STA 141 Project-Chance of Admit Prediction

Instructor: Peter Kramlinger

Author: Dongmin Wu 918110396, Yifei Zhan 919368587

Contribution:

We began by discussing and deciding on the appropriate dataset to use for the analysis, followed by collaboratively outlining the structure of the report. Then, we worked iteratively on writing, debugging, and analyzing the R code, engaging in multiple rounds of communication to refine logic about model selection. We also drafted initial interpretations and comments in R and Google Docs. Finally, we wrote the report collaboratively, discussing and aligning on the ideas and structure before drafting each paragraph.

I. Introduction:

1.1 Research Question:

As global higher education becomes more competitive, predicting graduate school admissions has become a hot topic. This study mainly focuses on: "What are the key features influencing admissions?" and "Estimate applicants' chances of being admitted using predictive models." After addressing these problems, we can help students better understand what is the probability of being admitted based on their current grades.

1.2 Objective:

This study uses the "Graduate Admissions 2" dataset from Kaggle to build predictive models and analyze how different factors influence the chances of admission. As students about to graduate from college and planning to apply for graduate school, this study is meaningful to me which helps me better understand how influential some key factors are in the admissions process and provides data-driven advice for other applicants as well.

1.3 Overview of Methods:

This study uses various regression models including the Linear Regression Model, Polynomial Regression Model, and Interaction Model. We also used a stepwise regression model by evaluating AIC during each step and PCA model applied to reduce issues with multicollinearity in the data. These models are also compared and evaluated with following methods:

-Cross-Validation (CV): Using CV to assess the model's generalization performance and check overfit.

-Model Comparison: Compare and evaluate different models using metrics such as RSE, MSE, R^2, and AIC/BIC.

II. Data Processing:

2.1 Data Acquisition:

We use the “Graduate Admissions 2” dataset from Kaggle (Graduate Admission 2) consists of 500 entries and includes variables such as:

-GRE Score: GRE scores, ranging from [290, 340].

-TOEFL Score: TOEFL scores, ranging from [92, 120].

-University Rating: University ratings, on a scale of 1 to 5.

-SOP: Ratings for the Statement of Purpose, ranging from 1 to 5.

-LOR: Ratings for the Letters of Recommendation, ranging from 1 to 5.

-CGPA: Undergraduate GPA, ranging from [6.8, 10].

-Research: Research experience (0 for no, 1 for yes).

Target Variable is:

-Chance of Admit: Probability of admission, ranging from [0, 1].

This dataset has strong correlation and comprehensive variables provide us with a good foundation for analyzing chances of admit.

2.2 Data Cleaning:

-Missing values:

We checked for missing values (NA) in the dataset and found none. Thus, no additional steps were required to handle missing data.

-Outliers:

We use visualizations such as boxplots and histograms to examine the distribution and outliers of each variable. The only notable outlier was a value of 1 in the **LOR** variable. Although this is an outlier, we believe it reflects real-world scenarios where some applicants might not have or have poor recommendation letters. Thus, we keep this data point without removing it.

2.3 Data Formatting:

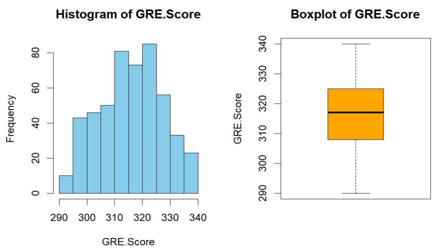
We convert University Ratings and Research into categorical variables to better align with the requirements of model building. By converting these variables into factors, we ensured that statistical modeling and visualization would appropriately treat them as categorical rather than continuous.

Besides, the data was transformed into a long format using the pivot_longer() function from the tidyr package. This transformation makes it easier to perform exploratory analysis and visualize trends for each variable. The long format dataset also fits better for detailed exploration of the relationships between each feature and the "Chance of Admit". This transformation is particularly useful when performing visualizations or comparing distributions across different groups, such as university ratings or research experience.

