum
Carreras	Carrera ->ALL	Sum

Requerimiento 4: Puntos críticos

Figura A2.5: Dimensiones de UnidadesCurriculares, TiempoCursos y TiempoExamenes.

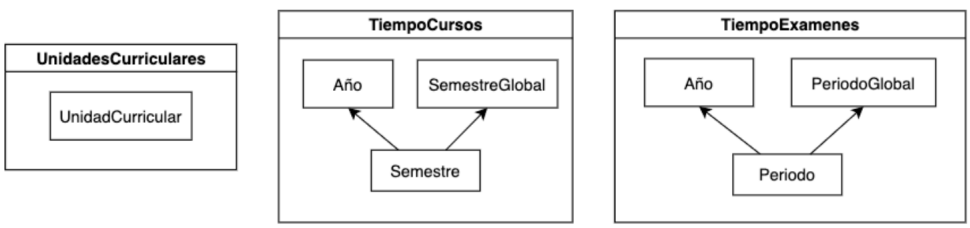


Figura A2.6: Relación dimensional PuntosCriticosCursos.

Cuadro A2.4: Tabla de aditividad de la relación dimensional PuntosCriticosCursos.

		Cantidad Aprobaron	Cantidad Inscriptos
UnidadesCurriculares	UnidadCurricular ->ALL	Sum	Sum
TiempoCursos	Semestre ->Año	Sum	Sum
	Año ->ALL	Sum	Sum
	Semestre ->SemestreGlobal	Sum	Sum
	SemestreGlobal ->ALL	Sum	Sum

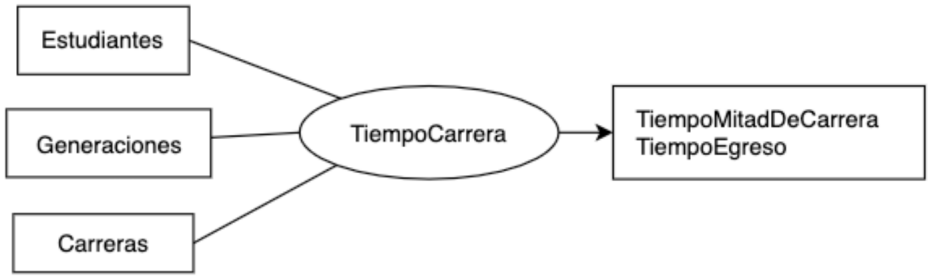
Figura A2.7: Relación dimensional PuntosCriticosExamenes.

Cuadro A2.5: Tabla de aditividad de la relación dimensional PuntosCriticosExamenes.

		Cantidad Aprobaron	Cantidad Inscriptos
UnidadesCurriculares	UnidadCurricular ->ALL	Sum	Sum
TiempoExamenes	Periodo ->Año	Sum	Sum
	Año ->ALL	Sum	Sum
	Periodo ->PeriodoGlobal	Sum	Sum
	PeriodoGlobal ->ALL	Sum	Sum

Requerimiento 5: Duración carreras

Figura A2.8: Relación dimensional TiempoCarrera.

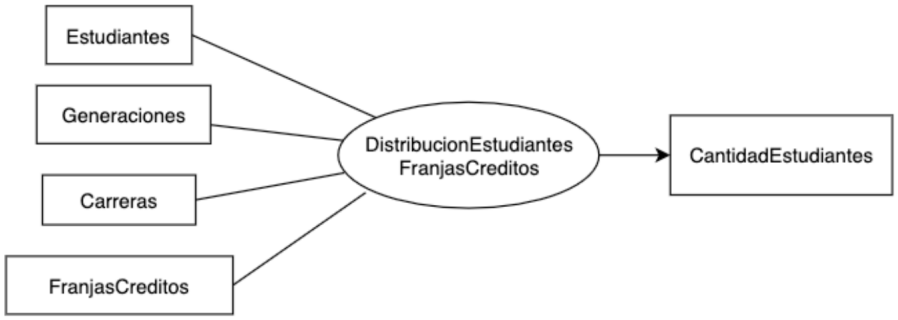


Cuadro A2.6: Tabla de aditividad de la relación dimensional TiempoCarrera.

		TiempoMitad DeCarrera	TiempoEgreso
Personas	Persona ->Sexo	Prom	Prom
	Sexo ->ALL	Prom	Prom
	Persona ->RangoEdadIngreso	Prom	Prom
	RangoEdadIngreso ->ALL	Prom	Prom
	Persona ->ProcedenciaGeográfica	Prom	Prom
	ProcedenciaGeográfica ->ALL	Prom	Prom
	Persona ->TipoInstituciónDeProcedencia	Prom	Prom
	TipoInstituciónDeProcedencia ->ALL	Prom	Prom
Generaciones	Año ->ALL	Prom	Prom
Carreras	Carrera ->ALL	Prom	Prom

Requerimiento 6: Avance por franja de créditos

Figura A2.9: Relación dimensional DistribucionEstudiantesFranjasCreditos.



Cuadro A2.7: Tabla de aditividad de la relación dimensional DistribucionEstudiantesFranjasCreditos.

