EDA of Foreign Trade Statistics - Vehicle and Trailer Tracking

Contents

EDA of Foreign Trade Statistics

These data sources contains Export, Import and Empty Entry data of Turkish and Foreign Vehicles.

There are six data sources from the TCMB site. One of the links is given below.

Dataset Link

Preprocess of Foreign Trade Statistics - Vehicle and Trailer Tracking

Import Data and Necessary Libraries

Our first step was downloading the excel data from the source and importing these files to RStudio.

Here is the code that we import and format the columns into proper data type.(There were character formatted columns which needed to be formatted into numeric.)

```
library(readxl) ## for importing excel files
library(tidyverse) ## for manipulating df's
library(countrycode) ## for conversion of country codes
library(dplyr) ## for data manipulation
library(kableExtra) # Pretty print DataFrame
library(zoo) # Used for converting year month character to date data type
library(wordcloud2)


df_tbg_entry <- read_excel("ProjectData/EVDS_TBG_ENTRY.xlsx")
df_tbg_entry <- df_tbg_entry %>% mutate_at(-c(1),funs(type.convert(as.numeric(.))))
df_tih_export <- read_excel("ProjectData/EVDS_TIH_EXPORT.xlsx")
df_tih_export <- df_tih_export %>% mutate_at(-c(1),funs(type.convert(as.numeric(.))))
df_tit_import <- read_excel("ProjectData/EVDS_TIT_IMPORT.xlsx")
df_tit_import <- df_tit_import %>% mutate_at(-c(1),funs(type.convert(as.numeric(.))))
df_ybg_entry<- read_excel("ProjectData/EVDS_YBG_ENTRY.xlsx")
df_ybg_entry <- df_ybg_entry %>% mutate_at(-c(1),funs(type.convert(as.numeric(.))))
df_yih_export<- read_excel("ProjectData/EVDS_YIH_EXPORT.xlsx")
df_yih_export <- df_yih_export %>% mutate_at(-c(1),funs(type.convert(as.numeric(.))))
```

```
df_yit_import<- read_excel("ProjectData/EVDS_YIT_IMPORT.xlsx")
df_yit_import <- df_yit_import %>% mutate_at(-c(1),funs(type.convert(as.numeric(.))))
```

Changing the Names of Columns and Ordering the Columns

There are some problems with these data frames, those we had to get over:

1. There were 169 columns of each data frames. We needed to split them by countries.
2. Column names were not defined clearly, so we needed to re-define the column names. Here is one of the raw data frame column names.

```
print(head(colnames(df_tbg_entry)))
```

```
## [1] "Tarih"                         "TP UNDNAKLIYE TBG AF ADET"
## [3] "TP UNDNAKLIYE TBG AF ADET-1" "TP UNDNAKLIYE TBG AF ADET-2"
## [5] "TP UNDNAKLIYE TBG AF ADET-3" "TP UNDNAKLIYE TBG AF ADET-4"
```

3. There are country codes, which is complex to read. We needed to convert country codes into country names.
4. And after we split those dataframes by countries, those were later merged into a single dataframe.

Here is the code we create a list which includes the column names those we wanted. The sequence related with raw dataframes.

```
colnames_fixed = c("Level", "PercentageChange","Difference","YearlyPercentageChange","YearlyDifference"
```