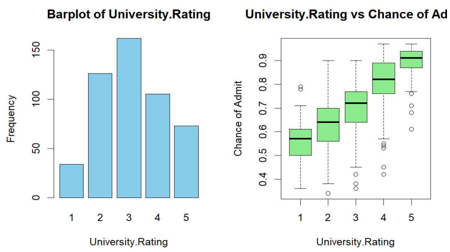
2.4 EDA

-Distribution and skewness:

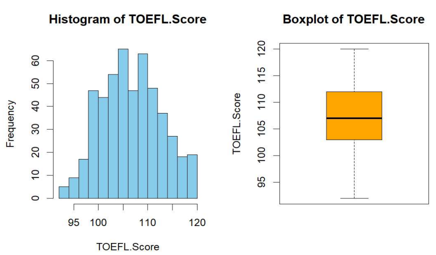
We use histograms and boxplots to visualize the distributions of continuous and categorical variables.



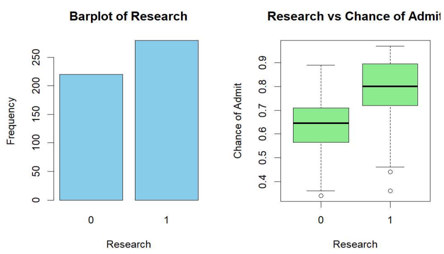
(figure 1)



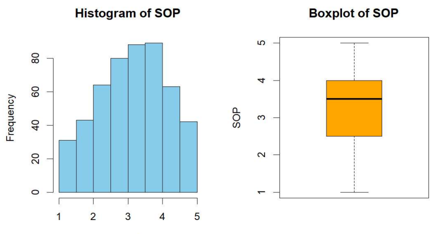
(figure 7)



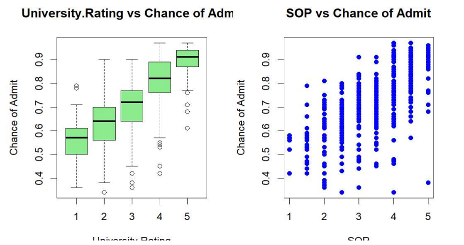
(figure 2)



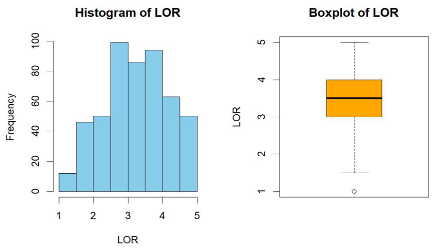
(figure 8)



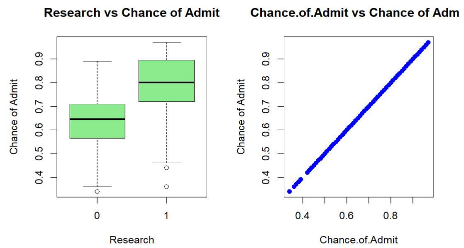
(figure 3)



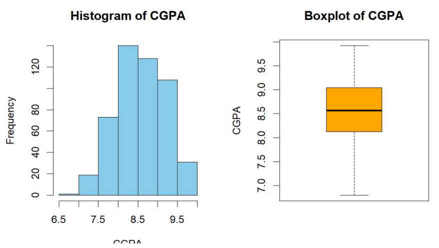
(figure 9)



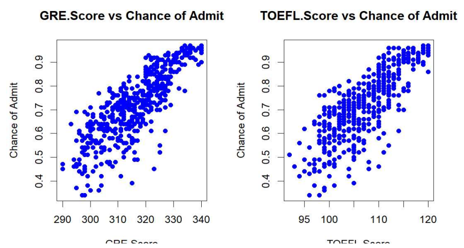
(figure 4)



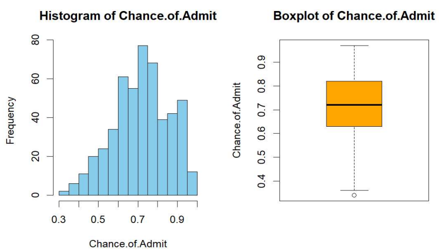
(figure 10)



