Configuration Manual

MSc Research Project
MSc in Fintech

Ogochukwu Pamela Ezeonugo
Student ID: X18211534

School of Computing
National College of Ireland

Supervisor: Noel Cosgrave

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Ogochukwu Pamela Ezeonugo …….

Student ID: X18211534 ………

Programme: Msc Fintech ……… **Year:** …………………2020………..

Module: Msc Research Project ………

Lecturer: Noel Cosgrave ………

Submission Due Date: ………15-08-2020 ………

Project Title: …Performance of Crowdfunding in Agricultural Campaigns ………

Word Count: ………………………17……………… **Page Count:** ……………………………2091……………………

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ………………………………………

Date: ………………………………………

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	

enalty Applied (if applicable):	

Configuration Manual

Ogochukwu Pamela Ezeonugo
Student ID: x1821534

1 Introduction

This configuration manual provides a detail report and system set up with specific software and hardware required to perform the research under the thesis titled: *The Performance of Crowdfunding in Agriculture an Evidence from Kickstarter.*

2 Dataset

The dataset is retrieved from a WebCrawler[1] that contains multiple excel files with different years from the Kickstarter platform.

3 System Requirements

Hardware Requirements

- System Model: HP Pavilion Laptop 13-an0xxx
- Processor: Processor Intel(R) Core(TM) i5-8265U CPU @ 1.60GHz, 1800 Mhz, 4 Core(s), 8 Logical Processor(s)
- Operating System: Microsoft Windows 10 Home
- Installed Physical Memory (RAM) 8.00 GB

