Introducción a R

con fundamentos de minería de datos

Blanca A. Vargas Govea
blanca.vg@gmail.com

13 de marzo de 2014

Contenido

Capítulo 1

Introducción

1.1. Minería de datos

1.1.1. ¿En dónde vemos la minería de datos?

Bancos: aprobación de crédito, lealtad de clientes: detectar cuáles son más probables de irse con la competencia, identificación de posibles clientes que respondan a promociones, detección de eventos fraudulentos, manufactura y producción: identificación de productos defectuosos, sistemas de recomendación: personalización; medicina: diagnósticos, relación entre enfermedades, Ítems frecuentes que la gente tiende a comprar juntas en el supermercado, filtrado colaborativo: ítems similares, usuarios similares, Películas, Amazon, Google

1.1.2. ¿Qué es?

En la actualidad existe un enorme y continuo flujo de datos que son recolectados y almacenados mediante sitios web, comercio electrónico, transacciones bancarias. La presión competitiva origina que se busquen maneras de proporcionar al usuario servicios personalizados y más acordes a sus necesidades que se refleje en un incremento en visitas/ventas.

En esa cantidad de datos a menudo existe información escondida, no implícita que puede ser muy valiosa para proporcionar al usuario lo que necesita. Comportamientos, relaciones, tendencias que tal vez ni siquiera el mismo usuario lo sepa.

Realizar este análisis de sin las técnicas adecuadas consumiría mucho tiempo.

En 1986 surgió el término: Descubrimiento de Conocimiento en Bases de Datos (KDD), proceso en el cual la minería de datos es un paso que consiste en aplicar análisis de datos y algoritmos de descubrimiento que bajo ciertas limitaciones producen un conjunto de patrones o modelos que describen los datos. Existen diversas definiciones de minería de datos, una de las clásicas y vigentes es: Extracción no trivial de información implícita, previamente desconocida y útil a partir de datos [Fayyad et al., 1996]. Las principales metas de los métodos de minería de datos son la predicción y descripción. El diagrama del proceso se ve en la Figura 1.1.

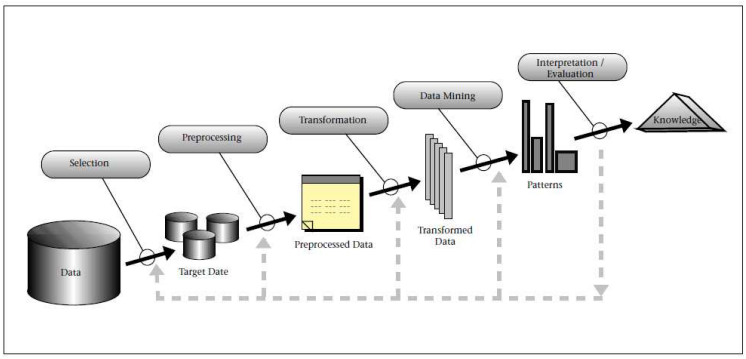


Figura 1.1: Pasos de KDD

Donde datos pueden ser por ejemplo, conjuntos de registros o textos obtenidos de sitios web, un patrón es una expresión en algún lenguaje que describe a un sub-conjunto de datos o a un modelo aplicable al sub-conjunto, por ejemplo, un conjunto de reglas. Aquí como extracción de un patrón nos referimos a ajustar un modelo a los datos, encontrar estructura a los datos; o en general, obtener cualquier descripción de alto nivel del conjunto de datos. Como no-trivial implica búsqueda o inferencia, esto es que no es un cálculo directo o valores predefinidos como el cálculo de un promedio de un conjunto de números. El término proceso incluye diversos pasos que involucran:

1. Entender el dominio de aplicación, conocimiento previo relevante e identificar la meta del proceso.

2. Seleccionar el conjunto de datos y variables.

3. Limpieza y pre-proceso de datos. Eliminar ruido si aplica, definir estrategias para los datos faltantes y tomar en cuenta información temporal y cambios conocidos.

4. Reducción de los datos. Encontrar los datos significativos para representar los datos dependiendo de la meta de la tarea.

5. Seleccionar el método apropiado para la meta: clasificación, regresión, agrupamiento,

6. Análisis exploratorio y selección del modelo e hipótesis. Seleccionar los algoritmos y modelos, los modelos sobre variables categóricas son distintos que sobre variables reales. Determinar qué se quiere, no es lo mismo obtener un modelo para entenderlo o interpretarlo que uno con capacidades de predicción.

7. Buscar patrones de interés, reglas de clasificación, árboles, regresión, agrupamiento.

8. Interpretación de los patrones. Visualización.

9. Usar el conocimiento descubierto. Incorporar el modelo en otro sistema, documentarlo, etc.

Los patrones descubiertos deben ser válidos y con cierto grado de certidumbre. Buscamos patrones que sean nóveles y potencialmente útiles, es decir, que proporcionen beneficios al usuario o tarea. Los patrones deben ser entendibles.

Al proceso completo se le conoce como KDD e incluye la recolección, abstracción y limpieza de los datos para encontrar patrones, validación y verificación de los patrones, visualización de los modelos y refinamiento del proceso de recolección. En la práctica, la parte de preparación de datos, correspondiente a los pasos 2, 3 y 4 suelen llevarse el 80 % del tiempo.

Otra definición frecuente dice que la minería de datos consiste en torturar los datos hasta que confiesen... y si torturas suficiente, puedes hacer que confiese

cualquier cosa. Aunque la definición suena divertida, la realidad es que es un riesgo pues puedes descubrir patrones que no tienen significado. Por otro lado, el principio de Bonferroni destruye la definición anterior: si buscas en más lugares de los que tu cantidad de datos puede soportar, lo más seguro es que encuentres basura.

Otros términos. Es frecuente encontrarnos con términos distintos que se utilizan para describir la actividad de analizar datos y encontrar patrones útiles, éstos son: análisis predictivo, estadística, análisis de negocios, descubrimiento de conocimiento, inteligencia de negocios, ciencia de los datos, como se muestra en la Figura 1.2.

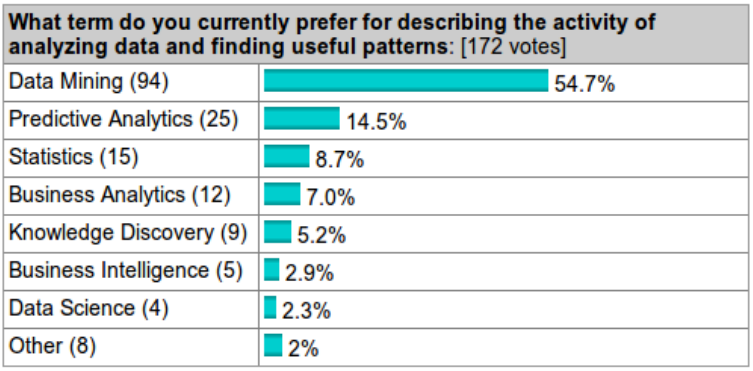


Figura 1.2: Términos para análisis de datos

De acuerdo al área de aplicación hay preferencias al uso de determinados términos. En la industria es más conocida como análisis predictivo, análisis de negocios; mientras que en el medio académico se le conoce como minería de datos, estadística, descubrimiento de conocimiento (ver Figura 1.3).

1.1.3. Origen

Aunque la minería de datos es la evolución de un campo con larga historia, el término fue introducido hasta los noventas. La minería de datos tiene sus orígenes en tres líneas. La más larga es la estadística clásica. La estadística es la base de muchas técnicas de minería de datos. La estadística clásica abarca conceptos como análisis de regresión, distribuciones, desviación estándar, varianza, análisis

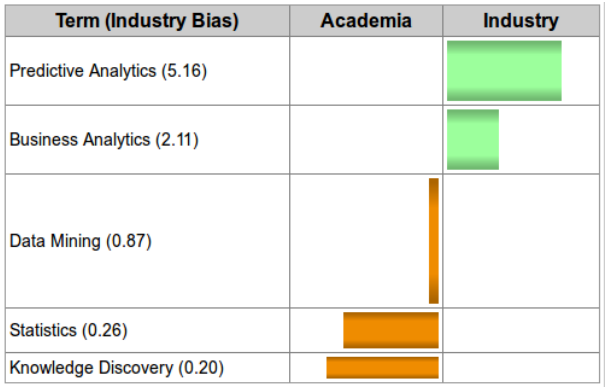


Figura 1.3: Términos para análisis de datos por área

de discriminantes, análisis de clusters, intervalos de confianza, todo lo que se usa para analizar datos y relaciones entre los datos.

