# NPTEL

## WEEK 4 ASSIGNMENT QUESTIONS

1) The power consumption of an individual house in a residential complex has been recorded for the previous year. This data is analysed to predict the power consumption for the next year. Under which type of machine learning problem does this fall under?

a) Classification

b) Regression

c) Reinforcement Learning

d) None of the above

Answers: b)

The prediction of power consumption in a house is a regression problem as the quantity predicted is continuous in nature.

2) A dataset contains data collected by the Tamil Nadu Pollution Control Board on environmental conditions (154 variables) from one of their monitoring stations. This data is further analyzed to understand the most significant factors that affect the Air Quality Index. The predictive algorithm that can be used in this situation is _____.

a. Logistic Regression
b. Simple Linear Regression
c. Multiple Linear Regression
d. None of the above

Answer: c)

The Multiple Linear Regression algorithm fits a linear relationship between a dependent continuous variable and more than one independent variable.

3) A regression model with the following function y = 60 + 5.2x was built to understand the impact of humidity (x) on rainfall (y). The humidity this week is 30 more than the previous week. What is the predicted difference in rainfall?

a. 156 mm

b. 15.6 mm

c. -156 mm

d. None of the above

Answer: a)

The difference in humidity H was by 30, so the other value is H+30.

Predicted difference in rainfall = Y2-Y1 = (60 + 5.2(H+30)) – (60+5.2H)

= 5.2 *30 = 156 mm

Read the information given below and answer the questions from 4 to 7:

**Data Description:**

An automotive service chain is launching its new grand service station this weekend. They offer to service a wide variety of cars. The current capacity of the station is to check 315 cars thoroughly per day. As an inaugural offer, they claim to freely check all cars that arrive on their launch day, and report whether they need servicing or not!

Unexpectedly, they get 450 cars. The servicemen will not work longer than the working hours, but the data analysts have to!

Can you save the day for the new service station?

How can a data scientist save the day for them?

He has been given a data set, '**ServiceTrain.csv'** that contains some attributes of the car that can be easily measured and a conclusion that if a service is needed or not.

Now for the cars they cannot check in detail, they measure those attributes and store them in **'ServiceTest.csv'**

**Problem Statement:**

Use machine learning techniques to identify whether the cars require service or not

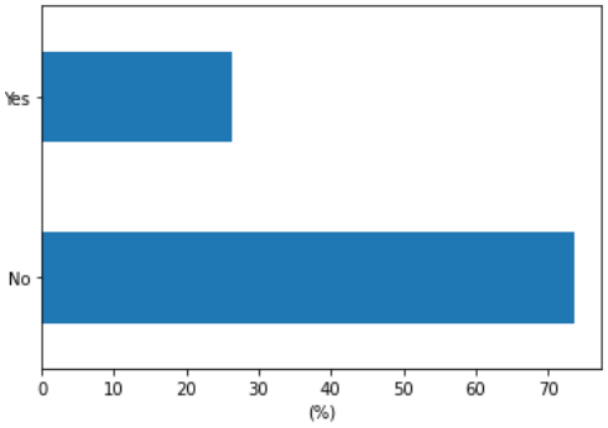
Read the given datasets **'ServiceTrain.csv'** and **'ServiceTest.csv'** as **train_data** and **test_data** respectively and import all the required packages for analysis.

4) Which of the following machine learning techniques would NOT be appropriate to solve the problem given in the problem statement?
   a. kNN
   b. Random Forest
   c. Logistic Regression
   d. Linear regression

Answers: d

Classification techniques such as kNN, random forest and logistic regression can be used to solve the given problem as the target variable is discrete in nature.

5) The plot shown below denotes the percentage distribution of the target column values within the **train_data** dataframe. Which of the following options are correct?



