Question 5

Tashen Naidoo

2023-11-27

Relevant packages

```
library(tidyverse);library(gt);library(tbl2xts);library(PerformanceAnalytics);library(lubridate);library
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.0     v purrr   1.0.1
## v tibble  3.2.1     v dplyr   1.1.3
## v tidyr   1.3.0     v stringr 1.5.0
## v readr   2.1.4     v forcats 0.5.2

## Warning: package 'tibble' was built under R version 4.2.3

## Warning: package 'dplyr' was built under R version 4.2.3

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

## Warning: package 'gt' was built under R version 4.2.3

## Loading required package: xts

## Warning: package 'xts' was built under R version 4.2.3

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 4.2.3

##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
##
##
## ######################### Warning from 'xts' package ##########################
## #                                                                             #
## # The dplyr lag() function breaks how base R's lag() function is supposed to  #
```

```
## # work, which breaks lag(my_xts). Calls to lag(my_xts) that you type or       #
## # source() into this session won't work correctly.                            #
## #                                                                             #
## # Use stats::lag() to make sure you're not using dplyr::lag(), or you can add #
## # conflictRules('dplyr', exclude = 'lag') to your .Rprofile to stop           #
## # dplyr from breaking base R's lag() function.                                #
## #                                                                             #
## # Code in packages is not affected. It's protected by R's namespace mechanism #
## # Set `options(xts.warn_dplyr_breaks_lag = FALSE)` to suppress this warning.  #
## #                                                                             #
## ###############################################################################
##
## Attaching package: 'xts'
##
## The following objects are masked from 'package:dplyr':
##
##     first, last
##
##
## Attaching package: 'PerformanceAnalytics'
##
## The following object is masked from 'package:graphics':
##
##     legend
##
## Loading required package: timechange
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union

## Warning: package 'RcppRoll' was built under R version 4.2.3

## Warning: package 'rugarch' was built under R version 4.2.3

## Loading required package: parallel
##
## Attaching package: 'rugarch'
##
## The following object is masked from 'package:purrr':
##
##     reduce
##
## The following object is masked from 'package:stats':
##
##     sigma

## Warning: package 'forecast' was built under R version 4.2.3

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

```
list.files('/code',full.names = T, recursive = T) %>%
    as.list() %>%
    walk(~source(.))
```

Question Five

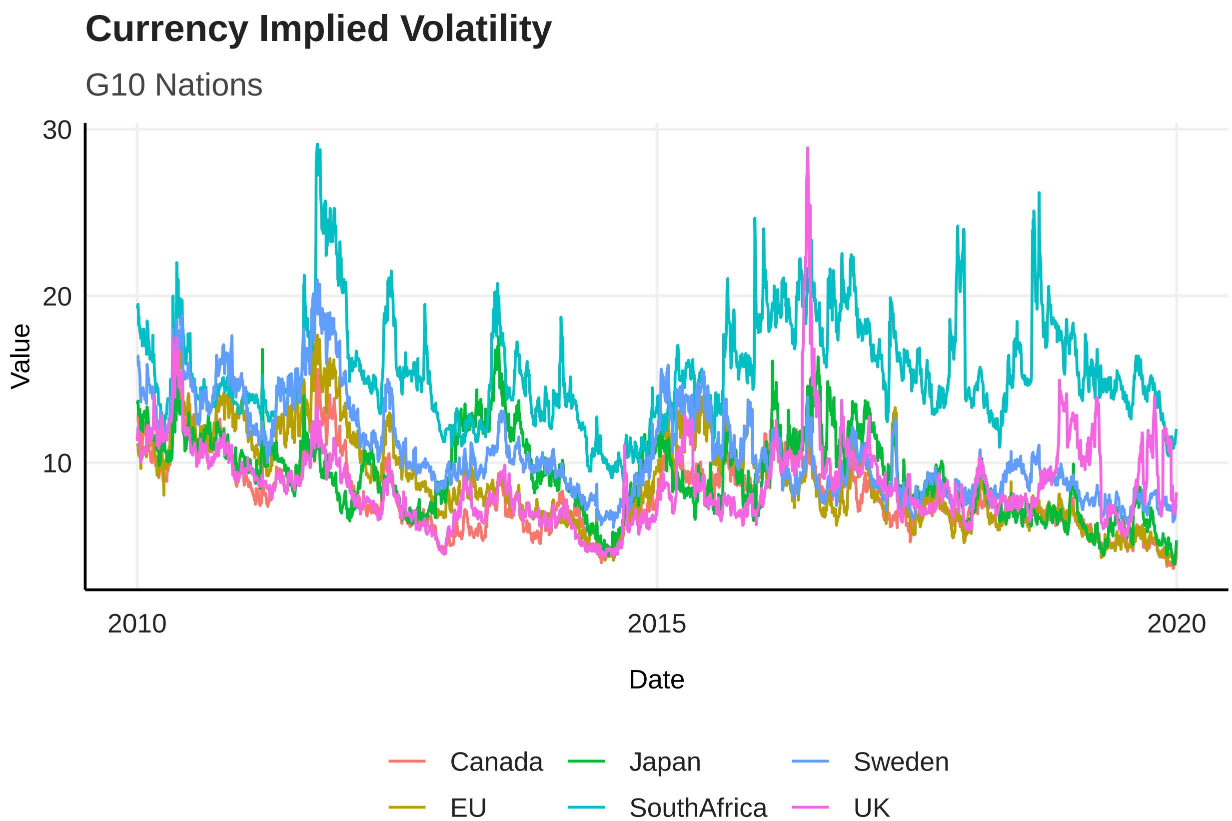
This question provides a lot of information relating to currencies. I have decided to utilise the *cncy_IV* and the *cncy* data sets, as I thought it was the most applicable to this question after reading the text files provided for each data set.

