

# Configuration Manual

Impact and Implications Analysis of  
Fintech and International Trade & Supply Chain Finance

MSc Research Project  
Programme Name: MSC FINTECH

Narasimhan Kannan  
Student ID: X18100040

School of Computing  
National College of Ireland

Supervisor: Mr. Victor Del Rosal

**National College of Ireland**  
**MSc Project Submission Sheet**  
**School of Computing**



**Student Name:** Narasimhan Kannan  
**Student ID:** X18100040  
**Programme:** MSC FINTECH **Year:** 2019  
**Module:** Research Project  
**Lecturer:** Mr. Victor Del Rosal  
**Submission Due Date:** 12th August 2019  
**Project Title:** Fintech and International Trade & Supply Chain Finance  
**Word Count:** 1508 **Page Count:** 7

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)	<input type="checkbox"/>
<b>Attach a Moodle submission receipt of the online project submission,</b> to each project (including multiple copies).	<input type="checkbox"/>
<b>You must ensure that you retain a HARD COPY of the project,</b> both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

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

<b>Office Use Only</b>	
Signature:	
Date:	
Penalty Applied (if applicable):	

# Configuration Manual

## Fintech and International Trade & Supply Chain Finance

Narasimhan Kannan  
Student ID: X18100040

### 1 General Information

Aim of this manual is to assist the other user to replicate the model developed in this study. The user's manual includes the modus operandi and algorithm processed in "R" studio. For this study two datasets are used, one from Kaggle trade statistics and other created virtual data sets from reliable sources. This manual consists of data background, application of CRISP-DM approach, model building, predictions, graph plotting.

### 2 Getting Started

#### 2.1 Application of Time series for data set 1 -for Traditional banking trade flow

#### 2.2 Data sets and background.

This data set is from Kaggle database and includes the imports and exports volumes of commodities of various countries in last 2 decades. The attributes consist of year, country, commodity, export, and import trade flow in USD, etc (10 columns and > 0.82 million rows). The rationale for choosing this data set is to arrive a benchmark for the trade flows and presume that these flows were transacted through banking channels where 80% to 90% utilizing trade finance facility (Xin Fin Organization, 2018) by buyers and sellers.

#### 2.3 Install Library:

Before the data sets imported into "R" following package to be loaded.

```
# Loading the libraries required for time series
install.packages('dplyr')
library(timeSeries)
library(timeDate)
library(tseries)
library(forecast)
library(dplyr)
library(ggplot2)
install.packages("psych")
library(psych)
```

#### 2.4 Ratio Split and Scientific notation (E)

In order to Train and Test the data sets, ratios of 80:20 is assigned and used "options(scipen=999)" "for scientific notation.

## 2.5 Read the data into R

As the data is large, data is split into two files and merged using “rbind” function in “R”

```
tradedata1 <- read.csv(file="D:/Research/Data/Data_part1.csv",header=TRUE, sep=",")
tradedata2 <- read.csv(file="D:/Research/Data/Data_part2.csv",header=TRUE,sep=",")
tradedata <- rbind(tradedata2,tradedata1)
```

## 2.6 Exploratory Data Analytics

### 2.6.1 To check Missing data and Header:

```
which(is.na(tradedata$country_or_area))
which(is.na(tradedata$year))
which(is.na(tradedata$trade_usd))
describe(na.omit(tradedata))
head(tradedata)
```

### 2.6.2 Identify the number of rows and columns

```
nrow(tradedata1) + nrow(tradedata2)
```

It is noted that this data set consists of 10 columns and 821707 observations and null missing values for key variables such as country, Year USD and Trade flow.

### 2.6.3 Descriptive statistics

```
summary(tradedata$trade_usd)
str(tradedata)
names(tradedata)
```

## 2.7 Identify of Top 10 countries including import and export in Trade - USD Aggregated.

Since the data includes the import and export in USD, values are aggregated based on the countries to find top 10 countries. The result is in USD billion (descending order) and displaced using Bar Chart.

- **# top 10 countries including import and export in Trade USD Aggregated.**

# Code

```
tradedata_top<-tradedata%>%group_by(country_or_area)
%>%summarise(USD=round(sum(trade_usd)/1000000000)) %>%arrange(USD)%>%tail(10)
```

- **# to Display Top 10 countries**

```
tradedata_top
```

- **# To display in GG plot for 10 countries**

```
p <- ggplot(df, aes(x= reorder(country_or_area,-USD), y=USD, fill=country_or_area)) +
  geom_bar(stat="identity")+theme(aspect.ratio = 1)+
  ggtitle("Top 10 countries in terms of trade flow in USD")
plot(p)
```

