Kobe Bryant Shot Selection

Bin Yu, Lu Cheng, Nuoya Rezsonya

Introduction

Kobe Bean Bryant played his entire 20-year career with the Los Angeles Lakers of the National Basketball Association (NBA) [1]. He marked his retirement from the NBA by scoring 60 points in his final game as a Los Angeles Laker on Wednesday, April 12, 2016. Bryant established a reputation for taking shots in the closing moments of tight games.

In this observation study, we are challenged to predict Kobe's shot, which shots will find the way to the net. The existing dataset, which includes 25,697 shots records with 27 exploratory variables, describes the location and circumstances of every field goal attempted by Kobe Bryant during his basketball career. We will use the current explanatory variables to predict the outcomes of 5,000 shot attempts.

In this project report, we will first list out data manipulation including missing values treatment, outlier recognition, multicollinearity identification and we will give explanations about decisions we made. Then we will arrive at a simple model to answer 5 specific research questions. At last, we will provide comparisons of three predictive models, Logistic Regression Model, LDA Model and Cluster/logistic Model, and the comparisons are in terms of AUC, AIC, Mis-Classification Rate, Sensitivity, Specificity and objective / loss function.

Data Description

The current datasets we have contains the location and circumstances of every field goal attempted by Kobe Bryant during his 20-year career. It has 30,697 observations of Kobe's shot information with 27 explanatory variables which including 11 continuous variables and 14 categorical variables. There is 1 binomial response variable (shot_made_flag) in the dataset. In this variable, there are 5,000 observations without value, which we will predict the proportion of the shot made for them. Please see below *(Table 1. Explanatory Variables)* for the detailed explanatory variable list. We have included the detailed variable definition and transformation in Appendix I.

Continuous Variables (11)	Numeric Categorical Variables (5)	Character Categorical Variables (9)	Variables not Used (2)
Lat Loc_x Loc_y Lon Minutes_remaining Seconds_remainnig Short_distance Attendance Arena_temp Avgnoisedb Arena_temp	Period Playoffs (binomial) Game_id game_event_id Shot_id	Action_Type Combined_Shot_Type Shot_Type Shot_zone_Area Shot_Zone_Basic Shot_Zone Range Season matchup opponent	Team_Name Team_id

Table 1. Explanatory Variables

Exploratory Analysis and Data Cleaning

Since the dataset is not good enough to use directly, necessary data manipulation is needed in order to make future modelling as accurate as possible. Hence, this falls into three major processes: potential transformation, outlier recognition and multicollinearity identification.

The Need for Any Potential Transformations

1. **Logit transformation on response variable.**

First of all, we have arrived at a scatter plot of the shot_distance vs. shot_made_flag, as shown on the *(Figure 1. Scatter Plot of shot_distance vs. shot_made_flag).* As you can see successful shots were made within a shot_distance of 50. However, it is not intuitive to show any relationship between shot distance and shot made/missed. In order to have a response that will be continuous across the real line, we decide to perform a logit transformation on the response, in other words, the future regression will be logistic regression with logit link function.

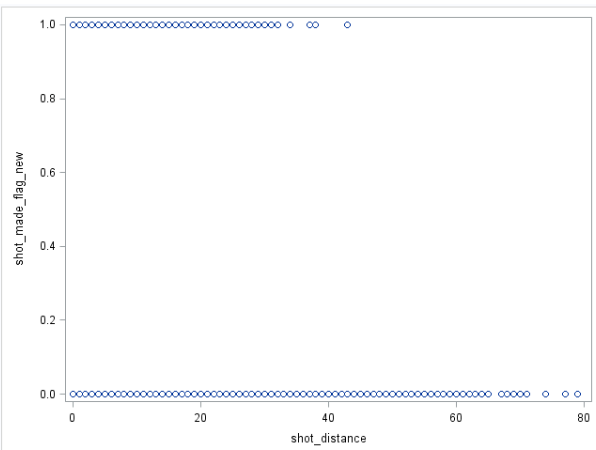


Figure 1. Scatter Plot of shot_distance vs. shot_made_flag

The MEANS Procedure

Variable	Minimum	Lower Quartile	Mean	Upper Quartile	Maximum	N Miss	N
game_event_id	2.0000000	110.0000000	249.1908004	368.0000000	659.0000000	0	30697
game_id	20000012.00	20500077.00	24764065.87	29600474.00	49900088.00	0	30697
lat	33.2533000	33.8843000	33.9531925	34.0403000	34.0883000	0	30697
loc_x	-250.0000000	-68.0000000	7.1104994	95.0000000	248.0000000	0	30697
loc_y	-44.0000000	4.0000000	91.1075349	160.0000000	791.0000000	0	30697
lon	-118.5198000	-118.3378000	-118.2626895	-118.1748000	-118.0218000	0	30697
minutes_remaining	0	2.0000000	4.8856240	8.0000000	11.0000000	0	30697
period	1.0000000	1.0000000	2.5194319	3.0000000	7.0000000	0	30697
playoffs	0	0	0.1465616	0	1.0000000	0	30697
seconds_remaining	0	13.0000000	28.3650845	43.0000000	59.0000000	0	30697
shot_distance	0	5.0000000	13.4374369	21.0000000	79.0000000	0	30697
team_id	1610612747	1610612747	1610612747	1610612747	1610612747	0	30697
game_date	13456.00	15484.00	16990.96	18336.00	20557.00	0	30697
shot_id	1.0000000	7675.00	15349.00	23023.00	30697.00	0	30697
attendance	11065.00	14314.00	15039.89	15737.00	20845.00	0	30697
arena_temp	64.0000000	69.0000000	70.0989022	71.0000000	79.0000000	0	30697
avgnoisedb	88.5600000	93.4000000	94.9495983	96.4900000	102.4300000	0	30697

Figure 2.SAS proc means Result

2. **Missing value treatment.**

There is no missing value of explanatory variables in the original data set. See *(Figure 2: SAS proc means Result).*

3. **Data type transformation**

Necessary transformations have been done to several categorical variables because they do not have actual numerical meaning. It is also very obvious that the response variable of this study should be binary categorical because Kobe can either make the shot or miss the shot. Variables being transformed are as: game_event_id, game_id, period, Playoffs, shot_made_flag, shot_id, game_date. But we will not use shot_id, game_event_id in any logistic modelling since they are just indexes and have no real influences on whether Kobe made the shot.

4. **Deleted variables**

There are two variables: team_id and team_name being deleted because Kobe has only played for Lakers and values in these two columns are the same.

5. **Variables combination and variable update**

There are 2 pairs of the location where Kobe made the shot. Lat and loc_y as well as lon and loc_x is high linearly correlated, as shown on Figure 3: Scatter Matrix of location variables. We decide that we will only keep loc_x and loc_y as location variables. The rest of the explanatory variables are not showing evidence of linear relationship between them.

We have combined minutes_remaining and seconds_remaining into Second_to_period_end so that the time remaining in each period can be all expressed in one format.

In order to get more information on whether the game is at LA, we split up this original column, matchup, based on “@” and “vs.” in it. Rows with “@” will be assigned level 0 since the game is not home. Rows with “vs.” will be assigned level 1 since the game is at home.

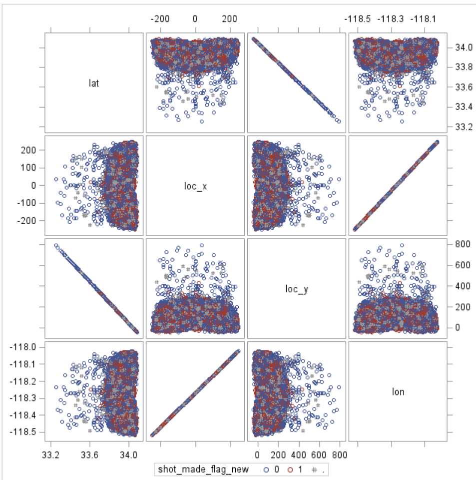


Figure 3: Scatter Matrix of location variables

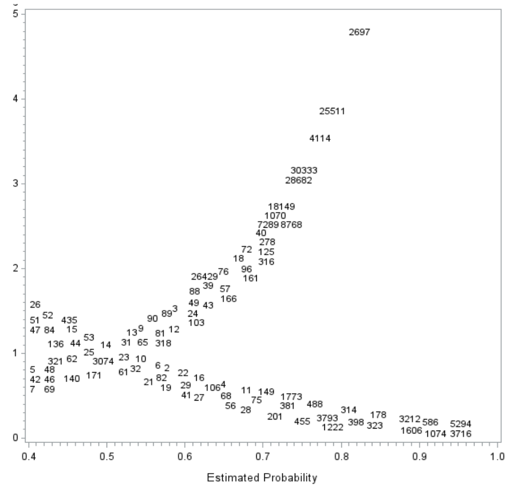


Figure 4: Plot of Influential Points

Outlier Recognition

We have also generated a plot to detect any influential points *(Figure 4: Plot of Influential Points)*. As you can see there are no points that are considered to be influential.

