# **Kobe Bryant Shot Selection**

Gautam Kapila, Angelo Bravo, Mrinmoy Bhaumik

## **Introduction**

Kobe Bryant was a prolific American professional basketball player, and is considered one of the greatest players in the history of the game. We are provided with the location and circumstances of every shot attempted by Bryant during his 20-year career.

The goal of the project is twofold:

1. Model the data to predict whether the shot was made successfully or not on the test dataset, given the model data set.
2. Provide arguments for or against following propositions
   1. The odds of Kobe making a shot decrease with respect to the distance he is from the hoop
   2. The probability of Kobe making a shot decreases linearly with respect to the distance he is from the hoop
   3. The relationship between the distance Kobe is from the basket and the odds of him making the shot is different if they are in the playoffs

## **Data description**

Data inputs are provided as part of two data sets:

### **Modeling dataset**

Consists of 20697 entries for each unique shot attempted. There is one explanatory variable (shot\_made\_flat) and twenty eight covariates.

shot\_made\_flag = 1 (or = 0) determines whether the shot was successful (or not)

### **Prediction dataset**

Consists of 5000 entries for each unique shot attempted. There are twenty eight covariates, and the shot outcome explanatory variable is absent (shot\_made\_flag)

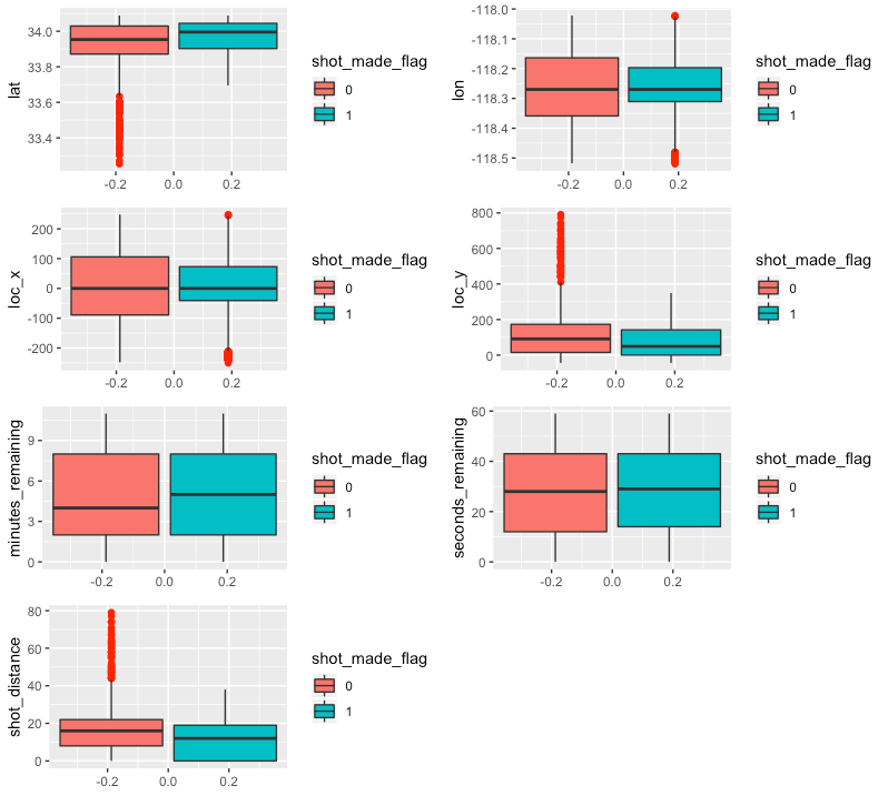
Refer Appendix 1 for detailed description on the variables

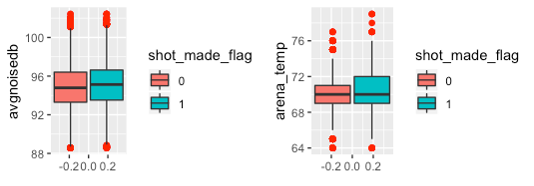
## **Exploratory data analysis**

### **Analysis of Binary Response Variable**

We observe that the response variable is fairly equally distributed and will not perform actions to deal with imbalanced response variables.

### **Outlier Analysis**

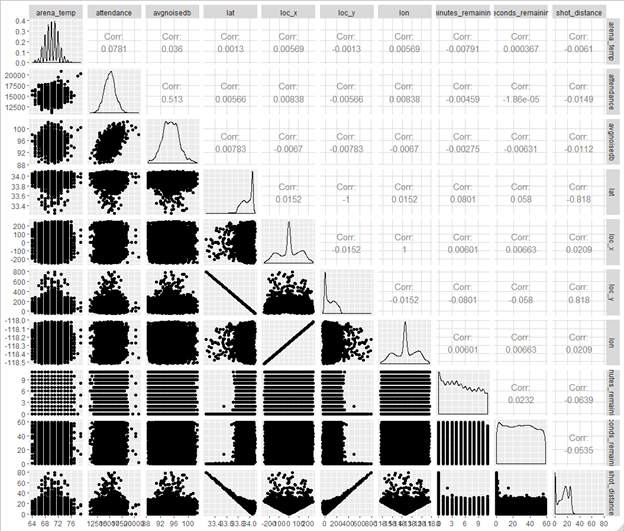




Most of the outliers seem to be associated with shot\_made\_flag = 0. E.g. for ‘lat’ covariate, shot misses seem to have outliers. We have not specially addressed outliers in our modeling.

