UNIVERSIDAD NACIONAL DE
ASUNCIÓN
FACULTAD POLITÉCNICA
INGENIERíA EN INFORMÁTICA

Técnica de pronóstico de la demanda basada en Business Intelligence y Machine Learning

Postulantes: Raúl Benítez
Alberto Garcete

Proyecto de Grado Para Obtener el Diploma Académico de Ingeniería en Informática.

Asesores: PhD. Diego P. Pinto Roa
Ing. Aditardo Vázquez

2017

Capítulo 1

Introducción

En las empresas retail o de ventas minoristas uno de los principales problemas con que se enfrentan es el manejo eficiente del stock de manera a evitar tener los productos en exceso en los depósitos que incurran en sobrecostos, o en el otro extremo la falta de dichos productos o ruptura de stock lo cual conlleva a pérdidas de oportunidades de ventas por no disponer del producto que puede generar insatisfacción de los clientes y a su vez repercute en las utilidades de la empresa. Uno de los mayores desafíos en empresas de este sector es pronosticar las ventas para el próximo período comercial.

Actualmente en el proceso de gestión de compras se utilizan ciertas técnicas de pronósticos para determinar las cantidades de las órdenes de compra, dichas técnicas pueden ser cuantitativas o cualitativas. Independientemente de la técnica elegida, el problema real de los pronósticos es su falta de confiabilidad, ya que por lo general no son precisos, entonces, la interrogante que siempre surge es si serán superiores o inferiores a la demanda real y en qué medida.

Con el presente trabajo se plantea una nueva técnica para estimar los volúmenes de ventas del siguiente período comercial, luego esta estimación servirá de apoyo en la toma de decisión de la cantidad establecida en las órdenes de compras para la reposición de stock. En este nuevo modelo se integran técnicas de Business Intelligence y Machine Learning.

En la etapa de Business Intelligence el objetivo principal es calcular los Indicadores Claves de Rendimiento (KPI - Key Performance Indicators) de los productos en base a los datos históricos obtenidos de la base de datos transaccional. Luego cada serie de KPI obtenidos pasan por un proceso de etiquetado, donde el experto en compras los analiza y determina qué nivel de compra conviene para cada serie de KPI.

En la etapa de Machine Learning se utilizan como entrada las series de KPI obtenidas en la etapa anterior de Business Intelligence y constituyen las instancias que alimentan los distintos algoritmos de clasificación del aprendizaje supervisado implementado. Luego tienen lugar los procesos propios de esta etapa que son el entrenamiento y testeo para finalmente evaluar los distintos desempeños a fin de determinar los algoritmos más adecuados que serán utilizados para pronosticar las ventas futuras.

1.1. Motivación

El problema en general de cualquier técnica actual de pronóstico de la demanda, es que por ejemplo en los pronósticos de demanda cuantitativos se depende en gran medida de la variable cantidad de ventas. En los pronósticos de demanda cualitativos se depende en gran medida de la opinión y experiencia del experto, que si bien pueden ser válidos se utilizan principalmente en casos donde la proyección es a largo plazo o ante carencia de datos históricos.

También podemos mencionar otra debilidad de las técnicas de pronósticos actuales, y es que no son adaptativas, en el sentido de que como se tratan de fórmulas genéricas los resultados son uniformes y no tienen en cuenta la evolución del mercado ni aspectos particulares como el tipo de producto, tipo de empresa, ubicación geográfica, etc.

La oportunidad que surge es poder implementar una técnica de pronóstico de demanda automática o con una mínima intervención manual. Esta técnica permitiría paliar las debilidades de los modelos actuales, y a su vez que integraría las características esenciales de las técnicas cualitativas y cuantitativas existentes.

Actualmente, en el proceso de gestión de compras para la reposición de stock se utilizan técnicas de pronósticos para determinar las cantidades de las órdenes de compra. Estas técnicas pueden estar basadas en pronósticos cuantitativos o cualitativos. Independientemente de la técnica elegida el problema real con los pronósticos es su falta de confiabilidad, ya que por lo general no son precisos. La interrogante que siempre surge en estos modelos es si sus resultados serán superiores o inferiores a la demanda real y en qué medida.

En este contexto, este trabajo apuesta a desarrollar una solución que automatice la toma de decisión de la reposición de stock utilizando técnicas basadas en Business Intelligence y Machine Learning en dos fases. En la primera fase, a partir del histórico del movimiento del stock y detalle de ventas el módulo Business Intelligence obtiene los KPI asociado al producto y periodo. En la segunda fase, estos datos son utilizados como entrada al módulo de Machine Learning para determinar el volumen de compra para el siguiente periodo de ventas.

1.2. Planteamiento del Problema

Dado un conjunto de productos que una empresa retail ofrece para la venta, este trabajo trata el problema del pronóstico de la demanda para la reposición de stock. Se considera que la empresa retail opera un stock cíclico, con períodos de reposición regulares y se analizan los productos que no son estacionales.

Se apuesta a desarrollar una solución que automatice la toma de decisiones de reposición de stock. Para lograrlo se ha modelado una técnica de pronóstico de la demanda basada en Business Intelligence y Machine Learning; que busca a través de Key Performance Indicators, de la opinión de un experto en compras y de algorimos de clasificación prever volúmenes eficientes de productos para la reposición de stock.

Se puede decir que se trata de una técnica de dos fases. En la primera fase, a partir del

histórico de movimientos de stock y de los detalles de ventas, el módulo de Business Intelligence obtiene los KPI asociados al producto y período. En la segunda fase, estos datos son utilizados como entradas al módulo de Machine Learning para determinar los volúmenes de compras para el siguiente periodo de ventas.

1.3. Objetivos

A continuación se dan a conocer los objetivos generales y específicos que se pretenden alcanzar con este trabajo.

1.3.1. Objetivos Generales

El aporte principal y objetivo general del presente trabajo es el siguiente:

- Modelar una nueva técnica de pronóstico de la demanda para la toma de decisión en la reposición de stock integrando herramientas y conceptos de Business Intelligence y Machine Learning.

1.3.2. Objetivos Específicos

Los objetivos particulares para el logro del objetivo principal son los siguientes:

- Estudiar la problemática del pronóstico de la demanda.
- Estudiar el funcionamiento de Business Intelligence.
- Estudiar el funcionamiento de Machine Learning.
- Realizar el proceso de Business Intelligence con una base de datos real, modelar el data-warehouse y obtener KPI (Key Performance Indicators).
- Realizar el proceso de etiquetado con la colaboración de un experto en compras y obtener así las instancias que se necesitan para la siguiente etapa.
- Realizar el proceso de Machine Learning y obtener los algoritmos de clasificación que mejor pronostiquen la demanda por cada producto.
- Analizar el rendimiento del modelo de pronóstico de demanda propuesto.
- Proponer enfoques complementarios al modelo propuesto.

1.4. Organización del libro

El libro se estructura de la siguiente manera:

- El Capítulo 2 presenta el concepto de Pronóstico de la Demanda y los principales modelos de pronóstico que son implementados en la actualidad.
- El Capítulo 3 presenta los conceptos que envuelven a Business Intelligence.
- El Capítulo 4 presenta los conceptos que envuelven a Machine Learning.
- En el Capítulo 5 se sigue el proceso de Business Intelligence sobre datos de fuentes reales de una empresa retail.
- En el Capítulo 6 por medio del proceso de Machine Learning se recibe la salida de Business Intelligence y se modela como un problema de clasificación multiclase.
- En el Capítulo 7 tienen lugar los procesos de entrenamiento y evaluación. Se muestran los resultados experimentales y se analiza el desempeño de los algoritmos.
- En el Capítulo 8 se extraen las conclusiones generales del trabajo, contrastando los objetivos propuestos inicialmente y los resultados obtenidos. También se describen las propuestas de trabajos futuros que complementarían la técnica propuesta.

Capítulo 2

Pronóstico de la Demanda

La elaboración de pronósticos de ventas precisos es uno de los retos más importantes en empresas del tipo retail. En un escenario inicial se tienen los depósitos llenos de productos listos para ser llevados a los mostradores. A medida que pasa el tiempo la cantidad en depósito va decreciendo por la demanda de los clientes y llegado un momento crítico hay que tomar la decisión de reponer el stock. Si bien la reposición de stock se lleva a cabo dentro de un proceso empresarial llamado *Administración de Compras*, hay un componente vital dentro de este proceso que es estimar la cantidad o volumen de productos a adquirir para reponer el stock. Es ahí donde entra en juego el pronóstico de la demanda.

A continuación se explicará sintéticamente el proceso de *Administración de Compras*, para luego analizar las principales técnicas de pronósticos de demanda que están vigentes en el mundo empresarial.

2.1. Administración de compras [JLF12]

Los términos compras, adquisiciones, administración de materiales, logística, abastecimiento, administración del suministro y administración de la cadena de suministro se utilizan de manera indistinta ya que no existe un concenso general sobre la terminología. El proceso de adquisición es el eje central de la actividad empresarial de administración de compras y del suministro. Cualquier organización requiere de proveedores por lo que es muy importante acoplarlos con efectividad al entorno organizacional, y que las decisiones de compras no contradigan las estrategias de la empresa.

Las empresas centran sus esfuerzos en aumentar sus ingresos, desminuir sus costos, o una combinación de ambos a fin de obtener ganacias de la forma más eficiente posible. Este trabajo intenta contribuir a lograr decisiones eficientes de compras basadas en pronósticos de demanda precisos. Se considera que es una decisión importantísima estimar o predecir eficientemente la cantidad o volumen de productos para reponer el stock y que sirvan para el período de ventas que está por llegar.

El stock o existencia de una empresa es el conjunto de materiales y artículos que se alma-

cenan, tanto aquellos que son necesarios para el proceso productivo como los destinados a la venta. La función que desempeña el stock o existencia en una empresa son:

- Evitar la escasez, ante la incertidumbre de la demanda o ante un posible retraso en la reposición o suministro de los pedidos.
- Aprovechar la disminución de los costes a medida que aumenta el volumen de compras o de fabricación.
- Lograr un equilibrio entre las compras y las ventas para alcanzar la máxima competitividad.

El proceso de compras o adquisiciones se trata de un conjunto de etapas: a) Detectar la necesidad, b) Traducir la necesidad en una especificación comercial, c) Buscar potenciales proveedores, d) Seleccionar el proveedor adecuado, e) Detallar la orden de compra y pactar el suministro, f) Recibir los productos, g) Pagar a los proveedores. En el detalle de la orden se ven reflejadas las estimaciones de las cantidades a comprar de los productos.

En el proceso de compras el caso ideal por supuesto sería poder adivinar por cada producto la cantidad que se va a vender en el siguiente periodo de venta. De ser así al finalizar cada periodo de ventas se dispondría de stock cero, con lo cual se llega a una máxima eficiencia en compras. Pero como adivinar es imposible, lo que si se puede hacer es predecir eficientemente la demanda futura.

Del por qué la importancia de estimar de forma correcta esta cantidad o volumen, los expertos en negocios explican que los productos parados en stock mientras no se venden es dinero en estantería, además que generan sobrecostos de mantenimiento como seguros, personal encargado, fecha de vencimiento de los productos, etc. Otro hecho no deseado es la ruptura de stock, es decir el no disponer de un producto en stock cuando haya clientes interesados en comprarlo, lo cual también es considerado pérdida para la empresa. Lo que se desea es mantener un nivel de stock óptimo, es decir, por una parte tener suficiente cantidad para satisfacer la demanda sin caer en roturas de stock y, por otra, evitar que haya un exceso inútil del mismo. Si bien el presente trabajo no está enfocado en medir los costos, lo que sí se busca es comprar de forma eficiente utilizando las herramientas de business intelligence y machine learning que ayudan a estimar lo que se va a vender en el siguiente periodo.

Una administración efectiva de las compras y del suministro contribuye de manera significativa al éxito organizacional. La función del suministro evoluciona a medida que la tecnología y el ambiente competitivo mundial requieren enfoques innovadores [JLF12].

Antes de realizar una compra surgen las siguientes preguntas:

- Cuándo debemos realizar un pedido?
- Qué cantidad debemos solicitar en cada pedido?
- Cuántas unidades de cada artículo debemos mantener en stock?

Para responder estas preguntas una de las herramientas que ayudan son las técnicas de pronósticos de demanda, entre las que se destacan los *Métodos de Pronósticos Cualitativos* y los *Métodos de Pronósticos Cuantitativo*s.

2.2. Métodos de pronósticos cualitativos [VAH11][HH08]

- Opinión del Gerente: El pronóstico se basa en la opinión, experiencia o el conocimiento técnico de las condiciones de un solo gerente. Pueden haber datos en los cuales el gerentes apoya su decisión.
- Junta de opinión ejecutiva: Similar al método anterior, la diferencia está en que se basa en un grupo de ejecutivos que intercambian opiniones, perspectivas y conocimientos, luego formulan y componen ideas comunes que sirven de base para emitir un pronóstico unificado, compartiendo de este modo la responsabilidad.
- Consulta a la fuerza de ventas: Esta técnica se basa en la experiencia del personal más cercano al cliente que es el cuerpo de vendedores de la empresa. Cada vendedor realiza una estimación de la demanda en su zona de influencia. Luego las estimaciones son revisadas por los mandos superiores, para obtener un pronóstico corporativo final.
- Encuesta en el mercado de consumo: Se encuesta a los clientes acerca de sus planes de compras, sus intereses por determinados productos o posibles nuevas características. La estimación se extrae de los resultados de las encuestas. Son útiles para elaborar planes de marketing, lazamiento de nuevos productos, etc.
- Método Delphi: Se basa en identificar un panel de expertos que pueden ser gerentes, empleados comunes, o expertos del sector. Se tiene un cuestionario donde cada uno de ellos lo completa de forma aislada. Se integran todas las respuestas, luego cada experto tiene acceso al set de respuestas y puede ajustar su respuesta conforme le parezca conveniente. Este proceso se repite iterativamentte hasta alcanzar un cierto nivel de consenso. Finalmente los resultados de este panel de expertos sirven de base para las decisiones de pronóstico de las personas que deben realizarlo.
- Analogía de productos similares: Se basa en el comportamiento de las ventas de un producto similar o modelo. Técnica útil para nuevos productos que se quieren introducir en el mercado y de los cuales no se dispone de información histórica de ventas, entonces se puede pronosticar haciendo analogía con productos sustitutos o complementarios.

2.3. Métodos de pronósticos cuantitativos

Estos modelos se basan en métodos de pronóstico estadísticos que a partir de los datos históricos de ventas y suponiendo que las tendencias históricas continuarán, son capaces de

anticipar la demanda futura [HH08]. En general, para modelar cuantitativamente se debe disponer de información sobre la variable a pronosticar, la información debe ser cuantificable y el patrón histórico de cierto modo se debe repetir en el futuro [ASW+11].

El pronóstico de la demanda de productos es sólo una aplicación importante de estos métodos. En otros casos, los pronósticos se podrían utilizar para evaluar los requerimientos de cantidades tan diversas como las partes de repuestos, el rendimiento de la producción y las necesidades de personal. Las técnicas de pronóstico se usan también frecuentemente para anticipar las tendencias económicas a nivel regional, nacional o incluso internacional [HH08].

En general, los métodos cuantitativos se clasifican en técnicas de series de tiempo y en pronósticos causales.

2.3.1. Métodos de series de tiempo

Una serie de tiempo es un conjunto de observaciones de la variable a pronosticar, medidas en puntos o períodos sucesivos del tiempo pasado [HH08]. El histórico de ventas de un producto donde se observan valores diarios de las cantidades vendidas constituye un buen ejemplo de serie de tiempo. Los datos históricos de la variable a predecir están limitados a sus valores pasados.

El objetivo del método es obtener una buena predicción del valor futuro de la variable a pronosticar, enmarcado por supuesto en la serie de tiempo. Para lograr el objetivo, el modelo debe descubrir el patrón dentro de la serie y luego ser capaz de extrapolarlo hacia el futuro [ASW+11]. De cierta manera, hay una suposición intrínseca al modelo de que los factores que influyen en las ventas pasadas y presentes continuarán a futuro.

Si bien el volumen de ventas es un buen indicador de la historia de la demanda, no toma en cuenta muchos aspectos del proceso entero de las ventas, como pueden ser la ruptura de stock, plazos de reposición de stock, precio del producto, la incidencia del marketing u otros. De igual modo se pueden descubrir tendencias, estacionalidad, ciclos, etc., en la historia de la demanda para luego extrapolarlo a un tiempo futuro. También hay que destacar que el intervalo del muestreo tiene mucha influencia en el pronóstico y por ende en los resultados obtenidos [PMAR07].

En el sentido estricto de la interpretación es erróneo hablar de pronosticar el siguiente valor de la observación en una serie de tiempo. Como este valor puede ser cualquiera y dependerá de circunstancias futuras que son ajenas al control humano, entonces es imposible predecirlo exactamente. El siguiente valor de una serie de tiempo es una variable al azar y tiene alguna distribución de probabilidades. Si ese siguiente valor es la media de la distribución de probabilidades acortaría el problema, pero se desconoce su distribución de probabilidades así como también su media. Lo mejor que se puede realizar es una estimación de la media tan cerca como sea posible, utilizando todos los datos disponibles. La meta de los métodos de pronóstico de series de tiempo es estimar la media de la distribución de probabilidades subyacente del siguiente valor de la serie de tiempo tan cerca como sea posible. Para una serie de tiempo que tiene exactamente la misma distribución para todos y cada uno de los periodos, el método

de pronóstico de promedios proporciona la mejor estimación de la media, pero en general se usan otros métodos de pronósticos porque la distribución cambia con el paso del tiempo. Si la distribución de probabilidad de una serie sigue siendo la misma en el siguiente periodo entonces se dice que es estable (puede haber cambios en la distribución pero deben ser pequeños). Si la distribución de probabilidad presenta cambios grandes y frecuentes entonces se dice que es inestable [HH08].