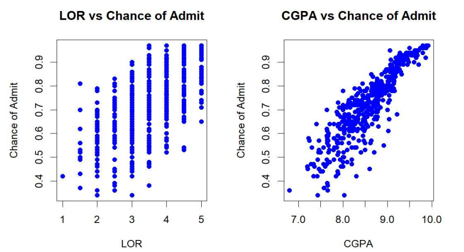
(figure 5)



(figure 11)



(figure 6)



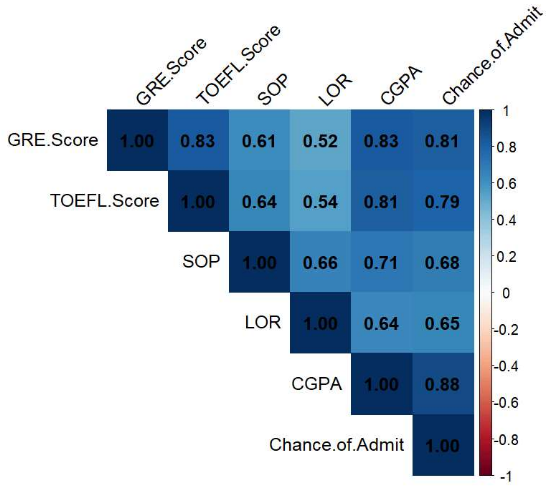
(figure 12)

From the histogram and box plot, we can tell that some variables (e.g., GRE, TOEFL, and CGPA) are close to normal distribution, while others (e.g., SOP and LOR) are slightly right-skewed since the skewness is mild and did not significantly affect model accuracy. We chose not to apply transformations such as log or box-cox transformations.

In terms of box-plot fig1-5, we see there is only one outlier in LOR, however, when traced back to data, the candidate with one LOR is still reasonable in admission. Thus, we decide not to remove this outlier.

For the categorical variable (Research, University Rating), we see a positive correlation with a chance of Admit from (fig 8,9). In addition, in terms of continuous variables (fig 9,11,12), scatter plot also shows a positive correlation with the target variable. Since the scatter plot for GRE, CGPA and TOEFL(fig 9,11,12) may be nonlinear and may require the use of polynomial regression. Therefore, besides linear regression, polynomial regression is considered.

We also plot a correlation matrix for numerical variables and find a relatively high correlation between them.



(Figure 13)

Each independent variable shows a positive correlation with the target dependent variable, the admission rate. Additionally, interaction terms are considered to account for potential relationships between the independent variables. For instance, we hypothesize that TOEFL scores and GRE scores may be highly correlated, as both reflect a student's performance on standardized tests.

III Methodology

We are performing linear regression to predict the target feature “chance of admit”.

As mentioned above in the EDA section, we performed linear regression as our basic model, polynomial regression, and interaction regression. Given the high correlations observed in the correlation matrix, we also applied Principal Component Analysis (PCA) to address multicollinearity and implemented Ridge Regularization to prevent overfitting and improve model robustness.

3.1 Linear Regression:

For the linear regression analysis, we included all variables—GRE Score, TOEFL Score, University Rating, SOP, LOR, CGPA, and Research—as our base model to gain a general understanding of the relationships between the dependent and independent variables. Additionally, during further model selection, we primarily compared the output results with this base model.

Based on the output of the linear regression model, the adjusted R-squared value indicates that approximately 81.98% of the variability in the "Chance of Admit" is explained by the model, suggesting a strong fit. Among all factors, CGPA appears to be the most influential predictor with the highest coefficient value (0.1179), followed by Research and GRE Score. The overall model is highly significant (F-statistic p-value $< 2e-16$), confirming that the predictors collectively have a substantial effect on the chance of admission.

In addition, since our number of independent variables is limited, thus, applying a reduced model might not be efficient in reducing model complexity. Thus, the reduced model is not considered in this case.

3.2 Polynomial Regression

We implemented a polynomial regression model by including the squared terms for GRE, CGPA, and TOEFL, as suggested by the EDA. The scatter plots (Figures 9, 11, and 12) indicated potential non-linear relationships, with the data clustering at higher scores and showing signs of saturation for these variables. Given their high correlation with the "Chance of Admit" (0.81, 0.83, and 0.83 for GRE, CGPA, and TOEFL, respectively), the polynomial model was chosen to better capture these non-linear patterns. This approach allows us to explore whether incorporating these quadratic terms improves the regression analysis and better accounts for the observed data structure.