La segunda área es la inteligencia artificial. Esta disciplina se construye con heurísticas, en oposición a la estadística, intenta aplicar el pensamiento humano como el procesamiento a problemas estadísticos. Como este enfoque requería gran poder de cómputo, no fue práctico hasta los 80s, cuando las computadoras empezaron a ofrecer poder útil a precios razonables. Algunos productos que adoptaron esto fueron los módulos de optimización de RDBMS Relational Database Management Systems.

La tercer área es el aprendizaje automático (*machine learning*), que es más precisamente descrita como la unión de estadística e IA. 80s y 90s, ml trata de que los programas aprendan con base en los datos que estudian. Al contrario de las técnicas estadísticas que requieren que el usuario tenga una hipótesis primero en mente, los algoritmos analizan datos e identifican relaciones entre atributos y entidades para construir los modelos que permiten a los expertos del dominio, no-estadísticos entender relaciones entre atributos y la clase. El paradigma de hipotetiza y prueba, se relajó a prueba e hipotetiza.

La minería de datos floreció a finales de los noventas y puede describirse como la unión de desarrollos históricos y actuales en estadística, IA, aprendizaje automático y visualización (Figura 1.4). Estas técnicas se usan juntas para estudiar datos y encontrar tendencias o patrones. Está teniendo gran aceptación

en la ciencia e industria para analizar grandes cantidades de datos y descubrir conocimiento que de otra forma no podría encontrarse.

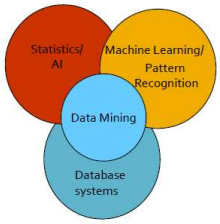


Figura 1.4: Data mining

¿Por qué si todo surgió de 30 a 40 a nos atrás es hasta ahora que estas técnicas están teniendo tanta atención. Los recursos tecnológicos actuales permiten flujos enormes de información, internet, dispositivos móviles, etc. que entonces no se tenían. Así, ha crecido hasta convertirse en lo que se conoce como: **Big data**.

Big data son datos que exceden la capacidad de procesamiento de sistemas de bases de datos convencionales. Es grande, se mueve rápido y no se ajusta a las estructuras de la bd. Big data tiene tres dimensiones: volumen, velocidad y variedad.

- Volumen - Mucha mucha mucha información. Terabytes y petabytes de información. Hadoop, MapReduce.
- Velocidad - Debe usarse como si fuera un flujo continuo para maximizar su valor.
- Variedad - Énfasis en los datos no estructurados: texto, audio, video, flujos, logs y más.

La mayoría de los procesos de minería de datos ocurren en desktops y laptops, como se ve en la Figura 1.5

Estas técnicas tienen diversas implicaciones, una de las más discutidas es la privacidad en los datos. Ética, privacidad en los datos. ¿De quién son los datos? ¿del usuario? ¿del sistema? ¿cómo garantizar que con el fin de construir mejores sistemas no se le de mal uso a los datos?

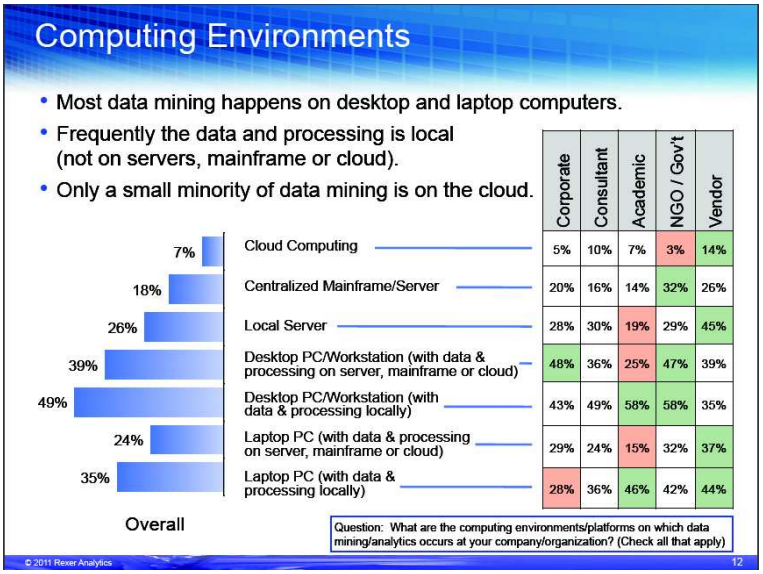


Figura 1.5: Plataformas

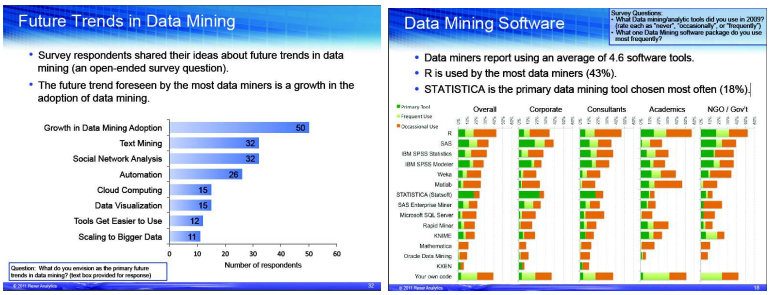
1.1.4. Tendencias

Las tendencias indican un crecimiento en la incorporación de minería a sus procesos (ver Figura 1.6). La minería de texto, el análisis de redes sociales, automatización, cómputo en la nube, visualización de datos, mejora en las herramientas y Big data se indican como aspectos principales.

En este panorama, la Figura 1.6 muestra que R es una de las herramientas para minería de datos que ha destacado recientemente. La información se obtuvo de un survey aparecido en 2011 con resultados recolectados en 2010. 735 usuarios de 60 países `http://rexeranalytics.com/`.

1.2. R

R es un lenguaje de script para manipulación de datos, análisis estadístico y visualización.



(a) Tendencias (b) Herramientas

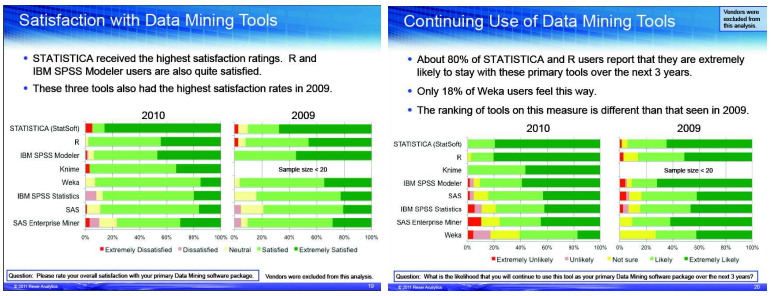
Figura 1.6: Tendencias y herramientas

1.2.1. Antecedentes

- Inspirado por el lenguaje S. Desarrollado por John Chambers en los laboratorios Bell. Inició en 1976 como ambiente interno de análisis y se implementó como librerías de Fortran.
- En 1988 se reescribió en C.
- La versión 4 se liberó en 1996 y desde 1998 no ha cambiado drásticamente. Ganó el premio ACM Software System award.
- En 1991 R fué creado por Ross Ihaka y Robert Gentleman de la Universidad de Aukland en Nueva Zelanda para facilitarse su curso de introducción al análisis de datos.
- Hay dos versiones del origen del nombre: 1) es la inicial de ambos autores y 2) por ser descendiente del lenguaje S.
- En 1995 los convencieron de poner el código disponible.
- En 1997 un grupo de voluntarios se unieron al proyecto y son ahora quienes controlan el código fuente y dirigen el proyecto. Ciclo de implementación de 6 meses.
- En el 2000 se libera la versión 1.0.0.
- 2012 Marzo 30 R version 2.15.0, 3736 paquetes. [R Development Core Team, 2010]

[Smith, 2010]
http://www.biostat.jhsph.edu/~rpeng/biostat776/lecture1.pdf

En los últimos años R se ha convertido en una de las tres herramientas de minería más satisfactorias para los usuarios. El 80 % de los usuarios de R reportaron que muy probablemente continuarían usándolo en los siguientes 3 años. Figura 1.7.