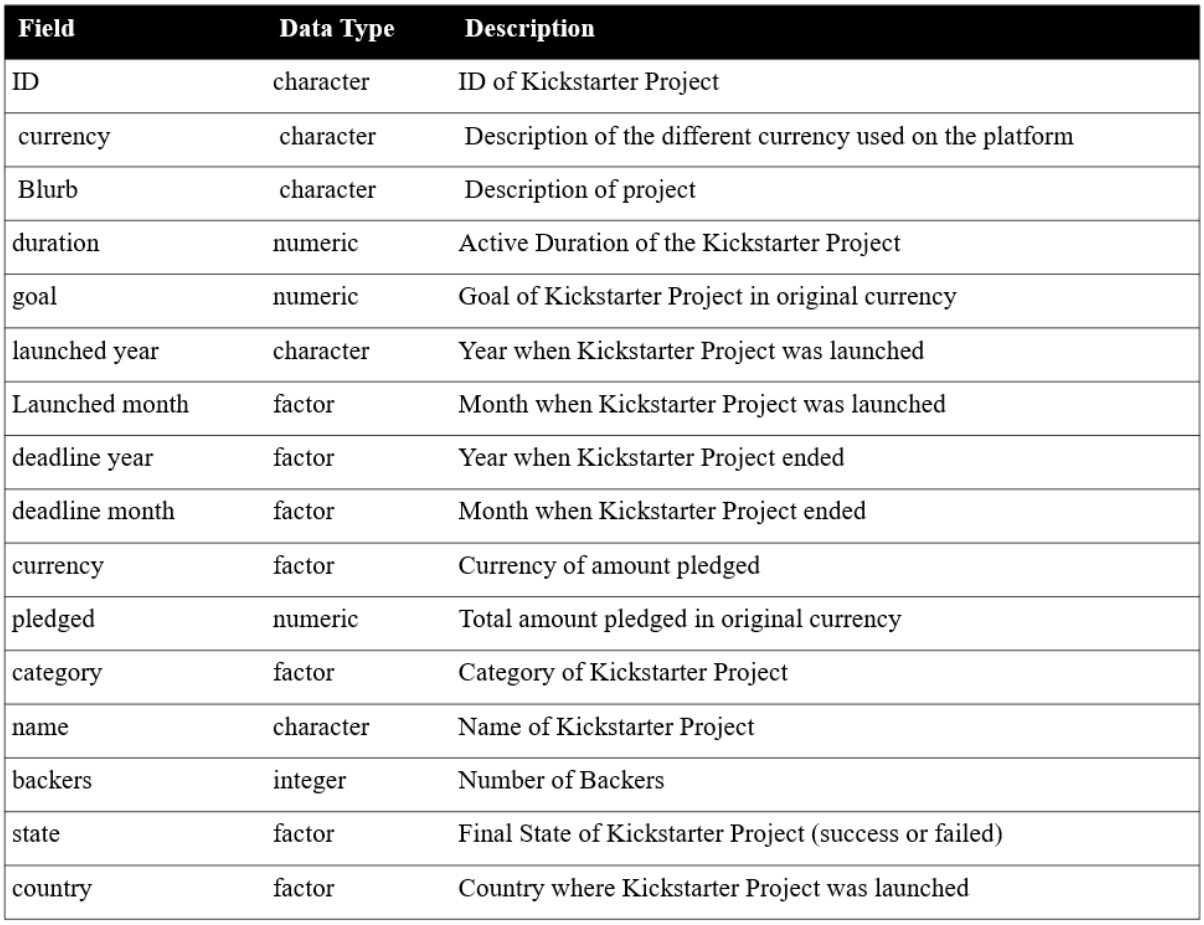
Software Requirements

- R Studio and programming language – R version 4.0.2

[1] https://webrobots.io/kickstarter-datasets/

4 Metadata

The description of different features in the data set.



Field	Data Type	Description
ID	character	ID of Kickstarter Project
currency	character	Description of the different currency used on the platform
Blurb	character	Description of project
duration	numeric	Active Duration of the Kickstarter Project
goal	numeric	Goal of Kickstarter Project in original currency
launched year	character	Year when Kickstarter Project was launched
Launched month	factor	Month when Kickstarter Project was launched
deadline year	factor	Year when Kickstarter Project ended
deadline month	factor	Month when Kickstarter Project ended
currency	factor	Currency of amount pledged
pledged	numeric	Total amount pledged in original currency
category	factor	Category of Kickstarter Project
name	character	Name of Kickstarter Project
backers	integer	Number of Backers
state	factor	Final State of Kickstarter Project (success or failed)
country	factor	Country where Kickstarter Project was launched

5 R Libraries

The following libraries were used in the R studio for data exploration, transformation and modelling.

Functions	Packages	Version
Retrieve excel data	library(readxl)	1.3.1
Verb of data manipulation	library(dplyr)	1.0.0
Use to tidy up messy data	library(tidyr)	1.1.0
Visualizing of data with plots	library(ggplot2)	3.3.2
Parsing of dates and time	library(lubridate)	1.7.8
Data manipulation	library(magrittr)	1.5
To create a formattable object	library(formattable)	0.2.0.1
Tool for dynamic report generation	library(knitr)	1.29
Summarize information of a model	library(broom)	0.5.6
For data bootstrapping	library(boot)	1.3-25
Logistic regression analysis	library(glmnet)	4.0-2
For handling generalized additive models	library(mgcv)	1.8-31

