EVALYNE LWOBA

ANDREW_ID : elwoba

DIAML

ASSIGNMENT 5

Libraries Used

```
import numpy as np
import pandas as pd

import matplotlib.pyplot as plt
from matplotlib.colors import ListedColormap

from scipy.stats import pearsonr
import seaborn as sns
import statsmodels.api as sm

from sklearn.linear_model import LogisticRegression
from sklearn.metrics import mean_squared_error
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report

from tabulate import tabulate


import warnings
warnings.filterwarnings('ignore')
```

Question 1

1. Statistical learning (25 points)

1.1 Describe at least four steps to implementing a rule-based approach to decision-making and give an example.

Rule based decisions are a set of observations and data that drive a certain decision to be made in a model. These decisions rely on repetition and observations. The system mimics a human since it takes instructions from them. The steps to follow when creating a rule are:[1]

Step 1:transform what we know as domain knowledge to numerical figures to make it easy for the model to interpret.
Step 2: create specific thresholds for select the next course of action.
Step 3:come up with a score which should be very simple and can be repeated
Step 4: the model should be applicable to any user and the same result obtain at any given time.

Example of a rule based system

The following are facts about the workplace where people who are married are not allowed to work in the same department.

a) John is a manager in the purchasing department
b) Peter is a manager in sales department
c) Joe, Mary and Ann are working under John
d) Simon, Tom and Eva are working under John
e) Ann and Tom are married

Rules

Rule 1: Manager(place,person) works at (place, person)
Rule 2: Works at(place,person) and manager(place,person) therefore boss of(people under manager)
Rule 3: works at(place,person) and works at(place, person) therefore not married(people)
Rule 4: married(a,b(persons)) therefore married(b,a(persons))
Rule 5:married(person,place) and works at(department,person) therefore insured by(place,insurance)

The above set of rules can help the model answer the following questions:

a) Name a person who works in the company
b) Name married people and working in same department
c) Name Simons boss
d) Name department bosses.
e) Name someone who is insured.

Is any domain knowledge required to establish a rule? Support your answer with an explanation.

Domain knowledge is required to establish rules because it plays a major role in accuracy and success of a model. For instance in the example above, domain knowledge or facts and policies of the company had to be known before coming up with rules based to build our model. Without domain knowledge the rules might be end up giving garbage or ambiguous for the required outputs.

1.2 Explain over-fitting and why it is a problem in statistical Learning.

Model fitting is a very vital part of quantitative measurement in science. Experiments are used to investigate the associations between measures. There could be problems that come with this experiments. According to Occam's Razor principle of parsimony, that if a model has 2 predictor variables to explain the y- variable then no other than the two predictors should be used in modelling. If the association among the predictors is linear then applying a quadratic breaches Occam's principle of parsimony. Overfitting happens when models infringe parsimony and use more than needed features. The model fits needless noise then that is called overfitting. There are two types of overfitting, (a) a multi linear regression that has unconnected predictors and (b)modelling with a data set that enhances difficulty instead of improving gains to the model performance.[2]

why it is a problem in statistical Learning.

Overfitting is a problem in statistical learning because it can; (a)encourages getting into unwanted inferences,(b)modify portability and (c)lead to including unrelated features which can lead to undesired forecasts.

If you have a small dataset containing ten data points, should you prefer a simple model with one parameter or a complex model with ten parameters? Support your answer with an explanation.

This is a small data set so I would prefer a simple model with 1 parameter because[3];

i. A complex model with a lot of parameters can lead to overfitting which can be avoided by using a simple model
ii. A complex model with many parameters can be difficult to read and make inference because the parameters might be interconnected.
iii. A simple model is trained using a smaller number of parameters is efficient and requires a smaller computational time.