### **Multicollinearity**

#### Correlation Plots -

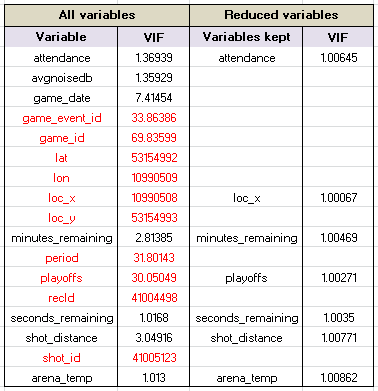


In summary, following variables have high correlation

1. shot\_distance with lat, shot\_distance with loc\_y
2. lat with loc\_y, lon with loc\_x
3. attendance with avgnoised

#### VIF analysis

Removing highly correlated variables addresses multicollinearity



Variables highlighted red have very high VIF. Removing then and keeping only one of the variables that correlate gives VIF ~ 1, taking care of multicollinearity

### **Transformation**

#### Log transformation

Initial correlation matrices show variables shot\_distance, lat, and loc\_y have visible skewness in distribution.

Log transformation of the three variables indicates a more normal distribution but still heavily skewed. *Refer to appendix for correlation matrix.*

Further, after doing the log transformation, a significant number of the observations within those variables became NA or infinite values so we decided not to go forward with the log transformation.

#### Ordinal transformation

Two of the categorical variables, shot\_type and shot\_zone\_range, were changed into ordinal data. Refer to the R code in the appendix to look at the order.

## **Data Modeling**

Shot classification models are built using (1) logistic regression model and (2) linear discriminant analysis (LDA) model, as described below. Logistic model provides better prediction model fit as described in ‘Final Model Selection Summary’

### **Logistic regression model**

#### ***Variable Selection***

id variables were removed as they don’t give any significant information about the shots. Team\_name variable is also removed as it was the Lakers for every single shot. From the correlation matrix we noticed there was a 1 to 1 correlation with the (loc\_x, lon) and (lat, loc\_y). So for model 1 we removed (loc\_x,loc\_y) and for model 2 we removed (lat, lon) and then we ran stepwise regression.

#### ***Modeling summary***

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Model #** | **Selection Technique** | **Input variable set** | **AIC** | **Miss-classification Error** | **AUC** | **Sensitivity** | **Specificity** | **Log Loss**  **Function** |
| Model 1 | Stepwise | All except loc\_x and loc\_y | 18830 | .3189 | .7053 | .4784 | .8491 | .6077 |
| **Model 2** | **Stepwise** | **All except lat and lon** | **18831** | **.3191** | **.7059** | **.4784** | **.8487** | **.6075** |
| Model 3 | Stepwise | All | 18902 | .3216 | .7049 | .4584 | .8607 | .6077 |
| Model 4 | EDA; Subject Knowledge | All | 18932 | .3206 | .7041 | .4498 | .8696 | .6077 |

Table 2: Model summary for logistic regression

We finalized Model 2 as our model for logistic regression, even though the Miss-classification error and AIC are slightly higher than found in model 1, it has the highest AUC and lowest Log Loss function out of all the models.

#### ***Model Prediction summary***

Total number of observations = 5000

Number of observations classified into shot\_made\_flat 1 = 1403

Number of observations classified into shot\_made\_flat 0 = 3597

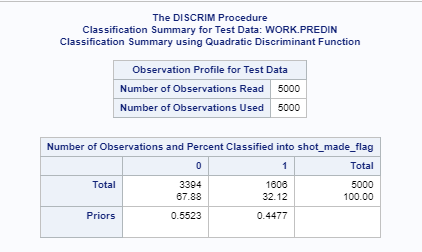
### **Linear discriminant analysis (LDA) model**

#### ***Variable Selection***

Numeric covariates based on correlation analysis (EDA) are used in LDA model for classification, i.e.(a) attendance (b) arena\_temp (c) minutes\_remaining (d) seconds\_remaining (e) shot\_distance

#### ***Modeling summary***

#### ***Model Prediction summary***



### **Final model selection summary**

Logistic regression model is used to build final predictions for the prediction data set due to (a) smaller loss function, misclassification rate, specificity, sensitivity, and

|  |  |  |
| --- | --- | --- |
| Classification Table Criterion | Logistic Regression (Model 2) | LDA model (cross validated) |
| Sensitivity (%) | 47.84 | 42.45 |
| Specificity (%) | 84.87 | 73.87 |
| Misclassification rate (%) | 31.91 | 40.19 |
| AUC (Area Under Curve) | 0.7059 | - |
| Loss Function | 0.6075 | 0.6633 |

## 

## **Statistical Inferences**

### Odds of Kobe making shots decreasing with respect to distance he is from the hoop

The odds ratio summary from logistic regression model is shown below:

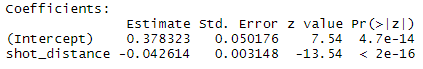


The shot\_distance has an **odds ratio of .95828** within a 95% confidence interval of (.95237,.9642).