```
cncy <- read_rds("data/currencies.rds")
cncy_Carry <- read_rds("data/cncy_Carry.rds")
cncy_value <- read_rds("data/cncy_value.rds")
cncyIV <- read_rds("data/cncyIV.rds")
bbdxy <- read_rds("data/bbdxy.rds")
```

The *cncyIV* data set provides an indication of the future volatility of a currency over time. These values are based on what the market foresees in the future. I strictly want to look at the periods between 2010 and 2020.

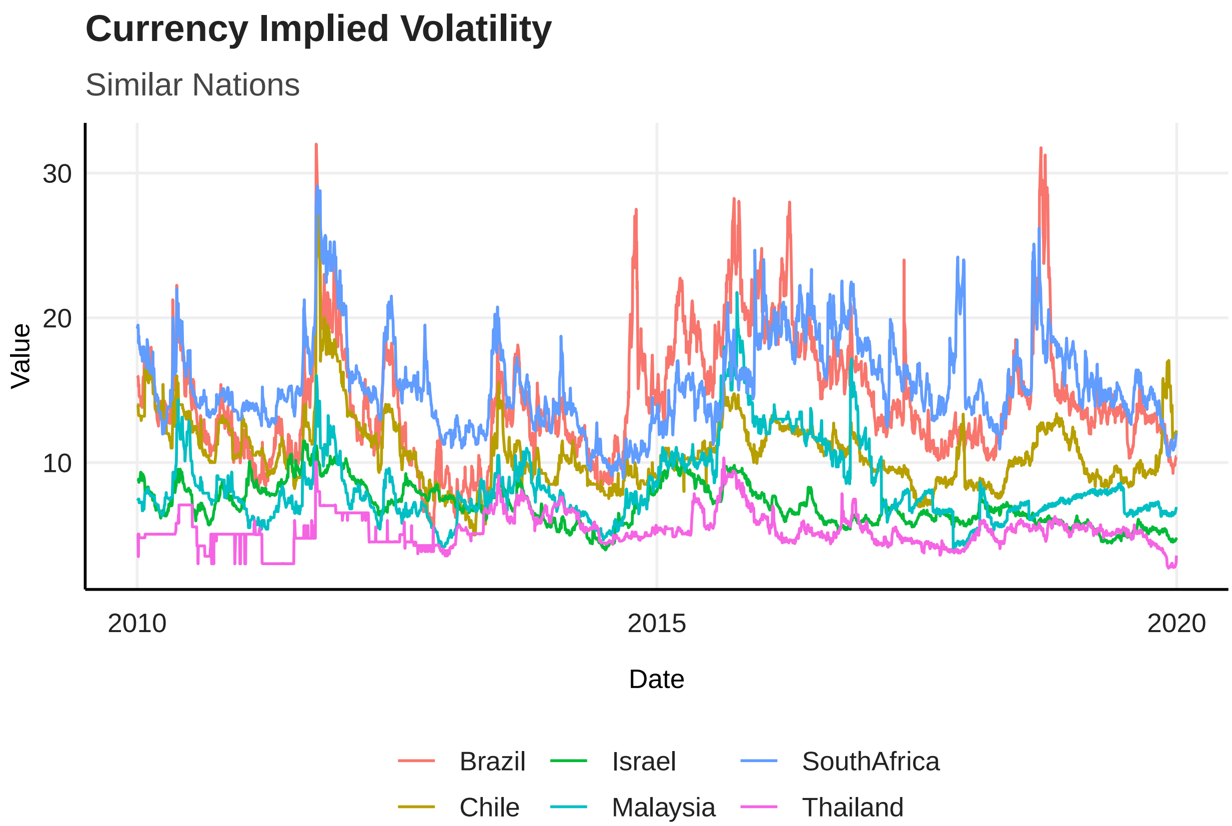
I restrict the number of nations to only look at the G10 nations, however, there are only five G10 nations present in the data set. I utlise the EU information as a proxy for France, as they are not in the data set, but use the Euro.

