ADMISSIONS USING LINEAR REGRESSION

Mod 4 Presentation

Lim Zheng Wei

INTRO

A common concern for applicants to university is

- What are the chances of admission to a university?
- What is his probability of admission?
- What are the key factors?

In this exercise, we look at building a model to assess this.

CONTENT

The dataset contains several parameters which are considered important during the application for Masters Programs.
The parameters included are :

1. GRE Scores (out of 340)
2. TOEFL Scores (out of 120)
3. University Rating (out of 5)
4. Statement of Purpose(SOP) and Letter of Recommendation(LOR) Strength (out of 5)
5. Undergraduate GPA (out of 10)
6. Research Experience (either 0 or 1)
7. Chance of Admit(COA) (ranging from 0 to 1)

TRUST THE PROCESS

EDA	• Import data • Check and clean data if necessary
Observe	• Seaborn to plot the variables • Read and identify the predictor and target variables
Feature Selection	• Create test/train model • Find the best predictor variable(s)
Predict	• Given the model we can now predict your odds

EDA

- Import relevant libraries and data

```
import pandas as pd
import numpy as np
import seaborn as sns
import statsmodels.api as sm
import matplotlib.pyplot as plt
from sklearn import linear_model
from sklearn import preprocessing
from sklearn.metrics import mean_squared_error
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.model_selection import KFold
from statsmodels.formula.api import ols
from mpl_toolkits.mplot3d import Axes3D

%matplotlib inline
```

```
admission_csv = 'C:/Users/zheng/Desktop/Data Science/Presentations/Mod 4//admission.csv'

admission = pd.read_csv(admission_csv)
```

- admission.head()

	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
0	337	118	4	4.5	4.5	9.65	1	0.92
1	324	107	4	4.0	4.5	8.87	1	0.76
2	316	104	3	3.0	3.5	8.00	1	0.72
3	322	110	3	3.5	2.5	8.67	1	0.80
4	314	103	2	2.0	3.0	8.21	0	0.65

EDA

- admission.describe()

	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
count	500.000000	500.000000	500.000000	500.000000	500.00000	500.000000	500.000000	500.00000
mean	316.472000	107.192000	3.114000	3.374000	3.48400	8.576440	0.560000	0.72174
std	11.295148	6.081868	1.143512	0.991004	0.92545	0.604813	0.496884	0.14114
min	290.000000	92.000000	1.000000	1.000000	1.00000	6.800000	0.000000	0.34000
25%	308.000000	103.000000	2.000000	2.500000	3.00000	8.127500	0.000000	0.63000
50%	317.000000	107.000000	3.000000	3.500000	3.50000	8.560000	1.000000	0.72000
75%	325.000000	112.000000	4.000000	4.000000	4.00000	9.040000	1.000000	0.82000
max	340.000000	120.000000	5.000000	5.000000	5.00000	9.920000	1.000000	0.97000

EDA

- admission.isnull().sum()

```
GRE Score            0
TOEFL Score          0
University Rating    0
SOP                  0
LOR                  0
CGPA                 0
Research             0
Chance of Admit      0
dtype: int64
```

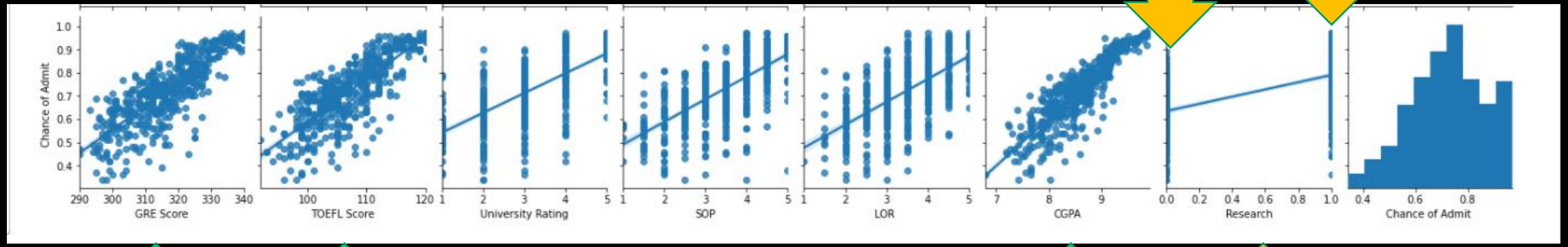
OBSERVE

- Select our predictor, target variables
- Y = admission[‘Chance of Admit‘]
- predictor_columns = [c for c in admission.columns if c != 'Chance of Admit ']
- X = pd.DataFrame(admission, columns = predictor_columns)

- admission.corr()