For statistical inference	Library(rcompanion)	2.3.25
Correlation plot	Library(GGally)	2.0.0
Verifying discrete data	library(verification)	1.4.2
For training data and classification regression	library(caret)	6.0-86
Functions as an assistant to install other packages	library(tidyverse)	1.3.0
For creating data frame	library(data. Table)	1.12.8
Create a data table	library(DT)	0.14
Text mining	library(tm)	0.7-7
Data visualization	library(plotly)	4.9.2.1
Convert country names and codes	library(countrycode)	1.2.0
Creation of high Charts	library(highcharter)	0.7.0
Contains	library(caTools)	1.18.0
Calculating statistics in logistic regression	library(InformationValue)	1.2.3
Plotting ROC curve	library(ROCR)	1.0-11
For calculating sequential test	library(car)	3.0-8
Carrying out sentiment analysis	Ilibrary(SentimentAnalysis)	1.3-3
Converting data into different shapes	library(splitstackshape)	1.4.8
Creation of data frames	library(tibble)	3.0.1
Transformation of word	library(tidytext)	0.2.5
Providing access to text data	library(textdata)	0.4.1
Visualizing and analyzing of text	library(wordcloud)	2.6
Transforms data from wide to long	library(reshape2)	1.4.4
To alter text by sentence	library(sentimentr)	2.7.1
Reading file into r	library(readr)	1.3.1
Balancing data	library(DMwR)	0.4.1
Contains multiple function	library(mlbench)	2.1.1
Used for multivariate analysis	library(psych)	1.9.12.31
Object Connection	library(curl)	4.3
Plotting ROC curve	library(PRROC)	1.3.1
For carrying out regression analysis	library(Metrics)	0.1.4
Table manipulation	library(kableExtra)	1.1.0
Parametric testing tools	library(lmtest)	0.9-37

6 Codes

Data Exploration & Transformation Codes

```
####### Importing data Set into R
CF_KS <- read.csv("~/CF_KS.csv")
```

Structure of Data

```
str(CF_KS)
```

Exploring Factors with Multiple Levels
##################### CURRENCY
##
str(CF_KS$currency)

```
CF_KS$currency<-as.character(CF_KS$currency)
CF_KS$currency[CF_KS$currency %in% c("JPY", "HKD", "SGD", "CHF", "NOK",
"DKK", "MXN", "NZD",  "SEK")] <- "Other"
CF_KS$currency <- as.factor(CF_KS$currency)
levels(CF_KS$currency)
```

###########Country#######################

```
str(CF_KS$country)
CF_KS$country<-as.character(CF_KS$country)
CF_KS$country[CF_KS$country %in% c("JP", "LU", "AT", "HK", "SG", "BE", "CH", "IE",
                              "NO", "DK",  "MX", "NZ", "SE", "ES", "IT", "NL", "FR", "DE")] <-
"Other"
CF_KS$country <- as.factor(CF_KS$country)
levels(CF_KS$country)
```

############ Dummy Encoding #######################

```
disable_communication_matrix<-model.matrix(data= CF_KS,state~ disable_communication-
1)
spotlight_matrix<-model.matrix(data= CF_KS,state~spotlight-1)
is_starrable_matrix<-model.matrix(data = CF_KS,state~ is_starrable-1)
currency_matrix<-model.matrix(data=CF_KS,state~currency-1)
country_matrix<-model.matrix(data=CF_KS,state~country-1)
currency_trailing_code_matrix<-model.matrix(data = CF_KS,state~ currency_trailing_code-
1)
staff_pick_matrix<-model.matrix(data=CF_KS,state~staff_pick-1)
```

########### Binding of the created matrix from Dummy encoding ################

```
CF_KS <- cbind(CF_KS[,1:38],currency_matrix,country_matrix,
            currency_trailing_code_matrix,disable_communication_matrix,
            spotlight_matrix,staff_pick_matrix,is_starrable_matrix)
```

########## Transformation ############

The codes here are used to convert the strings that came in form of a time stamp in the data as date. Al

######### Changing Strings to date ##################

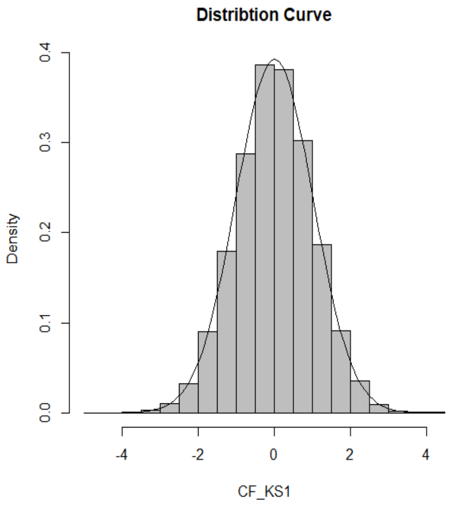
```
date_cols <- c("deadline","state_changed_at","created_at","launched_at")
```

