COMP 6315 Mineria de Datos

CLASE 3: Preprocesamiento de Datos

Dr. Edgar Acuna
Departmento de Matematicas

Universidad de Puerto Rico-Mayaguez

website: academic.uprm.edu/eacuna

Por qué preprocesar los datos?

- Los datos en el mundo real son “sucios”:
  - incompletos: falta valores en los atributos, carecen de algunos atributos de interés, o contienen sólo totalizaciones.
  - ruidosos: contienen errores o valores anómalos.
  - inconsistentes: contienen discrepancias en códigos o nombres (Notas: A,AB,B,C,D,F,W).
  - datos duplicados.
- Si no hay calidad en los datos, no hay calidad en los resultados!
  - Las decisiones de calidad deben estar basadas en datos de calidad.
  - Para hacer Data Warehouse se necesita integrar consistentemente datos de calidad.

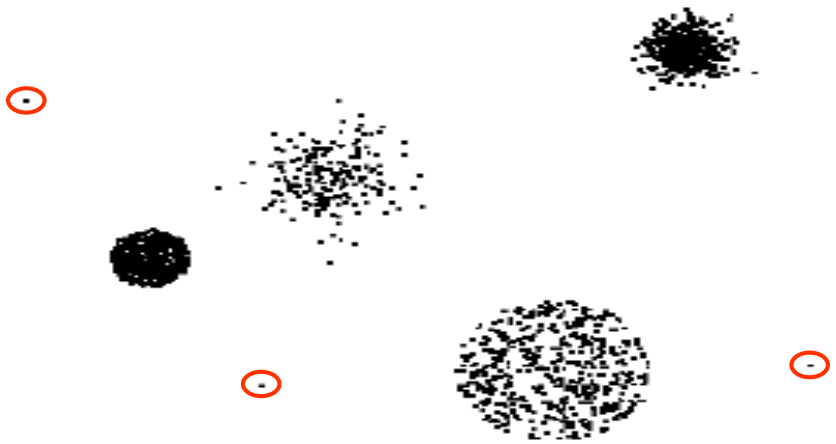
Ruido

- El ruido se refiere a la modificación de valores originales.
- Ejemplos: distorsión de la voz de una persona cuando habla por teléfono, “nieve” en la pantalla de televisión.

Two Sine Waves **Two Sine Waves + Noise**

“Outliers”

- Los “outliers” son casos con características que difieren considerablemente de la mayoría de los casos en el conjunto de datos.



Principales Tareas de Preprocesamiento de Datos

- Limpieza de datos
  - Completar valores faltantes, suavizar datos ruidosos, identificar o eliminar “outliers”, y resolver inconsistencias.
- Integración de datos
  - Integración de múltiples bases de datos.
- Transformación de datos
  - Normalización y totalización.
- Reducción de datos
  - Se obtiene una representación más reducida en volumen pero que produce los mismos o similares resultados analíticos.
- Discretización de datos
  - Parte de la reducción de datos pero que es particularmente importante para datos numéricos.

Limpieza de Datos

- Tareas de la limpieza de datos:
  - Completar valores faltantes.
  - Identificar “outliers” y suavizar datos ruidosos.
  - Corregir datos inconsistentes.

Preprocesamiento - Datos Faltantes

- Los datos no siempre están disponibles.
  - E.g., muchas filas no tienen registrados valores para muchos atributos, tales como los ingresos del cliente en datos de ventas.
- La falta de valores se puede deber a:
  - mal funcionamiento de equipos.
  - inconsistencia con otros datos registrados y por lo tanto han sido eliminados y considerados faltantes.
  - datos no ingresados debido a equivocaciones o malos entendidos.
  - algunos datos pudieron no considerarse importantes al momento de ingresar datos y fueron omitidos.
- Puede ser necesario estimar los valores faltantes.

Datos faltantes (cont)

- Los valores faltantes son un problema común en análisis estadístico.
- Se ha propuesto muchos métodos para el tratamiento de valores faltantes. Muchos de estos métodos fueron desarrollados para el tratamiento de valores faltantes en encuestas por muestreo.
- Bello (1995), tratamiento de valores faltantes in regression
- Troyanskaya et al (2001), tratamiento de datos faltantes en clasificacion no supervisada.

Datos faltantes (cont)

- Estudios relacionados con clasificación supervisada:
  - Chan and Dunn (1972) – Imputation en LDA para problemas con dos clases.
  - Dixon (1975) – Imputacion k-nn para lidiar con valores faltantes en clasificacion supervisada.
  - Tresp (1995)- el problema de valores faltantes en aprendizaje supervisado usando redes neurales.
- Impacto de valores faltantes:
  - 1% datos faltantes – trivial
  - 1-5% - manejable
  - 5-20% - requiere métodos sofisticados
  - 20% o mas- efecto perjudica las interpretaciónes

Mecanismos de valores faltantes

- Hay tres mecanismos de valores que tratan de explicara las razones que cusan que hayan datos faltantes:
- i) Valores faltantes completamente al azar *(MCAR):* La probabilidad que una instancia tenga un valor faltante para un atributo es la misma para todas las instancias. Es decir, esta probabilidad no depende ni de los valores observados ni de los valores faltantes. La mayoria de los valores faltantes no son MCAR.

  Este mecanismo es mas adecuado para datos a ser usados en clasificacion no supervisada.

  Asimismo es adecuado para conjuntos de datos con un bajo porcentaje de valores faltantes.

Mecanismos de valores faltantes

- ii) *Valores faltantes al azar (MAR)*: La probabilidad que una instancia tenga un valor faltante en un atributo depende de los valores observados, como por ejemplo la clase a la cual pertenece la instancia, pero no depende de los valores faltantes. Este mecanismo es mas adecuado para datos usados en clasificacion supervisada. Asimismo para simular datos faltantes.
- *iii) Valores faltantes no al azar o no ignorables(NMAR):* La probabilidad de que una instancia tenga un valor faltante en un atributo depende de los valores faltantes en el conjunto de datos. Ocurre cuando las personas entrevistadas no quieren revelar algo muy personal acerca de ellas. Este tipo de valores faltantes es el mas dificil de tratar y es el que ocurre mas frecuentemente. Se considera cuando hay una gran cantidad de datos faltantes

Conjunto de datos Census

Tambien conocido como **Adult.**

48842 instancias, contiene variables continuas , ordinales y nominales (entrenamiento=32561, prueba=16281).

Cuando se eliminan las instancias con valores faltantes quedan 45222 (entrenamiento=30162, prueba=15060).

Tamano=3.8MB (entrenamiento), 1.9MB(prueba)

Disponible en: http://archive.ics.uci.edu/ml/

Donantes: Ronny Kohavi y Barry Becker (1996).