El rol del analista es capturar los componentes del patrón de la demanda y luego traducirlo a un valor de pronóstico. Para Johnson [JLF12] este patrón tiene seis componentes básicos: valor constante (la fluctuación de los datos alrededor de una media constante), tendencia (el incremento o decremento sistemático de la media a lo largo del tiempo), variaciones estacionales, cíclicas, aleatorias y puntos críticos. Para Anderson [ASW+11] el patrón de los datos en una serie de tiempo tienen cuatro componentes separados: tendencia, cíclico, estacional e irregular, y que luego se combinan para generar los valores de la serie de tiempo. Se analizan estos cuatro componente:

1. Componente de tendencia: Los valores de la serie de tiempo pueden ir cambiando gradualmente, tendiendo hacia valores que incrementan o que disminuyen. Cuando estos incrementos o disminuciones se dan por periodos de tiempo prolongados se dice que la serie tiene un componente de tendencia. a) muestra una tendencia no lineal; en este caso la serie de tiempo indica poco crecimiento inicial, luego un periodo de rápido crecimiento y por último una estabilización. b) La tendencia lineal decreciente de la gráfica es útil para las series de tiempo que muestran una declinación constante en el tiempo. c) La línea horizontal en la gráfica representa una serie de tiempo que no tiene un aumento o disminución constante en el tiempo, y por tanto no muestra tendencia. Generalmente el componente tendencia se debe a factores a largo plazo.

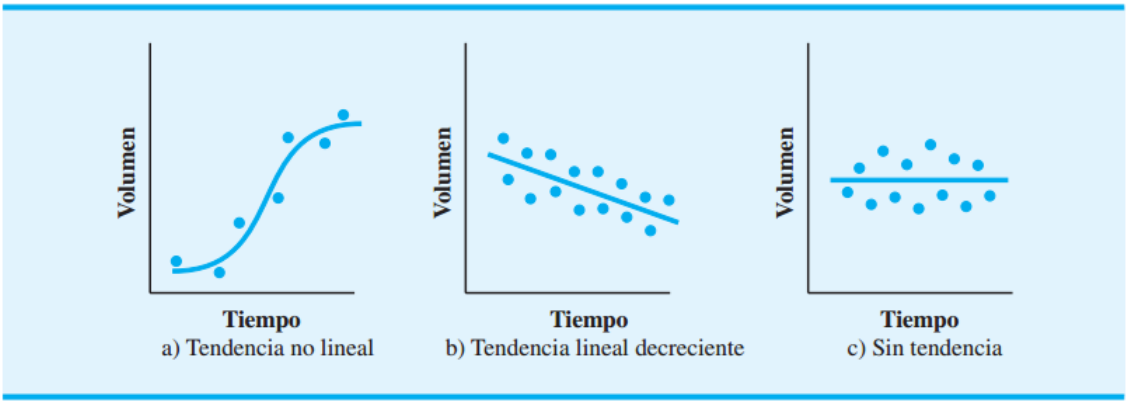


Figura 2.1: Ejemplos del componente tendencia en las series de tiempo.

2. Componente cíclico: es habitual que los puntos de la serie de tiempo se encuentren por encima o por debajo de la linea de tendencia. Cuando el patrón de puntos está de forma alterna por encima y por debajo de la linea de tendencia durante períodos mayor a un año, entonces estamos ante presencia del componente cíclico de la serie de tiempo.

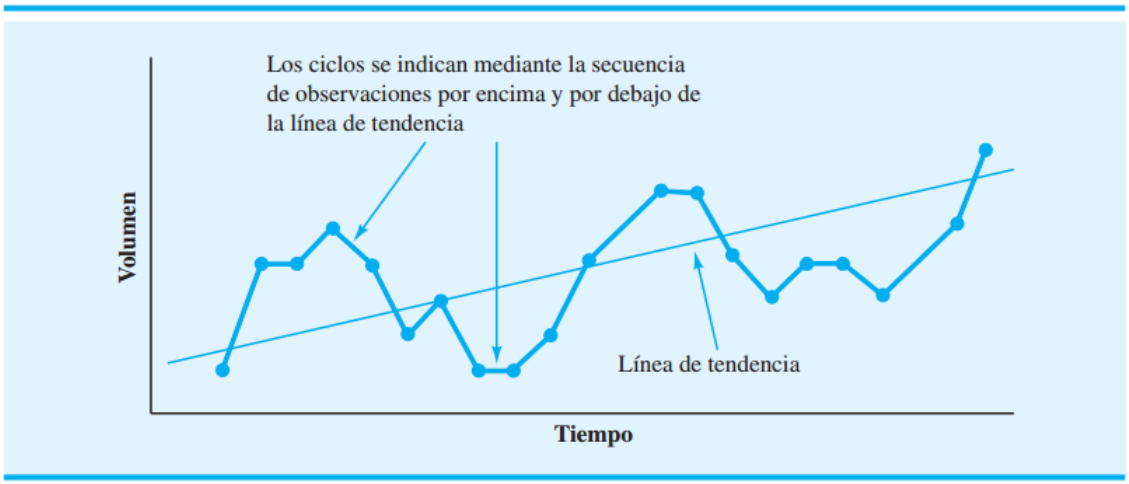


Figura 2.2: Ejemplo del componente cíclico y de tendencia en las series de tiempo.

3. Componente estacional: Cuando el patrón de puntos está por encima o por debajo de la linea de tendencia durante períodos alternos menores o igual a un año, entonces estamos ante presencia del componente estacional de la serie de tiempo. Las altas cantidades de ventas de abrigos durante el otoño e invierno son patrones que se repite debido a la influencia estacional. En ciertos tipos de productos, las ventas que se anticipan en un mes particular están influidas por la temporada del año. Por ejemplo, un producto que es popular en Navidad, podría tener ventas en diciembre que son dos veces mayores que las ventas de enero [HH08].

4. Componente irregular: Se refiere a la presencia de variabilidad aleatoria en la serie de tiempo. Corresponde a aquellos puntos que aparecieron desviados en relación a lo esperado de los efectos del componente de tendencia, cíclico y estacional. Como este componente es impredecible no se puede cuantificar el impacto que tiene en la serie de tiempo. Generalmente es debido a factores a corto plazo o a circunstancias casuales.

En general, los métodos de series de tiempo se clasifican en: método de pronóstico del último valor, método de pronóstico por promedios, método de pronóstico de promedio móvil, método de pronóstico por suavizamiento exponencial, método de suavizamiento exponencial con tendencia y el método ARIMA (AutoRegressive Integrated Moving Average) [HH08].

- El método de pronóstico del último valor: Este método utiliza solamente el último valor de la serie de tiempo como pronóstico del valor futuro. También es conocido como método ingenuo, porque sin mucho análisis aparentemente resulta ingenuo elegir un solo valor de toda la serie. Pero en ocasiones sí es una buena aproximación, como por ejemplo cuando hay demasiada fluctuación en la serie y entonces el último valor se convierte en el más fiable. Recomendable para series de tiempo inestables.

$$Pronóstico = último\ valor \tag{2.1}$$

- El método de pronóstico por promedios: En este caso se utilizan todos los valores de la

serie y luego se promedia para obtener el valor de pronóstico de la serie. Recomendable para series de tiempo estables, razón por la cual todos los valores tienen el mismo peso y son considerados relevantes.

$$Pronóstico = promedio\ de\ todos\ los\ valores\ hasta\ la\ fecha \tag{2.2}$$

- El método de pronóstico de promedio móvil: Consiste en considerar solamente los últimos n períodos y luego promediarlo para así obtener el valor de pronóstico de la serie. Recomendable para series de tiempo medianamente estables, razón por la cual se toman en cuenta únicamente n valores que tienen el mismo peso y que son considerados importantes.

$$Pronóstico = promedio\ de\ los\ últimos\ n\ valores \tag{2.3}$$

donde $n = número\ de\ periodos\ más\ recientes$

- El método de pronóstico por suavizamiento exponencial: Con este método se asignan pesos diferentes a los valores de la serie. El último período es el de mayor peso y así paulatinamente se van asignando pesos cada vez menores a los valores mas antiguos de la serie. Este resultado se puede obtener de forma simple y sintética mediante una combinación del último valor de la serie y del último pronóstico correspondiente a dicho último valor.

$$Pronóstico = \alpha\,(último\ valor) + (1\text{-}\alpha)\,(último\ pronóstico) \tag{2.4}$$

donde α (alfa) es una constante entre 0 y 1 llamada "constante de suavizamiento". Para series de tiempo estables es recomendable valores de α pequeños como 0,1, y para series de tiempo inestables valores mayores. En general en aplicaciones de hoy en día se utilizan valores entre 0,1 y 0,3.

- El método de suavizamiento exponencial con tendencia: El inconveniente del método de pronóstico por suavizamiento exponencial sin tendencia es que justamente se retrasa suficiente respecto de la tendencia ya que no lo toma en cuenta. Al considerar la tendencia se obtienen pronósticos más precisos. Se calcula la pendiente actual de la linea de tendencia para luego ajustar el nuevo pronóstico a la pendiente obtenida. Los valores más recientes de la serie de tiempo se utilizan para obtener la linea de tendencia actual y pueden tener dirección ascendente, descendente u horizontal.

$$Pronóstico = \alpha\,(último\ valor) + (1\text{-}\alpha)\,(último\ pronóstico) + tendencia\ estimada \tag{2.5}$$

$$tendencia\ estimada = \beta\,(última\ tendencia)\,(1\text{-}\beta)\,(estimación\ anterior) \tag{2.6}$$

$$\textit{última tendencia} = \alpha\,(\textit{último valor} - \textit{penúltimo valor}) + (1\text{-}\alpha)\,(\textit{último pronóstico} - \textit{penúltimo pronóstico}) \tag{2.7}$$

donde β (beta) es una constante de suavizamiento de tendencia entre 0 y 1. La elección del valor y rango de β tienen igual significado que α.

2.3.2. Pronósticos causales [HH08]

Ciertamente las series de tiempo se basan en un solo indicador clave, como lo es por ejemplo la variable ventas. Siguiendo el ejemplo, el objetivo de la serie de tiempo es encontrar un valor de pronóstico de la variable ventas a partir de valores pasados de la misma variable ventas. Ahora bien, si tenemos dos variables en relación causa-efecto las series de tiempo no nos sirven.

El pronóstico causal obtiene una proyección de la cantidad de interés (la variable dependiente) relacionándola directamente con una o más cantidades (las variables independientes) que impulsan la cantidad de interés. Por ejemplo, las promociones sobre uno o varios productos pueden ser la causa de una mayor cantidad de ventas en dichos artículos, como tal tenemos una relación causa (promociones)-efecto (mayores ventas), es decir las promociones provocan cambios en los niveles de ventas.

Una de las técnicas para resolver problemas de pronósticos causales es la *regresión lineal*. El objetivo de este método es encontrar la linea recta que más se aproxime a la relación entre la variable dependiente y la/s variables independiente/s. Cuando hay una sola variable independiente la forma de la ecuación es la de la recta:

$$y = a + bx \tag{2.8}$$

donde: $y = \textit{variable dependiente}$, $x = \textit{variable independiente}$, $a = \textit{intersección de la línea con el eje } y$, $b = \textit{pendiente de la línea}$

Para obtener a y b se utiliza el método llamado de *mínimos cuadrados*, que encuentra el par de valores a y b tal que la suma del cuadrado de los errores de estimación sea el menor posible. Para problemas donde se consideran varios indicadores clave como variables independientes, la ecuación presenta la siguiente forma:

$$y = a + b_1x_1 + b_2x_{2+\ldots+}b_nx_n \tag{2.9}$$

Donde el proceso de obtención de a y $b_1, b_2, ..., b_n$ es también por el método de *mínimos cuadrados*.

2.3.3. Precisión de los métodos cuantitativos

Para medir la desviación que hay entre el pronóstico y el valor real se utiliza normalmente el valor del *error promedio del pronóstico*, conocido como *Desviación Absoluta Media (MAD)* y que se calcula con la siguiente fórmula:

$$MAD = \frac{suma\ de\ los\ errores\ de\ pronóstico}{número\ de\ pronósticos} \tag{2.10}$$

Otra forma significativa de medir la precisión es a través del *Error Cuadrático Promedio (MSE)* y que se calcula de la siguiente forma:

$$MSE = \frac{suma\ de\ los\ cuadrados\ de\ los\ errores\ de\ pronóstico}{número\ de\ pronósticos} \tag{2.11}$$

El resultado de esta última fórmula pone de relieve los errores grandes de pronóstico, así como también destaca si el método de pronóstico es preciso. Se utiliza como complemento informativo a MAD.

2.4. Relación con el problema de estudio

Este capítulo pretendió dar a conocer las técnicas de pronósticos de demanda ampliamente conocidas y estudiadas en los diferentes textos consultados [JLF12][VAH11][HH08] [ASW+11]. Si bien el presente trabajo también busca encontrar una estimación de pronóstico como lo hacen las técnicas de esta sección, debe quedar claro que la solución propuesta es una alternativa distinta, es decir, no se trata de un método cuantitativo o cualitativo propiamente dicho, sin embargo toma ciertos aspectos de ambos y se implementa con conceptos, tecnologías y herramientas de solución totalmente diferentes.

En los capítulos tres y cuatro se combinan los conceptos, herramientas y tecnologías que modelan la solución de esta nueva técnica de pronóstico. En quinto capítulo se realizan las experimentaciones en base al caso de estudio y se evaluan los resultados obtenidos.

Capítulo 3

Business Intelligence

En la actualidad Business Intelligence está siendo cada vez más adoptado por las organizaciones debido a la necesidad de los mandos superiores de contar con información rápida y precisa necesaria para la toma de decisiones y su importancia a nivel estratégico y operativo, en este capítulo se presenta los conceptos y una introducción al Business Intelligence.

3.1. Definición

Business Intelligence no se trata ni de un producto ni de un sistema, es una arquitectura que engloba un conjunto de conceptos, técnicas de computación y herramientas para analizar y transformar los datos empresariales en información significativa y útil que permite ser de apoyo a las organizaciones en la toma de decisiones y brindarles una visión estratégica, táctica y operativa más efectivas mediante un acceso fácil a los datos empresariales. Las tecnologías de Business Intelligence ofrecen vistas históricas, actuales y predictivas de las operaciones, son procesos que se extienden en el tiempo, capaces de manejar grandes volúmenes de datos que ayudan a identificar, crear y desarrollar nuevas estrategias de negocios para mejorar la competitividad. La era actual de las tecnologías de la información ha llevado a la necesidad de tener mejores, más rápidas y más eficientes métodos de extraer los datos de una organización, transformarlo en información y distribuirlo a las cadenas de mando. Business Intelligence responde a dicha necesidad [MA03, JC10, Can07].

El primero que acuñó el término fue Howard Dresner, quién fue consultor de Gartner Group lo utilizó para describir un conjunto de conceptos y métodos que mejoran la toma de decisiones, partiendo de la información disponible acerca de los hechos. Entonces, partiendo de la definición del glosario de términos de Gartner [Gar06]:
"*Business Intelligence es un proceso interactivo para explorar y analizar información estructurada sobre un área (normalmente almacenada en un datawarehouse), para descubrir tendencias o patrones, a partir de las cuales derivar ideas y extraer conclusiones. Las áreas incluyen clientes, proveedores, productos, servicios y competidores. El proceso de business intelligence incluye la comunicación de los descubrimientos y efectuar los cambios*".

Una definición más formal que propone The Datawarehouse Institute es[EH05]:
"*Business Intelligence es un término paraguas que abarca los procesos, las herramientas y las tecnologías para convertir datos en información, información en conocimiento y planes para conducir de forma eficaz las actividades de los negocios. Business Intelligence abarca las tecnologías de datawarehousing, los procesos en el 'back end'*[1]*, consultas, informes, análisis y las herramientas para mostrar información (herramientas de Business Intelligence) y los procesos en el 'front end'*".

3.1.1. Objetivos

Según lo expuesto en la definición de Business Intelligence, tiene los siguientes objetivos principales[Can07]:

- Convertir datos en información, información en conocimiento y conocimiento en planes operativos o estratégicos.
- Facilitar la disponibilidad de información a los usuarios de negocios, que les ayude a tomar decisiones más rápidamente.
- Apoyar de forma sostenible y continuada a las organizaciones para mejorar su competitividad, ante el entorno de negocios cambiante de forma que puedan adaptarse a él.
- Ante la cantidad de información que va creciendo, disponer de más tiempo en analizarla, en vez de gastar mucho tiempo en prepararla, organizarla y estructurarla.
- Permitir a las organizaciones dirigir de mejor forma, decidir, medir, gestionar y optimizar el alcance de la eficiencia y los resultados financieros.
- Disminuir sustancialmente la incertidumbre que existe ante la toma de decisiones respecto a un plan estratégico.

