Group B : MACHINE LEARNING

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Assignment No: 1

**Title Name:** Predict the price of the Uber ride from a given pickup point to the agreed drop-off location.

**Name**: Aditi Shivani

**Class** : BE **Div**: 1 **Batch**: A

# Roll No: 405A005

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

In [1]:

**import** pandas **as** pd **import** numpy **as** np **import** seaborn **as** sns

**import** matplotlib.pyplot **as** plt

In [2]:

df **=** pd**.**read\_csv("uber.csv")

In [3]:

df**.**head()

df**.**info() *#To get the required information of the dataset*

df**.**columns *#TO get number of columns in the dataset*

df **=** df**.**drop(['Unnamed: 0', 'key'], axis**=** 1) *#To drop unnamed column as it isn't req*

df**.**head()

df**.**shape *#To get the total (Rows,Columns)* df**.**dtypes *#To get the type of each column* df**.**info()

df**.**describe() *#To get statistics of each columns*

<class 'pandas.core.frame.DataFrame'> RangeIndex: 200000 entries, 0 to 199999 Data columns (total 9 columns):

# Column Non-Null Count Dtype

|  |  |  |
| --- | --- | --- |
| 0 Unnamed: 0 | 200000 non-null | int64 |
| 1 key | 200000 non-null | object |
| 2 fare\_amount | 200000 non-null | float64 |
| 3 pickup\_datetime | 200000 non-null | object |
| 4 pickup\_longitude | 200000 non-null | float64 |
| 5 pickup\_latitude | 200000 non-null | float64 |
| 6 dropoff\_longitude | 199999 non-null | float64 |
| 7 dropoff\_latitude | 199999 non-null | float64 |
| 8 passenger\_count | 200000 non-null | int64 |

dtypes: float64(5), int64(2), object(2) memory usage: 13.7+ MB

<class 'pandas.core.frame.DataFrame'> RangeIndex: 200000 entries, 0 to 199999 Data columns (total 7 columns):

# Column Non-Null Count Dtype

|  |  |  |
| --- | --- | --- |
| 0 fare\_amount | 200000 non-null | float64 |
| 1 pickup\_datetime | 200000 non-null | object |
| 2 pickup\_longitude | 200000 non-null | float64 |
| 3 pickup\_latitude | 200000 non-null | float64 |
| 4 dropoff\_longitude | 199999 non-null | float64 |
| 5 dropoff\_latitude | 199999 non-null | float64 |
| 6 passenger\_count | 200000 non-null | int64 |

dtypes: float64(5), int64(1), object(1) memory usage: 10.7+ MB

Out[3]: **fare\_amount pickup\_longitude pickup\_latitude dropoff\_longitude dropoff\_latitude passen**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **count** | 200000.000000 | 200000.000000 | 200000.000000 | 199999.000000 | 199999.000000 | 2000 |
| **mean** | 11.359955 | -72.527638 | 39.935885 | -72.525292 | 39.923890 |  |
| **std** | 9.901776 | 11.437787 | 7.720539 | 13.117408 | 6.794829 |  |
| **min** | -52.000000 | -1340.648410 | -74.015515 | -3356.666300 | -881.985513 |  |
| **25%** | 6.000000 | -73.992065 | 40.734796 | -73.991407 | 40.733823 |  |
| **50%** | 8.500000 | -73.981823 | 40.752592 | -73.980093 | 40.753042 |  |
| **75%** | 12.500000 | -73.967154 | 40.767158 | -73.963658 | 40.768001 |  |

**fare\_amount pickup\_longitude pickup\_latitude dropoff\_longitude dropoff\_latitude passen max** 499.000000 57.418457 1644.421482 1153.572603 872.697628 2

In [4]:

df**.**isnull()**.**sum() df['dropoff\_latitude']**.**fillna(value**=**df['dropoff\_latitude']**.**mean(),inplace **= True**) df['dropoff\_longitude']**.**fillna(value**=**df['dropoff\_longitude']**.**median(),inplace **= True** df**.**isnull()**.**sum()

df**.**dtypes

Out[4]:

In [5]:

fare\_amount float64

pickup\_datetime object pickup\_longitude float64 pickup\_latitude float64 dropoff\_longitude float64 dropoff\_latitude float64 passenger\_count int64 dtype: object

df**.**pickup\_datetime **=** pd**.**to\_datetime(df**.**pickup\_datetime, errors**=**'coerce') df**.**dtypes

Out[5]: fare\_amount float64 pickup\_datetime datetime64[ns, UTC] pickup\_longitude float64

pickup\_latitude float64

dropoff\_longitude float64

dropoff\_latitude float64

passenger\_count int64 dtype: object

In [6]:

df**=** df**.**assign(hour **=** df**.**pickup\_datetime**.**dt**.**hour, day**=** df**.**pickup\_datetime**.**dt**.**day,

month **=** df**.**pickup\_datetime**.**dt**.**month,

year **=** df**.**pickup\_datetime**.**dt**.**year,

dayofweek **=** df**.**pickup\_datetime**.**dt**.**dayofweek) df**.**head()

Out[6]:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **fare\_amount pickup\_datetime pickup\_longitude pickup\_latitude dropoff\_longitude dropoff\_latit** | | | | | | |
| **0** | 7.5 | 2015-05-07  19:52:06+00:00 | -73.999817 | 40.738354 | -73.999512 | 40.723 |
| **1** | 7.7 | 2009-07-17  20: | -73.994355 | 40.728225 | -73.994710 | 40.750 |
| **2** | 12.9 | 2009-08-24  21:45:00+00:00 | -74.005043 | 40.740770 | -73.962565 | 40.772 |
| **3** | 5.3 | 2009-06-26  08:22:21+00:00 | -73.976124 | 40.790844 | -73.965316 | 40.803 |
| **4** | 16.0 | 2014-08-28  17:47:00+00:00 | -73.925023 | 40.744085 | -73.973082 | 40.761 |

In [7]:

df **=** df**.**drop('pickup\_datetime',axis**=**1) df**.**head()

df**.**dtypes

Out[7]:

In [8]:

fare\_amount float64 pickup\_longitude float64 pickup\_latitude float64 dropoff\_longitude float64 dropoff\_latitude float64 passenger\_count int64

hour int64

day int64

month int64

year int64

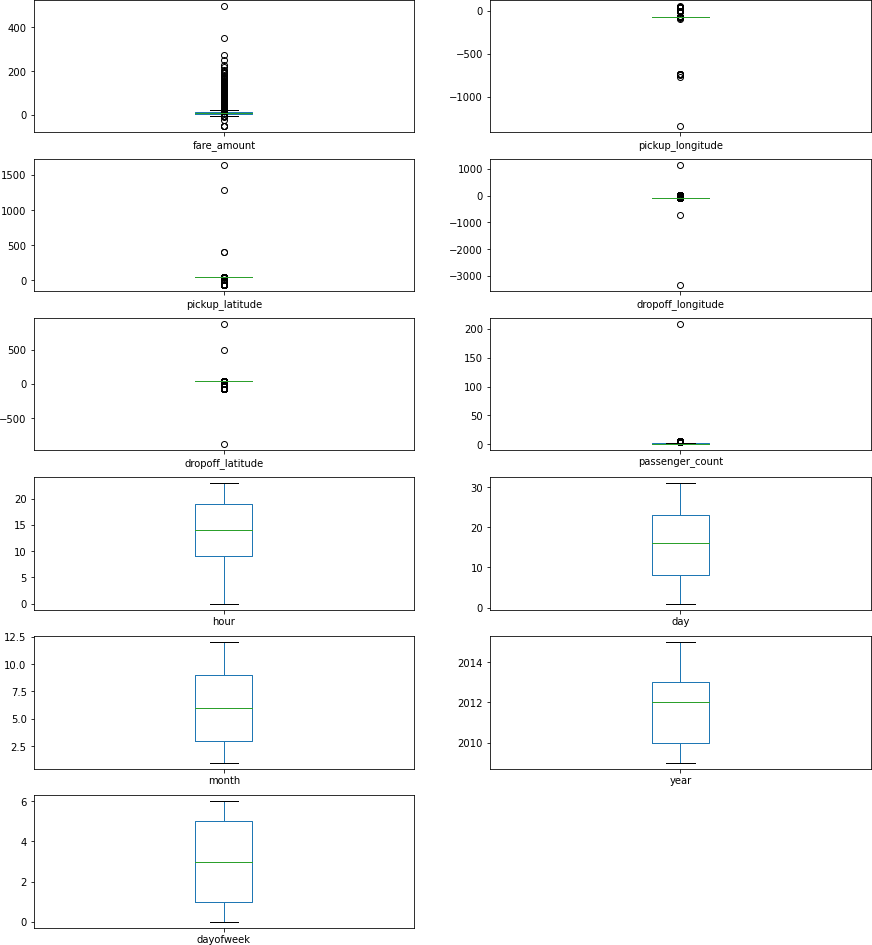
dayofweek int64 dtype: object

df**.**plot(kind **=** "box",subplots **= True**,layout **=** (7,2),figsize**=**(15,20))

Out[8]: fare\_amount AxesSubplot(0.125,0.787927;0.352273x0.0920732) pickup\_longitude AxesSubplot(0.547727,0.787927;0.352273x0.0920732) pickup\_latitude AxesSubplot(0.125,0.677439;0.352273x0.0920732) dropoff\_longitude AxesSubplot(0.547727,0.677439;0.352273x0.0920732) dropoff\_latitude AxesSubplot(0.125,0.566951;0.352273x0.0920732) passenger\_count AxesSubplot(0.547727,0.566951;0.352273x0.0920732) hour AxesSubplot(0.125,0.456463;0.352273x0.0920732)

day AxesSubplot(0.547727,0.456463;0.352273x0.0920732) month AxesSubplot(0.125,0.345976;0.352273x0.0920732) year AxesSubplot(0.547727,0.345976;0.352273x0.0920732)

dayofweek AxesSubplot(0.125,0.235488;0.352273x0.0920732) dtype: object



In [9]:

**def** remove\_outlier(df1 , col):

Q1 **=** df1[col]**.**quantile(0.25) Q3 **=** df1[col]**.**quantile(0.75)

IQR **=** Q3 **-** Q1

lower\_whisker **=** Q1**-**1.5**\***IQR upper\_whisker **=** Q3**+**1.5**\***IQR

df[col] **=** np**.**clip(df1[col] , lower\_whisker , upper\_whisker)

**return** df1

In [12]:

**def** treat\_outliers\_all(df1 , col\_list):

**for** c **in** col\_list:

df1 **=** remove\_outlier(df , c)

**return** df1

df **=** treat\_outliers\_all(df , df**.**iloc[: , 0::])

df**.**plot(kind **=** "box",subplots **= True**,layout **=** (7,2),figsize**=**(15,20)) *#Boxplot shows*

Out[12]: fare\_amount AxesSubplot(0.125,0.787927;0.352273x0.0920732) pickup\_longitude AxesSubplot(0.547727,0.787927;0.352273x0.0920732) pickup\_latitude AxesSubplot(0.125,0.677439;0.352273x0.0920732) dropoff\_longitude AxesSubplot(0.547727,0.677439;0.352273x0.0920732)

In [14]:

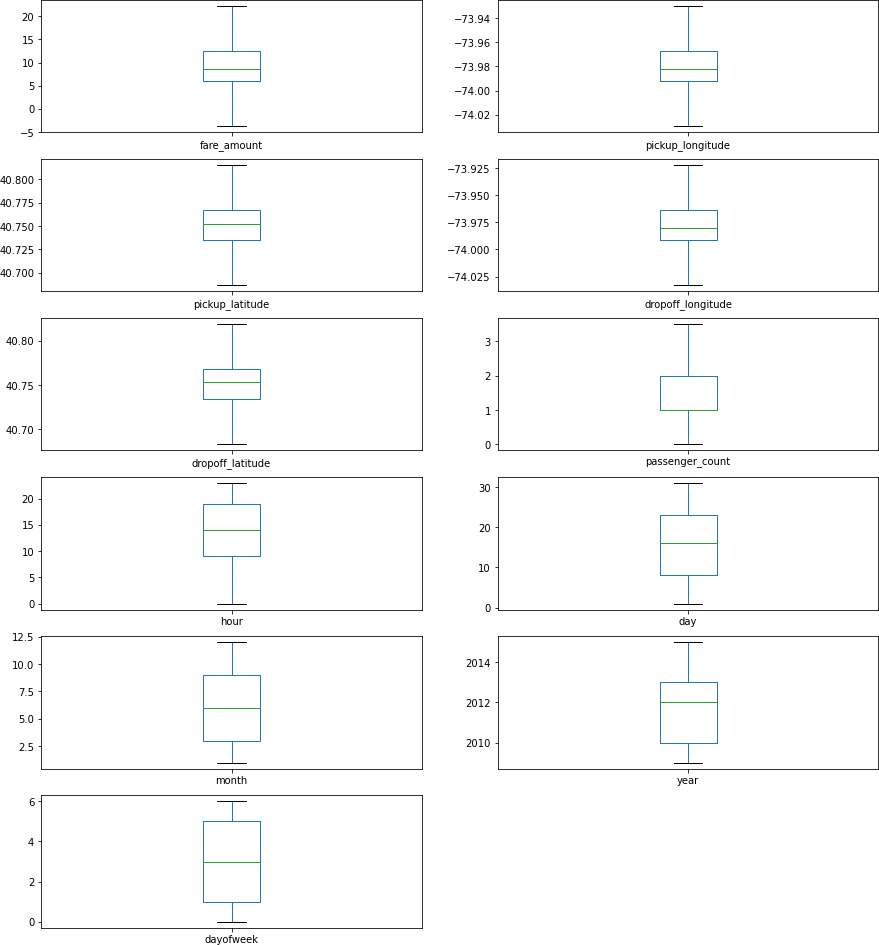
In [22]:

dropoff\_latitude AxesSubplot(0.125,0.566951;0.352273x0.0920732) passenger\_count AxesSubplot(0.547727,0.566951;0.352273x0.0920732) hour AxesSubplot(0.125,0.456463;0.352273x0.0920732)

day AxesSubplot(0.547727,0.456463;0.352273x0.0920732) month AxesSubplot(0.125,0.345976;0.352273x0.0920732)

year AxesSubplot(0.547727,0.345976;0.352273x0.0920732)

dayofweek AxesSubplot(0.125,0.235488;0.352273x0.0920732) dtype: object



pip install haversine

Collecting haversine

Downloading haversine-2.7.0-py2.py3-none-any.whl (6.9 kB) Installing collected packages: haversine

Successfully installed haversine-2.7.0

Note: you may need to restart the kernel to use updated packages.

**import** haversine **as** hs *#Calculate the distance using Haversine to calculate the dist*

travel\_dist **=** []

**for** pos **in** range(len(df['pickup\_longitude'])):

long1,lati1,long2,lati2 **=** [df['pickup\_longitude'][pos],df['pickup\_latitude'][pos loc1**=**(lati1,long1)

loc2**=**(lati2,long2)

c **=** hs**.**haversine(loc1,loc2) travel\_dist**.**append(c)

print(travel\_dist) df['dist\_travel\_km'] **=** travel\_dist df**.**head()

IOPub data rate exceeded.

The notebook server will temporarily stop sending output to the client in order to avoid crashing it.

To change this limit, set the config variable

`--NotebookApp.iopub\_data\_rate\_limit`.