Multicollinearity Identification

Like we discussed in the section of Variables combination and variable update. There are 2 pairs of the location where Kobe made the shot. Lat and loc_y as well as lon and loc_x is high linearly correlated, as shown on *(Figure 3: Scatter Matrix of location variables).* We decide that we will only keep loc_x and loc_y as location variables.

For the transformation details, please see Appendix I. Explanatory Variable Definition and Transformation.

Interpretation Models/Questions

Simple Logistic Regression Model

To use logistic regression, there are three assumptions needed to be met:

1. **Response is binary:** which has been met
2. **Log odds should be linearly related to the explanatory variables:** we will talk in research question 3, the relationship is not linear but very close, it is reasonable to proceed with this assumption met
3. **Observations should be independent:** assume they are independent

Analysis of Maximum Likelihood Estimates

Parameter		DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept		1	0.3460	0.0266	169.1083	<.0001
shot_distance		1	-0.0440	0.00141	976.9580	<.0001
playoffs	1	1	-0.0172	0.0362	0.2258	0.6347
Home_play	1	1	0.0477	0.0256	3.4546	0.0631

Odds Ratio Estimates

Effect	Point Estimate	95% Wald Confidence Limits	
shot_distance	0.957	0.954	0.960
playoffs 1 vs 0	0.983	0.916	1.055
Home_play 1 vs 0	1.049	0.997	1.103

Figure 5: Logistic Regression Result from SAS

The selected model has only 3 variables: shot_distance as numerical variable along with playoffs and Home_play as binary categorical variables. The model is using playoff = 0(the game is not in play off season) and Home_play =0(the game is not in LA) as the reference level. The response variable is using show_made_flag_new =0, Kobe missed the shot, as the reference level. The probability modeled is show_made_flag_new = 1, when Kobe made the shot.

The regression result from SAS is shown on *(Figure 5. Regression Result from SAS)*. The logistic regression equation is as below:

logit(πhat) = 0.3460 - 0.0440 * shot_distance - 0.0172 * (playoffs) + 0.0477 *(Home_play)

Research Question 1: Shot distance matters?

As the regression result shown above *(Figure 5: Regression result from SAS),* at significant level of 5%, the coefficient for the variable shot_distance is -0.0440 which is statistically significant with a p-value < 0.0001. This indicates that for a one-unit change in shot_distance, we are expecting an estimated -0.0440 increase in the log of the odds of the dependent variable shot_made_flag_new= 1 (Kobe made the shot). For one-unit increase in shot_distance variable, the estimated odds of having status shot made (vs having shot missed) increase by a factor of $e^{-0.0440}$ = 0.957, which indicates the odds going smaller as the distance going bigger, after accounting for playoffs and Home_play variables.

Conclusion: There is enough evidence to suggest that the shot distance increase on-unit the estimated odds of Kobe's shot made will decrease 0.957 (p-value <0.0001). 95% confidence interval for this difference is 0.954 to 0.960 *(Figure 6. confidence interval plot of odds ratio estimates)*.

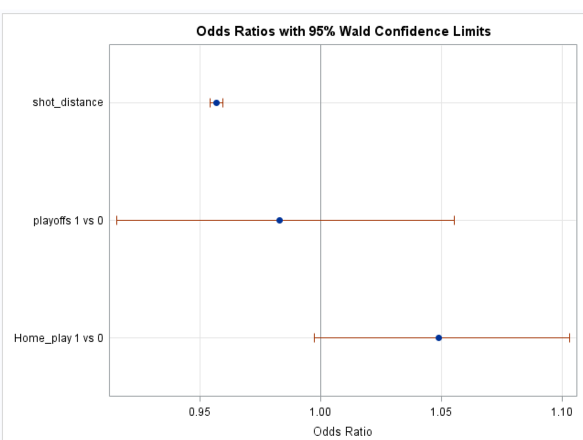


Figure 6: Confidence Interval Plot of odds ratio estimates

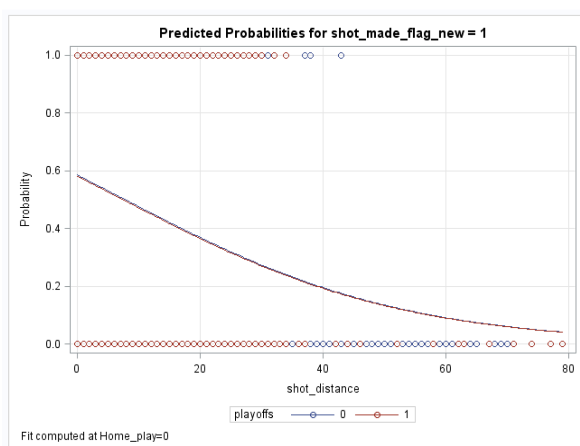


Figure 7: Plot of shot_distance vs. the proportion of shot

made

Research Question 2: Shot distance affects shot?

By observing the effect plot generated by SAS, the plot of shot_distance vs. the proportion of making a shot *(Figure 7: Plot of shot_distance vs. the proportion of making a shot)* verifies that as the shot distances going further, the probability of making a shot is going smaller and the rate of probability going smaller is getting slower. The relationship is not linear but close. Therefore, after the logit transformation the linearity of logit and explanatory variable assumption is met.

Research Question 3: Playoff game affects shot?

As the SAS logistic regression result shown *(Figure 5. Regression Result from SAS),* the odds of making a shot successfully when the game is in playoffs is estimated to be $e^{-0.0172}$= 0.983 times the odds of when the game is not in playoffs, with fixed levels of shot distance and home play. An approximate 95% confidence interval is 0.916 to 1.055. In other words, the relationship between the distance Kobe is from the hoop and the odds of him making the shot is different based on whether he is in the playoffs or not and it is lower when he is in playoffs. The odds of him making a shot successfully is lower when the game is in playoffs. For example, if the shot distance is 25 and the game is at home, he has an estimated $\frac{e^{(0.3460\,-\,0.0440\,*25\,-\,0.0172+\,0.0477)}}{1+e^{\quad(0.3764-0.0440*25+\,0.0172+0.0477)}}$=0.3266227 = 32.66227% probability of making the shot if he is in playoffs compared to an estimated $\frac{e^{(0.3764-0.0440*25+0.0477)}}{1+e^{\quad(0.3764-0.0440*25+0.0477)}}$=33.04169% probability of making a shot if he is not in playoffs (p-value=0.6347).

Conclusion: There is NOT enough evidence to suggest that Kobe will has more estimated odds of Kobe's shots in playoff games holding shot_distance and Home_Play constant (p-value=0.6347). 95% confidence interval is 0.916 to 1.055.

Research Question 4: Home game affects shot?

As the SAS logistic regression result shown *(Figure 5. Regression Result from SAS),* the odds of making a shot successfully when the game is in LA is estimated to be $e^{0.0477}$= 1.048856 times the odds of when the game is not in LA, with fixed levels of shot_distance and playoffs (p-value=0.0631). An approximate 95% confidence interval is 0.997 to 1.103. In other words, the odds of making a shot successfully when the game is at home is higher than when the game is not at home.

Conclusion: There is NOT enough evidence to suggest that Kobe will has more estimated odds of Kobe's shot in home games holding shot_distance and Playoffs constant (p-value=0. 0631). 95% confidence interval is 0.997 to 1.103

Research Question 5: Is Kobe a good clutch?

In this question we use the clutch as last 180 seconds of each period. We will use the all means from the first 9 mins of each period as a reference line and compare it with the last 3 mins percentage to it. The table of summary is as shown below *(Table 2: Clutch and Overall Average Comparison)*. We conclude that if there is no overtime in the game, he may be a good clutch since Kobe's shot percentage is always higher than the first 9 minutes. getting smaller as the game going toward the end.

