

IN LAB

LAB TASK 1 :

Getting dataset which is a csv file, into google collab to do some analysis of linear regression.

```
!wget
https://www.dropbox.com/s/veak3ugc4wj9luz/Alumni%20Giving%20Regression%20%28Edited%29.csv?dl=0
```



```
--2023-10-19 10:42:27-- https://www.dropbox.com/s/veak3ugc4wj9luz/Alumni%20Giving%20Regression%20%28Edited%29.csv?dl=0
Resolving www.dropbox.com (www.dropbox.com)... 162.125.6.18, 2620:100:601c:18::a27d:612
Connecting to www.dropbox.com (www.dropbox.com)|162.125.6.18|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: /s/raw/veak3ugc4wj9luz/Alumni%20Giving%20Regression%20%28Edited%29.csv [following]
--2023-10-19 10:42:28-- https://www.dropbox.com/s/raw/veak3ugc4wj9luz/Alumni%20Giving%20Regression%20%28Edited%29.csv
Reusing existing connection to www.dropbox.com:443.
HTTP request sent, awaiting response... 302 Found
Location: https://uc4bd04fec4eb8c4874ad8aaaa5.d1.dropboxusercontent.com/cd/0/inline/CF6fboxLSIkZbk0Zlgh0XVIHcZ0ad0m0dEz-ZitaoCyr-jyokAgu07XEh6fuB1kuvig0
--2023-10-19 10:42:28-- https://uc4bd04fec4eb8c4874ad8aaaa5.d1.dropboxusercontent.com/cd/0/inline/CF6fboxLSIkZbk0Zlgh0XVIHcZ0ad0m0dEz-ZitaoCyr-jyokAgu07XEh6fuB1kuvig0
Resolving uc4bd04fec4eb8c4874ad8aaaa5.d1.dropboxusercontent.com (uc4bd04fec4eb8c4874ad8aaaa5.d1.dropboxusercontent.com)... 162.125.6.15, 2620:100:601c:
Connecting to uc4bd04fec4eb8c4874ad8aaaa5.d1.dropboxusercontent.com (uc4bd04fec4eb8c4874ad8aaaa5.d1.dropboxusercontent.com)|162.125.6.15|:443... connec
HTTP request sent, awaiting response... 200 OK
Length: 3504 (3.4K) [text/plain]
Saving to: 'Alumni Giving Regression (Edited).csv?dl=0'

Alumni Giving Regre 100%[=====] 3.42K --.-KB/s in 0s

2023-10-19 10:42:28 (1.98 GB/s) - 'Alumni Giving Regression (Edited).csv?dl=0' saved [3504/3504]
```

```
from keras.models import Sequential
from keras.layers import Dense, Dropout
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
import numpy as np
from sklearn import linear_model
from sklearn import preprocessing
from sklearn import tree
from sklearn.ensemble import RandomForestRegressor,
GradientBoostingRegressor
import pandas as pd
import csv
```

LAB TASK 2:

Importing important file into the google collab using python commands.

```
from keras.models import Sequential
from keras.layers import Dense, Dropout
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
import numpy as np
from sklearn import linear_model
from sklearn import preprocessing
from sklearn import tree
from sklearn.ensemble import RandomForestRegressor,
GradientBoostingRegressor
import pandas as pd
import csv
```