**So this means the odds of Kobe making a shot decrease with respect to the distance he is from the hoop.**

### The probability of Kobe making a shot decreases linearly with respect to the distance he is from the hoop

The estimate of shot\_distance using logistic regression model is shown below:

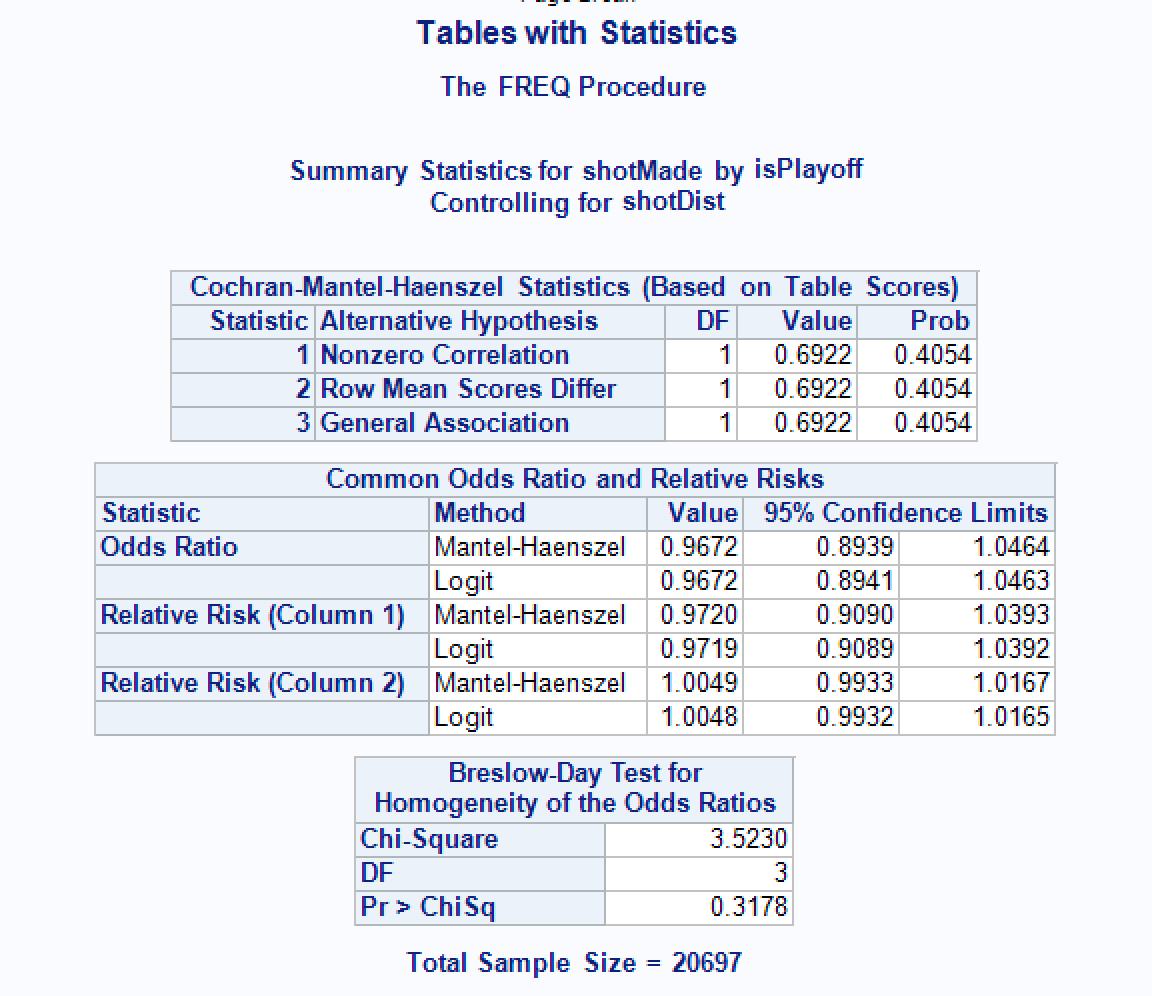


**The coefficient for shot\_distance is -.0426** so it shows a negative linear relationship with shot\_made\_flag therefore **implying that the probability of making a shot decreases linearly with respect to the distance Kobe is from the hoop**

### The relationship between the distance Kobe is from the basket and the odds of him making the shot is different if they are in the playoffs

