Data Modelling with Python

Name of student

Name of professor

University

Course

Date

Data modelling is a key aspect in the process of trying ton find key insights from the selected data inputs that a user has been given. Usually data modeling goes just beyond the steps of data analysis. Whereas data analysis involve the process of trying to find summaries from the fata and what type of information that the data is trying to speak to us about, data modeling on the other hand involves the process of applying algorithms to the dataset and then by using those algorithms, we are able to quickly summaries the data and then get the meaningful relationships from it.

The algorithms will try to apply a learning model on the data and then by using the 80/20 AB testing rule, generate another set of 20% dataset to the set and then be able to establish trends from the same dataset. Later, we can use samples to this dataset and be able to predict the next possible outputs from this data. This is what is called Machine learning. The inbuilt algorithms doing this work are called models. Some of the inbuilt models inside the python programming language include:

* Logistic regression
* KNN model
* Decision trees
* Arima model
* Random forest

In this particular exercise, we are going to apply a few of these models on the following datasets to this exercise and try to come up with the most satisfying models for the datasets supplied.

**Datasets descriptions;**

The dataset being used for this particular exercise is compose d of a total instance of 2064 records with the following columns variables as indicated;

#Dataset description;

 parents        usual, pretentious, great\_pret

 has\_nurs       proper, less\_proper, improper, critical, very\_crit

 form           complete, completed, incomplete, foster

 children       1, 2, 3, more

 housing        convenient, less\_conv, critical

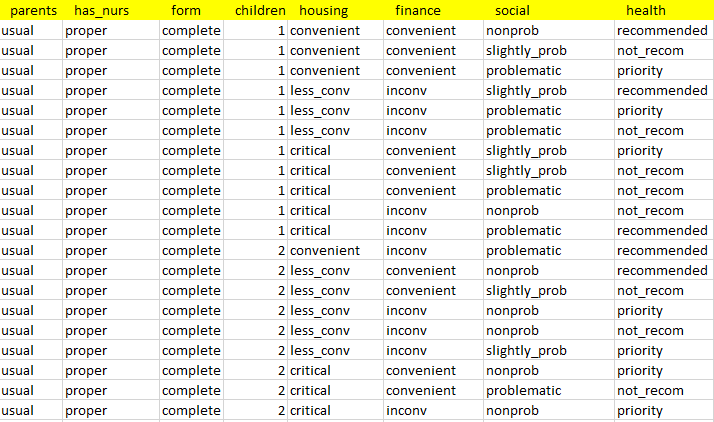
 finance        convenient, inconv

 social         non-prob, slightly\_prob, problematic

 health         recommended, priority, not\_recom

These columns variable sin the dataset present as an action or a behavior attribute related with each child. For instance, the data column indicates which type of parent the kid has, the nursing type category, the type of nursing that the kid is receiving, the number of children , whether or not there is a housing, the financial capabilities associated with the kid, the social fit of the kid and finally the health of the kid.

A quick summary and overview of this dataset in Microsoft excel file looks like below:



**Data analysis and modelling:**

In the next step, a series of models are applied on the dataset using the housing and parents states to try to establish models from the same dataset and answers given. Further, the analysis is done using python notebook as the development environment and the Python programming language.

The next step is to load the dataset into the IDE and read it as below:

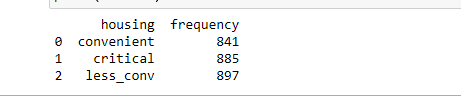
   #Load the housing dataset;

import pandas as pd;

import numpy as np;

dataset = pd.read\_csv("housing.csv")

print (dataset)



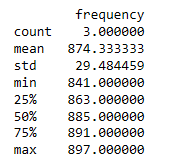
#Decsribe the data;

dataset = pd.read\_csv("housing.csv")

#print(dataset)

print(dataset.describe())

Using the describe method will print the summary of the measures of central tendencies of the dataset as below;



This kind of illustration indicates that based on the supplied housing data, the base measures of central tendencies were supplied and the correspond outputs were determined based on the mean, standard deviation, the min value, maximum value, the 25th percentile as well as the 7th percentile.

# Get the plot

def x\_pos(y\_pos):

    return x\_pos.astype(int)

housing = ['convenient','critical','less\_conv']

x\_pos = np.arange(len(housing))

CTEs = [housing,frequency]

fig, ax = plt.subplots()

ax.bar(x\_pos, CTEs, align='center', alpha=0.5)

ax.set\_ylabel('Frequency')

ax.set\_xticks(x\_pos)

ax.set\_xticklabels(the\_hour)

ax.set\_title('housing')

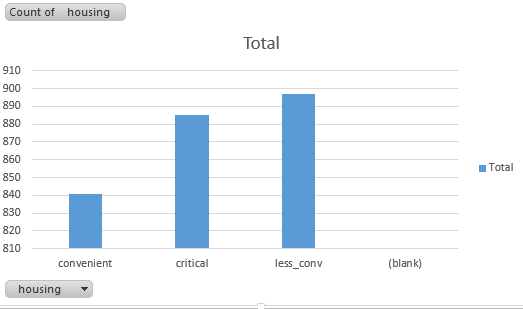
ax.yaxis.grid(True)

# Save the figure and show

plt.tight\_layout()

plt.savefig('bar\_chart.png')

plt.show()



# Modelling the data with Naive bayes algoritm

import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

import sklearn

from sklearn.preprocessing import LabelEncoder

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import StandardScaler

from sklearn.naive\_bayes import GaussianNB

from sklearn.metrics import confusion\_matrix,accuracy\_score

dataset = pd.read\_csv('nsr.csv')

X = dataset.iloc[:, [1, 2, 3]].values

y = dataset.iloc[:, -1].values

le = LabelEncoder()

X[:,0] = le.fit\_transform(X[:,0])

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.20, random\_state = 0)

sc = StandardScaler()

X\_train = sc.fit\_transform(X\_train)

X\_test = sc.transform(X\_test)

classifier = GaussianNB()

classifier.fit(X\_train, y\_train)

y\_pred  =  classifier.predict(X\_test)

y\_pred

y\_test

cm = confusion\_matrix(y\_test, y\_pred)

ac = accuracy\_score(y\_test,y\_pred)

Using the Naïve Bayes model above, we are trying to establish as to whether a child will receive proper health care if they are from a descent housing and are also based on the nature of parents that they have. We already know that the probability of a child getting proper housing if they have more than 1 parent is higher if they also have caring parents. We then use the confusion matrix to compute the accuracy of the predicted values and the test values, the accuracy returned using the confusion matrix is 0.85. What this means is that there is a higher positive correlation between the type of parents a child has, the type of housing and the type of healthcare that the child will get.