1.3 There are two commonly used approaches to avoid over-fitting; describe each one.

a. Feature selection.
   When we have several features to use to construct a model, some of them might have less impact on the prediction. We apply feature section to choose which feature is more meaningful or important in the training data and leave out the less significant one. This process aids in making the model simple in turn minimizes noise from the data. This can be done manually or automatically by algorithms.[4]

b. Regularization
   This is where a higher valued penalizing term is introduced into the dataset which aids in preventing overfitting. The penalizing term reduces the variance of the model nut slightly increases biasness.[4]

Provide two examples of metrics used to evaluate the performance of a model and give a formula for each one. Give two examples of applications and appropriate metrics for each case.

a. MSE(Mean square Error)– this is the measure of forecast performance.
   Formula is:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2$$

   Application for mean square error is when getting difference of actual values and estimated values.

b. MAE(Mean Absolute Error)- this is a forecast measure of how big an error is. The formula is:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i|$$

   Application: used when predicting the energy from wind.

1.5 Why are benchmarks useful in machine learning and give two examples.

Bench marking is useful in machine learning because it is used to evaluate and compare a models input and output to estimates from external data. It checks the model's ability to learn patterns in conditioned data sets.[3] Examples are:

a) Persistence – this is used with a hypothesis that all the changing aspects in the model are as a result of random move.
b) Unconditional average – this benchmark holds a hypothesis that no extra data can be got from time ordering of events.

Question 2

2. Machine Learning (25 points)

2.1 What is machine learning?

Machine learning addresses the question of creating a computer program that learns from familiarity with regards to some set tasks and performance evaluation on whether the evaluation gets better when applied to a new almost related assignment.[7]

Discuss its evolution over time and why is it popular?

In 1950, a test was done that reached a conclusion that a machine can learn if communicated. This led to the first computer game to be made in 1952 by Arthur Samuel. In 1957, neural networks were born through an invention of a perceptron by Frank Rosenblatt. 1990 gave rise to AI which encompassed a

blend of computer science and statistics. Big data came into play in 2010which led to an explosion of huge volumes of data requiring analysis and research. In 2014, APIs made it easy to access data without restrictions from anywhere.[7]
There have been huge leaps made in the evolution of machine learning over time because of the ability of networked and mobile computing systems to gather and transport huge a mounts big data. A diverse list of machine learning algorithms has been developed to cover the wide variety of data and problems being exhibited across different machine learning problems.[5]
Machine learning has been in the center of solutions to the problems of getting meaningful insights, predictions and inference from the big data. The size of the data pushed for evolution of machine learning algorithms to blend the findings. [6] Big data learning and multilayered networks in Deep learning led to the development of machine learning that has algorithms which gains knowledge from the big data.
Machine learning is important because with growth in AI, it can be easier to train a system by showing it examples of desired output than to manually program by anticipating the desired response for all possible inputs. Customer service, diagnosis of errors in systems and great impacts in data related fields, medicine and research has given the push to an increase in popularity of machine learning.

2.2 Give three examples of machine learning techniques that can be viewed as either supervised or unsupervised approaches.

a. Classification e.g KNM
b. Regression e.g Decision Trees
c. Clustering e.g PCA

2.3 What is the difference between classification and regression?

Classification is where the output expected is a classification or prediction of discreet values such as males, females, True and False.[7]
While
Regression the output expected is numerical. It is an appropriate model for predicting continuous values such as price, salary and age.[7]

2.4 What is the difference between supervised learning and unsupervised learning?

Supervised learning is where the experience has a labelled datapoints. The models finds the mapping function to map the input variable (x) with the output variable(y)

while

Unsupervised learning is where the experience has an unlabeled dataset. The model finds structured patters from input data on its own without supervision.[7]

2.5 Give examples of successful applications of machine learning and explain what technique is appropriate and what type of learning is involved?

a. Virtual assistants : there are very common today and assist users to looking for data over voice requests. They understand the human voice and accomplishes tasks as requested by a human. The technologies used is AI, machine learning and natural language processing which collect and analyze the information based on gathered experiences from the user. The learning involved is unsupervised and technique is classification.[7]

b. Information extraction : this is where data is fetched by an algorithm and presented in an organized manner. This data is then used to make decisions like for instance in the medical field. A series of symptoms can be linked to a certain disease and medication is recommended by as AI. The technique used here is classification and supervised learning.[7]

Question 3

3.1. Load the diabetes data into MATLAB or Python. Produce a correlation matrix of the explanatory variables

I loaded the data into my working environment then created a sub data frame with 10 independent variables (x)and one dependent variable(y) .