```
#G10 nations from data set
G10_nations <- c("Canada", "Japan", "Sweden", "UK", "EU", "SouthAfrica")

#Clean the data sets
cncy <- cncy %>%
    spread(Name, Price)
colnames(cncy) <- gsub("_Cncy", "", colnames(cncy))
colnames(cncy) <- gsub("_Inv", "", colnames(cncy))
cncy <- cncy %>%
    gather(Name, Price, -date)

cncyIV <- cncyIV %>%
    spread(Name, Price)
colnames(cncyIV) <- gsub("_IV", "", colnames(cncyIV))
cncyIV <- cncyIV %>%
    gather(Name, Price, -date)

#The data set is fairly large, so I want to further refine it by looking at specific nations.
cncyIV %>%
    filter(date > lubridate::ymd(20091231) & date < lubridate::ymd(20200101)) %>%
    filter(Name %in% G10_nations) %>%
    ggplot()+
    geom_line(aes(date, Price, color = Name))+
    ylab("Value")+
    xlab("Date")+
    labs(title = "Currency Implied Volatility",
         subtitle = "G10 Nations")+
    fmxdat::theme_fmx()
```



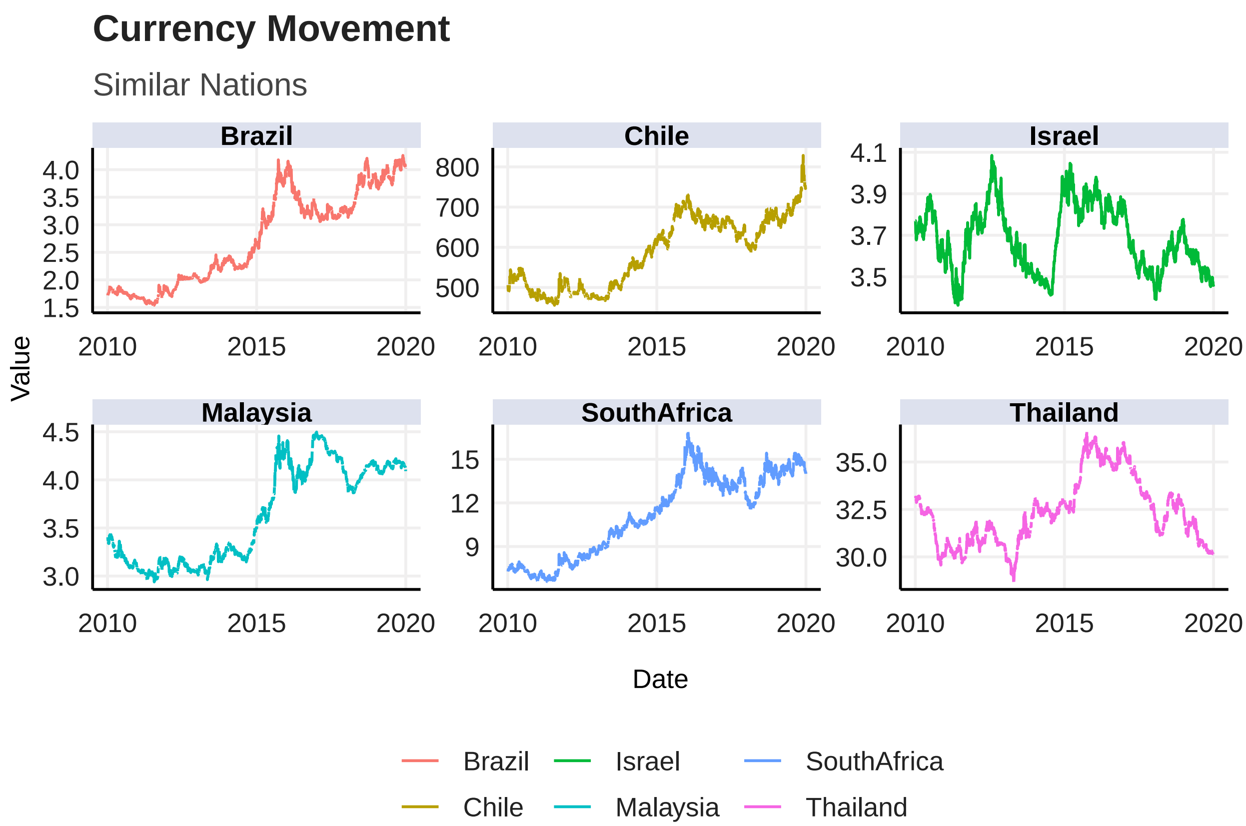
We note that South Africa has a high expected volatility compared to the rest of the G10 nations that are present in the data set. Let's look at the rankings of South Africa over time, compared to nations of similar status. I only intend to look at the periods between 2010 and 2020.