3.2. Componentes de Business Intelligence

Implementar un proyecto de Business Intelligence en una organización es un proceso que sigue una serie de pasos, cada paso puede verse como un componente. En la siguiente gráfica observamos los distintos componentes que forman parte de Business Intelligence.[Can07]

[1] *Los términos "back end" y "front end" comúnmente usados en Sistemas de Información significan, respectivamente, la parte más cercana al área tecnológica y la más cercana a los usuarios. Si hiciéramos un paralelismo con una tienda, serían la "trastienda" y el "mostrador*

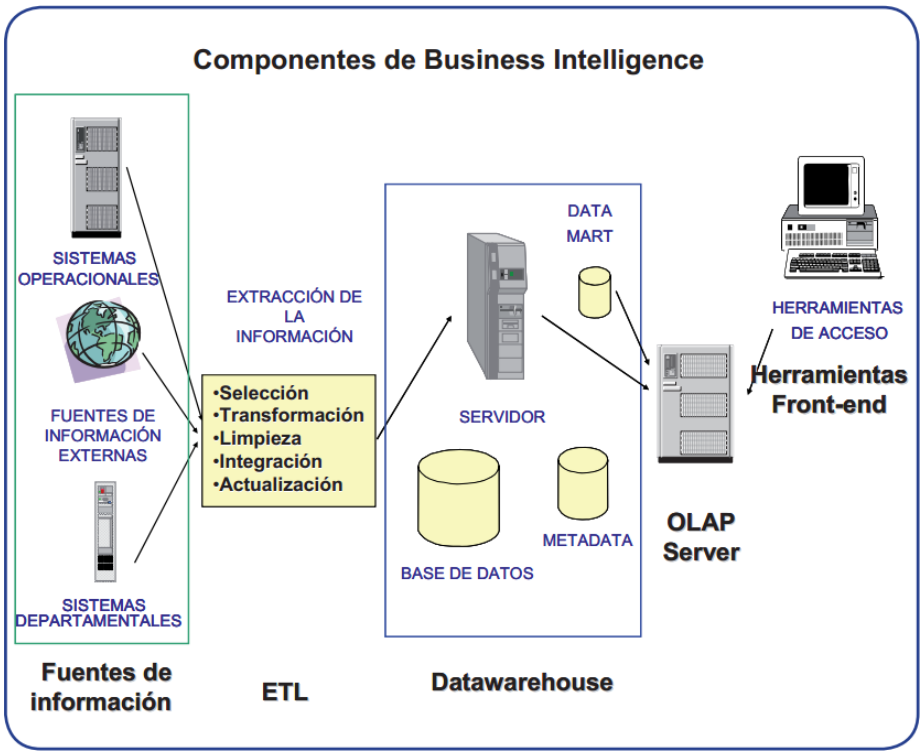


Figura 3.1: Componentes de Business Intelligence

A continuación, una breve descripción de los componentes de Business Intelligence:

3.2.1. Fuentes de Información

Las fuentes de información representan el origen de los datos con la cual se alimenta de información al datawarehouse. Estas pueden provenir de diferentes sistemas como:

- Sistemas operacionales o transaccionales, que incluyen aplicaciones desarrolladas a medida, ERP, CRM, SCM, etc.
- Sistemas de información departamentales: previsiones, presupuestos, hojas de cálculo, etc.
- Fuentes de información externa, en algunos casos comprada a terceros. Las fuentes de información externas podrían ser importantes para enriquecer la información acerca de los clientes. En algunos casos es interesante incorporar información referente, por ejemplo, a población, número de habitantes, etc.

También las fuentes de información son usualmente heterogéneas, pueden contener tipos de datos: [JC10]

- Estructurados: almacenados en las bases de datos.
- Semi estructurados: son formatos entendibles por los computadores como Html tabulado, Excel, CSV u otros que pueden ser obtenidos mediante técnicas estándar de extracción de datos.

- No estructurados: son formatos no legibles para computadoras como Word, Html no tabulado, PDF, etc que pueden obtenerse mediante técnicas avanzadas de extracción de datos.

3.2.2. Extracción, Transformación y Carga

La extracción, transformación y carga, comúnmente abreviado por las siglas ETL (*del inglés "Extract, Transform and Load"*) es un tipo de integración de datos que consiste en todo el proceso que se realiza entre las fuentes de información y el área de presentación de los datos[Kim92]. Es utilizado para extraer los datos de los sistemas de origen, transformarlos en función a los requerimientos del negocio y cargar los datos en el entorno de destino.

El proceso ETL se divide en 5 subprocesos[Can07]:

Extracción

La extracción es el primer paso en el proceso de obtención de los datos, recupera los datos físicamente de las distintas fuentes de información. En este punto se dispone de los datos en bruto. El principal objetivo es extraer aquellos datos de los sistemas transaccionales que son necesarios y prepararlos para el resto de los subprocesos de ETL. Para ello se deben determinar las mejores fuentes de información, las de mejor calidad. Para tal finalidad, se debe analizar las fuentes disponibles y escoger aquellas que sean mejores.

Limpieza

Este proceso recupera los datos en bruto y comprueba su calidad, elimina los duplicados y, cuando es posible, corrige los valores erróneos y completa los valores vacíos, es decir se transforman los datos -siempre que sea posible- para reducir los errores de carga. En este momento se disponen de datos limpios y de alta calidad.

Los sistemas transaccionales contienen datos que no han sido depurados y que deben ser limpiados. Algunas causas que provocan que los datos estén "sucios" son:

- Valores por defecto.
- Ausencia de valor.
- Campos que tienen distintas utilidades.
- Valores contradictorios.
- Uso inapropiado de los campos.
- Re utilización de claves primarias.
- Selección del primer valor de una lista.

- Problemas de carga de antiguos sistemas o de integración entre sistemas.

La limpieza de datos se divide en distintas etapas:

- ***Depurar los valores*** *(parsing)*: localiza e identifica los elementos individuales de información en las fuentes de datos. Por ejemplo: separa el nombre completo en: nombre, primer apellido, segundo apellido; o la dirección en: calle, número, etc.
- ***Corregir*** *(correcting)*: corrige los valores individuales de los atributos usando algoritmos de corrección y fuentes de datos externas. Por ejemplo: comprueba la dirección y su código postal correspondiente.
- ***Estandarizar*** *(standardizing)*: aplica rutinas de conversión para transformar valores en formatos definidos y consistentes. Por ejemplo: trato de Sra. o Sr. cambiar a sus correspondientes nombres completos.
- ***Relacionar*** *(matching)*: busca y relaciona los valores de registros, corrigiéndolos y estandarizándolos para eliminar duplicados. Por ejemplo: identificando nombres y direcciones similares.

Transformación

La transformación de los datos se realiza partiendo de los datos una vez limpios, se transforman los datos de acuerdo a las reglas y necesidades del negocio. El resultado de este proceso es la obtención de datos limpios, consistentes, sumarizados y útiles.

La transformación incluye:

- Cambios de formato.
- Sustitución de códigos.
- Valores derivados y agregados.

Los agregados como las sumas de las ventas normalmente se precalculan y se almacenan para conseguir mayores rendimientos. En este proceso también se ajusta el nivel de granularidad o detalle, por ejemplo: se puede tener detalles a nivel de lineas de factura en los datos extraídos, pero en el datawarehouse lo que se almacena son las ventas semanales o mensuales. La diferencia del nivel de detalle en el análisis es lo que se denomina granularidad.

Integración

Este proceso valida que los datos que cargamos en el datawarehouse son consistentes con las definiciones y formatos del datawarehouse; los integra en los distintos modelos de las distintas áreas de negocio que hemos definido en el mismo. Estos procesos pueden ser complejos.

Actualización

Este proceso es el que nos permite añadir los nuevos datos al datawarehouse, determina la periodicidad con el que haremos nuevas cargas de datos al datawarehouse.

3.2.3. DATAWAREHOUSE

El datawarehouse o almacén de datos son colecciones de datos acerca de los procesos de negocios de una organización, proporciona una visión global, común e integrada de los datos. Hoy en día se piensa que son las principales tecnologías que apoyan el entorno heterogéneo de toma de decisiones, el datawarehouse tiene las siguientes propiedades: no volátil, coherente, fiable y con información histórica [Inm92, JC10].

El profesor Hugh J. Watson [Wat06] lo define como:

"Un datawarehouse es una colección de información creada para soportar las aplicaciones de toma de decisiones. Datawarehousing es el proceso completo de extraer información, transformarla y cargarla en un datawarehouse y el acceso a esta información por los usuarios finales y las aplicaciones."

Bill Inmon [Inm92] definió las características que debe cumplir un datawarehouse:

- Orientado a un área: cada parte del datawarehouse está construida para resolver un problema de negocio. Por ejemplo: entender los hábitos de compra de clientes, analizar la calidad de los productos, analizar la productividad de una línea de fabricación.
- Integrado: la información debe ser transformada en medidas comunes, códigos comunes y formatos comunes para ser útil. Por ejemplo: la moneda en que están expresadas los importes es común.
- Indexado en el tiempo: se mantiene la información histórica. Ejemplo: analizar la evolución de las ventas en los periodos deseados.
- No volátil: los usuarios no la mantienen como lo harían en los entornos transaccionales. No se ve actualizado continuamente, sino periódicamente de forma preestablecida. La información se almacena para la toma de decisiones

Ralph Kimbal [Kim92] define los objetivos que debería cumplir un datawarehouse:

- El alcance de un datawarehouse puede ser bien un departamento o bien corporativo.
- El datawarehouse no es sólo información sino también las herramientas de consulta, análisis y presentación de la información.
- La información del datawarehouse es consistente.
- La calidad de información en el datawarehouse es el motor de business reengineering.

Se debe tener en cuenta que existen otros elementos en el contexto de un datawarehouse[JC10]:

- Datawarehousing: es el proceso de extraer y filtrar datos de las operaciones procedentes de los distintos sistemas de información operacionales y sistemas externos, para transformarlos, integrarlos y almacenarlos en un almacén de datos con el fin de acceder a ellos para dar soporte en el proceso de toma de decisiones.

- Data Mart: es un subconjunto de datos del datawarehouse cuyo objetivo es responder a un determinado análisis.

- Operational Data Store (ODS): es un tipo de almacén de datos que proporciona sólo los últimos valores de los datos y no su historial.

- Staging Área: es el sistema que permanece entre las fuentes de datos y el datawarehouse con el objetivo de:

  - Facilitar la extracción de datos desde fuentes de origen con una heterogeneidad y complejidad grande.
  - Mejorar la calidad de los datos.
  - Ser usado como caché de datos operacionales con el que posteriormente se realiza el proceso de datawarehousing.
  - Uso de la misma para acceder en detalle a información no contenida en el datawarehouse.

- Procesos ETL: tecnología de integración de datos basada en la consolidación de datos que se usa tradicionalmente para alimentar el datawarehouse, data mart, staging área y ODS.

- Metadatos: datos estructurados y codificados que describen características de instancias; aportan informaciones para ayudar a identificar, descubrir, valorar y administrar las instancias descritas.

Elementos de una datawarehouse[JC10]

La estructura relacional de una base de datos operacional sigue las formas normales en su diseño. Un datawarehouse no debe seguir ese patrón de diseño. La idea principal es que la información sea presentada desnormalizada para optimizar las consultas. Para ello se debe identificar, en la organización, los procesos de negocio, las vistas para el proceso de negocio y las medidas cuantificables asociadas a los mismos. Los elementos de un datawarehouse son:

- **Tabla de hecho**: es la representación en el datawarehouse de los procesos de negocio de la organización. A nivel de diseño es una tabla que permite guardar dos tipos de atributos diferenciados:

  - Medidas del proceso de trabajo que se pretende modelizar
  - Claves foráneas hacia registros de una tabla de dimensión

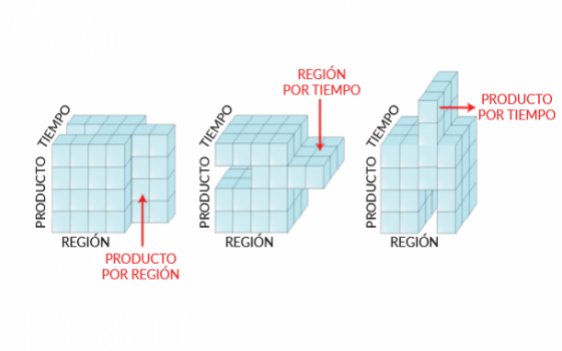


Figura 3.2: Cubo OLAP

- **Dimensión**: es la representación en el datawarehouse de una vista para un cierto proceso de negocio.
- **Métrica**: son los indicadores de negocio de un proceso de negocio. Aquellos conceptos cuantificables que permiten medir el proceso de negocio.

Tipos de esquemas para estructurar los datos en un datawarehouse

- **Esquema en estrella:** consiste en estructurar la información en procesos, vistas y métricas recordando a una estrella. A nivel de diseño, consiste en una tabla de hechos en el centro para el hecho objeto de análisis y una o varias tablas de dimensión por cada punto de vista del análisis que participan en la descripción de ese hecho.
- **Esquema en copo de nieve:** es un esquema de representación derivado del esquema en estrella, en el que las tablas de dimensión se normalizan en múltiples tablas. Por esta razón la tabla de hechos deja de ser la única tabla del esquema que se relaciona con otras tablas.

3.2.4. OLAP

Existen múltiples tecnologías que permiten analizar las información almacenada en un datawarehouse, uno de los más importantes es OLAP (*Online Analytical Processing*), esta tecnología permite realizar un análisis multidimensional de un hecho desde distintas perspectivas o dimensiones mediante consultas complejas que van desde pocas hasta docenas de operaciones de unión, filtrado, agrupación y agregación. El principal objetivo es la de agilizar la consulta de grandes cantidades de datos [Wre06, Can07, JC10].

Una definición formal de OLAP seria[JC10]:

"*Se entiende por OLAP o proceso analítico en linea, al método ágil y flexible para organilizar datos, especialmente metadatos, sobre un objeto o jerarquía de objetos como en un sistema u organización multidimensional, y cuyo objetivo es recuperar y manipular datos y combinaciones de los mismos a través de consultas o incluso informes*"

Una representación gráfica del OLAP son los que se conocen como cubos.

Existen distintos tipos de OLAP, las cuales difieren principalmente en la forma de guardar los datos:

- MOLAP (Multidimensional OLAP): es la forma tradicional del OLAP, accede directamente sobre una base de datos multidimesional, que utiliza estructuras de datos optimizadas para la recuperación de los mismos, es eficaz en los tiempos de respuestas de las consultas.
- ROLAP (Relational OLAP): accede directamente a las bases de datos relacionales que almacenan los datos base y las tablas dimensionales como tablas relacionadas.
- HOLAP (Hybrid OLAP): es una combinación de las dos anteriores, permite almacenar parte de los datos en una base de datos multidimensional y otra parte de en una relacional. En la base de datos relacional se guardan cantidades mas grandes de información detallada, mientras que en la multidimensional se almacenan datos menos detallados o agregados.

3.2.5. HERRAMIENTAS DE BI

Las principales herramientas de Business Intelligence son [EH05]:

- *Generadores de informes*: Utilizadas por desarrolladores profesionales para crear informes estándar para grupos, departamentos o la organización.
- *Herramientas de usuario final de consultas e informes*: Empleadas por usuarios finales para crear informes para ellos mismos o para otros; no requieren programación.
- *Herramientas OLAP*: Permiten a los usuarios finales tratar la información de forma multidimensional para explorarla desde distintas perspectivas y periodos de tiempo.
- *Herramientas de Dashboard y Scorecard*: Permiten a los usuarios finales ver información crítica para el rendimiento con un simple vistazo utilizando iconos gráficos y con la posibilidad de ver más detalle para analizar información detallada e informes, si lo desean.
- *Herramientas de planificación, modelización y consolidación*: Permite a los analistas y a los usuarios finales crear planes de negocio y simulaciones con la información de Business Intelligence. Pueden ser para elaborar la planificación, los presupuestos, las previsiones. Estas herramientas proveen a los dashboards y los scorecards con los objetivos y los umbrales de las métricas.
- *Herramientas datamining*: Permiten a estadísticos o analistas de negocio crear modelos estadísticos de las actividades de los negocios. Datamining es el proceso para descubrir e interpretar patrones desconocidos en la información mediante los cuales resolver problemas de negocio. Los usos más habituales del datamining son: segmentación, venta cruzada, sendas de consumo, clasificación, previsiones, optimizaciones, etc.

3.3. Indicadores Clave de Rendimiento

Los KPI *(Key Performance Indicators)* o Indicadores Clave de Rendimiento se tratan de indicadores que son decisivos para analizar de forma rápida la situación del negocio y que también facilitan la toma de decisiones. Una característica de los KPI es que todos los KPI son indicadores, pero no todos los indicadores son KPI, otra característica es que cada organización debe definir sus propios KPI que desean tener siempre presente para manejar su rumbo, estas varían de acuerdo según la actividad realizada, el tipo de producto o la estrategia de negocios, por dicho motivo los KPI no pueden copiarse de una organización a otra ya que cada organización es diferente y requiere de una reflexión estratégica de los cuales saldrán los correspondientes KPI. Un cuadro de gestión o de mando no debe excederse en la cantidad de KPI, porque puede darse el problema de "la parálisis por el análisis" que ocurre cuando se pasa de no tener ninguna información a contar con decenas de indicadores y una de las características del entorno competitivo actual es que se deben tomar decisiones de forma rápida y antes de que lo hagan los demás competidores[Alv13].