Here is the two functions work nested each other. "get_country_codes" function gets the converts the country code into country name. Split and combine uses "get_country_codes" function, and splits the main dataframe by countries as new dataframe merge all dataframes into one by their properties.(Is it foreign or Turkish vehicles or Import-Export etc.)

```
get_country_codes <- function(df){

  country_codes <- list()

  for(val in names(df))
  {
    country_code <- strsplit(val,split=" ")[[1]][4]

    if(!is.na(country_code) & !country_code %in% country_codes){

      country_codes <- append(country_codes,country_code)
    }

  }

  return(country_codes)
}
```

```r
split_and_combine <- function(df,vehicletype,exportimport){

  country_codes <- get_country_codes(df)

  datasets <- list()

  for(code in country_codes){

    df_corrected <- df %>% select(contains(paste(" ",code," A",sep="")))

    colnames(df_corrected) <- colnames_fixed

    df_corrected$Date <- df$Tarih

    df_corrected$ExportImportCountry <- code

    df_corrected$VehicleType <- vehicletype

    df_corrected$ExportImport <- exportimport

    df_corrected$ExportImportCountry <- countrycode(df_corrected$ExportImportCountry,origin = 'iso2c', 

    df_corrected$ExportImportRegion <- countrycode(sourcevar = df_corrected$ExportImportCountry,
                           origin = "country.name",
                           destination = "continent",custom_match = c('Kosovo'='Europe'))

    datasets <- append(datasets,list(df_corrected))

  }

  return(bind_rows(datasets))

}
```

In this last code. We use split and combine function for all of the dataframes and we merged all of data frames those we created with "split_and_combine" function via using bind_rows function.

```r
df_tih_export_cleaned <- split_and_combine(df_tih_export,'TIH','EXPORT')
df_yih_export_cleaned <- split_and_combine(df_yih_export,'YIH','EXPORT')

df_tit_import_cleaned <- split_and_combine(df_tit_import,'TIT','IMPORT')
df_yit_import_cleaned <- split_and_combine(df_yit_import,'YIT','IMPORT')

df_tbg_entry_cleaned <- split_and_combine(df_tbg_entry,'TBG','EMPTY ENTRY')
df_ybg_entry_cleaned <- split_and_combine(df_ybg_entry,'YBG','EMPTY ENTRY')

df_exportimport_final <- bind_rows(df_tih_export_cleaned, df_yih_export_cleaned,df_tit_import_cleaned,d
df_exportimport_final[is.na(df_exportimport_final)] <- 0
df_exportimport_final$Date <- as.yearmon(df_exportimport_final$Date)
```

Level	PercentageChange	Difference	YearlyPercentageChange	YearlyDifference	DtePreviousYearPercentageChang
1190	450.925926	974	37.57225	325	98.9966
1135	-4.621849	-55	100.53004	569	89.7993
1007	-11.277533	-128	-26.87001	-370	68.3946
866	-14.001986	-141	-49.08877	-835	44.8160
882	1.847575	16	-29.15663	-363	47.4916
284	-67.800454	-598	-62.33422	-470	-52.5083

Final Data Frame after Preprocess

Now we have the single data frame that includes all we wanted.

Here is a preview of our single data frame.

```
kable(tail(df_exportimport_final)) %>%
  kable_styling("striped", full_width = F) %>%
  scroll_box(width = "100%", height = "400px")
```

Exploratory Data Analysis

In this section we will analyze and find some interesting insights from our dataset.

Summary of Dataset

We have 14 columns. 4 of them is character data type, 1 of them is date data type and rest of all is numeric data type.

You can see detailed summary statistics from below.

```
summary(df_exportimport_final)
```

```
##      Level         PercentageChange      Difference
##  Min.   :    0.0   Min.   : -100.000   Min.   :-32764.00
##  1st Qu.:    0.0   1st Qu.:   -9.253   1st Qu.:    -4.00
##  Median :   19.0   Median :    0.000   Median :     0.00
##  Mean   :  623.3   Mean   :    8.911   Mean   :     2.59
##  3rd Qu.:  285.0   3rd Qu.:    8.086   3rd Qu.:     6.00
##  Max.   :62182.0   Max.   :27900.000   Max.   : 24903.00
##  YearlyPercentageChange YearlyDifference   DtePreviousYearPercentageChange
##  Min.   :  -100.00      Min.   :-32668.0   Min.   : -100.000
##  1st Qu.:     -4.53     1st Qu.:    -1.0   1st Qu.:  -11.305
##  Median :      0.00     Median :     0.0   Median :    0.000
##  Mean   :     46.76     Mean   :    15.7   Mean   :   20.535
##  3rd Qu.:     11.48     3rd Qu.:     9.0   3rd Qu.:    5.634
##  Max.   :293850.00      Max.   : 21013.0   Max.   :19850.000
##  DtePreviousYearPercentageDifference MovingAverage        MovingSum
##  Min.   :-31509.00                   Min.   :    0.00   Min.   :     0
##  1st Qu.:    -3.00                   1st Qu.:    0.17   1st Qu.:     2
```

```
##  Median :     0.00                  Median :   20.50   Median :   225
##  Mean   :   -17.89                  Mean   :  613.53   Mean   :  7068
##  3rd Qu.:     4.00                  3rd Qu.:  288.77   3rd Qu.:  3253
##  Max.   : 26171.00                  Max.   :53219.17   Max.   :638630
##       Date      ExportImportCountry VehicleType        ExportImport
##  Min.   :2012   Length:50112        Length:50112       Length:50112
##  1st Qu.:2014   Class :character    Class :character   Class :character
##  Median :2017   Mode  :character    Mode  :character   Mode  :character
##  Mean   :2017
##  3rd Qu.:2019
##  Max.   :2022
##  ExportImportRegion
##  Length:50112
##  Class :character
##  Mode  :character
##
##
##
```