Current values: NotebookApp.iopub\_data\_rate\_limit=1000000.0 (bytes/sec) NotebookApp.rate\_limit\_window=3.0 (secs)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Out[22]: | **fare\_amount** | **pickup\_longitude** | **pickup\_latitude** | **dropoff\_longitude** | **dropoff\_latitude passenger\_co** |
| **0** 7.5 | | -73.999817 | 40.738354 | -73.999512 | 40.723217 |
| **1** 7.7 | | -73.994355 | 40.728225 | -73.994710 | 40.750325 |
| **2** 12.9 | | -74.005043 | 40.740770 | -73.962565 | 40.772647 |
| **3** 5.3 | | -73.976124 | 40.790844 | -73.965316 | 40.803349 |
| **4** 16.0 | | -73.929786 | 40.744085 | -73.973082 | 40.761247 |

In [23]:

df**=** df**.**loc[(df**.**dist\_travel\_km **>=** 1) **|** (df**.**dist\_travel\_km **<=** 130)] print("Remaining observastions in the dataset:", df**.**shape)

Remaining observastions in the dataset: (200000, 12)

In [26]:

incorrect\_coordinates **=** df**.**loc[(df**.**pickup\_latitude **>** 90) **|**(df**.**pickup\_latitude **< -**90) (df**.**dropoff\_latitude **>** 90) **|**(df**.**dropoff\_latitude **< -**90) **|**

(df**.**pickup\_longitude **>** 180) **|**(df**.**pickup\_longitude **< -**180) **|**

(df**.**dropoff\_longitude **>** 90) **|**(df**.**dropoff\_longitude **< -**90)

]

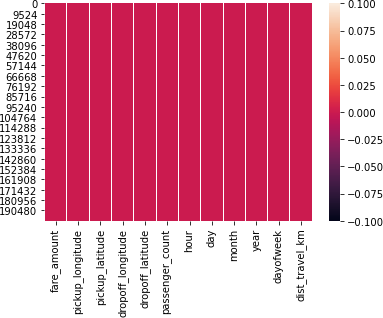
df**.**drop(incorrect\_coordinates, inplace **= True**, errors **=** 'ignore') df**.**head()

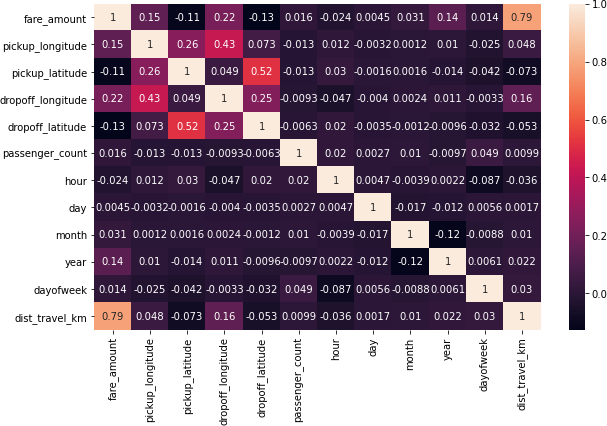
df**.**isnull()**.**sum()

sns**.**heatmap(df**.**isnull()) *#Free for null values* corr **=** df**.**corr() *#Function to find the correlation* corr

fig,axis **=** plt**.**subplots(figsize **=** (10,6)) sns**.**heatmap(df**.**corr(),annot **= True**)

Out[26]: <AxesSubplot:>





In [28]:

x **=** df[['pickup\_longitude','pickup\_latitude','dropoff\_longitude','dropoff\_latitude', y **=** df['fare\_amount']

In [29]:

**from** sklearn.model\_selection **import** train\_test\_split X\_train,X\_test,y\_train,y\_test **=** train\_test\_split(x,y,test\_size **=** 0.33)

In [31]:

**from** sklearn.linear\_model **import** LinearRegression regression **=** LinearRegression() regression**.**fit(X\_train,y\_train) regression**.**coef\_ *#To find the linear coeeficient*

regression**.**intercept\_ *#To find the linear intercept*

prediction **=** regression**.**predict(X\_test) *#To predict the target values*

print(prediction) y\_test

[17.28050585 11.44946862 13.22284482 ... 15.04497674 18.34524502

9.91445235]

|  |  |  |
| --- | --- | --- |
| Out[31]: | 30406 | 18.50 |
|  | 122525 | 13.00 |
|  | 145989 | 22.25 |
|  | 50071 | 17.50 |
|  | 2065 | 4.50  ... |
|  | 95147 | 4.50 |
|  | 107084 | 14.10 |
|  | 36958 | 11.50 |
|  | 65775 | 14.10 |
|  | 39173 | 8.50 |

In [33]:

Out[33]:

In [ ]:

Name: fare\_amount, Length: 66000, dtype: float64

**from** sklearn.metrics **import** r2\_score r2\_score(y\_test,prediction)

**from** sklearn.metrics **import** mean\_squared\_error MSE **=** mean\_squared\_error(y\_test,prediction) MSE

RMSE **=** np**.**sqrt(MSE) RMSE 3.156187085348032

3.156187085348032

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Assignment No: 2

**Title Name:** Classify the email using the binary classification method. Email Spam detection has two states: a) Normal State – Not Spam, b) Abnormal State – Spam. Use K-Nearest Neighbors and Support Vector Machine for classification. Analyze their performance.

**Name**: Aditi Shivani

**Class** : BE **Div**: 1 **Batch**: A

# Roll No: 405A005

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

In [4

|  |
| --- |
| ]:  **import** pandas **as** pd  **import** numpy **as** np  **import** seaborn **as** sns  **import** matplotlib.pyplot **as** plt  **%matplotlib** inline  **import** warnings warnings**.**filterwarnings('ignore')  **from** sklearn.model\_selection **import** train\_test\_split  **from** sklearn.svm **import** SVC **from** sklearn **import** metrics df**=**pd**.**read\_csv('emails.csv') df**.**head()  df**.**columns df**.**isnull()**.**sum() df**.**dropna(inplace **= True**)  df**.**drop(['Email No.'],axis**=**1,inplace**=True**) X **=** df**.**drop(['Prediction'],axis **=** 1)  y **=** df['Prediction']  **from** sklearn.preprocessing **import** scale X **=** scale(X)  *# split into train and test*  X\_train, X\_test, y\_train, y\_test **=** train\_test\_split(X, y, test\_size **=** 0.3, random\_st |
|  |
| ]:  **from** sklearn.neighbors **import** KNeighborsClassifier  knn **=** KNeighborsClassifier(n\_neighbors**=**7) knn**.**fit(X\_train, y\_train)  y\_pred **=** knn**.**predict(X\_test) print("Prediction",y\_pred)  print("KNN accuracy = ",metrics**.**accuracy\_score(y\_test,y\_pred)) print("Confusion matrix",metrics**.**confusion\_matrix(y\_test,y\_pred)) |

In [5

In [6]:

Prediction [0 0 1 ... 1 1 1]

KNN accuracy = 0.8009020618556701

Confusion matrix [[804 293]

[ 16 439]]

*# cost C = 1* model **=** SVC(C **=** 1) *# fit*

model**.**fit(X\_train, y\_train)

*# predict*

y\_pred **=** model**.**predict(X\_test) metrics**.**confusion\_matrix(y\_true**=**y\_test, y\_pred**=**y\_pred) print("SVM accuracy = ",metrics**.**accuracy\_score(y\_test,y\_pred))

SVM accuracy = 0.9381443298969072

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Assignment No: 3

**Title Name:** Given a bank customer, build a neural network-based classifier that can determine whether they will leave or not in the next 6 months

**Name**: Aditi Shivani

**Class** : BE **Div**: 1 **Batch**: A

# Roll No: 405A005

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

In [1]:

|  |
| --- |
| **import** pandas **as** pd **import** numpy **as** np **import** seaborn **as** sns  **import** matplotlib.pyplot **as** plt *#Importing the libraries*  df **=** pd**.**read\_csv("Churn\_Modelling.csv") |

In [2]:

|  |
| --- |
| df**.**head() df**.**shape df**.**describe() df**.**isnull() df**.**isnull()**.**sum() df**.**info() df**.**dtypes df**.**columns  df **=** df**.**drop(['RowNumber', 'Surname', 'CustomerId'], axis**=** 1) *#Dropping the unnecess*  df**.**head() |

<class 'pandas.core.frame.DataFrame'> RangeIndex: 10000 entries, 0 to 9999 Data columns (total 14 columns):

# Column Non-Null Count Dtype

1. RowNumber 10000 non-null int64
2. CustomerId 10000 non-null int64
3. Surname 10000 non-null object
4. CreditScore 10000 non-null int64
5. Geography 10000 non-null object
6. Gender 10000 non-null object
7. Age 10000 non-null int64
8. Tenure 10000 non-null int64
9. Balance 10000 non-null float64
10. NumOfProducts 10000 non-null int64
11. HasCrCard 10000 non-null int64
12. IsActiveMember 10000 non-null int64
13. EstimatedSalary 10000 non-null float64
14. Exited 10000 non-null int64 dtypes: float64(2), int64(9), object(3) memory usage: 1.1+ MB

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Out[2]: | **CreditScore** | **Geography** | **Gender** | **Age** | **Tenure** | **Balance** | **NumOfProducts** | **HasCrCard IsActiveM** |
| **0** | 619 | France | Female | 42 | 2 | 0.00 | 1 | 1 |
| **1** | 608 | Spain | Female | 41 | 1 | 83807.86 | 1 | 0 |
| **2** | 502 | France | Female | 42 | 8 | 159660.80 | 3 | 1 |
| **3** | 699 | France | Female | 39 | 1 | 0.00 | 2 | 0 |
| **4** | 850 | Spain | Female | 43 | 2 | 125510.82 | 1 | 1 |

In [3]:

**def** visualization(x, y, xlabel): plt**.**figure(figsize**=**(10,5))

plt**.**hist([x, y], color**=**['red', 'green'], label **=** ['exit', 'not\_exit']) plt**.**xlabel(xlabel,fontsize**=**20)

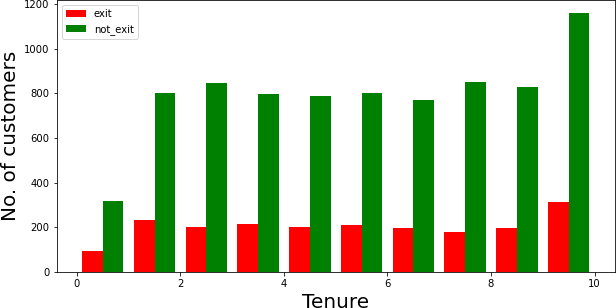
plt**.**ylabel("No. of customers", fontsize**=**20)

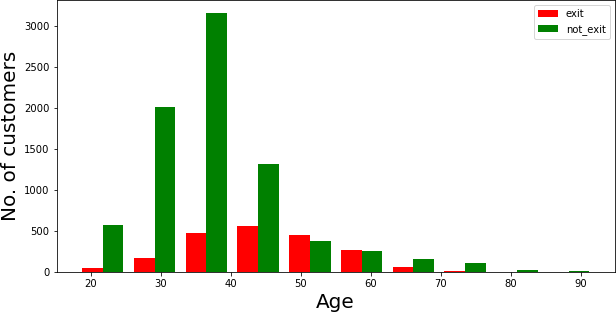
plt**.**legend()

In [4]:

df\_churn\_exited **=** df[df['Exited']**==**1]['Tenure'] df\_churn\_not\_exited **=** df[df['Exited']**==**0]['Tenure']

visualization(df\_churn\_exited, df\_churn\_not\_exited, "Tenure") df\_churn\_exited2 **=** df[df['Exited']**==**1]['Age'] df\_churn\_not\_exited2 **=** df[df['Exited']**==**0]['Age'] visualization(df\_churn\_exited2, df\_churn\_not\_exited2, "Age")





In [6]:

X **=** df[['CreditScore','Gender','Age','Tenure','Balance','NumOfProducts','HasCrCard', states **=** pd**.**get\_dummies(df['Geography'],drop\_first **= True**)

gender **=** pd**.**get\_dummies(df['Gender'],drop\_first **= True**) df **=** pd**.**concat([df,gender,states], axis **=** 1)

In [8]:

df**.**head()

X **=** df[['CreditScore','Age','Tenure','Balance','NumOfProducts','HasCrCard','IsActive y **=** df['Exited']

**from** sklearn.model\_selection **import** train\_test\_split X\_train,X\_test,y\_train,y\_test **=** train\_test\_split(X,y,test\_size **=** 0.30)

In [9]:

**from** sklearn.preprocessing **import** StandardScaler sc **=** StandardScaler()

X\_train **=** sc**.**fit\_transform(X\_train) X\_test **=** sc**.**transform(X\_test)

X\_train X\_test

|  |  |  |
| --- | --- | --- |
| Out[9]: | array([[ 0.08909172, | 2.03556129, -1.04195601, ..., 0.90636285, |
|  | -0.57581067, | 1.7581737 ], |
|  | [-0.6935785 , | -0.3592006 , 0.33616247, ..., 0.90636285, |
|  | -0.57581067, | -0.56877202], |
|  | [ 1.7066102 , | 3.18504699, -1.04195601, ..., -1.10331088, |
|  | -0.57581067, | 1.7581737 ], |
|  | ..., |  |
|  | [ 0.07865612, | -0.93394345, -0.35289677, ..., 0.90636285, |
|  | -0.57581067, | -0.56877202], |
|  | [-0.46399524, | -0.3592006 , -1.73101525, ..., -1.10331088, |
|  | -0.57581067, | -0.56877202], |
|  | [ 1.59181856, | -0.55078155, 0.33616247, ..., 0.90636285, |
|  | -0.57581067, | -0.56877202]]) |

In [10]:

**import** keras*#Can use Tenserflow as well but won't be able to understand the errors i*

**from** keras.models **import** Sequential *#To create sequential neural network*

**from** keras.layers **import** Dense *#To create hidden layers*

classifier **=** Sequential()

*#To add the layers*

*#Dense helps to contruct the neurons #Input Dimension means we have 11 features # Units is to create the hidden layers*

In [11]:

classifier**.**add(Dense(activation **=** "relu",input\_dim **=** 11,units **=** 6,kernel\_initializer classifier**.**add(Dense(activation **=** "relu",units **=** 6,kernel\_initializer **=** "uniform")) classifier**.**add(Dense(activation **=** "sigmoid",units **=** 1,kernel\_initializer **=** "uniform" classifier**.**compile(optimizer**=**"adam",loss **=** 'binary\_crossentropy',metrics **=** ['accurac classifier**.**summary() *#3 layers created. 6 neurons in 1st,6neurons in 2nd layer and 1* classifier**.**fit(X\_train,y\_train,batch\_size**=**10,epochs**=**50) *#Fitting the ANN to training* y\_pred **=**classifier**.**predict(X\_test)

y\_pred **=** (y\_pred **>** 0.5) *#Predicting the result*

**from** sklearn.metrics **import** confusion\_matrix,accuracy\_score,classification\_report cm **=** confusion\_matrix(y\_test,y\_pred)

cm

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

plt**.**figure(figsize **=** (10,7))

sns**.**heatmap(cm,annot **= True**) plt**.**xlabel('Predicted') plt**.**ylabel('Truth') print(classification\_report(y\_test,y\_pred))

Model: "sequential"

Layer (type) Output Shape Param #

=================================================================

|  |  |  |  |
| --- | --- | --- | --- |
| dense (Dense) | (None, | 6) | 72 |
| dense\_1 (Dense) | (None, | 6) | 42 |
| dense\_2 (Dense) | (None, | 1) | 7 |

=================================================================

Total params: 121

Trainable params: 121

Non-trainable params: 0

Epoch 1/50

700/700 [==============================] - 1s 675us/step - loss: 0.4841 - accuracy:

0.7970

Epoch 2/50

10/26/22, 5:24 PM ml\_3

700/700 [==============================] - 5s 711us/step - loss: 0.4220 - accuracy:

0.7970

Epoch 3/50

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 700/700 [==============================] | - 0s | 715us/step | - loss: | 0.4165 | - accuracy: |
| 0.7990  Epoch 4/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 625us/step | - loss: | 0.4129 | - accuracy: |
| 0.8277 |  |  |  |  |  |
| Epoch 5/50  700/700 [==============================] | - 0s | 634us/step | - loss: | 0.4109 | - accuracy: |
| 0.8280 |  |  |  |  |  |
| Epoch 6/50 |  |  |  |  |  |
| 700/700 [==============================]  0.8304 | - 1s | 736us/step | - loss: | 0.4093 | - accuracy: |
| Epoch 7/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 650us/step | - loss: | 0.4078 | - accuracy: |
| 0.8321  Epoch 8/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 653us/step | - loss: | 0.4065 | - accuracy: |
| 0.8319 |  |  |  |  |  |
| Epoch 9/50  700/700 [==============================] | - 0s | 640us/step | - loss: | 0.4053 | - accuracy: |
| 0.8350 |  |  |  |  |  |
| Epoch 10/50 |  |  |  |  |  |
| 700/700 [==============================]  0.8351 | - 0s | 621us/step | - loss: | 0.4044 | - accuracy: |
| Epoch 11/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 643us/step | - loss: | 0.4035 | - accuracy: |
| 0.8360  Epoch 12/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 700us/step | - loss: | 0.4024 | - accuracy: |
| 0.8360 |  |  |  |  |  |
| Epoch 13/50  700/700 [==============================] | - 0s | 648us/step | - loss: | 0.4016 | - accuracy: |
| 0.8349 |  |  |  |  |  |
| Epoch 14/50 |  |  |  |  |  |
| 700/700 [==============================]  0.8356 | - 0s | 649us/step | - loss: | 0.4010 | - accuracy: |
| Epoch 15/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 712us/step | - loss: | 0.4005 | - accuracy: |
| 0.8361  Epoch 16/50 |  |  |  |  |  |
| 700/700 [==============================] | - 1s | 721us/step | - loss: | 0.3997 | - accuracy: |
| 0.8367 |  |  |  |  |  |
| Epoch 17/50  700/700 [==============================] | - 1s | 716us/step | - loss: | 0.3996 | - accuracy: |
| 0.8364 |  |  |  |  |  |
| Epoch 18/50 |  |  |  |  |  |
| 700/700 [==============================]  0.8371 | - 0s | 703us/step | - loss: | 0.3988 | - accuracy: |
| Epoch 19/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 695us/step | - loss: | 0.3985 | - accuracy: |
| 0.8360  Epoch 20/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 650us/step | - loss: | 0.3973 | - accuracy: |
| 0.8364 |  |  |  |  |  |
| Epoch 21/50  700/700 [==============================] | - 0s | 641us/step | - loss: | 0.3978 | - accuracy: |
| 0.8369 |  |  |  |  |  |
| Epoch 22/50 |  |  |  |  |  |
| 700/700 [==============================]  0.8363 | - 0s | 615us/step | - loss: | 0.3975 | - accuracy: |
| Epoch 23/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 639us/step | - loss: | 0.3962 | - accuracy: |
| 0.8377  Epoch 24/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 649us/step | - loss: | 0.3966 | - accuracy: |
| 0.8359  Epoch 25/50 |  |  |  |  |  |

10/26/22, 5:24 PM ml\_3

700/700 [==============================] - 0s 666us/step - loss: 0.3961 - accuracy:

0.8371

Epoch 26/50

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 700/700 [==============================] | - 0s | 630us/step | - loss: | 0.3959 | - accuracy: |
| 0.8373  Epoch 27/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 666us/step | - loss: | 0.3957 | - accuracy: |
| 0.8369 |  |  |  |  |  |
| Epoch 28/50  700/700 [==============================] | - 0s | 644us/step | - loss: | 0.3956 | - accuracy: |
| 0.8391 |  |  |  |  |  |
| Epoch 29/50 |  |  |  |  |  |
| 700/700 [==============================]  0.8379 | - 0s | 594us/step | - loss: | 0.3956 | - accuracy: |
| Epoch 30/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 668us/step | - loss: | 0.3950 | - accuracy: |
| 0.8371  Epoch 31/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 715us/step | - loss: | 0.3949 | - accuracy: |
| 0.8393 |  |  |  |  |  |
| Epoch 32/50  700/700 [==============================] | - 0s | 705us/step | - loss: | 0.3950 | - accuracy: |
| 0.8376 |  |  |  |  |  |
| Epoch 33/50 |  |  |  |  |  |
| 700/700 [==============================]  0.8386 | - 0s | 652us/step | - loss: | 0.3955 | - accuracy: |
| Epoch 34/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 684us/step | - loss: | 0.3948 | - accuracy: |
| 0.8381  Epoch 35/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 661us/step | - loss: | 0.3944 | - accuracy: |
| 0.8387 |  |  |  |  |  |
| Epoch 36/50  700/700 [==============================] | - 0s | 678us/step | - loss: | 0.3947 | - accuracy: |
| 0.8394 |  |  |  |  |  |
| Epoch 37/50 |  |  |  |  |  |
| 700/700 [==============================]  0.8394 | - 0s | 663us/step | - loss: | 0.3942 | - accuracy: |
| Epoch 38/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 711us/step | - loss: | 0.3939 | - accuracy: |
| 0.8397  Epoch 39/50 |  |  |  |  |  |
| 700/700 [==============================] | - 1s | 996us/step | - loss: | 0.3941 | - accuracy: |
| 0.8383 |  |  |  |  |  |
| Epoch 40/50  700/700 [==============================] | - 1s | 799us/step | - loss: | 0.3939 | - accuracy: |
| 0.8404 |  |  |  |  |  |
| Epoch 41/50 |  |  |  |  |  |
| 700/700 [==============================]  0.8396 | - 1s | 863us/step | - loss: | 0.3938 | - accuracy: |
| Epoch 42/50 |  |  |  |  |  |
| 700/700 [==============================] | - 1s | 739us/step | - loss: | 0.3943 | - accuracy: |
| 0.8377  Epoch 43/50 |  |  |  |  |  |
| 700/700 [==============================] | - 1s | 761us/step | - loss: | 0.3933 | - accuracy: |
| 0.8374 |  |  |  |  |  |
| Epoch 44/50  700/700 [==============================] | - 1s | 919us/step | - loss: | 0.3938 | - accuracy: |
| 0.8389 |  |  |  |  |  |
| Epoch 45/50 |  |  |  |  |  |
| 700/700 [==============================]  0.8387 | - 0s | 630us/step | - loss: | 0.3938 | - accuracy: |
| Epoch 46/50 |  |  |  |  |  |
| 700/700 [==============================] | - 1s | 766us/step | - loss: | 0.3936 | - accuracy: |
| 0.8381  Epoch 47/50 |  |  |  |  |  |
| 700/700 [==============================] | - 0s | 623us/step | - loss: | 0.3935 | - accuracy: |
| 0.8386  Epoch 48/50 |  |  |  |  |  |

10/26/22, 5:24 PM ml\_3

700/700 [==============================] - 0s 614us/step - loss: 0.3937 - accuracy:

0.8380

Epoch 49/50

700/700 [==============================] - 1s 763us/step - loss: 0.3934 - accuracy:

0.8399

Epoch 50/50

700/700 [==============================] - 1s 837us/step - loss: 0.3935 - accuracy:

0.8391

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| precision | | recall | f1-score | support |
| 0 | 0.84 | 0.97 | 0.90 | 2384 |
| 1 | 0.72 | 0.29 | 0.41 | 616 |
| accuracy |  |  | 0.83 | 3000 |
| macro avg | 0.78 | 0.63 | 0.66 | 3000 |
| weighted avg | 0.82 | 0.83 | 0.80 | 3000 |



In [ ]:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Assignment No: 4

**Title Name:** Implement Gradient Descent Algorithm.

**Name**: Aditi Shivani

**Class** : BE **Div**: 1 **Batch**: A

# Roll No: 405A005

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

In [1]:

cur\_x **=** 3 *# The algorithm starts at x=3*

rate **=** 0.01 *# Learning rate*

precision **=** 0.000001 *#This tells us when to stop the algorithm*

previous\_step\_size **=** 1 *#*

max\_iters **=** 10000 *# maximum number of iterations*

iters **=** 0 *#iteration counter*

df **= lambda** x: 2**\***(x**+**5) *#Gradient of our function*

In [2]:

**while** previous\_step\_size **>** precision **and** iters **<** max\_iters: prev\_x **=** cur\_x *#Store current x value in prev\_x*

cur\_x **=** cur\_x **-** rate **\*** df(prev\_x) *#Grad descent* previous\_step\_size **=** abs(cur\_x **-** prev\_x) *#Change in x* iters **=** iters**+**1 *#iteration count*

print("Iteration",iters,"\nX value is",cur\_x) *#Print iterations*

print("The local minimum occurs at", cur\_x)