En la siguiente sección haremos uso de los conceptos, herramientas y tecnologías que nos provee Business Intelligence para obtener los datos que ayudarán a modelar una solución al problema de estudio sobre pronóstico de la demanda. Iniciaremos con una breve descripción de la fuente de información, el proceso ETL (Extracción, Transformación y Carga), la especificación del datawarehouse, la definición de los indicadores clave de rendimiento y finalmente el etiquetado a cada tupla de indicadores que se transformarán en datos de entrada para el proceso de aprendizaje automático.

3.4. Aplicaciones de BI

Los sistemas de Business Intelligence abarcan un grupo cada vez mayor de usuarios, desde especialistas en tareas de control, información financiera, personal de ventas, directivos y gerentes[**?**]. Entre los sectores que utilizan sistemas de Business Intelligence se encuentran compañias comerciales, compañias de seguros, entidades financieras, telecomunicaciones y empresas de manufactura.

Tabla 3.1: Áreas de aplicación de BI

Áreas de aplicación de BI	Casos de uso
Empresas Retail	▪ Proporcionar un análisis de las transacciones de los clientes. D promociones, hábitos de compras. ▪ Pronóstico. Uso de datos de escaneado para pronosticar la der definir los requisitos de inventario con mayor precisión
Inventario	▪ Planificación de Inventarios. Ayudar a identificar el nivel de in demanda de los clientes.
Gestión de Pedidos	▪ Pedido y reposición. Uso de la información para tomar decisio y determinar cantidades óptimas.
Contabilidad	▪ El uso de datos contables permite una mejor oportunidad de a operaciones, identificar ahorros de costos y oportunidades estr
Bancos, Financieras y Valores.	▪ Análisis de rentabilidad del cliente: Analizar la rentabilidad gl proporcionar la base para las ventas de alta rentabilidad y la l clientes de alto valor, reducir los costos para los clientes de ba nuevos productos y servicios. ▪ Gestión de créditos: Establecer patrones de progresión de prob clientes, alertar a los clientes para evitar problemas de crédito cartera crediticia del banco, reducir pérdidas crediticias. ▪ Atención en sucursales: Mejorar el servicio y la atención al cli fortalecer la lealtad del cliente.
Telecomunicaciones	▪ Perfil y segmentación de clientes. Determinar perfiles de produ clientes, proporcionar perfiles de clientes detallados e integrad de llamadas frecuentes, determinar futuras necesidades de los ▪ Previsión de la demanda del cliente. Prever las necesidades fu proporcionar una base para el análisis y control de la rotación
Transporte	
Educación	
Salud	▪ Analizar los resultados, identificar tendencias, detectar patron

Capítulo 4

Machine Learning

En este capítulo se realiza un breve repaso sobre conceptos básicos que envuelven a Machine Learning. El objetivo es visualizar qué aspectos de Machine Learning fueron tomados como componentes de solución al problema de estudio.

En las últimas dos décadas, Machine Learning se ha convertido en uno de los pilares de la tecnología de la información y, con eso, una parte bastante central de nuestra vida. Debido a que cada vez hay más datos disponibles disponibles, existen buenas razones para creer que el análisis inteligente de datos será aún más importante como un ingrediente necesario para el progreso tecnológico. El aprendizaje automático puede aplicarse en muchos aspectos. Ahora discutimos una serie de aplicaciones, los tipos de datos con los que se ocupan y, finalmente, formalizamos los problemas de una manera algo más estilizada. Lo último es clave si queremos evitar reinventar la rueda para cada nueva aplicación. En cambio, gran parte del arte del aprendizaje automático es reducir una gama de problemas bastante dispares a un conjunto de prototipos bastante reducido. Gran parte de la ciencia del aprendizaje automático es entonces resolver esos problemas y proporcionar buenas garantías para las soluciones.

4.1. Definición

En 1959 Arthur Samuel en una publicación escribió: *"Programming computers to learn from experience should eventually eliminate the need for much of this detailed programming effort"* [Sam59]. Este pionero de machine learning ya presagiaba que los programas, a partir del aprendizaje sobre datos históricos (la experiencia), podrían efectuar tareas de toma de decisiones sin ser programadas explícitamente dichas decisiones.

Samuel define machine learning como sigue: *"Machine Learning es un campo de estudio que da a las computadoras la capacidad de aprender sin ser explícitamente programadas"*. Otro investigador de machine learning Tom Mitchell propuso en 1998 la siguiente definición: *"Well posed Learning Problem: A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E"*. Donde se nos indica que el aprendizaje en las máquinas deberá

ser parecido al aprendizaje en los humanos, por ejemplo cuando una criatura comienza a hablar a través de la experiencia de pronunciar las palabras y de su interacción con otras personas, entonces sucede que su capacidad de hablar se va perfeccionando o mejorando.

"The purpose of machine learning is to learn from training data in order to make as good as possible predictions on new, unseen, data"[Pug16]. La dificultad radica en que debemos construir modelos que nos acerquen a una buena predicción sobre datos aún no conocidos o imprevistos. Peter Prettenhofer y Gille Louppe presentan la siguiente definición:

Data comes as...

- A set of examples $\{(x_i, y_i) \mid 0 \leq i < n\ samples\}$, with
  - Feature vector $x \in \mathbb{R}^{n\ features}$, and
  - Response $y \in \mathbb{R}$(regression) or $y \in \{-1, 1\}$ (classification)
- Goal is to...
  - Find a function $\hat{y} = f(x)$
  - Such that error $L(y, \hat{y})$on new (unseen) x is minimal

4.2. Formas de Aprendizaje [RN04]

Los algoritmos de aprendizaje automático se pueden agrupar según la forma en que se realiza el aprendizaje, pero teniendo en cuenta que todos reciben un conjunto de ejemplos del cuál aprender. Uno de los componentes más importantes al momento de diagnosticar la naturaleza del problema de aprendizaje es el tipo de retroalimentación disponible para el aprendizaje. Hay tres tipos distintos de aprendizaje: supervisado, no supervisado y por refuerzo.

4.2.1. Aprendizaje supervisado

En los problemas de apredizaje supervisado los algoritmos reciben como entrada datos de entrenamiento que ya tienen resultados conocidos o etiquetas. En el caso más general un instructor provee el valor correcto de la salida de cada ejemplo. Como resultado se aprende una función a partir de las entradas y salidas de los ejemplos. Como ejemplo de aplicación están los vehículos autoconducidos que deben aprender a diferenciar una calle de la que no es (salida booleana es calle o no es calle), también debe aprender a frenar (salida booleana frenar o no frenar), etc. Example problems are classification and regression para Supervised Learning. Example algorithms include Logistic Regression and the Back Propagation Neural Network.

El problema de estudio utiliza algoritmos de aprendizaje supervisado, donde el experto en compras da la respuesta correcta a cada ejemplo.

4.2.2. Aprendizaje no supervisado

En los problemas de apredizaje no supervisado los algoritmos reciben como entrada datos de entrenamiento que no tienen resultados conocidos o etiquetas. Se buscan estructuras presentes y como resultado se pueden extraer reglas generales, o reducir sistemáticamente la redundancia, o se pueden organizar los datos por similitud. Se aprende a partir de patrones de entrada de los que no se dispone de sus valores de salida, es decir a priori no hay etiquetas o respuesta correcta en los ejemplos. Como ejemplo de aplicación está el caso de la computadora que aprendió sola el concepto de un animal gato. Example problems are clustering, dimensionality reduction and association rule learning para Unsupervised Learning. Example algorithms include: the Apriori algorithm and k-Means.

4.2.3. Aprendizaje por refuerzo

El problema del aprendizaje por refuerzo es el más general de las tres categorías. En vez de que un instructor indique al agente qué hacer, el agente de aprendizaje por refuerzo debe aprender a partir del refuerzo o recompensa. Por ejemplo, la falta de propina al final del viaje (o una gran factura por golpear la parte trasera del coche de delante) da al agente algunas indicaciones de que su comportamiento no es el deseable. El aprendizaje por refuerzo típicamente incluye el subproblema de aprender cómo se comporta el entorno.

4.3. Algoritmos de Machine Learning

El Dr. Jason Brownlee es un especialista en aprendizaje automático, desarrollador, escritor y empresario. Ha trabajado en sistemas de aprendizaje automático para la defensa, startups y pronósticos meteorológicos. Tiene una comunidad en https://machinelearningmastery.com/, la cual empezó porque le apasiona ayudar a los desarrolladores profesionales a comenzar y aplicar con confianza el machine learning que les permita resolver problemas complejos. Los algoritmos de aprendizaje automático se pueden agrupar según la similaridad en términos de su forma o función, como por ejemplo los métodos basados en árboles y los métodos inspirados en redes neuronales. Se muestra en la siguiente figura lo propuesto por el Dr. Jason:

Figura 4.1: Algoritmos de machine learning

No hay un consenso general de cómo agrupar los algoritmos de machine learning en términos de su función o de cómo trabajan. La figura mostró un método útil de agrupación, que no es perfecto y ni exhaustivo en los grupos y algoritmos. Hay algoritmos que pueden encajar en varias categorías como Learning Vector Quantization que es a la vez un método inspirado en una red neuronal y un método basado en instancia. También hay categorías que tienen el mismo nombre que describen el problema y la clase de algoritmo como Regression y Clustering. Se podría manejar estos casos listando los algoritmos dos veces o insertando en el grupo al que subjetivamente se ajusta mejor. Se utiliza este último enfoque de no duplicar algoritmos.

4.3.1. Algoritmos de regresión

La regresión modela la relación que existe entre variables, se mejora iterativamente utilizando una medida de error en las predicciones hechas por el modelo. Los métodos de regresión son herramientas de las estadísticas que se han adoptado en el aprendizaje de la máquina. Podemos utilizar el término regresión para referirnos a la clase de problema y también a la clase de algoritmo. Para ser más exactos, la regresión es realmente un proceso.

4.3.2. Algoritmos basados en instancia

El modelo de aprendizaje basado en instancia es un problema de decisión con instancias o ejemplos de datos de entrenamiento que se consideran importantes o requeridos para el modelo. Estos métodos típicamente construyen una base de datos con ejemplos y los compara con los

nuevos datos utilizando una medida de similaridad para así encontrar la mejor coincidencia y hacer la predicción. Por esta razón, los métodos basados en la instancia también se llaman métodos de aprendizaje basado en memoria. El enfoque se pone en la representación de las instancias almacenadas y las medidas de similaridad utilizadas entre instancias.

4.3.3. Algoritmos de regularización

Comprende una extensión hecha a otros métodos (típicamente a los métodos de regresión). Penaliza los modelos basándose en sus complejidades, favoreciendo modelos más simples que también son mejores de generalizar. Son populares, potentes y en general simples modificaciones de otros métodos.

4.3.4. Algoritmos de árboles de decisión

Los métodos de árboles de decisión construyen un modelo de decisiones hechas en base a los valores de atributos en los datos. Las decisiones se bifurcan en la estructura del árbol hasta que se tome una decisión de predicción para un registro dado. Los árboles de decisión son entrenados en los datos para problemas de clasificación y regresión. Los árboles de decisión son a menudo rápidos y precisos y un gran favorito en aprendizaje automático.

4.3.5. Algoritmos bayesianos

Los métodos bayesianos son los que aplican explícitamente el teorema de Bayes para problemas tales como la clasificación y la regresión.

4.3.6. Algoritmos de agrupación

La agrupación así como también la regresión describen la clase de problema y la clase de método. Los métodos de agrupación suelen estar organizados según el enfoque del modelado, tales como los basados en centroides y los jerárquicos. Todos los métodos atañen a la utilización de las estructuras inherentes en los datos, para organizar dichos datos de la mejor manera posible en grupos de máxima uniformidad.

4.3.7. Algoritmos de aprendizaje de reglas de asociación

Los métodos de aprendizaje de reglas de asociación extraen reglas que mejor explican las relaciones observadas entre variables en los datos. Estas reglas pueden descubrir asociaciones importantes y comercialmente útiles, en grandes conjuntos de datos multidimensionales que pueden ser explotados por una organización.

4.3.8. Algoritmos de redes neurales artificiales

Las redes neuronales artificiales son modelos inspirados en la estructura y/o función de las redes neuronales biológicas. Son una clase de búsqueda de patrones que se utilizan comúnmente para problemas de regresión y clasificación. Es realmente un enorme subcampo compuesto de cientos de algoritmos y variaciones para todo tipo de tipos de problemas. Se ha separado el aprendizaje profundo de las redes neuronales debido a su enorme crecimiento y popularidad.

4.3.9. Algoritmos de aprendizaje profundo

Los métodos de aprendizaje profundo son una moderna actualización de las redes neuronales artificiales que explotan el abundante y barato poder de computación. Se ocupan en construir redes neuronales mucho más grandes y complejas y, muchos métodos se refieren a problemas de aprendizaje semi-supervisados donde grandes conjuntos de datos contienen muy pocos datos etiquetados.

4.3.10. Algoritmos de reducción de dimensionalidad

Al igual que los métodos de agrupación, la reducción de la dimensionalidad busca y explora la estructura inherente en los datos, pero en este caso de una manera no supervisada o en orden a resumir o describir los datos utilizando menos información. Esto puede ser útil para visualizar datos dimensionales o para simplificar datos que luego se pueden utilizar en un método de aprendizaje supervisado. Muchos de estos métodos pueden ser adaptados para su uso en clasificación y regresión.

4.3.11. Algoritmos ensamble

Métodos de ensamble son modelos compuestos por múltiples modelos más débiles, que son entrenados independientemente y cuyas predicciones son combinadas de alguna manera para hacer la predicción general. Mucho esfuerzo se pone en qué tipos de aprendices débiles combinar y las formas en que hay que combinarlos. Esta es una clase de técnica muy poderosa y como tal es muy popular.

4.3.12. Otros algoritmos

Algoritmos no lineales como:

- Support Vector Machines

Algoritmos para tareas especiales en el proceso del aprendizaje automático:

- Feature selection algorithms
- Algorithm accuracy evaluation

- Performance measures

Algoritmos para subcampos de especialidad de aprendizaje automático:

- Computational intelligence (evolutionary algorithms, etc.)
- Computer Vision (CV)
- Natural Language Processing (NLP)
- Recommender Systems
- Reinforcement Learning
- Graphical Models

4.4. Problemas de clasificación y regresión

En los problemas de clasificación el modelo creado debe predecir la clase, tipo o categoría de la salida.

La gama de problemas de aprendizaje es claramente grande, como vimos al discutir aplicaciones. Dicho esto, los investigadores han identificado un número cada vez mayor de plantillas que se pueden usar para abordar un gran conjunto de situaciones. Son esas plantillas las que facilitan el despliegue del aprendizaje automático en la práctica y nuestra discusión se centrará en gran medida en un conjunto de elección de tales problemas. Ahora ofrecemos una lista completa de plantillas.

4.4.1. Clasificación binaria

En su forma más simple se reduce a la siguiente cuestión: dado un patrón x extraído de un dominio X, estimar qué valor asumirá una variable aleatoria binaria asociada $y \in \{\pm 1\}$ [SV08].

La clasificación binaria es probablemente el problema más estudiado en el aprendizaje automático y ha dado lugar a una gran cantidad de desarrollos algorítmicos y teóricos importantes durante el siglo pasado. En su forma más simple, se reduce a la pregunta: dado un patrón x extraído de un dominio X, estimar qué valor asumirá una variable aleatoria binaria asociada y $\in \{\pm\ 1\}$.

Por ejemplo, si se muestran imágenes de manzanas y naranjas, podemos indicar si el objeto en cuestión es una manzana o una naranja. Igualmente bien, podríamos querer predecir si un propietario de vivienda podría incumplir su préstamo dado sus datos de ingresos y su historial de crédito, o si un correo electrónico determinado es spam o jamón. La capacidad de resolver este problema básico ya nos permite abordar una gran variedad de configuraciones prácticas. Existen muchas variantes con respecto al protocolo en el que estamos obligados a hacer nuestra estimación:

4.4.2. Clasificación multiclase

Es la extensión lógica de la clasificación binaria. La principal diferencia es que ahora $y \in \{1, 2, 3.., N\}$ puede asumir un rango de valores diferentes [SV08]. El problema de estudio utiliza clasificación multiclase, donde $y \in \{Nada, Poco, Medio, Mucho\}$

La clasificación multiclase es la extensión lógica de la clasificación binaria. La principal diferencia es que ahora y ∈ {1, ... , n} puede asumir un rango de valores diferentes. Por ejemplo, es posible que deseemos clasificar un documento de acuerdo con el idioma en que fue escrito (inglés, francés, alemán, español, hindi, japonés, chino, ...). La principal diferencia con anterioridad es que el costo del error puede depender en gran medida del tipo de error que cometamos. Por ejemplo, en el problema de evaluar el riesgo de cáncer, hace una diferencia significativa si clasificamos erróneamente una etapa temprana del cáncer como saludable (en cuyo caso es probable que el paciente muera) o como una etapa avanzada de cáncer (en en qué caso es probable que el paciente sufra molestias por un tratamiento excesivamente agresivo).

4.4.3. Regression

Es otra aplicación prototípica. Aquí el objetivo es estimar una variable de valor real y ∈ R dado un patrón x (ver, por ejemplo, la Figura 1.7). Por ejemplo, podríamos querer estimar el valor de un stock al día siguiente, el rendimiento de un fabuloso semiconductor dado el proceso actual, el contenido de hierro de las mediciones de espectroscopia de masas dadas por el mineral o la frecuencia cardíaca de un atleta, dada la información del acelerómetro. Una de las cuestiones clave en las que los problemas de regresión difieren entre sí es la elección de una pérdida. Por ejemplo, al estimar los valores de stock, nuestra pérdida para una opción de venta será decididamente unilateral. Por otro lado, a un deportista aficionado solo le importaría que nuestra estimación de la frecuencia cardíaca coincida con la media real.