Missing Values

We don't have any missing values in our tables as you can see below, that is because we have already fill NA values with 0 in preprocess step.

```
colSums(is.na(df_exportimport_final))
```

```
##                                Level                     PercentageChange
##                                    0                                    0
##                           Difference               YearlyPercentageChange
##                                    0                                    0
##                     YearlyDifference     DtePreviousYearPercentageChange
##                                    0                                    0
## DtePreviousYearPercentageDifference                        MovingAverage
##                                    0                                    0
##                            MovingSum                                 Date
##                                    0                                    0
##                  ExportImportCountry                          VehicleType
##                                    0                                    0
##                         ExportImport                   ExportImportRegion
##                                    0                                    0
```

Most Import-Export Countries

The Word Cloud graphs in the below shows us most imported and exported countries based on country name size.

Most import countries as we can see below are EU countries such as Germany, Italy, France etc. On the other hand countries close to our border like Iraq, Iran, Bulgaria etc.

Import Plot

```r
#Import
ImportFreq <- df_exportimport_final %>% filter(ExportImport == 'IMPORT') %>% group_by(ExportImportCount

wordcloud2(data=ImportFreq, size=0.8)
```

Most export countries as we can see below are countries close to our borders. Top countries are usually in middle east or Asia, followed by EU countries.