		Cantidad Estudiantes
Personas	Persona ->Sexo	Sum
	Sexo ->ALL	Sum
	Persona ->RangoEdadIngreso	Sum
	RangoEdadIngreso ->ALL	Sum
	Persona ->ProcedenciaGeográfica	Sum
	ProcedenciaGeográfica ->ALL	Sum
	Persona ->TipoInstituciónDeProcedencia	Sum
	TipoInstituciónDeProcedencia ->ALL	Sum
Generaciones	Año ->ALL	Sum
Carreras	Carrera ->ALL	Sum
FranjasCreditos	Franja ->ALL	Sum

Requerimiento 7: Distribución estudiantes por unidad curricular

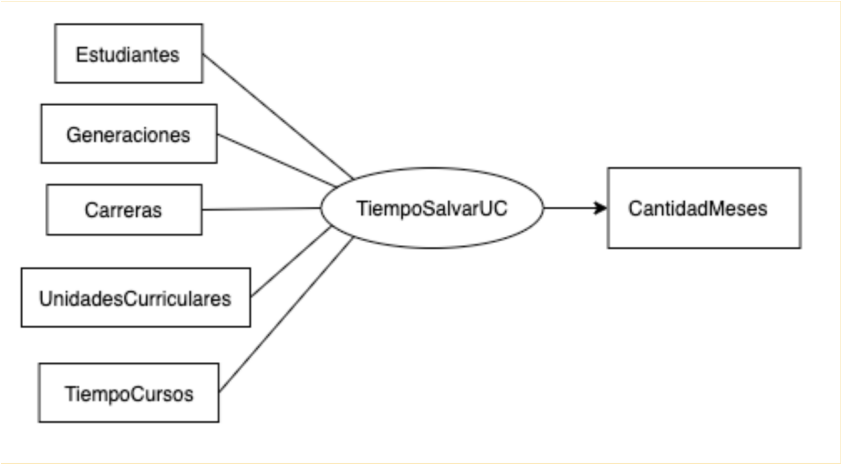
Cuadro A2.8: Tabla de aditividad de la relación dimensional DistribucionEstudiantesUC.

		Cant Reprobaron	Cant Vencidos	Cant Cursando	Cant Exonerados	Cant Salvaron Examen	Cant Recursaron	Cant Inscriptos
Personas	Persona -> Sexo	Sum	Sum	Sum	Sum	Sum	Sum	Sum
	Sexo -> ALL	Sum	Sum	Sum	Sum	Sum	Sum	Sum
	Persona -> RangoEdad Ingreso	Sum	Sum	Sum	Sum	Sum	Sum	Sum
	RangoEdad Ingreso ->ALL	Sum	Sum	Sum	Sum	Sum	Sum	Sum
	Persona -> Procedencia Geográfica	Sum	Sum	Sum	Sum	Sum	Sum	Sum
	Procedencia Geográfica ->ALL	Sum	Sum	Sum	Sum	Sum	Sum	Sum
	Persona -> Tipo Institución DeProcedencia	Sum	Sum	Sum	Sum	Sum	Sum	Sum
	TipoInstitucióDe Procedencia ->ALL	Sum	Sum	Sum	Sum	Sum	Sum	Sum
Generaciones	Año ->ALL	Sum	Sum	Sum	Sum	Sum	Sum	Sum
Carreras	Carrera ->ALL	Sum	Sum	Sum	Sum	Sum	Sum	Sum
Tiempo Cursos	Semestre ->Año	Sum	Sum	Sum	Sum	Sum	Sum	Sum
	Año ->ALL	Sum	Sum	Sum	Sum	Sum	Sum	Sum

	Semestre -> Semestre Global	Sum	Sum	Sum	Sum	Sum	Sum	Sum
	Semestre Global ->ALL	Sum	Sum	Sum	Sum	Sum	Sum	Sum
Unidades Curricu-lares	Unidad Curricular ->ALL	Sum	Sum	Sum	Sum	Sum	Sum	Sum

Requerimiento 8: Tiempo que lleva salvar una unidad curricular

Figura A2.10: Relación dimensional TiempoSalvarUC.



Cuadro A2.9: Tabla de aditividad de la relación dimensional TiempoSalvarUC.

		CantMeses
Personas	Persona ->Sexo	Prom, Mediana
	Sexo ->ALL	Prom, Mediana
	Persona ->RangoEdadIngreso	Prom, Mediana
	RangoEdadIngreso ->ALL	Prom, Mediana
	Persona ->ProcedenciaGeográfica	Prom, Mediana
	ProcedenciaGeográfica ->ALL	Prom, Mediana
	Persona -> TipoInstituciónDeProcedencia	Prom, Mediana
	TipoInstituciónDeProcedencia ->ALL	Prom, Mediana

Generaciones	Año ->ALL	Prom, Mediana
Carreras	Carrera ->ALL	Prom, Mediana
Unidades Curriculares	UnidadCurricular ->ALL	Prom, Mediana
TiempoCursos	Semestre ->Año	Prom, Mediana
	Año ->ALL	Prom, Mediana
	Semestre ->SemestreGlobal	Prom, Mediana
	SemestreGlobal ->ALL	Prom, Mediana

El diseño conceptual es muy útil para poder visualizar las estructuras dimensionales, pero no incluye detalles de implementación. A continuación abordaremos el diseño lógico de Data Warehouse tomando como entrada el diseño conceptual de esta sección.

A3. Diseño lógico

La etapa del Diseño Lógico tiene como objetivo definir el esquema lógico tomando como entrada no solo el esquema conceptual multidimensional presentado en el punto anterior, sino también estrategias para resolver los requerimientos no funcionales.