(a) Favoritas (b) Continuidad

Figura 1.7: Favoritas y continuidad

1.2.2. Ventajas

- Software libre con base en el respetado lenguaje S.
- Es comparable y a menudo superior en poder a productos comerciales.
- Disponibilidad Windows, Mac, Linux.
- Lenguaje de propósito general.
- Incorpora características encontradas en el enfoque orientado a objetos y lenguaje de programación funcional.
- El sistema almacena los conjuntos de datos entre sesiones por lo que no se necesita cargar nuevamente los datos cada vez. También guarda el historial de comandos.
- Es fácil encontrar ayuda de la comunidad, la cual provee de nuevas funciones.
- Proporciona disponibilidad instantánea de métodos nuevos y experimentales.
- Facilidad de exportar e importar datos en diversos formatos.

1.2.3. Desventajas

La principal es que no escala bien para conjuntos grandes de datos. Google y Facebook usan R a menudo como *sandbox* antes de usar otro lenguaje como C o Phyton. Los espacios de trabajo de R se almacenan en RAM por lo que son limitados. Algunas formas con las que han

Algunas soluciones que se dan: i) usar la conectividad a base de datos de R (e.g. RMySQL) y obtener solo porciones de datos, ii) obtener muestras menores, o (iii) ejecutar los scripts en la computadora de alguien obsesionado por el RAM o arrancar un servidor virtual en la nube de 15 Gigas.

http://www.dataspora.com/2009/02/predictive-analytics-using-r/

Capítulo 2

R. Lo básico: instalación y manejo de datos

2.1. Diseño conceptual

Las funciones están organizadas en librerías llamadas paquetes. R está dividido en dos partes:

1. El sistema base.
2. Los paquetes recomendados. Los paquetes se pueden descargar de CRAN, Debian, SourceForge, github, etc.

2.2. Instalación

- Instalar R [R Development Core Team, 2010] . Disponible para Linux, MacOS X y Windows:
  en `http://cran.r-project.org/mirrors.html`.
- Puede usarse en consola o instalar un IDE.
  Sugerencia: RStudio `http://www.rstudio.org/`.

El conjunto de datos que se usará es credit-g.csv [Hofmann, 1994].
En modo consola:

```
$ R
```

2.3. Cómo obtener ayuda

```
help.start()        # ayuda general
help(nombrefuncion) # detalles sobre la funcion
?funcion            # igual que el anterior
apropos("solve")    # lista las funciones que contienen "solve"
example(solve)      # muestra un ejemplo del uso de solve
help("*")
vignette()
vignette("foo")
data()              # muestra los conjuntos de datos disponibles
help(datasetname)   # detalles del conjunto de datos
```

2.4. Espacio de trabajo

Es el entorno de tu sesión actual en R e incluye cualquier objeto: vectores, matrices, dataframes, listas, funciones. Al finalizar la sesión se puede guardar el actual espacio de trabajo para que automáticamente se cargue en la siguiente.

Algunos comandos estándar para definir el espacio de trabajo son los siguientes:

```
getwd() # muestra el directorio actual
ls()    # lista los objetos en el espacio de trabajo
setwd(mydirectory)      # cambia el path a mydirectory
setwd("c:/docs/mydir")  # notar / en vez de \ en Windows
setwd("/usr/rob/mydir") # en Linuz
history() # despliega los 25 comandos recientes
history(max.show=Inf) # despliega los comandos previos
q() # quit R.
```

2.5. Paquetes

```
.libPaths() # obtiene ubicación de la librería
library()   # muestra los paquetes instalados
```

```
search()    # muestra los paquetes cargados
# descarga e instala paquetes del repositorio CRAN
install.packages("nombredelpaquete")
library(package) # carga el paquete
```

2.6. Scripts

```
# en Linux
R CMD BATCH [options] my_script.R [outfile]

# en ms windows (ajustar el path a R.exe)
"C:\Program Files\R\R-2.5.0\bin\R.exe" CMD BATCH
   --vanilla --slave "c:\my projects\my_script.R"

source("myfile")
sink("record.lis") # direcciona la salida al archivo record.lis
```

2.7. IDEs/GUIs

- RStudio,
  http://www.rstudio.org/
- StatET,
  http://www.walware.de/goto/statet/
- ESS (Emacs Speaks Statistics),
  http://ess.r-project.org/
- RapidMiner R extension,
  http://rapid-i.com/
- Tinn-R,
  http://www.sciviews.org/Tinn-R/
- Rattle GUI,
  http://rattle.togaware.com/
- JGR (Java GUI for R),
  http://cran.r-project.org/web/packages/JGR/index.html

RStudio, StatET y ESS, son IDEs, orientadas más a programación. La Figura 2.1 muestra las GUIs preferidas. La consola es la favorita.

Which R interfaces do you use frequently?	
built-in R console (225)	40%
RStudio (135)	24%
Eclipse with StatET (90)	16%
RapidMiner R extension (80)	14.2%
Tinn-R (62)	11%
ESS (Emacs Speaks Statistics) (59)	10.5%
Rattle GUI (53)	9.4%
R Commander (43)	7.7%
Revolution Analytics (31)	5.5%
RKWard (22)	3.9%
JGR (Java Gui for R) (21)	3.7%
RExcel (18)	3.2%
R via a data mining tool plugin (12)	2.1%
Red-R (8)	1.4%
SciViews-R (6)	1.1%
Other (44)	7.8%

Figura 2.1: GUIs

2.8. Datos y funciones básicas

2.9. Números

```
var1 <- 54
var1

var2 <- sqrt(var1*8)
var2
```

2.10. Vectores

```
vector <- c(1,2,3,4,5)
```

```
vector[0]
vector[1]

cadena <- "uno"
cadena

lcadena <- c("casa","manzana","uva")
lcadena

vlogico <- c(TRUE,FALSE,TRUE,TRUE,FALSE)
vlogico
```

2.11. Dataframes

```
c1 <- c(25,26,27,28)
c2 <- c("Ana", "Lola", "Luis", "Pedro")
c3 <- c(TRUE,TRUE,TRUE,FALSE)
mydata <- data.frame(c1,c2,c3)
names(mydata) <- c("ID","Nombre","Aprobado") # nombres de variables
```

Otros tipos de datos son: arrays, listas y factores.***Agregar

2.12. Exportar e importar datos

```
> setwd("path_name")
> credit <- read.csv(file = "../data/credit-g.csv", sep = ",",
                     na.strings = "NULL")
```

2.13. Despliegue de objetos

```
ls()  # lista de objetos
names(credit) # variables de credit
str(credit) # estructura de credit
```

```
levels(credit$foreign_worker) # niveles(valores de variable)
dim(credit) # dimensiones (ren x cols)
class(credit) # clase del objeto
credit  # objeto
head(credit, n=10)
tail(credit, n=10)
credit$purpose
#purpose
attach(credit)  # coloca la bd en el path
purpose
```

2.14. Renglones, columnas, mínimo y máximo

```
# Explorando conteo
ren <- nrow(credit)   # raw row count
col <- ncol(credit)

# Min y Max
agemin <- min(age,na.rm = TRUE) # min sin NULLS en age
agemax <- max(age,na.rm = TRUE)

ammin <- min(credit_amount,na.rm = TRUE) # min sin NULLS en age
ammax <- max(credit_amount,na.rm = TRUE)
```

2.15. Subconjuntos

```
# Filtrando por edad
mayor30 <- subset(credit, age >= 30)
renmayor30 <- nrow(mayor30)
renmayor30

en20y40 <- subset(credit, age >= 20  & age <= 40)
ren20y40 <- nrow(en20y40)
```

```
ren20y40
```

2.15.1. Muestra aleatoria

```
set.seed(32)
dsmall <- credit[sample(nrow(credit), 10), ]
dsmall
```

2.15.2. Definiendo rangos

```
# Filtrando por bins
agebin = cut(age,breaks = c(18,30,40,50,60,70,80))
agebinfile <- data.frame(purpose,credit_amount,personal_status,
              housing,job,age=agebin,class)
agebinfile
```

2.16. Únicos, frecuencia, orden

```
# unicos
unicos <- unique(class)
unicos
nhousing <- length(unique(class))
nhousing

# table
mycredit <- table(dsmall$class)
mycredit
mycredit <- table(dsmall$age,dsmall$credit_amount)
mycredit

# ordenar
mycredit <- table(dsmall$housing)
mycredit
mylist <- sort(mycredit,decreasing = TRUE)
mylist
```

2.17. Construyendo archivos de salida

```
oframe = data.frame(purpose,credit_amount,personal_status,
                    housing,job,age=agebin,class)
write.table(oframe, row.names = FALSE,
            sep = ";",quote = FALSE, file="../data/output01_data.csv")
```

Capítulo 3

Exploración de datos y visualización

3.1. Conceptos básicos

El análisis descriptivo o exploratorio tiene por objetivo describir los datos mediante técnicas estadísticas y gráficas. El análisis descriptivo nos da un primer acercamiento a los datos y extrae características importantes que se utilizarán para análisis posteriores. Este tipo de análisis es una herramienta útil para encontrar errores, ver patrones en los datos, encontrar y generar hipótesis. A continuación se describen conceptos básicos [Bartlein, 2009, Trochim, 2006, Quick, 2009, Wackerly et al., 2002] que se utilizan a lo largo del documento.