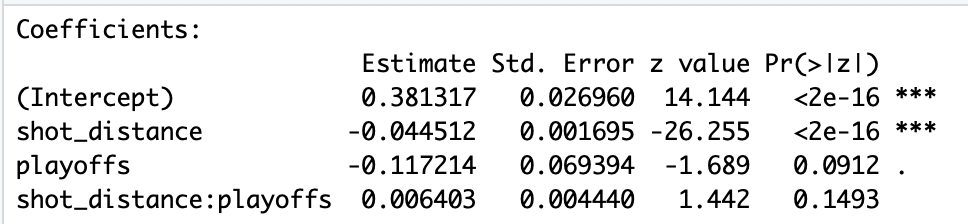
There is **not sufficient evidence t**o show that distance Kobe is from the basket and the odds of him making the shot is different if they are in the playoffs.

Results from Cochran-Mantel-Haesnzel test checking ‘if there was a difference in odds for making a shot in the playoffs vs. not in the playoffs while controlling for shot distance’ are captured below. It indicates that there was no relationship between the odds of making a shot in the playoffs vs. not playoffs.

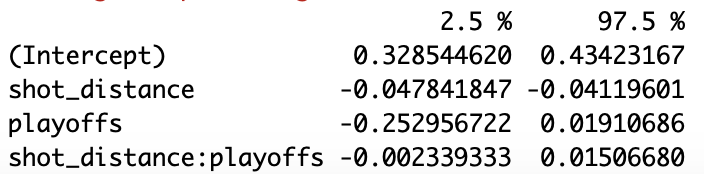


A simple logistic model with playoffs indicator and shot distance with their interaction shows following results:

**Estimates :**



**Confidence Intervals**

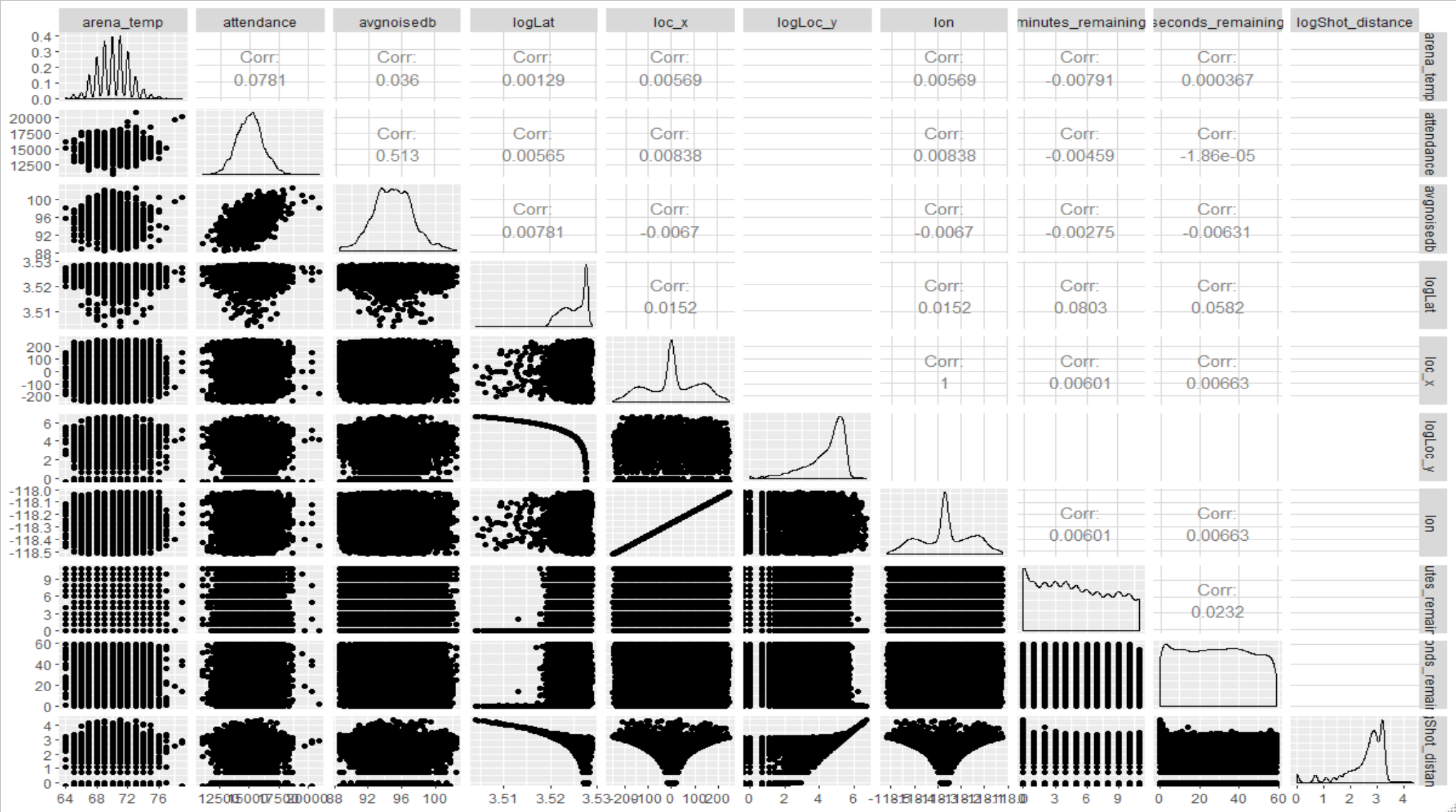


We failed to reject the null hypothesis that the interaction term (shot\_distance:playoffs) is equal to 0, indicating that there is no significant interaction between shot distance and playoffs.