The screenshot shows a Jupyter Notebook titled 'lab4Regression.ipynb'. The left sidebar displays a file explorer with a folder named 'sample_data' containing three files: 'Alumni Giving Regression (Edited)...', 'Alumni Giving Regression (Edited)...', and 'Alumni Giving Regression (Edited)...'. The main area shows the execution of two code cells. The first cell, labeled [1], displays the output of an HTTP request and a file saving operation: 'HTTP request sent, awaiting response... 200 OK', 'Length: 3504 (3.4K) [text/plain]', 'Saving to: 'Alumni Giving Regression (Edited).csv?dl=0'', and a progress bar for 'Alumni Giving Regre 100%[=====] 3.42K --KB/s in 0s'. The second cell, labeled [4], contains a list of imports for machine learning libraries: 'from keras.models import Sequential', 'from keras.layers import Dense, Dropout', 'from sklearn.metrics import classification_report, confusion_matrix', 'from sklearn.model_selection import train_test_split', 'from sklearn.metrics import mean_squared_error', 'import numpy as np', 'from sklearn import linear_model', 'from sklearn import preprocessing', 'from sklearn import tree', 'from sklearn.ensemble import RandomForestRegressor, GradientBoostingRegressor', 'import pandas as pd', and 'import csv'.

LAB TASK 3:

Displaying some rows and columns of data which are at the starting of the file.

```
np.random.seed(7)
df = pd.read_csv("Alumni Giving Regression (Edited).csv", delimiter =
",")
df.head()
```

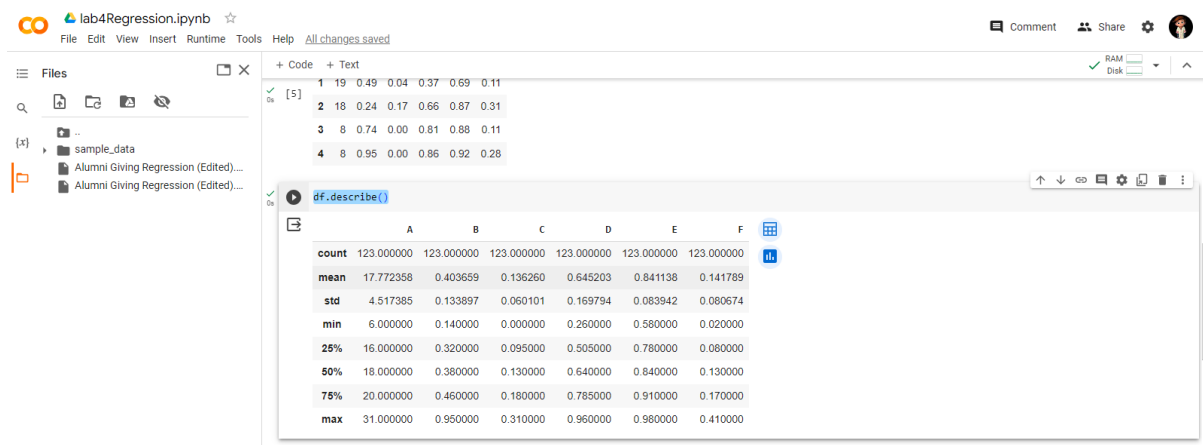
The screenshot shows the same Jupyter Notebook interface. The code cell [4] now includes the execution of 'np.random.seed(7)', 'df = pd.read_csv("Alumni Giving Regression (Edited).csv", delimiter = ",")', and 'df.head()'. Below the code, a preview of the first five rows of the data is displayed in a table format:

	A	B	C	D	E	F
0	24	0.42	0.16	0.59	0.81	0.08
1	19	0.49	0.04	0.37	0.69	0.11
2	18	0.24	0.17	0.66	0.87	0.31
3	8	0.74	0.00	0.81	0.88	0.11
4	8	0.95	0.00	0.86	0.92	0.28

LAB TASK 4:

This command will give us the whole analysis of the file with some parameters like count, mean, minimum, maximum etc

```
df.describe()
```



lab4Regression.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Files

sample_data

Alumni Giving Regression (Edited)...

Alumni Giving Regression (Edited)...

