**MATH 4432 Statistical Machine Learning Project 1**

**Classification of Football Betting Match Results**

By WU, Sen (20271863) & CHAN, Wing Kin [Collaborator] on 15 Mar. 2018

**Introduction**

This project is inspired by one of the best friends of one of the teammates. He realized that it is possible to predict the handicap results of football matches by simply reading the Home/Away/Draw odds, Home/Away handicap odds and change of odds before the match.

In fact, bookmakers over the world are in fact, generating profit over the years. The actuaries behind each bookmaker play crucial role in risk management. They make predictions of match results and suggest reasonable and attractive odds to boost the bets. They could make generous profits controlling the odds when some instances of major upsets happen.

In each football match, there are so many variables that could affect the match results. A lot of variables has been considered by actuaries before the match. Actuaries set the odds for making profit, thus we could do reverse engineering by odds from actuaries.

**Data Set**

We will only choose the match result data from some famous European leagues. We will take the match results in past 5-6 years (From Season 2011-12 to 2016-17) from England Premier, German Division 1, Italian Division 1, French Division 1 and Spanish Division 1 as training data. There is a total of 9471 observations for training. We will take the match results of the above five leagues generated in current season (2017-18) as test data. As of 6 Mar 2018, the number of training data is above 1000.

All data are obtainable from <http://www.football-data.co.uk/downloadm.php>. For data each year, it contains the match results and odds not only for the five leagues aforementioned, but also some lower-division leagues. We first download the Excel Workbook file .xls, then find the data of those five leagues from sheets named “E0”, “D1”, “I1”, “F1” and “SP1” respectively. The column “Referee” is removed from “E0” such that all five leagues have same variables. For data of Season 2014-15, apart from “Referee” column in “E0”, columns “SJH”, “SJD” and “SJA” for each of the leagues are also removed.

The detailed legend of data can be found at <http://www.football-data.co.uk/notes.txt>. We will try to find the interrelationship between a variety of factors that affects the game results, not only for odds, but also some important statistical data in football match, such as number of yellow and red cards, fouls, offsides etc.

**Analysis**

Unlike many other classification problems, the percentage of correct predictions does not reflect the true profitability a model. For example, it is not strange if a model predicts the match results of low-odd items with very high percentage of correctness. We hope this model could predict some incidences when some major upsets happen, which could be major source of profit generating.

For instance, we are interested in Handicap predictions. Handicap market has two options for gamblers to choose, Home and Away. The match results have been adjusted by a handicap value judged by the strength of the two teams. Say, if home team is much more competitive than the away team, the home team will receive a penalty handicap value, which means the final score of the team will be subtracted by a number such that the match seems fairer. As a result, the handicap odds are often attractive, and the handicap market are often harder to predict, as stronger teams are given disadvantages in this pool.

For instance, in our data, there is a column named BbAHh indicating the handicap size for home team. We found the following example for illustration why handicap market is so popular. In Spanish Division 1 match, Barcelona vs. Granada on 9 Jan. 2016, the home team Barcelona received an average handicap of –3.5, which means Barcelona need to win with goal difference of four such that gamblers bought Handicap Barcelona win could win the bet. This makes more exciting betting experience to gamblers.

Another important factor we need to consider is the partial betting mechanism for handicap market. The partial betting occurs when the bookmakers set a handicap value of 0.25 + 0.5*n*, where *n* is any integer. Such handicap value is equivalent to set two handicap values, which are given by 0.25 + 0.5*n* ± 0.25. The bet will be equally divided in two portions, one portion for each handicap value. For example, if the match correct score is 2:1 and the **home** handicap size is –0.75 (*n* = –2), consider a bet bought handicap home win. The bet is split into two portions: one for handicap –0.5 and another for handicap –1. Consider the handicap –0.5 bet, since 2 – 0.5 > 1, this bet is win. However, for the –1 bet, 2 – 1 = 1, which means this bet is draw and the bet will be refunded. Finally, the actual odd applied in dividend is given by:

True Odd = (Odd – 1) / 2 + 1

The true odd in the above case means that the bet for handicap home received a **partial win**. However, if someone bought handicap away win in this case, the bet will be split into two portions. One for handicap home –0.5, where 1 < 2 – 0.5 means this portion is lost. Another for handicap home –1, where 1 = 2 – 1 means this portion is draw and will be refunded. Thus, the final actual odd applied in the dividend is given by:

True Odd = 0.5

In this case, the bet for away win is classified as **partial lose**. In short, we could apply the following expression to determining the payout odds:



We applied a three-step test in our .csv file by using Excel to identify the paying odds of handicap home-win bets:

1. Draw Test

We first filter out the handicap draw cases as it will affect the future tests. We write the following formula in the handicap payout cell in Excel:

= IF(E2 + BE2 = F2, 1, *Win/Lose Test*)

where column E and F stores the Home Goal (HG) and Away Goal (AG) data, column BE contains the handicap size data applied to home team. We will assign a paying odd value of 1 when the handicap result is draw.

