Emerging Markets indecies correlation to GDP

An exploration into four emerging markets. This paper will explore the correlation of FTSE indicies and GDP.

Even Oscar Harlert

2023-10-18

Introduction

Emerging markets can be great investment opportunity. In order to make good investment decisions and understand the risk and return for any investment you have to analyze historical data.

To measure countries development there are several metrics that can be useful. In this paper we will use keep things simple and only use one metric and that is gross domestic product (GDP onwards). We could use a lot more metrics but to limit the scope of this paper we are only using GDP. In a separate dataset we will have the population for the countries in question.

Research question

This paper will have a limited scope. We are seeking to investigate indices data representing each country. The countries selected for this paper are India, Egypt, China and Brazil.

- How has the historical development been in indices funds been in India, Egypt, China and Brazil?

- What has the historical risk been?

- What degree of correlation does the indices have with the GDP for each country?

In this paper, we consider risk to be standard deviation of the dataset.

Data sets

There are three datasets acquired for this paper. The first one is downloaded from refinitive/datastream. For more information regarding the data provider you can follow this link. This will be refereed to as dataset one. The data in the file are time-series spanning from 19. September 2003 to 19. September 2023. The measurements are done with a monthly interval. All data points are represented in US dollar.

The second data set is downloaded from The World Bank. The data-set is actually divide into two files but will be combined in RStudio as they each only contain one metric of data, GDP and population.

Project libraries

This project will make use of several libraries in RStudio. In order to use them we need to load them. To avoid any problems with replicating the code I've made a file to load all. This file also contains a comment with an install command if you are missing any of them on your local setup. To access the install command you can mark the area in the file and press "CTRL+Shift+C". This will remove the comment and allow you to press CTRL+ENTER to execute the code and install dependencies.

To make the code cleaner, I've made a script folder and separated the required libraries in that file. The command above simply executes the commands in that file. The following are the lines that has been executed.

```
library(here)
```

```
here() starts at C:/Users/evenhar/Documents/GitHub/R_folder/Assignments/Emerging_markets
```

```
library(tidyverse)
```

```
-- Attaching core tidyverse packages ----------------------- tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.4
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.4.4     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.0
v purrr     1.0.2
```

```
-- Conflicts ---------------------------------------- tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
```

```r
library(readxl)
library(psych)
```

```
Attaching package: 'psych'

The following objects are masked from 'package:ggplot2':

    %+%, alpha
```

```r
library(janitor)
```

```
Attaching package: 'janitor'

The following objects are masked from 'package:stats':

    chisq.test, fisher.test
```

```r
library(readxl)
```

NB! We will add any library we add and check it at the end to assure that the project is not missing any dependencies.

With all the libraries loaded we are ready to start working with our data sets. To be able to replica the current sets I've made a github repo where you can access the files, they can be accessed here. Dataset two is accessible from World Bank DataBank. However, the refinitive dataset would required you to have a refinitive account. Therefore, to assure that the project is reproducible I made the repository linked above.

In all, we have three excel files in the folder datasets that can be accessed in the project folder.

Import, tidy & transformation

This section will be split into three sections:

1. Import, tidy and transform dataset one

2. Import tidy and transform dataset two

3. We join the datasets into dataframes we can use for visualzation and modelling.

Dataset one

Initially we want to start to access the data we collected in the dataset folder. The data is stored as a excel file that has been stored in the project folder. To make it possible to work with in r we start by importing the file and puting the data in a dataframe called index_emerging_markets.

```r
index_emerging_markets <- read_excel("datasets/FTSE 20.09.23 - EOH.xlsx", skip = 6)

index_emerging_markets <- index_emerging_markets |>
  rename(
    Date = Name
  )
```

We changed to name of the coulmn for dates to dates from names with the second row. The data in this file is currently in absolute values of the indices. To be able to make comparisons we need to normalize the data with the start of 100. In order to do so we are using the following line:

```r
normalized <- index_emerging_markets |>
  mutate(
    FTSEWorldN = FTSEWorld / (146.2 / 100),
    IndiaN = India / (332.5 / 100),
    EgyptN = Egypt / (59.04 / 100),
    ChinaN = China / (672.28 / 100),
    BrazilN = Brazil / (150.65 / 100),
  )
```

The code above adds 6 new columns where the data is normalized and starts at 100 on the 19. September 2003. The code snippet above can be generalized as the following:

Initial value / (Initial value / 100). The initial value divided by 100 will be exectued on each of the observation on each row. This results in a graph where we can compare each indicies including our benchmark (FTSE World).

