Configuration Manual

MSc Research Project
MSc in FinTech

Ashwani Teotia
Student ID: x17160715

School of Computing
National College of Ireland

Supervisor: Noel Cosgrave

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Ashwani Teotia
Student ID:	x17160715
Programme:	MSc in FinTech
Year:	2019
Module:	MSc Research Project
Supervisor:	Noel Cosgrave
Submission Due Date:	12/08/2019
Project Title:	Configuration Manual
Word Count:	1000
Page Count:	8

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:	15th September 2019

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:	
Penalty Applied (if applicable):	

Configuration Manual

Ashwani Teotia
x17160715

1 Introduction

This is the configuration manual to assist the user to configure the artifact produced as part of the thesis titled "Prediction of Crowdfunding Project Success Probability using Machine Learning". This manual will include the details regarding software and hardware used to realize the research artifact. This document will also contain the discussion regarding why the specific decisions are made to design and implement to assist the user for an overview. Code snippets are also provided for insights into logical implementation details. The Methodology used to complete this research project is CRISP-DM (Chapman, Clinton, Kerber, Khabaza, Reinartz, Shearer and Wirth; 2000).

2 Data Collection

Dataset [1] for this work is collected from Kaggle [2] website. Yu, Huang, Yang, Liu, Li and Tsai (2018) have used this dataset in research work. This data corresponds to reward-based crowd markets platform- Kickstarter [3]. Dataset has 378,661 observations and 15 features. Further to scope the project feasibility, only the projects related to the United Kingdom were selected. Scoped dataset has 29,453 observations and 6 features. Data is downloaded from Kaggle as "CSV" format and used as it is into R-Studio to prepare and process as required for modeling.

2.1 Dataset Metadata

Collected data have following metadata, extracted using R-language with RStudio.

[1]Dataset used: `https://www.kaggle.com/kemical/kickstarter-projects/version/7`
[2]Kaggle website: `http://www.kaggle.com`
[3]Kickstarter website: `http://www.kickstarter.com`

```
'data.frame':   378661 obs. of  15 variables:
 $ ID              : int  1000002330 1000003930 1000004038 1000007540 1000011046 1000014025 1000023410 1000030581 1000034518
 100004195 ...
 $ name            : Factor w/ 375755 levels "\177Not Twins - New EP! \"The View from Down Here\"",..: 332526 135661 364998
 344791 77315 206108 293449 69327 284125 290705 ...
 $ category        : Factor w/ 159 levels "3D Printing",..: 109 94 94 91 56 124 59 42 114 40 ...
 $ main_category   : Factor w/ 15 levels "Art","Comics",..: 13 7 7 11 7 8 8 8 5 7 ...
 $ currency        : Factor w/ 14 levels "AUD","CAD","CHF",..: 6 14 14 14 14 14 14 14 14 14 ...
 $ deadline        : Factor w/ 3164 levels "2009-05-03","2009-05-16",..: 2288 3042 1333 1017 2247 2463 1996 2448 1790 1863
 ...
 $ goal            : num  1000 30000 45000 5000 19500 50000 1000 25000 125000 65000 ...
 $ launched        : Factor w/ 378089 levels "1970-01-01 01:00:00",..: 243292 361975 80409 46557 235943 278600 187500 274014
 139367 153766 ...
 $ pledged         : num  0 2421 220 1 1283 ...
 $ state           : Factor w/ 6 levels "canceled","failed",..: 2 2 2 2 1 4 4 2 1 1 ...
 $ backers         : int  0 15 3 1 14 224 16 40 58 43 ...
 $ country         : Factor w/ 23 levels "AT","AU","BE",..: 10 23 23 23 23 23 23 23 23 23 ...
 $ usd.pledged     : num  0 100 220 1 1283 ...
 $ usd_pledged_real: num  0 2421 220 1 1283 ...
 $ usd_goal_real   : num  1534 30000 45000 5000 19500 ...
```

Figure 1: Dataset Metadata

3 System Setup

3.1 Hardware

The research work is implemented and deployed using laptop or a desktop machine with the following description:

- Laptop Machine:
  Processor: Intel(R) Core (TM) i5 2430M CPU @ 2.40 GHz Dual core
  8GB RAM
  500GB HDD
  GPU: Intel HD Graphics 3000

- Desktop Machine: Processor: Intel(R) Xenon(R) CPU E5-1620 v4 @ 3.50 GHz Quad core
  16GB RAM
  512GB SSD
  GPU: NVIDIA Quadro M2000 12GB

  The research work is started on a laptop, though with the need to tune models the desktop machine was used with a better configuration. As the data analysis grew with the tuning of models and the project data scope defined the artifacts could develop on either of the hardware.