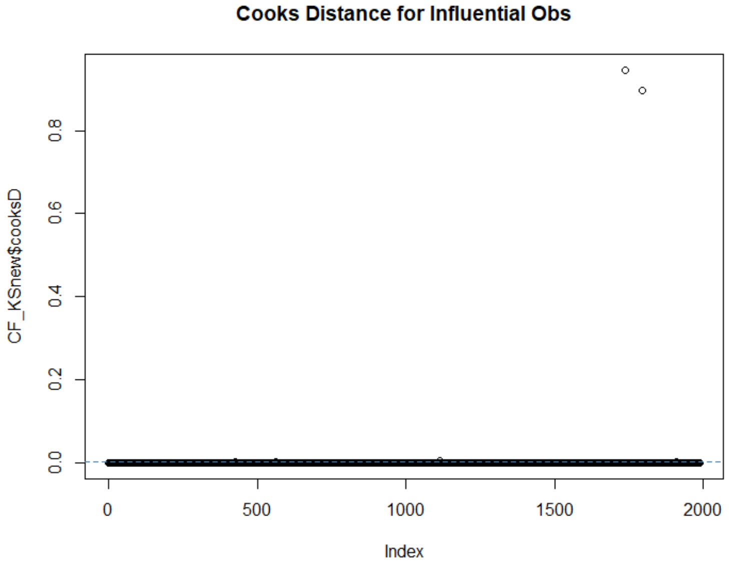
######## Feature Creation #############

########## Creation of Duration ##############

```
CF_KS$duration<-as.numeric(CF_KS$deadline - CF_KS$launched_at)
```

```
###### Date #########
CF_KS <- CF_KS %>%separate(col = "deadline", into = c("deadline_year",
"deadline_month", "deadline_day"), sep = "-") %>%
  separate(col = "launched_at", into = c("launched_year", "launched_month",
"launched_day"), sep = "-")

####### Assigning of  Labels #######
CF_KS$id <- as.character(CF_KS$id)
CF_KS$backers_count<-as.numeric(CF_KS$backers_count)
CF_KS$state_changed_at<- as.Date(CF_KS$state_changed_at)
CF_KS$created_at <- as.Date(CF_KS$created_at)

######Changing Dependent Variable to Factors###################
CF_KS$state<-as.factor(CF_KS$state)
levels(CF_KS$state) <- c("failed", "failed", "successful","successful","failed")
CF_KS$state= factor(CF_KS$state, levels = c('failed', 'successful'), labels = c(0,1))
levels(CF_KS$state)
str(CF_KS)

##########  Data Validation #####################
######## Finding Duplicates####################

which(duplicated(CF_KS$id))
CF_KS<-CF_KS[!duplicated(CF_KS$id), ]
sum(duplicated(CF_KS))

########### Missing Values #############

sapply(CF_KS, function(x) sum(is.na(x)))
sum(is.na(CF_KS))


############ Feature Engineering  ##################
CF_KS$namelength<-sapply(CF_KS$name, function(x) length(unlist(strsplit(as.character(x),
"\\W+"))))

########### Success Rate #########
CF_KS$Success_rate<-CF_KS$pledged/CF_KS$goal

########### Distribution of the data ###################
CF_KS1<-CF_KS
Descriptive
CF_KS1<- rnorm(6525, mean = 0, sd = 1)
mean(CF_KS1)
sd(CF_KS1)
descriptives <- function(CF_KS1) {
  u <- length(unique(CF_KS1))
  missing <- sum(is.na(CF_KS1)) / length(CF_KS1) * 100
  quantiles <- quantile(CF_KS1, na.rm = T)
  av <- mean(CF_KS1, na.rm = T)
  stdev <- sd(CF_KS1, na.rm = T)
  df <- data.frame(u, missing, quantiles[1], quantiles[2],
                   quantiles[3], quantiles[4], quantiles[5], av, stdev)
  names(df) <- c("no. unique", "% missing", "min", "first quartile",
                 "median", "third quartile", "max", "mean", "sd")
  rownames(df) <- NULL
  return(df)
}
df <- descriptives(CF_KS1)

######## histogram##################
hist(CF_KS1)
#distribution curve
hist(CF_KS1,freq=FALSE,col="Gray",xlab="CF_KS1",main="Distribtion Curve")
curve(dnorm(x,mean=mean(CF_KS1),sd=sd(CF_KS1)),add=TRUE,col="black")
```





```
########### ###Staff_Pick ###############
staff pick <- CF_KS %>%
          group_by(staff_pick, state) %>%
          summarise(count = n()) %>%
         mutate(percent_of_total_state = round(count / sum(count), digits = 4)) %>%
          filter(state == '1') %>%
         select(staff_pick, state, percent_of_total_state) %>%
           arrange(desc(percent_of_total_state))

kable(staff_pick, align = rep('c', 3)) %>%
  kable_styling(bootstrap_options = c("striped"), full_width = F)

########## Spotlight ############
spotlight <- CF_KS %>%
  group_by(spotlight, state) %>%
  summarise(count = n()) %>%
  mutate(percent_of_total_state = round(count / sum(count), digits = 4)) %>%
  filter(state == '1') %>%
  select(spotlight, state, percent_of_total_state) %>%
  arrange(desc(percent_of_total_state))

kable(spotlight, align = rep('c', 3)) %>%
  kable_styling(bootstrap_options = c("striped"), full_width = F)
```