I generated a correlation matrix using pearson corr() function of the explanatory variables(x) and plotted a heatmap.[1]

Results

	AGE	SEX	BMI	BP	S1	S2	S3	S4	S5	S6
AGE										
SEX	0.17									
BMI	0.19	0.09								
BP	0.34	0.24	0.40							
S1	0.26	0.04	0.25	0.24						
S2	0.22	0.14	0.26	0.19	0.90					
S3	-0.08	-0.38	-0.37	-0.18	0.05	-0.20				
S4	0.20	0.33	0.41	0.26	0.54	0.66	-0.74			
S5	0.27	0.15	0.45	0.39	0.52	0.32	-0.40	0.62		
S6	0.30	0.21	0.39	0.39	0.33	0.29	-0.27	0.42	0.46	

Findings

From the above heatmap, most of the features have a correlation between –1 and 1. The features with a correlation closer to 0 have no linear correlation. Those with a correlation closer to 1 have are more closely correlated . For instance S1 and S2 have the highest correlation of 0.9 which means they are highly correlated and as one increases the other increases too.

3.2. What is collinearity?

Collinearity is where two or more features of a regression model are closely linearly correlated.

What effect does collinearity amongst predictor variables have on their estimated coefficient value?

When 2 or more features of a model are closely correlated their estimated correlation coefficient is not unique. This makes it difficult to make inference of the model. This can also cause the R square value to increase. A large R square value increases the variance. To avoid this problem, when doing predictions, features with a R square > 0.8 should be avoided.

3.3. Create a multivariate model using all ten variables and a constant.

I added a column with a constant 1 in the Diabetes data frame. I chose 1 because it doesn't have any effect when multiplied with another value in the data frame. To create a multivariate model, I created a sub data frame as the x variable using all the 10 variables and the dependable variable as the y. I created a model using the statsmodels library OLS and passed in the x and y variables as augments. A prediction on the x variables created the corresponding prediction on y variables.

To calculate mean square error, I used the mean square error method of the sklearn.metrics library and passed the y-predicted and y values. From the model, I generated the p-values and adjusted r square value.

Results:

The Mean Squared Error and the adjusted R2 for model1 are:

Mean squared error is: 2859.69634758675
Adjusted R squared value is: 0.5065592904853231

Inference

The MSE is very large which means the pvalue coefficients will be less likely significant.

When a model has an adjusted R square value < 1, it means the data set had some variations. For this case, the adjusted R square value is 0.51 which indicate that the features used had some irregularity which cannot be explained.

Are all variables significant? Could this be a problem of collinearity?

To check the significance of our variables, we compare the p-values from model1 with the alpha(0.05)

Results

```
Constant    1.016617e-06
AGE         8.670306e-01
SEX         1.041671e-04
BMI         4.296391e-14
BP          1.024278e-06
S1          5.794761e-02
S2          1.603902e-01
S3          6.347233e-01
S4          2.734587e-01
S5          1.555899e-05
S6          3.059895e-01
dtype: float64
```

Inference.

All the variables that have a pvalue of less than 0.05 are significant . **Age, S2,S3,S4 and S6** have a pvalue that is greater than 0.05 hence are insignificant. This could be a problem of collinearity because it waters down the implication of explanatory variables in a model.

3.4. What is the difference between forward selection and backward selection?

There are two methods of stepwise regression: the forward selection and the backward selection. In the forward selection, the model starts with all coefficients at zero and builds the regression successively. The predictive features are added one by one until it gets the best fit. It selects the features that predicts the most on the dependent feature. The best features are added to the model one by one and repeated with the variable that then predicts the most on the dependent feature. This little procedure keeps going on until adding predictors does not add anything to the prediction model anymore. Then the features in the model are returned as the best.

In the backward selection all the predictor features chosen are added into the model on start and removed one by one. The variables that do not significantly predict anything on the dependent feature are removed from the model one by one. The backward method is generally the preferred method because the forward method produces so-called suppressor effects. These suppressor effects occur when predictor features are only significant when another predictor is held constant.