Población

Es el conjunto completo de elementos de nuestro interés. Las características de una población son las medidas estadísticas de cada uno de sus elementos.

Muestra

Es una proporción de la población. Una muestra posee las mismas características de la población si se obtiene aleatoriamente. Las muestras aleatorias deben cumplir con dos características: 1) todos los elementos deben tener la misma oportunidad de ser seleccionados y 2) la selección de un elemento debe ser independiente de la selección de otro.

La diferencia entre una población y una muestra es que con la población, nuestro interés es identificar sus características mientras que con la muestra, nuestro interés es hacer inferencias sobre las características de la población. Si la medida (e.g, media, mediana) se obtiene de la población completa, se le llama **parámetro** mientras que si se obtiene de una muestra, se le llama **estimación**.

Estadística descriptiva

Describe los datos sin ningún tipo de generalización. Ejemplo: porcentaje de menores de edad que utilizan redes sociales.

Inferencia estadística

Generaliza o induce algunas propiedades de la población a partir de la cual los datos se tomaron. Ejemplo: ¿es la satisfacción del usuario de sistemas de recomendación significativamente diferente entre hombres y mujeres?

Variables categóricas

No aparecen de forma numérica y tienen dos o más categorías o valores. Pueden ser nominales y ordinales. Una variable nominal no tiene un orden (e.g., rojo, amarillo, suave), mientras que la ordinal designa un orden (e.g., primer lugar, segundo lugar).

Variables numéricas

Son aquellas que pueden tomar cualquier valor dentro de un intervalo finito o infinito. Hay dos tipos de variable numérica: de intervalo y radio. Una variable de intervalo tienen valores cuyas diferencias son interpretables pero no tienen un cero verdadero, un ejemplo es la temperatura. Pueden ser sumadas o restadas pero no multiplicadas o divididas (e.g., hoy no es el doble de cálido que ayer). Una variable de radio tiene un cero verdadero y pueden sumarse, multiplicarse o dividirse (e.g., el peso).

Para cada tipo de variable hay diferentes técnicas de análisis.

El análisis puede ser **univariable**, en el cual se exploran las variables o atributos uno por uno o **bivariable**, en el cual simultáneamente se analizan dos variables para conocer la relación entre ellas, su fuerza o si hay diferencias entre éllas

y el significado de las mismas. El análisis bivariable puede ser entre dos variables numéricas, dos variables categóricas y una variáble numérica y una categórica. Los valores estadísticos de las variables son las características que se extraerán de los perfiles de usuario e ítems.

3.2. Explorando variables individuales

En este tipo de análisis se exploran las variables una por una y dependiendo de su tipo (i.e., categórica, numérica) se aplican distintos tipos de análisis y gráficas.

Para las **variables categóricas** el análisis exploratorio básicamente es un conteo del número de valores de la variable especificada y el porcentaje de valores de la variable específica. Se utilizan las gráficas de barras y de pay. Por otro lado, las **variables numéricas** se analizan calculando el mínimo, máximo, media, mediana, moda, rango, los cuantiles, la varianza, la desviación estándar, el coeficiente de variación, la asimetría y la curtosis. Se visualizan mediante histogramas y gráficas de caja.

El conjunto de medidas son las características que nos interesan para el modelo que queremos obtener por lo que es importante recordar lo que cada una representa.

Media

La media, en estadística, puede tener dos significados: la media aritmé- es una medida de tendencia central de un espacio muestral. En un conjunto de datos, es la suma de una lista dividida por el número de miembros de la lista. Se denota por $\bar{x}$ o μ y puede verse como la salida esperada $E(x)$ de un evento x tal que si la medida se realiza varias veces, el valor promedio sería la salida más común.

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i = \frac{1}{n}(x_1 + x_2 + ... + x_n)$$

La media describe la ubicación central de los datos y la desviación estándar describe la dispersión. La media de una muestra puede ser diferente de la media poblacional especialmente para muestras pequeñas pero a mayor

tama no de la muestra, es más probable que la media muestral se acerque a la media poblacional.

Es el primer momento de la distribución de una variable aleatoria. Un momento es una medida cuantitativa de la forma de un conjunto de puntos.

No es una medida muy robusta para describir la tendencia central de un conjunto de datos pues es fácilmente influenciada por valores atípicos. Para distribuciones asimétricas la media no concuerda con el centro por lo que en esos casos se prefiere la media para describir la tendencia central.

Mediana

Es el número que separa el conjunto de datos por la mitad. Para encontrarla se ordenan los números en orden ascendente o descendente y se escoge el que está a la mitad. Si la lista tiene un número par de elementos, la mediana se calcula obteniendo la media de los números centrales. Para cualquier distribución de probabilidad , la mediana satisface las siguientes desigualdades:

$$Pr(X \leq m) \geq \frac{1}{2} \text{ y } Pr(X \geq m) \geq \frac{1}{2}$$

Puede usarse como medida de localización de tendencia central cuando una distribución es asimétrica. Es igual al segundo cuartil.

Moda

La moda es el valor que ocurre con mayor frecuencia en un conjunto de datos. Puede diferir mucho de la media y mediana debido a la asimetría de las distribuciones.

Rango

El rango es la diferencia entre los valores máximo y mínimo de valores en un conjunto. Como solamente depende de dos observaciones es una medida débil de la dispersión excepto si la muestra es grande.

Varianza y desviación estándar

La varianza describe que tan lejos están los valores de la media. La varianza

y la desviación estándar (σ y σ^2 respectivamente) son indicadores de la dispersión de los datos dentro de una muestra o población. Como en el caso de la media, la varianza de la población y desviación estándar son las desviaciones esperadas. La varianza de una población se calcula usando la media de una población:

$$\sigma^2 = \frac{1}{n}\sum_{i=1}(x_i - \mu)^2$$

Para calcular la varianza muestral, encontramos los errores de todas las mediciones, ésto es, la diferencia entre cada medida y la media muestral, $x_i - \bar{x}$. Luego se eleva al cuadrado cada valor y se suman, se divide entre el número de observaciones menos 1. La desviación estándar muestral es la raíz cuadrada de la varianza muestral $\sqrt{s^2}$.

Es el segundo momento de una distribución.

Cuantiles

Son puntos tomados en intervalos regulares de la distribución acumulativa de una variable aleatoria. Dado un conjunto de datos ordenados, los cuantiles-q dividen el conjunto en q subconjuntos de igual tamaño; son los valores que marcan los límites entre subconjuntos consecutivos. Los casos especiales son: cuartiles, decentiles, centiles y percentiles.

Cuartiles. Un cuartil es cualquiera de los tres valores que dividen los datos ordenados en cuatro partes iguales. El primer cuartil (Q1) de un grupo de valores es el valor en el que cae el 25 % de los datos. El segundo cuartil Q2 es la mediana (el 50 % de los datos). El tercer cuartil Q3 de un grupo de valores es en el que cae el 75 % de los datos. La distancia entre el primero y el tercer cuartiles se conoce como el rango inter-cuartil RIC (Q3-Q1) y a veces se usa como una alternativa robusta a la desviación estándar.

Una gráfica muy útil para visualizar los cuartiles son los diagramas de caja; se les conoce también como diagramas de caja y bigotes (Figura 3.1). Este tipo de gráfica resume las siguientes medidas: mediana, cuartiles superior e inferior y valores mínimo y máximo. La caja contiene el 50 % central

de los datos. La parte superior de la caja representa el 75 percentil de los datos . La línea central representa la mediana y el primer y tercer cuartiles son las orillas de la caja. El área de la caja es el rango intercuartil. Los valores extremos, dentro de $1{,}5$ veces del rango intercuartil son los extremos de las líneas. Los puntos a mayor de $1{,}5$ veces el IQR de la mediana son potenciales valores atípicos.