Clutch Period	Local mean(proportion)	Grand mean	Comparing to the average
1	0.4715803	0.4512712	More than average
2	0.4662219	0.4171598	More than average

3	0.4651655	0.4192650	More than average
4	0.4273063	0.3862139	More than average
5	0.3586957	0.4840426	Less than average
6	0.5714286	0.4347826	More than average
7	0	0.6	Less than average

Table 2: Clutch and Overall Average Comparison

Predictive Models

In this part, we are going to introduce 3 predictive models we have achieved and then compare them in terms of AUC, AIC, Misclassification Rate, Sensitivity, Specificity and objective / loss function.

Logistic Regression Model

To use logistic regression, there are three assumptions needed to be met:

1. **Response is binary:** which has been met
2. **Log odds should be linearly related to the explanatory variables:** details in research question 3, the relationship is not linear but very close, it is reasonable to proceed with this assumption met)
3. **Observations should be independent:** assume they are independent

We have created several explanatory variables based on the original data:

End_Three_Sec	A binary variable. When the remaining seconds is less than 3 seconds, End_Three_Sec will be 1.
season_P2	To make the season data as a 6 by 2 matrix.
game_year	It is the year of that game

The selected logistic model equation comes back as below, and the probability modeled is show_made_flag_new = 1, when Kobe made the shot.

$$\begin{aligned}\mathbf{logit(\pi)} = &\ \beta_0 + \beta_1 \textbf{action\_type} + \beta_2 \textbf{End\_Three\_Sec} + \beta_3 \textbf{shot\_zone\_range} + \beta_4\ \textbf{season\_P2} \\ &+ \beta_5\ \textbf{shot\_zone\_area} + \beta_6\ \textbf{arena\_temp} + \beta_7\ \textbf{period} + \beta_8 \textbf{shot\_zone\_basic} + \beta_9 \textbf{attendance} + \beta_{10}\ \textbf{loc\_y} \\ &+ \beta_{11}\ \textbf{game\_year} + \beta_{12}\ \textbf{shot\_distance*action\_type} + \beta_{13}\ \textbf{period*Second\_to\_period\_end} \\ &+ \beta_{14}\ \textbf{loc\_y*shot\_distance} + \beta_{14}\ \textbf{shot\_zone\_basic*shot\_zone\_area}\end{aligned}$$

By checking the Hosmer and Lemeshow result, there is not enough evidence to suggest a lack of fit of this logistic regression model with a p-value of 0.9921. For all logistic regression fits output, please see as shown on *(Table3: HL test results and model fit statistics tables),* ROC curve and the log-loss result of 0.3931.

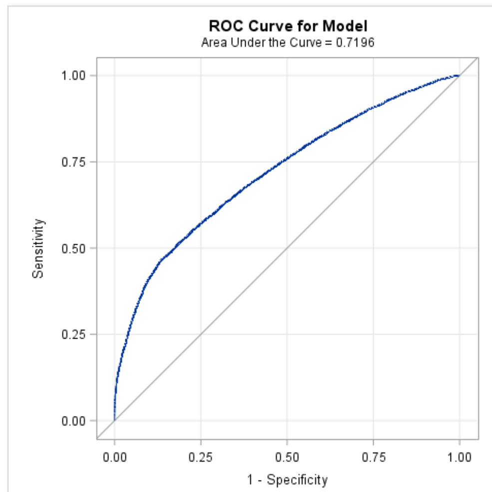
Model Fit Statistics		
Criterion	Intercept Only	Intercept and Covariates
AIC	35327.083	30968.146
SC	35335.237	32248.344
-2 Log L	35325.083	30654.146

Testing Global Null Hypothesis: BETA=0			
Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	4670.9372	156	<.0001
Score	4245.1391	156	<.0001
			)001

Hosmer and Lemeshow Goodness-of-Fit Test		
Chi-Square	DF	Pr > ChiSq
1.5341	8	0.9921

Analysis Variable : lossfuc				
N	Mean	Std Dev	Minimum	Maximum
5000	-0.3931884	0.1446333	-0.6931472	0

Joint Tests			
Effect	DF	Wald Chi-Square	Pr > ChiSq
action_type	54	1166.5906	<.0001
End_Three_Sec	1	67.4318	<.0001
shot_zone_range	4	2.6943	0.6102
season_p2	19	54.4643	<.0001
shot_zone_area	4	10.7291	0.0298
arena_temp	1	23.8230	<.0001
period	6	22.0816	0.0012
shot_zone_basic	5	4.4473	0.4870
attendance	1	174.6227	<.0001
loc_y	1	28.7954	<.0001
game_year	1	0.1487	0.6998
shot_dist*action_typ	48	115.6176	<.0001
Second_to_per*period	6	12.5767	0.0503
loc_y*shot_distance	1	44.1247	<.0001
shot_zone*shot_zone_	4	3.4928	0.4790



Fit Statistics for SCORE Data											
Data Set	Total Frequency	Log Likelihood	Error Rate	AIC	AICC	BIC	SC	R-Square	Max-Rescaled R-Square	AUC	Brier Score
WORK.KOBEDATA_LOG	25697	-15327.1	0.3131	30968.15	30969.77	32276.25	32248.34	0.166207	0.222475	0.719575	0.205919

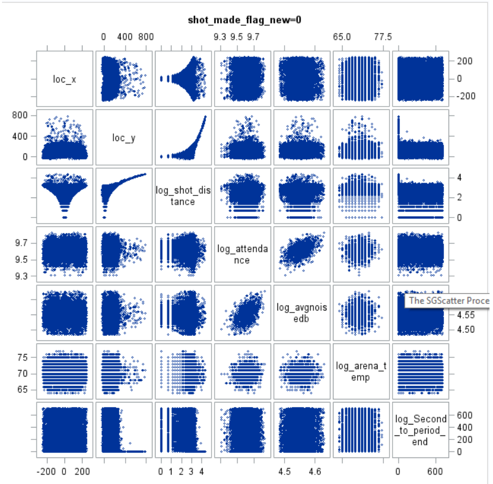
Table3. HL test result and model fit statistics tables.

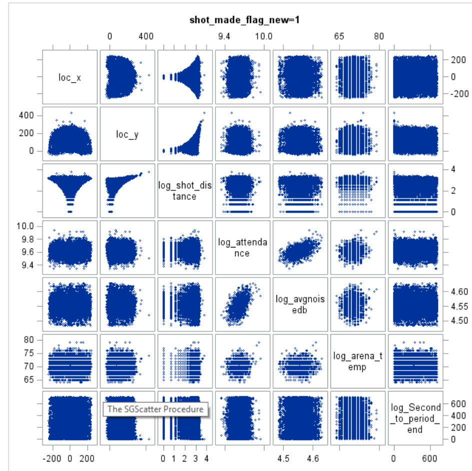
Linear Discriminant Analysis/Quadratic Discriminant Analysis

We used the variables selected from logistic regression as start point. Eight continues variables were chosen into the model.

Assumptions

1. **Multi Variate Normality**: The training dataset contains over 25,000 observations. Normality is robust for a large sample size according to CLT. There is no violation of normality.
2. **Independence:** We use a PCA model to create the new non-correlated numerical variables to avoid dependency. We assume that the independence assumption is met.
3. **Homogeneity**:





The SAS System

The DISCRIM Procedure
Test of Homogeneity of Within Covariance Matrices

Chi-Square	DF	Pr > ChiSq
999.810902	28	<.0001

It is hard to identify the ellipses due to the large dataset even after log transformation. Since the normality assumption is met, we run the Bartlett's test to check homogeneity. Significant evidence shows that at least one pair of covariances or variances are different in shot_made_flag. (p-value < 0.0001 in Bartlett's test). The Homogeneity assumption is false for LDA. Quadratic discriminant analysis is more suitable for this test.

Table 5. Ellipses Plots and Bartlett's Test Results.

Data preparation

1. **Categorical Variables**

The dataset we got has both continuous and categorical explanatory variables. To build a LDA model properly, we need to make proper adjustments on the categorical variables. There are many ways to transform the categorical variables, and we regroup the data accordingly. We divide the data into 32 groups by 3 categorical variables: *N_Combined_action_type*, *home_play* and *playoffs*. A description of *N_Combined_action_type* is in appendix I.

2. **Numeric Variables**

We used eight continues variables in the model. To reduce correlations between variables, we used the PCA model to re-create new non-correlated variables and added them into the model.

3. **Priors**

We checked the frequency of the shot_made_flag to determine the priors in the LDA model.