1. Win/Lose Test

After filtered all draw cases, we are going to conduct the Win/Lose Test by determining whether the home goal with handicap value applied is larger than that of away goal. If home wins, we could conduct Win All/Partially test. If home loses, we could conduct Lose All/Partially test:

IF(E2 + BE2 > F2,*Win A/P Test*,*Lose A/P Test*)

We assign the rest of the odd values by the Win A/P test and Lose A/P test.

1. Win/Lose All/Partially Test

We apply additional disadvantage handicap of –0.5 to home team to see if the home team’s handicap victory holds unbeaten. In another word, we will classify the match result as “home win partially” if home loses when additional disadvantage handicap of –0.5 is applied. The following is the formula of Win A/P test:

IF(E2 + BE 2 - 0.5 >= F2, BG2, (BG2 - 1)/2 + 1)

where the column BG contains the handicap winning odd for home team. Similarly, we can write the Lose A/P test code by applying additional advantage handicap of +0.5 to home team and check if away team’s handicap victory holds unbeaten:

IF(E2 + BE2 + 0.5 <= F2, 0, 0.5)

Finally, the code assigning paying odds for home-win bets is given by:

=IF(E2 + BE2 = F2, 1, IF(E2 + BE2 >F2, IF(E2 + BE2 - 0.5 >= F2, BG2, (BG2 - 1)/2 + 1), IF(E2 + BE2 + 0.5 <= F2, 0, 0.5)))

Similarly, we can assign the paying odds for away-win bets by the following code:

=IF(E2 + BE2 = F2, 1, IF(E2 + BE2 >F2, IF(F2 - BE2 + 0.5 <= E2, 0, 0.5), IF(F2 - BE2 - 0.5 >= E2, BI2, (BI2 - 1)/2 + 1)))

where column BI contains the handicap winning odd for away team. Later when processing the data using R, we shall calculate the following to evaluate the profitability:

> Profit = mean(*Paying\_odd* – 1)

Where the *Paying\_odd* is determined by the choice by the model trained on the test data.

Next, we should define the *correct choice* of the machine-learning algorithm. When a gambler is dedicated to bet in handicap market, they are given the odds and handicap size. They will choose one from the two available selections, home team and away team. Their choices decide the final paying odds. Thus, we need to define the correct choice in the training data, so that the algorithm can make Home/Away decision in the test data. Sometimes it is possible that no matter which side the gambler bets, the bet will be refunded. They are called “draw” cases. We will randomly assign the correct choice “Home” or “Away” to each of the “draw” case for sake of less statistical bias. The formula of correct choice in the training data is shown below:

=IF(BJ2 > 1, "H", IF(BJ2 < 1, "A", IF(RAND() >= 0.5, "H", "A")))

This ensures that the Home and Away results are evenly distributed in the draw cases. Up to this stage, we finished processing the training data files and we are free to begin our training and testing.

**Procedures**

Firstly, we import our .csv files into the R workspace environment:

> train = read.csv("C:/Users/Globas Wu/iCloudDrive/HKUST Notes/MATH 4432 (Year 4 Spring)/Project/Project 1/Football\_data\_training.csv")

> test = read.csv("C:/Users/Globas Wu/iCloudDrive/HKUST Notes/MATH 4432 (Year 4 Spring)/Project/Project 1/Football\_data\_test.csv")

We also load the libraries we used in previous assignments, namely ISLR, MASS and boot. We will now assign the following data sets:

> Football.train <- train

> Football.test <- test[test$CC !="D",]

We removed the rows in the test data with draw handicap result. That is, we turn our problem into biclassification problem. For train data, as aforementioned, we assigned home/away randomly to those matches with draw results. By names(train), we obtains the names inside the training and test data set:

[1] "ï..Div" "Date" "HomeTeam" "AwayTeam"

[5] "FTHG" "FTAG" "FTR" "HTHG"

[9] "HTAG" "HTR" "HS" "AS"

[13] "HST" "AST" "HF" "AF"

[17] "HC" "AC" "HY" "AY"

[21] "HR" "AR" "B365H" "B365D"