	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
GRE Score	1.000000	0.827200	0.635376	0.613498	0.524679	0.825878	0.563398	0.810351
TOEFL Score	0.827200	1.000000	0.649799	0.644410	0.541563	0.810574	0.467012	0.792228
University Rating	0.635376	0.649799	1.000000	0.728024	0.608651	0.705254	0.427047	0.690132
SOP	0.613498	0.644410	0.728024	1.000000	0.663707	0.712154	0.408116	0.684137
LOR	0.524679	0.541563	0.608651	0.663707	1.000000	0.637469	0.372526	0.645365
CGPA	0.825878	0.810574	0.705254	0.712154	0.637469	1.000000	0.501311	0.882413
Research	0.563398	0.467012	0.427047	0.408116	0.372526	0.501311	1.000000	0.545871
Chance of Admit	0.810351	0.792228	0.690132	0.684137	0.645365	0.882413	0.545871	1.000000

Recall corr value is 0.545871



- High correlation values among predictor variables will cause inprecise estimates because they interfere with one another

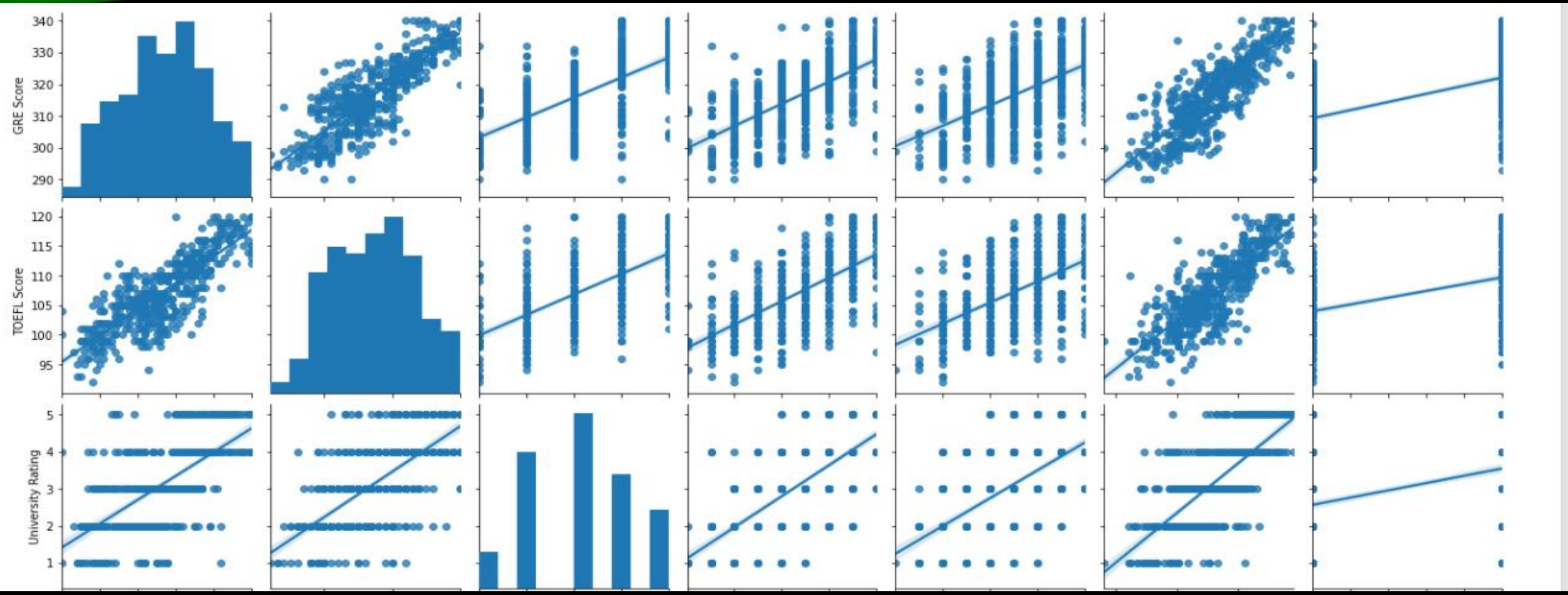
Degree of correlation:

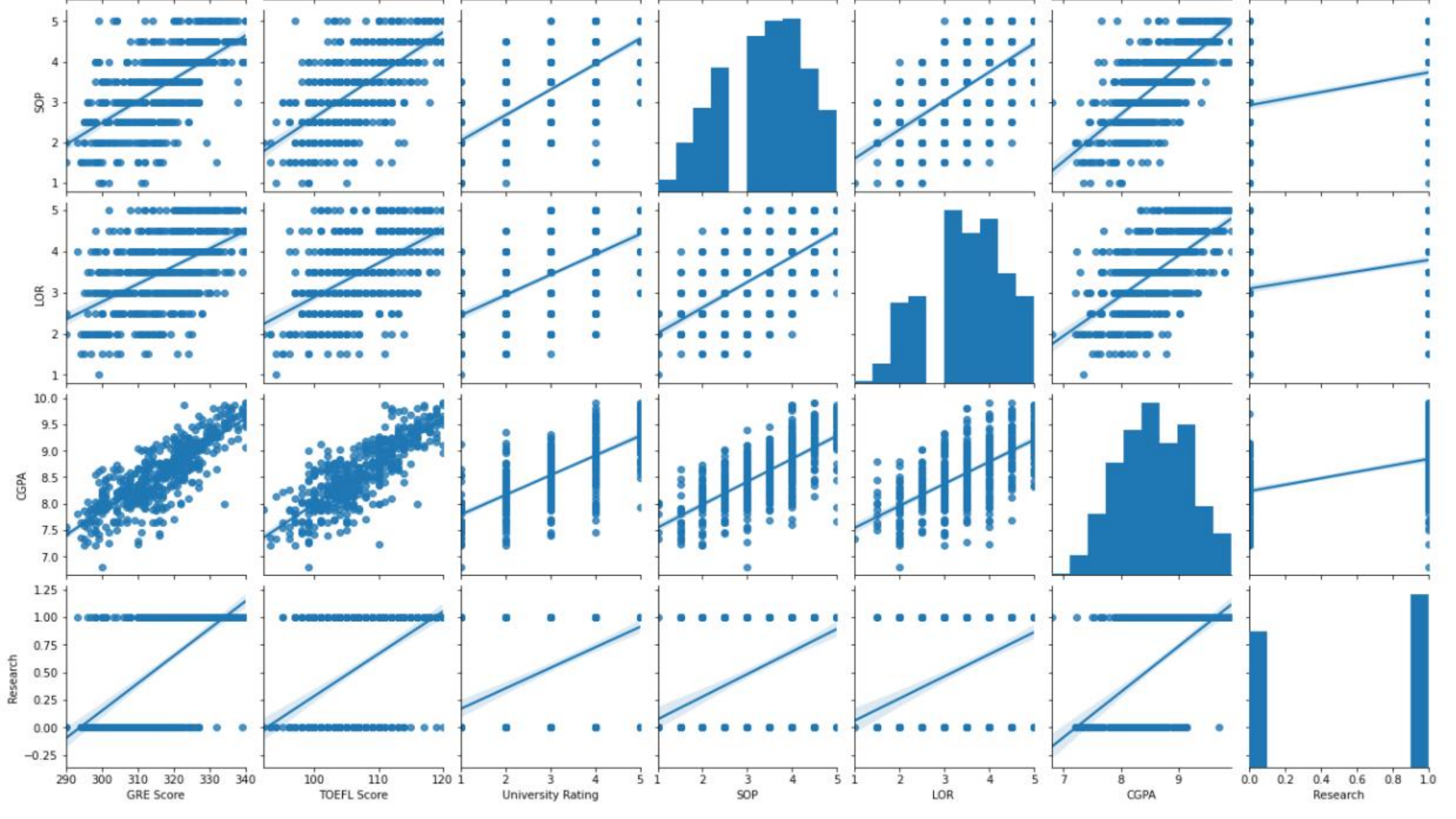
- Perfect: 1
- High : Between ± 0.50 and ± 1 i.e strong correlation.
- Moderate : Between ± 0.30 and ± 0.49 i.e medium correlation.
- Low : below ± .29 i.e small correlation.

- X.corr()

	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research
GRE Score	1.000000	0.827200	0.635376	0.613498	0.524679	0.825878	0.563398
TOEFL Score	0.827200	1.000000	0.649799	0.644410	0.541563	0.810574	0.467012
University Rating	0.635376	0.649799	1.000000	0.728024	0.608651	0.705254	0.427047
SOP	0.613498	0.644410	0.728024	1.000000	0.663707	0.712154	0.408116
LOR	0.524679	0.541563	0.608651	0.663707	1.000000	0.637469	0.372526
CGPA	0.825878	0.810574	0.705254	0.712154	0.637469	1.000000	0.501311
Research	0.563398	0.467012	0.427047	0.408116	0.372526	0.501311	1.000000

- Predict: Likely not all are key variables, Research could be one of the factors


GRE Score
TOEFL Score
University Rating
340
330
320
310
300
290
120
115
110
105
100
95
5
4
3
2
1
GRE Score
TOEFL Score
University Rating
SOP
LOR
CGPA
Research