Variables en Census

- 1- age: continua.
- 2- workclass: Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked. Nominal
- 3- fnlwgt (final weight) : Continua.
- 4- education: Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool. Ordinal.
- 5- education-num: continua.
- 6- marital-status: Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse. Nominal
- 7- occupation: Nominal

Variables en Census

- 8-relationship: Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried. Nominal
- 9-race: White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black. Nominal
- 10-sex: Female[0], Male[1]. Nominal-Binaria
- 11-capital-gain: continua.
- 12-capital-loss: continua.
- 13-hours-per-week: continua.
- 14-native-country: nominal
- 15 Salary: >50K [2], <=50K [1].

Ejemplo: Conjunto de datos census

age	employme	education	edun	marital	…	job	relation	race	gender	hour	country	wealth
					…							
39	State_gov	Bachelors	13	Never_mar	…	Adm_cleric	Not_in_fam	White	Male	40	United_Sta	poor
51	Self_emp_	Bachelors	13	Married	…	Exec_man	Husband	White	Male	13	United_Sta	poor
39	Private	HS_grad	9	Divorced	…	Handlers_c	Not_in_fam	White	Male	40	United_Sta	poor
54	Private	11th	7	Married	…	Handlers_c	Husband	Black	Male	40	United_Sta	poor
28	Private	Bachelors	13	Married	…	Prof_speci	Wife	Black	Female	40	Cuba	poor
38	Private	Masters	14	Married	…	Exec_man	Wife	White	Female	40	United_Sta	poor
50	Private	9th	5	Married_sp	…	Other_serv	Not_in_fam	Black	Female	16	Jamaica	poor
52	Self_emp_	HS_grad	9	Married	…	Exec_man	Husband	White	Male	45	United_Sta	rich
31	Private	Masters	14	Never_mar	…	Prof_speci	Not_in_fam	White	Female	50	United_Sta	rich
42	Private	Bachelors	13	Married	…	Exec_man	Husband	White	Male	40	United_Sta	rich
37	Private	Some_coll	10	Married	…	Exec_man	Husband	Black	Male	80	United_Sta	rich
30	State_gov	Bachelors	13	Married	…	Prof_speci	Husband	Asian	Male	40	India	rich
24	Private	Bachelors	13	Never_mar	…	Adm_cleric	Own_child	White	Female	30	United_Sta	poor
33	Private	Assoc_acd	12	Never_mar	…	Sales	Not_in_fam	Black	Male	50	United_Sta	poor
41	Private	Assoc_voc	11	Married	…	Craft_repai	Husband	Asian	Male	40	*MissingVa	rich
34	Private	7th_8th	4	Married	…	Transport_	Husband	Amer_India	Male	45	Mexico	poor
26	Self_emp_	HS_grad	9	Never_mar	…	Farming_fi	Own_child	White	Male	35	United_Sta	poor
33	Private	HS_grad	9	Never_mar	…	Machine_c	Unmarried	White	Male	40	United_Sta	poor
38	Private	11th	7	Married	…	Sales	Husband	White	Male	50	United_Sta	poor
44	Self_emp_	Masters	14	Divorced	…	Exec_man	Unmarried	White	Female	45	United_Sta	rich
41	Private	Doctorate	16	Married	…	Prof_speci	Husband	White	Male	60	United_Sta	rich
:	:	:	:	:	:	:	:	:	:	:	:	:

Leyendo files de datos en R

El comando basico es read.table(“filename”)

Si los datos estan guardados localmente y en el formato csv se usa
> a=read.csv("c://datos1.csv")
>a
Si los datos estan en Excel se usa la librerias **xlsx**, que tiene el comando read.xlsx(“filename”)
La libreria **gdata** que tiene el comando read.xlx o
La libreria **XLConnect**

Tambien se puede leer datos en otros formatos usando la libreria **foreign.**

Datos de hasta un millon de registros pueden ser leidos usando la funcion **fread** de la libreria **data.table** de R

Leyendo files de datos en R (cont)

Datos de hasta 2 GB pueden ser leidos pero no necesariamente analyzados directamente en R. El analisis requiere usar otras herramientas computacionales como Hadoop, y Spark.

Para leer datos de la internet se puede usar read.table("url"), donde url es la localizacion en la internet de los datos que se quieren leer.
Por ejemplo, para leer los datos de "adult" que estan en la UCI se usa
read.table("https://archive.ics.uci.edu/ml/machine-learning-databases/adult/adult.data",header=F,sep=",",na.strings="? ")

Donde header=F significa que ls columas de datos no tienen nombres
Sep="," significa que los elementos de cada registro estan separados por ","
Y na.strings="?", significa que "?" representa un "missing values".
Tambien descargando previamente el paquete curl se puede usar el commando
fread("https://archive.ics.uci.edu/ml/machine-learning-databases/adult/adult.data",header=F,sep=",",na.strings="?")

Datos Census en la libreria dprep

Asumiendo que se ha instalado la libreria dprep en el espacio de trabajo (Aun no corre en la ultima version de R, junio 2018)

```
>library(dprep)
>data(census)
>#Viendo la primera fila de los datos
>head(census)
```