Para almacenar los datos se utilizará una base de datos relacional y el esquema que se utilizará será en estrella. Por ende tendremos una tabla por cada dimensión y una tabla por cada relación dimensional con sus respectivas medidas, a las que llamaremos tablas de hechos. En las tablas de dimensión tenemos las jerarquías embebidas y en consecuencia están desnormalizadas. Para las tablas de hechos se tendrá una columna por cada dimensión participante donde se guardará el identificador de la misma. Para una mejor identificación a las tablas de dimensión se les agrega el `dim_` y a las tablas de hechos se le agrega el prefijo `fact_`.

A continuación se presentarán los diagramas estrella y los esquema de relación de cada tabla para cada requerimiento.

Requerimiento 1: Ingresos

Los estudiantes se identifican con la cédula, se conoce su nombre y además se tiene un identificador y nombre para cada una de las características por las cuales se agrupan (sexo, rango edad ingreso, procedencia geográfica y tipo de institución

de procedencia). Para dar soporte al requerimiento no funcional 2 se agrega el idPrimerCarrera, con el identificador de la primer carrera a la que se inscribió el estudiante. El esquema de relación es:

`dim_estudiantes(cedula, nombre, idSexo, nombreSexo, idRangoEdadIngreso, nomRangoEdadIngreso, idProcGeografica, nomProcGeografica, idTipoInstProc, nomTipoInstProc, idPrimerCarrera)`

Las generaciones se identifican por el año y además se les agrega un nombre. El esquema de relación es:

`dim_generaciones(idGeneracion, nombreGeneracion)`

De las carreras se conoce su identificador y el nombre. El esquema de relación es:

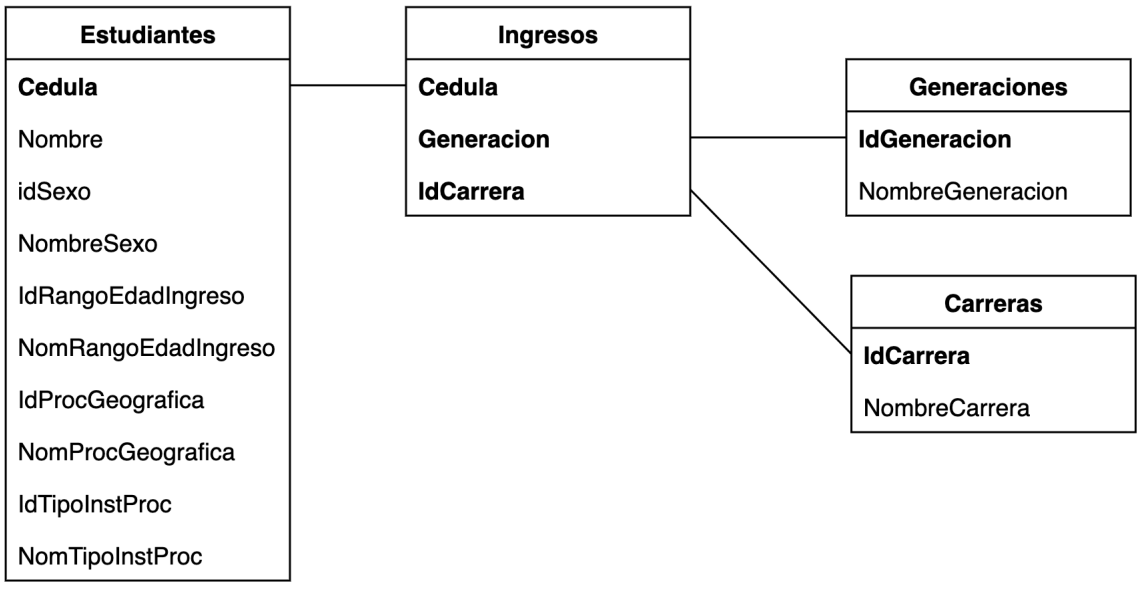
`dim_carreras(idCarrera, nombreCarrera);`

Para la tabla de hechos de ingresos la medida está implícita, y se calcula a partir de la cantidad de filas. El esquema de relación es:

`fact_ingresos(cedula, generacion, idCarrera)`

El esquema estrella resultante de la relación dimensional Ingresos se muestra en la figura A3.1.

Figura A3.1: Esquema estrella resultante de la relación dimensional de Ingresos.



Requerimiento 2: Egresos

Los esquemas de relación para los Estudiantes, Generaciones y Carreras son los mismos que para el Requerimiento 1.

Los años de egresos se identifican por el año y además se les agrega un nombre. El esquema de relación es:

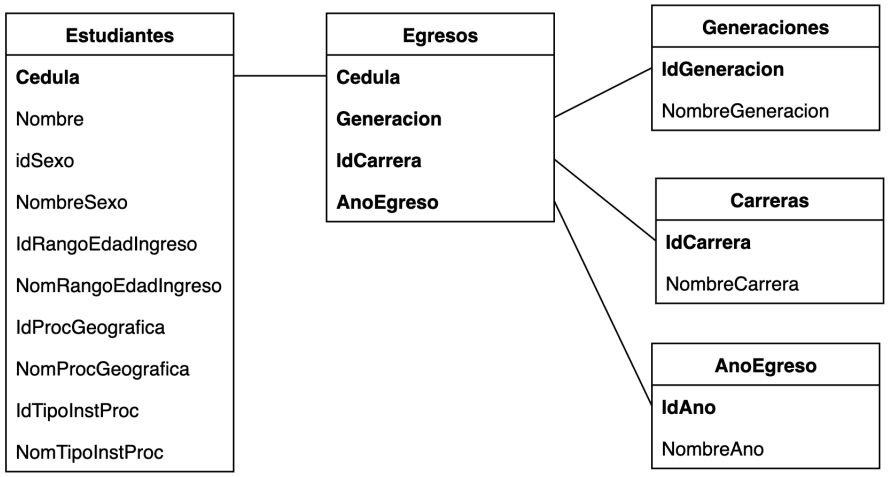
`dim_ano_egreso(idAno, nombreAno)`

Para la tabla de hechos de egresos la medida está implícita, y se calcula a partir de la cantidad de filas. El esquema de relación es:

`fact_egresos(cedula, generacion, idCarrera, anoEgreso)`

El esquema estrella resultante de la relación dimensional Egresos se muestra en la figura A3.2.

Figura A3.2: Esquema estrella resultante de la relación dimensional de Egresos.



Requerimiento 3 : Estudiantes activos

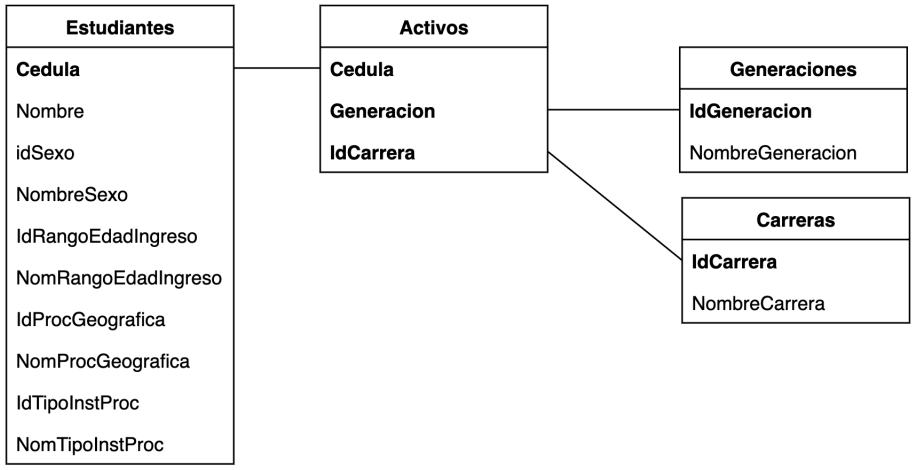
Los esquemas de relación para los Estudiantes, Generaciones y Carreras son los mismos que para el Requerimiento 1.

Para la tabla de hechos de activos la medida está implícita, y se calcula a partir de la cantidad de filas. El esquema de relación es:

`fact_activos(cedula, generacion, idCarrera)`

El esquema estrella resultante de la relación dimensional Activos se muestra en la figura A3.3

Figura A3.3: Esquema estrella resultante de la relación dimensional de Activos.



Requerimiento 4: Puntos críticos

Las unidades curriculares se identifican con un código y tienen un nombre. El esquema de relación es:

`dim_unidades_curriculares(idUnidadCurricular, nombreUnidadCurricular)`

El tiempo de los cursos se identifican por el identificador del semestre (que es una concatenación del semestre y el año en que fue dictado) y tienen un nombre. Además se tiene un identificador por los distintos criterios de agrupación (años o semestres globales). Para cumplir con el requerimiento no funcional 2 se agrega una columna para ordenar la dimensión.

`dim_tiempo_cursos(idSemestre, nombreSemestre, idAno, nombreAno, idSemestreGlobal, nombreSemestreGlobal, orderSemestre)`

El tiempo de los exámenes se identifican por el identificador del periodo (que es una concatenación del mes y el año en que fue tomado) y tienen un nombre. Además se tiene un identificador por los distintos criterios de agrupación (años o períodos globales). Para cumplir con el requerimiento no funcional 2 se agrega una columna para ordenar la dimensión.

`dim_tiempo_examenes(idPeriodo, nombrePeriodo, idAno, nombreAno, idPeriodoGlobal, nombrePeriodoGlobal, orderperiodo)`

El esquema de relación para la tabla de hechos de puntos críticos de los cursos es:

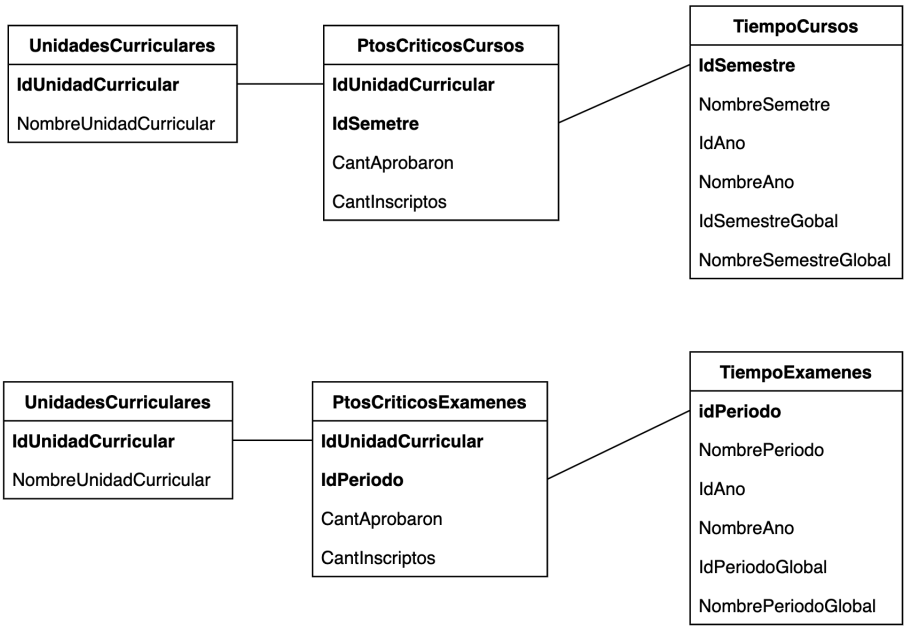
`fact_puntos_criticos_cursos(idUnidadCurricular, idSemestre, cantAprobaron, cantInscriptos)`