Building LDA model

Eigenvectors

	Prin1	Prin2	Prin3	Prin4	Prin5	Prin6	Prin7	Prin8
loc_x	0.005429	-.003889	-.410723	0.724061	-.534401	0.141131	0.019860	0.033177
loc_y	0.699555	0.056078	0.044288	0.034907	0.067614	0.015547	-.008632	0.706690
shot_distance	0.699670	0.049138	0.015048	0.075495	0.054657	0.003870	0.000587	-.706484
attendance	-.056946	0.698173	0.025708	0.031115	-.026474	-.062190	-.709271	-.006942
avgnoisedb	-.049017	0.692389	0.087599	0.029862	-.038647	-.122829	0.702172	0.001591
arena_temp	-.013909	0.163110	-.565497	-.191507	0.359528	0.695745	0.058385	-.003266
game_id	-.039409	0.017078	0.697765	0.153653	-.149481	0.681992	-.002071	-.014390
Second_to_period_end	-.116895	-.026091	0.118832	0.637710	0.743678	-.108151	-.002388	0.010035

Table 6 :PCA model.

We ran a LDA test on each group of data. Total 32 times of LDA tests have been run. We used cross validation and priors in the analysis. Results are merged by the LDA results of each data group. We calculate the evaluation parameters of the final results. In *Table 7. Results for data group 1*, we only listed the results for first group.

The SAS System

The DISCRIM Procedure
Classification Summary for Calibration Data: WORK.TRAIN
Cross-validation Summary using Quadratic Discriminant Function

Number of Observations and Percent Classified into shot_made_flag_new			
From shot_made_flag_new	0	1	Total
0	3505 74.99	1169 25.01	4674 100.00
1	1555 67.79	739 32.21	2294 100.00
Total	5060 72.62	1908 27.38	6968 100.00
Priors	0.55	0.45	

Error Count Estimates for shot_made_flag_new			
	0	1	Total
Rate	0.2501	0.6779	0.4426
Priors	0.5500	0.4500	

The SAS System

The DISCRIM Procedure
Classification Summary for Test Data: WORK.TEST
Classification Summary using Quadratic Discriminant Function

Observation Profile for Test Data	
Number of Observations Read	1310
Number of Observations Used	1310

Number of Observations and Percent Classified into shot_made_flag_new			
From shot_made_flag_new	0	1	Total
.	952 72.67	358 27.33	1310 100.00
Total	952 72.67	358 27.33	1310 100.00
Priors	0.55	0.45	

Error Count Estimates for shot_made_flag_new	
	Total
Rate	.
Priors	0.0000

* Final results are merged by results of each data group.

Table 7: Results for data group 1

Conclusion:

The LDA model predicts the probability of success rate of each Kobe's shot. The overall misclassification rate is 41.71% and the AUC is 0.608.

LDA model summary and Issue

1. Homogeneity assumption is not satisfied. The LDA is not a suitable model for this test. Quadratic discriminant analysis is running for this test.
2. There are 15 observations in the training dataset that are missing predictions. This problem may be caused by the data grouping. The test dataset of one of the groups is null. LDA could not give a prediction with null test data. However, this issue does not impact the prediction of test data. When calculating the final results, we set null prediction of the training dataset to 0.45 by default.

Cluster + Logistic Regression Model

We selected K-Means clustering to perform the clustering on the continuous variables. In SAS, FASTCLUS procedure performs a disjoint cluster analysis based on the distances computed from one or more quantitative variables and it is good for the big dataset as well. We selected below continuous variables for PROC FASTCLUS. To improve the performance, we used PROC STANDARD to standardize the variables to make sure they are on the same level.

loc_x loc_y shot_distance arena_temp avgnoisedb Second_to_period_end lat lon

Since there are 8 dimensions in the FASTCLUS model, it is hard to draw scatter plot to see the distribution of the clusters. We created a macro to save 9 outputs from the 9 FASTCLUS procedures. Then draw the R-square plot as Figure 8. Cluster for each cluster identified. The elbow is around cluster 3, that means it at least have 3 clusters in this model. We will select maxcluster=4 as the model for our prediction. In *Figure 9. First 4 clustering distribution*. you can see the cluster distribution (used PROC CANDISC to generate canonical variables and plot them). Please see Appendix II for the SAS codes.

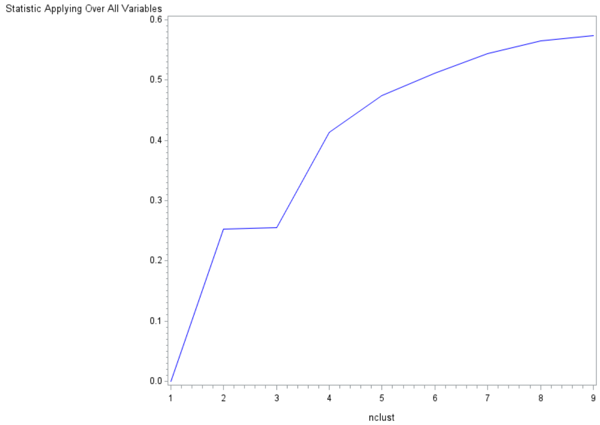


Figure 8. Cluster Plot for Each Cluster Identified

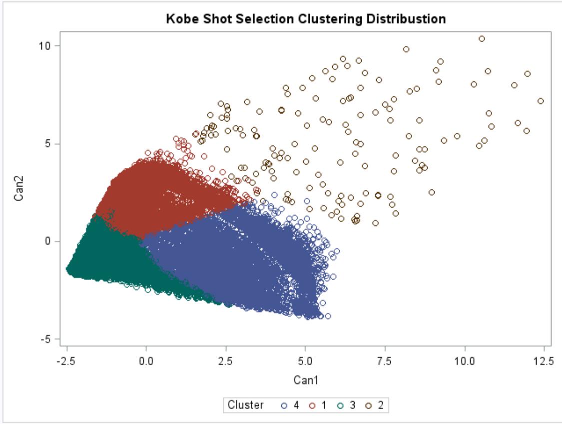


Figure 9. First 4 clustering distribution

After we have the clusters, we will need a model to predict the shot made. Since the response variable is binomial, logistic regression is supported in this dataset. As we discussed in the logistic regression, it satisfied below assumptions. We went ahead to build a logistic regression model and include the cluster as a explanatory variable.

1. **Response is binary:** which has been met
2. **Log odds should be linearly related to the explanatory variables**: we will talk in research question 3, the relationship is not linear but very close, it is reasonable to proceed with this assumption met
3. **Observations should be independent:** assume they are independent

Please see below Cluster + logistic model equation:

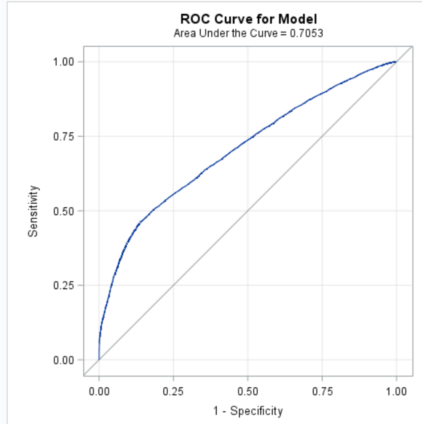
$$\textbf{logit}(\pi) = \beta 0 + \beta_1 \textbf{action\_type} + \beta_2 \textbf{Cluster} + \beta_3 \textbf{shot\_zone\_range} + \beta_4\ \textbf{season\_p2}$$
$$+ \beta_5\ \textbf{shot\_zone\_area} + \beta_6\ \textbf{period} + \beta_7\ \textbf{shot\_zone\_basic} + \beta_8\ \textbf{End\_Three\_Sec}$$
$$+ \beta_9 \textbf{shot\_zone\_basic*shot\_zone\_area}$$

We listed the results of the cluster logistic regression results in *Table 8 Cluster Logistic Regression Results*. You can find the SAS code in Appendix II.

Conclusion: This cluster + logistic model will have misclassification rate of 31.72%, AUC 0.705294. The log loss of this model is 0.3906. Based on the Hosmer and Lemeshow goodness of fit test, there is not enough evidence to suggest that there is a lack of fit issue with this model (p-value =0.6754 and rejected the H_0).

Future works: The cluster variable in the final logistic model is not significant, we need to improve the performance of the clustering. Maybe change the categorical variables to vector or remove some continuous variables from the clustering.