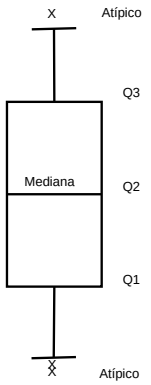


Figura 3.1: Diagrama de caja y bigotes

Valores atípicos. Un valor atípico es una observación numéricamente distante del resto de los datos y puede representar datos erróneos. Tomando como referencia la diferencia entre el primer cuartil (Q1) y el tercer cuartil Q3, o valor intercuartil, en un diagrama de caja se considera un valor atípico el que se encuentra 1,5 veces esa distancia de uno de esos cuartiles (atípico leve) o a 3 veces esa distancia (atípico extremo).

Asimetría

Es una medida de la asimetría de una distribución que se define por la siguiente fórmula:

$$\gamma_1 = \mu_3/{\mu_2}^{3/2}$$

El valor puede ser positivo, negativo o indefinido. Como regla, una asimetría negativa indica que la media de los datos es menor que la mediana y la

distribución está concentrada a la izquierda; la asimetría positiva indica que la media es mayor que la mediana y la distribución está concentrada a la derecha. Esta regla aplica solamente a distribuciones unimodales cuyos histogramas tienen un solo pico. Cualitativamente, una asimetría negativa indica que la cola del lado izquierdo de la distribución de probabilidad es más larga que la cola derecha y el grueso de los valores, incluyendo la mediana cae en el lado derecho de la media. Una asimetría positiva indica lo contrario. Un valor de cero indica que los valores son relativamente distribuidos igual a ambos lados de la media. Es el tercer momento de la media. Conocer la asimetría del conjunto de datos indica si las desviaciones de la media serán positivas o negativas.

Curtosis

Es el cuarto momento central y mide si la distribución es alta y delgada o corta y gruesa en comparación a la distribución normal de la misma varianza. Puede decirse que la curtosis mide qué tan puntiaguda es la distribución. Valores altos de curtosis indican que la varianza es el resultado de desviaciones extremas infrecuentes. Se define como:

$$\gamma_2 = \frac{\mu_4}{\delta^4} - 3$$

Una distribución con curtosis elevada tiene un pico agudo y largo, colas anchas, mientras que con baja curtosis tieen picos más redondeados y colas más delgadas y cortas.

Cargar los datos

```
> setwd("path_name")
> credit <- read.csv(file = "../data/credit-g.csv", sep = ",",
                     na.strings = "NULL")
> attach(credit)
```

Cargar librería moments

```
install.packages("moments")
library(moments)
```

3.3. Moda

```
Mode <- function (x) {
    cngtable <- table(x)
    n <- length(cngtable)
    mode <- as.double(names(sort(cngtable)[n]))
    mode
}

moda <- Mode(age)
moda
```

3.4. Rango

```
Rng <- function(x) {
    rangem <- diff(range(x))
    rangem
}

rango <- Rng(age)
rango
```

3.5. Cuantiles

```
Quantiles <- function(x) {
    quants <- quantile(x)
    quantval <- as.double(names(table(quants)))
    quantval
}

q <- Quantiles(sort(age))
q
```

3.6. Media

```
media <- round(mean(age),2)
media
```

3.7. Mediana

```
mediana <- round(median(age),2)
mediana
```

3.8. Varianza

```
varianza <- round(var(age),2)
varianza
```

3.9. Desviación estándar

```
sd <- round(sd(age),2)
sd
```

3.10. Curtosis

```
kurtosis <- round(kurtosis(age),2)
kurtosis
```

3.11. Explorando múltiples variables

Statistical analysis tool that estimates, on the basis of past (historical) data, the probability of an event occurring again.

Modelo probabilístico, es la forma que pueden tomar un conjunto de datos obtenidos de muestreos de datos con comportamiento que se supone aleatorio.

Pueden ser modelos probabilísticos discretos o continuos. Los primeros, en su mayoría se basan en repeticiones de pruebas de Bernoulli. Los más utilizados son:

Modelo de Bernoulli Modelo Binomial. Modelo Geométrico. Modelo Binomial negativo. Modelo Hipergeométrico. Modelo de Poisson.

Por otro lado, tal como se ha mencionado antes, existen modelos probabilísticos continuos, entre ellos destacamos:

Distribución Normal: usada ampliamente en muestras mayores a 30 datos. Distribución Chi Cuadrado: usada en muestras pequeñas. Distribución Exponencial: usada en duración o donde interviene el paso del tiempo. Distribución F-Snedecor: usada para controlar la varianza de 2 distribuciones.

Capítulo 4

Minería de datos

4.1. Clasificación

Las enormes cantidades de datos acumuladas en bases de datos pueden ser usadas para tomar decisiones. La clasificación y la predicción son dos formas de análisis de datos que pueden usarse para extraer modelos que describen clases o predicen tendencias futuras de los datos. Mientras que la clasificación predice etiquetas o variables categóricas, o valores discretos, los modelos predictivos lo hacen con funciones contínuas. Por ejemplo, un modelo de clasificación puede construirse para categorizar si un préstamo bancario es seguro o riesgoso, mientras que un modelo de predicción puede construirse para predecir gastos de clientes potenciales en equipo de cómputo dado su ingreso y ocupación.

Regresión lineal. En la regresión lineal, los datos se modelan usando una línea recta. La regresión lineal es la forma de regresión más simple. La regresión de dos variables modela una variable, Y, la variable de salida como función lineal de otra variable aleatoria, X, el predictor, i.e., $Y = \alpha + \beta X$

Regresión lineal en R.

Clasificación por retropropagación. La retropropagación es el algoritmo de aprendizaje de redes neuronales. El campo de las redes neuronales fue acu nado por psicólogos y neurobiólogos. Básicamente, una red neuronal es un conjunto

de unidades de entrada/salida conectadas en el que cada conexión tiene un peso asociado. Durante la fase de aprendizaje, la red aprende ajustando los pesos de tal manera que sea capaz de predecir la clase correcta de los ejemplos de entrada.

nnet: redes neuronales feed-forward

4.2. Agrupación

Al proceso de agrupar un conjunto de objetos físicos o abstractos en clases de objetos similares se le llama agrupación (clustering). Un cluster es una colección de objetos que son parecidos a otro dentro del mismo cluster y son diferentes a otros en otros clusters. Un cluster puede ser tratado colectivamente como un grupo en muchas aplicaciones.

A diferencia de la clasificación y predicción, que analizan objetos con un valor de clase, el agrupamiento analiza objetos que no tienen una clase conocida. En general, la clase no está presente por desconocerse. El agrupamiento puede usarse para generar tales etiquetas de clase. Los objetos son agrupados con base en el principio de maximización de similaridad intra-clase y minimización de la similaridad entre clases. Esto es, los clusters son formados de tal forma que dentro del cluster exista alta similaridad pero haya muy poco parecido con objetos de otros clusters. Cada cluster puede verse como una clase de objetos a partir de las cuales se pueden derivar reglas. El clustering puede facilitar la formación de taxonomías, esto es, la organización de observaciones en una jerarquía de clases.

Agrupación en R.

4.3. Reglas de asociación

La construcción de reglas de asociación encuentra asociaciones o relaciones de correlación entre conjuntos de ítems. Con las cantidades masivas de datos que contínuamente son colectados y almacenados en bases de datos, muchas industrias se han interesado en el minado de reglas de asociación. Por ejemplo, el descubrimiento de relaciones de asociación interesantes entre grandes cantidades

de transacciones registradas puede ser úti para el dise no de catálogos, mercadeo-cruzado y otros procesos de toma de decisiones. El ejemplo típico es el análisis de la canasta de mercado. Este proceso analiza los hábitos de compra de los clientes al encontrar las asociaciones entre los distintos ítems que los clientes colocan en sus bolsas o canastas de compra. El descubrimiento de tales asociaciones puede ayudar a los comerciantes a desarrollar estrategias de mercado al conocer qué ítems son comprados al mismo tiempo. Por ejemplo, si los clientes compran leche, ¿qué probabilidad hay de que también compren pan y qué tipo de pan en la misma visita al supermercado? Tal información puede llevar a un incremento en las ventas al planear qué productos se colocarán cerca en los anaqueles. Así, colocar el pan y la leche en anaqueles cercanos favorecerá las ventas de dichos artículos.