3.2 Software

The research work is implemented and deployed either on a Windows 7 professional or windows 10 professional version. GitHub [4] is used as a repository for the artifacts produced in this work to maintain versioning and the reviews feedback.

Zotero [5] is used as a reference management software to maintain the references and citations used in this research work.

64-bit RStudio version 1.2.1335 is used with R-version 3.5.3 in this work to complete the data analysis to produce research artifact. An issue is observed regarding RStudio version, 64-bit RStudio has a better model run performance with respect to RAM usage.

[4] GitHub website: `https://github.com/`
[5] zotero website: `https://www.zotero.org/`

Noticed 32-bit RStudio version was using less RAM, once RStudio upgraded to 64-bit the performance of the model was better.

Microsoft Excel 2017 with R language is used to get insight from the crowd markets data.

4 Software/ Libraries

RStudio with R language is the main data analysis software used in this research work. There are six R-code files used in this work, more details in section 6. Following is the version information of R language used:

```
platform       x86_64-w64-mingw32
arch           x86_64
os             mingw32
system         x86_64, mingw32
status
major          3
minor          5.3
year           2019
month          03
day            11
svn rev        76217
language       R
version.string R version 3.5.3 (2019-03-11)
nickname       Great Truth
```

Figure 2: R Version

4.1 R library and Windows Software Specification

R:

knitr – This package is used to draw data in tabular form using kable function.
plotly – This package is used to draw plots using ggplot2.
lubridate – This package is used to convert date in text format to date format.
Caret - This package is used as a wrapper to use models used in this research work. This library is used for k-fold cross validation as well as to evaluate models performances.

Windows:

RStudio 1.2.1335
Microsoft Excel and Word 2017
TeXstudio desktop LaTeX software is used as creating the final research paper.

5 High Level Design

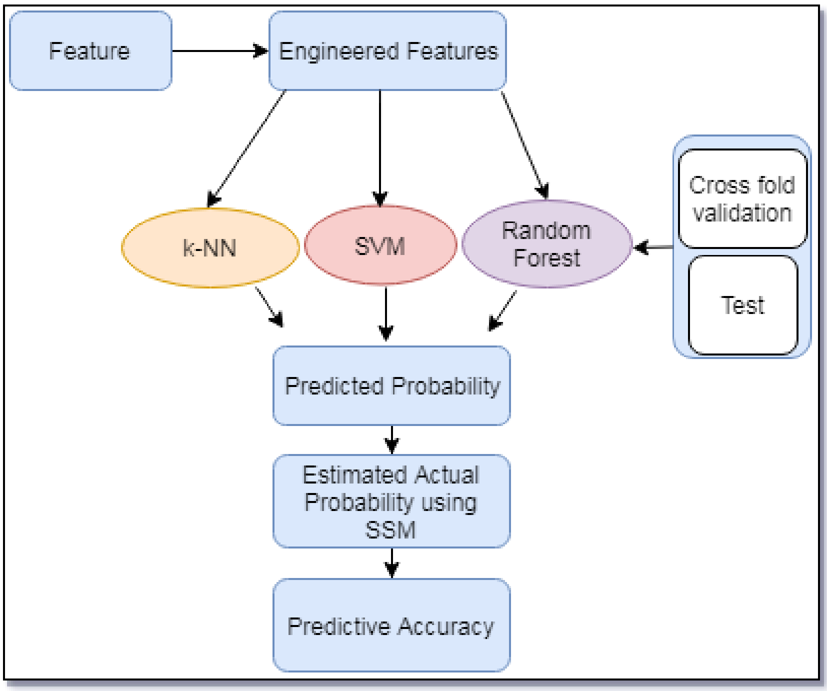

Figure 3: High Level Design

k-fold cross validation where k is kept as 10 is used to validate models result in this research. This technique is used to evaluate performance of the models and at the same time to avoid over and under-fitting of the models.

Seeding is used at the time of model execution, sampling and to one-hot encode variable. This is to maintain reproducibility of this research work.

