**ABSTRACT:**

This Project Graduate Admissions is all about predicting whether the student will get admission into the school or not .The analysis has been done by exploring the overall structure of the dataset and summary statistics of each variable. Also, analyzed the number of observations, their data type. As the target value is continuous so, I have used Regularized linear Regression( Lasso, Ridge, Enet), Neural Network, Random Forest, Gradient Boosted Trees and Linear Regression to find the results.

**GOAL:**

The objective is to help students to predict graduate admissions from an Indian Perspective. The target value shows chance of admit in a university for a student. Target variable is “Chance of Admit “.It ranges from 0 to 1.The dataset is inspired by the UCLA Graduate Dataset. The dataset is "Admission\_Predict" which is available on Kaggle's website . There are 400 observations of 9 variables. The attributes are  Serial No. , GRE Score, TOEFL Score, University Rating, SOP, LOR, CGPA, Research, Chance of Admit. All the given attributes are numerical. I have dropped the “Serial No.” variable as it is not a useful predictor. Created the factor of “Research” variable as it had binary values.

**RELATED WORK:**

There is one work similar to my project where “Chance of Admit” is the target variable and the objective is to help students to predict graduate admissions from an Indian Perspective.

<https://www.kaggle.com/mahmoud86/graduate-admissions-lasso-ridge-linear-regression>

The dataset is similar as of my dataset. The research variable has been made as a factor. The relationship among the other variables and target variable “Chance of admit” has been done.

Linear Regression, Lasso and Ridge has been used in the example to predict the “Chance of Admit”.

Ridge RMSE : 0.00340927467708808

Lasso RMSE : 0.00378190883711233

Linear Regression RMSE: 0.06761906

As per this example, Ridge has the lowest RMSE.

**DATA EXPLORATION AND PREPROCESSING:**

Correlation between “Chance of Admit” and the other predictors was examined . Side-by-side box plots and t-tests was used for categorical variables; a scatterplot and correlation matrix was used for numeric variables.

Chance of Admit has the highest correlation with CGPA as correlation coefficient between them is 0.87 and also, correlation coefficient with GRE Score is 0.80. After those two , it has fair correlation with TOEFL Score as correlation coefficient between them is 0.79.

Also, to confirm the correlation, we used scatterplot and we can see that linear dependency is there between Chance of Admit and CGPA, Chance of Admit and TOEFL Score. Chance of Admit and GRE Score also has linear dependency.

I did Side by side plot and t-test to find relation between categorical and numerical value. By doing Side by side plot test we see there is difference in the mean of Chance of Admit by Research. To confirm we did t-test and as p value is less than the significant value. We can say there is a relation between Chance of Admit and Research.

There were no missing values in my dataset. I used complete.cases to see the complete cases. All cases are complete. My dataset had 9 variables but I dropped “Serial No.” as it is not a useful predictor.

In Neural Network Model, to scale the attributes selectively, the scaled attributes and non-scaled attributes are stored in separate objects then combined using the cbind() function. This process is repeated for the test and validation sets using the normalization parameters from the training set.

**DATA ANALYSIS AND EXPERIMENTAL RESULTS:**

I used the following models:

* Regression Linear Model:

1.Lasso: A Lasso Linear Regression model is trained using 10-fold cross-validation and the lambda parameter is tuned using a sequence of 100 values between 0.1 and 100.

2.Ridge: A Ridge Regression Model is trained using 10-fold cross-validation and the lambda parameter is tuned using a sequence of 100 values between 0.1 and 100.

3.Enet: An Elastic Net Linear Regression Model is trained using 10-fold cross-validation. The alpha parameter is tuned using a sequence of 10 values between 0 and 1. The lambda parameter is tuned using a sequence of 100 values between 0.1 and 100.

* Neural Network: Used one validation set  and manually try out different values for hyper-parameters  and picked the combination which gave the best performance on the validation set.
* Random Forest: A random forest model is created using the Caret package. The model is auto-tuned using 10-fold cross-validation.
* GBM: A Gradient Boosted Tree is auto tuned using 10-fold cross-validation.
* Linear Regression: It is auto tuned using 10-fold cross-validation.
* *Predicting “ Chance of admit” using Regression Linear Model*

The dataset is partitioned into test(Admission\_Predict\_test) and train(Admission\_Predict\_train) data. The predictions are made on “Admission\_Predict\_test” data using lasso, ridge, enet.

RMSE:

Lasso: 0.06517812

Ridge: 0.07056521

Enet: 0.07056521

* *Predicting “Chance of Admit” Using Artificial Neural Networks*

The “Admission\_Predict\_train” dataset is split into 90% training and 10% validation sets. Tried various values for hyper parameters.

Model1 : An initial neural network is fitted using one hidden layer with 15 neurons, a drop out factor of .5, a mini-batch size of 50, and the Adam optimizer algorithm.

Model2: Number of neurons 20 and mini-batch size of 50.

Model 3: number of neurons to 50 and mini-batch size of 50.

Model4: 20 neurons and a mini-batch size of 100

Model5: 50 neurons and a mini-batch size of 100

RMSE Values:

Model1: 0.06393291

Model2: 0.06595817

Model3: 0.05839776

Model4: 0.06832892

Model5: 0.06105439

Least value of RMSE( 0.05839776) was of model3 so, I used that to find chance of admit values for Admission\_Predict\_test. The RMSE came 0.07151191. I added an extra layer to see if there was any improvement but there was no improvement in the performance.