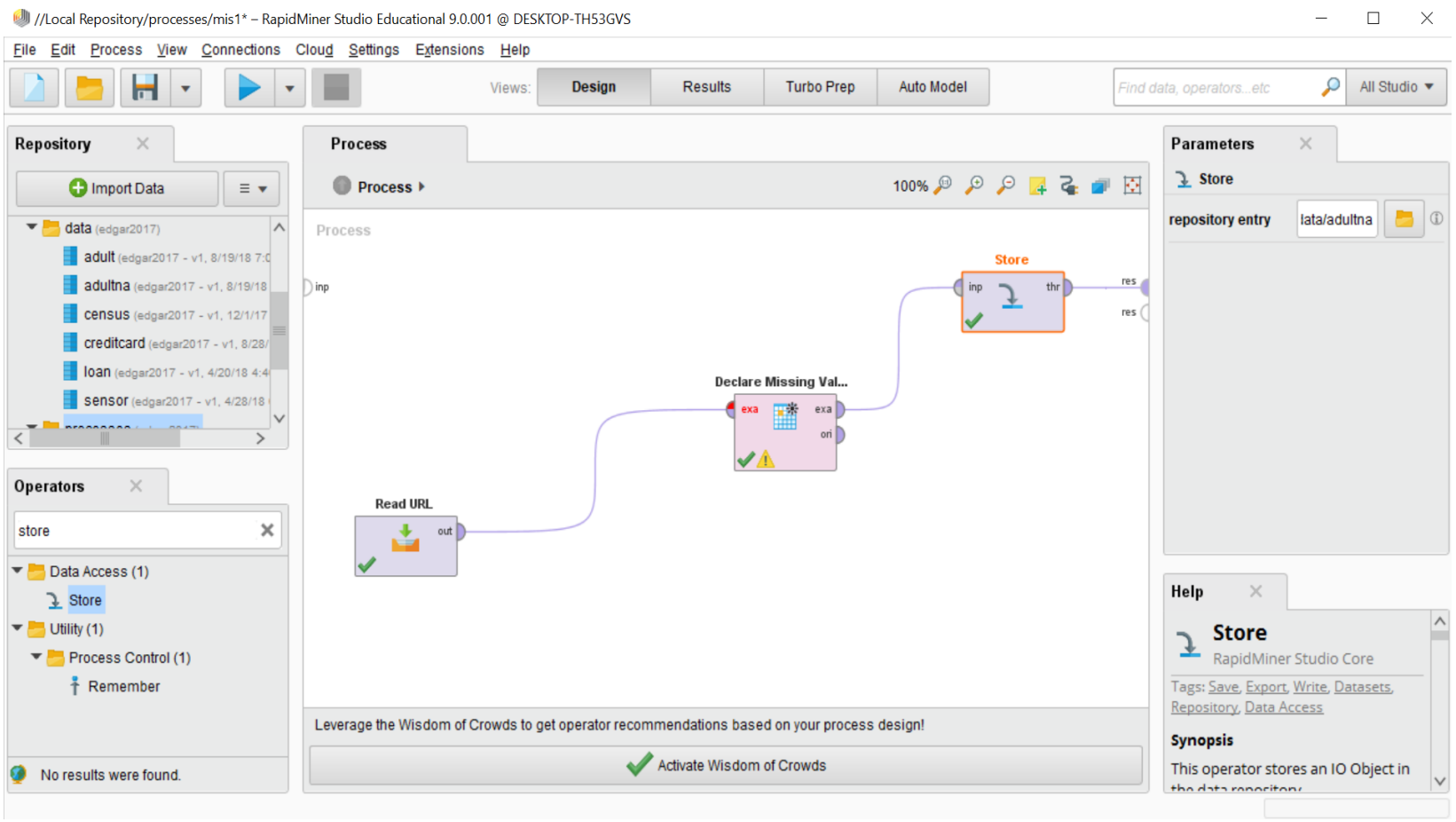
Leyendo los datos en Python

```
import pandas as pd
df = pd.read_csv('https://archive.ics.uci.edu/ml/machine-learning-
databases/adult/adult.data', header=None, sep=',‘,na_values=“ ?”)
df.columns=['v1', 'v2', 'v3', 'v4',v5','v6','v7','v8','v9','v10','v11','v12','v13','v14','class']
```

Otra forma:

```
 import pandas as pd
names=['v1','v2','v3','v4','v5','v6','v7','v8','v9','v10','v11','v12','v13','v14','clase']
data=pd.read_csv('https://archive.ics.uci.edu/ml/machine-learning-
databases/adult/adult.data’, names=names, na_values=“ ?”)
print(data.shape)
(32562, 15)
data.describe()
```

Leyendo los datos en Rapidminer



Funciones en R para Valores Faltantes

- Para detectar las columnas con missing values

  which(colSums(is.na(adult))!=0)
- Para detectar las filas con missing values

  rmiss=which(rowSums(is.na(adult))!=0,arr.ind=T)
- Para hallar el porcentaje de filas con missing values

  length(rmiss)*100/dim(adult)[1]
- Para hallar el porcentaje de missing values por columna

  colmiss=c(2,7,14)

  per.miss.col=100*colSums(is.na(adult[,colmiss]))/dim(adult)[1]
- Para eliminar los missing values

  adult.omit=na.omit(adult)

  dim(adult.omit)

  [1] 30162 15

Explorando el conjunto de datos usando

imagmiss() de la libreria dprep

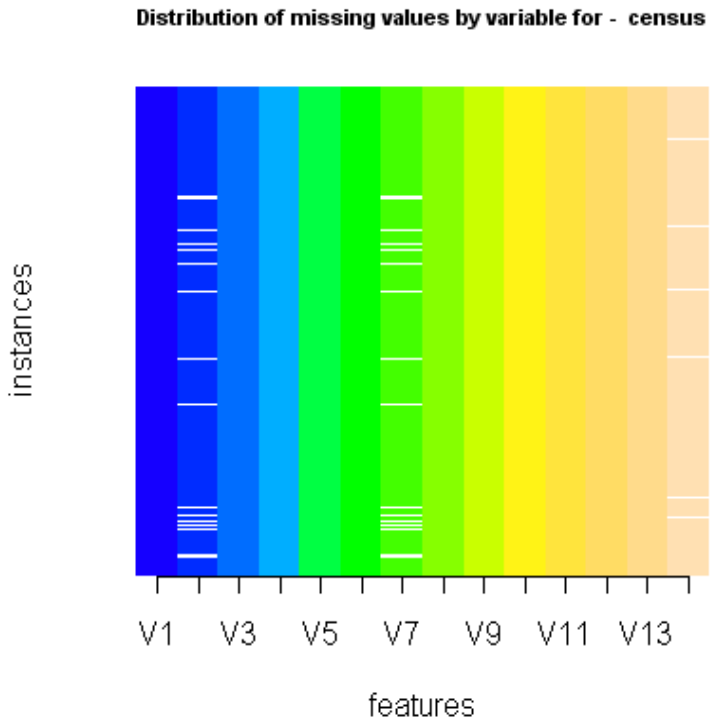
Instalar primero la libreria dprep de R

> imagmiss(data, name="dataname")

Report on missing values for Census :

Number of missing values overall: 4262
Percent of missing values overall: 0.9349485
Features with missing values (percent):
V2 V6 V13
5.638647 5.660146 1.790486

Percent of features with missing values:
21.42857
Number of instances with missing values: 2399
Percent of instances with missing values:
7.36771



La función clean

- Esta función elimina columnas y filas que tienen un gran número de valores faltantes.

```
census.cl=clean(adult,tol.col=.5,tol.row=.3,name="cl.adult")
```

	Variables	Percent.of.missing
1	V2	5.6386474616873
2	V6	5.66014557292466
3	V13	1.79048555019809

Maximum number of values to be imputed: 4262

Tratamiento de valores faltantes

- **Eliminación de casos**. Ignorar la fila que contiene datos faltantes. Usualmente es aplicado cuando el valor que falta es el de la clase (asumiendo que se esta haciendo clasificación). No es efectiva cuando el porcentaje de valores faltantes por atributo varía considerablemente.
- **Técnicas de Imputación**, donde los valores faltantes son reemplazados con valores estimados basados en la información disponible en el conjunto de datos.
- **Estimación de parámetros**, donde los procedimientos de Máxima Verosimilitud que usan variantes del algoritmo EM (Expectation-Maximization) pueden manejar la estimación de parámetros simultaneamente a lcon la sustitucion de valores faltantes.