[25] "B365A" "BWH" "BWD" "BWA"

[29] "IWH" "IWD" "IWA" "LBH"

[33] "LBD" "LBA" "PSH" "PSD"

[37] "PSA" "WHH" "WHD" "WHA"

[41] "VCH" "VCD" "VCA" "Bb1X2"

[45] "BbMxH" "BbAvH" "BbMxD" "BbAvD"

[49] "BbMxA" "BbAvA" "BbOU" "BbMx.2.5"

[53] "BbAv.2.5" "BbMx.2.5.1" "BbAv.2.5.1" "BbAH"

[57] "BbAHh" "BbMxAHH" "BbAvAHH" "BbMxAHA"

[61] "BbAvAHA" "HDCPOH" "HDCPOA" "CC"

[65] "PSCH" "PSCD" "PSCA"

We suspect the handicap market results is related to BbAH, BbAHh, BbMxAHH, BbAvAHH, BbMxAHA and BbAvAHA. In the following we will use four biclassification methods to determine Firstly, we perform the **logistic regression** to analyze the data. We used the following codes:

> fit.glm <- glm(CC ~ BbAH + BbAHh + BbMxAHH + BbAvAHH + BbMxAHA + BbAvAHA, data = Football.train, family = "binomial")

> summary(fit.glm)

The following is the summary of this logistic regression results:

Call:

glm(formula = CC ~ BbAH + BbAHh + BbMxAHH + BbAvAHH + BbMxAHA + BbAvAHA, family = "binomial", data = Football.train)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.511 -1.174 -1.004 1.174 1.514

Coefficients:

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

(Intercept) 1.0667043 2.2742014 0.469 0.6390

BbAH -0.0003065 0.0053590 -0.057 0.9544

BbAHh -0.0353516 0.0253111 -1.397 0.1625

BbMxAHH 1.2454150 0.9359638 1.331 0.1833

BbAvAHH -2.0381993 1.1822628 -1.724 0.0847 .

BbMxAHA -1.5095399 0.8909137 -1.694 0.0902 .

BbAvAHA 1.7552550 1.1310726 1.552 0.1207

---

Signif. codes:

0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 13130 on 9470 degrees of freedom

Residual deviance: 13092 on 9464 degrees of freedom

AIC: 13106

Number of Fisher Scoring iterations: 3

We used the following codes to check the percentage of correct predictions:

> probglm <- predict(fit.glm, Football.test, type = "response")

> pred.glm <- rep("A",length(probglm))

> pred.glm[probglm > 0.5] <- "H"

> mean(pred.glm != Football.test$CC)

The program shows the following results:

[1] 0.4625585

We found that the rate of correct prediction has a bias of around 3.7% from 50% expected rate of correctness. It means the highest percentage of correct prediction can approach around 53.7% if we buy the opposite team to that of the predicted team.

Next, we perform **linear discriminant analysis (LDA)** to the training data.

> HA.testdata <- Football.test$CC

> fit.lda <- lda(CC ~ BbAH + BbAHh + BbMxAHH + BbAvAHH + BbMxAHA + BbAvAHA, family = "binomial", data = Football.train)

> fit.lda

The following is the information of LDA fitting model:

Call:

lda(CC ~ BbAH + BbAHh + BbMxAHH + BbAvAHH + BbMxAHA + BbAvAHA, data = Football.train, family = "binomial")

Prior probabilities of groups:

A H

0.5012142 0.4987858

Group means:

BbAH BbAHh BbMxAHH BbAvAHH BbMxAHA BbAvAHA

A 24.49947 -0.3521698 2.005768 1.943592 1.987700 1.925338

H 24.48772 -0.3588061 1.988315 1.926880 2.004803 1.941901

Coefficients of linear discriminants:

LD1

BbAH -0.002466215

BbAHh -0.280892092

BbMxAHH 9.831357308

BbAvAHH -16.113965003

BbMxAHA -11.806660983

BbAvAHA 13.764021275

Then we construct confusion matrix applying fit.lda to the test data to check the effectiveness of LDA:

> pred.lda <- predict(fit.lda, Football.test)

> table(pred.lda$class, HA.testdata)

The following shows the confusion matrix:

HA.testdata

A D H

A 339 0 269

H 324 0 350

The percentage of correct predictions = 100%(689/1282) = 53.7%.