Joint Tests			
Effect	DF	Wald Chi-Square	Pr > ChiSq
action_type	54	2243.1922	<.0001
shot_zone_range	4	36.9833	<.0001
season_p2	19	99.1076	<.0001
shot_zone_area	4	18.9451	0.0008
period	6	32.1895	<.0001
shot_zone_basic	5	22.9328	0.0003
End_Three_Sec	1	93.3730	<.0001
CLUSTER	1	1.1252	0.2888
shot_zone*shot_zone_	4	3.5743	0.4667



Hosmer and Lemeshow Goodness-of-Fit Test		
Chi-Square	DF	Pr > ChiSq
5.7482	8	0.6754

Fit Statistics for SCORE Data											
Data Set	Total Frequency	Log Likelihood	Error Rate	AIC	AICC	BIC	SC	R-Square	Max-Rescaled R-Square	AUC	Brier Score
WORK.OUTDATA4	25697	-15514.2	0.3172	31226.49	31227.14	32051.34	32033.75	0.153971	0.206098	0.705294	0.208947

Table 8: Cluster Logistic Regression Results

Model Evaluation

In below table, we compared each competing model with the AUC, Misclassification Rate, Sensitivity, Specificity and objective / loss function. We manually calculated the log loss with below formula. Based on the results, we think the Model 3 (Logistic model) has better performance even though it has lower AUC and higher Misclassification rate, but it has better log loss, sensitivity and specificity.

$$-\frac{1}{N}\sum_{i=1}^{N}[y_i \log p_i + (1-y_i)\log(1-p_i)].$$

Where N is the total number classifications, y_i is the shot_made_flag and p_i are the probability from the model of each outcome (shot made or shot missed.)

Test Set Models	AUC	AIC	Misclassification Rate	Sensitivity	Specificity	Log Loss
Model 1 (Logistic)	0.71958	30968.15	0.3131	46.3498%	64.1502%	0.3931
Model 2 (LDA)	0.608		0.417	44.54%	72.15%	0.729
Model 3 (Cluster + Logistic)	0.70529	31226.49	0.3172	46.00%	64.00%	.3906

Table 9: Model Comparisons

References:

1. https://en.wikipedia.org/wiki/Kobe_Bryant
2. https://www.analyticsvidhya.com/blog/2015/11/easy-methods-deal-categorical-variables-predictive-modeling/
3. http://documentation.sas.com/?docsetId=grstatproc&docsetTarget=n1ow47onjbmpeln12zysuuihx3dg.htm&docsetVersion=9.4&locale=en
4. https://stats.idre.ucla.edu/sas/seminars/sas-logistic/proc-logistic-and-logistic-regression-models/
5. https://support.sas.com/documentation/cdl/en/statug/63033/HTML/default/viewer.htm#statug_fastclus_sect016.htm
6. http://support.sas.com/documentation/cdl/en/proc/61895/HTML/default/viewer.htm#standard-overview.htm
7. https://support.sas.com/documentation/cdl/en/statug/63033/HTML/default/viewer.htm#statug_candisc_sect004.htm

Appendix I

Data Definition and Transformation:

Variable Name	Variable Type from The Given Dataset	Transformation of data	Reasons of transformation
action_type	Categorical	Value adjustment	There is one type of action, Cutting Finger Roll Layup Shot, which we believe it is a typo. We have switched it to Finger Roll Layup Shot.
combined_shot_type	Categorical	N/A	NA
game_event_id	Numerical	Switch to categorical variable	These two columns describe in which game and time the shots were made. Therefore, there are duplicates in it. We believe it is better to use the column as categorical variable.
game_id	Numerical		
lat	Numerical	Only keep on pair of the location: loc_x and loc_y	These 4 columns describe the location where the shots were made. However, loc_y is highly linearly correlated with lat and the same situation applies to loc_x and lon, we decide that we will only keep one pair, loc_x and loc_y, in the following analysis.
loc_x	Numerical		
loc_y	Numerical		
lon	Numerical		
minutes_remaining	Numerical	Combined with seconds_remaining	We believe it is better to keep time remaining in one column instead of two.
period	Numerical	Switch to categorical variable	The column describes which period of the game when the shots were made. It is better to use the column as categorical variable instead of numerical.
playoffs	Numerical	Switch to	This column describes whether the team has been in

		categorical variable	the playoff games. We believe it is better to use the column as categorical variable.
season	Categorical	N/A	N/A
seconds_remaining	Numerical	Combined with minutes_remaining	We believe it is better to keep time remaining in one column instead of two.
shot_distance	Numerical	N/A	N/A
shot_made_flag	Numerical	Switch to categorical variable	This is the binary response variable. The outcome of any shot made by Kobe should be either he made the shot or he missed the shot. It shouldn't be numerical.
shot_type	Categorical	N/A	N/A
shot_zone_area	Categorical		
shot_zone_basic	Categorical		
shot_zone_range	Categorical		
team_id	Numerical	Deleted	Deleted because Kobe has only played for Lakers and values in these two columns are all the same.
team_name	Categorical		
game_date	Categorical	N/A	N/A
matchup	Categorical	Created a new column named HomeField based on this column	In order to get more information on whether the game is at LA, we split up this original column based on "@" and "vs." in it. Rows with "@" will be assigned level 0 since the game is not home. Rows with "vs." will be assigned level 1 since the game is at home.
opponent	Categorical	N/A	N/A
shot_id	Numerical	Switch to categorical variable	It is not appropriate to use the shot_id as numerical variable in this situation.
attendance	Numerical	N/A	N/A
arena_temp	Numerical		
avgnoisedb	Numerical		

LDA Model:

N_Combine_action_typ**e**:

N_Combine_a ction_type	**Combine_shot_type**	**action_type**	
0	Jump Shot	Jump Shot	• Categorical variable *N_Combined_action_type* is created by combining and adjusting two categorical variables (*action_type* and *combined_shot_type*). It is created to reasonably parse some levels of *combined_shot_type*.
1	Jump Shot	Others	
2	Layup Shot	Layup Shot	
3	Layup Shot	Driving Layup Shot	
4	Layup Shot	Others	
5	Dunk		
6	Tip Shot		
7	Others		

Appendix II

Data import and cleanup SAS script:

```
/*Import the data from csv file*/
FILENAME REFFILE
'C:\Users\yubin\OneDrive\MyWork\SMU\MSDS6372\Unit14Project\KobeDataProj2.csv';

PROC IMPORT DATAFILE=REFFILE
      DBMS=CSV Replace
      OUT=KobeData;
      GETNAMES=YES;
      GUESSINGROWS=MAX; /*we got some probelm on the season field, this will help us to
get through the error*/
RUN;
/*check the data, but only display the first 10 rows*/
proc print data=KobeData (obs=10);
Run;

/*build dummy columns*/
proc glmmod data=KobeData outdesign=GLMDesign outparm=GLMParm;
class action_type combined_shot_type game_event_id game_id period playoffs season
shot_type
  shot_zone_area shot_zone_basic shot_zone_range game_date opponent;
model shot_id= action_type minutes_remaining period season seconds_remaining
shot_zone_area
                        shot_zone_basic shot_zone_range attendance arena_temp;
run;
proc print data=GLMDesign (obs=10); run;
```

```
proc print data=GLMParm ; run;

/*Merge the dummy data*/
data KobeData;
  merge KobeData GLMDesign;
  run;

/*create new column to deal with NA in shot_made_flag field*/
data KobeData;
      set KobeData;
      if shot_made_flag='1' then shot_made_flag_new=1;
            else if shot_made_flag='0' then shot_made_flag_new=0;
            else shot_made_flag_new=.;
      Second_to_period_end = minutes_remaining * 60+ seconds_remaining;
      if find(matchup,'vs') then Home_play=1;
            else Home_play=0;
      game_year=Year(game_date);
      game_month=Month(game_date);
      game_quarter=QTR(game_date);
      season_p2 = substr(season,6,2);
      if minutes_remaining * 60+ seconds_remaining<3 then End_Three_Sec=1;
            else End_Three_Sec=0;
      if loc_y=0 then angle=0;
            else angle =atan(loc_x/abs(loc_y));
      action_type_first_word=scan(action_type,1,' ');
      action_type_last_word =scan(reverse(Tranwrd(Tranwrd(action_type,'shot',''),
'Shot','')),1,' ');

      /*matchup = translate(matchup,'@','vs.');*/ /*Replace @ to vs.*/
run;

/*Checking missing value*/
proc means data=KobeData Min Q1 Mean Q3 Max nmiss n;
run;

/*get training set*/
data train;
      set KobeData;
      if shot_made_flag_new ^=.;
run;

/*get Test set*/
data test;
      set KobeData;
      if shot_made_flag_new =.;
Run;
```