```
#similar nations
same <- c("Brazil","Chile","Israel","Malaysia", "Thailand", "SouthAfrica")
cncyIV %>% filter(date > lubridate::ymd(20091231) & date < lubridate::ymd(20200101)) %>%
    filter(Name %in% same) %>%
    ggplot()+
    geom_line(aes(date, Price, color = Name))+
    ylab("Value")+
    xlab("Date")+
    labs(title = "Currency Implied Volatility",
         subtitle = "Similar Nations")+
    fmxdat::theme_fmx()
```



South Africa remains to have this high expectation of volatility in the future periods. Let's take a look at the currency movement over time.

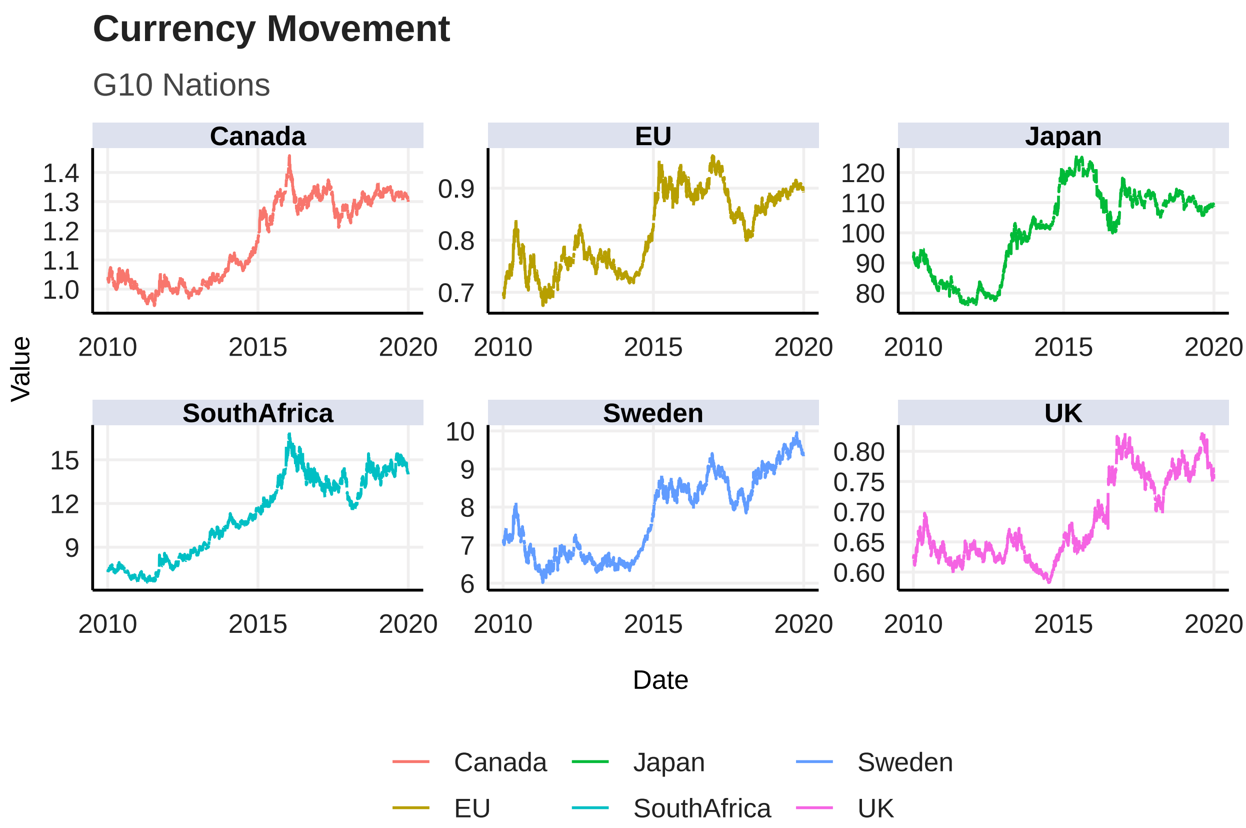
```
cncy %>% filter(Name %in% same) %>%
    filter(date > lubridate::ymd(20091231) & date < lubridate::ymd(20200101)) %>%
    ggplot()+
    geom_line(aes(date, Price, color = Name))+
    facet_wrap(~Name, scales = "free", ncol = 3)+
    ylab("Value")+
    xlab("Date")+
    labs(title = "Currency Movement",
         subtitle = "Similar Nations")+
    fmxdat::theme_fmx()
```



We note that the movement overall to nations similar to South Africa is fairly similar, except for Israel. This can seem puzzling of why the expected volatility of South Africa stands out compared to nations that have a similar movement as the Rand.

Let's look at the Rand compared to the G10 nations.