3.5. How does the approach stepwise work in the sense of selecting variables?

First the model starts with no feature. The 10 features initial pvalues are compared against the contant and below is the outcome:

```
const    7.433496e-31
SEX      3.664293e-01
dtype: float64
const    1.751618e-10
BMI      3.466006e-42
dtype: float64
const    0.001760
S1       0.000007
dtype: float64
const    7.126475e-17
S5       8.826459e-39
dtype: float64
const    4.437242e-12
AGE      7.055686e-05
dtype: float64
const    4.301065e-12
S2       2.359848e-04
dtype: float64
const    2.710866e-03
S6       7.580083e-17
dtype: float64
const    4.401020e-04
BP       1.649372e-22
dtype: float64
const    1.892632e-05
S4       2.304253e-21
dtype: float64
const    5.188287e-64
S3       6.162865e-18
dtype: float64

Add  BMI                    with p-value 3.46601e-42
```

Step1: Forward stepwise selection adds features into the model depending with the pvalues. From above we can see that **BMI** has the lowest pvalue and it is added into the model. The remaining models are compared again and the pvalues for the 9 remaining features are as below;

```
const    4.674330e-09
BMI      6.288631e-42
SEX      8.225591e-01
dtype: float64
const    3.792195e-10
BMI      2.319276e-38
S1       7.961267e-02
dtype: float64
const    5.249758e-29
BMI      1.168996e-23
S5       3.039635e-20
dtype: float64
const    2.455363e-11
BMI      1.309246e-39
AGE      3.638563e-02
dtype: float64
const    7.259699e-10
BMI      3.229807e-39
S2       5.757554e-01
dtype: float64
const    3.239627e-14
BMI      3.871059e-31
S6       1.165706e-05
dtype: float64
const    2.020878e-18
BMI      3.047885e-29
BP       1.725601e-09
dtype: float64
const    8.033051e-13
BMI      7.017075e-29
S4       5.888280e-08
dtype: float64
const    4.115392e-01
BMI      1.810748e-31
S3       4.055195e-07
dtype: float64
Add  S5                    with p-value 3.03963e-20
```

step2: S5 has the lowest pvalue among the remaining 9 features. This is added into the model and now we have **BMI and S5** in the model. The remaining 8 features are compared again.

```
const    2.010608e-27
BMI      8.002827e-24
S5       1.170986e-20
SEX      1.444255e-01
dtype: float64
const    6.316956e-28
```

BMI 2.867675e-24
S5 2.026974e-21
S1 3.333891e-03
dtype: float64
const 1.115576e-28
BMI 1.946870e-23
S5 2.974533e-19
AGE 8.157956e-01
dtype: float64
const 1.922565e-28
BMI 3.127332e-24
S5 9.208421e-21
S2 9.037458e-02
dtype: float64
const 1.488582e-27
BMI 1.838220e-21
S5 1.903978e-16
S6 1.467762e-01
dtype: float64
const 1.158977e-32
BMI 5.551045e-19
S5 5.557158e-16
BP 3.742620e-05
dtype: float64
const 7.972106e-25
BMI 2.884551e-22
S5 6.645447e-14
S4 3.914650e-01
dtype: float64
const 4.498733e-11
BMI 1.623888e-20
S5 2.053029e-16
S3 3.770117e-03
dtype: float64

Add BP with p-value 3.74262e-05

step3: BP has the lowest pvalue among the remaining 8 features. This is added into the model and now we have **BMI, S5 and BP** in the model. The remaining 7 features are compared again.

const 9.902446e-32
BMI 7.038410e-19
S5 1.380917e-16
BP 6.007440e-06
SEX 1.743245e-02
dtype: float64
const 6.715025e-32

BMI 1.873430e-19
S5 5.030845e-18
BP 1.700049e-05
S1 1.454431e-03
dtype: float64
const 1.915072e-32
BMI 5.540399e-19
S5 4.627944e-16
BP 2.754857e-05
AGE 4.112309e-01
dtype: float64
const 3.179296e-32
BMI 1.262581e-19
S5 1.028220e-16
BP 2.763333e-05
S2 6.225047e-02
dtype: float64
const 1.190385e-30
BMI 4.079354e-18
S5 3.762158e-14
BP 9.389986e-05
S6 5.284324e-01
dtype: float64
const 1.823435e-28
BMI 1.119057e-17
S5 9.518419e-11
BP 3.262679e-05
S4 3.096380e-01
dtype: float64
const 1.175155e-13
BMI 4.873755e-16
S5 1.640773e-12
BP 1.909239e-05
S3 1.851607e-03
dtype: float64