Code

```
1 19 0.49 0.04 0.37 0.69 0.11
2 18 0.24 0.17 0.66 0.87 0.31
3 8 0.74 0.00 0.81 0.88 0.11
4 8 0.95 0.00 0.86 0.92 0.28
```

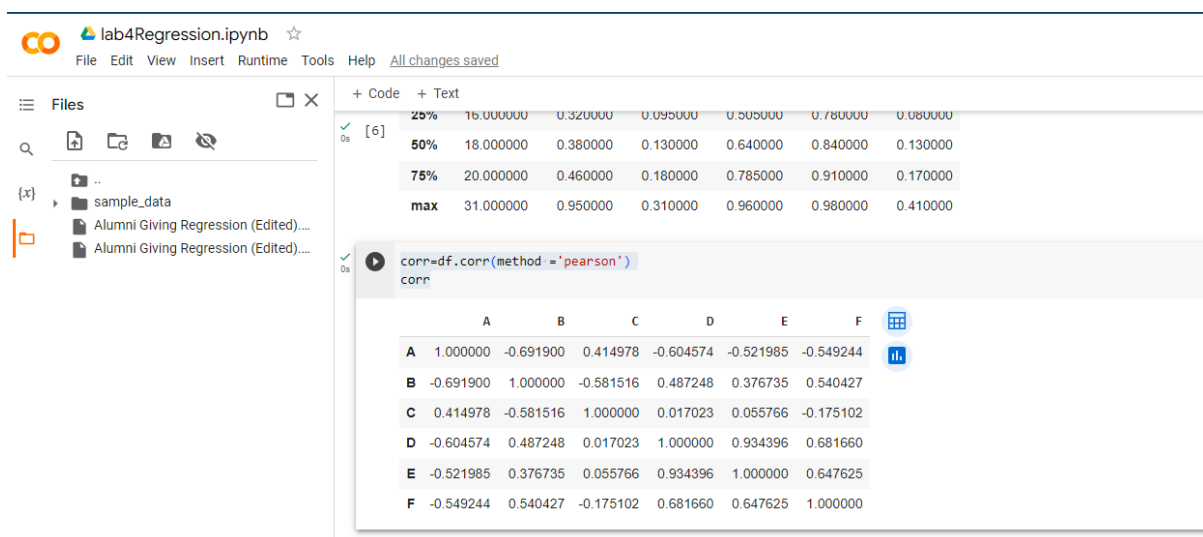
df.describe()

	A	B	C	D	E	F
count	123.000000	123.000000	123.000000	123.000000	123.000000	123.000000
mean	17.772358	0.403659	0.136260	0.645203	0.841138	0.141789
std	4.517385	0.133897	0.060101	0.169794	0.083942	0.080674
min	6.000000	0.140000	0.000000	0.260000	0.580000	0.020000
25%	16.000000	0.320000	0.095000	0.505000	0.780000	0.080000
50%	18.000000	0.380000	0.130000	0.640000	0.840000	0.130000
75%	20.000000	0.460000	0.180000	0.785000	0.910000	0.170000
max	31.000000	0.950000	0.310000	0.960000	0.980000	0.410000

LAB TASK 5:

Here we calculate the Pearson correlation coefficients between the columns (variables) in a DataFrame and store the resulting correlation matrix in the variable corr.

```
corr=df.corr(method='pearson')
corr
```



lab4Regression.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Files

sample_data

Alumni Giving Regression (Edited)...

Alumni Giving Regression (Edited)...

Code

```
corr=df.corr(method='pearson')
corr
```

	A	B	C	D	E	F
A	1.000000	-0.691900	0.414978	-0.604574	-0.521985	-0.549244
B	-0.691900	1.000000	-0.581516	0.487248	0.376735	0.540427
C	0.414978	-0.581516	1.000000	0.017023	0.055766	-0.175102
D	-0.604574	0.487248	0.017023	1.000000	0.934396	0.681660
E	-0.521985	0.376735	0.055766	0.934396	1.000000	0.647625
F	-0.549244	0.540427	-0.175102	0.681660	0.647625	1.000000

LAB TASK 6:

These lines of code prepare data for regression modeling by selecting the independent features (predictors) and the target variable

```
Y_POSITION = 5
model_1_features = [i for i in range(0,Y_POSITION)]
X = df.iloc[:,model_1_features]
Y = df.iloc[:,Y_POSITION]
```

```
X_train, X_test, y_train, y_test = train_test_split(X, Y,
test_size=0.20, random_state=2020)
```

lab4Regression.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Files

sample_data

Alumni Giving Regression (Edited)...