Export Plot

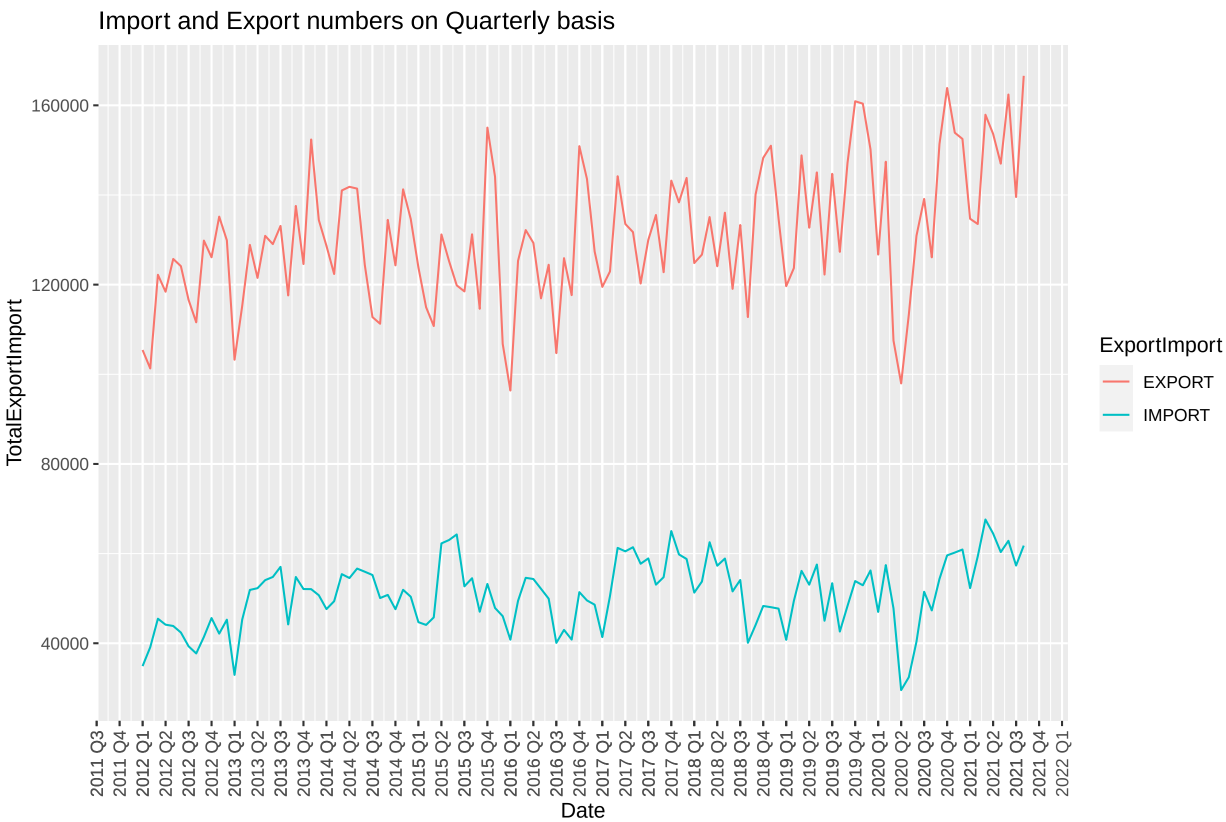
```r
#Export
ExportFreq <- df_exportimport_final %>% filter(ExportImport == 'EXPORT') %>% group_by(ExportImportCount

wordcloud2(data=ExportFreq, size=0.8)
```

Export-Import Trend

As seen in the graph below we can see parallel increase and decrease between Import and Export numbers on Quarterly basis. Moreover we can see drops on export and import numbers between fourth quarter and first quarter. We strongly believe this is due to winter conditions also we are seeing a huge drop in between 2020-Q1 and 2020-Q2 due to Corona Virus.

```
df_exportimport_final %>%
  filter(ExportImport == 'EXPORT' | ExportImport == 'IMPORT')  %>%
  group_by(Date,ExportImport) %>% summarize(TotalExportImport = sum(Level)) %>%
  ggplot( aes(x=Date, y=TotalExportImport, group=ExportImport, color=ExportImport)) +
  zoo::scale_x_yearqtr(n = 100,format = '%Y Q%q') +
  geom_line() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(title = "Import and Export numbers on Quarterly basis")
```