Iteration 1

X value is 2.84

The local minimum occurs at 2.84 Iteration 2

X value is 2.6832

The local minimum occurs at 2.6832 Iteration 3

X value is 2.529536

The local minimum occurs at 2.529536 Iteration 4

X value is 2.37894528

The local minimum occurs at 2.37894528 Iteration 5

X value is 2.2313663744

The local minimum occurs at 2.2313663744 Iteration 6

X value is 2.0867390469119997

The local minimum occurs at 2.0867390469119997 Iteration 7

X value is 1.9450042659737599

The local minimum occurs at 1.9450042659737599 Iteration 8

X value is 1.8061041806542846

The local minimum occurs at 1.8061041806542846 Iteration 9

X value is 1.669982097041199

The local minimum occurs at 1.669982097041199 Iteration 10

X value is 1.5365824551003748

The local minimum occurs at 1.5365824551003748 Iteration 11

X value is 1.4058508059983674

The local minimum occurs at 1.4058508059983674 Iteration 12

X value is 1.2777337898784

The local minimum occurs at 1.2777337898784 Iteration 13

X value is 1.152179114080832

10/26/22, 5:25 PM ml\_4

Iteration 17

X value is 0.6745741294451669

The local minimum occurs at 0.6745741294451669 Iteration 18

X value is 0.5610826468562635

The local minimum occurs at 0.5610826468562635 Iteration 19

X value is 0.44986099391913825

The local minimum occurs at 0.44986099391913825 Iteration 20

X value is 0.3408637740407555

The local minimum occurs at 0.3408637740407555 Iteration 21

X value is 0.23404649855994042

The local minimum occurs at 0.23404649855994042 Iteration 22

X value is 0.1293655685887416

The local minimum occurs at 0.1293655685887416 Iteration 23

X value is 0.026778257216966764

The local minimum occurs at 0.026778257216966764 Iteration 24

X value is -0.07375730792737258

The local minimum occurs at -0.07375730792737258 Iteration 25

X value is -0.1722821617688251

The local minimum occurs at -0.1722821617688251 Iteration 26

X value is -0.2688365185334486

The local minimum occurs at -0.2688365185334486 Iteration 27

X value is -0.36345978816277963

The local minimum occurs at -0.36345978816277963 Iteration 28

X value is -0.45619059239952403

The local minimum occurs at -0.45619059239952403 Iteration 29

X value is -0.5470667805515336

The local minimum occurs at -0.5470667805515336 Iteration 30

X value is -0.6361254449405029

The local minimum occurs at -0.6361254449405029 Iteration 31

X value is -0.7234029360416929

The local minimum occurs at -0.7234029360416929 Iteration 32

X value is -0.8089348773208591

The local minimum occurs at -0.8089348773208591 Iteration 33

X value is -0.8927561797744419

The local minimum occurs at -0.8927561797744419 Iteration 34

X value is -0.9749010561789531

The local minimum occurs at -0.9749010561789531 Iteration 35

X value is -1.055403035055374

The local minimum occurs at -1.055403035055374 Iteration 36

X value is -1.1342949743542665

The local minimum occurs at -1.1342949743542665 Iteration 37

X value is -1.2116090748671813

The local minimum occurs at -1.2116090748671813 Iteration 38

X value is -1.2873768933698377

The local minimum occurs at -1.2873768933698377 Iteration 39

X value is -1.361629355502441

10/26/22, 5:25 PM ml\_4

Iteration 40

X value is -1.4343967683923922

The local minimum occurs at -1.4343967683923922 Iteration 41

X value is -1.5057088330245443

The local minimum occurs at -1.5057088330245443 Iteration 42

X value is -1.5755946563640535

The local minimum occurs at -1.5755946563640535 Iteration 43

X value is -1.6440827632367725

The local minimum occurs at -1.6440827632367725 Iteration 44

X value is -1.711201107972037

The local minimum occurs at -1.711201107972037 Iteration 45

X value is -1.7769770858125964

The local minimum occurs at -1.7769770858125964 Iteration 46

X value is -1.8414375440963444

The local minimum occurs at -1.8414375440963444 Iteration 47

X value is -1.9046087932144176

The local minimum occurs at -1.9046087932144176 Iteration 48

X value is -1.9665166173501292

The local minimum occurs at -1.9665166173501292 Iteration 49

X value is -2.0271862850031264

The local minimum occurs at -2.0271862850031264 Iteration 50

X value is -2.0866425593030637

The local minimum occurs at -2.0866425593030637 Iteration 51

X value is -2.1449097081170025

The local minimum occurs at -2.1449097081170025 Iteration 52

X value is -2.2020115139546625

The local minimum occurs at -2.2020115139546625 Iteration 53

X value is -2.257971283675569

The local minimum occurs at -2.257971283675569 Iteration 54

X value is -2.312811858002058

The local minimum occurs at -2.312811858002058 Iteration 55

X value is -2.3665556208420164

The local minimum occurs at -2.3665556208420164 Iteration 56

X value is -2.419224508425176

The local minimum occurs at -2.419224508425176 Iteration 57

X value is -2.4708400182566725

The local minimum occurs at -2.4708400182566725 Iteration 58

X value is -2.521423217891539

The local minimum occurs at -2.521423217891539 Iteration 59

X value is -2.570994753533708

The local minimum occurs at -2.570994753533708 Iteration 60

X value is -2.619574858463034

The local minimum occurs at -2.619574858463034 Iteration 61

X value is -2.667183361293773

The local minimum occurs at -2.667183361293773 Iteration 62

X value is -2.713839694067898

10/26/22, 5:25 PM ml\_4

Iteration 63

X value is -2.75956290018654

The local minimum occurs at -2.75956290018654 Iteration 64

X value is -2.804371642182809

The local minimum occurs at -2.804371642182809 Iteration 65

X value is -2.8482842093391527

The local minimum occurs at -2.8482842093391527 Iteration 66

X value is -2.8913185251523696

The local minimum occurs at -2.8913185251523696 Iteration 67

X value is -2.9334921546493224

The local minimum occurs at -2.9334921546493224 Iteration 68

X value is -2.974822311556336

The local minimum occurs at -2.974822311556336 Iteration 69

X value is -3.015325865325209

The local minimum occurs at -3.015325865325209 Iteration 70

X value is -3.055019348018705

The local minimum occurs at -3.055019348018705 Iteration 71

X value is -3.093918961058331

The local minimum occurs at -3.093918961058331 Iteration 72

X value is -3.1320405818371646

The local minimum occurs at -3.1320405818371646 Iteration 73

X value is -3.1693997702004215

The local minimum occurs at -3.1693997702004215 Iteration 74

X value is -3.206011774796413

The local minimum occurs at -3.206011774796413 Iteration 75

X value is -3.2418915393004846

The local minimum occurs at -3.2418915393004846 Iteration 76

X value is -3.277053708514475

The local minimum occurs at -3.277053708514475 Iteration 77

X value is -3.3115126343441856

The local minimum occurs at -3.3115126343441856 Iteration 78

X value is -3.345282381657302

The local minimum occurs at -3.345282381657302 Iteration 79

X value is -3.378376734024156

The local minimum occurs at -3.378376734024156 Iteration 80

X value is -3.4108091993436727

The local minimum occurs at -3.4108091993436727 Iteration 81

X value is -3.4425930153567994

The local minimum occurs at -3.4425930153567994 Iteration 82

X value is -3.4737411550496633

The local minimum occurs at -3.4737411550496633 Iteration 83

X value is -3.50426633194867

The local minimum occurs at -3.50426633194867 Iteration 84

X value is -3.534181005309697

The local minimum occurs at -3.534181005309697 Iteration 85

X value is -3.563497385203503

10/26/22, 5:25 PM ml\_4

Iteration 86

X value is -3.5922274374994325

The local minimum occurs at -3.5922274374994325 Iteration 87

X value is -3.620382888749444

The local minimum occurs at -3.620382888749444 Iteration 88

X value is -3.6479752309744553

The local minimum occurs at -3.6479752309744553 Iteration 89

X value is -3.675015726354966

The local minimum occurs at -3.675015726354966 Iteration 90

X value is -3.7015154118278666

The local minimum occurs at -3.7015154118278666 Iteration 91

X value is -3.7274851035913095

The local minimum occurs at -3.7274851035913095 Iteration 92

X value is -3.7529354015194833

The local minimum occurs at -3.7529354015194833 Iteration 93

X value is -3.7778766934890937

The local minimum occurs at -3.7778766934890937 Iteration 94

X value is -3.8023191596193118

The local minimum occurs at -3.8023191596193118 Iteration 95

X value is -3.8262727764269258

The local minimum occurs at -3.8262727764269258 Iteration 96

X value is -3.8497473208983872

The local minimum occurs at -3.8497473208983872 Iteration 97

X value is -3.8727523744804193

The local minimum occurs at -3.8727523744804193 Iteration 98

X value is -3.895297326990811

The local minimum occurs at -3.895297326990811 Iteration 99

X value is -3.917391380450995

The local minimum occurs at -3.917391380450995 Iteration 100

X value is -3.939043552841975

The local minimum occurs at -3.939043552841975 Iteration 101

X value is -3.9602626817851356

The local minimum occurs at -3.9602626817851356 Iteration 102

X value is -3.981057428149433

The local minimum occurs at -3.981057428149433 Iteration 103

X value is -4.001436279586445

The local minimum occurs at -4.001436279586445 Iteration 104

X value is -4.021407553994716

The local minimum occurs at -4.021407553994716 Iteration 105

X value is -4.040979402914822

The local minimum occurs at -4.040979402914822 Iteration 106

X value is -4.060159814856525

The local minimum occurs at -4.060159814856525 Iteration 107

X value is -4.078956618559395

The local minimum occurs at -4.078956618559395 Iteration 108

X value is -4.097377486188207

10/26/22, 5:25 PM ml\_4

Iteration 109

X value is -4.115429936464443

The local minimum occurs at -4.115429936464443 Iteration 110

X value is -4.133121337735154

The local minimum occurs at -4.133121337735154 Iteration 111

X value is -4.150458910980451

The local minimum occurs at -4.150458910980451 Iteration 112

X value is -4.167449732760842

The local minimum occurs at -4.167449732760842 Iteration 113

X value is -4.1841007381056246

The local minimum occurs at -4.1841007381056246 Iteration 114

X value is -4.200418723343512

The local minimum occurs at -4.200418723343512 Iteration 115

X value is -4.216410348876642

The local minimum occurs at -4.216410348876642 Iteration 116

X value is -4.2320821418991095

The local minimum occurs at -4.2320821418991095 Iteration 117

X value is -4.247440499061128

The local minimum occurs at -4.247440499061128 Iteration 118

X value is -4.262491689079905

The local minimum occurs at -4.262491689079905 Iteration 119

X value is -4.277241855298307

The local minimum occurs at -4.277241855298307 Iteration 120

X value is -4.291697018192341

The local minimum occurs at -4.291697018192341 Iteration 121

X value is -4.305863077828494

The local minimum occurs at -4.305863077828494 Iteration 122

X value is -4.319745816271924

The local minimum occurs at -4.319745816271924 Iteration 123

X value is -4.333350899946486

The local minimum occurs at -4.333350899946486 Iteration 124

X value is -4.3466838819475555

The local minimum occurs at -4.3466838819475555 Iteration 125

X value is -4.359750204308605

The local minimum occurs at -4.359750204308605 Iteration 126

X value is -4.372555200222433

The local minimum occurs at -4.372555200222433 Iteration 127

X value is -4.385104096217984

The local minimum occurs at -4.385104096217984 Iteration 128

X value is -4.3974020142936245

The local minimum occurs at -4.3974020142936245 Iteration 129

X value is -4.409453974007752

The local minimum occurs at -4.409453974007752 Iteration 130

X value is -4.421264894527597

The local minimum occurs at -4.421264894527597 Iteration 131

X value is -4.432839596637045

10/26/22, 5:25 PM ml\_4

Iteration 132

X value is -4.444182804704305

The local minimum occurs at -4.444182804704305 Iteration 133

X value is -4.4552991486102185

The local minimum occurs at -4.4552991486102185 Iteration 134

X value is -4.466193165638014

The local minimum occurs at -4.466193165638014 Iteration 135

X value is -4.4768693023252535

The local minimum occurs at -4.4768693023252535 Iteration 136

X value is -4.487331916278748

The local minimum occurs at -4.487331916278748 Iteration 137

X value is -4.497585277953173

The local minimum occurs at -4.497585277953173 Iteration 138

X value is -4.50763357239411

The local minimum occurs at -4.50763357239411 Iteration 139

X value is -4.517480900946228

The local minimum occurs at -4.517480900946228 Iteration 140

X value is -4.527131282927304

The local minimum occurs at -4.527131282927304 Iteration 141

X value is -4.536588657268758

The local minimum occurs at -4.536588657268758 Iteration 142

X value is -4.545856884123382

The local minimum occurs at -4.545856884123382 Iteration 143

X value is -4.5549397464409145

The local minimum occurs at -4.5549397464409145 Iteration 144

X value is -4.563840951512097

The local minimum occurs at -4.563840951512097 Iteration 145

X value is -4.572564132481855

The local minimum occurs at -4.572564132481855 Iteration 146

X value is -4.581112849832218

The local minimum occurs at -4.581112849832218 Iteration 147

X value is -4.589490592835574

The local minimum occurs at -4.589490592835574 Iteration 148

X value is -4.597700780978863

The local minimum occurs at -4.597700780978863 Iteration 149

X value is -4.605746765359285

The local minimum occurs at -4.605746765359285 Iteration 150

X value is -4.6136318300521

The local minimum occurs at -4.6136318300521 Iteration 151

X value is -4.621359193451058

The local minimum occurs at -4.621359193451058 Iteration 152

X value is -4.628932009582036

The local minimum occurs at -4.628932009582036 Iteration 153

X value is -4.636353369390395

The local minimum occurs at -4.636353369390395 Iteration 154

X value is -4.643626302002588

10/26/22, 5:25 PM ml\_4

Iteration 155

X value is -4.650753775962536

The local minimum occurs at -4.650753775962536 Iteration 156

X value is -4.657738700443285

The local minimum occurs at -4.657738700443285 Iteration 157

X value is -4.664583926434419

The local minimum occurs at -4.664583926434419 Iteration 158

X value is -4.671292247905731

The local minimum occurs at -4.671292247905731 Iteration 159

X value is -4.6778664029476165

The local minimum occurs at -4.6778664029476165 Iteration 160

X value is -4.684309074888664

The local minimum occurs at -4.684309074888664 Iteration 161

X value is -4.6906228933908904

The local minimum occurs at -4.6906228933908904 Iteration 162

X value is -4.696810435523073

The local minimum occurs at -4.696810435523073 Iteration 163

X value is -4.702874226812612

The local minimum occurs at -4.702874226812612 Iteration 164

X value is -4.708816742276359

The local minimum occurs at -4.708816742276359 Iteration 165

X value is -4.714640407430832

The local minimum occurs at -4.714640407430832 Iteration 166

X value is -4.720347599282215

The local minimum occurs at -4.720347599282215 Iteration 167

X value is -4.725940647296571

The local minimum occurs at -4.725940647296571 Iteration 168

X value is -4.731421834350639

The local minimum occurs at -4.731421834350639 Iteration 169

X value is -4.736793397663627

The local minimum occurs at -4.736793397663627 Iteration 170

X value is -4.742057529710355

The local minimum occurs at -4.742057529710355 Iteration 171

X value is -4.747216379116147

The local minimum occurs at -4.747216379116147 Iteration 172

X value is -4.752272051533824

The local minimum occurs at -4.752272051533824 Iteration 173

X value is -4.757226610503148

The local minimum occurs at -4.757226610503148 Iteration 174

X value is -4.762082078293084

The local minimum occurs at -4.762082078293084 Iteration 175

X value is -4.766840436727223

The local minimum occurs at -4.766840436727223 Iteration 176

X value is -4.771503627992678

The local minimum occurs at -4.771503627992678 Iteration 177

X value is -4.776073555432824

10/26/22, 5:25 PM ml\_4

Iteration 178

X value is -4.780552084324168

The local minimum occurs at -4.780552084324168 Iteration 179

X value is -4.784941042637685

The local minimum occurs at -4.784941042637685 Iteration 180

X value is -4.7892422217849315

The local minimum occurs at -4.7892422217849315 Iteration 181

X value is -4.793457377349233

The local minimum occurs at -4.793457377349233 Iteration 182

X value is -4.7975882298022485

The local minimum occurs at -4.7975882298022485 Iteration 183

X value is -4.801636465206204

The local minimum occurs at -4.801636465206204 Iteration 184

X value is -4.805603735902079

The local minimum occurs at -4.805603735902079 Iteration 185

X value is -4.809491661184038

The local minimum occurs at -4.809491661184038 Iteration 186

X value is -4.813301827960357

The local minimum occurs at -4.813301827960357 Iteration 187

X value is -4.81703579140115

The local minimum occurs at -4.81703579140115 Iteration 188

X value is -4.820695075573127

The local minimum occurs at -4.820695075573127 Iteration 189

X value is -4.824281174061665

The local minimum occurs at -4.824281174061665 Iteration 190

X value is -4.827795550580431

The local minimum occurs at -4.827795550580431 Iteration 191

X value is -4.831239639568823

The local minimum occurs at -4.831239639568823 Iteration 192

X value is -4.834614846777447

The local minimum occurs at -4.834614846777447 Iteration 193

X value is -4.837922549841898

The local minimum occurs at -4.837922549841898 Iteration 194

X value is -4.84116409884506

The local minimum occurs at -4.84116409884506 Iteration 195

X value is -4.844340816868159

The local minimum occurs at -4.844340816868159 Iteration 196

X value is -4.847454000530796

The local minimum occurs at -4.847454000530796 Iteration 197

X value is -4.85050492052018

The local minimum occurs at -4.85050492052018 Iteration 198

X value is -4.853494822109776

The local minimum occurs at -4.853494822109776 Iteration 199

X value is -4.85642492566758

The local minimum occurs at -4.85642492566758 Iteration 200

X value is -4.859296427154229

10/26/22, 5:25 PM ml\_4

Iteration 201

X value is -4.862110498611145

The local minimum occurs at -4.862110498611145 Iteration 202

X value is -4.864868288638922

The local minimum occurs at -4.864868288638922 Iteration 203

X value is -4.867570922866143

The local minimum occurs at -4.867570922866143 Iteration 204

X value is -4.87021950440882

The local minimum occurs at -4.87021950440882 Iteration 205

X value is -4.872815114320644

The local minimum occurs at -4.872815114320644 Iteration 206

X value is -4.875358812034231

The local minimum occurs at -4.875358812034231 Iteration 207

X value is -4.877851635793546

The local minimum occurs at -4.877851635793546 Iteration 208

X value is -4.880294603077676

The local minimum occurs at -4.880294603077676 Iteration 209

X value is -4.882688711016122

The local minimum occurs at -4.882688711016122 Iteration 210

X value is -4.8850349367958

The local minimum occurs at -4.8850349367958 Iteration 211