Alumni Giving Regression (Edited)...

Code

[7] `corr=df.corr(method='pearson')`
`corr`

	A	B	C	D	E	F
A	1.000000	-0.691900	0.414978	-0.604574	-0.521985	-0.549244
B	-0.691900	1.000000	-0.581516	0.487248	0.376735	0.540427
C	0.414978	-0.581516	1.000000	0.017023	0.055766	-0.175102
D	-0.604574	0.487248	0.017023	1.000000	0.934396	0.681660
E	-0.521985	0.376735	0.055766	0.934396	1.000000	0.647625
F	-0.549244	0.540427	-0.175102	0.681660	0.647625	1.000000

[15] `Y_POSITION = 5`
`model_1_features = [i for i in range(0, Y_POSITION)]`
`X = df.iloc[:, model_1_features]`
`Y = df.iloc[:, Y_POSITION]`
`X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.20, random_state=2020)`

[15] `model1 = linear_model.LinearRegression()`

LAB TASK 6:

This is the main model. In this code, a linear regression model is trained using the training data and evaluated on both the training and testing datasets. It calculates and reports the Root Mean Squared Error (RMSE), a measure of the model's predictive performance. Additionally, it extracts and displays the coefficients of the features, indicating their influence on the target variable in the linear regression model.

```
model1 = linear_model.LinearRegression()
model1.fit(X_train, y_train)
y_pred_train1 = model1.predict(X_train)
print("Regression")
print("=====")
RMSE_train1 = mean_squared_error(y_train, y_pred_train1)
print("Regression Train set: RMSE {}".format(RMSE_train1))
print("=====")
y_pred1 = model1.predict(X_test)
RMSE_test1 = mean_squared_error(y_test, y_pred1)
print("Regression Test set: RMSE {}".format(RMSE_test1))
print("=====")

coef_dict = {}
for coef, feat in zip(model1.coef_, model_1_features):
    coef_dict[df.columns[feat]] = coef
print(coef_dict)
```

The screenshot shows a Jupyter Notebook interface with the following components:

- File Explorer (Left):** Displays the file structure, including 'sample_data', 'Alumni Giving Regression (Edited)...', and 'Alumni Giving Regression (Edited)...'.
- Code Editor (Center):** Contains the following Python code:


```

V_POSITION = 5
model_1_features = [i for i in range(0, V_POSITION)]
X = df.iloc[:, model_1_features]
Y = df.iloc[:, V_POSITION]
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.20, random_state=2020)

model1 = linear_model.LinearRegression()
model1.fit(X_train, y_train)
y_pred_train1 = model1.predict(X_train)
print("Regression")
print("=====")
RMSE_train1 = mean_squared_error(y_train, y_pred_train1)
print("Regression Train set: RMSE: {}".format(RMSE_train1))
print("=====")
y_pred1 = model1.predict(X_test)
RMSE_test1 = mean_squared_error(y_test, y_pred1)
print("Regression Test set: RMSE: {}".format(RMSE_test1))
print("=====")

coef_dict = {}
for coef, feat in zip(model1.coef_, model_1_features):
    coef_dict[df.columns[feat]] = coef
print(coef_dict)
      
```
- Output (Bottom):** Displays the execution results:


```

Regression
=====
Regression Train set: RMSE 0.002761693322289229
=====
Regression Test set: RMSE 0.004209824026356377
=====
{'A': -0.0009337757382416938, 'B': 0.16012156890162943, 'C': -0.044160015425349614, 'D': 0.15217907817100407, 'E': 0.17539950794101047}
      
```
- Status Bar (Bottom):** Shows 'Disk' usage and '80.85 GB available'.