4.5. Machine Learning y Demand Forecasting

Publicaciones que comprenden los últimos años, presentan trabajos muy interesantes en el ámbito de Machine Learning asociado al Demand Forecasting (Pronóstico de la Demanda). Se citan a continuación algunos de los problemas afrontados en publicaciones, que dan una idea del estado del arte en la conjunción de estos dos temas:

- Mejoramiento en la precisión de la previsión de demanda de agua urbana para la ciudad de Montreal – Canadá (2017).
- Proposición de un método de control inteligente para sistemas de calefacción y refrigeración. (2017).
- Proposición de un modelo predictivo probabilístico de consumo de energía, basado en datos, para la predicción del consumo en edificios residenciales (2017).

- Revisión de diferentes modelos de predicción de la carga eléctrica con un enfoque particular en modelos de regresión (2017).
- Aplicación de Machine Learning en la nube para encontrar conversaciones de los consumidores que influyen en las decisiones de compras (2016).
- Modelado de la demanda turística de España (2016).
- Predicción del Mercado de Valores (2016).
- Análisis para un minorista en línea: Previsión de la demanda y optimización de precios (2016).
- Demanda de calefacción residencial basado en el consumo total mensual de gas natural (2015).
- Predicción de la demanda de importación de crudo en Taiwán (2014).
- Predicción del desempeño de las estrategias de pronóstico para la demanda de repuestos navales (2012).

4.6. Modelado de clasificación multiclase [WFH11]

Para desarrollar un modelo o esquema de machine learning para resolver problemas de clasificación multiclase, es necesario conocer los componentes esenciales que la forman.

4.6.1. Ejemplos o instancias

La entrada de un esquema de aprendizaje automático es un conjunto de instancias. Estas instancias son las cosas que deben ser clasificadas, asociadas o agrupadas. En el escenario estándar, cada instancia es un ejemplo individual e independiente del concepto que se debe aprender. Para el problema de estudio el proceso de Business Intelligence es quien provee las instancias.

4.6.2. Características o atributos

Las instancias son caracterizadas mediante los valores de un conjunto predeterminado de atributos. Cada instancia proporciona una entrada al aprendizaje automático y es caracterizado por los valores de un conjunto fijo y predefinido de características o atributos.

4.6.3. Etiquetas

Las cantidades nominales tienen valores que son símbolos distintos. Los valores mismos sirven como etiquetas o nombres, de ahí el término nominal, que viene de la palabra latina para nombre. Los atributos nominales a veces se llaman categorizados, enumerados o discretos.

4.6.4. Conjunto de entrenamiento

El grupo de ejemplos utilizados en el proceso de entrenamiento de los algoritmos de aprendizaje automático constituyen el conjunto de entrenamiento.

4.6.5. Algoritmos de clasificación multiclase

Constituye el conjunto de algoritmos de machine que soportan problemas de clasificación multiclase.

Hipótesis, Parámetros, Función de costo, Objetivo.

Funcion Objetivo (f), Variables de entrada (X), Variable de salida (Y). Y = f(X)

4.6.6. Conjunto de prueba

Para predecir el rendimiento de un clasificador sobre nuevos datos, necesitamos evaluar su tasa de error en un conjunto de datos que no desempeñó ningún papel en la formación del clasificador. Este conjunto de datos independiente se denomina conjunto de prueba.

4.7. Algoritmos de clasificación en WEKA

Weka es una colección de algoritmos de aprendizaje automático para tareas de minería de datos. Los algoritmos pueden ser aplicados directamente a un conjunto de datos o llamados desde código Java. Weka contiene herramientas para pre-procesamiento de datos, clasificación, regresión, clustering, reglas de asociación y visualización. También es adecuado para desarrollar nuevos esquemas de aprendizaje automático [atUoW]. En el problema de estudio se utiliza el conjunto de algoritmos de clasificación de Weka [WFHP16]. Los algoritmos de clasificación de Weka que se utilizarán son los siguientes [htt]:

4.7.1. Clasificadores bayesianos

NaiveBayes implementa el clasificador probabilístico Naive Bayes. NaiveBayesSimple usa la distribución normal para modelar atributos numéricos. NaiveBayes puede usar estimadores de densidad del núcleo, que mejoran el rendimiento si la suposición de normalidad es groseramente incorrecta; También puede manejar atributos numéricos mediante discretización supervisada.

4.7.1.1. BayesNet

BayesNet constituye la clase Java base para un clasificador Bayes Network (Red Bayesiana). Aprende utilizando diversos algoritmos de búsqueda y medidas de calidad. Proporciona estructuras de datos (estructura de red, distribuciones de probabilidad condicional, etc.) y facilidades comunes a los algoritmos de aprendizaje Bayes Network tales como K2 y B [BFH$^+$16].

4.7.1.2. NaiveBayes

NaiveBayes constituye la clase Java base para un clasificador Naive Bayes usando estimadores de clases. Los valores de precisión del estimador numérico se eligen basándose en el análisis de los datos de entrenamiento. Por esta razón, el clasificador no es un UpdateableClassifier (que en el uso típico se inicializan con cero las instancias de entrenamiento) [JL95].

4.7.1.3. NaiveBayesMultinomial

NaiveBayesMultinomial constituye la clase Java para construir y utilizar un clasificador Naive Bayes multinomial. La ecuación central para este clasificador: $P\left[C_i|D\right] = \left(P\left[D|C_i\right] xP\left[C_i\right]\right) / P\left[D\right]$ (regla de Bayes), donde C_i es la clase i y D es un documento [MN98].

4.7.1.4. NaiveBayesMultinomialUpdateable

NaiveBayesMultinomial constituye la clase Java para construir y utilizar un clasificador Naive Bayes multinomial. La ecuación central para este clasificador: $P\left[C_i|D\right] = \left(P\left[D|C_i\right] xP\left[C_i\right]\right) / P\left[D\right]$ (regla de Bayes), donde C_i es la clase i y D es un documento. Es la versión incremental del algoritmo [MN98].

4.7.1.5. NaiveBayesUpdateable

NaiveBayesUpdateable constituye la clase Java para un clasificador Naive Bayes utilizando estimadores de clases. Esta es la versión actualizable de NaiveBayes. Este clasificador utiliza una precisión predeterminada de 0.1 para atributos numéricos cuando se invoca *buildClassifier* con cero instancias de entrenamiento [JL95].

4.7.2. Basado en funciones

Los algoritmos incluidos en la categoría de funciones incluyen un grupo variado de clasificadores que se pueden escribir como ecuaciones matemáticas de una manera razonablemente natural. Otros métodos, como los árboles de decisión y las reglas, no pueden (hay excepciones: Naive Bayes tiene una formulación matemática simple).

4.7.2.1. Logistic

Logistic constituye la clase Java para construir y utilizar un modelo multinomial de regresión logística con un estimador de cresta. Si hay k clases para n instancias con m atributos, la matriz de parámetros B a calcular será una matriz $m * (k - 1)$. Aunque la Regresión Logística original no se ocupa de los pesos de las instancias, se modifica el algoritmo para manejar los pesos de las instancias [lCvH92].

4.7.2.2. MultilayerPerceptron

MultilayerPerceptron es un clasificador que utiliza *back propagation* para clasificar instancias. Esta red puede ser monitorizada y modificada durante el tiempo de entrenamiento. Los nodos de esta red son todos *sigmoides* (excepto cuando la clase es numérica, en cuyo caso los nodos de salida se convierten en unidades lineales sin umbrales).

MultilayerPerceptron es una red neuronal que se entrena usando propagación posterior. Aunque se menciona en las funciones, difiere de los otros esquemas porque tiene su propia interfaz de usuario.

4.7.2.3. SimpleLogistic

SimpleLogistic constituye un clasificador para la construcción de modelos de regresión logística lineal. LogitBoost con funciones de regresión simples como base de aprendizaje se utiliza para ajustar los modelos logísticos. El número óptimo de iteraciones LogitBoost a realizar es validación cruzada, lo que conduce a la selección automática de atributos [LHF05] [SFH05].

4.7.2.4. SMO

Constituye el algoritmo Sequential Minimal Optimization para el entrenamiento de un clasificador Support Vector. Implementa el algoritmo de optimización mínima secuencial de John Platt. Esta implementación reemplaza globalmente todos los valores perdidos y transforma los atributos nominales en binarios. También normaliza todos los atributos por defecto. (En ese caso, los coeficientes de la salida se basan en datos normalizados y no en los datos originales). Los problemas multiclase se resuelven utilizando la clasificación Pairwise (aka 1-vs-1). En el caso multiclase, las probabilidades predichas se acoplan utilizando el método de acoplamiento Pairwise de Hastie y Tibshirani [Pla98] [KSBM01] [HT98].

4.7.3. Clasificadores perezosos (basados en instancia)

Los aprendices perezosos almacenan las instancias de entrenamiento y no realizan ningún trabajo real hasta el momento de la clasificación. El estudiante perezoso más simple es el clasificador k-nearest-neighbor, que implementa IBk. Se puede usar una variedad de algoritmos de búsqueda diferentes para acelerar la tarea de encontrar los vecinos más cercanos.

4.7.3.1. IBk

Clasificador K-nearest neighbours (K vecinos más cercanos). Puede seleccionar el valor apropiado de K basado en la validación cruzada. También se puede hacer ponderación de distancias [AK91].

4.7.3.2. KStar

K* es un clasificador basado en instancias, es decir, la clase de una instancia de prueba se basa en la clase de aquellas instancias de entrenamiento similares a ella, según lo determinado por alguna función de similitud. Se diferencia de otros esquemas de aprendizaje basados en instancia en que utiliza una función de distancia basada en entropía [CT95].

4.7.3.3. LWL

Locally Weighted Learning (Aprendizaje ponderado localmente) utiliza un algoritmo basado en instancias para asignar pesos de instancias que luego son utilizados por un especificado *WeightedInstancesHandler*. Puede hacer clasificación (por ejemplo, utilizando Naive Bayes) o regresión (por ejemplo, utilizando Linear Regression) [FHP03] [AMS96].

4.7.4. Meta algoritmos

Los algoritmos de Metalearning toman clasificadores y los convierten en aprendices más poderosos. Un parámetro especifica el o los clasificadores base; otros especifican el número de iteraciones para esquemas iterativos tales como bagging y boosting y una inicial semilla para el generador de números aleatorios.

4.7.4.1. AdaBoostM1

AdaBoostM1 constituye la clase Java para impulsar un clasificador de clase nominal utilizando el método Adaboost M1. Sólo se pueden abordar problemas de clase nominal. A menudo mejora dramáticamente el rendimiento, pero a veces sobreajusta [FS96].

4.7.4.2. AttributeSelectedClassifier

La dimensionalidad de los datos de entrenamiento y de prueba se reduce mediante la selección de los atributos antes de pasarlos a un clasificador.

4.7.4.3. Bagging

Bagging constituye la clase Java para capturar un clasificador que reduce la varianza. Puede hacer clasificación y regresión [Bre96].

4.7.4.4. ClassificationViaRegression

ClassificationViaRegression constituye la clase Java para hacer clasificación utilizando métodos de regresión. La clase es binarizada y se construye un modelo de regresión por cada valor de clase [FWI+98].

4.7.4.5. CVParameterSelection

CVParameterSelection constituye la clase Java para realizar la selección de parámetros mediante validación cruzada, para cualquier clasificador [Koh95a].

4.7.4.6. FilteredClassifier

FilteredClassifier constituye la clase Java para ejecutar un clasificador arbitrario en datos que se han pasado a través de un filtro arbitrario. Al igual que el clasificador, la estructura del filtro se basa exclusivamente en los datos de entrenamiento, y las instancias de prueba serán procesadas por el filtro sin cambiar su estructura.

4.7.4.7. IterativeClassifierOptimizer

IterativeClassifierOptimizer elige el mejor número de iteraciones para un IterativeClassifier tal como LogitBoost, utilizando validación cruzada. Optimiza el número de iteraciones del clasificador iterativo utilizando la validación cruzada.

4.7.4.8. LogitBoost

LogitBoost constituye la clase Java para realizar una regresión logística aditiva. Realiza clasificación utilizando un esquema de regresión como base del aprendizaje, y puede manejar problemas multiclase [FHT98].

4.7.4.9. MultiClassClassifier

MultiClassClassifier constituye un metaclasificador para manejar conjuntos de datos multiclase con clasificadores de 2 clases. Este clasificador también es capaz de aplicar códigos de salida de corrección de errores para aumentar la precisión.

4.7.4.10. MultiClassClassifierUpdateable

MultiClassClassifierUpdateable constituye un metaclasificador para manejar conjuntos de datos multiclase con clasificadores de 2 clases. Este clasificador también es capaz de aplicar códigos de salida de corrección de errores para aumentar la precisión. El clasificador base debe ser un clasificador actualizable.

4.7.4.11. MultiScheme

MultiScheme constituye la clase Java para seleccionar un clasificador entre varios, utilizando validación cruzada en los datos de entrenamiento. El rendimiento se mide en función del porcentaje de aciertos (clasificación) o del error medio cuadrático (regresión).

4.7.4.12. RandomCommittee

RandomCommittee constituye la clase Java para construir un conjunto aleatorizado de clasificadores base. Cada clasificador base se construye utilizando una semilla de números aleatorios diferentes (pero basado en los mismos datos). La predicción final es un promedio directo de las predicciones generadas por los clasificadores base individuales.

4.7.4.13. RandomizableFilteredClassifier

RandomizableFilteredClassifier constituye la clase Java para ejecutar un clasificador arbitrario en datos que han pasado a través de un filtro arbitrario. Al igual que el clasificador, la estructura del filtro se basa exclusivamente en los datos de entrenamiento, y las instancias de prueba serán procesadas por el filtro sin cambiar su estructura.

4.7.4.14. RandomSubSpace

Este método construye un clasificador basado en árbol de decisión que mantiene la mayor precisión en los datos de entrenamiento y mejora la precisión de generalización a medida que crece en complejidad. El clasificador consta de múltiples árboles construidos sistemáticamente mediante selección pseudoaleatoria de subconjuntos de componentes del vector de características, es decir, árboles construidos en subespacios elegidos aleatoriamente [Ho98].

4.7.4.15. Stacking

Stacking combina varios clasificadores utilizando el método de apilamiento. Puede hacer clasificación o regresión [Wol92].

4.7.4.16. Vote

Vote constituye la clase Java para combinar clasificadores. Se dispone de diferentes combinaciones de estimaciones de probabilidad para la clasificación [Kun04] [KHDM98].

4.7.4.17. WeightedInstancesHandlerWrapper

Envoltorio genérico alrededor de cualquier clasificador para permitir soporte de instancias ponderadas (weighted instances). Utiliza el remuestreo con pesos si el clasificador base no implementa la interfaz *weka.core.WeightedInstancesHandler* y hay otros pesos de instancias 1.0 presentes. De forma predeterminada, los datos de entrenamiento se pasan al clasificador base si puede manejar pesos de instancia. Sin embargo, es posible forzar el uso del remuestreo con pesos también.

4.7.5. Sistema de reglas

4.7.5.1. DecisionTable

Constituye la clase Java para la construcción y uso de un clasificador de mayoría de una tabla de decisión simple [Koh95b].

DecisionTable crea un clasificador de tablas de decisiones. Evalúa los subconjuntos de características utilizando la mejor búsqueda primero y puede usar la validación cruzada para la evaluación (Kohavi 1995b). Una opción es utilizar el método del vecino más cercano para determinar la clase para cada instancia que no está cubierta por una entrada de tabla de decisión, en lugar de la mayoría global de la tabla, basado en el mismo conjunto de características.

4.7.5.2. JRip

Implementa el aprendizaje de reglas proposicionales "Repeated Incremental Pruning to Produce Error Reduction" (RIPPER) o "Poda Incremental Repetida" para producir reducción de errores. Fue propuesto por William W. Cohen como una versión optimizada de IREP [Coh95].

4.7.5.3. OneR

Constituye la clase Java para construir y utilizar un clasificador 1R. Utiliza el atributo de error mínimo para la predicción, discretizando los atributos numéricos [Hol93].

4.7.5.4. PART

Utiliza dividir y conquistar. Construye un árbol de decisión C4.5 parcial. Constituye la clase Java para generar una lista de decisiones PART. Crea un árbol de decisión C4.5 parcial en cada iteración y convierte la "mejor" hoja en una regla [FW98].

4.7.5.5. ZeroR

Constituye la clase Java para construir y usar un clasificador 0-R. Predice la media (para una clase numérica) o la moda (para una clase nominal).

4.7.6. Árboles de decisión

De los clasificadores de árboles en WEKA, el J4.8 reimplementa C4.5. Puede construir un árbol binario en lugar de uno con varias ramas.

4.7.6.1. DecisionStump

Constituye la clase Java para construir y utilizar un tocón de decisión. Generalmente se utiliza en conjunción con un algoritmo de boosting. Realiza regresión (basado en el error cuadrático medio) o clasificación (basado en la entropía).