## 2.8 Year wise aggregation of trade flow for Top 5 countries

- #For Canada:

```
tradedata_sum<-tradedata%>%group_by(country_or_area,year)%>%summarise
(USD=round(sum(trade_usd)/1))%>%arrange(USD)%>%filter(country_or_area == 'Canada')
tradedata_canada<-data.frame(year=tradedata_sum$year,usd
=round(tradedata_sum$USD/1000000000))
```

similarly, replace the Canada to other 4 countries namely EU-28, China, Germany, and France

- #To check data frame:

```
class(tradedata_eu28)
"data.frame"
```

## 2.9 Converting to Timeseries for each of top 5 countries

As data identified on above steps is a data frame format and in order to pass the data into the timeseries model, the data is converted using ts function from start of year 1999. However, for EU, the start date is from 2000.

```
tradedata_canada.ts <- ts(tradedata_canada$usd,start=1989,end=2016)
tradedata_france.ts <- ts(tradedata_france$usd,start=1994,end=2016)
tradedata_germany.ts <- ts(tradedata_germany$usd,start=1991,end=2016)
tradedata_china.ts <- ts(tradedata_china$usd,start=1992,end=2016)
tradedata_eu28.ts <- ts(tradedata_eu28$usd,start=2000,end=2015)
```

**# to check whether Timeseries converted**

```
class(tradedata_canada.ts)
```

```
"ts"
```

## 2.10 Plot the Time Series

In order to identify the original trends for each country, the data are plotted using plot.ts.

```
plot.ts(tradedata_canada.ts, main="Canada - Time Series Original")
plot.ts(tradedata_france.ts, main="France - Time Series Original")
plot.ts(tradedata_germany.ts, main="Germany - Time Series Original")
plot.ts(tradedata_china.ts, main="China - Time Series Original")
plot.ts(tradedata_eu28.ts, main="EU-28 Time Series Original")
```

## 3 Model Building

To build the model, this data is trained and tested as defined by ratio (80:20) and performed for top 5 countries such as Canada, France, Germany, China, and EU-28 and then applied **Auto Arima model**

- # Time series Modelling using Train and Test

```
#Canada
samplesize = ratio * nrow(tradedata_canada)
trainrows <- floor(samplesize)
testrows <- nrow(tradedata_canada) - trainrows
```

```
train_can <- trainrows
test_can <- testrows
```

- # train and test split based on ratio.

```
train.ts.d <- ts(tradedata_canada.ts[1:trainrows],1988+1,end=1988+trainrows)
test.ts.d<-
ts(tradedata_canada.ts[trainrows+1:(trainrows+testrows)],start=1988+trainrows+1,end=1988
+trainrows+testrows)
```

- #Auto Fitting with seasonal

```
auto.fit1 <- auto.arima(train.ts.d,seasonal=T)
summary(auto.fit1)
auto.fcast1 <- forecast(auto.fit1,h=testrows)
plot(auto.fcast1,main="Canada")
auto.fcast1
```

- # root mean square error

```
pred_canada = auto.fcast1$mean
RMSE_canada = sqrt(sum(test.ts.d-pred_canada)^2/testrows)
test.ts.d
pred_canada
RMSE_canada
```

similarly, replace the Canada to other 4 countries namely EU-28, China, Germany, and France and list the Performance Matrix

## 4 Prediction for next 10 years trade flow using Auto ARIMA Model

Based on the past trends and using the auto ARIMA model project the future sales for next 10 years of trad flow.

```
# Prediction for next 10 years
#Auto Fitting with seasonal
# Canada
auto.fit1 <- auto.arima(tradedata_canada.ts,seasonal=T)
summary(auto.fit1)
auto.fcast1 <- forecast(auto.fit1,h=10)
plot(auto.fcast1,main="Canada - Next 10 years forecast")
auto.fcast1
```

Similarly, replace the Canada to other 4 countries namely EU-28, China, Germany, and France

---

## Data Set 2

### **5 Fitech data set 1 -To identify the interest rate to decide profitability (World Bank Data)**

#### **5.1 Read the values**

```
filedata <-read.csv(file="D:/Research/Data/ARIMA .csv",sep="," ,header=TRUE)
```

#### **5.2 To decide plot per frame**

```
par(mfrow=c(1,1))
```

#### **5.3 To assign value to create line**

```
x <- Fitechdata$Year  
y0 <- c(11.0, 10.0, 9.0, 8.5, 8.0, 7.0, 7.0, 6.0, 6.0, 6.0)  
y1 <- Fitechdata$FinTechRate  
y2 <- Fitechdata$GlobalAverage  
y3 <- Fitechdata$MoneyTranOper
```

#### **5.4 To Create a first and Subsequent line**

```
plot(x, y0, type = "l", frame = FALSE,  
     col = "white", xlab = "Year", ylab = "Rate",main="Cost as per % of total Remittance ")
```