```
cncy %>% filter(Name %in% G10_nations) %>%
    filter(date > lubridate::ymd(20091231) & date < lubridate::ymd(20200101)) %>%
    ggplot()+
    geom_line(aes(date, Price, color = Name))+
    facet_wrap(~Name, scales = "free", ncol = 3)+
    ylab("Value")+
    xlab("Date")+
    labs(title = "Currency Movement",
         subtitle = "G10 Nations")+
    fmxdat::theme_fmx()
```



The movement of the currency seems relatively the same to the G10 nations, however, this is vastly different from the Implied Volatility graph above.

Let's take a look at the financial table, as this will provide us with a better indication of the currencies over time.

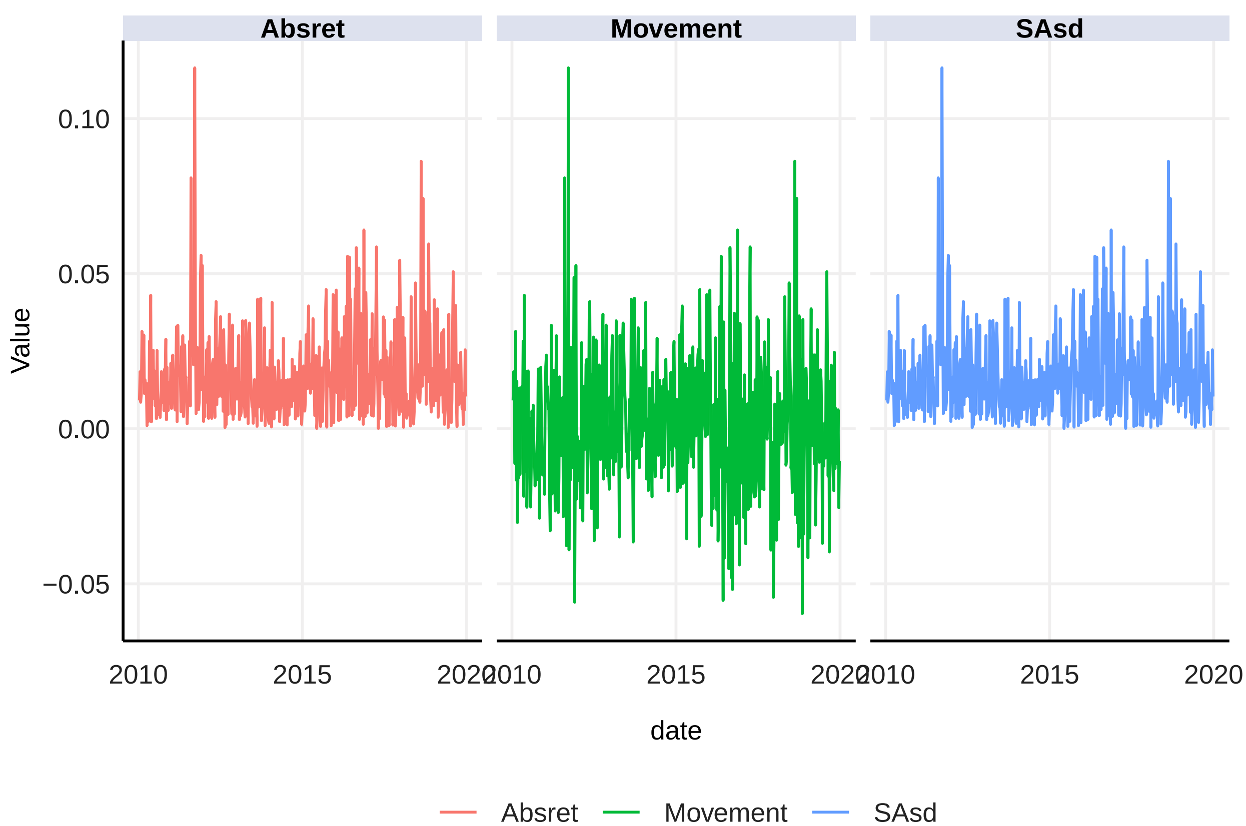
```
# #first need to create a simple return for this
# tem <- cncy %>%
#      filter(date > lubridate::ymd(20091231) & date < lubridate::ymd(20200101)) %>%
#     na.locf(cncy$Price, na.rm = FALSE,fromLast = TRUE, maxgap = 4) %>%
#     mutate(Year = format(date, "%Y"), Month = format(date, "%B"), Day = format(date, "%A")) %>%
#     group_by(Year, Month, Day) %>%
#     filter(Day == "Thursday") %>%
#     ungroup() %>%
#     group_by(Name) %>%
#     mutate(Return = Price/lag(Price)-1) %>%
#     filter(date>first(date))
#
# temg10 <- tem %>%
#     filter(Name %in% G10_nations) %>%
#     select(date, Name, Return) %>%
#     arrange(date)
#
# BM <- "SouthAfrica"
# Yrs_lookback <- 3
# NA_Check <- 0.9
#
```