Tratamiento de valores faltantes en clasificacion

Eliminacion de casos (CD) – Este método consiste en descartar todas las instancias (casos) con valores perdidos en por lo menos un atributo. Una variante de este método consiste en determinar el grado de valores faltantes en cada instancia y atributo, y eliminar las instancias y/o atributos con altos niveles de valores faltantes. Antes de eliminar cualquier atributo es necesario evaluar su relevancia en el análisis.

Tratamiento de valores faltantes

- Imputacion usando la media (MI) – Reemplazar los valores faltantes de un atributo dado por la media de todos los valores conocidos de ese atributo en la clase a la que la instancia con el valor faltante pertenece. Si la variable es nominal se usa la moda en lugar de la mediana
- Imputacion usando la mediana (MDI). Como la media se ve afectada por la presencia de outliers, parece natural usar la mediana en su lugar para asegurar robustez. En este caso los valores faltantes para un atributo dado es reemplazado por la mediana de todos los valores conocidos de ese atributo en la clase a la que la instancia con el valor faltante pertenece.

  adult.mimp=ce.mimp(adult ,"mean",1:14, c(2,7,14))

  adult.mdimp=ce.mimp(adult,"median",atr=1:14,nomatr=c(2,7,14))

Imputación con los k-vecinos mas cercanos (KNNI)

- Dividir el conjunto de datos D en dos partes. Sea *D*m el conjunto que contiene las instancias en las cuales falta por lo menos uno de los valores. Las demás instancias con información completa forman un conjunto llamado *D*c.

  - Para cada vector x en *D*m:

    A) Dividir el vector en dos partes: una la de información observada y otra la de información faltante, *x* = [*x*o; *x*m].

    B) Calcular la distancia entre *x*o y todos los vectores del conjunto *D*c. Usar solo aquellos atributos en los vectores de Dc que están observados en el vector x.

    C) Usar los K vectores más cercanos (K-nearest neighbors) y considerar la moda como un estimado de los valores faltantes para los atributos nominales. Para atributos continuos, reemplazar el valor faltante por la media del atributo en la vecindad de los k vecinos mas cercanos (k-nearest neighborhood).

Imputación con los k-vecinos mas cercanos (KNNI)[2]

A1	A2	A3	A4	Clase
4	5	NA	6	1
5	1	4	1	1
7	9	5	2	1
8	2	5	8	1
6	4	6	2	1

D(r1,r2)= 6.48 D(r1,r3)=6.40, D(r1,r4)=5.38, D(r1,r5)= 4.58
Luego, si se usa k=1 vecinos cercanos (r5), NA deberia ser reemplazado por 6,si se usa k=3 vecinos cercanos (r3,r4 y r5), NA deberia ser reemplazado por el promedio de 5,5 y 6 que es 5.33

Imputación con los k vecinos mas cercanos KNNI[3]

- El metodo no se podria aplicar si todas las filas de la matriz contienen al menos un valor faltante.
- Usualmente, se toma k igual a 10. Pero el numero de vecinos a usar es a lo sumo igual al numero de filas completas de la matriz.
- Cuando hay variables cuantitativas y categóricas se usa la distancia Gower en lugar de la distancia euclideana

.

Imputación con los k vecinos mas cercanos (KNNI)[4]

- Cuando la base de datos tiene atributos de distintos tipos, ya no se puede usar las distancias usuales como Euclideana y Manhattan que solo sirven para atributos numericos.
- Hay muchas alternativas para medir la distancia entre los registros (ver Wilson y Martinez).
- Sin embargo la distancia mas usada es la distancia Gower.
- data(crx)
- crx.knn=ec.knnimp(crx,nomatr=c(1,4:7,9:10,12:13),k=5)

  imputacion por KNN también esta implementada en RapidMiner.

Distancia Gower(disponible en la librería StatMatch)

- Supongamos que tenemos p atributos algunos de los cuales son cuantitativos y otros nominales no ordinales.

La distancia Gower entre los registros Xi y Xj

Xi=(xi1,xi2,……,xip) y Xj=(xj1,xj2,……..,xjp)

Estaa data por $D_G(Xi,XJ)= \sum_{k=1}^{p} d(xik,xjk)/p$

Donde d(xik,xjk)=|xik-xjk|/rango(Xk) si la variable es continua y d(xik,xjk)=1 si la variable Xk es nominal y sus valores son distintos y d(xik,xjk)=0 si sus valores son iguales.

El valor de la distancia Gower esta estandarizado entre 0 y 1.

Insertando aleatoriamente valores faltantes[1]

```
> mat1=cbind(c("a","b","ab",“b",“b","ab"),c("ac","ad",“ac","ab","ac","ab"))
> mat1
     [,1] [,2]
[1,] "a"  "ac"
[2,] "b"  "ad"
[3,] "ab" “ac"
[4,] “b"  "ab"
[5,] “b"  "ac"
[6,] "ab" "ab"
> dim(mat1)
[1] 6 2
```

Insertando aleatoriamente datos faltantes[2]

```
>#convirtiendo la matriz en un arreglo vectorial
>mat2=as.vector(mat1)
> mat2
 [1] "a"  "b"  “ab" “b"  “b"  "ab" "ac" "ab" “ac" "ab" "ac" “ab"
>#insertando al azar 6 valores faltantes
>mat2[sample(1:12,6)]=NA
> mat2
 [1] NA   "b"  "ab" NA   NA   "ab" "ac" NA   “ac" NA   NA   "ab“
>#reconstruyendo la matriz
>mat2=matrix(mat2,ncol=2)
```

Insertando aleatoriamente datos faltantes[3]

```
>#anadiendole una columna ficticia de clases, porque la imputación
>en dprep solo es para datos supervisados
>mat3=cbind(mat3,rep(1,6))
>#convirtiendo mat3 en dataframe
>mat3=data.frame(mat3)
> mat3
   X1     X2 X3
1 <NA>   ac  1
2    b <NA>  1
3   ab      ac       1
4 <NA> <NA>  1
5 <NA> <NA>  1
6   ab   ab  1
```

Imputando los datos faltantes

```
>ce.mimp(mat3,"mean",1:2,nomatr=c(1,2))
Summary of imputations using substitution of  mean (mode for nominal features):
    Row Column Class Imput.value
[1,] "1" "1"    "1"   "ab"
[2,] "2" "2"    "1"   "ac"
[3,] "4" "1"    "1"   "ab"
[4,] "4" "2"    "1"   "ac"
[5,] "5" "1"    "1"   "ab"
[6,] "5" "2"    "1"   "ac"

Total number of imputations per class:

1
6
```

Otros métodos de imputación.