Exploratory Analysis and Research question 1-5:

```
/**** Exploratory Analysis and Data Cleaning ***************/
data KobeData_exp;
set KobeData;
informat shot_made_flag_new $2.;run;
proc sgplot data=KobeData_exp;
scatter x=shot_distance y=shot_made_flag_new; run;
/*split to groups, scatter matrix*/
```

```
proc sgscatter data=KobeData_exp;
  matrix lat loc_x loc_y lon / group=shot_made_flag_new;
run;
 proc sgscatter data=KobeData_exp;
  matrix Second_to_period_end shot_distance attendance arena_temp avgnoisedb/
group=shot_made_flag_new;
run;
/*******************************influence points plot
**********************************************/
ods graphics on;
proc logistic data = KobeData  ;
class playoffs(ref="0") home_play(ref="0") / param = ref;
model shot_made_flag_new = shot_distance playoffs Home_play
;
output out=dinf prob=p resdev=dr h=pii reschi=pr difchisq=difchi;
run;
 ods graphics off;
quit;
goptions reset = all;
symbol1 pointlabel = ("#shot_id" h=1 )  value=none;
proc gplot data = dinf;
  plot difchi*p;
run;
quit;

/**************************question 1-4 can be answered by using the code
below************************************************/
/*loc_x,loc_y and lat, lon, can only keep one pair,will keep loc_x,loc_y*/
data KobeData_1(drop=lat lon);
set KobeData_exp;
run;
proc logistic data = KobeData_1 PLOTS = ALL ;
class action_type combined_shot_type game_event_id game_id period playoffs(ref="0")season
shot_made_flag_new(ref="0") shot_type
        shot_zone_area shot_zone_basic shot_zone_range game_date Home_play(ref="0")
opponent shot_id/ param = ref;
model shot_made_flag_new = shot_distance playoffs Home_play/ lackfit ctable;
effectplot slicefit;
run;

/**********************************question 5 proportion
comparison********************************************/
data clutch_q5(keep= period Second_to_period_end shot_made_flag_new);
set train;
where Second_to_period_end <180;run;
data noclutch_q5(keep= period Second_to_period_end shot_made_flag_new);
set train;
where Second_to_period_end >180;run;
proc means data = clutch_q5;
class period;
var shot_made_flag_new;run;
proc means data = noclutch_q5;
class period;
var shot_made_flag_new;run;
```

```
/*************************************************************************************/
/************************************logistic
regression*********************************************/
data KobeData_log;
set KobeData_1;/*without lon and lat*/
run;
ods graphics on;
/*logistic regression*/
proc logistic data = KobeData_log outmodel=results outest=betas plots =all covout;
class action_type combined_shot_type  shot_zone_range season_p2 shot_zone_area period
shot_zone_basic End_Three_Sec(ref="0")
      ;
model shot_made_flag_new (event='1') = action_type  End_Three_Sec shot_zone_range
season_P2 shot_zone_area      arena_temp  period shot_zone_basic attendance loc_y
game_year
                                    shot_distance*action_type
                                    period*Second_to_period_end
                                          loc_y*shot_distance
                                          shot_zone_basic*shot_zone_area
            /lackfit ctable;
score out=KobePredictLog fitstat;
run;
ods graphics off;
/*********************************logloss
calculation*********************************************/
data logloss;
set KobePredictLog;
where shot_made_flag_new=.; /*only the test data*/
if P_1 = 0 then P_1_new  = 0.000001;
      else if P_1 = 1 then P_1_new  = 0.9999999999; /*in case there is log(0)*/
      else if P_1 = . then P_1_new  = 0.5;/*missing value =0.5*/
      else P_1_new = P_1;
if I_shot_made_flag_new = . then I_shot_made_flag_new  = 0.5;/*missing value =0.5*/
lossfuc= I_shot_made_flag_new*log(P_1_new) +(1-I_shot_made_flag_new)*log(1-P_1_new);
run;
proc means data =logloss;
var lossfuc;run;
/*********************************calculate percentage of
prediction*********************************/
data log_submit_perc(keep = shot_id I_shot_made_flag_new P_1);
set KobePredictLog;
where shot_made_flag_new=.;
informat I_shot_made_flag_new 2.;
if P_1 = 0 then P_1_new  = 0.000001;
      else if P_1 = 1 then P_1_new  = 0.9999999999;
      else if P_1 = . then P_1_new  = 0.5;
      else P_1_new = P_1;
if I_shot_made_flag_new = . then I_shot_made_flag_new  = 0.5;
run;
data log_submit;
      set log_submit_perc;
      if P_1 = . then P_1 =0.5;
            else P_1= P_1;
```

```
run;
proc means data = log_submit Mean ;run;
/*write to local*/
Proc export data=log_submit(keep = shot_id P_1 rename=(P_1=shot_made_flag))
outfile='H:\MSDS6372 stats2\project2\KobeLogistic.csv'
DBMS=CSV Replace;quit;
/******************************************to see
sensitivity*********************************************/
data sensitivity(keep = shot_id F_shot_made_flag_new I_shot_made_flag_new sen_count);
set KobePredictLog;
/*if there is any missing value, just let it to be 0 so it won't be included in the
calculation*/
if I_shot_made_flag_new = . then I_shot_made_flag_new  = 0;
if F_shot_made_flag_new = . then F_shot_made_flag_new  = 0;
if F_shot_made_flag_new=1 and I_shot_made_flag_new =1 then sen_count =1;
      else sen_count =0;
run;
Proc export data=sensitivity
outfile='H:\MSDS6372 stats2\project2\KobeSensi.csv'
DBMS=CSV Replace;quit;

/*********************************to see
speciality***************************************/
data specialicity(keep = shot_id F_shot_made_flag_new I_shot_made_flag_new spec_count);
set KobePredictLog;
if F_shot_made_flag_new=0 and I_shot_made_flag_new =0 then spec_count =1;
else spec_count =0; run;
Proc export data=specialicity
outfile='H:\MSDS6372 stats2\project2\KobeSpec.csv'
DBMS=CSV Replace;quit;


/*Cluster model SAS code:  */
/* standard procedure to bring the continuous variables to the same level */
proc standard data = KobeData out = clustvar mean =0 std =1;
var loc_x loc_y shot_distance arena_temp avgnoisedb Second_to_period_end lat lon ;
run;
proc print data=clustvar (obs=10);
Run;
proc means data=clustvar Min Q1 Mean Q3 Max nmiss n;
run;

%macro kmean(K);

/* k-means cluster / partition */
proc fastclus data = clustvar out=outdata&K.  outstat = clusterStat&K. maxclusters= &K.
maxiter=300;
  var loc_x loc_y shot_distance arena_temp avgnoisedb Second_to_period_end lat lon
      ;
  run;
%mend;

%kmean(1);
%kmean(2);
```

```
%kmean(3);
%kmean(4);
%kmean(5);
%kmean(6);
%kmean(7);
%kmean(8);
%kmean(9);

data clust1;
set clusterStat1;
nclust=1;

if _type_ = 'RSQ';
keep nclust over_all;
run;



 data clust2;
set clusterStat2;
nclust=2;

if _type_ = 'RSQ';
keep nclust over_all;
run;
data clust3;
set clusterStat3;
nclust=3;

if _type_ = 'RSQ';
keep nclust over_all;
run;
data clust4;
set clusterStat4;
nclust=4;

if _type_ = 'RSQ';
keep nclust over_all;
run;
data clust5;
set clusterStat5;
nclust=5;

if _type_ = 'RSQ';
keep nclust over_all;
run;
data clust6;
set clusterStat6;
nclust=6;

if _type_ = 'RSQ';
keep nclust over_all;
run;
data clust7;
```

```
set clusterStat7;
nclust=7;

if _type_ = 'RSQ';
keep nclust over_all;
run;
data clust8;
set clusterStat8;
nclust=8;

if _type_ = 'RSQ';
keep nclust over_all;
run;
data clust9;
set clusterStat9;
nclust=9;

if _type_ = 'RSQ';
keep nclust over_all;
run;

data clusrsquare;
set clust1 clust2 clust3 clust4 clust5 clust6 clust7 clust8 clust9;
run;

proc print data =clusrsquare;
run;

/* plot elbow curve using r-square values;*/
symbol1 color=blue interpol=join;
proc gplot data = clusrsquare;
plot over_all * nclust;
run;


/**/
proc candisc data = outdata4 out=clustcan;
class cluster;
var loc_x loc_y shot_distance arena_temp avgnoisedb Second_to_period_end home_play;
run;

proc sgplot data = clustcan;
scatter y=can2 x=can1 / group =cluster;
run;

proc logistic data = outdata4;
class  action_type combined_shot_type  shot_zone_range season_p2 shot_zone_area period
shot_zone_basic End_Three_Sec(ref="0") shot_type;
Model shot_made_flag_new  (ref ='0') =  action_type combined_shot_type  shot_zone_range
season_p2 shot_zone_area period shot_zone_basic End_Three_Sec shot_type cluster
            / selection = stepwise;
run;
Quit;
```

```
proc logistic data = outdata4 outmodel=results plots =all;
class  action_type combined_shot_type  shot_zone_range season_p2 shot_zone_area period
shot_zone_basic End_Three_Sec(ref="0") ;
model shot_made_flag_new  (ref ='0') =  action_type combined_shot_type  shot_zone_range
season_p2 shot_zone_area period shot_zone_basic End_Three_Sec  cluster
action_type*combined_shot_type
shot_zone_basic*shot_zone_area
/ lackfit ctable;
score out=KobePredictLog fitstat;
run;

data logloss;
set KobePredictLog;
where shot_made_flag_new=.; /*only the test data*/
if P_1 = 0 then P_1_new  = 0.000001;
      else if P_1 = 1 then P_1_new  = 0.9999999999; /*in case there is log(0)*/
      else if P_1 = . then P_1_new  = 0.5;/*missing value =0.5*/
      else P_1_new = P_1;
if I_shot_made_flag_new = . then I_shot_made_flag_new  = 0.5;/*missing value =0.5*/
lossfuc= I_shot_made_flag_new*log(P_1_new) +(1-I_shot_made_flag_new)*log(1-P_1_new);
run;
proc means data =logloss;
var lossfuc;run;
/***********************************calculate percentage of
prediction*********************************/
data log_submit_perc(keep = shot_id I_shot_made_flag_new p_1);
set KobePredictLog;
where shot_made_flag_new=.;
informat I_shot_made_flag_new 2.;
if P_1 = 0 then P_1_new  = 0.000001;
      else if P_1 = 1 then P_1_new  = 0.9999999999;
      else if P_1 = . then P_1_new  = 0.5;
      else P_1_new = P_1;
if I_shot_made_flag_new = . then I_shot_made_flag_new  = 0.5;
run;
data log_submit_perc(keep = shot_id I_shot_made_flag_new P_1);
set KobePredictLog;
where shot_made_flag_new=.;
informat I_shot_made_flag_new 2.;
if P_1 = 0 then P_1_new  = 0.000001;
      else if P_1 = 1 then P_1_new  = 0.9999999999;
      else if P_1 = . then P_1_new  = 0.5;
      else P_1_new = P_1;
if I_shot_made_flag_new = . then I_shot_made_flag_new  = 0.5;
run;
data log_submit;
      set log_submit_perc;
      if P_1 = . then P_1 =0.5;
            else P_1= P_1;
run;
proc means data = log_submit Mean ;run;
/*write to local*/
Proc export data=log_submit(keep = shot_id P_1 rename=(P_1=shot_made_flag))
outfile='C:\Users\yubin\OneDrive\MyWork\SMU\MSDS6372\Unit14Project\KobeCluster.csv'
```

```
DBMS=CSV Replace;quit;

Proc export data=KobePredictLog(keep = shot_id p_1 RENAME=(p_1=shot_made_flag))
outfile='C:\Users\yubin\OneDrive\MyWork\SMU\MSDS6372\Unit14Project\KobeClusterKaggle.csv'
DBMS=CSV Replace;quit;

/*****************************************to see
sensitivity*********************************************/
data sensitivity(keep = shot_id F_shot_made_flag_new I_shot_made_flag_new sen_count);
set KobePredictLog;
/*if there is any missing value, just let it to be 0 so it wont be included in the
calculation*/
if I_shot_made_flag_new = . then I_shot_made_flag_new  = 0;
if F_shot_made_flag_new = . then F_shot_made_flag_new  = 0;
if F_shot_made_flag_new=1 and I_shot_made_flag_new =1 then sen_count =1;
      else sen_count =0;
run;
Proc export data=sensitivity
outfile='C:\Users\yubin\OneDrive\MyWork\SMU\MSDS6372\Unit14Project\KobeClusterSensi.csv'
DBMS=CSV Replace;quit;

/********************************to see
specialicity****************************************/
data specialicity(keep = shot_id F_shot_made_flag_new I_shot_made_flag_new spec_count);
set KobePredictLog;
if F_shot_made_flag_new=0 and I_shot_made_flag_new =0 then spec_count =1;
else spec_count =0; run;
Proc export data=specialicity
outfile='C:\Users\yubin\OneDrive\MyWork\SMU\MSDS6372\Unit14Project\KobeClusterSpec.csv'
DBMS=CSV Replace;quit;
```