However, the results indicate that implementing squared coefficients did not lead to any improvement. The adjusted R-squared value is 0.8181, reflecting a good fit, but it remains slightly lower than that of the base model. Additionally, the high p-values for the squared terms suggest that they are not statistically significant in the analysis. This could be attributed to collinearity among the variables, indicating that further steps, such as applying PCA might be needed.

Overall, the polynomial model does not provide additional insights or advantages over the simpler linear model. The significant predictors remain the same, and the inclusion of quadratic terms does not enhance the model's fit or predictive performance.

3.3 Interaction Regression

The interaction model was developed to examine potential relationships between independent variables that could jointly influence the "Chance of Admit." Specifically, it includes interaction terms such as GRE Score × University Rating and CGPA × Research to investigate whether the effect of one variable depends on the level of another. Intuitively, admissions committees may perceive students from higher-rated universities as having stronger academic reputations and integrity, making high GRE scores more convincing. Similarly, students with higher CGPAs are likely to contribute more effectively to research projects compared to those with lower CGPAs. This reasoning suggests the possibility of interaction effects between these variables.

The interaction model reveals that CGPA, TOEFL Score, and LOR (Letter of Recommendation Strength) are the key predictors of the "Chance of Admit," with significant positive effects and p-values of < 2e-16, 0.001176, and 3.42e-05, respectively. However, other variables, including GRE Score, Research, SOP, and all levels of University Rating, as well as the interaction terms (e.g., GRE Score × University Rating, CGPA × Research), are not statistically significant (p-values > 0.05). This suggests that the combined effects of these variables do not meaningfully influence the admission chances in this model. The adjusted R-squared value of 0.8195 indicates that approximately 81.95% of the variability in the "Chance of Admit" is explained, comparable to simpler models like the linear regression model. The residual standard error of 0.05997 also shows no significant improvement in predictive accuracy. Overall, while the interaction model accounts for potential relationships between predictors, it does not enhance the model's explanatory power or provide additional insights beyond the simpler models.

3.4 PCA

PCA was applied due to high correlations among features, which were grouped conceptually into three categories: (PC1, PC2, and PC3). These three might explain students' ability on three dimensions, academic skills and soft skills background, and others. These components were combined with the categorical variables, University Rating and Research, to construct a linear regression model.

The results indicate that PC2, PC3, PC4, PC5, and PC31 are statistically significant predictors ($p < 0.05$), with strong positive coefficients, suggesting they play an important role in explaining the "Chance of Admit." However, PC1 is only marginally significant ($p = 0.095782$). The adjusted R-squared value of 0.553 indicates that the model explains 55.3% of the variability in the admission chances, which is considerably lower than previous models. This suggests that while PCA helps address multicollinearity and reduce dimensionality, it may result in some loss of predictive power compared with our base model.

In summary, PCA provides an effective way to simplify the model by reducing redundancy among highly correlated features, but further refinement may be needed.

3.5 Ridge

Ridge Regression was chosen because of the high correlation observed between the predictor variables, such as TOEFL and GRE scores. This reduces the model's sensitivity to multicollinearity and helps prevent overfitting, ensuring better generalization on new data. Additionally, in the Ridge model, cross-validation was employed to determine the optimal value of the regularization parameter lambda, balancing model complexity and performance. We observed that the Ridge model achieved an adjusted R^2 of 0.8189, indicating a good fit.

3.6 Ridge Optimization

Considering the benefits of Ridge Regression, we applied Ridge optimization to address the issue of multicollinearity in other regression models, such as linear, polynomial, and interaction models. From the results, Ridge Regression maintained a high R^2 while significantly reducing AIC and BIC, demonstrating its ability to simplify the model’s complexity and prevent overfitting.