El esquema de relación para la tabla de hechos de puntos críticos de exámenes:

`fact_puntos_criticos_examenes(idUnidadCurricular, idPeriodo, cantAprobaron, cantInscriptos)`

El esquema estrella resultante de las relaciones dimensionales PuntosCriticosCursos y PuntosCriticosExamenes se muestra en la figura A3.4.

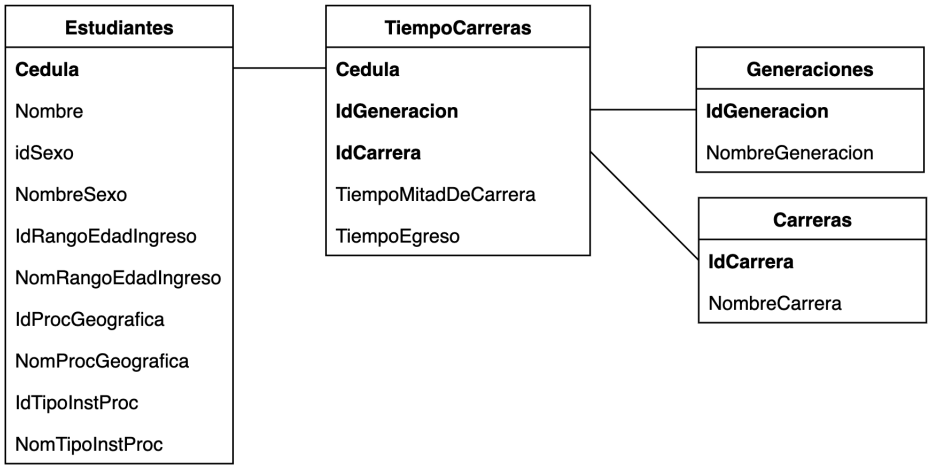
Figura A3.4: Esquema estrella resultante de la relación dimensional de Activos.



Requerimiento 5: Duración carreras

El esquema estrella resultante de la relación dimensional TiempoCarreras se muestra en la figura A3.5.

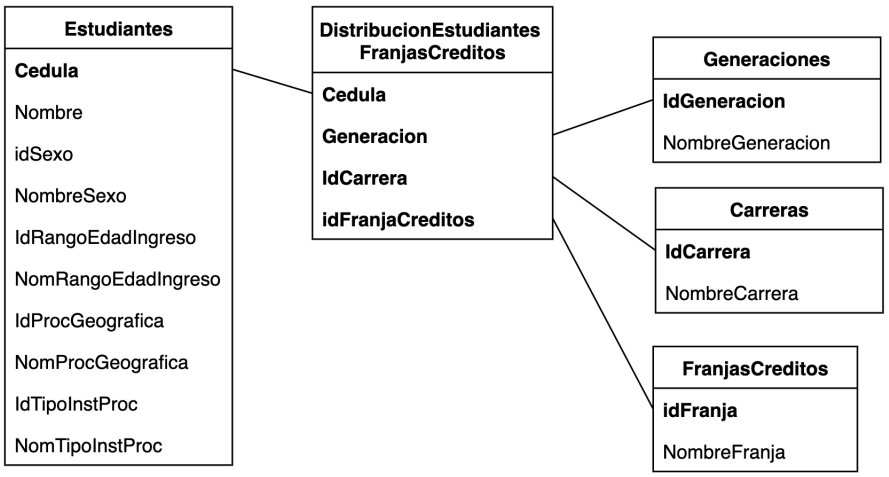
Figura A3.5: Esquema estrella resultante de la relación dimensional de TiempoCarreras.



Requerimiento 6: Avance por franja de créditos

El esquema estrella resultante de la relación dimensional DistribucionEstudiantesFranjasCreditos se muestra en la figura A3.6.

Figura A3.6: Esquema estrella resultante de la relación dimensional de DistribucionEstudiantesFranjasCreditos.