- **Hot deck and Cold deck. [nces.ed.gov/statprog].** En Cold deck se usan valores de estudios similares para reemplazar valores perdidos en el estudio actual. En Hot deck se usa valores de atributos correlacionados con el atributo que contiene el valor faltante para sustituirlos.
- **Modelo predictivo:** Regresión Lineal (atributos continuos), Regresión Logística (atributos binarios), logística Polychotomous (atributos nominales). El atributo con valor faltante es usado como la variable de respuesta y los demás atributos son considerados predictoras. En general, se puede usar modelos de Classification and Regression Trees.
- Desventajas: Puede crear sesgo, requiere correlación alta entre predictoras. Cómputo lento.

Otros métodos de imputación.

- Imputación Múltiple. Se imputan varias veces los valores faltantes con valores simulados de una distribucion que se asume para cada variable.
- Metodo de la SVD ($X=U\Sigma V'$). Iterativamente, considerando inicialmente a los missings como las medias de cada columna. Y sustituyendo los missings en cada iteración, hasta que la norma de la matriz converja.
- Algoritmo EM. Asumir un disribucion para las predictoras y estimando los parametros iterativamente hasta alcanzar convergencia.
- Los árboles de decisión tienen su propio enfoque para tratar valores faltantes.

Imputacion en Python

La libreria pandas tiene funciones que permiten hacer imputacion por la media y/o mediana en caso de que los atributos sean cuantitativos y por la moda en el caso de atributos categoricos . La function imputer de la libreria sklearn solo hace imputacion por vecinos mas cercanos con datos cuantitativos. Si hubieran variables categoricas estan deberian convertirse en numericas antes de imputar por knn

Imputacion en Rapidminer

Rapidminer hace imputacion por la media o mediana en caso de de que el atributo sea continuo o por la moda en caso de que el atributo sea nominal.
Primero se llama al conjunto de datos que contiene los valores perdidos y donde se va a aplicar la imputacion. Elegir de la lista de operadores, Data cleansing y de alli Replace Missing values. Unir el port out del operador de datos con el port exa del operador Replace missing values. Finalmente, unir el port exa de este operador con el port res y ejecutar.

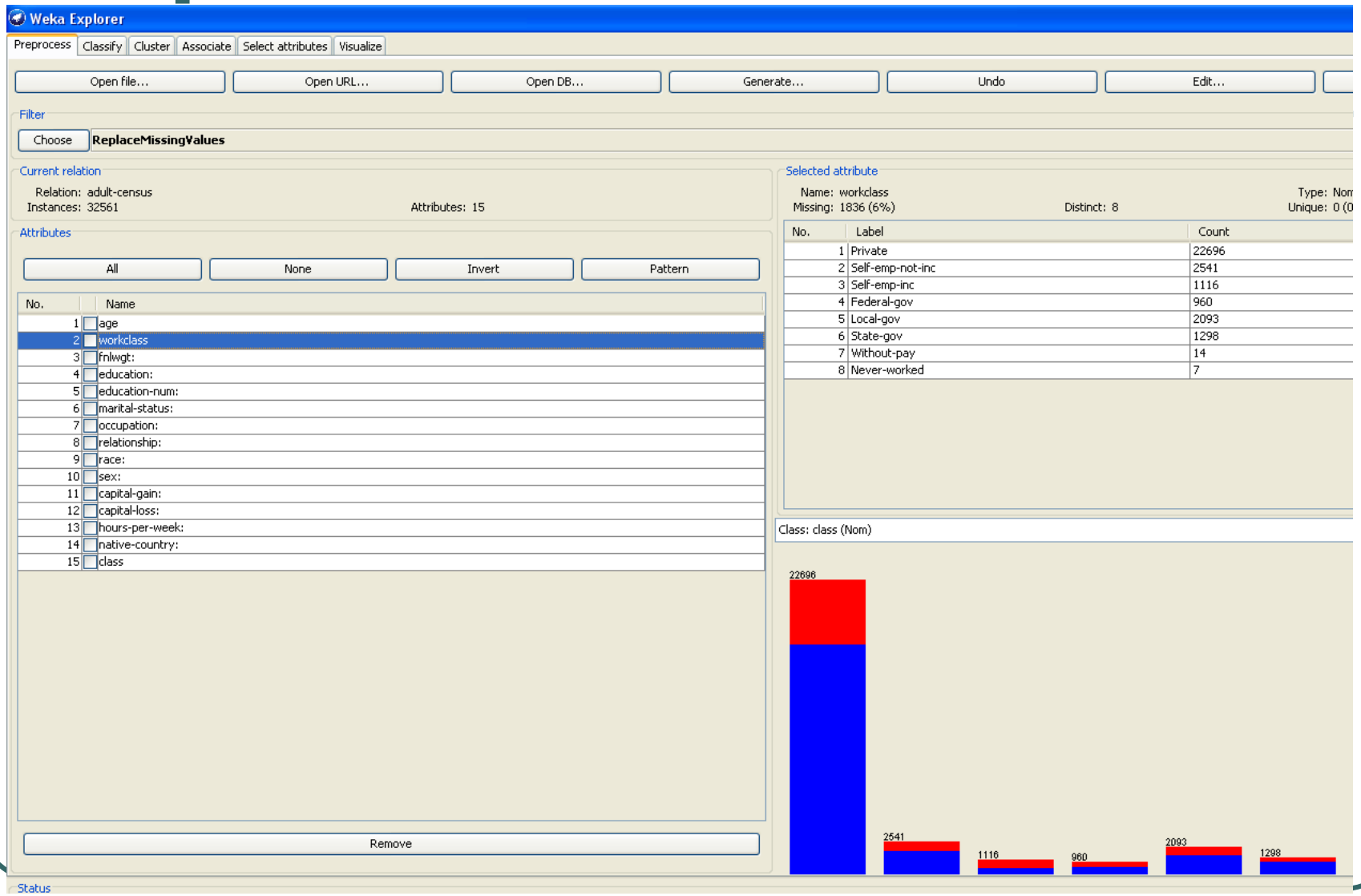
El proceso es rapido al igual que en Weka.