Apriori es el algoritmo para minería de reglas de asociación. En general, la minería de reglas de asociación puede verse como un proceso de dos pasos: 1) Encontrar todos los conjuntos de ítems frecuentes: Por definición, cada conjunto ocurrirá al menos tan frecuentemente como lo indique un conteo predeterminado de soporte mínimo.

2) Generar reglas fuertes de asociación de los conjuntos de datos frecuentes: Por definición, esas reglas deben satisfacer un mínimo de soporte y confianza.

arules: Reglas de asociación

Un árbol de decisión es como un diagrama de flujo con estructura de árbol en el que cada nodo interno denota una prueba en un atributo. Cada rama representa la salida de una prueba, y los nodos hoja representan clases. Para clasificar un ejemplo desconocido, los valores de atributo se prueban contra el árbol. Se traza un camino desde la raíz hasta el nodo hoja que indica la clase. Los árboles de decisión pueden convertirse fácilmente a reglas de clasificación.

Particionamiento recursivo y árboles de regresión
Clasificación y árboles de regresión

4.4. Selección de atributos

La selección de atributos es una de las técnicas más usadas en preprocesamiento de datos para minería de datos. Elimina datos relevantes, redundantes

o ruidosos y brinda aplicación inmediata. La selección de atributos ha sido un campo fértil de investigación y desarrollo desde 1970 en reconocimiento de patrones, aprendizaje automático, minería de datos y ampliamente aplicada en categorización de textos, recuperación de imágenes, detección de intrusis y análisis del genoma. [Liu and Yu, 2005]

Los conjuntos de datos para análisis pueden contener cientos de atributos, muchos de los cuales pueden ser irrelevantes o redundantes. Por ejemplo, si la tarea es clasificar clientes que son o no posibles compradores de un nuevo libro, cuando se registra una venta, los atributos del cliente tales como el número de teléfono puede ser irrelevante en contraste con atributos como la edad o las preferencias de lectura. Aunque un experto puede seleccionar atributos útiles, esto puede ser una tarea difícil y lenta. Descartar los atributos relevantes y conservar los irrelevantes puede causar confusión al algoritmo de minería empleado lo que puede generar resultados de poca calidad. Además, agregar atributos irrelevantes o redundantes puede hacer lento el proceso de minería.

La selección de atributos reduce el tama no del conjunto de datos al eliminar atributos irelevantes o redundantes. La meta de la selección de atributos es encontrar el conjunto mínimo de atributos tales que la distribución de probabilidad resultante de las clases sea tan cercana posible a la distribución original obtenida con todos los atributos. Al reducir el número de atributos se facilita también la interpretación del modelo resultante.

Es común que antes de utilizar los algoritmos se realice un proceso de selección de atributos.

Hacerl manual requiere de un profundo conocimiento del problema de aprendizaje y de lo que los atributos representan.

Beneficios [Guyon and Elisseeff, 2003]

- Mejora del desempe no predictivo.
- Reducción de tiempo de entrenamiento/proceso.
- Reducción de las necesidades de almacenamiento.
- Facilitar la visualización de los datos y comprensión de los datos.

4.4.1. Pasos

Un proceso típico de selección de atributos consiste en cuatro pasos básicos 4.1:

1. Generación de sub-conjuntos. Subset generation is a search procedure [48, 53] that produces candidate feature subsets for evaluation based on a certain search strategy. Es esencialmente un proceso de búsqueda heurística con cada estado en el espacio de búsqueda definiendo un sub-conjunto candidato para evaluación. Este proceso está definido por dos aspectos: dirección de la búsqueda (forward, backward, bi-direccional) y la estrategia de búsqueda: completa, secuencial y random.

2. Evaluación de sub-conjuntos. Each candidate subset is evaluated and compared with the previous best one according to a certain evaluation criterion. If the new subset turns out to be better, it replaces the previous best subset. PUede categorizarse en dos grupos basándose en su dependencia de algoritmos. Criterio independiente:

3. Criterio de paro. Determina cuándo el proceso de selección debe parar. Algunos criterios frecuentes son: 1) búsqueda completa, 2) se alcanza un límite que puede ser determinado por un número de atributos o iteraciones, 3) no hay mejora en el sub-conjunto al agregar o eliminar un atributo, y 4) se selecciona un sub-conjunto suficientemente bueno. The process of subset generation and evaluation is repeated until a given stopping criterion is satisfied.

4. Validación de resultados. Then the selected best subset usually needs to be validated by prior knowledge or different tests via synthetic and/or real-world data sets.

4.4.2. Categorías por criterio de evaluación

Caen dentro de las siguientes categorías:

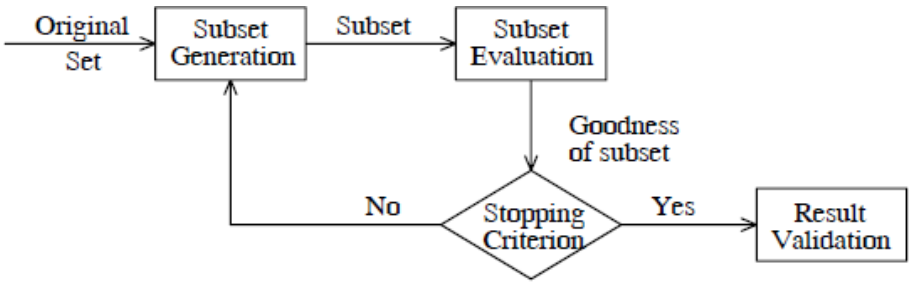


Figure 1: Four key steps of feature selection

Figura 4.1: Pasos Selección de atributos

- Filtros. Hacer una evaluación independiente con base en las características de los datos para evaluar al sub-conjunto. Operan independientemente del algoritmo de aprendizaje. Se les llama métodos de filtro porque el conjunto de atributos es filtrado para generar el subconjunto más prometedor antes de que el aprendizaje empiece.

  Filtrado. Ranking de atributos. No solamente permiten mejorar el desempe no predictivo, los rankings pueden proporcionar información sobre el mérito individual de cada atributo.

  Algunas medidas de criterio independiente son las medidas de distancias, de información, de dependencia y de consistencia.

  Métodos de ranking:

  - Ganancia de información. Es uno de los más simples y rápidos métodos de ranking y se usa a menudo en aplicaciones de categorización de texto. Si A es un atributo y C es la clase, las siguientes ecuaciones dan la entropía de la clase antes y después de observar el atributo.

$$H(C) = -\sum_{c \in C} p(c) log_2 p(c) \tag{4.1}$$

$$H(C|A) = -\sum_{a \in A} p(a) \sum_{c \in C} p(c|a) log_2 p(c|a) \tag{4.2}$$

La cantidad por la cual la entropía de la clase decrece refleja la información adicional acerca de la clase que proporciona el atributo y se le llama ganancia de información. A cada atributo A_i se le asigna un score con base en la ganancia de información entre él y la clase.

$$IG_i = H(C) - H(C|A_i) \tag{4.3}$$

$$IG_i = H(A_i) - H(A_i|C) \tag{4.4}$$

$$IG_i = H(A_i) + H(C) - H(A_i, C) \tag{4.5}$$

- Relief. Es un esquema de rankeo de atributos. Funciona muestreando aleatoriamente una instancia y ubicar sus vecinos cercanos de la misma clase y de clase opuesta. Los valores de los atributos de los vecinos cercanos son comparados a la instancia muestreada y usada para actualizar los scores de relevancia para cada atributo. El proceso se repite para un número determinado de instancias. La idea es que un atributo útil pueda diferenciar entre instancias de clases distintas y tenga el mismo palor para instancias de la misma clase.

  Se pueden tomar muestras aleatorias de instancias y buscar los vecinos con la misma clase (near-hit) clases distintas (near-miss). Si un near-hit tiene un valor de atributo diferente, entonces se considera irrelevante y su peso se decrementa. Por otra parte, si un near-miss tiene un valor de atributo diferente, el atributo se considera relevante y su peso se incrementa. Después de repetir la operación, se seleccionan solamente los atributos con pesos positivos.
- Componentes principales. Es una técnica estadística que puede reducir la dimensionalidad como producto de una transformación del espacio original de atributos y luego extraer sus eigenvectores. Los eigenvectores (componentes principales) definen una transformación lineal del espacio original de atributos a un nuevo espacio en el cual los atributos no están correlacionados.

Filtrado. Subconjuntos.