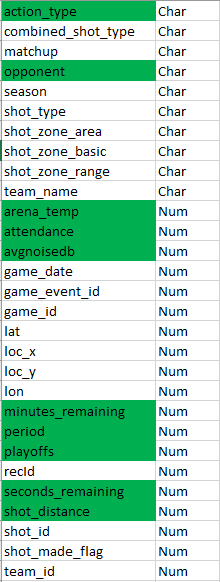
**From this we can conclude that the odds of Kobe making a shot are not different if they are in the playoffs.**

## APPENDIX 1

**Correlation matrix after log transformation:**



**Additional Variable Selection**

****

**\*** The variables highlighted in green were the ones selected to be used in Model 4 due to their statistical significance in previous models or inferences made from general basketball knowledge

## 

## APPENDIX 2

The appendix contains the code used to derive results for the report.

### SAS Code for VIF analysis ###

libname xl XLSX '/home/u41977007/DS6372/project2KobeData.xlsx';

proc reg data = xl.modelData;

model shot\_made\_flag = attendance avgnoisedb game\_date game\_event\_id game\_id lat lon loc\_x loc\_y minutes\_remaining period playoffs recId seconds\_remaining shot\_distance shot\_id team\_id arena\_temp/ vif tol collin;

run;

proc reg data = xl.modelData;

model shot\_made\_flag = attendance loc\_x minutes\_remaining playoffs seconds\_remaining shot\_distance arena\_temp/ vif tol collin;

run;

### SAS Code for LDA modeling and prediction ###

proc discrim data = dataIn pool=test crossvalidate testData= predIn testout=LDAPredOutStep out=LDAOutStep;

class shot\_made\_flag;

var attendance arena\_temp minutes\_remaining seconds\_remaining shot\_distance;

priors proportional;

run;

proc export data=LDAOutStep dbms=xlsx

outfile='/home/u41977007/DS6372/Project2\_LDAOutStepProp.xlsx' replace;

run;

proc export data=LDAPredOutStep dbms=xlsx

outfile='/home/u41977007/DS6372/Project2\_LDAPredOutStepProp.xlsx' replace;

run;

R Simple Logistic Model to Test Interaction:

df %>% lm(shot\_made\_flag~shot\_distance\*playoffs, data =.)

glm.fit <- df %>% glm(shot\_made\_flag ~ shot\_distance\*playoffs, data =., family=binomial(link="logit"))

summary(glm.fit)

confint(glm.fit)

R Boxplots:

##creating box plots for continuous random variables

lat\_box <- ggplot(data = df,aes(y = lat, fill = shot\_made\_flag)) +

geom\_boxplot(outlier.colour = 'red', outlier.shape = 16, outlier.size = 2, notch=FALSE)

long\_box <- ggplot(data = df,aes(y = lon, fill = shot\_made\_flag)) +

geom\_boxplot(outlier.colour = 'red', outlier.shape = 16, outlier.size = 2, notch=FALSE)

locx\_box <- ggplot(data = df,aes(y = loc\_x, fill = shot\_made\_flag)) +

geom\_boxplot(outlier.colour = 'red', outlier.shape = 16, outlier.size = 2, notch=FALSE)

locy\_box <- ggplot(data = df,aes(y = loc\_y, fill = shot\_made\_flag)) +

geom\_boxplot(outlier.colour = 'red', outlier.shape = 16, outlier.size = 2, notch=FALSE)

min\_rem\_box <- ggplot(data = df,aes(y = minutes\_remaining, fill = shot\_made\_flag)) +

geom\_boxplot(outlier.colour = 'red', outlier.shape = 16, outlier.size = 2, notch=FALSE)

secs\_rem\_box <- ggplot(data = df,aes(y = seconds\_remaining, fill = shot\_made\_flag)) +

geom\_boxplot(outlier.colour = 'red', outlier.shape = 16, outlier.size = 2, notch=FALSE)

shot\_dist\_box <- ggplot(data = df,aes(y = shot\_distance, fill = shot\_made\_flag)) +

geom\_boxplot(outlier.colour = 'red', outlier.shape = 16, outlier.size = 2, notch=FALSE)

attendance\_box <- ggplot(data = df,aes(y = attendance, fill = shot\_made\_flag)) +

geom\_boxplot(outlier.colour = 'red', outlier.shape = 16, outlier.size = 2, notch=FALSE)