SOP
LOR
CGPA
Research
GRE Score
TOEFL Score
University Rating
SOP
LOR
CGPA
Research

FEATURE SELECTION

- model1 =
- print(mo

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    COA   R-squared:                       0.657
Model:                            OLS   Adj. R-squared:                  0.656
Method:                 Least Squares   F-statistic:                     952.5
Date:                Sat, 15 Aug 2020   Prob (F-statistic):          1.09e-117
Time:                        22:38:44   Log-Likelihood:                 537.30
No. Observations:                 500   AIC:                            -1071.
Df Residuals:                     498   BIC:                            -1062.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept     -2.4828      0.104    -23.896      0.000      -2.687      -2.279
GRE            0.0101      0.000     30.862      0.000       0.009       0.011
==============================================================================
Omnibus:                       61.111   Durbin-Watson:                   0.904
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               90.883
Skew:                          -0.822   Prob(JB):                     1.84e-20
Kurtosis:                       4.288   Cond. No.                     8.89e+03
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 8.89e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Recall: t = coef/std error

RECAP

You want all

- High R2 score
- High t-score
- Low P > |t| score
- The P value is the probability of seeing a result as extreme as the one you are getting (a t value as large as yours) in a collection of random data in which the variable had no effect

Hypothesis Testing

Two-tailed

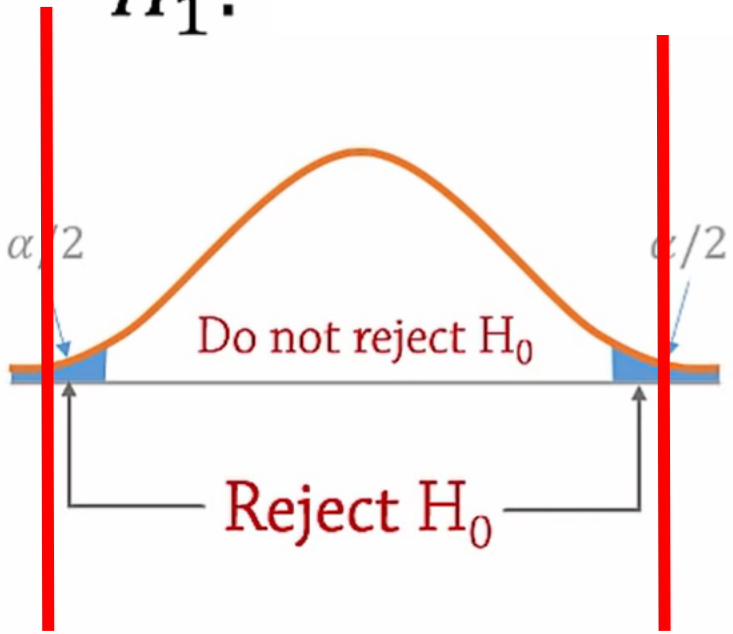
One-tailed

Left-tailed

Right-tailed

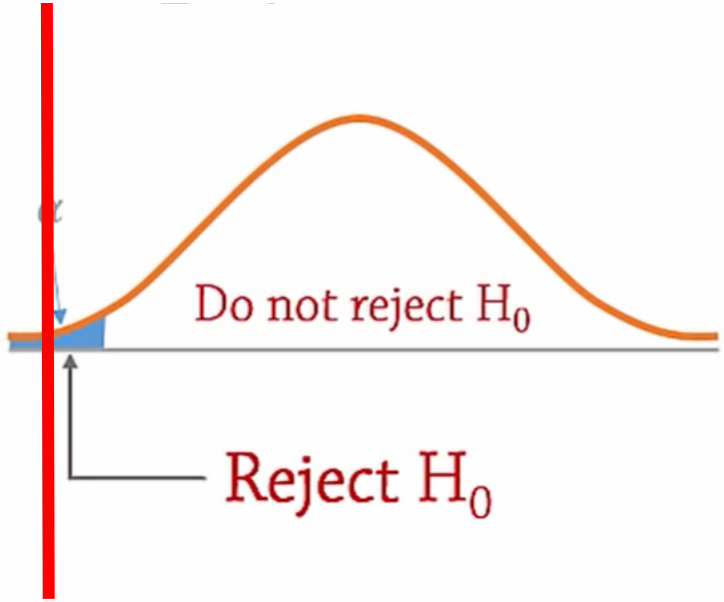
H_0: The variable has no effect on Y

H_1: The variable has an effect on Y