X value is -4.887334238059884

The local minimum occurs at -4.887334238059884 Iteration 212

X value is -4.8895875532986866

The local minimum occurs at -4.8895875532986866 Iteration 213

X value is -4.891795802232712

The local minimum occurs at -4.891795802232712 Iteration 214

X value is -4.893959886188058

The local minimum occurs at -4.893959886188058 Iteration 215

X value is -4.896080688464297

The local minimum occurs at -4.896080688464297 Iteration 216

X value is -4.898159074695011

The local minimum occurs at -4.898159074695011 Iteration 217

X value is -4.9001958932011105

The local minimum occurs at -4.9001958932011105 Iteration 218

X value is -4.902191975337089

The local minimum occurs at -4.902191975337089 Iteration 219

X value is -4.904148135830347

The local minimum occurs at -4.904148135830347 Iteration 220

X value is -4.90606517311374

The local minimum occurs at -4.90606517311374 Iteration 221

X value is -4.907943869651465

The local minimum occurs at -4.907943869651465 Iteration 222

X value is -4.909784992258436

The local minimum occurs at -4.909784992258436 Iteration 223

X value is -4.911589292413267

10/26/22, 5:25 PM ml\_4

Iteration 224

X value is -4.913357506565002

The local minimum occurs at -4.913357506565002 Iteration 225

X value is -4.915090356433702

The local minimum occurs at -4.915090356433702 Iteration 226

X value is -4.9167885493050285

The local minimum occurs at -4.9167885493050285 Iteration 227

X value is -4.918452778318928

The local minimum occurs at -4.918452778318928 Iteration 228

X value is -4.920083722752549

The local minimum occurs at -4.920083722752549 Iteration 229

X value is -4.921682048297498

The local minimum occurs at -4.921682048297498 Iteration 230

X value is -4.923248407331548

The local minimum occurs at -4.923248407331548 Iteration 231

X value is -4.9247834391849175

The local minimum occurs at -4.9247834391849175 Iteration 232

X value is -4.926287770401219

The local minimum occurs at -4.926287770401219 Iteration 233

X value is -4.927762014993195

The local minimum occurs at -4.927762014993195 Iteration 234

X value is -4.929206774693331

The local minimum occurs at -4.929206774693331 Iteration 235

X value is -4.930622639199464

The local minimum occurs at -4.930622639199464 Iteration 236

X value is -4.932010186415474

The local minimum occurs at -4.932010186415474 Iteration 237

X value is -4.933369982687164

The local minimum occurs at -4.933369982687164 Iteration 238

X value is -4.934702583033421

The local minimum occurs at -4.934702583033421 Iteration 239

X value is -4.936008531372753

The local minimum occurs at -4.936008531372753 Iteration 240

X value is -4.937288360745298

The local minimum occurs at -4.937288360745298 Iteration 241

X value is -4.938542593530392

The local minimum occurs at -4.938542593530392 Iteration 242

X value is -4.939771741659784

The local minimum occurs at -4.939771741659784 Iteration 243

X value is -4.940976306826588

The local minimum occurs at -4.940976306826588 Iteration 244

X value is -4.942156780690056

The local minimum occurs at -4.942156780690056 Iteration 245

X value is -4.943313645076255

The local minimum occurs at -4.943313645076255 Iteration 246

X value is -4.94444737217473

10/26/22, 5:25 PM ml\_4

Iteration 247

X value is -4.945558424731236

The local minimum occurs at -4.945558424731236 Iteration 248

X value is -4.946647256236611

The local minimum occurs at -4.946647256236611 Iteration 249

X value is -4.947714311111879

The local minimum occurs at -4.947714311111879 Iteration 250

X value is -4.9487600248896415

The local minimum occurs at -4.9487600248896415 Iteration 251

X value is -4.949784824391848

The local minimum occurs at -4.949784824391848 Iteration 252

X value is -4.950789127904011

The local minimum occurs at -4.950789127904011 Iteration 253

X value is -4.951773345345931

The local minimum occurs at -4.951773345345931 Iteration 254

X value is -4.952737878439012

The local minimum occurs at -4.952737878439012 Iteration 255

X value is -4.953683120870232

The local minimum occurs at -4.953683120870232 Iteration 256

X value is -4.954609458452827

The local minimum occurs at -4.954609458452827 Iteration 257

X value is -4.955517269283771

The local minimum occurs at -4.955517269283771 Iteration 258

X value is -4.956406923898095

The local minimum occurs at -4.956406923898095 Iteration 259

X value is -4.957278785420133

The local minimum occurs at -4.957278785420133 Iteration 260

X value is -4.958133209711731

The local minimum occurs at -4.958133209711731 Iteration 261

X value is -4.958970545517496

The local minimum occurs at -4.958970545517496 Iteration 262

X value is -4.959791134607146

The local minimum occurs at -4.959791134607146 Iteration 263

X value is -4.960595311915003

The local minimum occurs at -4.960595311915003 Iteration 264

X value is -4.9613834056767026

The local minimum occurs at -4.9613834056767026 Iteration 265

X value is -4.962155737563169

The local minimum occurs at -4.962155737563169 Iteration 266

X value is -4.962912622811905

The local minimum occurs at -4.962912622811905 Iteration 267

X value is -4.963654370355667

The local minimum occurs at -4.963654370355667 Iteration 268

X value is -4.964381282948554

The local minimum occurs at -4.964381282948554 Iteration 269

X value is -4.965093657289583

10/26/22, 5:25 PM ml\_4

Iteration 270

X value is -4.965791784143791

The local minimum occurs at -4.965791784143791 Iteration 271

X value is -4.966475948460915

The local minimum occurs at -4.966475948460915 Iteration 272

X value is -4.967146429491697

The local minimum occurs at -4.967146429491697 Iteration 273

X value is -4.967803500901863

The local minimum occurs at -4.967803500901863 Iteration 274

X value is -4.968447430883826

The local minimum occurs at -4.968447430883826 Iteration 275

X value is -4.969078482266149

The local minimum occurs at -4.969078482266149 Iteration 276

X value is -4.969696912620826

The local minimum occurs at -4.969696912620826 Iteration 277

X value is -4.970302974368409

The local minimum occurs at -4.970302974368409 Iteration 278

X value is -4.970896914881041

The local minimum occurs at -4.970896914881041 Iteration 279

X value is -4.97147897658342

The local minimum occurs at -4.97147897658342 Iteration 280

X value is -4.972049397051752

The local minimum occurs at -4.972049397051752 Iteration 281

X value is -4.972608409110717

The local minimum occurs at -4.972608409110717 Iteration 282

X value is -4.973156240928502

The local minimum occurs at -4.973156240928502 Iteration 283

X value is -4.973693116109932

The local minimum occurs at -4.973693116109932 Iteration 284

X value is -4.974219253787734

The local minimum occurs at -4.974219253787734 Iteration 285

X value is -4.974734868711979

The local minimum occurs at -4.974734868711979 Iteration 286

X value is -4.975240171337739

The local minimum occurs at -4.975240171337739 Iteration 287

X value is -4.975735367910985

The local minimum occurs at -4.975735367910985 Iteration 288

X value is -4.976220660552765

The local minimum occurs at -4.976220660552765 Iteration 289

X value is -4.976696247341709

The local minimum occurs at -4.976696247341709 Iteration 290

X value is -4.977162322394875

The local minimum occurs at -4.977162322394875 Iteration 291

X value is -4.977619075946977

The local minimum occurs at -4.977619075946977 Iteration 292

X value is -4.978066694428038

10/26/22, 5:25 PM ml\_4

Iteration 293

X value is -4.978505360539477

The local minimum occurs at -4.978505360539477 Iteration 294

X value is -4.978935253328687

The local minimum occurs at -4.978935253328687 Iteration 295

X value is -4.979356548262113

The local minimum occurs at -4.979356548262113 Iteration 296

X value is -4.979769417296871

The local minimum occurs at -4.979769417296871 Iteration 297

X value is -4.980174028950934

The local minimum occurs at -4.980174028950934 Iteration 298

X value is -4.980570548371915

The local minimum occurs at -4.980570548371915 Iteration 299

X value is -4.980959137404477

The local minimum occurs at -4.980959137404477 Iteration 300

X value is -4.981339954656387

The local minimum occurs at -4.981339954656387 Iteration 301

X value is -4.981713155563259

The local minimum occurs at -4.981713155563259 Iteration 302

X value is -4.982078892451994

The local minimum occurs at -4.982078892451994 Iteration 303

X value is -4.9824373146029535

The local minimum occurs at -4.9824373146029535 Iteration 304

X value is -4.982788568310895

The local minimum occurs at -4.982788568310895 Iteration 305

X value is -4.983132796944677

The local minimum occurs at -4.983132796944677 Iteration 306

X value is -4.983470141005784

The local minimum occurs at -4.983470141005784 Iteration 307

X value is -4.983800738185668

The local minimum occurs at -4.983800738185668 Iteration 308

X value is -4.984124723421955

The local minimum occurs at -4.984124723421955 Iteration 309

X value is -4.984442228953515

The local minimum occurs at -4.984442228953515 Iteration 310

X value is -4.984753384374445

The local minimum occurs at -4.984753384374445 Iteration 311

X value is -4.985058316686956

The local minimum occurs at -4.985058316686956 Iteration 312

X value is -4.9853571503532175

The local minimum occurs at -4.9853571503532175 Iteration 313

X value is -4.985650007346153

The local minimum occurs at -4.985650007346153 Iteration 314

X value is -4.9859370071992295

The local minimum occurs at -4.9859370071992295 Iteration 315

X value is -4.986218267055245

10/26/22, 5:25 PM ml\_4

Iteration 316

X value is -4.98649390171414

The local minimum occurs at -4.98649390171414 Iteration 317

X value is -4.986764023679857

The local minimum occurs at -4.986764023679857 Iteration 318

X value is -4.98702874320626

The local minimum occurs at -4.98702874320626 Iteration 319

X value is -4.987288168342134

The local minimum occurs at -4.987288168342134 Iteration 320

X value is -4.987542404975292

The local minimum occurs at -4.987542404975292 Iteration 321

X value is -4.987791556875786

The local minimum occurs at -4.987791556875786 Iteration 322

X value is -4.98803572573827

The local minimum occurs at -4.98803572573827 Iteration 323

X value is -4.988275011223505

The local minimum occurs at -4.988275011223505 Iteration 324

X value is -4.988509510999035

The local minimum occurs at -4.988509510999035 Iteration 325

X value is -4.988739320779054

The local minimum occurs at -4.988739320779054 Iteration 326

X value is -4.988964534363473

The local minimum occurs at -4.988964534363473 Iteration 327

X value is -4.989185243676204

The local minimum occurs at -4.989185243676204 Iteration 328

X value is -4.98940153880268

The local minimum occurs at -4.98940153880268 Iteration 329

X value is -4.989613508026626

The local minimum occurs at -4.989613508026626 Iteration 330

X value is -4.989821237866094

The local minimum occurs at -4.989821237866094 Iteration 331

X value is -4.990024813108772

The local minimum occurs at -4.990024813108772 Iteration 332

X value is -4.9902243168465965

The local minimum occurs at -4.9902243168465965 Iteration 333

X value is -4.990419830509665

The local minimum occurs at -4.990419830509665 Iteration 334

X value is -4.990611433899471

The local minimum occurs at -4.990611433899471 Iteration 335

X value is -4.990799205221482

The local minimum occurs at -4.990799205221482 Iteration 336

X value is -4.990983221117052

The local minimum occurs at -4.990983221117052 Iteration 337

X value is -4.991163556694711

The local minimum occurs at -4.991163556694711 Iteration 338

X value is -4.991340285560817

10/26/22, 5:25 PM ml\_4

Iteration 339

X value is -4.9915134798496

The local minimum occurs at -4.9915134798496 Iteration 340

X value is -4.991683210252608

The local minimum occurs at -4.991683210252608 Iteration 341

X value is -4.991849546047556

The local minimum occurs at -4.991849546047556 Iteration 342

X value is -4.992012555126605

The local minimum occurs at -4.992012555126605 Iteration 343

X value is -4.992172304024073

The local minimum occurs at -4.992172304024073 Iteration 344

X value is -4.992328857943591

The local minimum occurs at -4.992328857943591 Iteration 345

X value is -4.99248228078472

The local minimum occurs at -4.99248228078472 Iteration 346

X value is -4.992632635169025

The local minimum occurs at -4.992632635169025 Iteration 347

X value is -4.9927799824656445

The local minimum occurs at -4.9927799824656445 Iteration 348

X value is -4.992924382816332

The local minimum occurs at -4.992924382816332 Iteration 349

X value is -4.993065895160005

The local minimum occurs at -4.993065895160005 Iteration 350

X value is -4.993204577256805

The local minimum occurs at -4.993204577256805 Iteration 351

X value is -4.993340485711669

The local minimum occurs at -4.993340485711669 Iteration 352

X value is -4.993473675997436

The local minimum occurs at -4.993473675997436 Iteration 353

X value is -4.993604202477487

The local minimum occurs at -4.993604202477487 Iteration 354

X value is -4.993732118427937

The local minimum occurs at -4.993732118427937 Iteration 355

X value is -4.993857476059379

The local minimum occurs at -4.993857476059379 Iteration 356

X value is -4.993980326538191

The local minimum occurs at -4.993980326538191 Iteration 357

X value is -4.9941007200074266

The local minimum occurs at -4.9941007200074266 Iteration 358

X value is -4.994218705607278

The local minimum occurs at -4.994218705607278 Iteration 359

X value is -4.994334331495133

The local minimum occurs at -4.994334331495133 Iteration 360

X value is -4.994447644865231

The local minimum occurs at -4.994447644865231 Iteration 361

X value is -4.994558691967926

10/26/22, 5:25 PM ml\_4

Iteration 362

X value is -4.994667518128567

The local minimum occurs at -4.994667518128567 Iteration 363

X value is -4.994774167765996

The local minimum occurs at -4.994774167765996 Iteration 364

X value is -4.9948786844106765

The local minimum occurs at -4.9948786844106765 Iteration 365

X value is -4.994981110722463

The local minimum occurs at -4.994981110722463 Iteration 366

X value is -4.995081488508014

The local minimum occurs at -4.995081488508014 Iteration 367

X value is -4.995179858737854

The local minimum occurs at -4.995179858737854 Iteration 368

X value is -4.995276261563097

The local minimum occurs at -4.995276261563097 Iteration 369

X value is -4.995370736331835

The local minimum occurs at -4.995370736331835 Iteration 370

X value is -4.9954633216051985

The local minimum occurs at -4.9954633216051985 Iteration 371

X value is -4.995554055173095

The local minimum occurs at -4.995554055173095 Iteration 372

X value is -4.995642974069633

The local minimum occurs at -4.995642974069633 Iteration 373

X value is -4.99573011458824

The local minimum occurs at -4.99573011458824 Iteration 374

X value is -4.995815512296476

The local minimum occurs at -4.995815512296476 Iteration 375

X value is -4.995899202050547

The local minimum occurs at -4.995899202050547 Iteration 376

X value is -4.995981218009535

The local minimum occurs at -4.995981218009535 Iteration 377

X value is -4.996061593649345

The local minimum occurs at -4.996061593649345 Iteration 378

X value is -4.996140361776358

The local minimum occurs at -4.996140361776358 Iteration 379

X value is -4.996217554540831

The local minimum occurs at -4.996217554540831 Iteration 380

X value is -4.996293203450014

The local minimum occurs at -4.996293203450014 Iteration 381

X value is -4.996367339381013

The local minimum occurs at -4.996367339381013 Iteration 382

X value is -4.996439992593393

The local minimum occurs at -4.996439992593393 Iteration 383

X value is -4.996511192741525

The local minimum occurs at -4.996511192741525 Iteration 384

X value is -4.996580968886694

10/26/22, 5:25 PM ml\_4

Iteration 385

X value is -4.99664934950896

The local minimum occurs at -4.99664934950896 Iteration 386

X value is -4.9967163625187805

The local minimum occurs at -4.9967163625187805 Iteration 387

X value is -4.996782035268405

The local minimum occurs at -4.996782035268405 Iteration 388

X value is -4.996846394563037

The local minimum occurs at -4.996846394563037 Iteration 389

X value is -4.996909466671776

The local minimum occurs at -4.996909466671776 Iteration 390

X value is -4.996971277338341

The local minimum occurs at -4.996971277338341 Iteration 391

X value is -4.997031851791574

The local minimum occurs at -4.997031851791574 Iteration 392

X value is -4.997091214755742

The local minimum occurs at -4.997091214755742 Iteration 393

X value is -4.997149390460628

The local minimum occurs at -4.997149390460628 Iteration 394

X value is -4.997206402651415

The local minimum occurs at -4.997206402651415 Iteration 395

X value is -4.997262274598387

The local minimum occurs at -4.997262274598387 Iteration 396

X value is -4.997317029106419

The local minimum occurs at -4.997317029106419 Iteration 397

X value is -4.997370688524291

The local minimum occurs at -4.997370688524291 Iteration 398

X value is -4.997423274753805

The local minimum occurs at -4.997423274753805 Iteration 399

X value is -4.997474809258729

The local minimum occurs at -4.997474809258729 Iteration 400

X value is -4.997525313073554

The local minimum occurs at -4.997525313073554 Iteration 401

X value is -4.997574806812083

The local minimum occurs at -4.997574806812083 Iteration 402

X value is -4.997623310675841

The local minimum occurs at -4.997623310675841 Iteration 403

X value is -4.997670844462324

The local minimum occurs at -4.997670844462324 Iteration 404

X value is -4.997717427573078

The local minimum occurs at -4.997717427573078 Iteration 405

X value is -4.997763079021617

The local minimum occurs at -4.997763079021617 Iteration 406

X value is -4.997807817441185

The local minimum occurs at -4.997807817441185 Iteration 407

X value is -4.997851661092361

10/26/22, 5:25 PM ml\_4

Iteration 408

X value is -4.997894627870514

The local minimum occurs at -4.997894627870514 Iteration 409

X value is -4.997936735313104

The local minimum occurs at -4.997936735313104 Iteration 410

X value is -4.9979780006068415

The local minimum occurs at -4.9979780006068415 Iteration 411

X value is -4.998018440594705

The local minimum occurs at -4.998018440594705 Iteration 412

X value is -4.998058071782811

The local minimum occurs at -4.998058071782811 Iteration 413

X value is -4.998096910347155

The local minimum occurs at -4.998096910347155 Iteration 414

X value is -4.998134972140212

The local minimum occurs at -4.998134972140212 Iteration 415

X value is -4.998172272697408

The local minimum occurs at -4.998172272697408 Iteration 416

X value is -4.9982088272434595

The local minimum occurs at -4.9982088272434595 Iteration 417