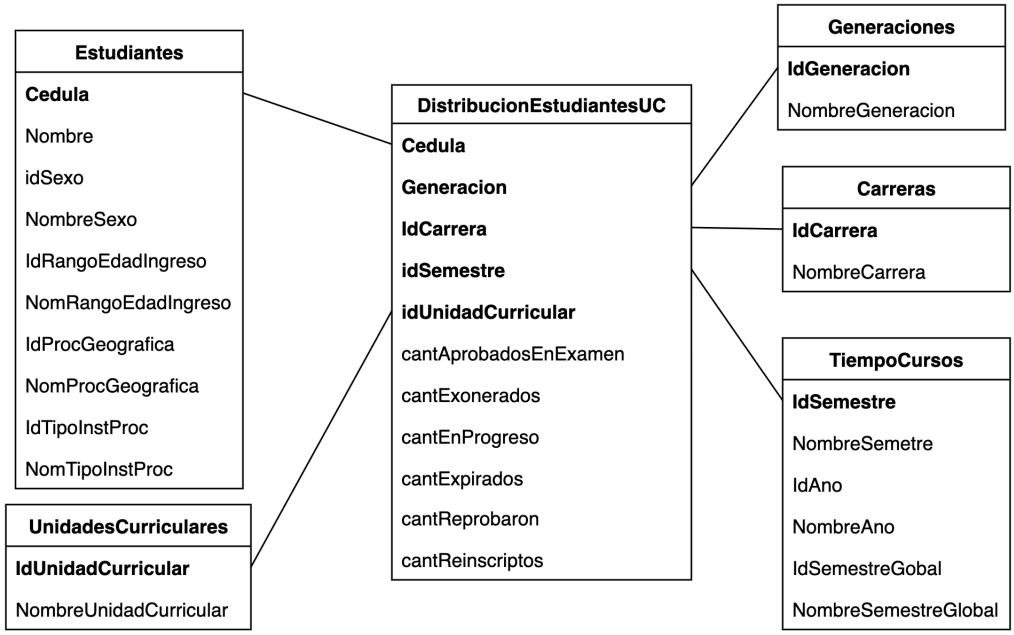


Requerimiento 7: Distribución estudiantes por unidad curricular

Los esquemas de relación para los Estudiantes, Generaciones y Carreras son los mismos que para el Requerimiento 1. El esquema de relación para las Unidades Curriculares es el mismo que en el Requerimiento 4. El esquema de relación para el Tiempo de los cursos es el mismo que para el requerimiento 4.

El esquema estrella resultante de la relación dimensional DistribucionEstudiantesUC se muestra en la figura A3.7.

Figura A3.7: Esquema estrella resultante de la relación dimensional de DistribucionEstudiantesUC.

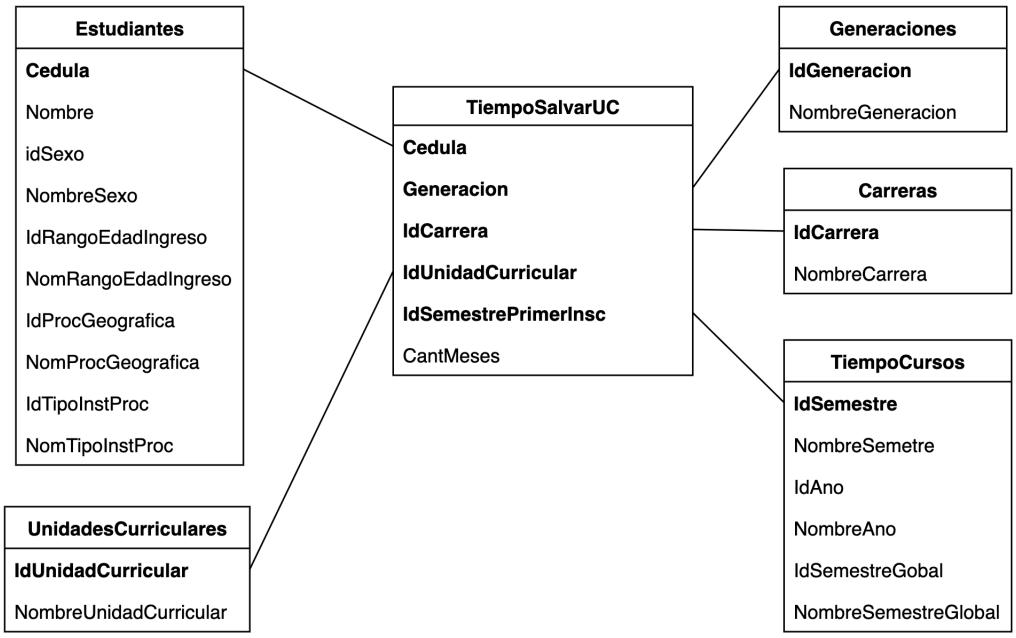


Requerimiento 8: Tiempo que lleva salvar una unidad curricular

Los esquemas de relación para los Estudiantes, Generaciones y Carreras son los mismos que para el Requerimiento 1. El esquema de relación para las Unidades Curriculares es el mismo que en el Requerimiento 4. El esquema de relación para el Tiempo de los cursos es el mismo que para el requerimiento 4.

El esquema estrella resultante de la relación dimensional TiempoSalvarUc se muestra en la figura A3.8.

Figura A3.8: Esquema estrella resultante de la relación dimensional de TiempoSalvarUc.



Una vez realizado el diseño lógico especificando las tablas a utilizar, se abre paso a la etapa de diseño físico donde se entrará aun mas en detalle en algunos aspectos de implementación.

A4. Diseño físico

En esta etapa de diseño se toman en cuenta particularidades de la base de datos utilizada con el fin de realizar optimizaciones para garantizar un buen rendimiento. Estas optimizaciones pueden ser la utilización de índices, vistas materializadas o particionamiento de las tablas [49].