6 R Code Files

R programming language is mainly used for data analysis and specifically have packages related to various statistical and machine learning tasks. Six R code files are developed in this research work as ICT artifact. R code files implemented in this research work are described as follows:

- Explore.R: This file contains the code related to the exploration of the dataset. Code in this R-file prepares data quality reports (DQR) (Kelleher, Namee and D'Arcy; 2015) , does data cleanup and explores data using correlation and graphs. This file realizes data understanding and data preparation phases of CRISP-DM. Following outputs are prepared in this R-code file;

  DQR before Data processing:

  DQR for numerical features:

Feature	Instances	Missing	Cardinality	Min	FirstQuantile	Median	ThirdQuantile	Max	Mean	Stdev
goal	33672	0	1505	1.00	1000.000	3000.000	10000.000	100000000	28149.14796	842940.5539
pledged	33672	0	9437	0.00	20.000	342.000	2145.080	3771475	4795.73025	38106.1574
backers	33672	0	1170	0.00	2.000	11.000	49.000	73206	84.92477	618.7751
usd.pledged	33672	0	23479	0.00	12.190	268.535	2092.582	5342789	5458.09062	50708.2977
usd_pledged_real	33672	0	24460	0.00	29.040	506.195	3203.395	5494493	7026.23382	56073.8233
usd_goal_real	33672	0	14868	1.22	1402.078	4745.860	15383.200	166361391	42774.12439	1337008.1220

Figure 4: DQR:Numeric Feature

DQR for categorical features:

	% Missing	% Unique	Mode1	Mode2	Mode1Freq	Mode2Freq	Mode1Percent	Mode2Percent
main_category	0.0000000	15	Film & Video	Games	5782	4012	17.1715372	11.9149442
category	0.0000000	159	Product Design	Video Games	1842	1533	5.4704205	4.5527441
status	0.0000000	5	failed	successful	17387	12067	51.6363744	35.8368971
location	0.0000000	1	GB	GB	33672	0	100.0000000	0.0000000
prjname	0.0029698	33552	Inspiration Magazine	That's What She Said Magazine - Issue #14	3	3	0.0089095	0.0089095

Figure 5: DQR:Categorical Feature

DQR after Data processing:

DQR for numerical features:

Feature	Instances	Missing	Cardinality	Min	FirstQuantile	Median	ThirdQuantile	Max	Mean	Stdev
goal	29453	0	13578	1.22	1307.53	4434.26	14382.85	166361391	3.929636e+04	1.353678e+06
backers	29453	0	1145	0.00	2.00	14.00	55.00	73206	9.276359e+01	6.588320e+02
duration	29453	0	62	0.00	29.00	29.00	30.00	89	3.075626e+01	1.107673e+01
name_length	29453	0	65	1.00	19.00	32.00	48.00	66	3.326476e+01	1.613746e+01
status	29453	0	2	0.00	0.00	0.00	1.00	1	4.097036e-01	4.917873e-01

Figure 6: DQR:Numeric Feature

DQR for categorical features:

	% Missing	% Unique	Mode1	Mode2	Mode1Freq	Mode2Freq	Mode1Percent	Mode2Percent
category	0	15	Film & Video	Games	5216	3245	17.70957	11.01755
location	0	1	GB	GB	29453	0	100.00000	0.00000

Figure 7: DQR:Categorical Feature

- ABT.R: This file contains code related to creating the analysis base table for modelling. Following analysis base table is produced in this R-code file;
  ABT for metadata feature:

```
'data.frame':   29453 obs. of  6 variables:
 $ goal       : num  1534 6470 17490 143 5181 ...
 $ backers    : int  0 761 0 27 1 0 1 6 9 1 ...
 $ duration   : num  58 27 29 29 29 29 29 42 52 29 ...
 $ name_length: int  31 57 20 52 17 28 58 30 11 15 ...
 $ status     : num  0 1 0 1 0 0 0 0 0 0 ...
 $ category   : Factor w/ 15 levels "Art","Comics",..: 13 9 3 2 8 14 3 5 6 7 ...
```

Figure 8: Analysis Base Table Metadata

- k-NN.R: This file contains code related to tuning the k-NN model. Following is the code snippet used to perform model tuning;

```
set.seed(457)
training.Control <- trainControl(method = "cv", number = 10)
model_Fit <- train(status ~goal + backers + duration +prj_name_length,
    data = training, method = "knn",
trControl=training.Control,
preProcess = c("center", "scale"),
tuneLength=30,
na.action=na.exclude)
```