avgnoisedb\_box <- ggplot(data = df,aes(y = avgnoisedb, fill = shot\_made\_flag)) +

geom\_boxplot(outlier.colour = 'red', outlier.shape = 16, outlier.size = 2, notch=FALSE)

arena\_temp\_box <- ggplot(data = df,aes(y = arena\_temp, fill = shot\_made\_flag)) +

geom\_boxplot(outlier.colour = 'red', outlier.shape = 16, outlier.size = 2, notch=FALSE)

ggarrange(lat\_box,long\_box,locx\_box,locy\_box,min\_rem\_box,

secs\_rem\_box,shot\_dist\_box,attendance\_box,avgnoisedb\_box,

arena\_temp\_box,ncol = 2, nrow = 4)

R Counts for Mantel-Haenszel Test:

nrow(df[which(df$playoffs == 1 & df$shot\_made\_flag == 1 & df$shot\_zone\_range == "16-24 ft."),])

nrow(df[which(df$playoffs == 1 & df$shot\_made\_flag == 1 & df$shot\_zone\_range == "24+ ft."),])

nrow(df[which(df$playoffs == 1 & df$shot\_made\_flag == 1 & df$shot\_zone\_range == "8-16 ft."),])

nrow(df[which(df$playoffs == 1 & df$shot\_made\_flag == 1 & df$shot\_zone\_range == "Back Court Shot"),])

nrow(df[which(df$playoffs == 1 & df$shot\_made\_flag == 1 & df$shot\_zone\_range == "Less Than 8 ft."),])

nrow(df[which(df$playoffs == 1 & df$shot\_made\_flag == 0 & df$shot\_zone\_range == "16-24 ft." ),])

nrow(df[which(df$playoffs == 1 & df$shot\_made\_flag == 0 & df$shot\_zone\_range == "24+ ft." ),])

nrow(df[which(df$playoffs == 1 & df$shot\_made\_flag == 0 & df$shot\_zone\_range == "8-16 ft." ),])

nrow(df[which(df$playoffs == 1 & df$shot\_made\_flag == 0 & df$shot\_zone\_range == "Back Court Shot"),])

nrow(df[which(df$playoffs == 1 & df$shot\_made\_flag == 0 & df$shot\_zone\_range == "Less Than 8 ft." ),])

nrow(df[which(df$playoffs == 0 & df$shot\_made\_flag == 1 & df$shot\_zone\_range == "16-24 ft."),])

nrow(df[which(df$playoffs == 0 & df$shot\_made\_flag == 1 & df$shot\_zone\_range == "24+ ft." ),])

nrow(df[which(df$playoffs == 0 & df$shot\_made\_flag == 1 & df$shot\_zone\_range == "8-16 ft."),])

nrow(df[which(df$playoffs == 0 & df$shot\_made\_flag == 1 & df$shot\_zone\_range == "Back Court Shot"),])

nrow(df[which(df$playoffs == 0 & df$shot\_made\_flag == 1 & df$shot\_zone\_range == "Less Than 8 ft."),])

nrow(df[which(df$playoffs == 0 & df$shot\_made\_flag == 0 & df$shot\_zone\_range == "16-24 ft."),])

nrow(df[which(df$playoffs == 0 & df$shot\_made\_flag == 0 & df$shot\_zone\_range == "24+ ft."),])

nrow(df[which(df$playoffs == 0 & df$shot\_made\_flag == 0 & df$shot\_zone\_range == "8-16 ft."),])

nrow(df[which(df$playoffs == 0 & df$shot\_made\_flag == 0 & df$shot\_zone\_range == "Back Court Shot"),])

nrow(df[which(df$playoffs == 0 & df$shot\_made\_flag == 0 & df$shot\_zone\_range == "Less Than 8 ft."),])

SAS Mantel-Haenszel Test:

data tableDat;

input isPlayoff $ shotMade $ shotDist $ count @@;

datalines;

playoff shot 16\_24ft 308

playoff shot over24ft 195

playoff shot 8\_16ft 321

playoff shot back\_court\_shot 0

playoff shot less\_than\_8ft 521

playoff noShot 16\_24ft 473

playoff noShot over24ft 367

playoff noShot 8\_16ft 403

playoff noShot back\_court\_shot 12

playoff noShot less\_than\_8ft 434

noPlayoff shot 16\_24ft 1928

noPlayoff shot over24ft 1199

noPlayoff shot 8\_16ft 1662

noPlayoff shot back\_court\_shot 0

noPlayoff shot less\_than\_8ft 3132

noPlayoff noShot 16\_24ft 2832

noPlayoff noShot over24ft 2413

noPlayoff noShot 8\_16ft 2152

noPlayoff noShot back\_court\_shot 48

noPlayoff noShot less\_than\_8ft 2297

;

run;

proc freq data=tableDat order=data;

tables shotMade\*shotDist\*isPlayoff / CMH chisq riskdiff(equal var=null) relrisk;

exact pchi or fisher;

weight count;

run;