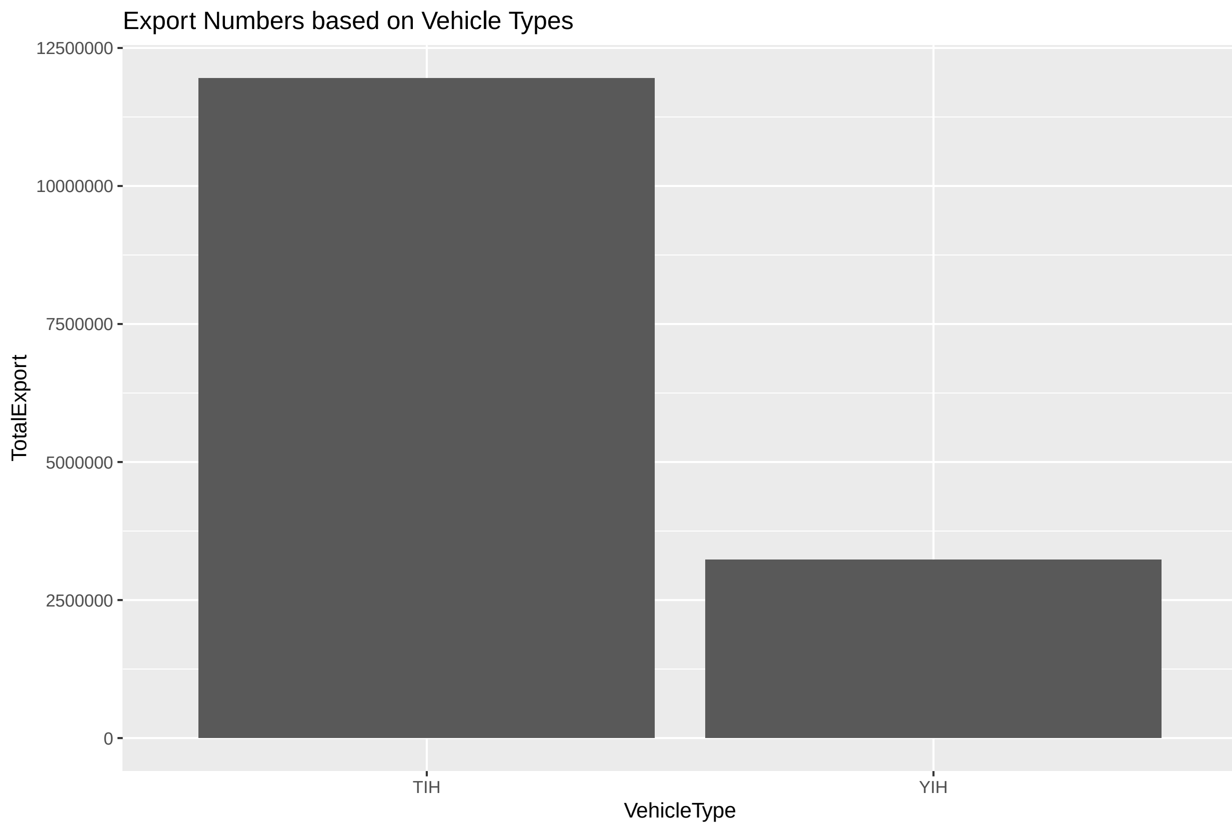


Export-Import Numbers Based On Vehicle Type (Turkish or Foreign Vehicles)

We can see in bar charts below Turkish vehicles mostly used for imports and exports.

Imports and Exports Percentages based on Vehicle Type also shown in tables.