4.7.6.2. HoeffdingTree

Un árbol Hoeffding (VFDT) es un algoritmo de inducción de árbol de decisión incremental que es capaz de aprender de flujos de datos masivos, suponiendo que la distribución de la generación de los ejemplos no cambian con el tiempo. Los árboles Hoeffding explotan el hecho de que una pequeña muestra puede a menudo ser suficiente para elegir un atributo de división óptimo. Esta idea está apoyada matemáticamente por el límite de Hoeffding, que cuantifica el número de observaciones (en nuestro caso, ejemplos) necesarios para estimar algunas estadísticas dentro de una precisión prescrita (en nuestro caso, la bondad de un atributo). Una característica teóricamente atractiva de Hoeffding Trees no compartida por otros aprendizajes por árboles de decisión incremental es que tiene garantías sólidas de rendimiento. Utilizando el límite de Hoeffding se puede demostrar que su salida es asintóticamente casi idéntica a la de un aprendizaje no incremental usando infinidad de ejemplos [HSD01].

4.7.6.3. J48

Constituye la clase Java para generar un árbol de decisión C4.5 podado o no podado [Qui93].

4.7.6.4. LMT

"Árboles de Modelos Logísticos" o "Logistic Model Trees" (LMT). Clasificador para la construcción de árboles de modelos logísticos, que son árboles de clasificación con funciones de regresión logística en las hojas. El algoritmo puede manejar variables binarias y multiclases, atributos numéricos y nominales y valores faltantes [LHF05] [SFH05].

4.7.6.5. RandomForest

Constituye la clase Java para construir un "Bosque de Árboles Aleatorios" o "Forest of Random Trees" [Bre01].

4.7.6.6. RandomTree

Constituye la clase Java para construir un árbol que considera K atributos elegidos al azar en cada nodo. No realiza poda. También tiene una opción que permite la estimación de probabilidades de clase (o media objetivo en el caso de regresión) basado en un conjunto de retención (backfitting).

4.7.6.7. REPTree

Aprendizaje rápido con árboles de decisión. Construye un árbol de decisión/regresión utilizando la información de ganancia/varianza y la elimina utilizando poda de reducción de errores (con backfitting-ajuste posterior). Sólo ordena valores para atributos numéricos. Los valores faltantes se tratan dividiendo las instancias correspondientes en fragmentos (es decir, como en C4.5).

4.8. Evaluación del aprendizaje

La evaluación es la clave para lograr avances reales en el aprendizaje automático. Entre las técnicas de evaluación se destacan la Validación Cruzada (Cross-Validation) y la Validación Cruzada k-pliegues Estratificado (Stratified k-fold Cross-Validation).

La técnica de Cross-Validation consiste en dividir los datos en un número de pliegues o particiones, si por ejemplo elegimos cuatro, entonces cada partición se utiliza para las pruebas y las demás para el entrenamiento, al repetir este proceso 4 veces se consigue que cada partición se haya utilizado una vez como conjunto de pruebas.

La técnica estándar para predecir la tasa de error es Stratified k-fold Cross-Validation, donde la estratificación se refiere al proceso de reorganizar los datos de tal manera a asegurar que cada pliegue sea una buena representación del conjunto. Comúnmente se acepta que 10 es el número de pliegues con el que se obtiene la mejor estimación de error, idea basada en diversas pruebas sobre conjuntos de datos diferentes y para distintas técnicas de aprendizaje [WFH11].

Otra técnica es el Porcentaje de División (Percentage Split) con el que puede retener para la prueba un determinado porcentaje de los datos. Es una alternativa utilizar un conjunto de pruebas separado o una división porcentual de los datos de entrenamiento. Si elegimos 60 % como porcentaje de división, entonces el conjunto de prueba se constituirá con el 40 % de las instancias y el conjunto de entrenamiento con el 60 % de las instancias.

4.9. Métricas de desempeño [WFH11]

Para los problemas de clasificación, es natural medir el rendimiento de un clasificador en términos de la tasa de error (error rate). El clasificador predice la clase de cada instancia: si es correcta se cuenta como un éxito, sino se cuenta como un error. La tasa de error es sólo la proporción de errores cometidos sobre un conjunto de instancias, y mide el rendimiento general del clasificador. Por supuesto, lo que nos interesa es el probable desempeño futuro en nuevos datos, no el rendimiento pasado en datos antiguos.

Para predecir el rendimiento de un clasificador en nuevos datos, necesitamos evaluar su tasa de error en un conjunto de datos que no desempeñó ningún papel en la formación del clasificador. Este conjunto de datos independiente se denomina conjunto de prueba. En tales situaciones se suele hablar de tres conjuntos de datos: los datos de entrenamiento, los datos de validación y los datos de prueba.

Los datos de entrenamiento son utilizados por uno o más esquemas de aprendizaje para conocer clasificadores. Los datos de validación se utilizan para optimizar los parámetros de los clasificadores, o para seleccionar uno determinado. A continuación, los datos de prueba se utilizan para calcular la tasa de error del método final optimizado. Cada uno de los tres conjuntos debe ser independiente: El conjunto de validación debe ser diferente del conjunto de entrenamiento para obtener un buen desempeño en la etapa de optimización o selección y el conjunto de pruebas debe ser diferente de ambos para obtener una estimación confiable de la

tasa de error real.

4.9.1. Aciertos

Número de instancias correctamente clasificadas.

4.9.2. Porcentaje de Aciertos

Porcentaje de instancias correctamente clasificadas.

4.9.3. Estadística Kappa (Kappa Statistic)

En problemas de clasificación para aplicaciones reales normalmente los errores cuestan diferentes cantidades. Por ejemplo en bancos y financieras el costo de prestar a una persona que no paga sus deudas es mayor que el costo de rechazar un préstamo a una persona que es pagadora. Los Verdaderos Positivos (True Positive - TP) y Verdaderos Negativos (True Negative - TN) son clasificaciones correctas. Un Falso Positivo (False Positive - FP) es cuando el resultado se predice incorrectamente como sí (o positivo) cuando es realmente no (o negativo). Un Falso Negativo (False Negative - FN) es cuando el resultado se predice incorrectamente como negativo cuando es realmente positivo. En la predicción multiclase, cada elemento de la matriz de confusión muestra el número de ejemplos de prueba para los que la clase real es la fila y la clase prevista es la columna. Son buenos resultados los grandes números en la diagonal principal e idealmente cero fuera de la diagonal principal. "Kappa se utiliza para medir el acuerdo entre la predicción y la observación de las categorizaciones de un conjunto de datos, mientras que se corrige para un acuerdo que ocurre por casualidad". Si los evaluadores están totalmente de acuerdo Kappa alcanza un valor máximo igual a 1. Si no hay total acuerdo entre los evaluadores, entonces Kappa tiene un valor < 1.

$$\kappa = \frac{Pr\,(a) - Pr\,(e)}{1 - Pr\,(e)} \tag{4.1}$$

Donde: $Pr\,(a)$ es el acuerdo observado relativo entre los observadores y $Pr\,(e)$ es la probabilidad hipotética de acuerdo al azar utilizando los datos observados para calcular las probabilidades de que cada observador clasifique aleatoriamente cada categoría.

4.9.4. Sensibilidad (Recall)

Calcula la sensibilidad con respecto a una clase en particular, esto se define como: positivos correctamente clasificados / positivos totales.

$$Recall = \frac{TP}{TP + FN} \tag{4.2}$$

4.9.5. Precisión (Precision)

Calcula la precisión con respecto a una clase en particular, esto se define como: positivos correctamente clasificados / total predicho como positivo.

$$Precision = \frac{TP}{TP + FP} \tag{4.3}$$

4.9.6. Puntuación-F (F-Measure)

La Puntuación-F es una medida de la exactitud de una prueba. La Puntuación-F puede interpretarse como un promedio ponderado de la precisión y sensibilidad, donde alcanza su mejor valor en 1 y el peor en 0. Se define como: 2 * Recall * Precision / (Recall + Precision).

$$F - Measure = \frac{2 * Recall * Precision}{(Recall + Precision)} \tag{4.4}$$

Capítulo 5

Modelado del problema

5.1. Planteamiento del problema

Esta sección se centra en los tres primeros componentes de Business Intelligence siguiendo el proceso de modelado dimensional[Kim92]. Uno de los principales problemas con los que se enfrentan las empresas retail[1], trata acerca de pronosticar la demanda, es decir, determinar la cantidad de productos que se deben disponer para satisfacer la demanda de los clientes en el siguiente periodo. Utilizando los conceptos y herramientas de Business Intelligence, se parte con el análisis de una fuente de información auténtica obtenida de una empresa retail; se diseña el datawarehouse que será poblado con información de las transacciones operacionales diarias del negocio; posteriormente se definen los indicadores claves de rendimiento como métricas, se calculan los valores de los indicadores con los datos históricos almacenados en el datawarehuose para cada periodo de tiempo establecido y se asigna una etiqueta a cada tupla de valores KPI que finalmente servirán como parámetros de entrada de las herramientas de aprendizaje automático para diseñar una solución que pronostique la demanda.

5.1.1. Fuente de Información

Para el presente trabajo, se cuenta con una base de datos relacional Oracle 10g con las operaciones transaccionales de una empresa retail dedicada a la venta de productos alimenticios y artículos de limpieza, algunas de las líneas de productos con que cuenta la empresa son: aceites corporales, acondicionadores, aromatizantes, cuidado corporal, desodorantes, limpiadores, salud e higiene, salud y belleza, aguas, gaseosas, cervezas, vinos, chocolates, galletitas, enlatados, lácteos, yerbas y varias líneas de productos más. La base de datos almacena datos de la operaciones comprendidas entre noviembre de 2013 y octubre de 2016. A continuación una reseña de las principales tablas tenidas en cuenta para el diseño del datawarehouse.

- **Tabla de Productos**: almacena datos como descripción, unidad de medida, categoría,

[1]Una empresa retail es cualquier comercio que vende sus productos al consumidor final, desde un supermercado a una tienda de barrio, desde un negocio de electrodomésticos a una franquicia textil, ya sea con cientos de puntos de venta o con un solo establecimiento.

tipo de impuesto, línea del producto, marca, código de barras, proveedor, costo, precio de venta, fecha de registro, etc., de los productos disponibles para la venta, la tabla cuenta con 13.200 artículos registrados.

- **Tabla Proveedores**: almacena datos como denominación, dirección, teléfono, ruc, email, página web, ciudad, país, propietario, etc., de los proveedores de la empresa, la tabla cuenta con 1.623 proveedores registrados.
- **Tabla de Ventas Cabecera**: es una de las tablas principales donde se registran los movimientos de ventas de la empresa. Contiene datos como número de factura, moneda, tipo de comprobante, caja, usuario, fecha, cliente, monto total, monto gravado, monto impuesto, monto exenta, etc., la tabla cuenta con 301.316 registros de ventas correspondientes al periodo mencionado previamente.
- **Tabla de Ventas Detalle**: contiene los registros de los productos que fueron comercializados, cada detalle está relacionado a un registro de la tabla venta cabecera. Contiene datos de la fecha, el producto, precio de costo unitario, precio de venta unitario, cantidad, importe grabado, importe del impuesto entre otros datos, la tabla cuenta con 981.402 detalles de ventas registrados en el periodo mencionado previamente.
- **Tabla de Movimientos de Stock:** contiene los registros de movimientos de stock de ventas y compras detallado. Contiene datos de fecha, producto, cantidad, tipo de movimiento, costo unitario, precio unitario, entre otro datos, la tabla cuenta con 1.062.440 movimientos de compra y ventas registradas.

5.1.2. Proceso ETL.

En el proceso ETL posterior a la extracción y previa a la carga de los datos al datawarehouse fueron realizadas algunas transformaciones y limpieza sobre los datos de origen.

- **Tabla de Productos**: se detectaron registros de artículos con las siguientes inconsistencias:
  - Datos del proveedor con valores nulos, los cuales eran completados con un proveedor por defecto de la tabla dimensional de proveedores.
  - Artículos con datos de costo nulo, en tales casos los valores eran asignados con un costo promedio tomados de la tabla de Ventas Detalle.
  - Artículos con datos donde el costo eran mayores al precio de venta unitario, en dichos casos los datos fueron completados con el costo promedio tomados de la tabla de Ventas Detalle.
  - Artículos cuyo precio de venta unitario era nulo, en los cuales los datos eran completados con el precio de venta mas reciente tomado de la tabla de Ventas Detalle.

- **Tabla de Ventas Cabecera**: se encontraron registros donde los datos del cliente eran nulos, en dichos casos fueron asignados un cliente por defecto tomados de la tabla dimensional de clientes.
- **Tabla de Ventas Detalle**: se detectaron registros con las siguientes falencias:
  - Registros de detalles donde los valores de costo eran iguales a cero, los cuales eran modificados por el costo promedio de la tabla dimensional de productos.
  - Registros de detalle donde el costo unitario eran mayores al precio de venta unitario, las cuales fueron corregidas con el costo promedio, tomados de la tabla dimensional de productos.

5.1.3. Datawarehouse

El datawarehouse se diseña partiendo de la definición de las tablas de hechos y dimensiones utilizando el modelo en esquema estrella [Kim92].

Tablas de hechos

De las tablas transaccionales se definen 3 tablas de hechos a partir de las cuales se definen y obtienen los valores de los KPI que luego se utilizarán para el modelado de la solución propuesta.

- **Tabla de hechos Cabecera:** almacena los datos históricos de las ventas, cada registro de la tabla de hechos guarda datos de: fecha, cliente, caja, número de factura y monto total, monto exento, monto gravado IVA. Las métricas definidas para la tabla de hechos son: monto total, monto exento, monto gravado IVA.
- **Tabla de hechos Detalles:** almacena los datos históricos en detalle por cada producto vendido, cada registro guarda información del número de comprobante, fecha, producto, proveedor, cliente, cantidad, costo unitario, precio unitario, impuesto e importe total. Las métricas asociadas a la tabla de hechos son: cantidad, precio unitario, impuesto, costo unitario y el importe total.
- **Tabla de hechos Stock:** almacena los datos históricos de cada movimiento ya sea compra o de venta realizada, cada registro contiene información como fecha, producto, tipo de movimiento, cantidad, precio unitario y costo unitario. Las métricas definidas para la tabla de hechos son: cantidad, precio unitario y costo unitario.

Dimensiones

Las tablas de dimensiones diseñadas para el modelado del datawarehouse y que se encuentran relacionadas a las tablas de hechos son:

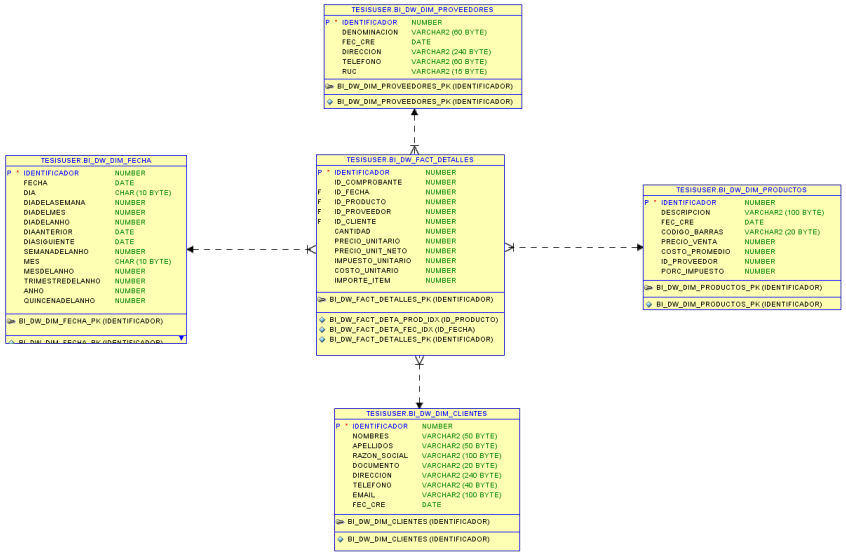


Figura 5.1: Tabla de hechos Detalles

- **Dimensión Fecha:** La tabla de dimensión fecha está ligada a todas las tablas de hechos, sirve para limitar o agrupar los datos de las tablas de hechos al momento de realizar consultas sobre estas en el tiempo. Con la dimensión fecha se pueden establecer niveles jerárquicos en días, semanas, meses, trimestres, semestres y años.
- **Dimensión Productos:** La tabla de dimensión producto está relacionada a las tablas de hechos Detalles y Stock, contiene los atributos o campos (ej.: nombre del producto) por la cual se pueden filtrar o agrupar datos al realizar consultas sobre la tabla de hechos.
- **Dimensión Proveedores:** La tabla de dimensión proveedores está relacionada a la tabla de hechos Detalles, contiene los atributos o campos (ej.: nombre del proveedor) por la cual se puede filtrar o agrupar datos al realizar consultas sobre la tabla de hechos.
- **Dimensión Clientes:** La tabla de dimensión Clientes está relacionada a las tablas de hechos Cabecera y Detalles, contiene los atributos o campos (ej.: nombre del cliente) por la cual se pueden filtrar o agrupar datos al realizar consultas sobre la tabla de hechos.
- **Dimensión Cajas:** La tabla de dimensión Cajas está relacionada a la tabla de hechos Cabecera, contiene los atributos o campos (ej.: número de caja) por la cual se pueden filtrar o agrupar datos al realizar consultas sobre la tabla de hechos.

En la siguiente figura observamos un ejemplo del modelo en esquema estrella de la tabla de hechos Detalles.