Then, we think of using Ridge to optimize our PCA model as well.

```
Importance of components:
                          PC1    PC2    PC3     PC4     PC5     PC6
Standard deviation     2.135 0.7871 0.5708 0.46403 0.40586 0.33868
Proportion of Variance 0.760 0.1032 0.0543 0.03589 0.02745 0.01912
Cumulative Proportion  0.760 0.8632 0.9175 0.95343 0.98088 1.00000
PCA Model Metrics:
R2: 0.8617249
MSE: 0.002749016
Residual Standard Error: 0.05243106
AIC: -2944.256
BIC: -2935.827
```

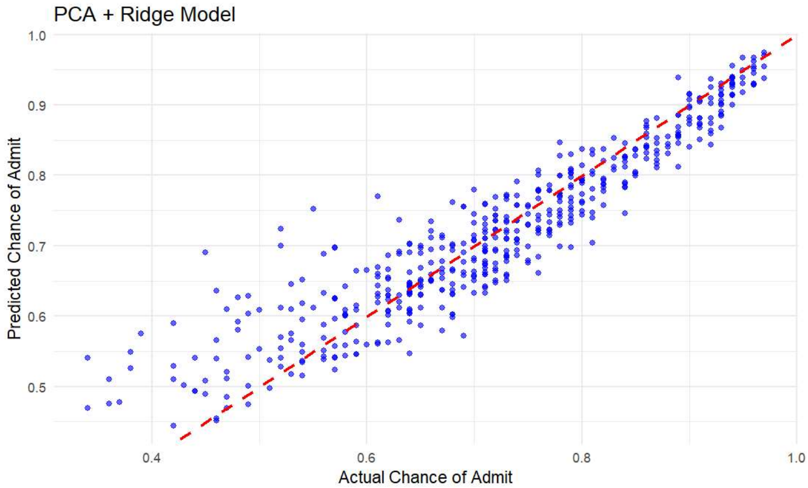
Surprised, we can tell that PC1 explains 76.0% of the variance, while the first four components collectively explain over 95.34% of the total variance. This indicates that most of the information in the data can be captured using a reduced dimensionality of four components.

PCA Ridge Model Performed pretty well. Besides, R^2 is 0.8617, which means the model explains 86.17% of the variance in the target variable, demonstrating excellent predictive power. It also has low MSE, Residual Standard Error (RSE), and AIC/BIC showing strong model fit.

Model <chr>	R2 <dbl>	MSE <dbl>	RSE <dbl>	AIC <dbl>	BIC <dbl>
Linear	0.8233646	0.003511647	0.05992196	-1382.8965	-1332.3212
Polynomial	0.8213479	0.003551741	0.05959649	-1379.2201	-1332.8594
Interaction	0.8249153	0.003480819	0.05899846	-1377.3053	-1305.6570
Ridge	0.8189615	0.003599185	0.05999321	-2799.5240	-2770.0217
Ridge Linear	0.8204914	0.003568770	0.05973918	-2803.7672	-2774.2650
Ridge Polynomial	0.8194925	0.003588629	0.05990517	-2792.9925	-2746.6318
Ridge Interaction	0.8204817	0.003568963	0.05974079	-2785.7402	-2718.3064
PCA Linear	0.5583335	0.008780672	0.09370524	-932.6627	-898.9458
PCA + Ridge	0.8617249	0.002749016	0.05243106	-2944.2562	-2935.8270

IV Model Decision

Therefore, we performed cross-validation to assess whether the model is overfitted, and the results indicated no signs of overfitting. We also draw a scatter plot to visualize the predicted chance of admit with actual chance of admit, and most of them are aligned. Based on this, we finalized the PCA Ridge model as our chosen model due to its excellent performance and ability to balance dimensionality reduction, multicollinearity handling, and prediction accuracy.



PCA + Ridge Model
Predicted Chance of Admit
Actual Chance of Admit
0.5
0.6
0.7
0.8
0.9
1.0
0.4
0.6
0.8
1.0

V Appendix:

````
```{r}
library(ggplot2)
library(dplyr)
library(tidyr)
library(readr)
library(gmodels)
library(GGally)
library(caret)
library(ggcorrplot)
library(car)
library(corrplot)
library(glmnet)
```


```{r}

# load data
data<-read.csv("Admission_Predict_Ver1.1.csv")

#0 na
print(sum(is.na(data)))

#0 duplicate
print(sum(duplicated(data)))

#set random seed for later cross validation
set.seed(42)