En el Data Warehouse construido no se aplican técnicas especiales para la optimización de consultas. El único aspecto relevante a destacar es que las tablas no están vinculadas con claves foraneas con el fin de facilitar el proceso de ETL.

Apéndice B

Minería de procesos

En este apéndice se puede encontrar información extra relacionada al proceso deETL sobre los datos crudos que se encuentran en la base de datos de Bedelías,estos procesos son: el análisis y unificación de unidades curriculares y la generaciónde la tabla de logs a utilizar en ProM Tools.

A1. Unificación de UC

Dentro de esta sección se encuentran todas las unificaciones de unidades curriculares que fueron realizadas durante el proceso de ETL para luego aplicar el proceso de minería, en la tabla A1.1 se puede ver todos los casos de unificación y su correspondiente código y nombre unificado.

Cuadro A1.1: Unificación de UC.

Código	**Unidad curricular**	**Grupo**
1020	CALCULO 1	CALCULO 1
1070	CALCULO 1	
1061	CALCULO DIF. E INTEGRAL EN UNA VARIABLE	
1030	GEOMETRIA Y ALGEBRA LINEAL 1	GEOMETRIA Y ALGEBRA LINEAL 1
1071	GEOMETRIA Y ALGEBRA LINEAL 1	
1021	ALGEBRA	
1151	FISICA 1	

1171	FISICA 1	FISICA 1
1120	FISICA GENERAL 1	
1121	FISICA GENERAL 2	
1022	CALCULO 2	CALCULO 2
1072	CALCULO 2	
1062	CALCULO DIF. E INT. EN VARIAS VARIABLES	
1031	GEOMETRIA Y ALGEBRA LINEAL 2	GEOMETRIA Y ALGEBRA LINEAL 2
1058	GEOMETRIA Y ALGEBRA LINEAL 2	
1320	PROGRAMACION 1	PROGRAMACION 1
1322	PROGRAMACION 1	
1372	PROGRAMACION 1	
1023	MATEMATICA DISCRETA 1	MATEMATICA DISCRETA 1
1025	PROBABILIDAD Y ESTADISTICA	PROBABILIDAD Y ESTADISTICA
1075	PROBABILIDAD Y ESTADISTICA	
1026	MATEMATICA DISCRETA 2	MATEMATICA DISCRETA 2
1027	LOGICA	LOGICA
1321	PROGRAMACION 2	PROGRAMACION 2
1403	INT. A LA ARQUITECTURA DE COMPUTADORES	INT. A LA ARQUITECTURA DE COMPUTADORES
1443	ARQUITECTURA DE COMPUTADORAS	
1424	ARQUITECTURA DE COMPUTADORES 1	
1323	PROGRAMACION 3	PROGRAMACION 3
1033	METODOS NUMERICOS	METODOS NUMERICOS
1079	METODOS NUMERICOS	
1610	INT. A LA INVESTIGACION DE OPERACIONES	INT. A LA INVESTIGACION DE OPERACIONES
1511	SISTEMAS OPERATIVOS	SISTEMAS OPERATIVOS
1518	SISTEMAS OPERATIVOS	
1532	SISTEMAS OPERATIVOS	
1537	SISTEMAS OPERATIVOS	
1324	PROGRAMACION 4	PROGRAMACION 4

1325	TEORIA DE LENGUAJES	TEORIA DE LENGUAJES
1911	FUNDAMENTOS DE BASES DE DATOS	FUNDAMENTOS DE BASES DE DATOS
1327	TALLER DE PROGRAMACION	TALLER DE PROGRAMACION
1446	REDES DE COMPUTADORAS	REDES DE COMPUTADORAS
1716	INT. A LA INGENIERIA DE SOFTWARE	INT. A LA INGENIERIA DE SOFTWARE
1721	PROYECTO DE INGENIERIA DE SOFTWARE	PROYECTO DE INGENIERIA DE SOFTWARE
1730	PROYECTO DE GRADO	PROYECTO DE GRADO

A2. Generación de datos del log a analizar

En esta sección podemos encontrar la consulta SQL realizada sobre los datos presentes en bedelías para la obtención y generación del log a ser utilizado por ProM Tools para la generación y análisis de los diferentes procesos analizados.

A2.1. DDL para definición de tabla de logs

Aquí se presenta el DDL asociado a la creación de la tabla intermedia de logs, esta tabla es la base para todo el análisis y generación de los diferentes modelos.

```
CREATE TABLE public.log (
    cedula bigint,
    materia character varying,
    fecha_inicio date,
    fecha_fin date,
    estado character varying
);
```