5.1.4. Definición de los Indicadores Claves de Rendimiento (KPI)

En el marco de este trabajo, en esta sección se definen los KPIs y se obtienen los valores que serán utilizados como datos de entrada en las herramientas de aprendizaje automático para el modelado de una solución para pronosticar la demanda y que servirá de apoyo en

la toma de decisiones para la reposición de stock del siguiente periodo[Alv13] (Ej.: cantidad a comprar para satisfacer la demanda de la siguiente semana, quincena, o mes), los KPIs definidos mas adelante fueron adaptados a la solución planteada debido a que en los textos consultados los KPI engloban a toda la organización en áreas como compras, ventas, marketing, recursos humanos y otros. Cada KPI mide un valor obtenido de los datos históricos almacenados en el datawarehouse. El cálculo de cada valor se realiza para cada producto y en un periodo de tiempo (semanal, quincenal o mensual), es decir, cada producto tendrá un valor distinto para cada uno de los KPI citados a continuación.

Ticket Medio. Es la cantidad media por cada transacción de venta que se realiza de un determinado producto. El indicador viene determinado por dos variables: La cantidad total vendida del producto y el total de tickets en las que fue vendido el producto. Aplicando la siguiente fórmula obtenemos el valor de la cantidad media de venta para cada producto.

$$X = \frac{\sum (Cantidad)}{Total\,Tickets\,Periodo} \tag{5.1}$$

Cifra de Ventas La cifra de ventas es un KPI que sirve para explicar el importe total de ventas que se ha obtenido para un producto. Se obtiene de la siguiente fórmula.

$$X = \sum (Precio * Cantidad) \tag{5.2}$$

Margen Comercial Es la razón entre el precio de venta y precio de costo del producto, es un indicador que permite conocer el porcentaje de rentabilidad del producto. Se obtiene de la siguiente fórmula.

$$X = \frac{\sum ((Precio - Costo) * Cantidad)}{\sum (Precio * Cantidad)} * 100 \tag{5.3}$$

Rotación de Stock Este indicador mide la cantidad de veces que el stock del producto se renueva durante un determinado ciclo comercial, es decir, la cantidad de veces que se recupera la inversión. Se obtiene de la siguiente fórmula.

$$X = \frac{\sum (Total\,Ventas\,Periodo)}{\left(\frac{Stock\,Inicial - Stock\,Final}{2}\right)} \tag{5.4}$$

Coeficiente de Rentabilidad El indicador mide la rentabilidad obtenida por la empresa basada en el margen y la rotación, el objetivo de toda empresa retail es aumentar los niveles de rotación. El coeficiente se obtiene de la siguiente fórmula.

$$X = \left(\sum (Precio - Costo) * Cantidad\right) * Rotacion\,Stock \tag{5.5}$$

Cobertura de Stock Este indicador muestra el periodo de tiempo (habitualmente se expresa en días o semanas) que el negocio puede continuar vendiendo con el stock de que dispone en el momento, sin incorporar nuevas cantidades de ese producto.

$$X = \frac{Stock\,Actual}{Promedio\,Cantidad\,Venta\,Ultimos\,3\,Periodos} \tag{5.6}$$

5.1.5. Cálculo de valores para los Indicadores Clave de Desempeño.

Definidos los KPI a ser utilizados, se procede a obtener los valores de cada KPI para cada producto y periodo de los datos almacenados en las tablas de hechos del datawarehouse, para ello las fórmulas descritas en la sección anterior se codifican a sentencias SQL y los resultados obtenidos de la ejecución fueron almacenamos en una tabla. Además de los valores de los KPI, en cada registro adicionalmente se guarda la información de la cantidad, fecha, año, mes, quincena y semana. Durante el cálculo de los valores de los KPI se establecieron ciertas restricciones, si un producto no fue vendido durante un número consecutivo de periodos esta era descartada, ya que los cálculos para cada KPI daban como resultado un valor cero, el cual no tiene relevancia para el modelado.

PERIODOS DE TIEMPO

Agrupamos el conjunto de los valores de los KPI de cada producto en 3 periodos de tiempo: semanal, quincenal y mensual.

SEMANAL En la figura 5.2se puede observar un extracto de los valores obtenidos para un producto por el rango de tiempo semanal.

El campo SEMANA se completa de acuerdo al periodo calculado, en el campo PERIODO, se guarda el valor S para semanal.

KPI_TIKET_MEDIO	KPI_CIFRA_VENTAS	KPI_MARGEN_COMERCIAL	KPI_ROTACION_STOCK	KPI_COEF_RENTABILIDAD	KPI_COBERTURA_STOCK	CANTIDAD	FECHA	ANHO	MES	QUINCENA	SEMANA	ID_PRODUCTO	PERIODO
5000	25000	7590,952380952381	0,7962307962307961	3477,416477616582	1,2	6	01/12/2013	2013	12		49	133	S
5000	20000	5904,761904761919	0,5714285714285771	3374, 14965986395	2,07692307692308	4	08/12/2013	2013	12		50	133	S
5000	15000	4428,571428571413	0,2727272727272723	1207,782207792221	0,9375	3	15/12/2013	2013	12		51	133	S
5000	10000	2952,380952380905	0,125	380,6479190476719	4,25	2	22/12/2013	2013	12		52	133	S
5000	10000	2952,380952380905	0,1428571428571443	421,7697074820983	5	2	29/12/2013	2013	12		53	133	S
5000	5000	1476,190476190468	0,08	110,9523809523839	5,571428571428577	1	01/01/2014	2014	1		1	133	S
5000	15000	4428,571428571413	0,2857142857142886	1265,306122448988	7,2	3	05/01/2014	2014	1		2	133	S
5000	20000	5904,761904761919	0,5714285714285771	3374, 14965986395	4,5	4	12/01/2014	2014	1		3	133	S
5000	20000	5904,761904761919	1,3333333333333333	7873,015873015871	1,875	4	19/01/2014	2014	1		4	133	S
5000	10000	2952,380952380905	0,2	590,4761904761919	0,2727272727272723	2	26/01/2014	2014	1		5	133	S

Figura 5.2: Tabla de Valores KPI Semanal

QUINCENAL En la figura 5.3 se puede observar un extracto de los valores obtenidos para un producto por el rango de tiempo quincenal.

El campo QUINCENA se completa de acuerdo al periodo calculado, en el campo PERIODO, se guarda el valor Q para quincenal.

	KPI_TIKET_MEDIO	KPI_CIFRA_VENTAS	KPI_MARGEN_COMERCIAL	KPI_ROTACION_STOCK	KPI_COEF_RENTABILIDAD	KPI_COBERTURA_STOCK	CANTIDAD	FECHA	ANHO	MES	QUINCENA	SEMANA	ID_PRODUCTO	PERIODO
1	5000	45000	13285,714285714	2	2857,142857142	1,2	9	01/12/2013	2013	12	23			133 Q
2	5000	35000	10333,333333333	0,7777777777778	8037,037037037	0,789473684210526	7	16/12/2013	2013	12	24			133 Q
3	5000	35000	10333,333333333	0,7368421052631	7614,035087719	1,625	7	01/01/2014	2014	1	1			133 Q
4	5000	35000	10333,333333333	0,56	5786,666666667	0,782608695652174	7	16/01/2014	2014	1	2			133 Q
5	5000	35000	10333,333333333	0,4516129032258	4666,666666667	1,714285714285	7	01/02/2014	2014	2	3			133 Q
6	5000	15000	4428,571428571413	0,2857142857142886	1265,306122448908	1,714285714285571	3	16/02/2014	2014	2	4			133 Q
7	5000	5000	1476,190476190468	0,1176470588235259	173,6696477307115	1,166315136413705	1	01/03/2014	2014	3	5			133 Q
8	0	0	0	0	0	2,181818181818181	0	16/03/2014	2014	3	6			133 Q
9	0	0	0	0	0	6	0	01/04/2014	2014	4	7			133 Q
10	2500	5000	1476,190476190468	0,1333333333333333	196,8253968253397	24	1	16/04/2014	2014	4	8			133 Q
11	5000	10000	2952,380952380905	0,3333333333333333	984,1269841269864	21	2	01/05/2014	2014	5	9			133 Q
12	5000	10000	2952,380952380905	0,5	1476,190476190468	5	2	16/05/2014	2014	5	10			133 Q
13	5000	10000	2952,380952380905	1	2952,380952380905	1,8	2	01/06/2014	2014	6	11			133 Q

Figura 5.3: Tabla de Valores KPI Quincenal

MENSUAL En la figura 5.4 se puede observar un extracto de los valores obtenidos para un producto por el rango de tiempo mensual.

El campo MES se completa para todos los casos del periodo calculado, en el campo PERIODO, se guarda el valor M para mensual.

KPI_TIKET_MEDIO	KPI_CIFRA_VENTAS	KPI_MARGEN_COMERCIAL	KPI_ROTACION_STOCK	KPI_COEF_RENTABILIDAD	KPI_COBERTURA_STOCK	CANTIDAD	FECHA	ANHO	MES	QUINCENA	SEMANA	ID_PRODUCTO	PERIODO
5000	60000	23616,047619047619	1,882352941176471	44436,3037351014	1,2	16	01/12/2013	2013	12			133	M
5000	70000	29666,666666666667	0,875	18000,3333333333	1,5	14	01/01/2014	2014	1			133	M
5000	50000	14761,904761904762	0,714285714285714	10544,2178078748	1,425	10	01/02/2014	2014	2			133	M
5000	5000	1476,1904761904762	0,117647058823529	170,666667707715	0,075	1	01/03/2014	2014	3			133	M
2500	5000	1476,1904761904762	0,133333333333333	196,8253966025397	0,96	1	01/04/2014	2014	4			133	M
5000	20000	5904,761904761905	0,8	4722,8095238095252	1,75	4	01/05/2014	2014	5			133	M
5000	15000	4428,571428571743	3	13285,7142857143	1,5	3	01/06/2014	2014	6			133	M
0	0	0	0	0	-0,375	0	01/07/2014	2014	7			133	M
0	0	0	0	0	-0,428571428571428	0	01/08/2014	2014	8			133	M
5000	80000	8571,428571428557	0,363636363636364	3116,883116883312	19	8	01/09/2014	2014	9			133	M
5000	70000	20000	2	40000	7	14	01/10/2014	2014	10			133	M

Figura 5.4: Tabla de Valores KPI Mensual

5.1.6. Asignación de etiquetas

A cada tupla de valores KPI obtenidos para cada producto se le debe asignar una etiqueta, el cual es uno de los puntos focales más importantes para el modelado mediante el aprendizaje automático. Para una mayor fiabilidad esta asignación de etiquetas debe ser realizada y revisada por el experto del área de compras (que podría ser el gerente de administración de compras u otra persona a cargo de la reposición de stock), sin embargo para el presente trabajo el etiquetado fue realizado en forma empírica, sin la intervención de un experto por la dificultad de contar con una persona especializada en el área. La estrategia utilizada para el etiquetado es de la siguiente manera:

Para cada KPI se definen un rango de valores y se asigna una letra (a, b, c ,d, e, f, g, h, i, ..., u) de acuerdo al valor obtenido.

Tabla 5.1: **RANGO KPI TICKET MEDIO**

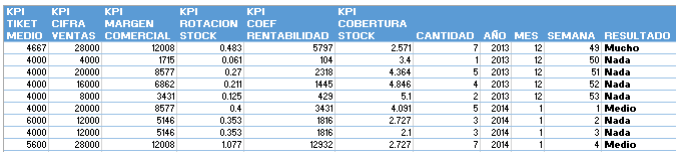
(=) igual a 0	a
> (mayor) a 0 y < (menor) a 1	b
>= (mayor o igual) a 1 y <= (menor o igual) a 3	c
> (mayor) a 3	d

Tabla 5.2: **RANGO KPI CIFRA VENTAS (%)**

>= (mayor o igual) a 0 y <= (menor o igual) a 20	e
> (mayor) a 20 y <= (menor o igual) a 50	f
> (mayor) a 50 y <= (menor o igual) a 80	g
> (mayor) a 80 y <= (menor o igual) a 100	h

Tabla 5.3: **RANGO KPI MARGEN COMERCIAL (%)**

>= (mayor o igual) a 0 y <= (menor o igual) a 20	i
> (mayor) a 20 y <= (menor o igual) a 50	j
> (mayor) a 50 y <= (menor o igual) a 80	k
> (mayor) a 80 y <= (menor o igual) a 100	l



KPI TIKET MEDIO	KPI CIFRA VENTAS	KPI MARGEN COMERCIAL	KPI ROTACION STOCK	KPI COEF RENTABILIDAD	KPI COBERTURA STOCK	CANTIDAD	AÑO	MES	SEMANA	RESULTADO
4667	28000	12008	0.483	5797	2.571	7	2013	12	49	Mucho
4000	4000	1715	0.061	104	3.4	1	2013	12	50	Nada
4000	20000	8577	0.27	2318	4.364	5	2013	12	51	Nada
4000	16000	6862	0.211	1445	4.846	4	2013	12	52	Nada
4000	8000	3431	0.125	429	5.1	2	2013	12	53	Nada
4000	20000	8577	0.4	3431	4.091	5	2014	1	1	Medio
6000	12000	5146	0.353	1816	2.727	3	2014	1	2	Nada
4000	12000	5146	0.353	1816	2.1	3	2014	1	3	Nada
5600	28000	12008	1.077	12932	2.727	7	2014	1	4	Medio

Figura 5.5: Etiquetado de KPI del periodo semanal

Tabla 5.4: **RANGO KPI ROTACIÓN STOCK**

(=) igual a 0	m
> (mayor) a 0 y < (menor) a 1	n
>= (mayor o igual) a 1 y <= (menor o igual) a 3	o
> (mayor) a 3	p

Tabla 5.5: **RANGO KPI COBERTURA STOCK**

(=) igual a 0	q
> (mayor) a 0 y < (menor) a 1	r
>= (mayor o igual) a 1 y <= (menor o igual) a 3	s
> (mayor) a 3 y <= (menor o igual) a 10	t
> (mayor) a 10	u

Una vez asignado las letras “a”, “b”, “c”, “d”, “e” hasta “u” se busca la combinación de letras correspondientes en la tabla de etiquetado realizado por el experto y se asigna el valor de la etiqueta correspondiente.

Tabla 5.6: **TABLA DE ETIQUETADO POR EL EXPERTO**

aeimq	Nada	bejnq	Poco	bejoq	Poco
aeimr	Nada	bejnr	Poco	bejpq	Medio
aeims	Nada	bejns	Nada	beknq	Poco
aeimt	Nada	bejnt	Nada	beknr	Poco
aeimu	Nada	bejnu	Nada	bekns	Nada
...	...	...	...	...	...

Una vez finalizado el etiquetado de la totalidad de las tuplas de KPI por cada producto y periodo, lo resultados son exportados a archivos con extensión csv, para cada producto se crea 3 archivos, uno por cada periodo (semanal, quincenal, mensual) que tiene como nombre el Identificador del producto y que contiene los valores de los resultados para los KPI. Estos archivos son los datos que sirven como entrada para crear el modelo de pronóstico para la reposición de stock mediante algoritmos de aprendizaje automático.

En la figura a continuación un ejemplo del etiquetado para los valores de los KPI correspondientes al periodo semanal.

5.2. Resumen de la sección

Hemos visto como los componentes de Business Intelligence ayudaron a obtener el conjunto de valores para los KPI a partir de los datos históricos almacenados en el dataware y el etiquetado a cada tupla de valores KPI, estos datos serán utilizados como un conjunto de entrenamiento por los algoritmos de aprendizaje automático del cual se obtendrá un modelo de solución al problema de estudio planteado de pronóstico de la demanda.

Capítulo 6

Experimentación

6.1. Experimentación

Se describirá cómo es la implementación del proceso de aprendizaje automático para este caso de estudio. Se mostrará primeramente cómo está constituida la salida del proceso de Business Intelligence, que en esencia proveen las instancias necesarias para la entrada del proceso de aprendizaje automático. También se verá qué clasificadores fueron utilizados, cómo se realizó el proceso de entrenamiento y de evaluación, y cuáles son las métricas de evaluación consideradas para medir el rendimiento de los clasificadores.

Para resumir la técnica propuesta, en la siguiente figura se ilustra el mapa mental general.

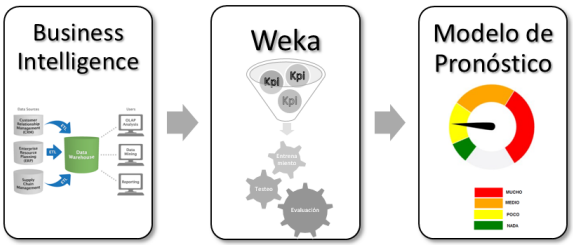


Figura 6.1: Mapa mental general

6.1.1. Datos proveídos por Business Intelligence

La salida de Business Intelligence provee tres conjuntos de datos independientes que se corresponden con los períodos de análisis: Mensuales, Quincenales y Semanales.

- Períodos Mensuales: Se analizaron 309 productos diferentes y por cada producto se tiene un máximo de 34 instancias. Cada instancia tiene los siguientes atributos: Ticket Medio, Cifra de Ventas, Margen Comercial, Rotación de Stock, Coeficiente de Rentabilidad, Cobertura de Stock, Cantidad, Año, Mes. La clase de cada instancia está definido por $y \in \{Nada, Poco, Medio, Mucho\}$.

- Períodos Quincenales: Se analizaron 228 productos diferentes y por cada producto se tiene un máximo de 68 instancias. Cada instancia tiene los siguientes atributos: Ticket Medio,

Cifra de Ventas, Margen Comercial, Rotación de Stock, Coeficiente de Rentabilidad, Cobertura de Stock, Cantidad, Año, Quincena. La clase de cada instancia está definido por $y \in \{Nada, Poco, Medio, Mucho\}$.