**R Code for Building Logistic Regression Models**

library(readxl)

library(dplyr)

library(GGally)

library(MASS)

library(InformationValue)

library(regclass)

library(MLmetrics)

library(questionr)

Kobe <-read\_excel("C:/Users/Mrinmoy/Documents/School/Applied Statistics/Project 2/project2KobeData.xlsx",sheet = 1)

colnames(Kobe)[colSums(is.na(Kobe)) > 0]

KobePred <-read\_excel("C:/Users/Mrinmoy/Documents/School/Applied Statistics/Project 2/project2KobeData.xlsx",sheet = 2)

#Convert timestamp

Kobe$game\_date <- do.call("c",lapply(Kobe$game\_date, function(x) as.Date(x, origin = "1899-12-30")))

KobePred$game\_date <- do.call("c",lapply(KobePred$game\_date, function(x) as.Date(x, origin = "1899-12-30")))

#Changing Categorical Data to Ordinal

KobeTrain$shot\_type <- factor (Kobe$shot\_type, ordered = TRUE,

levels = c("3PT Field Goal","2PT Field Goal"))

KobeTrain$shot\_zone\_range <- factor(Kobe$shot\_zone\_range, ordered = TRUE,

levels = c("Back Court Shot","24+ ft.","16-24 ft."

,"8-16 ft.","Less Than 8 ft."))

KobeTest$shot\_type <- factor (Kobe$shot\_type, ordered = TRUE,

levels = c("3PT Field Goal","2PT Field Goal"))

KobeTest$shot\_zone\_range <- factor(Kobe$shot\_zone\_range, ordered = TRUE,

levels = c("Back Court Shot","24+ ft.","16-24 ft."))

KobePred$shot\_type <- factor (KobePred$shot\_type, ordered = TRUE,

levels = c("3PT Field Goal","2PT Field Goal"))

KobePred$shot\_zone\_range <- factor(KobePred$shot\_zone\_range, ordered = TRUE,

levels = c("Back Court Shot","24+ ft.","16-24 ft."

,"8-16 ft.","Less Than 8 ft."))

set.seed(3)

ind <- sample(seq(1,dim(Kobe)[1],1), round(.75 \*dim(Kobe)[1]))

KobeTrain <- Kobe[ind,]

KobeTest <- Kobe[-ind,]

KobeFinalTrain <-Kobe

#Remove ID columns

KobeTrain$game\_event\_id <- NULL

KobeTrain$game\_id <- NULL

KobeTrain$recId <- NULL

KobeTrain$shot\_id <- NULL

KobeTrain$team\_id <- NULL

KobeTrain$team\_name<-NULL

KobeFinalTrain$game\_event\_id <- NULL

KobeFinalTrain$game\_id <- NULL

KobeFinalTrain$recId <- NULL

KobeFinalTrain$shot\_id <- NULL

KobeFinalTrain$team\_id <- NULL

KobeFinalTrain$team\_name<-NULL

KobeTest$game\_event\_id <- NULL

KobeTest$game\_id <- NULL

KobeTest$recId <- NULL

KobeTest$shot\_id <- NULL

KobeTest$team\_id <- NULL

KobeTest$team\_name<-NULL

#Multi-Colinearity

Kobe %>% select(arena\_temp,attendance,avgnoisedb,lat,loc\_x,loc\_y,lon,minutes\_remaining,seconds\_remaining,shot\_distance) %>% ggpairs(diag = list(discrete = "barDiag"))

#Log Transformation

KobeLog <- Kobe

KobeLog$logShot\_distance <- log(Kobe$shot\_distance)

KobeLog$logLat <- log(KobeLog$lat)

KobeLog$logLoc\_y <- log(KobeLog$loc\_y)

KobeLog$logShot\_distance <- log(KobeLog$shot\_distance)

KobeLog %>% dplyr::select(arena\_temp,attendance,avgnoisedb,logLat,loc\_x,logLoc\_y,lon,minutes\_remaining,seconds\_remaining,logShot\_distance) %>% ggpairs(diag = list(discrete = "barDiag"))

#Remove loc\_x and loc\_y

KobeTrain1 <- KobeTrain

KobeTest1 <- KobeTest

KobeTrain1$loc\_x <- NULL

KobeTrain1$loc\_y <- NULL

KobeTest1$loc\_x <- NULL

KobeTest1$loc\_y <- NULL

#Remove lon and lat

KobeTrain2 <- KobeTrain

KobeTest2 <- KobeTest

KobeTrain2$lon <- NULL

KobeTrain2$lat <- NULL

KobeTest2$lon <- NULL

KobeTest2$lat <- NULL

KobeFinalTrain$lon <- NULL

KobeFinalTrain$lat <- NULL

#Log transformation

KobeTrain3 <- KobeTrain

KobeTest3 <- KobeTest

KobeTrain3$logShot\_distance <- log(KobeTrain3$shot\_distance)