Add S1 with p-value 0.00145443

step4: S1 has the lowest pvalue among the remaining 7 features. This is added into the model and now we have **BMI, S5, BP and S1** in the model. The remaining 6 features are compared again.

const 7.119361e-31
BMI 2.234253e-19
S5 6.931194e-19
BP 1.988410e-06
S1 7.959889e-04
SEX 9.230560e-03
dtype: float64
const 9.115538e-32

```
BMI      2.006583e-19
S5       5.240804e-18
BP       2.025146e-05
S1       1.953247e-03
AGE      6.943615e-01
dtype: float64
const    8.603175e-32
BMI      2.629916e-16
S5       1.004489e-18
BP       1.254786e-05
S1       3.860581e-04
S2       1.467846e-02
dtype: float64
const    1.647864e-30
BMI      2.147538e-18
S5       1.824265e-16
BP       5.806463e-05
S1       1.091835e-03
S6       3.299199e-01
dtype: float64
const    2.494710e-25
BMI      2.340084e-17
S5       4.189496e-13
BP       9.511717e-06
S1       1.753044e-04
S4       2.539962e-02
dtype: float64
const    1.327705e-14
BMI      9.467696e-17
S5       5.897037e-13
BP       1.252640e-05
S1       2.303291e-02
S3       2.976717e-02
dtype: float64

Add  SEX                 with p-value 0.00923056
```

step5: SEX has the lowest pvalue among the remaining 6 features. This is added into the model and now we have **BMI, S5, BP ,S1 and SEX** in the model. The remaining 5 features are compared again.

```
const    8.683421e-31
BMI      2.438120e-19
S5       8.285996e-19
BP       3.142438e-06
S1       9.750504e-04
SEX      1.005930e-02
AGE      8.983645e-01
dtype: float64
```

```
const    2.750430e-30
BMI      6.687185e-15
S5       1.938635e-21
BP       2.786882e-07
S1       3.122573e-06
SEX      1.758474e-04
S2       2.723024e-04
dtype: float64
const    3.778883e-30
BMI      4.271883e-18
S5       3.835141e-17
BP       8.143499e-06
S1       4.980919e-04
SEX      5.930660e-03
S6       1.851017e-01
dtype: float64
const    2.096840e-21
BMI      3.110875e-16
S5       1.166289e-12
BP       1.811919e-07
S1       9.556135e-06
SEX      2.102307e-04
S4       5.434864e-04
dtype: float64
const    4.582918e-10
BMI      2.431181e-15
S5       2.128982e-12
BP       2.760103e-07
S1       4.361403e-02
SEX      2.023114e-04
S3       6.054228e-04
dtype: float64

Add  S2                    with p-value 0.000272302
```

step6: S2 has the lowest pvalue among the remaining 5 features. This is added into the model and now we have **BMI, S5, BP ,S1,SEX and S2** in the model. The remaining4 features are compared again.

```
const    3.065167e-29
BMI      4.083348e-14
S5       1.202844e-19
BP       1.149518e-06
S1       3.415822e-06
SEX      1.240442e-04
S2       3.705089e-04
S6       2.721107e-01
dtype: float64
```

```
const    1.857504e-06
BMI      7.932792e-15
S5       1.319999e-05
BP       2.702627e-07
S1       1.087649e-01
SEX      1.679425e-04
S2       2.025559e-01
S3       7.136963e-01
dtype: float64
const    3.295026e-30
BMI      7.170514e-15
S5       2.309666e-21
BP       5.082306e-07
S1       3.480635e-06
SEX      1.980671e-04
S2       2.786322e-04
AGE      9.514711e-01
dtype: float64
const    3.110632e-21
BMI      7.248161e-15
S5       6.398602e-09
BP       1.817887e-07
S1       2.810070e-03
SEX      1.066132e-04
S2       1.104828e-01
S4       2.619190e-01
dtype: float64

['BMI', 'S5', 'BP', 'S1', 'SEX', 'S2']
```

The model stops adding other features at step 6 because all the remaining features have a pvalue that is greater than 0.05.