```
# Statistics_table <- function(temg10, BM, Yrs_lookback, NA_Check){
#
#     data_included <- temg10 %>%
#         filter(date >= fmxdat::safe_year_min(datesel = last(date), N = Yrs_lookback))
#
#     data_considered <- data_included %>%
#         group_by(Name) %>%
#         summarise(N_noNA = sum(!is.na(Return))/length(unique(data_included$date))) %>%
#         filter(N_noNA > NA_Check) %>%
#         pull(Name)
#
#     data_xts <- data_included %>%
#         filter(Name %in% data_considered) %>%
#         tbl_xts(cols_to_xts = Return, spread_by = Name, Colnames_Exact = T)
#
#     Bench_xts <- data_included %>%
#         filter(Name %in% BM) %>%
#         tbl_xts(cols_to_xts = Return, Colnames_Exact = T)
#
#     library(PerformanceAnalytics)
#
#     Moments <- bind_rows(
#         data.frame(Return.cumulative(data_xts) ) %>% round(., 3),
#         data.frame(Return.annualized(data_xts, scale = 12, geometric = T)) %>% round(., 3),
#         data.frame(PerformanceAnalytics::Return.annualized.excess(data_xts, Bench_xts) ) %>% round(.,
#         data.frame(sd.annualized(data_xts, scale = 12, geometric = T)) %>% round(., 3),
#         data.frame(PerformanceAnalytics::AdjustedSharpeRatio( data_xts ) ) %>% round(., 3),
#         data.frame(PainIndex(data_xts, scale = 12, geometric = T)) %>% round(., 3),
#         data.frame(AverageDrawdown(data_xts, scale = 12)) %>% round(., 3),
#         data.frame(fmxdat::Safe_TE(Ra = data_xts, Rb = Bench_xts, scale = 12)) %>% round(., 3),
#         data.frame(PerformanceAnalytics::InformationRatio(Ra = data_xts, Rb = Bench_xts)) %>% round(.
#         data.frame(PerformanceAnalytics::CAPM.beta(Ra = data_xts, Rb = Bench_xts, Rf = 0)) %>% round(
#         data.frame(PerformanceAnalytics::CAPM.beta.bull(Ra = data_xts, Rb = Bench_xts, Rf = 0)) %>% r
#         data.frame(PerformanceAnalytics::CAPM.beta.bear(Ra = data_xts, Rb = Bench_xts, Rf = 0)) %>% r
#         data.frame(PerformanceAnalytics::UpDownRatios(Ra = data_xts, Rb = Bench_xts, method = "Percen
#         data.frame(PerformanceAnalytics::CVaR(R = data_xts, p = 0.05, method = "modified")) %>%
#             round(., 3)
#     ) %>%
#
#     tibble::rownames_to_column("Info") %>%
#         mutate(Period = glue::glue("Last {Yrs_lookback} Years"),
#                Info = c("Cum Return", "Return (Ann.)", "Return Excess (Ann.)", "SD (Ann.)", "Adj. Sha
#         relocate(Period, .before = Info) %>%
#         as_tibble()
#
#     # This line replaces the `.` with a space.
#     # Note the forward slashes, as `.` there means everything, `\\.` means a full-stop
#     colnames(Moments) <- gsub("\\.", " ", colnames(Moments))
#     Moments
# }
#
#
#
```

```
# Table <- bind_rows(
# Statistics_table(temg10, BM, Yrs_lookback = 3, NA_Check),
# Statistics_table(temg10, BM, Yrs_lookback = 5, NA_Check) )
#
# Make_perc <-c( "Cum Returns", "Returns (Ann.)", "Information Ratio", "SD (Ann.)",
# "Avg DD", "Modified CVaR")
#
# Rows_to_Perc <-
# Table %>%
#     mutate(RN = row_number()) %>%
#     filter(Info %in% Make_perc) %>%
#     pull(RN)
# colnams <- colnames(Table)[-1:-2]
# Cols_length <- ncol(Table)
# library(gt)
#
#
# tab <-
# Table %>%
# gt::gt(groupname_col = 'Period', caption = 'Currency Moments Comparison') %>%
# tab_header(title = glue::glue("Currency Statistics: Relative to {BM}")) %>%
# fmt_percent(
# columns = 3:Cols_length,
# rows = Rows_to_Perc,
# decimals = 1
# ) %>%
# sub_missing(
# columns = all_of(colnams),
# missing_text = '-'
# ) %>%
# tab_style(
# style = list(
# cell_fill(color = 'gray27', alpha = 0.15),
# cell_text(size = 'large', weight = 'bold',align = 'left')
# ),
# locations = cells_row_groups())
# tab %>%
# tab_options(data_row.padding = px(4),table.width = pct(100),
# column_labels.font.size = pct(50),
# column_labels.vlines.width = 1, table.font.size = pct(80)) %>%
# tab_options(data_row.padding = px(6),
# column_labels.font.size = pct(100)) %>%
# tab_style(style = cell_text(weight = 1200, align = 'left'),locations = cells_title(groups = 'title'))
# 5
# tab_style(style = cell_text(color = 'darkgrey', transform = 'uppercase', align = 'center'),
# locations = cells_column_labels(everything()))
#
```

Unfortunately, I ran into errors with this table and I am not sure why. I would like to look at the rand movement over time and assess the volatility of it to provide more insight into whether Volatility Index is correct for the ZAR. The GARCH for the South African Rand.

```r
SA <- cncy %>%
    filter(Name == "SouthAfrica") %>%
     filter(date > lubridate::ymd(20091231) & date < lubridate::ymd(20200101)) %>%
    na.locf(cncy$Price, na.rm = FALSE,fromLast = TRUE, maxgap = 4) %>%
    mutate(Year = format(date, "%Y"), Month = format(date, "%B"), Day = format(date, "%A")) %>%
    filter(Day == "Wednesday") %>% #look at weekly data, as daily rates may not change whcih will provi
    mutate(Movement = Price/lag(Price)-1) %>%
    filter(date > first(date)) %>%
    select(-Year, -Month, -Day)