Rapidminer Tambien hace imputacion por k-nn

Imputacion en WEKA

Weka hace solo imputacion por la media en caso de de que el atributo sea continuo o por la moda en caso de que el atributo sea nominal.
Para esto despues de abrir un archivo arff se elige el boton Choose de filter y se usa la siguiente secuencia filters> unsupervised >attribute>ReplaceMissingValues y luego se da Apply.
El conjunto de datos con valores faltantes es completado y luego se continua el analisis.
Para reemplazar los datos de Census se tarda un minuto.

Imputacion en WEKA



Efecto de valores faltantes en clasificacion supervisada

- Para conjuntos de datos con una pequena cantidad de valores faltantes se observa poca diferencia entre la eliminacion de casos y otros metodos de imputacion.
- Cuando se usa eliminacion de casos la variabilidad del estimado del error de clasificacion aumenta.
- Casi no hay diferencia entre usar imputacion por la media e imputacion por la mediana.
- El efecto de los valores faltantes depende de la forma que se distribuyen en la matriz de datos y en su localizacion con respecto a las variables mas importantes.
- El porcentaje de instancias con valores faltantes tiene mayor efecto en el proceso de clasificacion que el porcentaje total de valores faltantes en la matriz de datos
- El tratamiento de los valores faltantes en el procesos de clasificacion depende del clasificador que esta siendo usado.

Preprocesamiento-Normalización

- La normalización de datos consiste en re-escalar los valores de los datos dentro de un rango especificado, tal como -1 a 1 o 0 a 1.
- También es conocido como “normalizacion del rango”.

Razones para normalizar

- Normalizar los datos de entrada ayudará a acelerar la fase de aprendizaje.
- Los atributos con rangos grandes de valores tendran mas peso que los atributos con rangos de valores mas pequenos, y entonces dominarán la medida de distancia. Por ejemplo, el clasificador K-nn sando la medida de distancia euclideana depende de que todas las dimensiones de los valores de entrada estén en la misma escala.
- También puede ser necesario aplicar algún tipo de normalización de datos para evitar problemas numéricos tales como pérdida de precisión y desbordamientos aritméticos (overflows).

Conjunto de datos Bupa

Número de instancias: 345

Número de atributos: 7

Descripción de los atributos:

1. Mcv volumen corpuscular
2. alkphos fosfotasa alcalina
3. sgpt alamine aminotransferase
4. sgot aspartate aminotransferase
5. gammagt gamma-glutamyl transpeptidase
6. drinks numero de bebidas alcohoilicas
7. Class: 1 (higado enfermo) y 2 (higado sano)

> bupa[1:20,]

	V1	V2	V3	V4	V5	V6	V7
1	85	92	45	27	31	0	1
2	85	64	59	32	23	0	2
3	86	54	33	16	54	0	2
4	91	78	34	24	36	0	2
5	87	70	12	28	10	0	2
6	98	55	13	17	17	0	2
7	88	62	20	17	9	0.5	1
8	88	67	21	11	11	0.5	1
9	92	54	22	20	7	0.5	1
10	90	60	25	19	5	0.5	1
11	89	52	13	24	15	0.5	1
12	82	62	17	17	15	0.5	1
13	90	64	61	32	13	0.5	1
14	86	77	25	19	18	0.5	1
15	96	67	29	20	11	0.5	1
16	91	78	20	31	18	0.5	1
17	89	67	23	16	10	0.5	1
18	89	79	17	17	16	0.5	1
19	91	107	20	20	56	0.5	1
20	94	116	11	33	11	0.5	1

Normalización Z-score

Los valores V son normalizados en base a la media y desviación estándar.

V' = (V - mean) / std

Este método trabaja bien en los casos en que no se conoce el máximo y mínimo de los datos de entrada pero no cuando existen outliers que tienen un gran efecto en el rango de los datos.

> zbupa=rangenorm(bupa,'znorm',superv=T)

Normalización Min-Max

Este método realiza una transformación lineal de los datos originales V en el intervalo especificado [newmin,newmax]

V' = (V - min) * (newmax - newmin) / (max - min) + newmin

La ventaja de este método es que preserva exactamente todas las relaciones entre los datos. No introduce ningún potencial sesgo en los datos. La desventaja es que se encontrará un error “fuera del límite” ("out of bounds“) si un futuro ingreso de datos cae fuera del rango original.

- > mmbupa=rangenorm(bupa,mmnorm,superv=TRUE)

Normalización por escalamiento decimal

Este método realiza la normalización moviendo el punto decimal de los valores. El número de puntos decimales movidos depende del máximo valor absoluto.

$V' = V / 10^j$

donde j es el entero mas pequeño tal que Max(|V'|) < 1. Sólo es útil cuando los valores de los atributos son mayores que 1 en valor absoluto.

- > dsbupa=rangenorm(bupa,'dscale',superv=T)

Normalización Sigmoidal

Este método realiza una transformación no linear de los datos de entrada en el rango -1 a 1, usando una función sigmoidal.

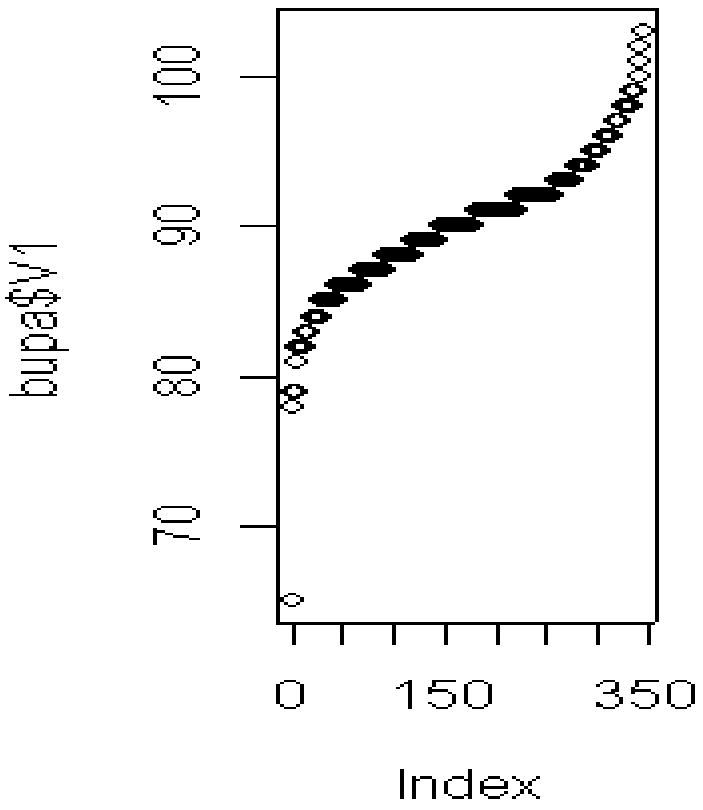
V' = (1 - e^(-a)) / (1 + e^(-a)) donde a = (V - mean) / std