The model selects the following features:

['BMI', 'S5', 'BP', 'S1', 'SEX', 'S2']

The selected features pvalues is as below:

```
forward selection useful features are:
Constant    2.750430e-30
SEX         1.758474e-04
BMI         6.687185e-15
BP          2.786882e-07
S1          3.122573e-06
S2          2.723024e-04
S5          1.938635e-21
dtype: float64
```

What is the MSE and R2 value for this new model?

```
MSE value for useful forward regression variables: 2876.683251787016
R_square value is: 0.5148837959256445
```

Findings

An R square value that is less than 1 indicate that there is some irregularity in the data set which cannot be reported for. Our model gives the Rsquare is 0.51 which implies that 50% of the data irregularity cannot be clarified by the model.

The MSE value is big which means that the

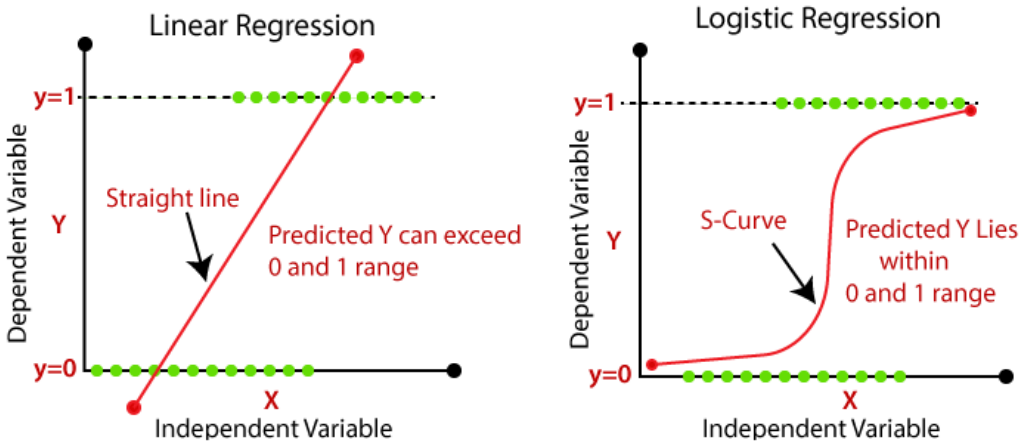
Question 4

4.1 What is the difference between logistic regression and linear regression?

Linear Regression is a commonly used supervised Machine Learning algorithm that predicts continuous values and used to solve regression problems. Linear Regression assumes that there is a linear relationship present between dependent and independent variables. It fits a line that explains two or more variables, the slope is a straight line showing the relation of the two variables as shown in the figure below[7]

While

Logistic Regression is another supervised Machine Learning algorithm that seeks to solve classification problems. It predicts categorical dependent variables using the independent variable y. the result of this model is either 0 or 1 with an s curve graph.The figures below illustrates the difference[7].



4.2 Load in the titanic dataset and calculate the probability of survival for a passenger on the titanic.

I loaded the data set into my working environment and check the count of people in the whole data set using count() method. I noticed some columns had some null values.

To get the survival probability of the passengers, I passed the count() method on the sex column to get the total number of people on the titanic. I also counted the number of survivors by passing the count () method on the survivors column. To compute the probability, I divided the number of survivors by the total count of passengers.

Results:4.2

```
The probability of a passenger surviving is 0.382
```

4.3 Provide a table giving survival probabilities broken down by passenger class, gender, and age.

I used the pd. pivot table() method from pandas library to create a tables for the number of survivors.

For passengers who survival or died by gender, I counted the number of tickets bought using survivor column as my index and categorized by gender tickets bought gender wise and saved in a pivot table

For the passengers who survived or died class wise, I counted the number of tickets using survivor column as my index categorized by the passenger class column and saved in a pivot table.

The ages in the data set have a wide distribution so, I created bins and labels to form categories for the ages. This was stored in a new column agegroup on the Titanic data frame. All ages were put in a group corresponding to the set bins. For the survival by age I counted the number of survivors and deaths using survival column as my index categorized by age group and saved in a pivot table.