   SA %>%
    mutate(SAsd = sqrt(Movement^2)) %>%
    mutate(Absret = abs(Movement)) %>%
    select(Movement, SAsd, Absret, date) %>%
    gather(Type, Value, -date) %>%
    arrange(date) %>%
    ggplot() +
geom_line(aes(date, Value, color = Type)) +
facet_wrap(~Type) +
fmxdat::theme_fmx() +
    labs(color = F)
```

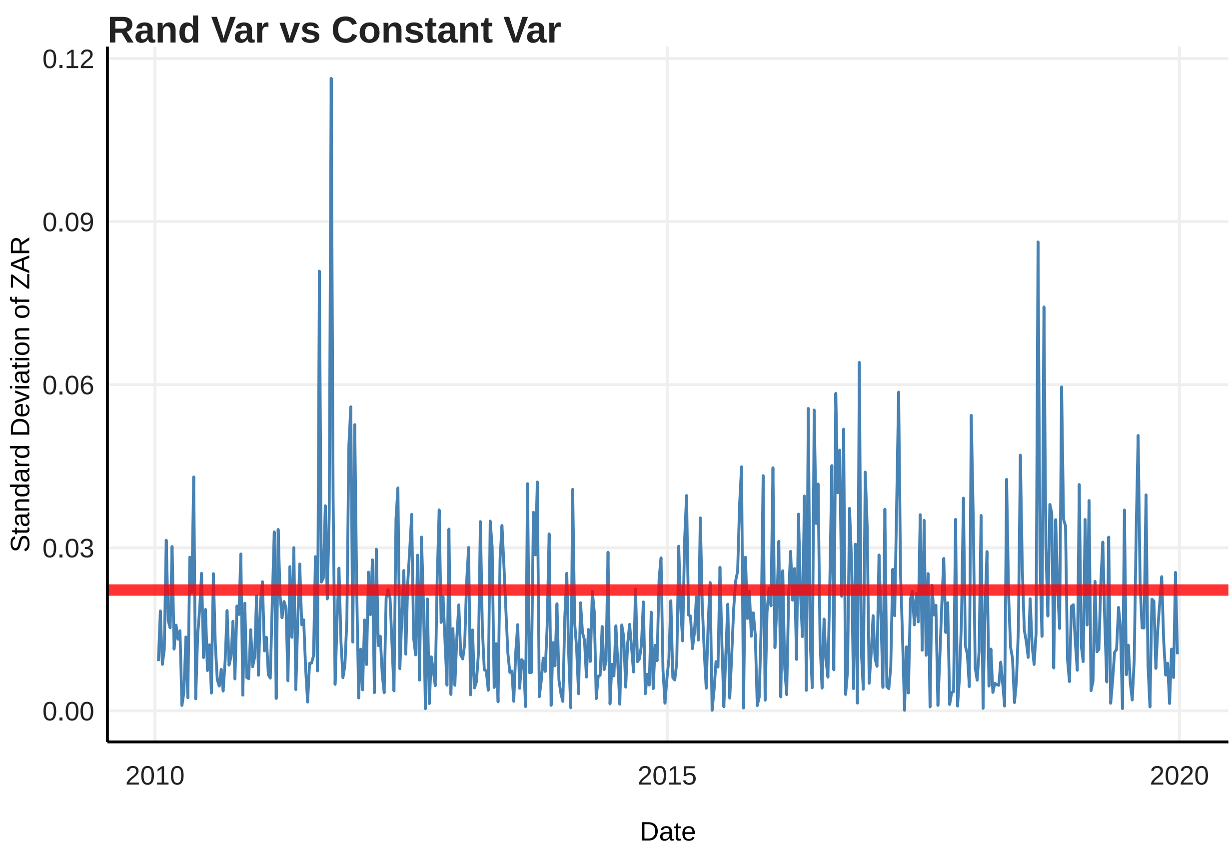


As the figure shows, there is high movement in the Rand. The standard deviation is volatile, which means that the Rand value is constantly fluctuating. Let's look at the Unconditional volatility.

```
SA %>%
mutate(SAsd = sqrt(Movement^2)) %>%
mutate(Constant_var = sd(Movement)) %>%
ggplot() +
geom_line(aes(date, SAsd), color = "steelblue") +
geom_hline(aes(date, yintercept = mean(Constant_var)), color = "red",
    alpha = 0.8, size = 2) +
    ylab("Standard Deviation of ZAR")+
    xlab("Date")+
fmxdat::theme_fmx() +
labs(title = "Rand Var vs Constant Var")
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