- CFS (Correlation-based Feature Selection). Es el primero de los métodos que evalúa sub-conjuntos de atributos. Es una heurística de evaluación de conjuntos que toma en cuenta la utilidad individual de los atributos para predecir la clase junto con el nivel de inter-correlación entre ellos. La siguiente heurística asigna altos scores a sub-conuntos que contienen atributos que son altamente correlacionados con la clase y tienen baja inter-correlación entre ellos. Los atributos redundantes son discriminados ya que están estrechamente correlacionados con uno m ás atributos. Como los atributos se tratan de forma independiente, CFS no puede identificar atributos que interactúan fuertemente. Después de calcular una matriz de correlación, CFS aplica una estrategia de búsqueda para encontrar un buen conjunto de atributos. Se usa una búsqueda forward-selection que produce una lista de atributos ranqueados de acuerdo a su contribución de bondad del conjunto.
- Consistency-based Subset Evaluation. Varios enfoques para selección de sub-conjuntos de atributos usan la consistencia de clase como una métrica de evaluación. Estos métodos buscan combinaciones de atributos cuyos valores dividan los datos en sub-conjuntos que contengan una clase mayoritaria. Generalmente la búsqueda favorece sub-conjuntos peque nos con alta consistencia de clase. Usa una búsqueda forward selection se usa para producir una lista de atributos, rankeados de acuerdo a su contribución total a la consistencia del conjunto de atributos.

[Hall and Holmes, 2003]

- Envoltura Evaluar los atributos usando la precisión de un algoritmo de aprendizaje. Se le conoce como método de envoltura porque el algoritmo está envuelto en el proceso de selección. Se usa validación cruzada para proporcionar un estimado de la precisión de un clasificador sobre datos nuevos. Se usa una búsqueda forward selection para producir una lista de

atributos, rankeados de acuerdo a su contribución a la precisión. Los wrappers generalmente dan mejores resultados que los filtros pero a costa de un costo computacional alto.

Algoritmos como C4.5 pueden usarse para selección de atributos. Se puede usar un árbol y el resultado usarlo en otro algoritmo de aprendizaje. Capaz que se comporta mejor que el árbol.

4.4.3. Espacio de búsqueda

Muchos métodos de selección de atributos involucran una búsqueda en el espacio de atributos para los sub-conjuntos que tienen más probabilidad de predecir mejor la clase. La figuar muestra el espacio de búsqueda para el ejemplo del clima. El número de posibles sub-conjuntos atributos incrementa exponencialmente con el número de atributos haciendo impráctica una búsqueda exhaustiva. Generalmente el espacio es buscado de forma greedy en una de dos direcciones: del top al bottom y del bottom al top. En cada estado se hace un cambio local al conjunto de atributos ya sea agregando o borrando un solo atributo.

A la dirección hacia abajo, donde se empieza sin atributos y se agrega uno a uno, se le llama selección hacia adelante (forward). La dirección hacia arriba, en la que se inicia con el conjunto completo y se van eliminando atributos uno a uno, se le llama eliminación hacia atrás (backward).

Los algoritmos greedy funcionan tomando decisiones que parecen las mejores en un momento dado, nunca se reconsidera.

En la selección hacia adelante, cada atributo que no está en el conjunto actual, se agrega y el sub-conjunto resultante es evaluado usando por ejemplo, cross-validation. Esta evaluación produce una medida numérica esperada del desempe no del sub-conjunto. El efecto de a nadir cada atributo en turno es cuantificado por esa medida, el mejor sub-conjunto es escogido y el procedimiento continúa. Sin embargo, si no hay mejora al aagregar el atributo, la búsqueda termina. Esta búsqueda garantiza un sub-conjunto óptimo local pero no un óptimo global.

La eliminación hacia atrás opera de forma análoga.

Existen esquemas de búsqueda más sofisticados que hacen una búsqueda

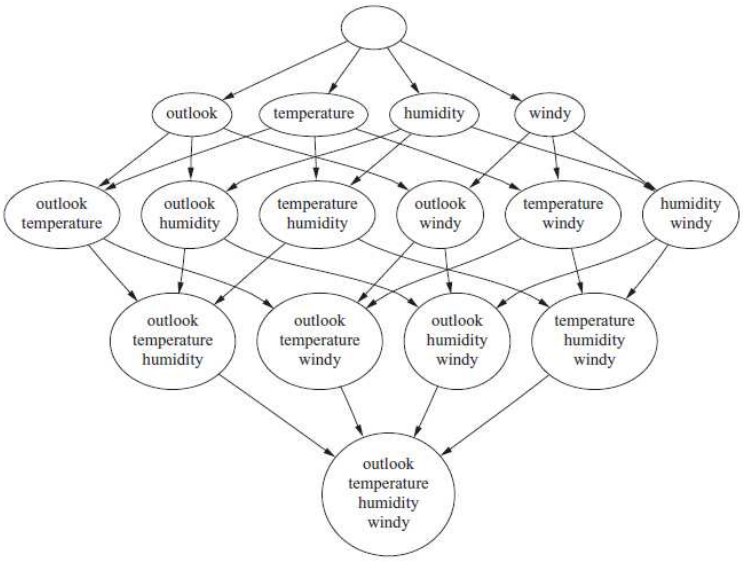


FIGURE 7.1

Attribute space for the weather dataset.

Figura 4.2: Espacio de búsqueda

bidireccional. Best-first termina cuando el desempe no empieza a decaer pero mantiene una lista de todos los sub-conjuntos evaluados. Beam search es similar pero trunca la lista y solament contiene un número fijo de candidatos.

FSelector: Selecting attributes

Caret: librerías diversas

4.5. Minería de textos

Una gran cantidad de la información disponible está almacenada en bases de datos de texto o de documentos. Este tipo de bases de datos están compuestas por documentos de diversas fuentes tales como artículos, libros, mensajes de correo y páginas Web entre otras.

Los datos almacenados en la mayoría de las bases de datos de texto se encuentran en forma semi-estructurada. Por ejemplo, un documento puede contener unos cuantos campos estructurados como por ejemplo, el título, autores,

fechas de publicación y categoría pero también contiene una gran cantidad de componentes no estructurados como por ejemplo, el resumen y los contenidos. Existen numerosos estudios en el modelado y la implementación de datos semi-estructurados, además, las técnicas de recuperación de información como los métodos de indexación se han desarrollado para manejar documentos no estructurados. Las técnicas tradicionales de recuperación de información se vuelven inadecuadas debido a la enorme cantidad de información que día a día se incrementa. Típicamente, solamente una peque na fracción de todos los documentos disponibles será relevante a un usuario. Sin conocer lo que pueden contener los documentos, es difícil formular queries efectivos para analizar y extraer información útil a partir de los datos. Los usuarios necesitan herramientas para comparar documentos, ordenarlos de acuerdo a su importancia o encontrar patrones y tendencias entre los múltiples documentos. Ante este panorama, la minería de texto se ha vuelto popular y un tópico fundamental en la minería de datos.

tm: librería de minería de texto

4.6. Sistemas de recomendación

Una de las aplicaciones que actualmente están en creciente uso son los sistemas de recomendación. Este tipo de sistemas en los que se generan recomendaciones personalizadas y se predicen los ratings tienen también mucho tiempo de existir. El primer sistema reportado es Information Lense, de 1987. Los sistemas de recomendación aplican técnicas estadísticas y de descubrimiento de conocimiento para recomendar productos o servicios con base en datos registrados. Estos sistemas pueden clasificarse en dos grupos:

- Basados en contenido. Toma en cuenta los atributos del usuario y los ítems. Si a un usuario le gusta la comida china, le recomendará el ítem que coincida con esa preferencia. Ventaja: no importa que existan pocos usuarios para generar recomendaciones. Desventaja: poca novedad, tal vez sea muy preciso pero no favorece conocer otros productos o servicios. These approaches recommend items that are similar in content to items the user has liked in the past, or matched to attributes of the user.

- Filtrado colaborativo. Recomienda los ítems con base en la similaridad entre usuarios y/o los ítems. Los ítems recomendados a un usuario son aquellos que prefieren usuarios similares.

Se mostrará un modelo basado en una matriz de preferencias. Se introduce el concepto de long-tail que explica la ventaja de los vendedores online sobre el resto.

