**MA – 5790 Predictive Modelling**

**FIFA 2017 – Code**

# installing the packages

installNewPackage <- function(packageName) {

if(packageName %in% rownames(installed.packages()) == FALSE)

{

install.packages(packageName, repos = "http://cran.us.r-project.org", dependencies=TRUE)

}

}

installNewPackage("Amelia")

installNewPackage("wordcloud")

installNewPackage("caret")

installNewPackage("corrplot")

installNewPackage("moments")

installNewPackage("e1071")

# Calling the libraries

library(Amelia)

library(wordcloud)

library(caret)

library(corrplot)

library(moments)

library(e1071)

# Read the data

fifa\_2017 <- read.csv(file = "data/FullData.csv", na.strings=c("","NA"))

# Check the dimension

dim(fifa\_2017)

# Setting the seed

set.seed(1)

# Getting the column names

col\_name <- colnames(fifa\_2017)

# Generate random numbers for frequency for the word cloud

col\_name\_values <- 1

# Create the Word cloud for all variables

word\_cloud <- data.frame(col\_name, col\_name\_values)

wordcloud(word\_cloud$col\_name, freq = word\_cloud$col\_name\_values, random.order = TRUE, random.color = FALSE, scale = c(1.25, 0.5), col = brewer.pal(9, "Oranges"))

# Setting the seed

set.seed(1)

# Filtering the dataset

subset\_colclasses <- function(DF, colclasses="numeric") {

DF[,sapply(DF, function(vec, test) class(vec) %in% test, test=colclasses)]

}

fifa\_2017\_factor <- subset\_colclasses(fifa\_2017, c("factor"))

fifa\_2017\_integer <- subset\_colclasses(fifa\_2017, c("integer", "num"))

# Getting the column names

col\_name\_factor <- colnames(fifa\_2017\_factor)

col\_name\_integer <- colnames(fifa\_2017\_integer)

# Generate random numbers for frequency for the word cloud

col\_name\_factor\_values <- 1

col\_name\_integer\_values <- 1

# Create the Word cloud for all variables

word\_cloud\_factor <- data.frame(col\_name\_factor, col\_name\_factor\_values)

wordcloud(word\_cloud\_factor$col\_name\_factor, freq = word\_cloud\_factor$col\_name\_factor\_values, random.order = TRUE, random.color = FALSE,

scale = c(3.4, 0.5), col = brewer.pal(9, "Oranges"))

# Word cloud for Continuous variable

word\_cloud\_integer <- data.frame(col\_name\_integer, col\_name\_integer\_values)

wordcloud(word\_cloud\_integer$col\_name\_integer, freq = word\_cloud\_integer$col\_name\_integer\_values, random.order = TRUE, random.color = FALSE,

scale = c(1.4, 0.5), col = brewer.pal(9, "Oranges"))

# Missmap

missmap(fifa\_2017, col = c("white", "brown3"), main = "Missing Map | Before Processing")

# Get the missing variables alone

missing\_count[missing\_count > 0]

```

# Imputation of missing values

# Remove unwanted predictors

fifa\_2017$National\_Position <- NULL

fifa\_2017$National\_Kit <- NULL

# Impute values by median

fifa\_2017$Club\_Kit = ifelse(is.na(fifa\_2017$Club\_Kit),ave(fifa\_2017$Club\_Kit, FUN = function(x) median(x, na.rm = TRUE)),fifa\_2017$Club\_Kit)

fifa\_2017$Contract\_Expiry = ifelse(is.na(fifa\_2017$Contract\_Expiry),ave(fifa\_2017$Contract\_Expiry, FUN = function(x) median(x, na.rm = TRUE)),fifa\_2017$Contract\_Expiry)

# Impute categorical values by mode

Mode <- function(x) {

ux <- unique(x)

ux[which.max(table(match(x, ux)))]

}

# Calling the corresponding impute function

fifa\_2017$Club\_Position = ifelse(is.na(fifa\_2017$Club\_Position), ave(fifa\_2017$Club\_Position, FUN = function(x) Mode(x)), fifa\_2017$Club\_Position)

fifa\_2017$Club\_Joining = ifelse(is.na(fifa\_2017$Club\_Joining), ave(fifa\_2017$Club\_Joining, FUN = function(x) Mode(x)),fifa\_2017$Club\_Joining)

# Conversion of Height and Weight to numerical values

fifa\_2017$Height <- as.numeric(gsub(fifa\_2017$Height,pattern = " cm",replacement = ""))

fifa\_2017$Weight <- as.numeric(gsub(fifa\_2017$Weight,pattern = " kg",replacement = ""))

# Check for missing map after processing

missmap(fifa\_2017, col = c("white", "darkolivegreen3"),main = "Missing Map | After Processing")

# Intialize variable

skew = list()

# Selecting only numerical values

fifa\_num = sapply(X = fifa\_2017,is.numeric)

# Calculating Skewness for the variables

skew = round(abs(apply(fifa\_2017[,fifa\_num], 2, skewness)), 3)

high\_skew <- list()

high\_skew\_value <- list()

par(mfrow=c(3,3))

# Select skewness values greater than 1 and plot histogram

for(i in 1:length(skew)){

if(skew[i] > 1){

high\_skew <- c(high\_skew, names(skew[i]))

high\_skew\_value <- c(high\_skew\_value, as.numeric(skew[i]))

hist(fifa\_2017[,names(skew[i])],main = paste(names(skew[i]), "|", skew[i]), xlab = names(skew[i]))

}

}

# Preprocessing the data and applying Box Cox transformation for highly skewed variables

fifa\_trans <- preProcess(x = fifa\_2017[unlist(high\_skew)],method = "BoxCox")

fifa\_data <- predict(fifa\_trans,newdata = fifa\_2017[unlist(high\_skew)])

# Calculating Skewness after Box Cox transformation and Plotting histogram for the transformed variables

skew\_measures <- matrix()

par(mfrow = c(3,3))

for (i in 1:ncol(fifa\_data)){

skew\_measures[i] <- round(skewness(fifa\_data[,i]), 3)

hist(fifa\_data[,i], main = paste(colnames(fifa\_data)[i], "|", skew\_measures[i]), xlab = colnames(fifa\_data)[i])

}

data.frame(cbind(column = colnames(fifa\_data), skew\_measures))

par(xpd = TRUE)

# Plot the correlation plot

corrplot(cor(fifa\_2017\_integer), order="hclust", tl.cex = 1)

# Finding out which predictors to elliminate since they have too large correlations

highCorr = findCorrelation(cor(fifa\_2017\_integer), cutoff = 0.90)

fifa\_2017\_integer\_rem <- fifa\_2017\_integer[,-highCorr]

removed\_col <- names(fifa\_2017\_integer)[highCorr]

# Matrix has no values > cutoff=0.9

corrplot(cor(fifa\_2017\_integer\_rem), order="hclust", tl.cex = 1)

# Print the list of variables removed

print("The variables that are removed after performing correlation check are")

print(removed\_col)