# process category variables
data$University.Rating <- as.factor(data$University.Rating)
data$Research <- as.factor(data$Research)

features <- names(data[0,-1])
#features
```

```{r}

for(feature in features) {
  #distinguish continuous and categorical variables
  if (is.numeric(data[[feature]])) {
    #continuous histogram
    hist(data[[feature]],main=paste("Histogram of",feature),xlab=feature,col="skyblue",border="black")
    #boxplot
    boxplot(data[[feature]],main=paste("Boxplot of",feature),ylab=feature,col= "orange")
  }
  else {
    #catego histogram
    barplot(table(data[[feature]]),main=paste("Barplot of",feature),col="skyblue",xlab= feature,ylab="Frequency")
    #boxplot
    boxplot(data$Chance.of.Admit~data[[feature]],main=paste(feature,"vs Chance of Admit"),xlab=feature,ylab="Chance of Admit",col=
"lightgreen")
  }
}

#scatter or box of features vs chance
for(feature in features) {
  if(is.numeric(data[[feature]])){
    #continuous scatter
    plot(data[[feature]],data$Chance.of.Admit, main=paste(feature,"vs Chance of Admit"),xlab= feature,ylab="Chance of Admit",
col="blue",pch=19)#pch fill
  }
  else{
    #catego box
    boxplot(data$Chance.of.Admit~data[[feature]],main=paste(feature,"vs Chance of Admit"),xlab=feature,ylab="Chance of
Admit",col="lightgreen")
  }
}
````
````

```

```{r}
#get continous variables
num_features<-data[,-1] #exclude serial number
num_features<-num_features[,sapply(num_features,is.numeric)]

#correlation matrix
cor_matrix<-cor(num_features)
corrplot(cor_matrix,method="color",type="upper",tl.col="black",tl.srt=45,addCoef.col="black")

```

#Linear Model
```{r}
#basic model
linear_model<-lm(Chance.of.Admit~.,data=data[,-1])
summary(linear_model)
```

#Polynomial Model
```{r}
#polynomial model
poly_model<-lm(Chance.of.Admit~GRE.Score+I(GRE.Score^2)+TOEFL.Score+I(TOEFL.Score^2)+CGPA+I(CGPA^2)+SOP+LOR+Research,data=data[,-1])
summary(poly_model)
```

#Interaction Model
```{r}
inter_model<-lm(Chance.of.Admit~GRE.Score*University.Rating+CGPA*Research+TOEFL.Score+SOP+LOR,data=data[,-1])
summary(inter_model)
```

#PCA model
```{r}
#preprocess to PCA, with 3 components
pca_lm<-preProcess(data[,c("GRE.Score","TOEFL.Score","CGPA","SOP","LOR")],method="pca",pcaComp=3)
pca_lm_data<-predict(pca_lm,data)

# change variable names for later formula calculation
colnames(pca_lm_data)<-c("PC1","PC2","PC3")

#combine pca components and data
pca_cbdata <- cbind(data, pca_lm_data)

#model implement
model_pca<-lm(Chance.of.Admit~PC1+PC2+PC3+University.Rating+Research,data=pca_cbdata)
summary(model_pca)

#statistical value for later model comparison
pca_pred<-predict(model_pca,pca_cbdata)
pca_r2<-summary(model_pca)$r.squared
pca_mse<-mean((data$Chance.of.Admit-pca_pred)^2)
pca_rse<-sqrt(pca_mse)
pca_aic<-AIC(model_pca)
pca_bic<-BIC(model_pca)

#reference: https://rstudio-pubs-static.s3.amazonaws.com/92006_344e916f251146daa0dc49fef94e2104.html?utm_source=chatgpt.com
```

#Ridge Regression Model
```{r}

ridge<-cv.glmnet(as.matrix(data[, -c(1, 9)]),data$Chance.of.Admit,alpha = 0)
ridge_pred<-predict(ridge,s=ridge$lambda.min,newx=as.matrix(data[,-c(1,9)]))