A2.2. SQL para generación de datos

En esta sección se presenta la consulta SQL encargada de popular la tabla de logs descrita en la sección anterior, esta consulta es la encargada de generar las escolaridades para todos los estudiantes a analizar.

```
SELECT DISTINCT estudiantes.cedula,
  asignaturas.grupoasignatura AS materia,
  min(actividades.fecha) AS fecha_inicio,
```

```
  max(actividades.fecha) AS fecha_fin,
  CASE
    WHEN rel_est_carr.fechaegr IS NOT NULL THEN 'RECIBIDO'::text
    WHEN NOT (EXISTS (
      SELECT actividades_1.cedula,
        actividades_1.asignatura,
        actividades_1.tipoactividad,
        actividades_1.nota,
        actividades_1.fecha,
        actividades_1.curricular,
        actividades_1.tipogenerado,
        actividades_1.periodo,
        actividades_1.dictada
      FROM actividades actividades_1
      WHERE estudiantes.cedula = actividades_1.cedula
        AND actividades_1.fecha >= ('2019-04-01'::date - '2 years'::interval)))
    THEN 'DESVINCULADO'::text
    ELSE 'CURSANDO'::text
  END AS estado
FROM actividades
  JOIN estudiantes ON actividades.cedula = estudiantes.cedula
  JOIN rel_est_carr ON rel_est_carr.cedula = estudiantes.cedula
  JOIN asignaturas ON
    asignaturas.asignatura::text = actividades.asignatura::text
    AND asignaturas.carrera = rel_est_carr.carrera
WHERE rel_est_carr.carrera = 72
  AND rel_est_carr.generacion >= 1997
  AND asignaturas.grupoasignatura IS NOT NULL AND NOT (EXISTS (
    SELECT a2.cedula,
      a2.asignatura,
      a2.tipoactividad,
      a2.nota,
      a2.fecha,
      a2.curricular,
      a2.tipogenerado,
      a2.periodo,
      a2.dictada
    FROM actividades a2
    WHERE estudiantes.cedula = a2.cedula
      AND a2.tipogenerado::text = 'R'::text))
```

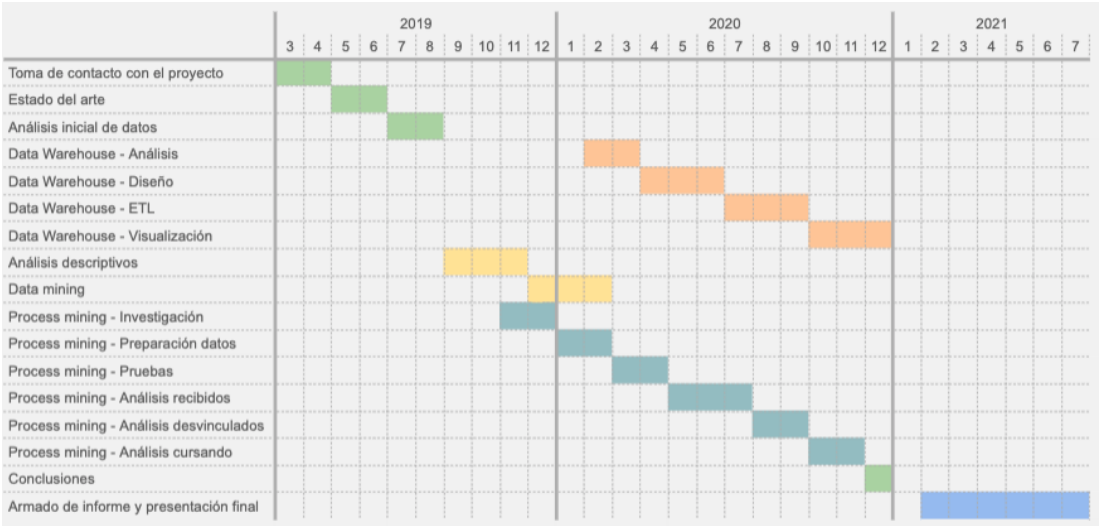
```
    AND (EXISTS (
      SELECT 1
      FROM actividades a2
        JOIN estudiantes e2 ON a2.cedula = e2.cedula
        JOIN rel_est_carr r2 ON r2.cedula = e2.cedula
        JOIN asignaturas g2 ON g2.asignatura::text = a2.asignatura::text
                           AND g2.carrera = r2.carrera
      WHERE estudiantes.cedula = a2.cedula
        AND g2.grupoasignatura::text = asignaturas.grupoasignatura::text
        AND
            CASE
              WHEN
                 a2.tipoactividad::text = 'E'::text AND a2.nota >= 3
                OR a2.tipoactividad::text = 'C'::text AND a2.nota >= 3
                  AND g2.tipoexo::text = 'C'::text
                OR a2.tipoactividad::text = 'C'::text AND a2.nota >= 6
                  AND a2.nota <= 12 AND g2.tipoexo::text = 'E'::text
                THEN true
              ELSE false
            END))
GROUP BY estudiantes.cedula,
  asignaturas.grupoasignatura,
  rel_est_carr.fechaegr;
```

Apéndice C

Cronograma del proyecto

En este apéndice se puede encontrar el desglose de las tareas que se ejecutaron en este proyecto a lo largo del tiempo, diagramadas en un formato gantt como se ve en la figura A0.1

Figura A0.1: Diagrama de Gantt de las tareas ejecutadas en el marco de este proyecto



La ejecución del proyecto comenzó con la toma de contacto del mismo, donde mantuvimos reuniones con la tutora para conocernos y conocer el proyecto de primera mano. Luego de eso se realizó un análisis del estado del arte plasmado en el capítulo 2. En una siguiente instancia tomamos contacto con los datos de Bedelías que nos fueron otorgados por la UEFI los cuales analizamos y los plasmamos en el capítulo 3. Con dichos datos se comenzaron los análisis descriptivos, la ejecución de minería de datos y la minería de procesos plasmados en el capítulo 5.

En paralelo a estos puntos se realizaron todos los pasos para la implementación del Data Warehouse descripto en el capitulo 4. Por último se redactó el presente informe y se preparó la presentación para la defensa.