LDA model SAS code:

```
/**************************/
/*LDA MODEL(grouped) - CL*/
/**************************/

/*******************/
/*1.Data Preparation*/
/*******************/

/*Categorical Variables Transformation*/
data KobeData;
set KobeData;
/*Number Label*/

      /*combined_shot_type and action_type to N_Combined_action_type*/
      if Combined_shot_type = "Jump Shot" then
            do;
            if action_type = "Jump Shot" then N_Combined_action_type = 0;
            else N_Combined_action_type = 1;
            end;
      else if Combined_shot_type = "Layup" then
            do;
```

```
		if action_type = "Layup Shot" then N_Combined_action_type = 2;
		else if action_type = "Driving Layup Shot" then N_Combined_action_type = 3;
		else N_Combined_action_type = 4;
		end;
	else if Combined_shot_type = "Dunk" then N_Combined_action_type = 5;
	else if Combined_shot_type = "Tip Shot" then N_Combined_action_type = 6;
	else N_Combined_action_type = 7;
	*else if Combined_shot_type = "Hook Shot" then N_Combined_action_type = 7;
	*else if Combined_shot_type ="Bank Shot" then N_Combined_action_type = 8;

	/*combined_shot_type*/
	if Combined_shot_type = "Jump Shot" then N_Combined_shot_type = 0;
	else if Combined_shot_type = "Layup" then N_Combined_shot_type = 1;
	else if Combined_shot_type = "Dunk" then N_Combined_shot_type = 2;
	else if Combined_shot_type = "Tip Shot" then N_Combined_shot_type = 3;
	else if Combined_shot_type = "Hook Shot" then N_Combined_shot_type = 4;
	else if Combined_shot_type = "Bank Shot" then N_Combined_shot_type = 5;

	/*shot_zone_area*/
	if shot_zone_area = "Center(C)" then N_shot_zone_area = 0;
	else if shot_zone_area = "Right Side Center(RC)" then N_shot_zone_area = 1;
	else if shot_zone_area = "Right Side(R)" then N_shot_zone_area = 2;
	else if shot_zone_area = "Left Side Center(LC)" then N_shot_zone_area = 3;
	else if shot_zone_area = "Left Side(L)" then N_shot_zone_area = 4;
	else if shot_zone_area = "Back Court(BC)" then N_shot_zone_area = 5;

	/*shot_zone_basic*/
	if shot_zone_basic = "Mid-Range" then N_shot_zone_basic = 0;
	else if shot_zone_basic = "Restricted Area" then N_shot_zone_basic = 1;
	else if shot_zone_basic = "Above the Break 3" then N_shot_zone_basic = 2;
	else if shot_zone_basic = "In The Paint (Non-RA)" then N_shot_zone_basic = 3;
	else if shot_zone_basic = "Right Corner 3" then N_shot_zone_basic = 4;
	else if shot_zone_basic = "Left Corner 3" then N_shot_zone_basic = 5;
	else if shot_zone_basic = "Backcourt" then N_shot_zone_basic = 6;

	/*season*/
	if season = "1996-97" then N_season = 0;
	else if season = "1996-97" then N_season = 1;
	else if season = "1997-98" then N_season = 2;
	else if season = "1998-99" then N_season = 3;
	else if season = "1999-00" then N_season = 4;
	else if season = "2000-01" then N_season = 5;
	else if season = "2001-02" then N_season = 6;
	else if season = "2002-03" then N_season = 7;
	else if season = "2003-04" then N_season = 8;
	else if season = "2004-05" then N_season = 9;
	else if season = "2005-06" then N_season = 10;
	else if season = "2006-07" then N_season = 11;
	else if season = "2007-08" then N_season = 12;
	else if season = "2008-09" then N_season = 13;
	else if season = "2009-10" then N_season = 14;
	else if season = "2010-11" then N_season = 15;
	else if season = "2011-12" then N_season = 16;
	else if season = "2012-13" then N_season = 17;
```

```
	else if season = "2013-14" then N_season = 18;
	else if season = "2014-15" then N_season = 19;
	else if season = "2015-16" then N_season = 20;

	/*Add shot_range_lower and shot_range_upper variables. Replaced categorical 
variable shot_zone_range. */
	if shot_zone_range = "8-16 ft." then
		do
			shot_range_lower = 8;
			shot_range_upper = 16;
		end;
	else if shot_zone_range = "16-24 ft." then
		do
			shot_range_lower = 16;
			shot_range_upper = 24;
		end;
	else if shot_zone_range = "Less Than 8 ft." then
		do
			shot_range_lower = 0;
			shot_range_upper = 8;
		end;
	else if shot_zone_range = "24+ ft." then
		do
			shot_range_lower = 24;
			shot_range_upper = 71;
		end;
	else if shot_zone_range = "Back Court Shot" then
		do
			shot_range_lower = 71;
			shot_range_upper = 94;
		end;
run;

/*Check Point*/
/*
proc print data=KobeData (obs=10);run;
proc contents data=KobeData;run;
proc means data=KobeData Min Q1 Mean Q3 Max nmiss n;run;
*/

/**********************/
/*2.Assumption Checking*/
/**********************/

data train;
set KobeData;
if shot_made_flag_new^=.;
run;

proc sort data=train;
by shot_made_flag_new;run;quit;

/*Check ellips for Homoscedasticity assumption*/
proc sgscatter data=train;
```

```
by shot_made_flag_new;
matrix
loc_x loc_y shot_distance attendance avgnoisedb arena_temp game_id Second_to_period_end
/ellipse=(type = mean alpha =.05);
run;quit;

/*Check ChiSq for Homoscedasticity assumption*/
proc discrim data=train pool=test;
                        class shot_made_flag_new;

                        var loc_x loc_y shot_distance attendance avgnoisedb arena_temp
Second_to_period_end
                        ; run; quit;

/*Run PCA to get non-correlated variances*/
proc princomp data = KobeData out = PCAKobeData std;
var loc_x loc_y shot_distance attendance avgnoisedb arena_temp game_id
Second_to_period_end;
run;
quit;

data Train;
      set PCAKobeData;
      if shot_made_flag_new ^=.;
run;

/*Get prior information*/
proc freq data = train;
table shot_made_flag_new; run; quit;

proc print data=PCAKobeData (obs=10); run;

/***********************/
/*3. Building LDA Model*/
/***********************/

/*Use macro to create train and test datasets in loop and run LDA models. Merge all
results*/
%macro LDAData;
/*Initializing results datasets*/
data TestClassify;
      set PCAKobeData;
      if shot_made_flag_new =.;
run;

data TrainClassify;
      set PCAKobeData;
      if shot_made_flag_new ^=.;
run;
/*Building models*/
%local i j k;
ods graphics off;
ods exclude all;
ods noresults;
```

```
%do i=0 %to 7;
      %do j=0 %to 1;
            %do k=0 %to 1;
                  /*Grouping data by categorical variables*/
                  data train;
                        set PCAKobeData;
                        if N_Combined_action_type=&i & home_play=&j & playoffs=&k &
shot_made_flag_new ^=.;
                  run;
                  data test;
                        set PCAKobeData;
                        if N_Combined_action_type=&i & home_play=&j & playoffs=&k &
shot_made_flag_new =.;
                  run;
                  /*Run LDA on current group of data.*/
                  proc discrim data=train pool=test testdata = test testout =
shotTestClassify out=shotTrainClassify crossvalidate;
                  class shot_made_flag_new;
            var Prin1-Prin8;
                  priors "0"=.55 "1"=.45;
                        /*var loc_x loc_y shot_distance attendance avgnoisedb arena_temp
game_id Second_to_period_end*/
                   run; quit;

            /*Collect group results by merging results classify dataset: */
                  proc sort data=shotTestClassify;by shot_id;run;
                  proc sort data=TestClassify;by shot_id;run;
                  proc sort data=shotTrainClassify;by shot_id;run;
                  proc sort data=TrainClassify;by shot_id;run;
                  *Test dataset for predicting results;
                  data TestClassify;
                        merge TestClassify shotTestclassify; by shot_id; run;
                  *Train dataset for model evaluation;
                  data TrainClassify;
                        merge TrainClassify shotTrainclassify; by shot_id; run;
            %end;
      %end;
%end;
ods graphics on;
ods exclude none;
ods results;
%mend;

%LDAData;

/*Check Point*/
/*
proc print data = TestClassify(obs=10); run;
proc print data = TrainClassify(obs=10); run;
proc means data=TestClassify Min Q1 Mean Q3 Max nmiss n;run;
proc means data=TrainClassify Min Q1 Mean Q3 Max nmiss n;run;
proc sql;
select shot_id,action_type,combined_shot_type,N_Combined_action_type,Home_play,playoffs
from TrainClassify where _1=.;
```

```
run;quit;
*/

/********************/
/*4. LDA Final OutPut*/
/********************/

/*Output Prediction data*/
data KobeTestOut (KEEP=shot_id _1 RENAME=(_1=shot_made_flag));
   set TestClassify;
 run ;

/* Export to file for summit */
proc export data=KobeTestOut
outfile='C:\SAS_Files\datatable2\MSDS6372_Project2\LDATestFinal_g.csv'
dbms=csv replace;
delimiter=',';
run;

/*Output Model evaluation data*/
data KobeTrainOut (KEEP=shot_id shot_made_flag_new _1 _INTO_ RENAME=(_1=predict
_INTO_=into));
   set TrainClassify;
 run ;

/* Export to file for evaluation */
proc export data=KobeTrainOut
outfile='C:\SAS_Files\datatable2\MSDS6372_Project2\LDATrainFinal_g.csv'
dbms=csv replace;
delimiter=',';
run;

/********************/
/*End of LDA Part -CL*/
/********************/
```