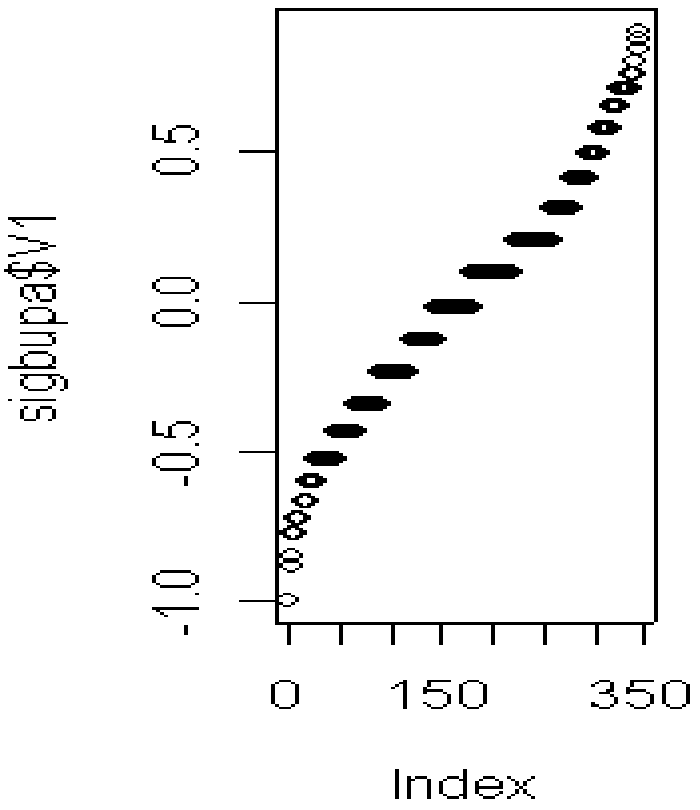
Los datos dentro de una desviación estándar de la media son mapeados a la región casi linear del sigmoide. Los puntos anómalos son comprimidos a lo largo de las colas de la función sigmoidal.

La normalización sigmoidal es especialmente apropiada cuando se tienen datos anómalos que se desean incluir en el conjunto de datos. Este previene que los valores que ocurren más comúnmente sean comprimidos en los mismos valores, sin perder la habilidad de representar grandes valores anómalos.

```
> sigbupa=rangenorm(bupa, 'signorm',superv=T)
> plot(sort(bupa$V1))
> plot(sort(sigbupa$V1))
```

Visualización del efecto de la transformación sigmoidal





Normalización Softmax

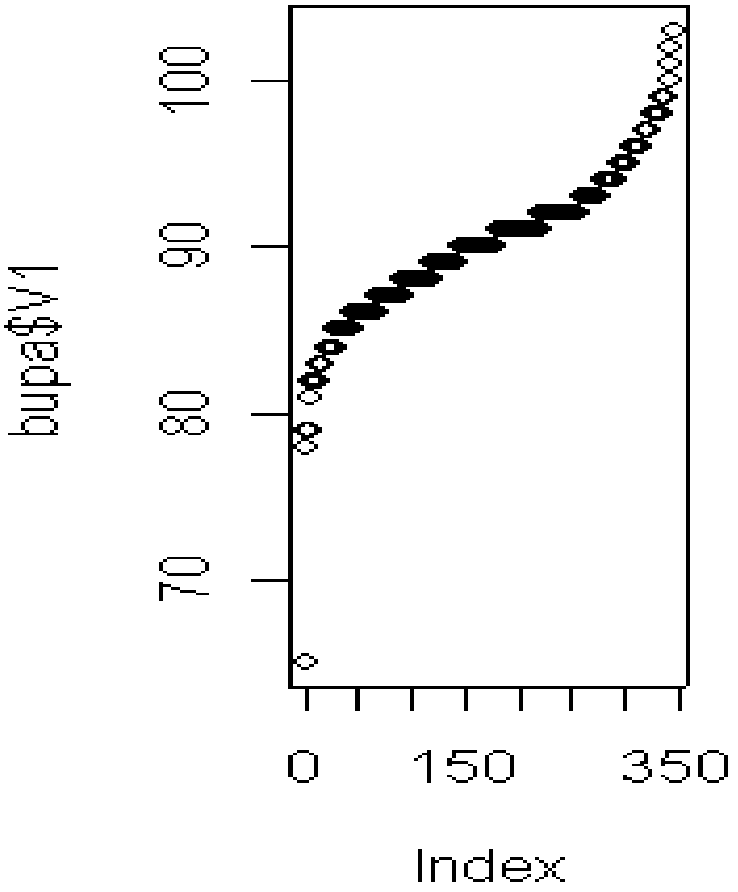
Se llama así porque llega suavemente hacia su valor máximo y mínimo. La transformación es mas o menos lineal en el rango medio, y tiene una ligera no linealidad a ambos extremos. El rango total cubierto es 0 a 1 y la transformación asegura que no ocurran valores futuros que caigan fuera del rango.

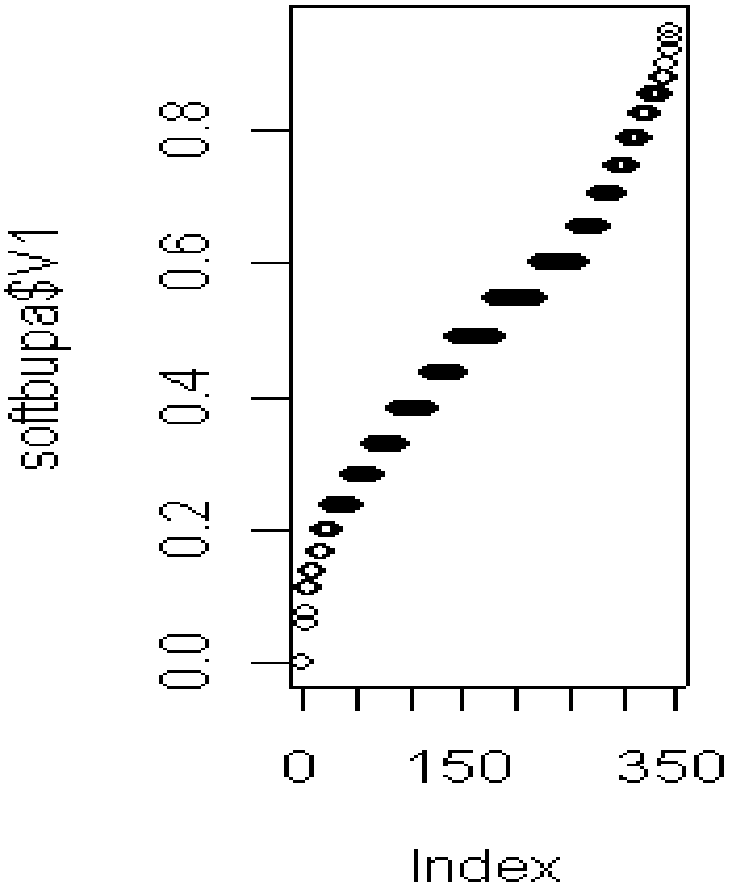
V' = 1 / (1 + e^(-a))

donde a= (V - mean) / std

- softbupa=softmaxnorm(bupa)
- zbupa=rangenorm(bupa,method="znorm")

