R Notebook

#Install packages here

#import libraries  
library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5 v purrr 0.3.4  
## v tibble 3.1.0 v dplyr 1.0.7  
## v tidyr 1.1.3 v stringr 1.4.0  
## v readr 1.4.0 v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(tidyr)  
library(dplyr)  
library(forecast)

## Registered S3 method overwritten by 'quantmod':  
## method from  
## as.zoo.data.frame zoo

library(broom)  
library(ggplot2)  
library(stargazer)

##   
## Please cite as:

## Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.

## R package version 5.2.2. https://CRAN.R-project.org/package=stargazer

library(mltools)

##   
## Attaching package: 'mltools'

## The following object is masked from 'package:tidyr':  
##   
## replace\_na

library(data.table)

##   
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':  
##   
## between, first, last

## The following object is masked from 'package:purrr':  
##   
## transpose

library(caret)

## Loading required package: lattice

##   
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':  
##   
## lift

library(fitdistrplus)

## Loading required package: MASS

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':  
##   
## select

## Loading required package: survival

##   
## Attaching package: 'survival'

## The following object is masked from 'package:caret':  
##   
## cluster

library("Hmisc")

## Loading required package: Formula

##   
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:dplyr':  
##   
## src, summarize

## The following objects are masked from 'package:base':  
##   
## format.pval, units

#Import dataset  
library(readr)  
data <- read\_csv("data2.csv")

##   
## -- Column specification --------------------------------------------------------  
## cols(  
## GRIP = col\_double(),  
## ARM = col\_double(),  
## RATINGS = col\_double(),  
## SIMS = col\_double()  
## )

head(data)

## # A tibble: 6 x 4  
## GRIP ARM RATINGS SIMS  
## <dbl> <dbl> <dbl> <dbl>  
## 1 106. 80.5 31.8 1.18  
## 2 106. 93 39.8 0.94  
## 3 94 81 46.8 0.84  
## 4 90.5 33.5 52.5 -2.45  
## 5 104 47.5 31.2 1   
## 6 171 126. 46.6 4.38

#Create new variables that will be used  
#Log form of token number  
#Count of categories  
#Count of supported currecies  
  
  
data\_x = data %>%  
 mutate(log\_ratings = log(RATINGS))  
#data\_x = na.omit(data\_x)  
#data\_x = data\_x  
  
head(data\_x)

## # A tibble: 6 x 5  
## GRIP ARM RATINGS SIMS log\_ratings  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 106. 80.5 31.8 1.18 3.46  
## 2 106. 93 39.8 0.94 3.68  
## 3 94 81 46.8 0.84 3.85  
## 4 90.5 33.5 52.5 -2.45 3.96  
## 5 104 47.5 31.2 1 3.44  
## 6 171 126. 46.6 4.38 3.84

#Get unique variables and their count  
cat("The platofrms used:", length(unique(data$GRIP)))

## The platofrms used: 102

cat("\n")

cat("The total number of countries:", length(unique(data$ARM)))

## The total number of countries: 96

cat("\n")

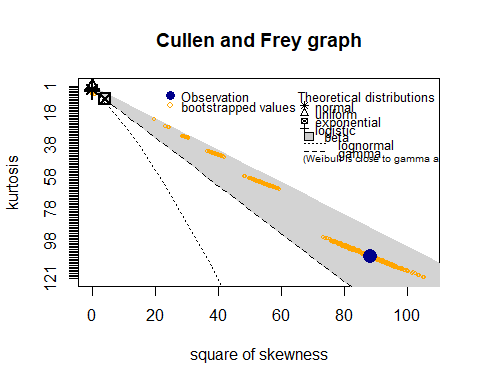
cat("The total number of token types is: ", length(unique(data$SIMS)))

## The total number of token types is: 136

#Getting Summary descriptive statistics of some key variables   
  
summary(data\_x)

## GRIP ARM RATINGS SIMS   
## Min. : 29.00 Min. : 19.00 Min. :21.60 Min. :-4.1700   
## 1st Qu.: 94.25 1st Qu.: 64.50 1st Qu.:34.80 1st Qu.:-0.9650   
## Median :111.50 Median : 81.50 Median :41.30 Median : 0.1600   
## Mean :115.01 Mean : 78.75 Mean :41.01 Mean : 0.2018   
## 3rd Qu.:125.50 3rd Qu.: 94.00 3rd Qu.:47.70 3rd Qu.: 1.0700   
## Max. :778.00 Max. :132.00 Max. :57.20 Max. : 5.1700   
## log\_ratings   
## Min. :3.073   
## 1st Qu.:3.550   
## Median :3.721   
## Mean :3.691   
## 3rd Qu.:3.865   
## Max. :4.047

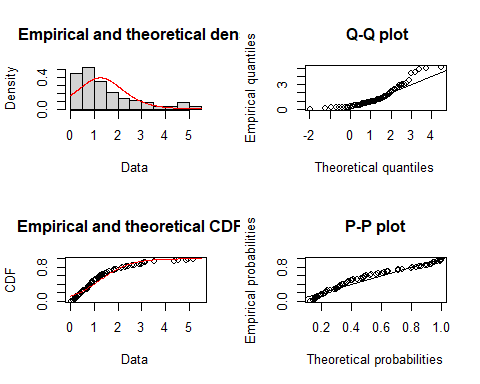
#Fitting and plotting distributions on some variables using Maximum Likelihood Estimation (MLE)  
# Log Normal distribution would be mostly targetted towards   
  
descdistribution <- function(data){  
 x = na.omit(data)  
 x = x[!is.infinite((x))]  
 x <- x[x > 0]  
 desc = descdist(x, boot = 1000)  
 print(summary(desc))  
}  
  
descdistribution(data\_x$GRIP)