R CODE FOR evaluation

```
trainFinal<-
read.csv("C:\\SAS_Files\\datatable2\\MSDS6372_Project2\\LDATrainFinal_g.csv",header=TRUE,
sep=',')

#logloss
trainFinal$predict[is.na(trainFinal$predict)]<-0.45
trainFinal$predict[trainFinal$predict==0]<-0.000001
trainFinal$predict[trainFinal$predict==1]<-0.999999
trainFinal$logloss<-trainFinal$shot_made_flag_new*log(trainFinal$predict)+(1-
trainFinal$shot_made_flag_new)*log(1-trainFinal$predict)
logloss<--mean(trainFinal$logloss)
logloss

#AUC
#library(pROC)
```

```
#trainFinal<-
read.csv("C:\\SAS_Files\\datatable2\\MSDS6372_Project2\\LDATrainFinal.csv",,header=TRUE,s
ep=',')
response=trainFinal$shot_made_flag_new
predictor=trainFinal$predict
roc_obj <-roc(response, predictor)
auc(roc_obj)

#Mis-Classification Rate
#trainFinal<-
read.csv("C:\\SAS_Files\\datatable2\\MSDS6372_Project2\\LDATrainFinal.csv",,header=TRUE,s
ep=',')
misclass<-
nrow(trainFinal[(trainFinal$shot_made_flag_new!=trainFinal$into)&(trainFinal$shot_made_fl
ag_new==1),])/nrow(trainFinal[trainFinal$shot_made_flag_new==1,])*0.5 +

nrow(trainFinal[(trainFinal$shot_made_flag_new!=trainFinal$into)&(trainFinal$shot_made_fl
ag_new==0),])/nrow(trainFinal[trainFinal$shot_made_flag_new==0,])*0.5
misclass

#sensitivity
#trainFinal<-
read.csv("C:\\SAS_Files\\datatable2\\MSDS6372_Project2\\LDATrainFinal.csv",,header=TRUE,s
ep=',')
sensitivity<-
nrow(trainFinal[(trainFinal$shot_made_flag_new==trainFinal$into)&(trainFinal$shot_made_fl
ag_new==1),])/nrow(trainFinal[trainFinal$shot_made_flag_new==1,])*100
sensitivity

#Specificity
#trainFinal<-
read.csv("C:\\SAS_Files\\datatable2\\MSDS6372_Project2\\LDATrainFinal.csv",,header=TRUE,s
ep=',')
Specificity<-
nrow(trainFinal[(trainFinal$shot_made_flag_new==trainFinal$into)&(trainFinal$shot_made_fl
ag_new==0),])/nrow(trainFinal[trainFinal$shot_made_flag_new==0,])*100
Specificity
```