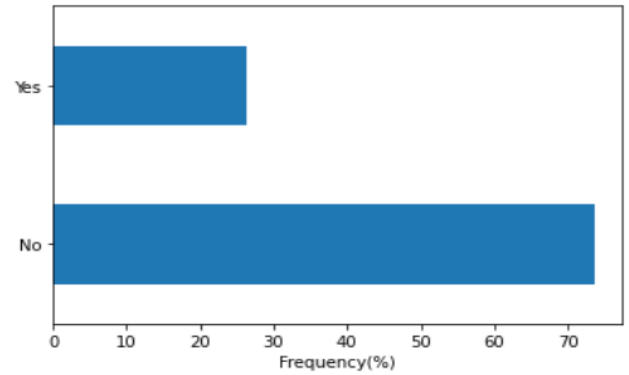

a) Yes > 20, No > 60
b) No > 70, Yes > 20
c) Yes > 30, No > 70
d) Yes > 70, No > 30

Answer: b)

```
pdst = train_data['Service'].value_counts(normalize = True)*100
print(pdst)
ax = pdst.plot(kind = "barh")
ax.set_xlabel("Frequency(%)")
```

```
No     73.650794
Yes    26.349206
Name: Service, dtype: float64

Text(0.5, 0, 'Frequency(%)')
```




Prepare the data by following the steps given below, and answer questions 6 and 7

- Encode categorical variable, ***Service*** - **Yes** as 1 and **No** as 0 for both the train and test datasets
- Split the set of independent features and the dependent feature on both the train and test datasets
- Set **random_state** for the instance of the logistic regression class as 0

6) After applying logistic regression, what is/are the correct observations from the resultant confusion matrix?

   a. True Positive = 29, True Negative = 94

   b. True Positive = 94, True Negative = 29

   c. False Positive = 5, True Negative = 94

   d. None of the above

Answers: a and c

7) The logistic regression model built between the input and output variables is checked for its prediction accuracy of the test data. What is the accuracy range (in %) of the predictions made over test data?

a) 60 - 79
b) 90 - 95

c) 30 – 59
d) 80 – 89

Answer: b)

```
import pandas as pd

# Importing library for logistic regression
from sklearn.linear_model import LogisticRegression

# Importing performance metrics - accuracy score & confusion matrix
from sklearn.metrics import accuracy_score, confusion_matrix

# importing the library of KNN
from sklearn.neighbors import KNeighborsClassifier

# Sklearn - package to split data into train & test
from sklearn.model_selection import train_test_split

#Importing visualization libraries
import matplotlib.pyplot as plt
```

```
train_data = pd.read_csv("ServiceTrain.csv")
test_data = pd.read_csv("ServiceTest.csv")
train_data.head()
```

Data preparation

Encoding the categorical variable, 'Service' - 'Yes' as 1 and 'No' as 0

```
train_data['Service'] = train_data['Service'].map({'Yes':1,'No':0})
train_data['Service']
```

```
0      0
1      1
2      1
3      0
4      0
      ..
310    0
311    0
312    1
313    1
314    1
Name: Service, Length: 315, dtype: int64
```

```
# Separating out the input and output features of train data
train_x11 = train_data.drop(['Service'], axis=1)
train_y11 = train_data['Service']
```

```
# Checking the size of input and output features of train data
print(train_x11.shape)
print(train_y11.shape)
```

```
(315, 5)
(315,)
```

```
# Creating an instance of the LogisticRegression class
logistic_mod = LogisticRegression(random_state = 0)

# Fitting the logistic regression model using input and output features of train data
logistic_mod.fit(train_x11,train_y11)
```