VI)Data Visualization

```
################ Chi-square TEST ################
chisq.test(CF_KS$state, CF_KS$duration)
chisq.test(CF_KS$state, CF_KS$country)
chisq.test(CF_KS$state, CF_KS$currency)

########## VISUALIZATION OF  DATA #######################
###### State of the Project #####
ggplot(CF_KS, aes(state, fill = state)) +
  geom_bar() +
  ylab("# of Projects") + xlab("Final State") +
  theme(legend.position = "bottom") +
  ggtitle("State of Projects")

######### Projects By Year #####################################
ggplot(CF_KS, aes(x = launched_year, fill = CF_KS$state)) +
  geom_bar() +
  theme(legend.position = "bottom") +
  ylab("Number of projects") + xlab("Year launched") +
  ggtitle("Agricultural Crowdfunding  projects by Year")
```

```
########Projects launched Over the years######
df1 <- as.data.frame(table(CF_KS$launched_year, CF_KS$launched_month))
names(df1) <- c('Launched_Year','Launched_Month', 'Freq')
df1 <- df1 %>% group_by(Launched_Year) %>% arrange(desc(Freq))

ggplot(df1, aes(x = Launched_Year, y = `Freq`, fill = Launched_Month)) +
  geom_bar(stat = "identity", position = "dodge") +
  ggtitle("Crowdfunding Agricultural Projects launched over time") +
  scale_y_continuous("# of Projects launched", labels = scales::comma) +
  scale_x_discrete("Year")

#########  Correlation  ##########
corMat <- cor(CF_KS[, c(1,4,17,18,30,38,40,64,65,66)])
corMat<-round(corMat,2)
ggcorr(corMat, nbreaks=8, palette='RdGy', label=TRUE, label_size=5, label_color='white')


#######Removal of rows and  irrelevant rows & columns with missing values and URLs
##################

CF_KS<- CF_KS%>% select(-
c(category,pledged,country,converted_pledged_amount,created_at,creator,currency,
                        currency_symbol,currency_trailing_code,current_currency,
                        deadline_year,deadline_day,deadline_month,disable_communication,
                        friends,static_usd_rate,launched_year,
                       launched_month,launched_day,location,name,permissions,photo,profile,slug,
                        is_backing,is_starred,is_starrable,urls,
                        spotlight,staff_pick,state_changed_at,source_url,
                        usd_type,usd_pledged,X,))
str(CF_KS)

####### Re-ordering the data set variables to a meaningful order ########
CF_KS <- CF_KS[, c(6,1:5,7:30)]
##################Normalizing the Dataset###########################
normalize<- function(x) {   return ((x - min(x)) / (max(x) - min(x))) }
CF_KS$goal<-normalize(CF_KS$goal)
CF_KS$backers_count<-normalize(CF_KSmodel$backers_count)
CF_KS$fx_rate<-normalize(CF_KS$fx_rate)
CF_KS$duration<-normalize(CF_KS$duration)
CF_KS$namelength<-normalize(CF_KS$namelength)
CF_KS$Success_rate<-normalize(CF_KS$Success_rate)

######## Creation of new data set to handle Assumptions  and Modelling ##############
CF_KSnew<-CF_KS
CF_KSmodel<-CF_KS
CF_KSsentiment<-CF_KS
Model Assumptions
############## Removal of Columns Blurb and ID #############
CF_KSnew<- CF_KSnew[, -c(3,6)]
```

```
#### Using Cook’s Distance to Check for Influential  Values
CF_KSnew$state<-as.numeric(CF_KSnew$state)

outliers <- glm(state ~ ., data = CF_KSnew)
CF_KSnew$cooksD <- cooks.distance(outliers)
n <- nrow(CF_KSnew)
plot(CF_KSnew$cooksD, main = "Cooks Distance for Influential Obs")
abline(h = 4/n, lty = 2, col = "steelblue")
CF_KSnew[ which( CF_KSnew$cooksD >= 0.5 ) , ]

############# Variance influence Factors  ##################
mydata<-data.frame(CF_KSnew[, -c(5:25)])
mydata<-mydata[,-c(8)]
round(cor(mydata),2)
mymodel<-lm(state~.,mydata)
mymodel
summary(mymodel)
vif(mymodel)

########## DURBIN WATSON TEST #################
durbinWatsonTest(mymodel)

#########NON -CONSTANT VARIANCE SCORE ###############################
ncvTest(mymodel)

##### Descriptive Statistics  #######
stat<- stat.desc(CF_KSnew[,c(2,3,4,27,28,29)])
```