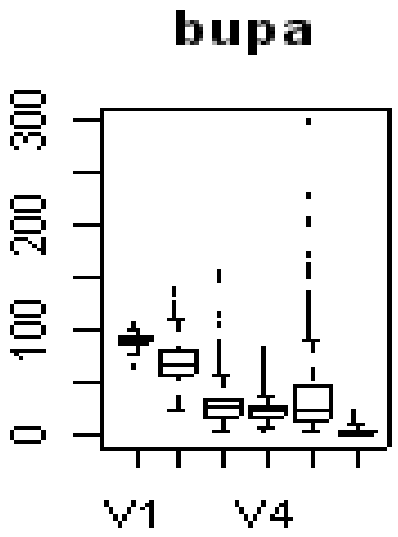
Visualizacion del efecto de la transformacion softmax

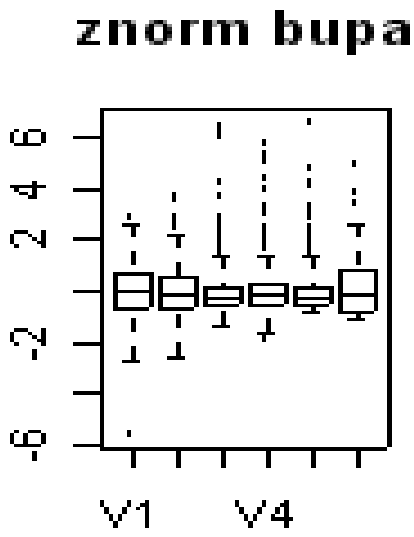


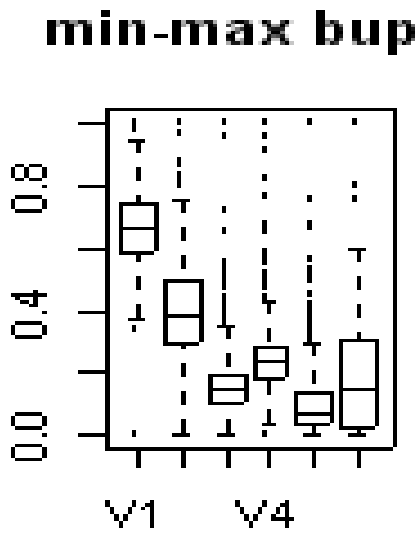


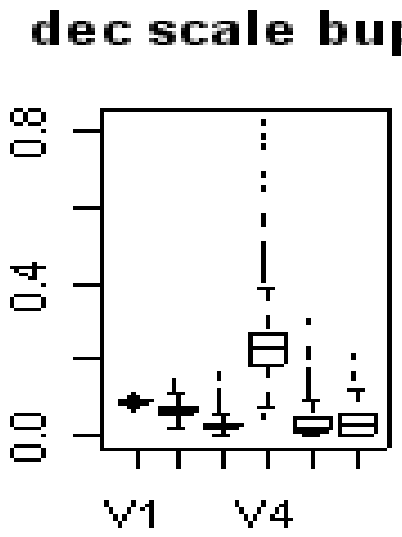
Boxplots que muestran el efecto de la normalización

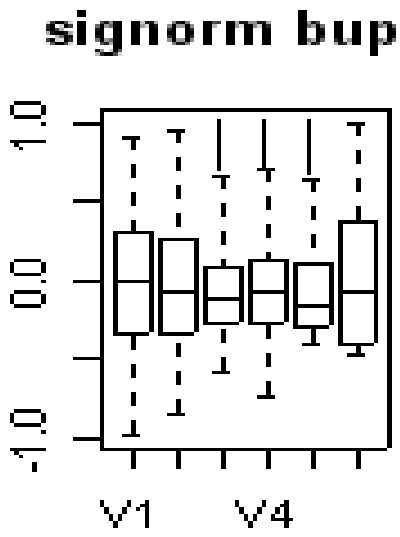
```
> par(mfrow=c(2,3))
> boxplot(bupa[,1:6],main="bupa")
> boxplot(zbupa[,1:6],main="znorm bupa")
> boxplot(mmbupa[,1:6],main="min-max bupa")
> boxplot(dsbupa[,1:6],main="dec scale bupa")
> boxplot(sigbupa[,1:6],main="signorm bupa")
> boxplot(softbupa[,1:6],main="softmax bupa")
```

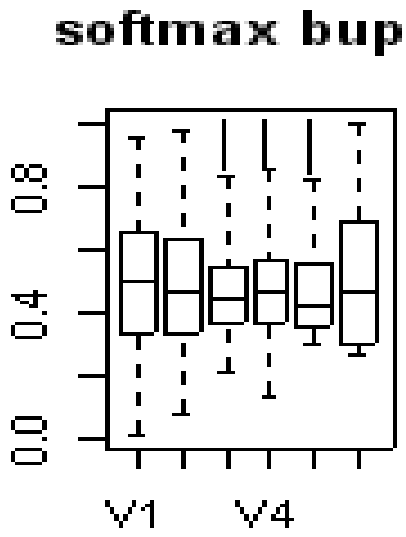

bupa
0
100
200
300
V1
V4


znorm bupa
-6
-2
2
4
6
V1
V4


min-max bupa
0.0
0.4
0.8
V1
V4


dec scale bupa
0.0
0.4
0.8
V1
V4


signorm bupa
-1.0
0.0
1.0
V1
V4


softmax bupa
0.0
0.4
0.8
V1
V4

Normalización en Rapidminer

Se llama al operador Data transformation and luego Value Modification, despues Numerical Value Modification y luego Normalize. Solo hace Normalizacion z-score y Min-Max

Normalizacion en Weka

Weka realiza normalizacion de los atributos numericos transformando los valores originales a valores en el intervalo [0,1].

Despues de abrir un archivo arff siga la siguiente secuencia filters>unsupervised>attributes>Normalize.

Z-normalization es llevada a cabo por el filter Standardize