**Course Code:** ANL252

**Title:** ANL252 Python for Data Analytics – End-of-Course Assessment

**SUSS PI No:** B2272579

**Full Name:** Pereira Kathlyn Therese

**Submission Date:** 3 November 2023

**Question 1)**

##Obtaining Data in a table format

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

ECA = pd.read\_csv(r"D:\for school\ANL252-Python\ECA\ECA.csv")

display(ECA)

**A screenshot of a data

Description automatically generated**

##Step 1: finding missing data and calculating the sum of missing data.

Missing = ECA.isnull().any()

print(Missing)

Sum\_Missing = ECA.isnull().sum ()

print (Sum\_Missing)

PersonID False

age True

sex False

bmi False

children False

smoker False

region False

charges False

dtype: bool

PersonID 0

age 123

sex 0

bmi 0

children 0

smoker 0

region 0

charges 0

dtype: int64

##Step 2: Removing and filling missing data

nECA = ECA.fillna (value = 0)

## Checking again to ensure a clean dataset

nECA.isnull().sum ()

PersonID 0

age 0

sex 0

bmi 0

children 0

smoker 0

region 0

charges 0

dtype: int64

## Step 3: replacing all “F” and “M” to female and male respectively

B = nECA.replace ('F', "female")

A = B.replace("M", "male")

After importing the data into Jupyter notebook and importing pandas, we must search for any missing data as missing data can affect the outcome of any equation that is needed when extracting the data. As such, the first task that we need to do is to search for the missing data. From the table above, we can note that the “age” column has missing data. We then go further into the investigation by identifying the sum of missing data using the .sum() function where we can find out that there are123 missing data under the column “age”. By finding out the missing data, we can then move on to the next step of filling the data.

The second step is fill the data using the fillna () function so that any missing data will be replaced with the value 0. Once this is done, we can check if there are any other missing data that might have been missed. However, from the above table, we can conclude that there are no other missing data and that all missing data under the column “age” has been replaced with the value 0 as mentioned above.

The last step is upon investigation, we note that participants have detailed their sex in two different ways. For females, they have indicated it as “F” and “female”, and for males, they indicated as “M” and “male”. However, for a clean data, we need to ensure consistency in the outputs. By using the .replace () function, we are able to replace all “F” and “M” inputs to “female” and “male” outputs respectively. [265 words]

**Question 2)**

import matplotlib.pyplot as plt

##Scatter Plot

d= A ["charges"]

e= A ["bmi"]

plt.scatter(d,e)

plt.title ("How does bmi correlate to the health insurance costing")

plt.xlabel ("Medical costs billed by health insurance")

plt.ylabel ("BMI (kg / m ^ 2)")

plt.show ()

A diagram of blue dots

Description automatically generated

The scatter plot above is meant to find how bmi affects the health insurance costing. From the results above, we can determine that most people have a bmi between 20 – 38 kg/m2 and have insurance billings that are mostly $10000 and below with a few outliers who have higher costings. We can also note that those who have a bmi of 50kg/m2 and above have less health insurance costing which could be due to other factors such as age where insurance may not be as popular hence less usage of it.

# Bar Graph (Gender + Age) - Comparison

##plotting a bar graph

plt.bar(ECA\_e["sex"], ECA\_e["age"])

plt.title ("Age comparison between male and female")

plt.xlabel ("Gender")

plt.ylabel ("Age")

plt.bar(A["sex"], A["age"])

plt.title ("Age comparison between male and female")

plt.xlabel ("Gender")

plt.ylabel ("Age")

plt.annotate("52 females ages 60 and above", xy = ("female", 52),

xytext = ("female", 30),

arrowprops = dict (facecolor = 'red',

shrink = 0.05))

plt.annotate("55 males aged 60 and above", xy = ("male", 55),

xytext = ("male", 37),

arrowprops = dict (facecolor = 'red',

shrink = 0.05))

plt.show ()

A graph of age comparison between male and female

Description automatically generated

From the graph above, we are comparing the age difference between males and females and from this, we can infer that there are more males who are aged 60 and above as opposed to females. It is almost equal but the males have about 3 more people who are aged 60 and above as compared to females who only have 52 people.