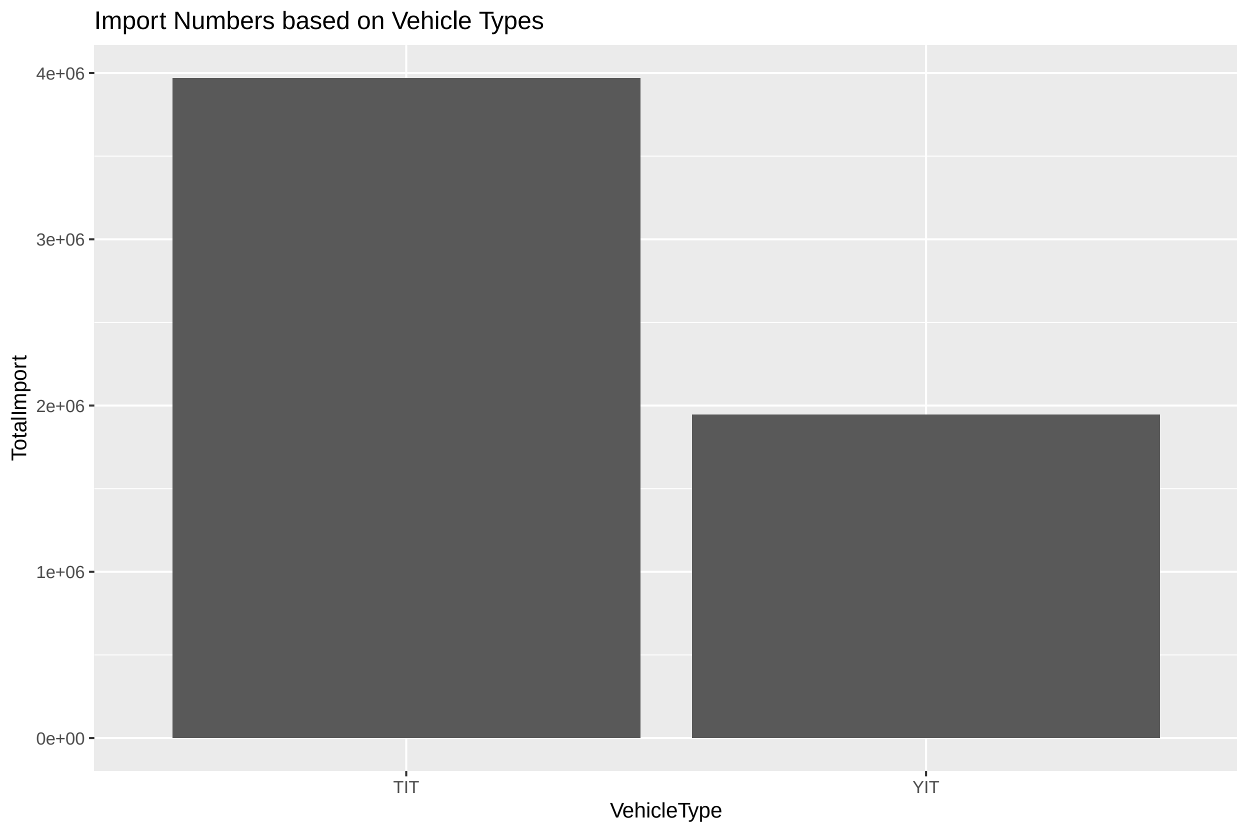
```
# Export
ExportVehicles <- df_exportimport_final %>%  filter(ExportImport == 'EXPORT') %>% group_by(VehicleType)
ggplot(ExportVehicles , aes(y=TotalExport, x=VehicleType)) +
  geom_bar(position="dodge", stat="identity") + labs(title = "Export Numbers based on Vehicle Types")
```




```
df_exportimport_final %>%  filter(ExportImport == 'EXPORT') %>% group_by(VehicleType) %>% summarize(Tota
```

```
## # A tibble: 2 x 2
##   VehicleType Percentage
##   <chr>            <dbl>
## 1 TIH               78.7
## 2 YIH               21.3
```

```
#Import
ImportVehicles <- df_exportimport_final %>%  filter(ExportImport == 'IMPORT') %>% group_by(VehicleType)
ggplot(ImportVehicles , aes(y=TotalImport, x=VehicleType)) +
  geom_bar(position="dodge", stat="identity") + labs(title = "Import Numbers based on Vehicle Types")
```