X value is -4.998244650698591

The local minimum occurs at -4.998244650698591 Iteration 418

X value is -4.998279757684619

The local minimum occurs at -4.998279757684619 Iteration 419

X value is -4.998314162530927

The local minimum occurs at -4.998314162530927 Iteration 420

X value is -4.998347879280309

The local minimum occurs at -4.998347879280309 Iteration 421

X value is -4.998380921694703

The local minimum occurs at -4.998380921694703 Iteration 422

X value is -4.998413303260809

The local minimum occurs at -4.998413303260809 Iteration 423

X value is -4.998445037195593

The local minimum occurs at -4.998445037195593 Iteration 424

X value is -4.998476136451681

The local minimum occurs at -4.998476136451681 Iteration 425

X value is -4.998506613722648

The local minimum occurs at -4.998506613722648 Iteration 426

X value is -4.998536481448195

The local minimum occurs at -4.998536481448195 Iteration 427

X value is -4.998565751819231

The local minimum occurs at -4.998565751819231 Iteration 428

X value is -4.998594436782846

The local minimum occurs at -4.998594436782846 Iteration 429

X value is -4.998622548047189

The local minimum occurs at -4.998622548047189 Iteration 430

X value is -4.998650097086245

10/26/22, 5:25 PM ml\_4

Iteration 431

X value is -4.9986770951445205

The local minimum occurs at -4.9986770951445205 Iteration 432

X value is -4.99870355324163

The local minimum occurs at -4.99870355324163 Iteration 433

X value is -4.998729482176797

The local minimum occurs at -4.998729482176797 Iteration 434

X value is -4.998754892533261

The local minimum occurs at -4.998754892533261 Iteration 435

X value is -4.998779794682596

The local minimum occurs at -4.998779794682596 Iteration 436

X value is -4.998804198788944

The local minimum occurs at -4.998804198788944 Iteration 437

X value is -4.998828114813166

The local minimum occurs at -4.998828114813166 Iteration 438

X value is -4.998851552516903

The local minimum occurs at -4.998851552516903 Iteration 439

X value is -4.998874521466565

The local minimum occurs at -4.998874521466565 Iteration 440

X value is -4.998897031037234

The local minimum occurs at -4.998897031037234 Iteration 441

X value is -4.998919090416489

The local minimum occurs at -4.998919090416489 Iteration 442

X value is -4.99894070860816

The local minimum occurs at -4.99894070860816 Iteration 443

X value is -4.998961894435997

The local minimum occurs at -4.998961894435997 Iteration 444

X value is -4.998982656547277

The local minimum occurs at -4.998982656547277 Iteration 445

X value is -4.999003003416331

The local minimum occurs at -4.999003003416331 Iteration 446

X value is -4.999022943348004

The local minimum occurs at -4.999022943348004 Iteration 447

X value is -4.999042484481044

The local minimum occurs at -4.999042484481044 Iteration 448

X value is -4.999061634791423

The local minimum occurs at -4.999061634791423 Iteration 449

X value is -4.999080402095594

The local minimum occurs at -4.999080402095594 Iteration 450

X value is -4.999098794053682

The local minimum occurs at -4.999098794053682 Iteration 451

X value is -4.999116818172609

The local minimum occurs at -4.999116818172609 Iteration 452

X value is -4.999134481809157

The local minimum occurs at -4.999134481809157 Iteration 453

X value is -4.999151792172974

10/26/22, 5:25 PM ml\_4

Iteration 454

X value is -4.999168756329515

The local minimum occurs at -4.999168756329515 Iteration 455

X value is -4.999185381202924

The local minimum occurs at -4.999185381202924 Iteration 456

X value is -4.999201673578866

The local minimum occurs at -4.999201673578866 Iteration 457

X value is -4.999217640107289

The local minimum occurs at -4.999217640107289 Iteration 458

X value is -4.999233287305143

The local minimum occurs at -4.999233287305143 Iteration 459

X value is -4.9992486215590395

The local minimum occurs at -4.9992486215590395 Iteration 460

X value is -4.999263649127859

The local minimum occurs at -4.999263649127859 Iteration 461

X value is -4.999278376145302

The local minimum occurs at -4.999278376145302 Iteration 462

X value is -4.999292808622396

The local minimum occurs at -4.999292808622396 Iteration 463

X value is -4.999306952449948

The local minimum occurs at -4.999306952449948 Iteration 464

X value is -4.999320813400949

The local minimum occurs at -4.999320813400949 Iteration 465

X value is -4.99933439713293

The local minimum occurs at -4.99933439713293 Iteration 466

X value is -4.999347709190272

The local minimum occurs at -4.999347709190272 Iteration 467

X value is -4.9993607550064665

The local minimum occurs at -4.9993607550064665 Iteration 468

X value is -4.999373539906337

The local minimum occurs at -4.999373539906337 Iteration 469

X value is -4.99938606910821

The local minimum occurs at -4.99938606910821 Iteration 470

X value is -4.9993983477260455

The local minimum occurs at -4.9993983477260455 Iteration 471

X value is -4.999410380771525

The local minimum occurs at -4.999410380771525 Iteration 472

X value is -4.999422173156094

The local minimum occurs at -4.999422173156094 Iteration 473

X value is -4.9994337296929725

The local minimum occurs at -4.9994337296929725 Iteration 474

X value is -4.999445055099113

The local minimum occurs at -4.999445055099113 Iteration 475

X value is -4.999456153997131

The local minimum occurs at -4.999456153997131 Iteration 476

X value is -4.999467030917188

The local minimum occurs at -4.999467030917188

10/26/22, 5:25 PM ml\_4

Iteration 477

X value is -4.9994776902988445

The local minimum occurs at -4.9994776902988445 Iteration 478

X value is -4.999488136492867

The local minimum occurs at -4.999488136492867 Iteration 479

X value is -4.99949837376301

The local minimum occurs at -4.99949837376301 Iteration 480

X value is -4.99950840628775

The local minimum occurs at -4.99950840628775 Iteration 481

X value is -4.999518238161995

The local minimum occurs at -4.999518238161995 Iteration 482

X value is -4.999527873398756

The local minimum occurs at -4.999527873398756 Iteration 483

X value is -4.99953731593078

The local minimum occurs at -4.99953731593078 Iteration 484

X value is -4.999546569612165

The local minimum occurs at -4.999546569612165 Iteration 485

X value is -4.999555638219921

The local minimum occurs at -4.999555638219921 Iteration 486

X value is -4.999564525455523

The local minimum occurs at -4.999564525455523 Iteration 487

X value is -4.999573234946412

The local minimum occurs at -4.999573234946412 Iteration 488

X value is -4.9995817702474845

The local minimum occurs at -4.9995817702474845 Iteration 489

X value is -4.999590134842535

The local minimum occurs at -4.999590134842535 Iteration 490

X value is -4.999598332145684

The local minimum occurs at -4.999598332145684 Iteration 491

X value is -4.99960636550277

The local minimum occurs at -4.99960636550277 Iteration 492

X value is -4.999614238192715

The local minimum occurs at -4.999614238192715 Iteration 493

X value is -4.999621953428861

The local minimum occurs at -4.999621953428861 Iteration 494

X value is -4.999629514360284

The local minimum occurs at -4.999629514360284 Iteration 495

X value is -4.999636924073078

The local minimum occurs at -4.999636924073078 Iteration 496

X value is -4.999644185591617

The local minimum occurs at -4.999644185591617 Iteration 497

X value is -4.999651301879784

The local minimum occurs at -4.999651301879784 Iteration 498

X value is -4.999658275842188

The local minimum occurs at -4.999658275842188 Iteration 499

X value is -4.999665110325345

10/26/22, 5:25 PM ml\_4

Iteration 500

X value is -4.999671808118838

The local minimum occurs at -4.999671808118838 Iteration 501

X value is -4.9996783719564615

The local minimum occurs at -4.9996783719564615 Iteration 502

X value is -4.999684804517332

The local minimum occurs at -4.999684804517332 Iteration 503

X value is -4.999691108426985

The local minimum occurs at -4.999691108426985 Iteration 504

X value is -4.999697286258446

The local minimum occurs at -4.999697286258446 Iteration 505

X value is -4.9997033405332765

The local minimum occurs at -4.9997033405332765 Iteration 506

X value is -4.999709273722611

The local minimum occurs at -4.999709273722611 Iteration 507

X value is -4.999715088248159

The local minimum occurs at -4.999715088248159 Iteration 508

X value is -4.999720786483196

The local minimum occurs at -4.999720786483196 Iteration 509

X value is -4.999726370753532

The local minimum occurs at -4.999726370753532 Iteration 510

X value is -4.999731843338461

The local minimum occurs at -4.999731843338461 Iteration 511

X value is -4.999737206471692

The local minimum occurs at -4.999737206471692 Iteration 512

X value is -4.999742462342258

The local minimum occurs at -4.999742462342258 Iteration 513

X value is -4.999747613095413

The local minimum occurs at -4.999747613095413 Iteration 514

X value is -4.999752660833504

The local minimum occurs at -4.999752660833504 Iteration 515

X value is -4.999757607616834

The local minimum occurs at -4.999757607616834 Iteration 516

X value is -4.999762455464498

The local minimum occurs at -4.999762455464498 Iteration 517

X value is -4.999767206355208

The local minimum occurs at -4.999767206355208 Iteration 518

X value is -4.999771862228104

The local minimum occurs at -4.999771862228104 Iteration 519

X value is -4.999776424983542

The local minimum occurs at -4.999776424983542 Iteration 520

X value is -4.9997808964838715

The local minimum occurs at -4.9997808964838715 Iteration 521

X value is -4.999785278554194

The local minimum occurs at -4.999785278554194 Iteration 522

X value is -4.9997895729831106

10/26/22, 5:25 PM ml\_4

Iteration 523

X value is -4.999793781523448

The local minimum occurs at -4.999793781523448 Iteration 524

X value is -4.999797905892979

The local minimum occurs at -4.999797905892979 Iteration 525

X value is -4.999801947775119

The local minimum occurs at -4.999801947775119 Iteration 526

X value is -4.999805908819617

The local minimum occurs at -4.999805908819617 Iteration 527

X value is -4.999809790643225

The local minimum occurs at -4.999809790643225 Iteration 528

X value is -4.99981359483036

The local minimum occurs at -4.99981359483036 Iteration 529

X value is -4.999817322933753

The local minimum occurs at -4.999817322933753 Iteration 530

X value is -4.999820976475077

The local minimum occurs at -4.999820976475077 Iteration 531

X value is -4.999824556945576

The local minimum occurs at -4.999824556945576 Iteration 532

X value is -4.999828065806665

The local minimum occurs at -4.999828065806665 Iteration 533

X value is -4.9998315044905315

The local minimum occurs at -4.9998315044905315 Iteration 534

X value is -4.999834874400721

The local minimum occurs at -4.999834874400721 Iteration 535

X value is -4.999838176912706

The local minimum occurs at -4.999838176912706 Iteration 536

X value is -4.999841413374452

The local minimum occurs at -4.999841413374452 Iteration 537

X value is -4.999844585106963

The local minimum occurs at -4.999844585106963 Iteration 538

X value is -4.999847693404824

The local minimum occurs at -4.999847693404824 Iteration 539

X value is -4.999850739536727

The local minimum occurs at -4.999850739536727 Iteration 540

X value is -4.999853724745993

The local minimum occurs at -4.999853724745993 Iteration 541

X value is -4.999856650251073

The local minimum occurs at -4.999856650251073 Iteration 542

X value is -4.999859517246051

The local minimum occurs at -4.999859517246051 Iteration 543

X value is -4.99986232690113

The local minimum occurs at -4.99986232690113 Iteration 544

X value is -4.999865080363108

The local minimum occurs at -4.999865080363108 Iteration 545

X value is -4.999867778755846

10/26/22, 5:25 PM ml\_4

Iteration 546

X value is -4.999870423180729

The local minimum occurs at -4.999870423180729 Iteration 547

X value is -4.999873014717115

The local minimum occurs at -4.999873014717115 Iteration 548

X value is -4.999875554422772

The local minimum occurs at -4.999875554422772 Iteration 549

X value is -4.999878043334316

The local minimum occurs at -4.999878043334316 Iteration 550

X value is -4.99988048246763

The local minimum occurs at -4.99988048246763 Iteration 551

X value is -4.999882872818278

The local minimum occurs at -4.999882872818278 Iteration 552

X value is -4.999885215361912

The local minimum occurs at -4.999885215361912 Iteration 553

X value is -4.999887511054674

The local minimum occurs at -4.999887511054674 Iteration 554

X value is -4.999889760833581

The local minimum occurs at -4.999889760833581 Iteration 555

X value is -4.999891965616909

The local minimum occurs at -4.999891965616909 Iteration 556

X value is -4.999894126304571

The local minimum occurs at -4.999894126304571 Iteration 557

X value is -4.999896243778479

The local minimum occurs at -4.999896243778479 Iteration 558

X value is -4.999898318902909

The local minimum occurs at -4.999898318902909 Iteration 559

X value is -4.999900352524851

The local minimum occurs at -4.999900352524851 Iteration 560

X value is -4.9999023454743545

The local minimum occurs at -4.9999023454743545 Iteration 561

X value is -4.999904298564868

The local minimum occurs at -4.999904298564868 Iteration 562

X value is -4.9999062125935705

The local minimum occurs at -4.9999062125935705 Iteration 563

X value is -4.999908088341699

The local minimum occurs at -4.999908088341699 Iteration 564

X value is -4.9999099265748645

The local minimum occurs at -4.9999099265748645 Iteration 565

X value is -4.999911728043367

The local minimum occurs at -4.999911728043367 Iteration 566

X value is -4.9999134934825

The local minimum occurs at -4.9999134934825 Iteration 567

X value is -4.99991522361285

The local minimum occurs at -4.99991522361285 Iteration 568

X value is -4.999916919140593

10/26/22, 5:25 PM ml\_4

Iteration 569

X value is -4.999918580757781

The local minimum occurs at -4.999918580757781 Iteration 570

X value is -4.999920209142625

The local minimum occurs at -4.999920209142625 Iteration 571

X value is -4.999921804959773

The local minimum occurs at -4.999921804959773 Iteration 572

X value is -4.9999233688605775

The local minimum occurs at -4.9999233688605775 Iteration 573

X value is -4.999924901483366

The local minimum occurs at -4.999924901483366 Iteration 574

X value is -4.999926403453699

The local minimum occurs at -4.999926403453699 Iteration 575

X value is -4.999927875384625

The local minimum occurs at -4.999927875384625 Iteration 576

X value is -4.999929317876933

The local minimum occurs at -4.999929317876933 Iteration 577

X value is -4.999930731519394

The local minimum occurs at -4.999930731519394 Iteration 578

X value is -4.999932116889006

The local minimum occurs at -4.999932116889006 Iteration 579

X value is -4.999933474551226

The local minimum occurs at -4.999933474551226 Iteration 580

X value is -4.999934805060202

The local minimum occurs at -4.999934805060202 Iteration 581

X value is -4.999936108958998

The local minimum occurs at -4.999936108958998 Iteration 582

X value is -4.999937386779818

The local minimum occurs at -4.999937386779818 Iteration 583

X value is -4.999938639044221

The local minimum occurs at -4.999938639044221 Iteration 584

X value is -4.999939866263337

The local minimum occurs at -4.999939866263337 Iteration 585

X value is -4.99994106893807

The local minimum occurs at -4.99994106893807 Iteration 586

X value is -4.999942247559309

The local minimum occurs at -4.999942247559309 Iteration 587

X value is -4.999943402608123

The local minimum occurs at -4.999943402608123 Iteration 588

X value is -4.9999445345559606

The local minimum occurs at -4.9999445345559606 Iteration 589

X value is -4.999945643864842

The local minimum occurs at -4.999945643864842 Iteration 590

X value is -4.999946730987545

The local minimum occurs at -4.999946730987545 Iteration 591

X value is -4.999947796367794

10/26/22, 5:25 PM ml\_4

Iteration 592

X value is -4.999948840440438

The local minimum occurs at -4.999948840440438 Iteration 593

X value is -4.999949863631629

The local minimum occurs at -4.999949863631629 Iteration 594

X value is -4.999950866358997

The local minimum occurs at -4.999950866358997 Iteration 595

X value is -4.9999518490318176

The local minimum occurs at -4.9999518490318176

In [ ]:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Assignment No: 5

**Title Name:** Implement K-Nearest Neighbors algorithm

**Name**: Aditi Shivani

**Class** : BE **Div**: 1 **Batch**: A

# Roll No: 405A005

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**import** numpy **as** np **import** pandas **as** pd **import** math

**import** matplotlib.pyplot **as** plt

**import** seaborn **as** sns

**from** sklearn.neighbors **import** KNeighborsClassifier

**from** sklearn **import** metrics

**from** sklearn.model\_selection **import** cross\_val\_score

**from** sklearn.model\_selection **import** KFold

**%matplotlib** inline

location **=** 'diabetes.csv' f **=** pd**.**read\_csv(location) data **=** pd**.**DataFrame(f) data**.**head()

In [1]:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Out[1]: | **Pregnancies** | | **Glucose** | **BloodPressure** | **SkinThickness** | **Insulin** | **BMI** | **Pedigree** | **Age** | **Outcome** |
|  | **0** | 6 | 148 | 72 | 35 | 0 | 33.6 | 0.627 | 50 | 1 |
|  | **1** | 1 | 85 | 66 | 29 | 0 | 26.6 | 0.351 | 31 | 0 |
|  | **2** | 8 | 183 | 64 | 0 | 0 | 23.3 | 0.672 | 32 | 1 |
|  | **3** | 1 | 89 | 66 | 23 | 94 | 28.1 | 0.167 | 21 | 0 |
|  | **4** | 0 | 137 | 40 | 35 | 168 | 43.1 | 2.288 | 33 | 1 |

In [2]:

cols\_clean **=** ['Glucose','BloodPressure','SkinThickness','Insulin','BMI','Pedigree']

*# with this function , i dealt with missing values and NaN values*

**for** i **in** cols\_clean:

data[i] **=** data[i]**.**replace(0,np**.**NaN) cols\_mean **=** int(data[i]**.**mean(skipna**=True**)) data[i] **=** data[i]**.**replace(np**.**NaN, cols\_mean)

data1 **=** data data1**.**head()**.**style**.**highlight\_max(color**=**"lightblue")**.**highlight\_min(color**=**"red")

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Out[2]: | **Pregnancies** | **Glucose** | **BloodPressure** | **SkinThickness** | **Insulin** | **BMI** | **Pedigree** | **Age O** |
| **0** | 6 | 148.000000 | 72.000000 | 35.000000 | 155.000000 | 33.600000 | 0.627000 | 50 |
| **1** | 1 | 85.000000 | 66.000000 | 29.000000 | 155.000000 | 26.600000 | 0.351000 | 31 |
| **2** | 8 | 183.000000 | 64.000000 | 29.000000 | 155.000000 | 23.300000 | 0.672000 | 32 |
| **3** | 1 | 89.000000 | 66.000000 | 23.000000 | 94.000000 | 28.100000 | 0.167000 | 21 |



**4**

137.000000

33

35.000000 168.000000 43.100000 2.288000

40.000000

0

In [3]:

print(data1**.**describe())