Encoding the categorical variable, 'Service' - 'Yes' as 1 and 'No' as 0

```
test_data['Service'] = test_data['Service'].map({'Yes':1,'No':0})
test_data['Service']
```

```
# Separating out the input and output features of test data
test_x = test_data.drop(['Service'], axis=1)
test_y = test_data['Service']
```

```
# Predicting the 'logistic' model on test_data
predict_log = logistic_mod.predict(test_x)
print(predict_log)
```

```
[1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 1 1 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0
 0 0 0 0 1 0 1 0 0 0 1 0 0 1 1 1 0 0 0 0 0 1 0 0 0 0 0 1 1 1 0 0 0 0 1 0 0
 0 0 0 0 1 1 1 0 0 0 0 1 0 1 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
 1 0 1 0 1 1 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0]
```

```
# Confusion matrix
confusion_matrix_logr = confusion_matrix(test_y, predict_log)
tn,fp,fn,tp = confusion_matrix_logr.ravel()
print(confusion_matrix_logr)
print('tp:', tp, 'tn:', tn, 'fp:',fp,'fn:',fn)
```

```
[[94  5]
 [ 7 29]]
tp: 29 tn: 94 fp: 5 fn: 7
```

```
lr_precision = tp/(tp+fp)
lr_recall = tp/(tp+fn)
lr_f1_score = 2/(1/lr_precision + 1/lr_recall)
print("Precision is: ",round(lr_precision*100,2),
      "Recall is: ", round(lr_recall*100,2),
      "F1 Score is: ",round( lr_f1_score *100,2))
```

```
Precision is:  85.29 Recall is:  80.56 F1 Score is:  82.86
```

```
print('Misclassified samples: %d' % (test_y != predict_log).sum())
```

```
Misclassified samples: 12
```

```
# Accuracy
acc_score_logr = accuracy_score(test_y, predict_log)
print(round(acc_score_logr*100,2))
print('Misclassified samples in Logistic Regression classification: %d' % (test_y != predict_log).sum())
```

```
91.11
Misclassified samples in Logistic Regression classification: 12
```

8) In a KNN model, by which means do we handle categorical variables?
   a) Standardization
   b) Dummy variables
   c) Correlation
   d) None of the above

Answer: b) Dummy variables can be used to encode the different values contained in a particular categorical independent feature.

The Global Happiness Index report contains the Happiness Score data with multiple features (namely – Economy, Family, Health, Freedom) that could possibly affect the target variable value.

Prepare the data by following the steps given below, and answer questions 9 and 10

- Split the set of independent features and the dependent feature on the given dataset
- Create training and testing data from the set of independent features and dependent feature by splitting the original data in the ratio 3:1 respectively, and set the value for ***random_state*** of the training/test split method's instance as 1

9) A multiple linear regression model is built on the Global Happiness Index dataset "GHI_Report.csv". What is the RMSE of the baseline model?
   a. 2.00

b. 0.50
c. 1.06
d. 0.75

Answer:c)

```
# Set the features and the target
features = list(set(data_ghi.columns)-set(["H_Score"]))
target = list(['H_Score'])

print(features)
print(target)
```

```
['Freedom', 'Health', 'Economy', 'Fam']
['H_Score']
```

```
X = data_ghi.loc[:,features]
y = data_ghi.loc[:,target]
train_x, test_x, train_y,test_y = train_test_split(X,y,test_size = 0.25, random_state = 1)
```

```
# Base Model with test data mean values
base_pred = np.mean(test_y)
print(base_pred)

#repeat the same for all samples in test data
base_pred = np.repeat(base_pred,len(test_y))

# Find Baseline model RMSE
base_rmse = np.sqrt(mean_squared_error(test_y,base_pred))
print("Base RMSE : ",round(base_rmse,2))
```

```
H_Score    5.343225
dtype: float64
Base RMSE :  1.06
```

10) X and Y are two variables that have a strong linear relationship. Which of the following statements are incorrect?
    a) There cannot be a negative relationship between the two variables.
    b) The relationship between the two variables is purely causal.
    c) One variable may or may not cause a change in the other variable.
    d) The variables can be positively or negatively correlated with each other.

Answers: a) and b)

A strong linear relationship means there is either a strong negative or positive correlation between variables. Similarly this necessarily doesn't imply that changes in one variable "causes" the other variable to change as well.