Programación para Data Science

Unidad 6: Preprocesamiento de datos en Python - Ejercicios para practicar

Cargad los datos del fichero `bank_edited.csv` en un dataframe. Este conjunto de datos recoge datos sobre una campaña de márketing de un banco portugués. El conjunto original puede encontrarse en el repositorio de datos de Machine Learning de la UC Irvine (http://archive.ics.uci.edu/ml/datasets/Bank+Marketing), pero el conjunto que usaremos tiene algunas modificaciones.

Nota: revisad la documentación de la función `read_csv` (https://pandas.pydata.org/pandas-docs/stable/generated/pandas.read_csv.html) para ver de qué parámetros disponemos para ajustar el proceso de carga de datos

Ejercicio 1

Los valores del estado civil (atributo `marital`) contienen errores tipográficos e incluyen el uso de distintas nomenclaturas. En este ejercicio, unificaremos la nomenclatura de los valores de este atributo.

a) ¿Cuántos valores distintos toma el atributo `marital` en el conjunto de datos? Mostrad cuáles son estos valores.

```
In [1]: #Respuesta
```

b) Unificad los valores del atributo `marital` para que sean únicamente: "single", "married" o "divorced".

```
In [5]: #Respuesta
```

c) ¿Qué columnas contienen valores perdidos?

```
In [3]: # Respuesta
```

d) Calcula el primero i el tercer cuartil del atributo "balance".

In [2]:
```
# Respuesta
```

Ejercicio 2

El atributo `poutcome` contiene información sobre si el cliente del banco contrató un depósito a largo plazo en el banco. Calcula la correlación entre el atributo `poutcome` y el resto de atributos mediante la función 'corr' (http://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.corr.html). ¿Cuál es la variable más correlacionada con `poutcome` ?

In [22]:
```
# Respuesta
```

Ejercicios para la PEC

A continuación encontraréis los ejercicios que debéis completar en esta PEC y que forman parte de la evaluación de esta unidad

Ejercicio 1

Cargad los datos del fichero `Salarios.csv` en un dataframe. Este conjunto de datos recoge el salario académico de nueve meses de 2008-09 para profesores adjuntos, profesores asociados y profesores en una universidad de EUA. El conjunto original puede encontrarse en el repositorio de datos del Departament d'Estadística i Investigació operativa de la Universitat Politècnica de Catalunya (http://www-eio.upc.edu/~pau/cms/rdata/datasets.html), pero el conjunto que usaremos tiene algunas modificaciones. **(Total 2 puntos)**

Nota: Debes de eliminar la primera columna. La primera columna corresponde al número de línea.

a) Listar la longitud del objecto `Salarios.csv` y el tipo de atributos que contiene. **(0,5 puntos)**

In [1]:
```
#Respuesta
```

b) Inputar los valores NA de la columna `Salary` con la media aritmética del salario. **(0,5 puntos)**

In [2]:
```
#Respuesta
```

c) Cread dos nuevas columnas, `disc_1` y `disc_2` , en el dataframe con el resultado de discretizar el artributo `salary` . La columna `disc_1` contendrá la discretización en 10 intervalos que contengan el mismo número de elementos cada uno. La columna `disc_2` contendrá la discretización en 10 intervalos del mismo tamaño. Imprimir por pantalla las columnas `salary` , `disc_1` y `disc_2` de los 20 primeros ítems. **(1 punto)**

In [3]:
```
#Respuesta
```

Ejercicio 2

Crea 3 nuevos atributos binarios (`Salary_T1` , `Salary_T2` , `Salary_T3`) que indiquen en qué tercil se encuentra el valor de salario de cada muestra. Así, para una muestra con un valor de salario en el segundo tercil, el atributo `Salary_T2` será 1 y los atributos `Salary_T1` y `Salary_T3` serán 0. Computa el número de profesores en cada tercil agrupando según su género. **(2 puntos)**

Nota: Usa el dataframe con los NA imputados con la media aritmética del salario.

In [4]:
```
#Respuesta
```

Ejercicio 3

Selecciona aleatoriamente un subconjunto mínimo de 60 muestras del dataframe, de manera que el dataframe resultante esté balanceado con respecto al atributo `type` . El nuevo dataframe debe tener el mismo número de muestras para cada posible valor del atributo `type` . **(2 puntos)**

In [5]:
```
#Respuesta
```

Ejercicio 4

Cargad los datos del fichero `Baseball.csv` en un dataframe. Este conjunto de datos recoge 322 observaciones y 25 variables de la liga de Beisbol en EUA. El conjunto original puede encontrarse en el repositorio de datos del Departament d'Estadística i Investigació operativa de la Universitat Politècnica de Catalunya (http://www-eio.upc.edu/~pau/cms/rdata/datasets.html), pero el conjunto que usaremos tiene algunas modificaciones. **(Total 2 puntos)**

Nota: Debes de eliminar la primera columna. La primera columna corresponde al número de línea.

a) Definid una función para calcular el número de muestras que tienen al menos dos valores perdidos. ¿ Cúantos muestras tienen al menos dos valores perdidos? **(0,5 puntos)**

In [6]:
```
#Respuesta
```

b) Eliminad las muestras que tengan algún valor perdido en cualquiera de los atributos. **(0,5 puntos)**