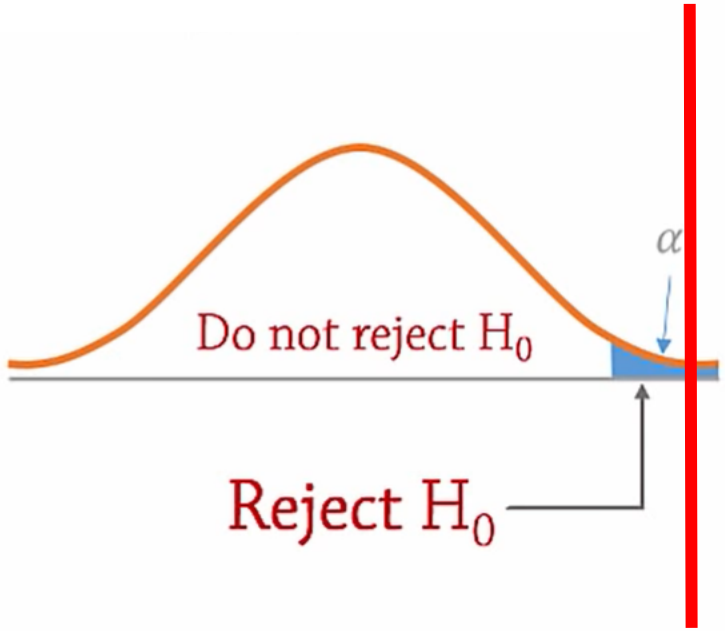


P < -t

P> t



P < -t



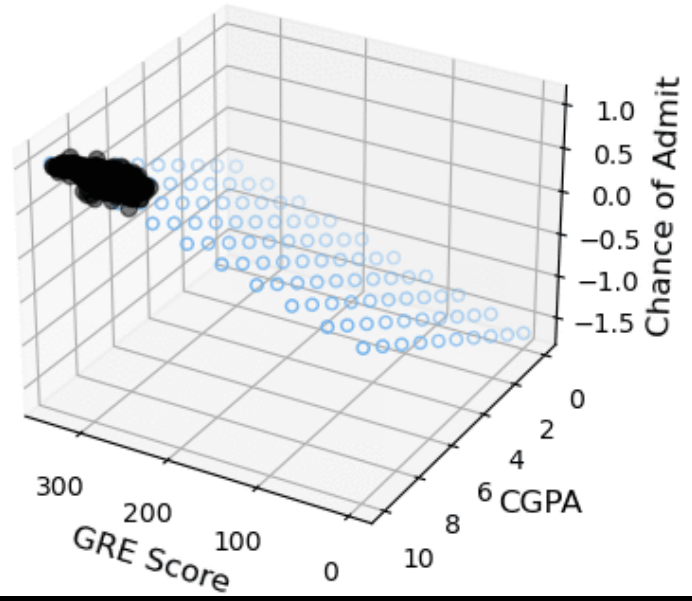
P> t

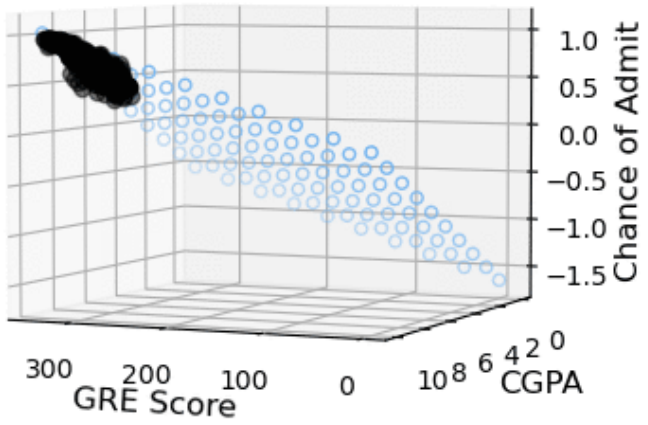
RESULTS AGAINST CHANCE OF ADMIT

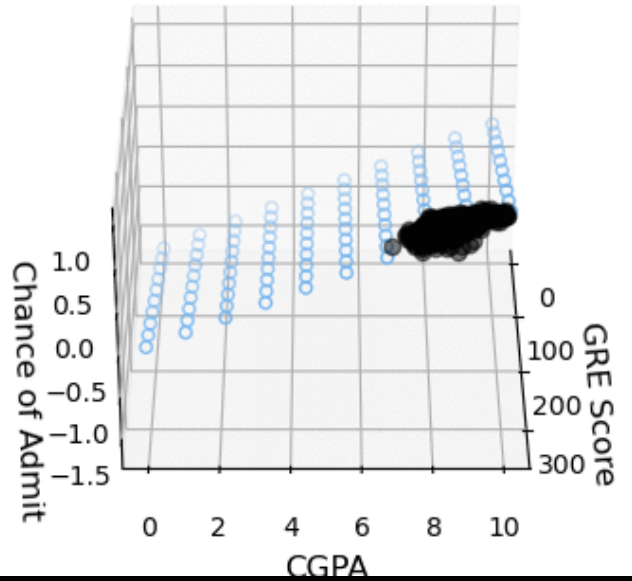
	GRE Score	TOEFL Score	Uni Ranking	SOP	LOR	CGPA	Research
R2 Value	0.657	0.628	0.476	0.468	0.416	0.779	0.298
Adjusted R2 Value	0.656	0.627	0.475	0.467	0.415	0.778	0.297
Coeff	0.101	0.0184	0.0852	0.0974	0.0984	0.2059	0.1551
Std Error	0.000	0.001	0.004	0.005	0.005	0.005	0.011
T-value	30.862	28.972	21.281	20.932	18.854	41.855	14.539
P > |t|	0.000	0.000	0.000	0.000	0.000	0.000	0.000

STRONG FEATURES

$R^2 = 0.80$

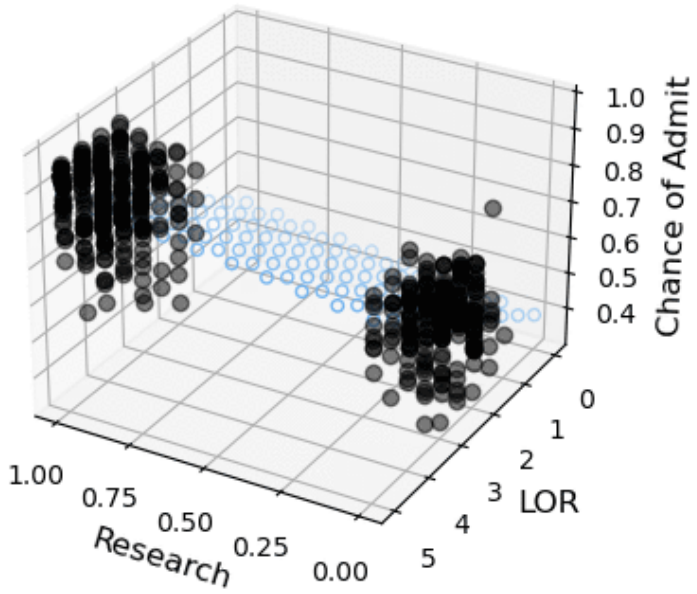


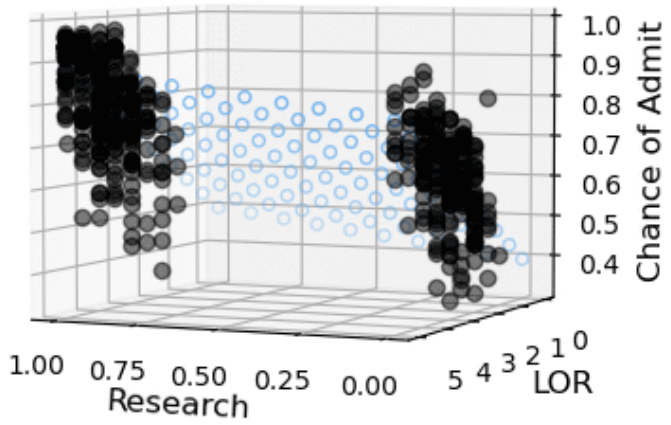


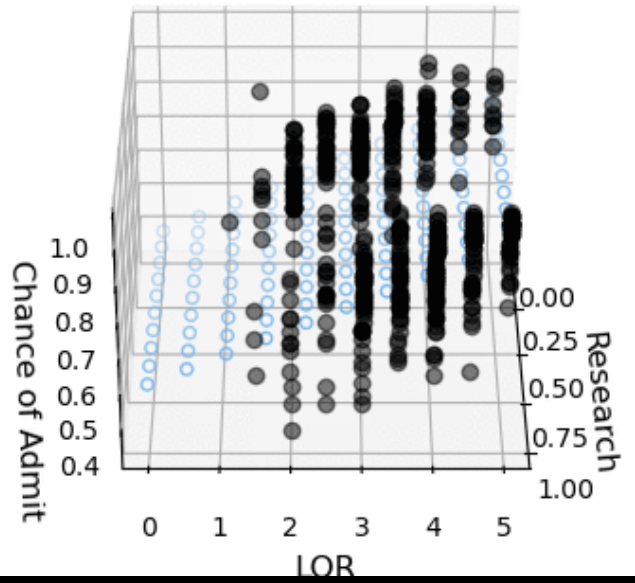


WEAK FEATURES

$R^2 = 0.52$







- Trial and Error is too tiring
- Total possible ways:
- $6C1 + 6C2 + 6C3 + 6C4 + 6C5 + 6C6 = 6 + 15 + 20 + 15 + 6 + 1 = 63$ ways