#reference: https://glmnet.stanford.edu/articles/glmnet.html
```
```

````
#PCA Ridge
```{r}
pca_result<-prcomp(num_features,scale.=TRUE)
summary(pca_result)

pca_data<-as.data.frame(pca_result$x[,1:2])
colnames(pca_data)<-c("PC1","PC2")
pca_data$Chance.of.Admit<-data$Chance.of.Admit

X_pca<-as.matrix(pca_data[,-3])
y_pca<-pca_data$Chance.of.Admit
ridge_pca<-cv.glmnet(X_pca,y_pca,alpha=0,nfolds=5)
ridge_pca_pred<-predict(ridge_pca,s=ridge_pca$lambda.min,newx=X_pca)
ridge_pca_mse<-mean((y_pca-ridge_pca_pred)^2)


ridge_pca_r2<-1-sum((y_pca-ridge_pca_pred)^2)/sum((y_pca-mean(y_pca))^2)
ridge_pca_rse<-sqrt(mean((y_pca-ridge_pca_pred)^2))
ridge_pca_mse<-mean((y_pca-ridge_pca_pred)^2)

n_pca<-length(y_pca)  # # samples
p_pca<-ncol(X_pca)  # # pca components
ridge_pca_aic<-n_pca*log(ridge_pca_mse)+2*p_pca
ridge_pca_bic<-n_pca*log(ridge_pca_mse)+log(n_pca)*p_pca

```



#Ridge Linear, Ridge Poly, Ridge Interaction
```{r}

X<-model.matrix(Chance.of.Admit~ . ,data=data[,-1])
y <- data$Chance.of.Admit


ridge_linear<-cv.glmnet(X,y,alpha=0,nfolds=5)


poly_features<-model.matrix(~GRE.Score+I(GRE.Score^2)+TOEFL.Score+I(TOEFL.Score^2)+CGPA+I(CGPA^2)+SOP+LOR+Resear
ch,data=data[,-1])
ridge_poly<-cv.glmnet(poly_features, data$Chance.of.Admit, alpha = 0)


interaction_features<-model.matrix(~GRE.Score*University.Rating+CGPA*Research+TOEFL.Score+SOP+LOR,data=data[,-1])
ridge_interaction<-cv.glmnet(interaction_features, data$Chance.of.Admit, alpha = 0)


```



#Calculate all AIC BIC MSE R2 RSE
```{r}
# Ridge Model Metrics


ridge_r2<-1-sum((y-ridge_pred)^2)/sum((y-mean(y))^2)
ridge_rse<-sqrt(mean((y-ridge_pred)^2))
ridge_mse<-mean((y-ridge_pred)^2)

ridge_coef <- coef(ridge, s = ridge$lambda.min)
n<-length(y)  # # samples
p<-length(ridge_coef)-1  # #coeff remove intercept
ridge_aic<-n*log(mean((y-ridge_pred)^2))+2*p
ridge_bic<-n*log(mean((y-ridge_pred)^2))+log(n)*p

####################################
#linear
linear_pred<-predict(linear_model,data[,-1])