En un sistema de recomendación hay dos clases de entidades: usuarios e ítems. Los usuarios tienen preferencias por ciertos ítems, y esas preferencias se representan en los datos. Los datos se representan como una matriz de utilidad para cada par usuario-ítem. El valor representa lo que se conoce sobre el grado de preferencia del usuario por ese ítem. Suponemos que la matriz es incompleta pues muchos ratings son desconocidos.

Ejemplo de matriz de utilidad

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

Cuadro 4.1: Matriz de utilidad

Una de las tareas de los sistemas de recomendación es predecir lo desconocido en la matriz. El top-n

No necesariamente se tiene que predecir cada valor faltante. Solamente se necesita descubrir altunas entradas que probablemente sean altas y a veces ni sino solamente encontrar un sub-conjunto grande de aquellos con los ratings mayores.

En este tema, un fenómeno importante es el long-tail. En el mundo real las tiendas solamente pueden mostrar al cliente una parte de la mercancía pues por limitaciones físicas no las puede tener todas. Por el contrario, las tiendas en línea pueden mostrar todo. Ejemplos: librerías, periódicos. La recomendación en el mundo físico es muy simple. simple. Generalmente las librerías desplegarán los libros más populares y el periódico publicará los artículos que cree que a sus lec-

tores les gustarán. En el primer caso, las ventas gobiernan las elecciones, en el segundo, el juicio editorial.

La diferencia, viendo la gráfica de long tail es que las tiendas físicas proporcionan solamente los ítems más populares, los que están a la izquierda. Las tiendas online proporcionan toda la cola además de los ítems populares.

El fenómeno de long-tail hace necesario a los negocios online, recomendar ítems a usuarios individuales. No es razonable esperar que los usuarios tengan que ver todos los ítems que pueden gustarles.

Ejemplo del efecto long-tail y un buen sistema de recomendación: Touching the Void 1998 no se había vendido, apareció Into thin air, un libro parecido y Amazon empezó a recomendarlo. Posteriormente Touching the void se hizo muy popular, por méritos propios.

Hay dos enfoques generales para obtener los valores sobre los ítems:

- Ratings explícitos. Se pide directamente.
- Se infiere por el comportamiento. Si el usuario compra el producto, mira el video, se supone que le gustó o al menos manifestó interés en el artículo. En este caso los valores serían 1 o 0.

4.6.1. Filtrado colaborativo

El filtrado colaborativo utiliza información de ratigs de ítems dados por usuarios como la base para predecir ratings y crear una lista de top-N recomendaciones para un usuario dado. Las dos tareas típicas son predecir ratinga para todos los ítems y crear una lista de las mejores recomendaciones. Formalmente, predecir los ratings faltantes es calcular una fila completa de la matriz donde los ratings faltantes se calculan estimándolos de otros datos.

Crear la lista top-N puede verse como un segundo paso, se toman los N ítems con los ratings más altos.

Generalmente tratamos con un gran número de items con ratings desconocidos lo que hace muy costoso el cálculo. Típicamente se dividen en dos grupos:

- Basados en memoria. Usan la base de datos completa o una gran muestra para crear recomendaciones. El algoritmo principal es el basado en el

usuario. La disventaja es la escalabilidad pues toda la base de datos tiene que ser procesada en línea.

- Basados en el modelo. Usan la base de datos para aprender un modelo más compacto que se usará posteriormente para crear recomendaciones. Ejemplo: clusters de usuarios con preferencias similares.

Los algoritmos de filtrado colaborativo miden la similaridad de renglones y/o columnas de la matriz de utilidad. La distancia Jaccard es apropiada cuando la matriz sea solamente de 1s y blancos. La distancia del coseno funciona mejor para valores más generales. A menudo es útil normalizar la matriz de utilidad restando el promedio, por renglón, por columna o ambos antes de medir la distancia coseno.

Los usuarios son parecidos si los vectores son cercanos de acuerdo a alguna distancia tal como Jaccard o el coseno. Recomendaciones para un usuario se hacen buscando a los usuarios más parecidos y recomendando ítems que les gustaríasn a ellos. Al proceso de identificar usuarios similares y recomendar lo que les gustaría se le llama filtrado colaborativo.

4.6.2. Basado en el usuario

Es un algoritmo basado en memoria que trata de imitar el comportamiento boca-a-boca. La suposición es que los usuarios con preferencias similares darán ratings similares. Así, los ratings faltantes para un usuario pueden ser calculados encontrando primero a sus vecinos (k vecinos cercanos) y luego agregar los ratings de esos usuarios a la predicción.

La vecindad se define en términos de similaridad entre usuarios, ya sea tomando un número de vecinos o a todos los usuarios dentro de un umbral. Las medidas de similaridad más populares son el coeficiente de correlación de Pearson y la similaridad del coseno. Recientemente se ha analizado que la correlación de Spearman es la más indicada. Estas medidas de similaridad se definen entre dos usuarios u_x y u_y como:

$Sim_{pearson}(x, y) = \frac{\sum_{i \in I}(x_i \overline{x})(y_i \overline{y})}{(|I|-1)sd(x)sd(y)}$

$Sim_{cosine}(x, y) = \frac{x \cdot y}{\|x\|\|y\|'}$

Se obtienen los vecinos para el usuario activo. Una vez que son encontrados, se agregan en forma de rating al usuario activo. La forma más simple es el promedio de los ratings de los vecinos.

El top-N se obtiene al ordenar los ítems de acuerdo al valor del rating para el vecino activo.

El algoritmo puede mejorarse eliminando el bias del rating. Esto se hace normalizando los datos del rating antes. La normalización se usa para eliminar bias de usuarios que consistentemente usan bajos o altos ratings que otros usuarios. Un método popular es restar la media de todos los ratings en el renglón.

Los principales problemas del CF basado en el usuario es que se tiene que tener la base de datos en memoria y que el cálculo de similaridad es costoso.

4.6.3. Basado en el ítem

Es un enfoque basado en el modelo que genera recomendaciones basadas en las relaciones entre ítems inferidas de la matriz de ratings. La suposición detrás de este enfoque es que los usuarios prefieren ítems que sean similares a otros ítems que les gusten.

El paso de la construcción del modelo consiste en calcular una matriz de similaridad conteniendo todas las similaridades ítem-ítem usando una medida. Las medidas populares son las mismas. Para reducir el modelo, para cada ítem solamente una lista de los k ítems más parecidos y valores de similaridad se guardan. Los k ítems más parecidos al ítem i son los vecinos del ítem. Esto mejora el rendimiento pero se sacrifica calidad en la recomendación.

Para hacer una recomendación basada en el modelo se usan las similaridades para calcular una suma pesada de ratings del usuario para ítems relacionados.

Es más eficiente que el basado en usuario ya que el modelo es reducido y puede ser pre-calculado. Se aplica con éxito en sistemas de recomendación grandes como Amazon.

Bibliografía

[Bartlein, 2009] Bartlein, P. (2009). Data analysis and visualization. http://geography.uoregon.edu/bartlein/old_courses/geog414f03/lectures/lec04.htm.

[Fayyad et al., 1996] Fayyad, U., Piatetsky-shapiro, G., and Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI Magazine*, 17:37–54.

[Guyon and Elisseeff, 2003] Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182.

[Hall and Holmes, 2003] Hall, M. A. and Holmes, G. (2003). Benchmarking attribute selection techniques for discrete class data mining. *IEEE Trans. on Knowl. and Data Eng.*, 15(6):1437–1447.

[Hofmann, 1994] Hofmann, H. (1994). UCI machine learning repository. http://bit.ly/laDAFC.

[Liu and Yu, 2005] Liu, H. and Yu, L. (2005). Toward integrating feature selection algorithms for classification and clustering. *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, 17:491–502.

[Quick, 2009] Quick, J. M. (2009). R tutorial series: Summary and descriptive statistics. http://www.r-bloggers.com/r-tutorial-series-summary-and-descriptive-statistics/.

[R Development Core Team, 2010] R Development Core Team (2010). R: A language and environment for statistical computing. http://www.R-project.org.

[Smith, 2010] Smith, D. (2010). R is hot. *Executive White Paper*. www.revolutionanalytics.com/R-is-Hot/.

[Trochim, 2006] Trochim, W. M. (2006). Research methods knowledge base. http://geography.uoregon.edu/bartlein/old_courses/geog414f03/lectures/lec04.htm.

[Wackerly et al., 2002] Wackerly, D. D., Scheaffer, R. L., and Mendenhall, W. (2002). *Estadística matemática con aplicaciones*. México, 6 edition.