- Used forward selection model

```
Added feature CGPA with R^2 = 0.771 and adjusted R^2 = 0.769
Added feature GRE Score with R^2 = 0.800 and adjusted R^2 = 0.796
Added feature LOR  with R^2 = 0.811 and adjusted R^2 = 0.805
Added feature TOEFL Score with R^2 = 0.816 and adjusted R^2 = 0.808
Added feature Research with R^2 = 0.821 and adjusted R^2 = 0.811

Resulting features:
CGPA, GRE Score, LOR, TOEFL Score, Research
```

Q: Does transformation help improve the fit?

My guess is not much

OLS Regression Results

Dep. Variable:	COA	R-squared:	0.821
Model:	OLS	Adj. R-squared:	0.819
Method:	Least Squares	F-statistic:	452.1
Date:	Sat, 15 Aug 2020	Prob (F-statistic):	9.97e-182
Time:	23:43:18	Log-Likelihood:	699.65
No. Observations:	500	AIC:	-1387.
Df Residuals:	494	BIC:	-1362.
Df Model:	5		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-1.3357	0.099	-13.482	0.000	-1.530	-1.141
CGPA	0.1230	0.009	13.221	0.000	0.105	0.141
GRE	0.0019	0.001	3.760	0.000	0.001	0.003
TOEFL	0.0030	0.001	3.501	0.001	0.001	0.005
LOR	0.0193	0.004	5.092	0.000	0.012	0.027
Research	0.0252	0.007	3.814	0.000	0.012	0.038

Omnibus:	109.027	Durbin-Watson:	0.800
Prob(Omnibus):	0.000	Jarque-Bera (JB):	248.874
Skew:	-1.130	Prob(JB):	9.07e-55
Kurtosis:	5.615	Cond. No.	1.23e+04

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.23e+04. This might indicate that there are strong multicollinearity or other numerical problems.

Normal

OLS Regression Results

Dep. Variable:	COA	R-squared:	0.821
Model:	OLS	Adj. R-squared:	0.819
Method:	Least Squares	F-statistic:	452.1
Date:	Sat, 15 Aug 2020	Prob (F-statistic):	9.97e-182
Time:	23:43:18	Log-Likelihood:	699.65
No. Observations:	500	AIC:	-1387.
Df Residuals:	494	BIC:	-1362.
Df Model:	5		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-1.3357	0.099	-13.482	0.000	-1.530	-1.141
CGPA	0.1230	0.009	13.221	0.000	0.105	0.141
GRE	0.0019	0.001	3.760	0.000	0.001	0.003
TOEFL	0.0030	0.001	3.501	0.001	0.001	0.005
LOR	0.0193	0.004	5.092	0.000	0.012	0.027
Research	0.0252	0.007	3.814	0.000	0.012	0.038

Omnibus:	109.027	Durbin-Watson:	0.800
Prob(Omnibus):	0.000	Jarque-Bera (JB):	248.874
Skew:	-1.130	Prob(JB):	9.07e-55
Kurtosis:	5.615	Cond. No.	1.23e+04

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.23e+04. This might indicate that there are
strong multicollinearity or other numerical problems.

Scaled

OLS Regression Results

Dep. Variable:	COA	R-squared:	0.821
Model:	OLS	Adj. R-squared:	0.819
Method:	Least Squares	F-statistic:	452.1
Date:	Sat, 15 Aug 2020	Prob (F-statistic):	9.97e-182
Time:	23:58:44	Log-Likelihood:	699.65
No. Observations:	500	AIC:	-1387.
Df Residuals:	494	BIC:	-1362.
Df Model:	5		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	0.7217	0.003	268.651	0.000	0.716	0.727
CGPA_scale	0.0743	0.006	13.221	0.000	0.063	0.085
GRE_scale	0.0213	0.006	3.760	0.000	0.010	0.032
TOEFL_scale	0.0183	0.005	3.501	0.001	0.008	0.029
LOR_scale	0.0179	0.004	5.092	0.000	0.011	0.025
Research_scale	0.0125	0.003	3.814	0.000	0.006	0.019

Omnibus:	109.027	Durbin-Watson:	0.800
Prob(Omnibus):	0.000	Jarque-Bera (JB):	248.874
Skew:	-1.130	Prob(JB):	9.07e-55
Kurtosis:	5.615	Cond. No.	4.77