In general though, to make conclusion about the hyper-parameters of neural network, we need to run each model several times and then look at the average performance. We ran the best model(model3) several times and the RMSE was 0.07578 .

* *Predicting Chance of Admit Using Random Forest and Gradient Boosted Trees:*

The “Admission\_Predict\_train” and “Admission\_Predict\_test” sets are recreated to undo the changes made to the variables in the previous section.

RMSE:

Random Forest: 0.07007131

GBM: 0.06721274

I also used , varImp() used to get the variable importance rankings:

CGPA

`GRE Score`

Research1

`TOEFL Score`

`University Rating`

LOR

SOP

* *Predicting Chance of Admit Using Linear Regression:*

From the linear regression model result below, we can say that variables CGPA and LOR have very statistically significant effect on the outcome as their p value is between 0 and 0.001.

Linear Regression

362 samples

7 predictor

No pre-processing

Resampling: Cross-Validated (10 fold)

Summary of sample sizes: 325, 325, 327, 325, 326, 327, ...

Resampling results:

RMSE Rsquared MAE

0.06344847 0.7970136 0.04605844

Tuning parameter 'intercept' was held constant at a value of TRUE

Call:

lm(formula = .outcome ~ ., data = dat)

Residuals:

Min 1Q Median 3Q Max

-0.26379 -0.02245 0.01022 0.03608 0.15865

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -1.243422 0.132008 -9.419 < 2e-16 \*\*\*

`\\`GRE Score\\`` 0.001787 0.000624 2.864 0.004433 \*\*

`\\`TOEFL Score\\`` 0.002440 0.001125 2.169 0.030760 \*

`\\`University Rating\\`` 0.005966 0.004916 1.213 0.225754

SOP -0.001982 0.005751 -0.345 0.730619

LOR 0.021850 0.005739 3.807 0.000166 \*\*\*

CGPA 0.120930 0.012833 9.424 < 2e-16 \*\*\*

Research1 0.023823 0.008357 2.851 0.004619 \*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.06368 on 354 degrees of freedom

Multiple R-squared: 0.8006, Adjusted R-squared: 0.7967

F-statistic: 203 on 7 and 354 DF, p-value: < 2.2e-16

[1] 0.06497345

* *Compared the cross validation error of all models using “resamples” except for NN model.*

Models: L, R, E, RF, G, LR

Number of resamples: 10

MAE

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

L 0.03236388 0.04369070 0.04522799 0.04597964 0.05015462 0.05626655 0

R 0.04063627 0.04554356 0.05086051 0.04943335 0.05338244 0.05742906 0

E 0.04063627 0.04554356 0.05086051 0.04943335 0.05338244 0.05742906 0

RF 0.03678497 0.04504175 0.04734278 0.04701579 0.05009922 0.05405172 0

G 0.03518413 0.04550091 0.04780267 0.04776030 0.05075082 0.05829829 0

LR 0.03232238 0.04367081 0.04539900 0.04605844 0.05037574 0.05626523 0

RMSE

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

L 0.04174099 0.05845083 0.06397825 0.06336428 0.07203883 0.07503867 0

R 0.04843849 0.06138249 0.06952456 0.06755757 0.07268680 0.08037119 0

E 0.04843849 0.06138249 0.06952456 0.06755757 0.07268680 0.08037119 0

RF 0.04737921 0.06327525 0.06892781 0.06505320 0.07029598 0.07073750 0

G 0.04653415 0.06346101 0.06642888 0.06589791 0.07177888 0.08131939 0

LR 0.04178179 0.05875300 0.06384613 0.06344847 0.07202710 0.07508969 0

Rsquared

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

L 0.6866038 0.7840596 0.8095110 0.7976470 0.8338251 0.8883798 0

R 0.7164219 0.7552235 0.7907375 0.7860081 0.8111622 0.8549721 0

E 0.7164219 0.7552235 0.7907375 0.7860081 0.8111622 0.8549721 0

RF 0.7308889 0.7640821 0.7894545 0.7898480 0.8120174 0.8582328 0

G 0.6650785 0.7703803 0.8038175 0.7828863 0.8142711 0.8616184 0

LR 0.6866363 0.7840804 0.8085245 0.7970136 0.8326733 0.8884973 0

From the comparison of performance from the cross validation error,looking at mean of RMSE, we can say that Ridge and Enet gave the worst cross validation error. Linear regression and Lasso gave the least cross validation error.

* *Calculated the RMSES:*

RMSE

Lasso 0.06517812

Ridge 0.07056521

Enet 0.07056521

NN 0.07151191

RF 0.07007131

GBM 0.06721274

Linear Regression 0.06497345

It can be concluded that Linear Regression model performed the best on this dataset.

**CONCLUSION:**

While processing the data , I found that Neural Networks, Random Forest, GBM are really good models in achieving the state of the art performance. Also, in Neural Networks, the values get changed whenever we run the model. So ,we need to run each model several times and then look at the average performance.

**REFERENCES:**

<https://www.kaggle.com/mohansacharya/graduate-admissions>

The dataset is owned by Mohan S Acharya.