# Histograph (Children)

##Ploting a histogram graph

plt.hist (A["children"], bins = 3, align = "mid”)

plt.title ("Distribution of the amount of children")

plt.xlabel ("Regularity")

plt.ylabel ("No.of Children")

plt.show ()

A graph with blue squares

Description automatically generated

By plotting a histogram, we can see the how the number of children is distributed to show the most preferred amount of children that the participants have. From the right-skewed data above, we can thus infer that most people preferred having 0 -1 children as opposed to having 2 children and above.

**Question 3)**

##Importing of scikit

import sklearn

from sklearn import tree

from sklearn.tree import DecisionTreeRegressor

from sklearn import datasets

from sklearn.preprocessing import StandardScaler

##replacing the strings no and yes to values 0 and 1 respectively

no = A.replace ("no", "0")

yes = no.replace ("yes", "1")

#determining the feature

Feature = yes [["smoker", "bmi"]]

x= Feature

y = yes["age"].values

print (x.head())

print (y[0:5])

print (x.shape, y.shape)

smoker bmi

0 1 27.900

1 0 33.770

2 0 33.000

3 0 22.705

4 0 28.880

[19. 18. 28. 33. 32.]

(1340, 2) (1340,)

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

random\_state = 0

test\_size = 0.3

x\_train, x\_test, y\_train, y\_test = train\_test\_split (x,y,test\_size = test\_size, random\_state = random\_state)

print("Train Set:", x\_train.shape, y\_train.shape)

print (x\_train["smoker"][0:5])

print("Test Set:", x\_test.shape, y\_test.shape)

print(x\_test ["smoker"][0:5])

Train Set: (938, 2) (938,)

466 0

1185 0

471 0

984 1

97 0

Name: smoker, dtype: object

Test Set: (402, 2) (402,)

574 0

661 0

458 0

1023 1

958 1

Name: smoker, dtype: object

##Normalising the data

from sklearn import preprocessing

X\_train = preprocessing.StandardScaler ().fit(x\_train).transform (x\_train)

X\_test = preprocessing.StandardScaler().fit(x\_test).transform(x\_test)

print("Normalized x Training Set:", X\_train[0:5])

print ("Normalized x Testing Set:", X\_test [0:5])

Normalized x Training Set: [[-0.50233101 -0.91658057]

[-0.50233101 -0.56047109]

[-0.50233101 -1.09341296]

[ 1.99071921 -0.79842067]

[-0.50233101 1.21926137]]

Normalized x Testing Set: [[-0.52553827 2.12940985]

[-0.52553827 -0.24990883]

[-0.52553827 -0.02500637]

[ 1.90281099 0.12105575]

[ 1.90281099 0.08454022]]

##Finding the min\_split for Decision Tree

from sklearn.model\_selection import GridSearchCV

maxd = [{"max\_depth": [x for x in range(1,12)], "min\_samples\_split": [x\*5 for x in range (1,30)]}]

DTree = DecisionTreeRegressor (criterion = "friedman\_mse")

tree = GridSearchCV (DTree, maxd, cv= 2)

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

tree.best\_estimator\_

DecisionTreeRegressor

DecisionTreeRegressor(criterion='friedman\_mse', max\_depth=5,

min\_samples\_split=20)

##Doing another round of prediction of data set to get the top data model.

tree = DecisionTreeRegressor ()

tree = tree.fit (X\_train, y\_train)

ztrain = tree.predict (X\_train)

ztest = tree.predict (X\_test)

The approach that I have taken is the decision tree regression. This is because I want to attain future predicted data outputs that are meaningful where I can determine the insights found. It also provides a clear and visualisation of the data. It is useful for the dependant variable that I am using which is the column “smoker”. I am trying to predict the number of smokers from the age and bmi and determine what kind of output I can get.

After replacing all the strings from the “smoker” column to integers, I am now able to use the data to train and set it where I am checking the data to ensure that it can be used for the decision tree regression.

After attaining the data, I have normalised the data so that there will be no disproportionate features that could affect the data (Bhandari, 2023). The next step is to use GridSearchCV to find the maximum depth and the minimum samples which we can see is 5 and 20 respectively.

At this point, the data model is almost the best model that I can use to plot the Decision Tree. However, before I plot the tree, I have done a fit to use the best data model to do another prediction.