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| count | Pregnancies  768.000000 | Glucose  768.000000 | BloodPressure  768.000000 | SkinThickness  768.000000 | Insulin \  768.00000 |
| mean | 3.845052 | 121.682292 | 72.386719 | 29.108073 | 155.28125 |
| std | 3.369578 | 30.435999 | 12.096642 | 8.791221 | 85.02155 |
| min | 0.000000 | 44.000000 | 24.000000 | 7.000000 | 14.00000 |
| 25% | 1.000000 | 99.750000 | 64.000000 | 25.000000 | 121.50000 |
| 50% | 3.000000 | 117.000000 | 72.000000 | 29.000000 | 155.00000 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 75% | 6.000000 | 140.250000 | 80.000000 | | 32.000000 155.00000 |
| max | 17.000000 | 199.000000 | 122.000000 | | 99.000000 846.00000 |
|  | BMI | Pedigree | Age | Outcome | |
| count | 768.000000 | 768.000000 | 768.000000 | 768.000000 | |
| mean | 32.450911 | 0.471876 | 33.240885 | 0.348958 | |
| std | 6.875366 | 0.331329 | 11.760232 | 0.476951 | |
| min | 18.200000 | 0.078000 | 21.000000 | 0.000000 | |
| 25% | 27.500000 | 0.243750 | 24.000000 | 0.000000 | |
| 50% | 32.000000 | 0.372500 | 29.000000 | 0.000000 | |
| 75% | 36.600000 | 0.626250 | 41.000000 | 1.000000 | |
| max | 67.100000 | 2.420000 | 81.000000 | 1.000000 | |

In [4]:

*# for the purpose of simplicity and analysing the most relevent data , we will sele # Glucose , Insulin and BMI*

q\_cols **=** ['Glucose','Insulin','BMI','Outcome']

*# defining variables and features for the dataset for splitting*

df **=** data1[q\_cols] print(df**.**head(2))

Glucose Insulin BMI Outcome 0 148.0 155.0 33.6 1

1 85.0 155.0 26.6 0

In [5]:

*# let's split the data into training and testing datasets*

split **=** 0.75 *# 75% train and 25% test dataset*

total\_len **=** len(df)

split\_df **=** int(total\_len**\***split)

train, test **=** df**.**iloc[:split\_df,0:4],df**.**iloc[split\_df:,0:4] train\_x **=** train[['Glucose','Insulin','BMI']]

train\_y **=** train['Outcome']

test\_x **=** test[['Glucose','Insulin','BMI']] test\_y **=** test['Outcome']

In [6]:

a **=** len(train\_x) b **=** len(test\_x)

print(' Training data =',a,'\n','Testing data =',b,'\n','Total data length = ',a**+**b)

In [7]:

Training data = 576 Testing data = 192 Total data length = 768

**def** knn(x\_train, y\_train, x\_test, y\_test,n): n\_range **=** range(1, n)

results **=** []

**for** n **in** n\_range:

knn **=** KNeighborsClassifier(n\_neighbors**=**n) knn**.**fit(x\_train, y\_train)

*#Predict the response for test dataset*

predict\_y **=** knn**.**predict(x\_test)

accuracy **=** metrics**.**accuracy\_score(y\_test, predict\_y)

*#matrix = confusion\_matrix(y\_test,predict\_y)*

*#seaborn\_matrix = sns.heatmap(matrix, annot = True, cmap="Blues",cbar=True)*

results**.**append(accuracy)

**return** results

In [8]:

n**=** 500

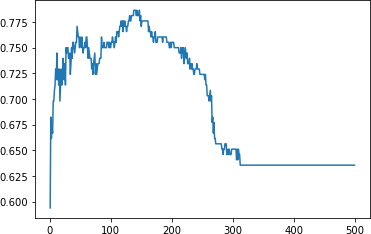
output **=** knn(train\_x,train\_y,test\_x,test\_y,n) n\_range **=** range(1, n)

plt**.**plot(n\_range, output)

Out[8]:

In [ ]:

[<matplotlib.lines.Line2D at 0x1f812813580>]



# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Assignment : ML Mini Project**

**Title Name:** Build a machine learning model that predicts the type of people who survived the Titanic shipwreck using passenger data (i.e. name, age, gender, socio-economic class, etc.).

**Name**: Aditi Shivani

**Class** : BE **Div**: 1 **Batch**: A

# Roll No: 405A005

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Title**: Build a machine learning model that predicts the type of people who survived the Titanic shipwreck using passenger data (i.e. name, age, gender, socio-economic class, etc.).

**Problem Statement:** Write a program for building a model that predicts the type of people who survived the Titanic shipwreckusing passenger data (i.e. name, age, gender, socio-economic class, etc.).

**Prerequisites:** Random Forest Classifer

**Objectives:** To build a machine learning model that predicts the type of people who survived the Titanic shipwreck using passenger data (i.e. name, age, gender, socio-economic class, etc.).

# Theory:

**Random Forest Classifier:**

Random Forest is a supervised learning algorithm.

It uses the *ensemble learning technique.*(Ensemble learning is using multiple algorithms at a time or a single algorithm multiple times to make a model more powerful) to build several decision trees at random data points. Then their predictions are averaged. Taking the average value of predictions made by several decision trees and then predict the final result.You can take reference from the above image.

Types of Random Forest models:

1. Random Forest Prediction for a classification problem.
2. Random Forest Prediction for a regression problem. What is Random Forest Classification?
3. It is an ensemble tree-based learning algorithm.
4. The Random Forest Classifier is a set of decision trees from randomly selected subset of training set.
5. It *aggregates the votes from different decision trees* to decide the final class of the test object.
6. *Random forest algorithm creates decision trees on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting.*

Structure of Random Forest

How does Random Forest Algorithm works?

Let us understand the working of Random Forest algorithm with the help of following steps −

* Step 1 − First, start with the selection of random samples from a given dataset.
* Step 2 − Next, this algorithm will construct a decision tree for every sample. Then it will get the prediction result from every decision tree.
* Step 3 − In this step, voting will be performed for every predicted result.
* Step 4 − At last, select the most voted prediction result as the final prediction result.

In [1]:

*# linear algebra*

**import** numpy **as** np

*# data processing*

**import** pandas **as** pd

*# data visualization*

**import** seaborn **as** sns

**%matplotlib** inline

**from** matplotlib **import** pyplot **as** plt

**from** matplotlib **import** style

*# Algorithms*

**from** sklearn **import** linear\_model

**from** sklearn.linear\_model **import** LogisticRegression **from** sklearn.ensemble **import** RandomForestClassifier **from** sklearn.linear\_model **import** Perceptron

**from** sklearn.linear\_model **import** SGDClassifier **from** sklearn.tree **import** DecisionTreeClassifier **from** sklearn.neighbors **import** KNeighborsClassifier **from** sklearn.svm **import** SVC, LinearSVC

**from** sklearn.naive\_bayes **import** GaussianNB

**from** sklearn.datasets **import** make\_classification **from** sklearn.model\_selection **import** train\_test\_split **from** sklearn.pipeline **import** make\_pipeline

**from** sklearn.preprocessing **import** StandardScaler

**from** sklearn **import** preprocessing

In [2]:

test\_df **=** pd**.**read\_csv("test.csv") train\_df **=** pd**.**read\_csv("train.csv")

In [3]:

train\_df**.**info()

<class 'pandas.core.frame.DataFrame'> RangeIndex: 891 entries, 0 to 890 Data columns (total 12 columns):

# Column Non-Null Count Dtype

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0 |  | PassengerId | 891 | non-null |  | int64 |
| 1 |  | Survived | 891 | non-null |  | int64 |
| 2 |  | Pclass | 891 | non-null |  | int64 |
| 3 |  | Name | 891 | non-null |  | object |
| 4 |  | Sex | 891 | non-null |  | object |
| 5 |  | Age | 714 | non-null |  | float64 |
| 6 |  | SibSp | 891 | non-null |  | int64 |
| 7 |  | Parch | 891 | non-null |  | int64 |

|  |  |  |
| --- | --- | --- |
| 8 Ticket 891 | non-null | object |
| 9 Fare 891 | non-null | float64 |
| 10 Cabin 204 | non-null | object |
| 11 Embarked 889 | non-null | object |
| dtypes: float64(2), | int64(5), | object(5) |

memory usage: 83.7+ KB

In [4]:

train\_df**.**describe()

Out[4]: **PassengerId Survived Pclass Age SibSp Parch Fare count** 891.000000 891.000000 891.000000 714.000000 891.000000 891.000000 891.000000

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **PassengerId** | **Survived** | **Pclass** | **Age** | **SibSp** | **Parch** | **Fare** |
| **mean** | 446.000000 | 0.383838 | 2.308642 | 29.699118 | 0.523008 | 0.381594 | 32.204208 |
| **std** | 257.353842 | 0.486592 | 0.836071 | 14.526497 | 1.102743 | 0.806057 | 49.693429 |
| **min** | 1.000000 | 0.000000 | 1.000000 | 0.420000 | 0.000000 | 0.000000 | 0.000000 |
| **25%** | 223.500000 | 0.000000 | 2.000000 | 20.125000 | 0.000000 | 0.000000 | 7.910400 |

**50%** 446.000000 0.000000 3.000000 28.000000 0.000000 0.000000 14.454200

**75%** 668.500000 1.000000 3.000000 38.000000 1.000000 0.000000 31.000000

**max** 891.000000 1.000000 3.000000 80.000000 8.000000 6.000000 512.329200

In [5]:

train\_df**.**head(8)

Out[5]: **PassengerId Survived Pclass Name Sex Age SibSp Parch Ticket Fare Cabin**

**0** 1 0 3

Braund, Mr. Owen

Harris

male 22.0 1 0

A/5 7.2500 NaN

21171

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | Cumings, Mrs. John |  | | | | | | |
| **1** | 2 | 1 | 1 | Bradley (Florence | female | 38.0 | 1 | 0 | PC 17599 | 71.2833 | C85 |
|  |  |  |  | Briggs |  |  |  |  |  |  |  |
|  |  |  |  | Th... |  |  |  |  |  |  |  |
| **2** | 3 | 1 | 3 | Heikkinen,  Miss.  Laina | female | 26.0 | 0 | 0 | STON/O2. 3101282 | 7.9250 | NaN |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Futrelle,  Mrs.  **3** 4 1 1 Jacques female 35.0 1 0 113803 53.1000 C123  Heath (Lily May  Peel)  Allen, Mr. | | | | | | |
| **4** 5 0 3 Will am male Henry | 35.0 | 0 | 0 | 373450 | 8.0500 | NaN |
| **5** 6 0 3 Mo an, male | NaN | 0 | 0 | 330877 | 8.4583 | NaN |
| McCar hy,  **6** 7 0 1 Mr. male | 54.0 | 0 | 0 | 17463 | 51.8625 | E46 |

Timothy J

**7**

8

0

3

Palsson, Master. Gosta Leonard

male 2.0

3

1 349909 21.0750 NaN

Mr. Ja es

In [6]:

|  |  |  |
| --- | --- | --- |
|  |  |  |
| total **=** train\_df**.**isnull()**.**sum()**.**sort\_values(ascending**=False**) percent\_1 **=** train\_df**.**isnull()**.**sum()**/**train\_df**.**isnull()**.**count()**\***100 percent\_2 **=** (round(percent\_1, 1))**.**sort\_values(ascending**=False**)  missing\_data **=** pd**.**concat([total, percent\_2], axis**=**1, keys**=**['Total', '%']) missing\_data**.**head(5) | | |

Out[6]:

In [7]:

train\_df**.**columns**.**values

**Total %**

**Cabin** 687 77.1

**Age** 177 19.9

**Embarked** 2 0.2

**PassengerId** 0 0.0

**Survived** 0 0.0

Out[7]:

In [8]:

array(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp', 'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'], dtype=object)

survived **=** 'survived' not\_survived **=** 'not survived'

fig, axes **=** plt**.**subplots(nrows**=**1, ncols**=**2,figsize**=**(10, 4)) women **=** train\_df[train\_df['Sex']**==**'female']

men **=** train\_df[train\_df['Sex']**==**'male']

ax **=** sns**.**distplot(women[women['Survived']**==**1]**.**Age**.**dropna(), bins**=**18, label **=** survive ax **=** sns**.**distplot(women[women['Survived']**==**0]**.**Age**.**dropna(), bins**=**40, label **=** not\_sur ax**.**legend()

ax**.**set\_title('Female')

ax **=** sns**.**distplot(men[men['Survived']**==**1]**.**Age**.**dropna(), bins**=**18, label **=** survived, ax **=** sns**.**distplot(men[men['Survived']**==**0]**.**Age**.**dropna(), bins**=**40, label **=** not\_surviv ax**.**legend()

\_ **=** ax**.**set\_title('Male')

C:\Users\hp\anaconda3\lib\site-packages\seaborn\distributions.py:2557: FutureWarnin g: `distplot` is a deprecated function and will be removed in a future version. Plea se adapt your code to use either `displot` (a figure-level function with similar fle xibility) or `histplot` (an axes-level function for histograms).

warnings.warn(msg, FutureWarning)

C:\Users\hp\anaconda3\lib\site-packages\seaborn\distributions.py:2557: FutureWarnin g: `distplot` is a deprecated function and will be removed in a future version. Plea se adapt your code to use either `displot` (a figure-level function with similar fle xibility) or `histplot` (an axes-level function for histograms).

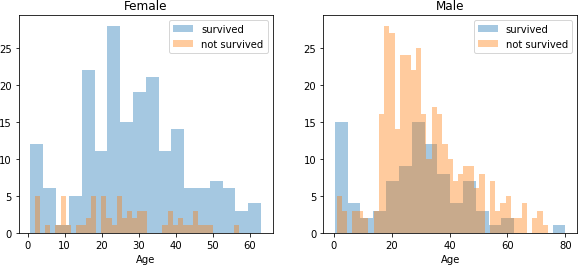
warnings.warn(msg, FutureWarning)

C:\Users\hp\anaconda3\lib\site-packages\seaborn\distributions.py:2557: FutureWarnin g: `distplot` is a deprecated function and will be removed in a future version. Plea se adapt your code to use either `displot` (a figure-level function with similar fle xibility) or `histplot` (an axes-level function for histograms).

warnings.warn(msg, FutureWarning)

C:\Users\hp\anaconda3\lib\site-packages\seaborn\distributions.py:2557: FutureWarnin g: `distplot` is a deprecated function and will be removed in a future version. Plea se adapt your code to use either `displot` (a figure-level function with similar fle xibility) or `histplot` (an axes-level function for histograms).

warnings.warn(msg, FutureWarning)



In [9]:

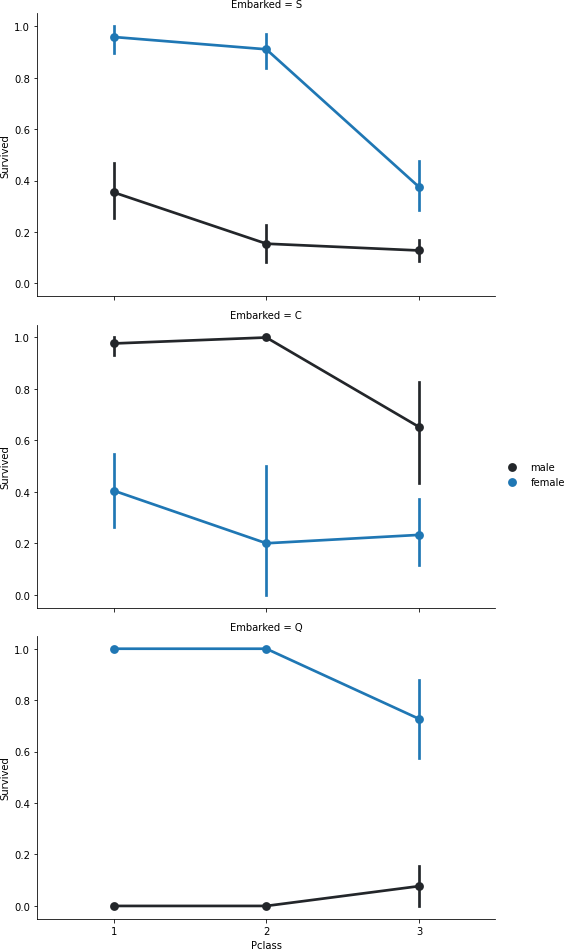
FacetGrid **=** sns**.**FacetGrid(train\_df, row**=**'Embarked', size**=**4.5, aspect**=**1.6) FacetGrid**.**map(sns**.**pointplot, 'Pclass', 'Survived', 'Sex', palette**=None**, order**=None**, FacetGrid**.**add\_legend()

Out[9]:

C:\Users\hp\anaconda3\lib\site-packages\seaborn\axisgrid.py:316: UserWarning: The `s ize` parameter has been renamed to `height`; please update your code.

warnings.warn(msg, UserWarning)