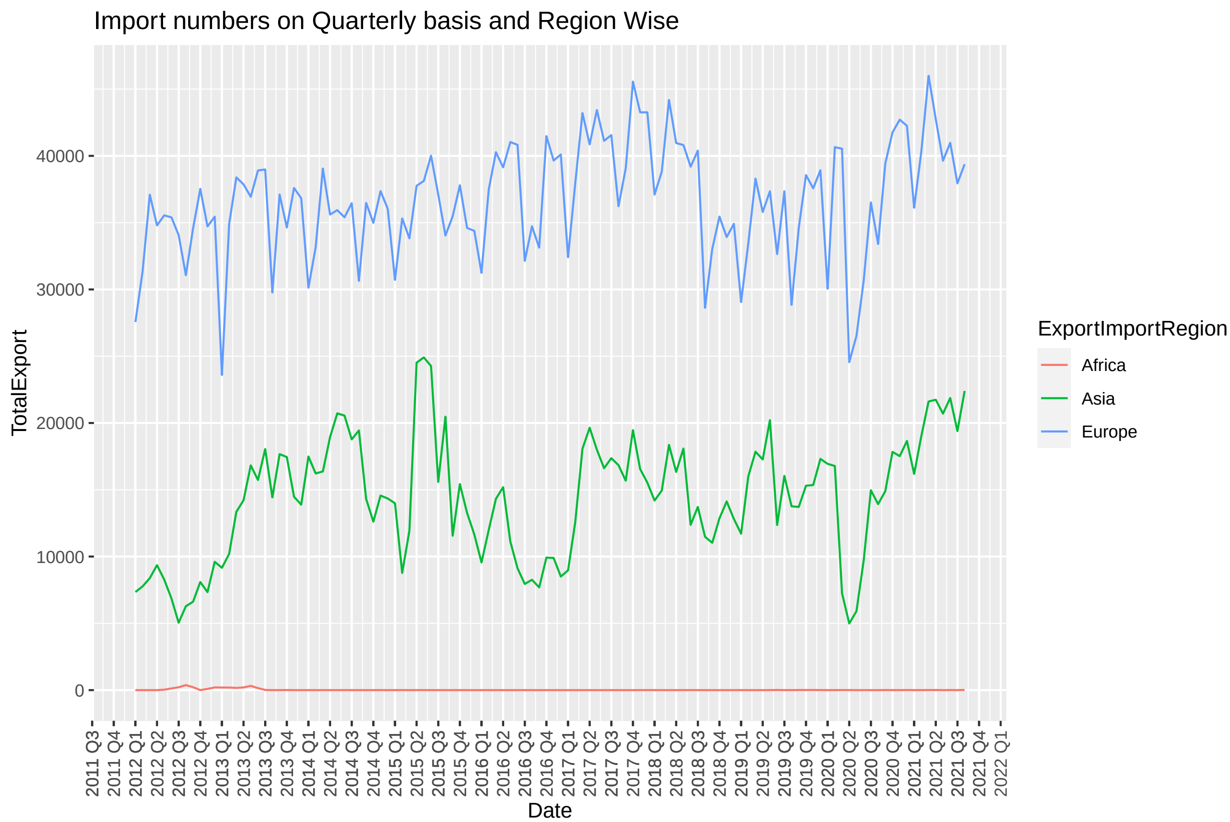

```
df_exportimport_final %>%  filter(ExportImport == 'IMPORT') %>% group_by(VehicleType) %>% summarize(Tota
```

```
## # A tibble: 2 x 2
##   VehicleType Percentage
##   <chr>            <dbl>
## 1 TIT               67.1
## 2 YIT               32.9
```