- Períodos Semanales: Se analizaron 127 productos diferentes y por cada producto se tiene un máximo de 151 instancias. Cada instancia tiene los siguientes atributos: Ticket Medio, Cifra de Ventas, Margen Comercial, Rotación de Stock, Coeficiente de Rentabilidad, Cobertura de Stock, Cantidad, Año, Semana. La clase de cada instancia está definido por $y \in \{Nada, Poco, Medio, Mucho\}$.

6.1.2. Esquema general de procesamiento

Se implementa el siguiente esquema de procesamiento con el dataset:

- Instancias de Períodos Mensuales: Por cada producto, se realiza el entrenamiento y testeo de sus instancias con todos los algoritmos de clasificación posibles, luego se analizan las métricas de desempeño arrojadas por cada algoritmo y finalmente se elige el mejor clasificador que será utilizado para la predicción de la demanda en períodos mensuales futuros.

- Instancias de Períodos Quincenales: Por cada producto, se realiza el entrenamiento y testeo de sus instancias con todos los algoritmos de clasificación posibles, luego se analizan las métricas de desempeño arrojadas por cada algoritmo y finalmente se elige el mejor clasificador que será utilizado para la predicción de la demanda en períodos quincenales futuros.

- Instancias de Períodos Semanales: Por cada producto, se realiza el entrenamiento y testeo de sus instancias con todos los algoritmos de clasificación posibles, luego se analizan las métricas de desempeño arrojadas por cada algoritmo y finalmente se elige el mejor clasificador que será utilizado para la predicción de la demanda en períodos semanales futuros.

6.1.3. Entrenamiento y evaluación

Como se mencionó en el esquema general de procesamiento, el entrenamiento y testeo se realiza con todos los algoritmos de clasificación posibles, para ello se utiliza la herramienta WEKA y los algoritmos de clasificación que implementa según la tabla **??**. Otra forma de categorizar los clasificadores incluidos en WEKA es como sigue:

- Bayesianos: BayesNet, NaiveBayes, NaiveBayesUpdateable.

- Basados en funciones: Logistic, MultilayerPerceptron, SimpleLogistic, SMO.

- Basados en reglas: OneR, DecisionTable, JRip, PART, ZeroR.

- Basados en árboles: DecisionStump, J48, LMT, RandomForest, RandomTree, REPTree.

En el siguiente pseudocódigo se presenta la estrategia de aprendizaje y selección de los clasificadores.

Algoritmo 6.1 Pseudocódigo para el proceso de clasificación.

```
for cada período de análisis {mensual,quincenal,semanal}:
    for cada producto con sus instancias:
        establecer conjunto de entrenamiento;
        establecer conjunto de testeo;
        for cada algoritmo de clasificación:
            construir clasificador (conjunto de entrenamiento);
            evaluar clasificador (conjunto de testeo);
            obtener métricas de evaluación;
        endfor;
        criterios de línea de base (ZeroR, criterios del experto u otro);
        seleccionar mejor clasificador (max(Kappa));
        guardar clasificador;
    endfor;
endfor;
```

La evaluación se hace por el método Stratified k-fold Cross Validation para un valor de k igual a 10 y las métricas de desempeño consideradas son el *Porcentaje de Aciertos* y la *Estadística Kappa*. El criterio de línea de base utilizado fue el clasificador ZeroR, el cual es uno de los criterios de línea de base más representativos para problemas de clasificación. Se considera que también puede resultar conveniente que el experto en compras establezca su propio criterio de línea de base, como puede ser un umbral mínimo de porcentaje de aciertos aceptado.

Por cada producto y período de análisis se elige como clasificador aquel que haya alcanzado el mayor valor de *Kappa*. En la siguiente figura se muestra como quedó la distribución de clasificadores para períodos mensuales. Por ejemplo, el clasificador *Logistic* resultó una mejor solución para 149 productos.

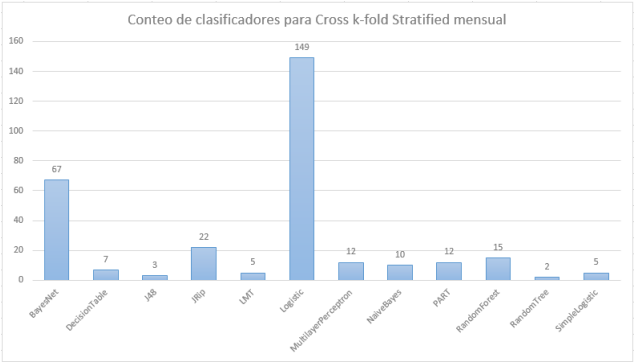


Figura 6.2: Conteo de clasificadores para período mensual.

Se puede observar en la siguiente gráfica de barras que la técnica propuesta alcanza altos porcentajes de aciertos en promedio, tanto para periodos mensuales, quincenales como semanales.

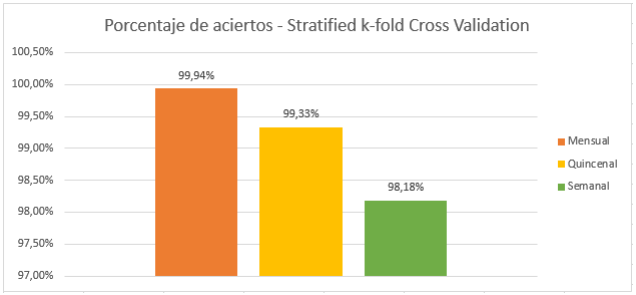


Figura 6.3: Promedio de porcentaje de aciertos para los tres períodos de análisis.

Como se trata de una prueba exhaustiva, por cada producto se intenta con todos los algoritmos de clasificación posible y se evalúa con el método Stratified 10-fold Cross Validation. Estas métricas de desempeño preliminares dan indicio de que la técnica propuesta en este trabajo puede alcanzar altos grados de confiabilidad. Obtener buenos resultados depende en gran medida de que los valores de *KPI* hayan sido obtenidos correctamente y también que el etiquetado haya sido realizado por un experto en compras.

6.1.4. Cómo hacer las predicciones

En el siguiente pseudocódigo se muestra el mecanismo para obtener el pronóstico de la demanda en un ambiente de producción.

Algoritmo 6.2 Pseudocódigo para el proceso de pronóstico de la demanda.

```
for cada próximo período a pronosticar {mensual,quincenal,semanal}:
      for cada producto:
            obtener KPIs del período actual finalizado;
            ejecutar su mejor clasificador (KPIs);
            obtener etiqueta {nada,poco,medio,mucho}
            extrapolar a valores continuos(criterio experto);
      endfor;
endfor;
```

Como se mencionó anteriormente, el modelo propuesto arroja como resultado un valor discreto $y \in \{Nada, Poco, Medio, Mucho\}$. Luego en función de la etiqueta resultante, del tipo de producto y del período seleccionado, el experto extrapola a un valor continuo que representa la cantidad en la orden de compra. El significado de las etiquetas varía

6.2. Discusión

6.2.1. Impacto del período de análisis

Una de las decisiones que se debe tomar es acerca del tiempo asignado al período de análisis. En este trabajo se analizaron tres períodos distintos: mensuales, quincenales y semanales con propósitos experimentales y por ser los más comunes en el ámbito comercial. En la práctica, la elección del período es una decisión estratégica a nivel gerencial que depende en gran medida del sector y tamaño de la empresa, tipos de productos, etc.

En el presente trabajo, por tratarse de períodos de tiempo muy cercanos (1, 2 y 4 semanas) no se observan diferencias significativas en el porcentaje de aciertos. Otro factor a tener en cuenta es que para períodos de tiempo muy extensos (6, 12 meses) existe mayor incertidumbre en el pronóstico.

6.2.2. Impacto del etiquetado

La técnica propuesta se trata de un sistema parametrizado, donde las variables principales son el período comercial y las etiquetas seleccionadas para la clasificación. Por cuestiones de practicidad y generalidad se eligió para este trabajo un enfoque de problema de clasificación. El etiquetado proporciona mayor flexibilidad al sistema y un entorno más controlable, en comparación a un sistema de asignación de valores continuos. La flexibilidad del sistema permitió emular la opinión del experto en compras y encontrar una cantidad eficiente de etiquetas.

Capítulo 7

Conclusiones

7.1. Conclusiones

Este trabajo se enfocó en proponer una nueva técnica de estimación de la demanda de productos, para reposición de stock en empresas retail. Como se mencionó en la Sección 2, la gestión de compras es uno los ejes centrales en la actividad empresarial y la decisión del volumen de compras para cada producto es un desafío que enfrentan las empresas al momento de reponer el stock. Partiendo de esta premisa y analizando las técnicas de pronóstico de la demanda empleadas en la actualidad, y el creciente incremento del uso de tecnologías de Business Intelligence en las organizaciones, se encontró la oportunidad de desarrollar una nueva técnica de pronóstico. En esta nueva técnica se utilizan los Indicadores Claves de Rendimiento y apoyados en la experiencia de un experto en compras (gerente o encargado de compras) se realiza el modelado utilizando algoritmos de clasificación de Machine Learning.

De acuerdo a los resultados experimentales se obtuvieron altas tasas de aciertos, haciendo pruebas exhaustivas con varios algoritmos de clasificación y evaluando con un método ampliamente aceptado. La técnica propuesta pretende que este nuevo modelo se convierta en una herramienta de apoyo en la toma de decisiones del gerente de compras en el proceso de reposición de stock.

Bibliografía

[AK91] D. Aha and D. Kibler. Instance-based learning algorithms. *Machine Learning*, 6:37–66, 1991. 36

[Alv13] Marcos Alvarez. *Cuadro de Mando Retail*. Profit, 2013. 23, 49

[AMS96] C. Atkeson, A. Moore, and S. Schaal. Locally weighted learning. *AI Review*, 1996. 37

[ASW+11] David R. Anderson, Dennis J. Sweeney, Thomas A. Williams, Jeffrey D. Camm, and Kipp Martin. *Métodos cuantitativos para los negocios*. © D.R. 2011 por Cengage Learning Editores, S.A. de C.V., una compañía de Cengage Learning, Inc, 11 edition, 2011. 8, 9, 13

[atUoW] Machine Learning Group at the University of Waikato. Weka 3: Data mining software in java. 34

[BFH+16] Remco R. Bouckaert, Eibe Frank, Mark Hall, Richard Kirkby, Peter Reutemann, Alex Seewald, and David Scuse. *WEKA Manual for Version 3-8-0*, April 2016. 34

[Bre96] Leo Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996. 37

[Bre01] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001. 41

[Can07] Josep Lluís Cano. *Busines Intelligence: Competir con información*. ESADE, Banesto, Banesto Pyme, 2007. 14, 15, 17, 21

[Coh95] William W. Cohen. Fast effective rule induction. In *Twelfth International Conference on Machine Learning*, pages 115–123. Morgan Kaufmann, 1995. 40

[CT95] John G. Cleary and Leonard E. Trigg. K*: An instance-based learner using an entropic distance measure. In *12th International Conference on Machine Learning*, pages 108–114, 1995. 37

[EH05] Wayne W. Eckerson and Cindi Howson. Enterprise business intelligence: Strategies and technologies for deploying bi on an enterprise scale tdwi report series. 2005. 15, 22

[FHP03] Eibe Frank, Mark Hall, and Bernhard Pfahringer. Locally weighted naive bayes. In *19th Conference in Uncertainty in Artificial Intelligence*, pages 249–256. Morgan Kaufmann, 2003. 37

[FHT98] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. Technical report, Stanford University, 1998. 38

[FS96] Yoav Freund and Robert E. Schapire. Experiments with a new boosting algorithm. In *Thirteenth International Conference on Machine Learning*, pages 148–156, San Francisco, 1996. Morgan Kaufmann. 37

[FW98] Eibe Frank and Ian H. Witten. Generating accurate rule sets without global optimization. In J. Shavlik, editor, *Fifteenth International Conference on Machine Learning*, pages 144–151. Morgan Kaufmann, 1998. 40

[FWI+98] E. Frank, Y. Wang, S. Inglis, G. Holmes, and I.H. Witten. Using model trees for classification. *Machine Learning*, 32(1):63–76, 1998. 37

[Gar06] Gartner. Glosario de gartner, www.gartner.com, enero 2006. gartner es una consultora internacional especializada en tecnologías de información y comunicación, January 2006. 14

[HH08] Frederick S. Hillier and Mark S. Hillier. *Métodos cuantitativos para administración.* Tercera edition, 2008. 7, 8, 9, 10, 12, 13

[Ho98] Tin Kam Ho. The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832–844, 1998. 39

[Hol93] R.C. Holte. Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11:63–91, 1993. 40

[HSD01] Geoff Hulten, Laurie Spencer, and Pedro Domingos. Mining time-changing data streams. In *ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining*, pages 97–106. ACM Press, 2001. 41

[HT98] Trevor Hastie and Robert Tibshirani. Classification by pairwise coupling. In Michael I. Jordan, Michael J. Kearns, and Sara A. Solla, editors, *Advances in Neural Information Processing Systems*, volume 10. MIT Press, 1998. 36

[htt] http://weka.sourceforge.net/doc.dev/weka/classifiers/Classifier.html. Interface classifier. 34

[Inm92] W.H. Inmon. *Building the datawarehouse.* QED Press, 1992. 19

[JC10] Jordi Conesa Josep Curto. *Introducción al Business Intelligence*. Editorial UOC, 2010. 14, 16, 19, 20, 21

[JL95] George H. John and Pat Langley. Estimating continuous distributions in bayesian classifiers. In *Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 338–345, San Mateo, 1995. Morgan Kaufmann. 35

[JLF12] P. Fraser Johnson, Michiel R. Leenders, and Anna E. Flynn. *Administración de compras y abastecimientos*. McGRAW-HILL / INTERAMERICANA EDITORES, S.A. DE C.V, 2012. 5, 6, 9, 13

[KHDM98] J. Kittler, M. Hatef, Robert P.W. Duin, and J. Matas. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226–239, 1998. 39

[Kim92] Ralph Kimball. *The datawarehouse Toolkit*. John Wiley & Sons, Inc, 1992. 17, 19, 45, 47

[Koh95a] R. Kohavi. *Wrappers for Performance Enhancement and Oblivious Decision Graphs*. PhD thesis, Stanford University, Department of Computer Science, Stanford University, 1995. 38

[Koh95b] Ron Kohavi. The power of decision tables. In *8th European Conference on Machine Learning*, pages 174–189. Springer, 1995. 40

[KSBM01] S.S. Keerthi, S.K. Shevade, C. Bhattacharyya, and K.R.K. Murthy. Improvements to platt's smo algorithm for svm classifier design. *Neural Computation*, 13(3):637–649, 2001. 36

[Kun04] Ludmila I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. John Wiley and Sons, Inc., 2004. 39

[lCvH92] S. le Cessie and J.C. van Houwelingen. Ridge estimators in logistic regression. *Applied Statistics*, 41(1):191–201, 1992. 35

[LHF05] Niels Landwehr, Mark Hall, and Eibe Frank. Logistic model trees. 95(1-2):161–205, 2005. 36, 41

[MA03] L.T. Moss and S. Atre. *Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-support Applications*. Addison-Wesley information technology series. Addison-Wesley, 2003. 14

[MN98] Andrew Mccallum and Kamal Nigam. A comparison of event models for naive bayes text classification. In *AAAI-98 Workshop on 'Learning for Text Categorization'*, 1998. 35

[Pla98] J. Platt. Fast training of support vector machines using sequential minimal optimization. In B. Schoelkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*. MIT Press, 1998. 36

[PMAR07] Arley Pérez, Alberto Medina, Pavel Alonso, and Nguyen Ramírez. Métodos y técnicas para la previsión de la demanda. *Universidad de Matanzas Camilo Cienfuegos - Facultad Industrial-Economía*, 2007. 8

[Pug16] Jean Francois Puget. What is machine learning?, May 2016. 26

[Qui93] Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA, 1993. 41

[RN04] Stuart Russell and Peter Norvig. *Inteligencia Artificial. Un Enfoque Moderno. Segunda Edición*. PEARSON EDUCACIÓN, S.A., 2004. 26

[Sam59] Arthur Samuel. Some studies in machine learning using the game of checker. *IBM Journal 3, 211-229*, 1959. 25

[SFH05] Marc Sumner, Eibe Frank, and Mark Hall. Speeding up logistic model tree induction. In *9th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 675–683. Springer, 2005. 36, 41

[SV08] Alex Smola and S.V.N. Vishwanathan. *Introduction to Machine Learning*. The Press Syndicate of The University of Cambridge, 2008. 31, 32

[VAH11] Naim Caba Villalobos, Oswaldo Chamorro Altahona, and Tomás José Fontalvo Herrera. *Gestión de la Producción y Operaciones*. 2011. 7, 13

[Wat06] Hugh James Watson. Recent developments in datawarehousing: A tutorial. 2006. 19

[WFH11] Ian H. Witten, Eibe Frank, and Mark A. Hall. *Data Mining - Practical Machine Learning Tools and Techniques - Third Edition*. Copyright © 2017 Elsevier Inc. All rights reserved, tercera edition, 2011. 33, 42

[WFHP16] Ian H. Witten, Eibe Frank, Mark A. Hall, and Christopher J. Pal. *Data Mining - Practical Machine Learning Tools and Techniques - Fourth Edition*. Cuarta edition, 2016. 34

[Wol92] David H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992. 39

[Wre06] R. Wrembel. *Data Warehouses and OLAP: Concepts, Architectures and Solutions: Concepts, Architectures and Solutions*. Gale virtual reference library. IRM Press, 2006. 21