KobeTest3$logShot\_distance <- log(KobeTest3$shot\_distance)

KobeTrain3$loc\_x <- NULL

KobeTrain3$loc\_y <- NULL

KobeTest3$loc\_x <- NULL

KobeTest3$loc\_y <- NULL

#Step-Wise Models

model1 <- glm(shot\_made\_flag ~.,data= KobeTrain1,family = binomial) %>% stepAIC(trace = FALSE)

model2 <- glm(shot\_made\_flag ~.,data= KobeTrain2,family = binomial) %>% stepAIC(trace = FALSE)

newModel2 <- glm(shot\_made\_flag ~.,data= KobeFinalTrain,family = binomial) %>% stepAIC(trace = FALSE)

orgModel <- glm(shot\_made\_flag ~.,data= KobeTrain,family = binomial) %>% stepAIC(trace = FALSE)

model3 <- glm(shot\_made\_flag ~.,data= KobeTrain3,family = binomial) %>% stepAIC(trace = FALSE)

prediction1 <- predict(model1,KobeTest1,type = "response")

optCutOff1 <- optimalCutoff(KobeTest1$shot\_made\_flag, prediction1)[1]

misClassError(KobeTest1$shot\_made\_flag, prediction1, threshold = optCutOff1)

plotROC(KobeTest1$shot\_made\_flag, prediction1)

sensitivity(KobeTest1$shot\_made\_flag, prediction1, threshold = optCutOff1)

specificity(KobeTest1$shot\_made\_flag, prediction1, threshold = optCutOff1)

LogLoss(prediction1,KobeTest1$shot\_made\_flag)

prediction2 <- predict(model2,KobeTest2,type = "response")

optCutOff2 <- optimalCutoff(KobeTest2$shot\_made\_flag, prediction2)[1]

misClassError(KobeTest2$shot\_made\_flag, prediction2, threshold = optCutOff2)

plotROC(KobeTest2$shot\_made\_flag, prediction2)

sensitivity(KobeTest2$shot\_made\_flag, prediction2, threshold = optCutOff2)

specificity(KobeTest2$shot\_made\_flag, prediction2, threshold = optCutOff2)

LogLoss(prediction2,KobeTest2$shot\_made\_flag)

testPrediction <- predict(testModel,KobeTest2,type = "response")

optCutOffTest <- optimalCutoff(KobeTest2$shot\_made\_flag, testPrediction)[1]

misClassError(KobeTest2$shot\_made\_flag, testPrediction, threshold = optCutOffTest)

plotROC(KobeTest2$shot\_made\_flag, testPrediction)

sensitivity(KobeTest2$shot\_made\_flag, testPrediction, threshold = optCutOffTest)

specificity(KobeTest2$shot\_made\_flag, testPrediction, threshold = optCutOffTest)

LogLoss(testPrediction,KobeTest2$shot\_made\_flag)

predictionOrg <- predict(orgModel,KobeTest,type = "response")

optCutOffOrg <- optimalCutoff(KobeTest$shot\_made\_flag, predictionOrg)[1]

misClassError(KobeTest$shot\_made\_flag, predictionOrg, threshold = optCutOffOrg)

plotROC(KobeTest$shot\_made\_flag, predictionOrg)

sensitivity(KobeTest$shot\_made\_flag, predictionOrg, threshold = optCutOffOrg)

specificity(KobeTest$shot\_made\_flag, predictionOrg, threshold = optCutOffOrg)

LogLoss(predictionOrg,KobeTest$shot\_made\_flag)

#No Step-wise

model4 <- glm(shot\_made\_flag ~ action\_type + opponent + arena\_temp + attendance

+ avgnoisedb + minutes\_remaining + period + playoffs + seconds\_remaining,data= KobeTrain,family = binomial)

prediction4 <- predict(model4,KobeTest,type = "response")

optCutOff4 <- optimalCutoff(KobeTest$shot\_made\_flag, prediction4)[1]

misClassError(KobeTest$shot\_made\_flag, prediction4, threshold = optCutOff4)

plotROC(KobeTest$shot\_made\_flag, prediction4)

sensitivity(KobeTest$shot\_made\_flag, prediction4, threshold = optCutOff4)

specificity(KobeTest$shot\_made\_flag, prediction4, threshold = optCutOff4)

LogLoss(predictionOrg,KobeTest$shot\_made\_flag)

#Odds Ratio Objective Model

oddsModel <- glm(shot\_made\_flag~shot\_distance, data = KobeTest, family = binomial)

questionr::odds.ratio(oddsModel, level = .95)

#Write Prediction file

finalPrediction <- predict(model2, KobePred, type = "response")

finalPredictionValue <- sapply(finalPrediction, function(x) if (x > optCutOff2){1} else {0})

TotalPrediction <- data.frame(finalPrediction,finalPredictionValue)

write.csv(TotalPrediction,file = "C:/Users/Mrinmoy/Documents/School/Applied Statistics/Project 2/PredictionData.csv")