<seaborn.axisgrid.FacetGrid at 0x2903cbbd970>



In [10]:

sns**.**barplot(x**=**'Pclass', y**=**'Survived', data**=**train\_df)

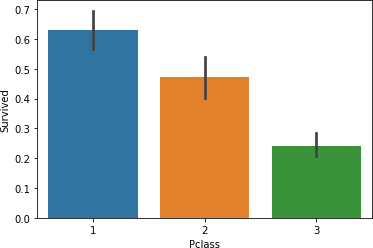
Out[10]:

In [11]:

grid **=** sns**.**FacetGrid(train\_df, col**=**'Survived', row**=**'Pclass', size**=**2.2, aspect**=**1.6) grid**.**map(plt**.**hist, 'Age', alpha**=.**5, bins**=**20)

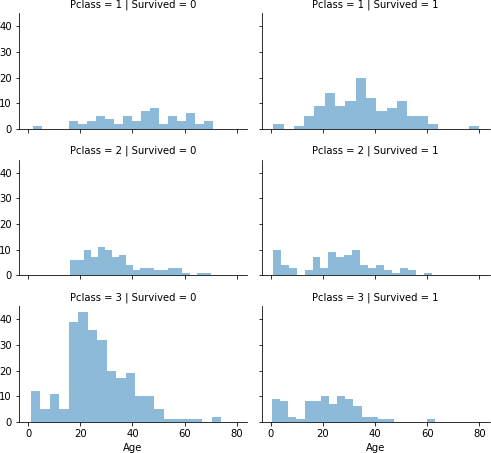
grid**.**add\_legend();

<AxesSubplot:xlabel='Pclass', ylabel='Survived'>



C:\Users\hp\anaconda3\lib\site-packages\seaborn\axisgrid.py:316: UserWarning: The `s ize` parameter has been renamed to `height`; please update your code.

warnings.warn(msg, UserWarning)



In [12]:

data **=** [train\_df, test\_df]

**for** dataset **in** data:

dataset['relatives'] **=** dataset['SibSp'] **+** dataset['Parch'] dataset**.**loc[dataset['relatives'] **>** 0, 'not\_alone'] **=** 0

dataset**.**loc[dataset['relatives'] **==** 0, 'not\_alone'] **=** 1 dataset['not\_alone'] **=** dataset['not\_alone']**.**astype(int)

train\_df['not\_alone']**.**value\_counts()

Out[12]:

In [13]:

1 537

0 354

Name: not\_alone, dtype: int64

axes **=** sns**.**factorplot('relatives','Survived',

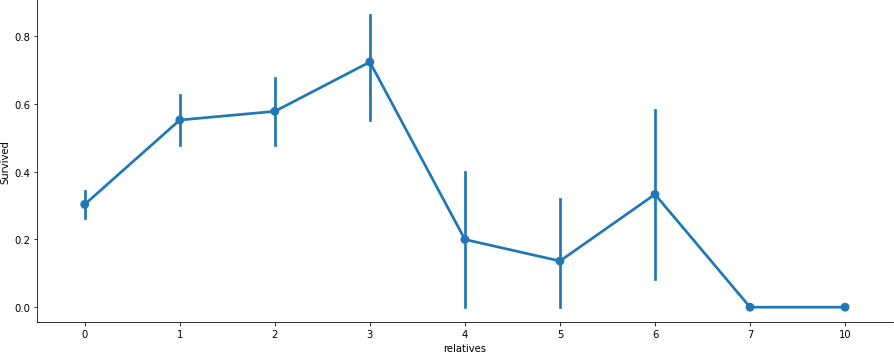
data**=**train\_df, aspect **=** 2.5, )

C:\Users\hp\anaconda3\lib\site-packages\seaborn\categorical.py:3714: UserWarning: Th e `factorplot` function has been renamed to `catplot`. The original name will be rem oved in a future release. Please update your code. Note that the default `kind` in ` factorplot` (`'point'`) has changed `'strip'` in `catplot`.

warnings.warn(msg)

C:\Users\hp\anaconda3\lib\site-packages\seaborn\\_decorators.py:36: FutureWarning: Pa ss the following variables as keyword args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.

warnings.warn(

In [14]:

train\_df **=** train\_df**.**drop(['PassengerId'], axis**=**1)

In [15]:

**import** re

deck **=** {"A": 1, "B": 2, "C": 3, "D": 4, "E": 5, "F": 6, "G": 7, "U": 8}

data **=** [train\_df, test\_df]

**for** dataset **in** data:

dataset['Cabin'] **=** dataset['Cabin']**.**fillna("U0")

dataset['Deck'] **=** dataset['Cabin']**.**map(**lambda** x: re**.**compile("([a-zA-Z]+)")**.**searc dataset['Deck'] **=** dataset['Deck']**.**map(deck)

dataset['Deck'] **=** dataset['Deck']**.**fillna(0) dataset['Deck'] **=** dataset['Deck']**.**astype(int)

*# we can now drop the cabin feature* train\_df **=** train\_df**.**drop(['Cabin'], axis**=**1) test\_df **=** test\_df**.**drop(['Cabin'], axis**=**1)

In [16]:

data **=** [train\_df, test\_df]

**for** dataset **in** data:

mean **=** train\_df["Age"]**.**mean() std **=** test\_df["Age"]**.**std()

is\_null **=** dataset["Age"]**.**isnull()**.**sum()

*# compute random numbers between the mean, std and is\_null*

rand\_age **=** np**.**random**.**randint(mean **-** std, mean **+** std, size **=** is\_null) *# fill NaN values in Age column with random values generated* age\_slice **=** dataset["Age"]**.**copy()

age\_slice[np**.**isnan(age\_slice)] **=** rand\_age dataset["Age"] **=** age\_slice

dataset["Age"] **=** train\_df["Age"]**.**astype(int) train\_df["Age"]**.**isnull()**.**sum()

Out[16]: 0

In [17]:

train\_df['Embarked']**.**describe()

|  |  |  |
| --- | --- | --- |
| Out[17]: | count | 889 |
|  | unique | 3 |
|  | top  freq | S  644 |

Name: Embarked, dtype: object

In [18]:

common\_value **=** 'S'

data **=** [train\_df, test\_df]

**for** dataset **in** data:

dataset['Embarked'] **=** dataset['Embarked']**.**fillna(common\_value)

In [19]:

train\_df**.**info()

<class 'pandas.core.frame.DataFrame'> RangeIndex: 891 entries, 0 to 890 Data columns (total 13 columns):

# Column Non-Null Count Dtype

1. Survived 891 non-null int64
2. Pclass 891 non-null int64
3. Name 891 non-null object
4. Sex 891 non-null object
5. Age 891 non-null int32
6. SibSp 891 non-null int64
7. Parch 891 non-null int64
8. Ticket 891 non-null object
9. Fare 891 non-null float64
10. Embarked 891 non-null object
11. relatives 891 non-null int64
12. not\_alone 891 non-null int32
13. Deck 891 non-null int32

dtypes: float64(1), int32(3), int64(5), object(4) memory usage: 80.2+ KB

In [20]:

data **=** [train\_df, test\_df]

**for** dataset **in** data:

dataset['Fare'] **=** dataset['Fare']**.**fillna(0) dataset['Fare'] **=** dataset['Fare']**.**astype(int)

In [21]:

data **=** [train\_df, test\_df]

titles **=** {"Mr": 1, "Miss": 2, "Mrs": 3, "Master": 4, "Rare": 5}

**for** dataset **in** data:

*# extract titles*

dataset['Title'] **=** dataset**.**Name**.**str**.**extract(' ([A-Za-z]+)\.', expand**=False**)

*# replace titles with a more common title or as Rare*

dataset['Title'] **=** dataset['Title']**.**replace(['Lady', 'Countess','Capt', 'Col','D

'Major', 'Rev', 'Sir', 'Jonkheer', 'Dona dataset['Title'] **=** dataset['Title']**.**replace('Mlle', 'Miss')

dataset['Title'] **=** dataset['Title']**.**replace('Ms', 'Miss') dataset['Title'] **=** dataset['Title']**.**replace('Mme', 'Mrs') *# convert titles into numbers*

dataset['Title'] **=** dataset['Title']**.**map(titles) *# filling NaN with 0, to get safe* dataset['Title'] **=** dataset['Title']**.**fillna(0)

train\_df **=** train\_df**.**drop(['Name'], axis**=**1) test\_df **=** test\_df**.**drop(['Name'], axis**=**1)

In [22]:

genders **=** {"male": 0, "female": 1} data **=** [train\_df, test\_df]

**for** dataset **in** data:

dataset['Sex'] **=** dataset['Sex']**.**map(genders)

In [23]:

train\_df['Ticket']**.**describe()

Out[23]: count 891

unique681 top 347082

freq 7

Name: Ticket, dtype: object

In [24]:

train\_df **=** train\_df**.**drop(['Ticket'], axis**=**1) test\_df **=** test\_df**.**drop(['Ticket'], axis**=**1)

In [25]:

ports **=** {"S": 0, "C": 1, "Q": 2}

data **=** [train\_df, test\_df]

**for** dataset **in** data:

dataset['Embarked'] **=** dataset['Embarked']**.**map(ports)

In [26]:

data **=** [train\_df, test\_df]

**for** dataset **in** data:

dataset['Age'] **=** dataset['Age']**.**astype(int) dataset**.**loc[ dataset['Age'] **<=** 11, 'Age'] **=** 0

dataset**.**loc[ dataset['Age'] **>** 66, 'Age'] **=** 6

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| dataset**.**loc[(dataset['Age'] | **>** 11) **&** | (dataset['Age'] | **<=** | 18), | 'Age'] **=** | 1 |
| dataset**.**loc[(dataset['Age'] | **>** 18) **&** | (dataset['Age'] | **<=** | 22), | 'Age'] **=** | 2 |
| dataset**.**loc[(dataset['Age'] | **>** 22) **&** | (dataset['Age'] | **<=** | 27), | 'Age'] **=** | 3 |
| dataset**.**loc[(dataset['Age'] | **>** 27) **&** | (dataset['Age'] | **<=** | 33), | 'Age'] **=** | 4 |
| dataset**.**loc[(dataset['Age'] | **>** 33) **&** | (dataset['Age'] | **<=** | 40), | 'Age'] **=** | 5 |
| dataset**.**loc[(dataset['Age'] | **>** 40) **&** | (dataset['Age'] | **<=** | 66), | 'Age'] **=** | 6 |

In [27]:

train\_df['Age']**.**value\_counts()

|  |  |  |
| --- | --- | --- |
| Out[27]: | 4 | 164 |
|  | 6 | 164 |
|  | 5 | 139 |
|  | 3 | 137 |
|  | 2 | 122 |
|  | 1 | 97 |
|  | 0 | 68 |

Name: Age, dtype: int64

In [28]:

train\_df**.**head(10)

10/26/22, 5:36 PM Titanic

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Out[28]: | **Survived** | **Pclass** | **Sex** | **Age** | **SibSp** | **Parch** | **Fare** | **Embarked** | **relatives** | **not\_alone** | **Deck** | **Title** |
| **0** |  | 0 | 3 | 0 | 2 | 1 | 0 | 7 | 0 | 1 | 0 | 8 | 1 |
| **1** |  | 1 | 1 | 1 | 5 | 1 | 0 | 71 | 1 | 1 | 0 | 3 | 3 |
| **2** |  | 1 | 3 | 1 | 3 | 0 | 0 | 7 | 0 | 0 | 1 | 8 | 2 |
| **3** |  | 1 | 1 | 1 | 5 | 1 | 0 | 53 | 0 | 1 | 0 | 3 | 3 |
| **4** |  | 0 | 3 | 0 | 5 | 0 | 0 | 8 | 0 | 0 | 1 | 8 | 1 |
| **5** |  | 0 | 3 | 0 | 1 | 0 | 0 | 8 | 2 | 0 | 1 | 8 | 1 |
| **6** |  | 0 | 1 | 0 | 6 | 0 | 0 | 51 | 0 | 0 | 1 | 5 | 1 |
| **7** |  | 0 | 3 | 0 | 0 | 3 | 1 | 21 | 0 | 4 | 0 | 8 | 4 |
| **8** |  | 1 | 3 | 1 | 3 | 0 | 2 | 11 | 0 | 2 | 0 | 8 | 3 |
| **9** |  | 1 | 2 | 1 | 1 | 1 | 0 | 30 | 1 | 1 | 0 | 8 | 3 |

In [29]:

data **=** [train\_df, test\_df]

**for** dataset **in** data:

dataset**.**loc[ dataset['Fare'] **<=** 7.91, 'Fare'] **=** 0

dataset**.**loc[(dataset['Fare'] **>** 7.91) **&** (dataset['Fare'] **<=** 14.454), 'Fare'] **=** 1 dataset**.**loc[(dataset['Fare'] **>** 14.454) **&** (dataset['Fare'] **<=** 31), 'Fare'] **=** 2 dataset**.**loc[(dataset['Fare'] **>** 31) **&** (dataset['Fare'] **<=** 99), 'Fare'] **=** 3 dataset**.**loc[(dataset['Fare'] **>** 99) **&** (dataset['Fare'] **<=** 250), 'Fare'] **=** 4 dataset**.**loc[ dataset['Fare'] **>** 250, 'Fare'] **=** 5

dataset['Fare'] **=** dataset['Fare']**.**astype(int)

In [30]:

data **=** [train\_df, test\_df]

**for** dataset **in** data:

dataset['Age\_Class']**=** dataset['Age']**\*** dataset['Pclass']

In [31]:

**for** dataset **in** data:

dataset['Fare\_Per\_Person'] **=** dataset['Fare']**/**(dataset['relatives']**+**1) dataset['Fare\_Per\_Person'] **=** dataset['Fare\_Per\_Person']**.**astype(int)

*# Let's take a last look at the training set, before we start training the models.*

train\_df**.**head(10)

Out[31]: **Survived Pclass Sex Age SibSp Parch Fare Embarked relatives not\_alone Deck Title A 0** 0 3 0 2 1 0 0 0 1 0 8 1

**1** 1 1 1 5 1 0 3 1 1 0 3 3

**2** 1 3 1 3 0 0 0 0 0 1 8 2

**3** 1 1 1 5 1 0 3 0 1 0 3 3

**4** 0 3 0 5 0 0 1 0 0 1 8 1

**5** 0 3 0 1 0 0 1 2 0 1 8 1

**6** 0 1 0 6 0 0 3 0 0 1 5 1

**7** 0 3 0 0 3 1 2 0 4 0 8 4

**8** 1 3 1 3 0 2 1 0 2 0 8 3

10/26/22, 5:36 PM Titanic

**Survived Pclass Sex Age SibSp Parch Fare Embarked relatives not\_alone Deck Title A**

**9** 1 2 1 1 1 0 2 1 1 0 8 3

X\_train **=** train\_df**.**drop("Survived", axis**=**1) Y\_train **=** train\_df["Survived"]

X\_test **=** test\_df**.**drop("PassengerId", axis**=**1)**.**copy()

In [32]:

In [33]:

sgd **=** linear\_model**.**SGDClassifier(max\_iter**=**5, tol**=None**) sgd**.**fit(X\_train, Y\_train)

Y\_pred **=** sgd**.**predict(X\_test) sgd**.**score(X\_train, Y\_train)

acc\_sgd **=** round(sgd**.**score(X\_train, Y\_train) **\*** 100, 2)

In [34]:

random\_forest **=** RandomForestClassifier(n\_estimators**=**100) random\_forest**.**fit(X\_train, Y\_train)

Y\_prediction **=** random\_forest**.**predict(X\_test) random\_forest**.**score(X\_train, Y\_train)

acc\_random\_forest **=** round(random\_forest**.**score(X\_train, Y\_train) **\*** 100, 2)

In [35]:

X, y **=** make\_classification(random\_state**=**42)

X\_train, X\_test, y\_train, y\_test **=** train\_test\_split(X, y, random\_state**=**42) pipe **=** make\_pipeline(StandardScaler(), LogisticRegression()) pipe**.**fit(X\_train, y\_train) *# apply scaling on training data #steps=[('standardscaler', StandardScaler()),*

*# ('logisticregression', LogisticRegression())])*

pipe**.**score(X\_test, y\_test) *# apply scaling on testing data, without leaking trainin*

Out[35]:

In [ ]:

0.96