In [8]:
```
#Respuesta
```

c) Definid una función que seleccione aquellas filas que no contegan valores atípicos (*outliers*) para un determinado atributo. Aplicar el significado de valor atípico leve (https://es.wikipedia.org/wiki/Valor_atípico). Los parámetros de entrada de la función deben ser el dataframe original y el nombre de la columna a evaluar. La función tiene que devolver el dataframe modificado (si corresponde), eliminando las filas que tengan valores atípicos en el atributo indicado. Utilizad la función anterior para calcular el número de valores no atípicos que tiene el atributo `outs86` . **(1 punto)**

In [9]:
```
#Respuesta
```

Ejercicio 5

Define una función para normalizar (escalar los valores entre [0,1]) un dataframe mediante la aproximación del Mín y Max. `(eq 1)`
`X'=(X-Xmin)/(Xmax-Xmin)` . Aplícalo a los datos del fichero `Baseball.csv` con valores perdidos eliminados y seleccionando los datos númericos. **(2 puntos)**

In [10]:
```
#Respuesta
```

Ejercicio Opcional

A partir de los datos del fichero `Baseball.csv` crudos, obtener los valores de covarianza (https://es.wikipedia.org/wiki/Covarianza) de las diferentes atributos y representar mediante un heatmap (https://en.wikipedia.org/wiki/Heat_map) los valores de covarianza obtenidos.

Pista: Quizás la función cov (http://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.cov.html (http://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.cov.html)) de pandas puede ser de utilidad.

In [11]:
```
#Respuesta
```

Soluciones a los ejercicios para practicar

Ejercicio 1

a) ¿Cuántos valores distintos toma el atributo marital en el conjunto de datos? Mostrad cuáles son estos valores.

In [24]:
```
import pandas as pd
import numpy as np

data = pd.read_csv("data/bank_edited.csv", sep=";", dtype={"balance
":np.float})

import numpy as np

# Utilizamos unique para recuperar los valores únicos:
v = data.marital.unique()

print "There are {} different values in marital:\n{}".format(len(v)
, v)
```

```
There are 11 different values in marital:
['married' 'single' 'marrid' 'divorced' 'maried' 'sing' 'Married'
'MARRIED'
 'DIVORCED' 'Single' 'SINGLE']
```

b) Unificad los valores del atributo marital para sean únicamente: "single", "married" o "divorced".

In [40]:
```
data.loc[(data.marital == "Married") | (data.marital == "maried") |
(data.marital == "MARRIED") |
         (data.marital == "marrid"), "marital"] = "married"
data.loc[(data.marital == "Single") | (data.marital == "SINGLE") |
(data.marital == "sing"), "marital"] = "single"
data.loc[(data.marital == "DIVORCED"), "marital"] = "divorced"

# Comprobación

v = data.marital.unique()
print "There are {} different values in marital:\n{}".format(len(v)
, v)
```

```
There are 3 different values in marital:
['married' 'single' 'divorced']
```

c) ¿Qué columnas contienen valores perdidos?

In [26]:
```
# Aplicamos la función any_isna a cada columna del dataframe
print data.isna().any()
```

```
age          False
job          False
marital      False
education    False
default      False
balance       True
housing      False
loan         False
contact      False
day           True
month        False
duration      True
campaign     False
pdays        False
previous     False
poutcome     False
y            False
dtype: bool
```

In [ ]:
```
d) Calcula el primero i el tercer cuartil del atributo "balance".
```

In [25]:
```
print data['balance'].quantile(0.25)
print data['balance'].quantile(0.75)
```

```
68.0
1476.0
```

Ejercicio 2

El atributo `poutcome` contiene información sobre si el cliente del banco contrató un depósito a largo plazo en el banco. Calcula la correlación entre el atributo `poutcome` y el resto de atributos mediante la función 'corr' (http://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.corr.html). ¿Cuál es la variable más correlacionada con `poutcome` ?

In [28]:
```
# Respuesta

# Visualizamos los valores de la columna poutcome
import pandas as pd
import numpy as np
import warnings
warnings.filterwarnings('ignore')

data.poutcome.unique()
```

Out[28]:
```
array(['unknown', 'failure', 'other', 'success'], dtype=object)
```

In [29]:
```
#Seleccionamos únicamente las muestras que contengan información precisa sobre si el cliente va contractó o no el depósit

data_pout = data[data.poutcome.isin(("failure","success"))]

# Discretitzamos la columna poutcome para poder calcular la correlación

data_pout['poutcome_cat'] = data_pout.poutcome.astype("category").cat.codes

#Calculamos la correlación poutcome_cat con el resto de columnas

data_pout.corr()["poutcome_cat"]
```

Out[29]:
```
age             0.090540
balance         0.039791
day             0.009252
duration        0.142385
campaign       -0.059986
pdays          -0.276853
previous        0.023411
poutcome_cat    1.000000
Name: poutcome_cat, dtype: float64
```

Seleccionamos la columna `pdays` por tener mayor correlación con poutcome_cat. Notad que la correlación en esto caso es negativa, pero como hemos asignado un 0 o un 1 "failure" y "succes" de forma arbitrária, el signo no es significativo en este caso.