(210 Words)

**Question 4)**

xx = x.columns.values

yy = y

fig, axes = plt.subplots (nrows = 1, ncols = 1, figsize =(4, 3), dpi = 1300)

plot\_tree(tree, feature\_names = xx, class\_names = yy, fontsize = 4, filled = True, rounded = True, impurity = 0.0)

plt.show ()

**Question 5)**

I think that to a small extent, it can be used for exploratory data analysis. This is because decision trees. According to an article by Pratik (2023), exploratory data analysis consists of “calculating summary statistics, visualizing data distributions, identifying outliers, exploring relationships between variables, and performing hypothesis testing”. With its prediction and classification features, decision trees can be used to perform hypothesis testing with checks to ensure the accuracy of the data. Decision trees are also relatively resistant to outliers and as such there may not need to identify outliers (“Are Decision Trees Robust To Outliers”, n.d.).

While it does not explore the relationships between variables, it handles correlations due to its nature of “recursive splitting and feature selection” which means that the correlations do not affect the data and there is no need to remove any correlations (Sen, 2023). This can be a fault depending on the outcome of data that we are testing for.

Not only this, but also though, the decision tree does provide data distributions in a sense, it is able to provide some form of summary statistics but there are also limitations as datasets can become overfitted whenever there are large and complicated trees thus leading to more issues (Research Gate, n.d. & “Decision tree limitations, 2023)

Thus, decision trees can be used for exploratory data analysis to a small extent as while it can do certain exploratory data analysis, there can be more issues that crop up unless there are new ways to modify the machine learning tool that would enable decision trees to be used for exploratory data analysis.

(266 words)

**References:**

*Are Decision Trees Robust to Outliers*. (n.d.). Inside Learning Machines. <https://insidelearningmachines.com/decision_trees_robust_to_outliers/#:~:text=In%20 this%20article%20you%20have,become%20more%20affected%20by%20outliers>.

Bhandari, A. (2023, October 27). *Feature Engineering: Scaling, Normalization, and Standardization (Updated 2023)*. Analytics Vidhya. [https://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning- normalization- standardization/#:~:text=A.%20Normalization%20helps%20in%20scaling,be%20imp ortant%20for%20some%20algorithms](https://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning-%09normalization-%09standardization/#:~:text=A.%20Normalization%20helps%20in%20scaling,be%20imp ortant%20for%20some%20algorithms).

Decision tree limitations. (2023, March 13). EDUBCA. [https://www.educba.com/decision- tree-limitations/](https://www.educba.com/decision-%09tree-limitations/)

Kumar, M. (2023). *Study Unit 3: Arrays and Plots*  [PowerPoint Slides]. Singapore University of Social Sciences.

Pratik, N (2023, March 16). *Exploratory Data Analysis (EDA) and Visualization Techniques*. Analytics Vidhya. [https://www.analyticsvidhya.com/blog/2021/08/exploratory-data-analysis-and- visualization-techniques-in-data- science/#:~:text=EDA%20techniques%20may%20include%20calculating,further%20analysis %20or%20decision%2Dmaking](https://www.analyticsvidhya.com/blog/2021/08/exploratory-data-analysis-and-%09visualization-techniques-in-data-%09science/#:~:text=EDA%20techniques%20may%20include%20calculating,further%20analysis %20or%20decision%2Dmaking).

Research Gate. (n.d.). [https://www.researchgate.net/figure/The-decision-tree-regression-model- calculated-on-summary-statistics-for-each-of- six\_fig1\_335999595#:~:text=license%3A%20CC%20BY-, The%20decision%20tree%20regression%20model%20calculated%20on%20summary%20stat istics%20for,to%20stop%20at%20this%20point](https://www.researchgate.net/figure/The-decision-tree-regression-model-%09calculated-on-summary-statistics-for-each-of-%09six_fig1_335999595#:~:text=license%3A%20CC%20BY-, The%20decision%20tree%20regression%20model%20calculated%20on%20summary%20stat istics%20for,to%20stop%20at%20this%20point).

Sen, V. (2023, October 5). "*NO" need to check for multicollinearity or remove correlated variables explicitly when using decision trees*. LinkedIn. [https://www.linkedin.com/pulse/need-check- multicollinearity-remove-correlated-variables- sen#:~:text=In%20summary%2C%20while%20multicollinearity%20is,recursive%20splitting% 20and% 20feature%20selection](https://www.linkedin.com/pulse/need-check-%09multicollinearity-remove-correlated-variables-%09sen#:~:text=In%20summary%2C%20while%20multicollinearity%20is,recursive%20splitting% 20and% 20feature%20selection).