Results

```
Survival by gender
sex       female  male
survived
0            127   682
1            339   161

Survival by class
pclass      1    2    3
survived
0         123  158  528
1         200  119  181

Survival by age
AgeGroup  Infant  Toddler  Kid  Teen  Adult
survived
0             18       14   16   141    430
1             33       17   11    84    282
```

The calculate the probability of survival according to gender sex and age, I fetched the index of the corresponding survival value from the sub data frame and divided by the corresponding column sum to GET THE PROBABILITY OF SURVIVAL. For instance to calculate the number of survivors in Class 1:

Number of survivors is 200

Number of deaths = 123

Total number of passengers in class1 : 123+200 = 323

To Get probability of survival used indexing to fetch the number of survivors and divide by the sum of passengers in class1(surviving + dead)

200/323 = 0.619295 .

The table below summarizes all the probabilities of surviving depending on gender, class and age.

survival probability

Female	Male	Clas1	Clas2	Clas3	Infant	Toddler	Kid	Teen	Adult
0.727468	0.190985	0.619195	0.429603	0.255289	0.647059	0.548387	0.407407	0.373333	0.396067

Findings

From the above table we can infer that :

i. Females had a higher probability of survival than the males.
ii. Passenger who rode in class 1 had a higher probability of survival than those in class 3.
iii. Infants had the highest probability of survival as compared to teens who the lowest probability of survival.

4.4 Build a logistic regression model for survival rates based on passenger class, sex, and age.

To be able to do a construct a regression model, I had to deal with the nan values in the age column. I created a function that fetches all the nan values and replaces them with the mean age which was

computed by passing the mean() method on the age column of the Titanic dataframe. After calling the function, all the ages that were nan values were replaced by the mean age.

The sex column values are in strings. Inorder to calculate a regression using the data set I converted the data into numerical values. I first created a two columns, female and male, using get_dummies function from pandas. I passed in the sex column of the Titanic dataset as my variable which generated a sub data frame with female being represented by a zero and male represented by one. I then dropped the first column because only one column is enough to represent both men and women.i finally used the concat ()method to join or add the sub data frame to the main data frame Titanic.

To build the logistic regression line, I used the LogisticRegression method from sklearn.linear_model. I created x and y variables using the passenger class, sex and age as the explanatory variables(x variables) and survived as the dependent variable(y variable).

I ran a prediction on the x variable and got the corresponding y predicted values.

What are the parameter estimates and are these parameters statistically significant?

Results: parameter estimates:

The p values for this model are:

[[-1.10976475 -2.43420108 -0.03341127]]

If a pvalue of a feature is > than 0.05, we infer that the feature is not statistically significant. In our case, all the pvalue are all less than alpha which means that passenger class, sex and age are all statistically significant features.

4.5 What is the performance of the model, measured by classification accuracy (number of correct classifications divided by total number of classifications) based on confusion matrix?

To measure the classification accuracy, I generated a confusion matrix using confusion matrix () method from sklearn.metrics library. The results are as below;

Confusion matrix.

```
array([[685, 124],
       [155, 345]], dtype=int64)
```

To calculate the classification accuracy, we add the number of correct classification and divide by the total number of classifications as below:

True positive result prediction is 685

True negative result prediction is 345

False positive result prediction is 124

False negative result prediction is 155

((685+345)/(685+345+124+155)) *100

Result

Our model is 78.686 % accurate.

Reference

1. Heatmap - https://seaborn.pydata.org/generated/seaborn.heatmap.html
2. A. Worster, J. Fan, and A. Ismaila, "Understanding linear and logistic regression analyses," *Canadian Journal of Emergency Medicine*, vol. 9, no. 2, pp. 111–113, 2007.
3. https://towardsdatascience.com/simplicity-vs-complexity-in-machine-learning-finding-the-right-balance-c9000d1726fb
4. Atsushi Inoue & Lutz Kilian (2005) In-Sample or Out-of-Sample Tests of Predictability: Which One Should We Use?, Econometric Reviews, 23:4, 371-402, DOI: 10.1081/ETC-200040785
5. T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (Springer, New York, 2011). ….question 2
6. K. Murphy, *Machine Learning: A Probabilistic Perspective* (MIT Press, Cambridge, MA, 2012
7. https://www.javatpoint.com/overfitting-in-machine-learning