#### **# Add a first line**

```
lines(x, y1, pch = 20, col = "red", type = "b", lty = 2)
```

#### **# Add a second line**

```
lines(x, y2, pch = 20, col = "blue", type = "b", lty = 2)
```

#### **# Add a third line**

```
lines(x, y3, pch = 20, col = "green", type = "b", lty = 2)
```

#### **5.5 # Add a legend to the plot**

```
legend("topright", legend=c("FinTech Rate", "Global Average","MoneyTranOper"),  
      col=c("red", "blue","green"), lty = 1:2, cex=0.8)
```

## **6 Data Set2 – For Checking Liquidity**

### **6.1 Read the values**

```
filedata <-read.csv (file="D:/Research/Data/Data set 2 Fitech.csv",sep="," ,header=TRUE)
```

### **6.2 For Global Online Disbursement**

#### **Create a first line**

```
plot(x, y0, type = "l", frame = FALSE,  
     col = "white", xlab = "Year", ylab = "Disbursement",main="Global online Disbursement USD  
(Million)")
```

```

# Add a first line
lines(x, y1, pch = 20, col = "red", type = "b", lty = 2)

# Add a second line
lines(x, y2, pch = 20, col = "blue", type = "b", lty = 2)

# Add a second line
lines(x, y3, pch = 20, col = "green", type = "b", lty = 2)

# Add a second line
lines(x, y4, pch = 20, col = "brown", type = "b", lty = 2)

# Add a legend to the plot
legend("topleft", legend=c("EU-28", "France","Germany","Canada"),
      col=c("red", "blue","green","brown"), lty = 1:2, cex=0.8)

```

### 6.3 To plot china separately - due to wide range in values where China is in Billion and other countries in Million

```

# to decide plot per frame
par(mfrow=c(1,1))

x <- Fintechdata1$Year
y0 <- c(0, 10000, 20000, 24000, 30000, 31000, 28000, 500, 550, 520000)
y5 <- Fintechdata1$China_Disb

# Create a first line
plot(x, y0, type = "l", frame = FALSE,
     col = "white", xlab = "Year", ylab = "Disbursement",main="Global online Disbursement USD (Million)")

lines(x, y5, pch = 20, col = "black", type = "b", lty = 2)

# Add a legend to the plot
legend("topleft", legend=c("China"),
     col=c("black"), lty = 1:2, cex=0.8)

```

### 6.4 For Invoice Trading Disbursement

```

# to decide plot per frame
par(mfrow=c(1,1))

# To assign value to create line
x <- Fintechdata1$Year
y0 <- c(0, 100, 200, 240, 300, 400, 450, 500, 550, 780)
y1 <- Fintechdata1$EU28_InvTrade
y2 <- Fintechdata1$France_InvTrade
y3 <- Fintechdata1$Germany_InvTrade
y4 <- Fintechdata1$Canada_InvTrade
# Create a first line
plot(x, y0, type = "l", frame = FALSE,

```

```
col = "white", xlab = "Year", ylab = "Disbursement",main="Invoice Trading Disbursement USD (Million)")
```

```
# Add a first line
```

```
lines(x, y1, pch = 20, col = "red", type = "b", lty = 2)
```

```
# Add a second line
```

```
lines(x, y2, pch = 20, col = "blue", type = "b", lty = 2)
```

```
# Add a second line
```

```
lines(x, y3, pch = 20, col = "green", type = "b", lty = 2)
```

```
# Add a second line
```

```
lines(x, y4, pch = 20, col = "brown", type = "b", lty = 2)
```

```
# Add a legend to the plot
```

```
legend("topleft", legend=c("EU-28", "France", "Germany", "Canada"),  
col=c("red", "blue", "green", "brown"), lty = 1:2, cex=0.8)
```

## **6.5 To plot china- due to wide range in values where China is in Billion and other countries in Million**

```
# to decide plot per frame
```

```
par(mfrow=c(1,1))
```

```
x <- Fintechdata1$Year
```

```
y0 <- c(0, 1000, 2000, 2400, 3000, 3100, 2800, 500, 550, 5000)
```

```
y5 <- Fintechdata1$China_InvTrade
```

```
# Create a first line
```

```
plot(x, y0, type = "l", frame = FALSE,
```

```
col = "white", xlab = "Year", ylab = "Disbursement",main="Invoice Trading Disbursement USD (Million)")
```

```
lines(x, y5, pch = 20, col = "black", type = "b", lty = 2)
```

```
# Add a legend to the plot
```

```
legend("topleft", legend=c("China"),  
col=c("black"), lty = 1:2, cex=0.8)
```

---

### **References**

Xin Fin Organization, 2018. Blockchain technology for global trade and finance. Xin Fin Organization