Sentiment Analysis

```
######## Sentiment Analysis ################
str(CF_KSsentiment)

CF_KSsentiment<- CF_KSsentiment[, -c(2,4,5,7,8:30)]
#Arranging in Specific Order
CF_KSsentiment <- CF_KSsentiment[, c(3,1,2)]
levels(CF_KSsentiment$state)

######### Split Blurb #########
CF_KSsentiment<-cSplit(CF_KSsentiment,"blurb", sep =" ", direction ="long")

#Special character Seperations
CF_KSsentiment<-cSplit(CF_KSsentiment,"blurb", sep ="/ ", direction ="long")
CF_KSsentiment

names(CF_KSsentiment)<-c("id","state","word")

#Cleaning Special characters
CF_KSsentiment$word<-gsub("$","", CF_KSsentiment$word)
CF_KSsentiment$word<-gsub(":","", CF_KSsentiment$word,fixed = TRUE)
```

```
#Show data stop words and most common words
tail(names(sort(table(CF_KSsentiment$word))),100)

######## Frequency of Words #########
CF_KSsentimentpairs<- CF_KSsentiment %>%
pairwise_count(word, id, sort = TRUE, upper = FALSE)
CF_KSsentimentpairs

#Convert to a tibble
CF_KSsentimentTibble<-as_data_frame(CF_KSsentiment)
class(CF_KSsentimentTibble)
CF_KSsentimentTibble

###############using the Bing Sentiment #######################
bing<-CF_KSsentiment%>%
  inner_join((get_sentiments("bing")))
bing

#Common Sentiment Words in Bing
bing_word_counts <- CF_KSsentiment %>%
  inner_join(get_sentiments("bing")) %>%
  count(word, sentiment, sort = TRUE) %>%
  ungroup()
```