## Length Class Mode   
## min 1 -none- numeric   
## max 1 -none- numeric   
## median 1 -none- numeric   
## mean 1 -none- numeric   
## sd 1 -none- numeric   
## skewness 1 -none- numeric   
## kurtosis 1 -none- numeric   
## method 1 -none- character

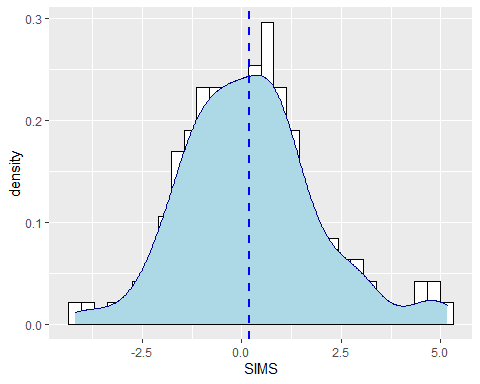
range01 <- function(x){(x-min(x))/(max(x)-min(x))}  
  
fitdistribution <- function(data){  
 x <- data[data > 0]  
 x = na.omit(x)  
 x = x[!is.infinite((x))]  
 fit <- fitdist(x, "logis")  
 print(summary(fit))  
 plot(fit)  
   
}  
  
fitdistribution(data\_x$SIMS)

## $start.arg  
## $start.arg$location  
## [1] 1.444605  
##   
## $start.arg$scale  
## [1] 0.6717385  
##   
##   
## $fix.arg  
## NULL  
##   
## Fitting of the distribution ' logis ' by maximum likelihood   
## Parameters :   
## estimate Std. Error  
## location 1.2568014 0.12631586  
## scale 0.6403893 0.06273465  
## Loglikelihood: -119.8129 AIC: 243.6257 BIC: 248.2872   
## Correlation matrix:  
## location scale  
## location 1.0000000 0.1242175  
## scale 0.1242175 1.0000000



rand\_x = na.omit(data\_x)  
rand\_x = rand\_x[!is.infinite(rowSums(rand\_x)),]  
  
# Basic density  
ggplot(rand\_x, aes(x = SIMS)) + geom\_histogram(aes(y=..density..), colour="black", fill="white")+  
 geom\_density(color="darkblue", fill="lightblue") + geom\_vline(aes(xintercept=mean(SIMS)),  
 color="blue", linetype="dashed", size=1)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



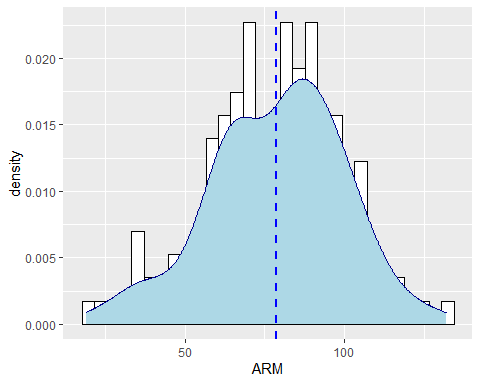
ggplot(rand\_x, aes(x = GRIP)) + geom\_histogram(aes(y=..density..), colour="black", fill="white")+  
 geom\_density(color="darkblue", fill="lightblue") + geom\_vline(aes(xintercept=mean(GRIP)),  
 color="blue", linetype="dashed", size=1)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## 

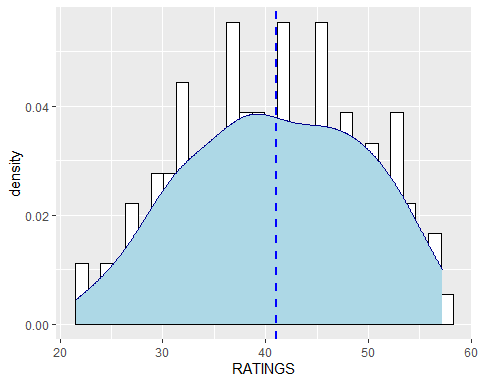
ggplot(rand\_x, aes(x = ARM)) + geom\_histogram(aes(y=..density..), colour="black", fill="white")+  
 geom\_density(color="darkblue", fill="lightblue") + geom\_vline(aes(xintercept=mean(ARM)),  
 color="blue", linetype="dashed", size=1)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



ggplot(rand\_x, aes(x = RATINGS)) + geom\_histogram(aes(y=..density..), colour="black", fill="white")+  
 geom\_density(color="darkblue", fill="lightblue") + geom\_vline(aes(xintercept=mean(RATINGS)),  
 color="blue", linetype="dashed", size=1)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



corr\_matrix <- rcorr(as.matrix(rand\_x))  
corr\_matrix

## GRIP ARM RATINGS SIMS log\_ratings  
## GRIP 1.00 0.21 -0.01 0.16 0.00  
## ARM 0.21 1.00 0.22 0.69 0.23  
## RATINGS -0.01 0.22 1.00 0.17 0.99  
## SIMS 0.16 0.69 0.17 1.00 0.17  
## log\_ratings 0.00 0.23 0.99 0.17 1.00  
##   
## n= 147   
##   
##   
## P  
## GRIP ARM RATINGS SIMS log\_ratings  
## GRIP 0.0095 0.9174 0.0483 0.9701   
## ARM 0.0095 0.0072 0.0000 0.0053   
## RATINGS 0.9174 0.0072 0.0424 0.0000   
## SIMS 0.0483 0.0000 0.0424 0.0354   
## log\_ratings 0.9701 0.0053 0.0000 0.0354