In the following we will use **quadratic discriminant analysis (QDA)** to the training data:

> fit.qda <- qda(CC ~ BbAH + BbAHh + BbMxAHH + BbAvAHH + BbMxAHA + BbAvAHA, data = Football.train)

> fit.qda

The following is the information of the QDA fitting:

Call:

qda(CC ~ BbAH + BbAHh + BbMxAHH + BbAvAHH + BbMxAHA + BbAvAHA,

data = Football.train)

Prior probabilities of groups:

A H

0.5012142 0.4987858

Group means:

BbAH BbAHh BbMxAHH BbAvAHH BbMxAHA BbAvAHA

A 24.49947 -0.3521698 2.005768 1.943592 1.987700 1.925338

H 24.48772 -0.3588061 1.988315 1.926880 2.004803 1.941901

Then we construct the confusion matrix:

> pred.qda <- predict(fit.qda, Football.test)

> table(pred.qda$class, HA.testdata)

The following is the confusion matrix generated:

HA.testdata

A D H

A 465 0 420

H 198 0 199

The percentage of correct predictions = 100%(664/1282) = 51.8%.

Next, we try ***k-*th nearest neighbour (KNN)** with values of *K* from 1 to 200 on the training data:

> HA.testdata <- factor(HA.testdata)

> attach(Football.train)

> train.X = cbind(BbAH, BbAHh, BbMxAHH, BbAvAHH, BbMxAHA, BbAvAHA)

> train.HA = CC

> attach(Football.test)

> test.X = cbind(BbAH, BbAHh, BbMxAHH, BbAvAHH, BbMxAHA, BbAvAHA)

> set.seed(1)

> for(i in 1:200){

+ knn.pred = knn(train.X, test.X, train.HA, k = i)

+ print(mean(knn.pred != HA.testdata))

+ }

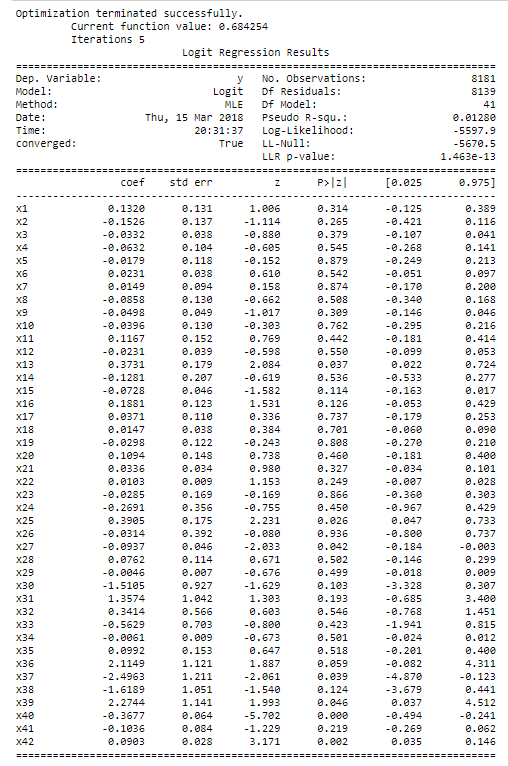
From the above code, we obtained 200 percentage of correct predictions from *K* = 1 to *K* = 200. The percentage with largest shift occurs at *K* = 12, which is shifted 3.4% away. In short, by adopting *only* the BbAH, BbAHh, BbMxAHH, BbAvAHH, BbMxAHA and BbAvAHA predictors, the highest shift attainable from **logistic regression**, **linear discriminant analysis (LDA)**, **quadratic discriminant analysis (QDA)** and ***k-*th nearest neighbour (KNN)** is around 3.7%, which is a bit unsatisfactory because the percentage means it is still quite close to purely guessing the match results. As a result, more advanced statistical learning methods should be adopted to further increase the percentage shift of correct predictions from 50%. However, at this stage we do expect there exists a limit of the highest available shift due to the lack of predictors. The presence of more predictors other than the odds would make the percentage of correct predictions differ further from 50%.

Now, we attempt to include other non-trivial predictors for predicting the result.

We exclude all parameters that will not be available before the start of the match (e.g. "AS", "HS", "AF", "HF") and used logistic regression to retrieve the significance (p-value) of different predictors.

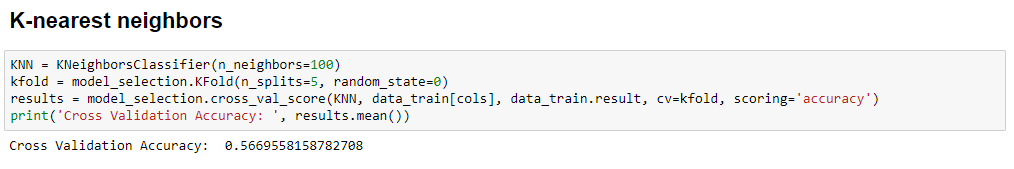
The "draw" results are removed to simplify the problem into a binary classification problem.