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

CROSS VALIDATION

```
# Create linear regression object
reg = LinearRegression()
```

```
# Set up 5-fold cross validation
k_fold = KFold(5, shuffle=True)
train_scores = []
train_rmse = []
test_scores = []
test_rmse = []

for k, (train, test) in enumerate(k_fold.split(X)): # Get training and test sets for X1 and y
    print("Train:", train, "Validation:",test)
    X1_train, X1_test = X1.iloc[train], X1.iloc[test]
    Y_train, Y_test = Y.iloc[train],  Y.iloc[test]

    reg.fit(X1_train, Y_train) # Fit model with training set

    y_pred_train = reg.predict(X1_train) # Make predictions with training and test set
    y_pred_test = reg.predict(X1_test)

    train_rmse.append(mean_squared_error(Y_train, y_pred_train, squared=False))    # Score R2 and RMSE on training and test sets
    test_rmse.append(mean_squared_error(Y_test, y_pred_test, squared=False))
    train_scores.append(reg.score(X1_train, Y_train))
    test_scores.append(reg.score(X1_test, Y_test))
```

	R2 Test Scores	RMSE Test Scores	R2 Train Scores	RMSE Train Scores
0	0.858612	0.051204	0.810446	0.061830
1	0.772919	0.067273	0.831216	0.057895
2	0.774110	0.069819	0.833499	0.056916
3	0.829672	0.053878	0.818350	0.061133
4	0.844665	0.058557	0.812892	0.060144

	R2 Test Scores	RMSE Test Scores	R2 Train Scores	RMSE Train Scores
count	5.000000	5.000000	5.000000	5.000000
mean	0.815996	0.060146	0.821281	0.059584
std	0.040110	0.008157	0.010540	0.002105
min	0.772919	0.051204	0.810446	0.056916
25%	0.774110	0.053878	0.812892	0.057895
50%	0.829672	0.058557	0.818350	0.060144
75%	0.844665	0.067273	0.831216	0.061133
max	0.858612	0.069819	0.833499	0.061830

- X1 = admission[['GRE', 'TOEFL', 'LOR', 'CGPA', 'Research',]]
- X1.head()

	GRE	TOEFL	LOR	CGPA	Research
0	337	118	4.5	9.65	1
1	324	107	4.5	8.87	1
2	316	104	3.5	8.00	1
3	322	110	2.5	8.67	1
4	314	103	3.0	8.21	0

CREATE MODEL

- X1_train, X1_test, Y_train, Y_test = train_test_split(X1, Y, test_size=0.2, random_state=42)

```
reg = LinearRegression().fit(X1_train, Y_train) # Train the model using the training sets
print(reg.intercept_)
print(reg.coef_)
reg.score(X1_test, Y_test)
```

RESULTS

Chance of Admit =

0.00219 **GRE** + 0.00316 **TOEFL** + 0.0187 **LOR** + 0.113 **CGPA** + 0.0264 **Research** – 1.355

R2 Score: 0.845

*rounded off to 3 d.p

PREDICT

- E.g GRE score = 300, TOEFL = 100, LOR = 4, CGPA = 8.5 Research = 1

```
X1_pred = np.array([300, 100, 4, 8.5, 1])
X1_pred = X1_pred.reshape(-1, len(X1_pred))

reg.predict(X1_pred)
```

```
array([0.67148219])
```

```
X1 = [[300, 100, 4, 8.5, 1]]

reg.predict(X1)
```

```
array([0.67148219])
```

- GRE = 320, TOEFL = 110, LOR = 0, CGPA = 9.6, Research = 0

```
X1_pred = np.array([320, 110, 0, 9.6, 0])
X1_pred = X1_pred.reshape(-1, len(X1_pred))

reg.predict(X1_pred)
```

```
array([0.7790727])
```

```
X1 = [[320, 110, 0, 9.6, 0]]

reg.predict(X1)
```

```
array([0.7790727])
```

CONCLUSION

- The key factor = CGPA
- Having a good GRE, TOEFL, SOP, LOR and Research help your chances more
- Probability of admission = 0.00219 **GRE** + 0.00316 **TOEFL** + 0.0187 **LOR** + 0.112 **CGPA** + 0.0264 **Research** – 1.355

REFERENCES

- https://www.statisticssolutions.com/pearsons-correlation-coefficient/#:~:text=High%20degree%3A%20If%20the%20coefficient,to%20be%20a%20small%20correlation.
- Lab 4.3
- https://www.youtube.com/watch?v=DIwOTOydeyk
- https://aegis4048.github.io/mutiple_linear_regression_and_visualization_in_python
- https://dss.princeton.edu/online_help/analysis/interpreting_regression.htm