linear_r2<-summary(linear_model)$r.squared
linear_rse<-summary(linear_model)$sigma
linear_mse<-mean((data$Chance.of.Admit-linear_pred)^2)
linear_aic<-AIC(linear_model)
````
````

```
linear_bic<-BIC(linear_model)

########################################
#poly
poly_pred<-predict(poly_model,data)

poly_r2<-1-sum((data$Chance.of.Admit-poly_pred)^2)/sum((data$Chance.of.Admit-mean(data$Chance.of.Admit))^2)
poly_rse<-sqrt(mean((data$Chance.of.Admit-poly_pred)^2))
poly_mse<-mean((data$Chance.of.Admit-poly_pred)^2)
poly_aic<-AIC(poly_model)
poly_bic<-BIC(poly_model)

#########################################
# Interaction Model Metrics
interaction_pred<-predict(inter_model,data[,-1])

interaction_r2<- 1-sum((data$Chance.of.Admit-interaction_pred)^2)/sum((data$Chance.of.Admit-mean(data$Chance.of.Admit))^2)
interaction_rse<-sqrt(mean((data$Chance.of.Admit-interaction_pred)^2))
interaction_mse<-mean((data$Chance.of.Admit-interaction_pred)^2)
interaction_aic<-AIC(inter_model)
interaction_bic<-BIC(inter_model)

#########################################
#Ridge liner
ridge_linear_pred<-predict(ridge_linear,s=ridge_linear$lambda.min,newx = X)

ridge_linear_r2<- 1-sum((data$Chance.of.Admit-ridge_linear_pred)^2)/sum((data$Chance.of.Admit-mean(data$Chance.of.Admit))^2)
ridge_linear_mse<-mean((data$Chance.of.Admit-ridge_linear_pred)^2)
ridge_linear_rse<-sqrt(ridge_linear_mse)

n<-nrow(data)# #samples
p<-ncol(data[,-c(1, 9)])##features
ridge_linear_aic<-n*log(ridge_linear_mse)+2*p
ridge_linear_bic<-n*log(ridge_linear_mse)+log(n)*p
#########################################
#Ridge poly
poly_features<-model.matrix(~GRE.Score+I(GRE.Score^2)+TOEFL.Score+I(TOEFL.Score^2)+CGPA+I(CGPA^2)+SOP+LOR+Resear
ch,data[,-1])

ridge_poly<-cv.glmnet(poly_features,data$Chance.of.Admit, alpha = 0)

ridge_poly_pred<- predict(ridge_poly,s=ridge_poly$lambda.min,newx=poly_features)

ridge_poly_r2<- 1-sum((data$Chance.of.Admit-ridge_poly_pred)^2)/sum((data$Chance.of.Admit-mean(data$Chance.of.Admit))^2)
ridge_poly_mse<-mean((data$Chance.of.Admit-ridge_poly_pred)^2)
ridge_poly_rse<-sqrt(ridge_poly_mse)

p <- ncol(poly_features)##features
ridge_poly_aic<-n*log(ridge_poly_mse)+2*p
ridge_poly_bic<-n*log(ridge_poly_mse)+log(n)*p
#########################################
#Ridge interact
interaction_features<-model.matrix(~GRE.Score*University.Rating+CGPA*Research+TOEFL.Score + SOP + LOR, data)

ridge_interaction<-cv.glmnet(interaction_features,data$Chance.of.Admit,alpha = 0)

ridge_interaction_pred<-predict(ridge_interaction, s = ridge_interaction$lambda.min, newx=interaction_features)


ridge_interaction_r2<-1
-sum((data$Chance.of.Admit-ridge_interaction_pred)^2)/sum((data$Chance.of.Admit-mean(data$Chance.of.Admit))^2)
ridge_interaction_mse<-mean((data$Chance.of.Admit-ridge_interaction_pred)^2)
ridge_interaction_rse<-sqrt(ridge_interaction_mse)

p <- ncol(interaction_features)
ridge_interaction_aic<-n*log(ridge_interaction_mse)+2*p
ridge_interaction_bic<-n*log(ridge_interaction_mse)+log(n)*p
#########################################



results <- data.frame(
  Model = c("Linear", "Polynomial","Interaction","Ridge", "Ridge Linear","Ridge Polynomial","Ridge Interaction","PCA Linear", "PCA
```

```
+ Ridge"),
  R2 = c(linear_r2, poly_r2, interaction_r2, ridge_r2, ridge_linear_r2, ridge_poly_r2, ridge_interaction_r2, pca_r2, ridge_pca_r2),
  MSE = c(linear_mse, poly_mse, interaction_mse, ridge_mse, ridge_linear_mse, ridge_poly_mse, ridge_interaction_mse, pca_mse,
ridge_pca_mse),
  RSE = c(linear_rse, poly_rse, interaction_rse, ridge_rse, ridge_linear_rse, ridge_poly_rse, ridge_interaction_rse, pca_rse,
ridge_pca_rse),
  AIC = c(linear_aic, poly_aic, interaction_aic, ridge_aic, ridge_linear_aic, ridge_poly_aic, ridge_interaction_aic, pca_aic,
ridge_pca_aic),
  BIC = c(linear_bic, poly_bic, interaction_bic, ridge_bic, ridge_linear_bic, ridge_poly_bic, ridge_interaction_bic, pca_bic,
ridge_pca_bic)
)


print(results)
```