- RF.R: This file contains code related to tuning the random forest model. Following is the code snippet used to perform model tuning;

```
set.seed(4541)

# Random Search
control <- trainControl(method="cv", number=10, search="random")
mtry <- sqrt(ncol(Training))
rf_random <- train(status~., data=Training, method="rf",
    metric="metric", tuneLength=15, trControl=control)
print(rf_random)
plot(rf_random)


set.seed(4541)
control <- trainControl(method="cv", number=10, search="grid")

tunegrid <- expand.grid(.mtry=c(12:19))
rf_gridsearch <- train(status~., data=Training, method="rf",
    metric="metric", tuneGrid=tunegrid, trControl=control)
```

```
print(rf_gridsearch)
plot(rf_gridsearch)
```

- SVM.R: This file contains code related to tuning the support vector machine model. Following is the code snippet used to perform model tuning;

```
set.seed(4581)
trctrl <- trainControl(method = "cv", number = 10,classProbs=TRUE)


# svm_Linear <- train(status ~., data = Training, method = "svmLinear",
#                    trControl=trctrl,
#                    preProcess = c("center", "scale"),
#                    tuneLength = 10)



# grid <- expand.grid(C = c(0,0.01, 0.05, 0.1, 0.25, 0.5, 0.75, 1, 1.25,
    1.5, 1.75, 2,5))
# svm_Linear_Grid <- train(status ~., data = Training, method =
    "svmLinear",
#                          trControl=trctrl,
#                          preProcess = c("center", "scale"),
#                          tuneGrid = grid,
#                          tuneLength = 10)


svm_Radial <- train(status ~., data = Training, method = "svmRadial",
trControl=trctrl,
preProcess = c("center", "scale"),
tuneLength = 10)



grid_radial <- expand.grid(sigma = c(0,0.01, 0.02, 0.025, 0.03, 0.04,
0.05, 0.06, 0.07,0.08, 0.09, 0.1, 0.25, 0.5, 0.75,0.9),
C = c(120, 123,125,128,130,132,135))

svm_Radial_Grid <- train(status ~., data = Training, method =
    "svmRadial",
trControl=trctrl,
preProcess = c("center", "scale"),
tuneGrid = grid_radial,
tuneLength = 10)
```

- Model.R: This file contains code to include all the models with hyperparameter tuned. This file also contains implementation of sorting smoothing method. Code related to performance matrices of the model accuracy is also included in this file. Further this file contains linear regression code to calculate predictive accuracy of the probability, generated as the output from the models.

```
probability.failed <- prediction.probability[,1]
probability.success <- prediction.probability[,2]

d.f <- data.frame(probability.success, Y=predictedBool)
d.f <- d.f[order(probability.success),]

names(d.f)[1] <- "V1predprob"
names(d.f)[2] <- "Yclass"

smoothing.valid.range <- (1: dim(d.f)[1])

for (probability.single in 1:dim(d.f)[1]) {
smoothing.required.range <- (probability.single-50):
    (probability.single+50)
smoothing.selected.range <- smoothing.valid.range[(smoothing.valid.range
    %in% smoothing.required.range)]
d.f[probability.single, "V2"] <-
    sum(d.f$Yclass[smoothing.selected.range])/ (2*50 +1)
}

plot( d.f[,"V1predprob"], d.f[,"V2"], main="k-NN",
xlab="Predicted probability", ylab="Estimated Actual probability", pch=1)

abline(lm(d.f[,"V2"]~d.f[,"V1predprob"]), col="blue",lty=1) # regression
    line (y~x)
lines(lowess(d.f[,"V1predprob"],d.f[,"V2"]), col="red", lty=1) # lowess
    line (x,y)

mod <- lm(d.f[,"V2"]~d.f[,"V1predprob"])
ex.cs1 <- expression(Regression, Lowess) # 2 ways
#utils::str(legend(.01, .95, ex.cs1, lty = 1:1, plot = FALSE)) # adj y !
legend(.01, .95, ex.cs1, lty = c(1,1), col=c("blue","red"))
summary(mod)
```

References

Chapman, P., Clinton, J. M., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. R. H. and Wirth, R. (2000). Crisp-dm 1.0: Step-by-step data mining guide.

Kelleher, J. D., Namee, B. M. and D'Arcy, A. (2015). *Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies*, The MIT Press.

Yu, P., Huang, F., Yang, C., Liu, Y., Li, Z. and Tsai, C. (2018). Prediction of crowdfunding project success with deep learning, *2018 IEEE 15th International Conference on e-Business Engineering (ICEBE)*, pp. 1–8.