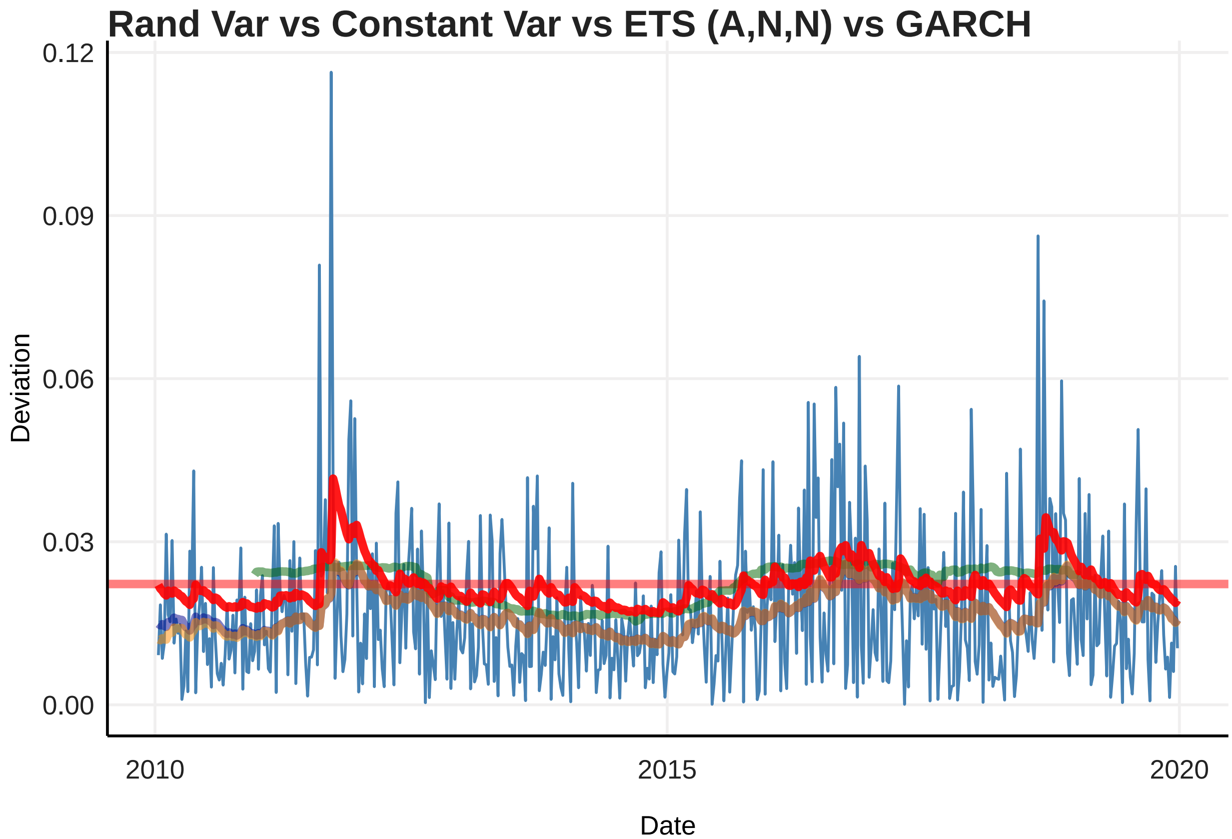
```
## Warning in geom_hline(aes(date, yintercept = mean(Constant_var)), color =
## "red", : Ignoring unknown aesthetics: x
```



We note a massive difference between the constant variance (red horizontal line) and the movement of the Rand throughout the period. This solidifies that the variance should not be used as a true indicator of risk/volatility. The red line represents heteroskedasticity, we note that the rand variance, in blue is not heteroskedastic.

```r
SA <- SA %>%
    mutate(SAsd = sqrt(Movement^2))
Randxts <- SA %>%
    tbl_xts(., cols_to_xts = Movement)

garch_model <- fGarch::garchFit(formula= ~arma(1,1) + aparch(1,1), data = Randxts , trace = FALSE)
ets_sd <- ets(SA %>% tbl_xts(cols_to_xts = SAsd), model = "ANN")
ets_sd_mult <- ets(1e-8 + SA %>% tbl_xts(cols_to_xts = SAsd), model = "MNN")
std_t <- garch_model@sigma.t

left_join(
  SA,
  tibble( date = unique(SA$date), GARCH = std_t), by = "date") %>%

left_join(
  .,
  tibble( date = unique(SA$date), ETS = as.numeric(ets_sd$fitted)), by = "date") %>%

  left_join(.,tibble( date = unique(SA$date), ETS_Mult = as.numeric(ets_sd_mult$fitted)), by = "date")
  mutate(Constant_var = sd(Movement)) %>%
  mutate(Roller = roll_sd(Movement, n = 100, fill = NA) ) %>%

  ggplot() +

  geom_line(aes(date, SAsd), color = "steelblue" ) +

  geom_hline(aes(date, yintercept = mean(Constant_var)), color = "red", alpha  = 0.5, size = 1.5) +

  geom_line(aes(date, y = Roller), color = "darkgreen", alpha  = 0.5, size = 1.5) +

  geom_line(aes(date, y = ETS), color = "darkblue", alpha  = 0.5, size = 1.5) +

  geom_line(aes(date, y = ETS_Mult), color = "darkorange", alpha  = 0.5, size = 1.5) +

  geom_line(aes(date, y = GARCH), color = "red", alpha  = 0.9, size = 1.5) +

  fmxdat::theme_fmx() +

  labs(title = "Rand Var vs Constant Var vs ETS (A,N,N) vs GARCH") +
    ylab("Deviation")+
    xlab("Date")+
    fmxdat::theme_fmx()
```

```
## Warning in geom_hline(aes(date, yintercept = mean(Constant_var)), color =
## "red", : Ignoring unknown aesthetics: x

## Warning: Removed 99 rows containing missing values (`geom_line()`).
```



The figure above displays the true movement of the Rand over time. We note that the Exponential Smoothing State Space Model (in darkorange) is the best model to use. It displays an accurate depiction of the ZAR throughout the period.