Export-Import Numbers Based On Region Wise

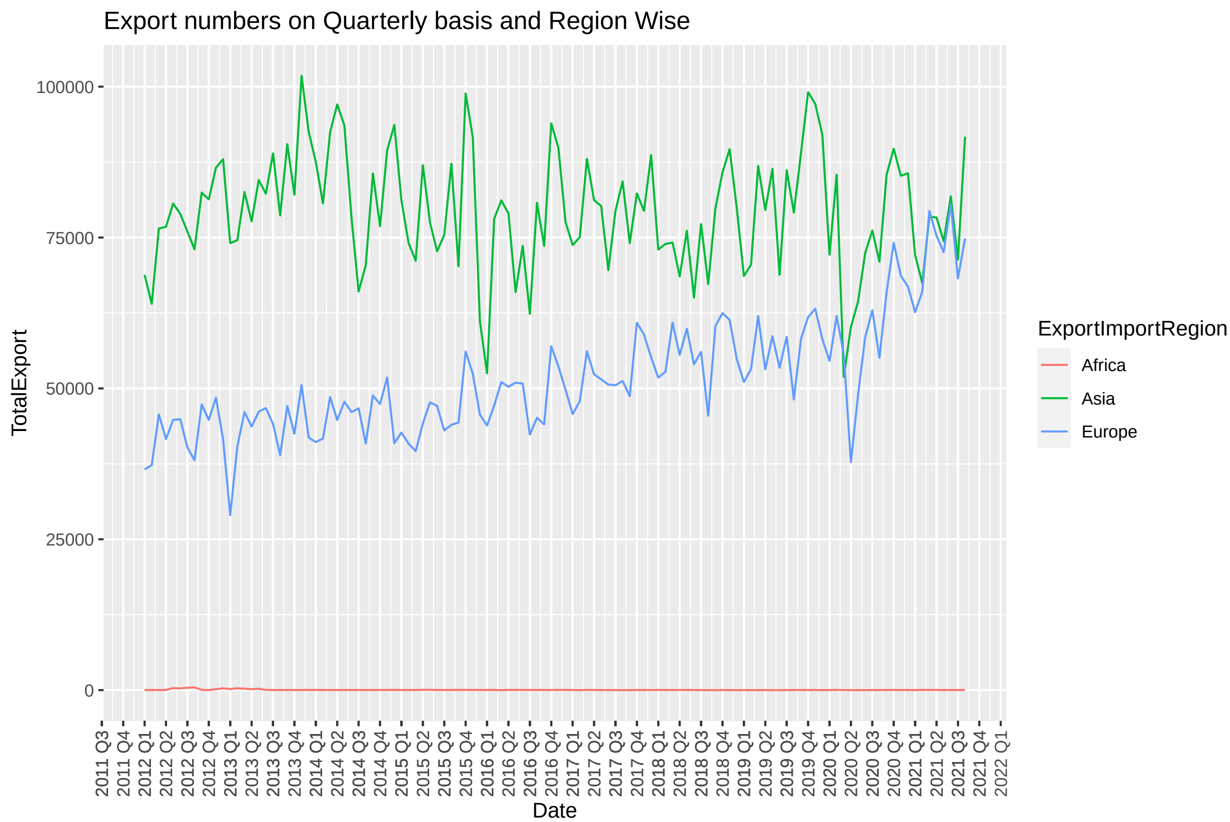
In this part we analyse our Import data based on Regions. As we can see in the plot below, import levels in Europe are way greater than Asia and Africa regions. Africa has the lowest import numbers among all. When we analyze the line running by quarters, import lines are quite bumpy. This irregularity can be considered as a factor of seasonal and political changes. In Africa region the line runs steadily.

```
df_exportimport_final %>%
  filter(ExportImport == 'IMPORT')  %>%
  group_by(Date,ExportImportRegion) %>% summarize(TotalExport = sum(Level)) %>%
  ggplot( aes(x=Date, y=TotalExport, group=ExportImportRegion, color=ExportImportRegion)) +
  zoo::scale_x_yearqtr(n = 100,format = '%Y Q%q') +
  geom_line() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(title = "Import numbers on Quarterly basis and Region Wise")
```



In the second plot we analyse our Export data based on Regions. As we can see in the plot below, export levels in Asia are greater than Europe region in the beginnings of our line chart, but export numbers of Europe has caught Asia numbers lately. Africa's line runs steady and has the lowest export numbers among all.

```
df_exportimport_final %>%
  filter(ExportImport == 'EXPORT')  %>%
  group_by(Date,ExportImportRegion) %>% summarize(TotalExport = sum(Level)) %>%
  ggplot( aes(x=Date, y=TotalExport, group=ExportImportRegion, color=ExportImportRegion)) +
  zoo::scale_x_yearqtr(n = 100,format = '%Y Q%q') +
  geom_line() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  labs(title = "Export numbers on Quarterly basis and Region Wise")
```


Export numbers on Quarterly basis and Region Wise
TotalExport
100000
75000
50000
25000
0
Date
2011 Q3
2011 Q4
2012 Q1
2012 Q2
2012 Q3
2012 Q4
2013 Q1
2013 Q2
2013 Q3
2013 Q4
2014 Q1
2014 Q2
2014 Q3
2014 Q4
2015 Q1
2015 Q2
2015 Q3
2015 Q4
2016 Q1
2016 Q2
2016 Q3
2016 Q4
2017 Q1
2017 Q2
2017 Q3
2017 Q4
2018 Q1
2018 Q2
2018 Q3
2018 Q4
2019 Q1
2019 Q2
2019 Q3
2019 Q4
2020 Q1
2020 Q2
2020 Q3
2020 Q4
2021 Q1
2021 Q2
2021 Q3
2021 Q4
2022 Q1
ExportImportRegion
Africa
Asia
Europe