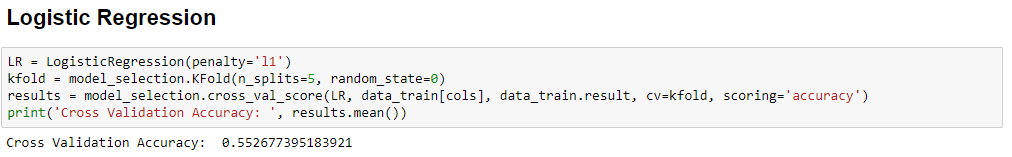
Below shows the logistic regression report for predictors selection:

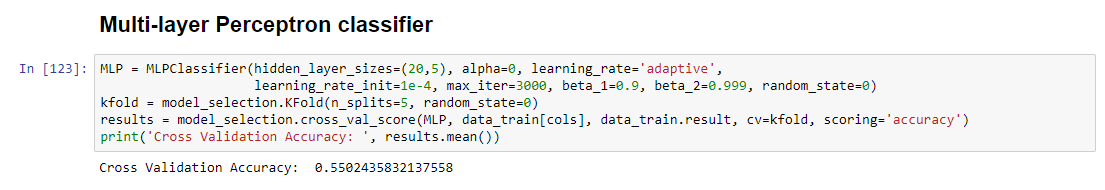


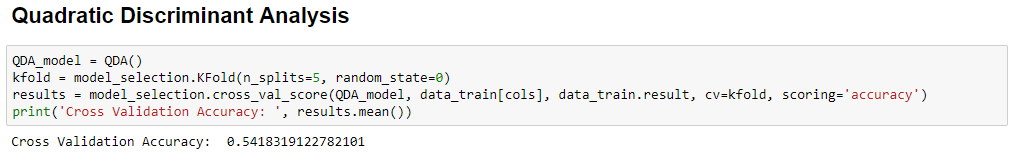
By examine the p-value, the predictors with p-value smaller than 0.05 are selected, which includes (i) 'BbMxAHH', (ii) 'BbAvAHH', (iii) 'BbAvAHA', (iv) 'PSCH', (v) 'PSCA' and (vi) 'BbMxA'.

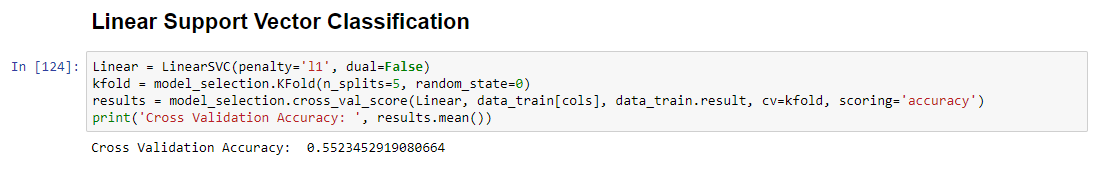
After selecting significant predictors, we used these predictors to fit multiple models.







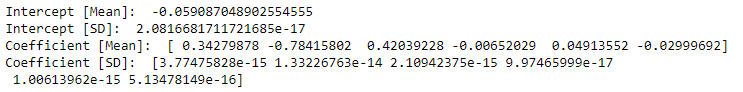




For models with hyperparameters, they are tuned to optimize the performance of different models. For the above models, their performances are similar, with K-nearest neighbour with the best performance – 56.7% accuracy on the testing data, following up with Logistic Regression – 55.3%. To further determine the accuracy of the model, bootstrap method is adopted. In here, we are going to examine how sensitive the Logistic Regression is with bootstrap method.



With the above code, we resample from the training data with replacement for 1000 times to fit the logistic regression model. After fitting with the logistic regression model, the coefficients are extracted. Their mean and standard deviation is calculated to estimate consistency of the fitted model.



From the bootstrap estimation of coefficient, the standard deviation is low for all predictors, which indicates the stability of the model. So, it indicates that logistic regression could be a promising model for predicting the outcome of the result.

**Conclusion**

For the prediction of outcome of matches, over 50% accuracy brings the possibility of gaining profits from betting matches.

With limited predictors and logistic regression model, we are already able to attain an approximately 56% accuracy.

In the above analysis, "draw" is omitted to simplify the prediction

As the number of predictors could be expanded to improve the accuracy of the model, further investigation will be conducted to improve the profitability of the model.