4

To save some space we will combine a couple of lines of code:

```
normalized_trimmed <- normalized |>
  select(Date, FTSEWorldN:BrazilN)

normalized_trimmed <- normalized_trimmed |>
  mutate(
    FTSEWorld_Log = c(NA, log(FTSEWorldN[-1] / FTSEWorldN[-nrow(normalized)])),
    India_Log = c(NA, log(IndiaN[-1] / IndiaN[-nrow(normalized)])),
    EgyptN_Log = c(NA, log(EgyptN[-1] / EgyptN[-nrow(normalized)])),
    China_Log = c(NA, log(ChinaN[-1] / ChinaN[-nrow(normalized)])),
    Brazil_Log = c(NA, log(BrazilN[-1] / BrazilN[-nrow(normalized)])),
  )

normalized_trimmed <- normalized_trimmed |>
  mutate(
    FTSE_per = FTSEWorld_Log * 100,
    India_Per = India_Log * 100,
    Egpyt_per = EgyptN_Log * 100,
    China_per = China_Log * 100,
    Brazil_per = Brazil_Log * 100,
  )

log_return <- select(normalized_trimmed,
    -"FTSEWorldN":-"BrazilN"
  )

log_return <- log_return[-1, ]

desc_data <- describe(normalized_trimmed) |>
  t()
```

Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf

Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

```
desc_data <- as.data.frame(desc_data)

desc_data <- desc_data[-c(1, 5:9), ]

desc_data <- select(desc_data,
```

```
                         -"FTSE_per":-"Brazil_per"
                         )

final_desc_data_log <- select(desc_data,
                              FTSEWorld_Log:Brazil_Log
                              )

write.csv(final_desc_data_log, "exported_data/desc_data_log.csv")
```

The lines above does several things, in short:

- We add rows for log(10) returns (most commonly used while working with historical returns in finance).

- We create new dataframes that separates data.

- We use describe() (a function from the psych package), it returns descriptive statistics from the dataset and we store this in a new dataframe called desc_data.

- A few more selective operations followed by an export to a csv if we at a later point would like to use the descriptive data from the project.

At this point we have dataframes with the data organized to make further analysis later in the project.

As mentioned earlier we used a function called describe, it returns the data in a frame where we have some abbreviations, for anyone not familiar with statistics they are the following:

Table 1: Describ() explination

Short	Definition
n	Number of observations
mean	The avarage
sd	Standard deviation (this is considered risk in financial terms)
skew	Skewness is the degree of asymmetry observed in a probability distribution (source).
kurtosis	An indicator of the tail of the distribution (source).
se	Standard error, and estimation of the standard deviation.

Dataset two

The second dataset is actually two excel files downloaded from the World Bank. Again, we need to load the data into a dataframe:

```
countries <- c("India", "Egypt, Arab Rep.", "China", "Brazil")

gdp <- read_excel("datasets/gdp.xls", skip = 3)

population <- read_excel("datasets/population.xls", skip = 3)
```

In addition to loading two dataframes with the data from the files, we also stored the four values (the names of the countries we are working with.

The datsets are filled with data we do not have any use for therefore we need to tidy it:

```
gdp <- clean_names(gdp)
population <- clean_names(population)


gdp_clean <- select(gdp,
  -"x1960":-"x2002",
  -"indicator_code",
  -"indicator_name",
  -"country_code"
  ) |>
  filter(
    country_name %in% countries
  ) |>
  t() |>
  row_to_names(row_number = 1) |>
  as.data.frame()

population_clean <- select(population,
  -"x1960":-"x2002",
  -"indicator_code",
  -"indicator_name",
  -"country_code"
  ) |>
  filter(
    country_name %in% countries
  ) |>
  t() |>
  row_to_names(row_number = 1) |>
  as.data.frame()
```

Above we are simply cleaning out and making usable data frames for later in the project. There are simply two frames we are currently interested in here. Is population_clean and

gdp_clean. We removed all years before 2003 as we are using this data together with the data from dataset one.

Join dataset one and two

Now we have a decent amount of dataframes from the two first datasets. in order to make

Visualzation

Until now we only have several dataframes and descriptive statistics from the datasets. In this section we will visualize the data to improve our understanding of what the data represent.
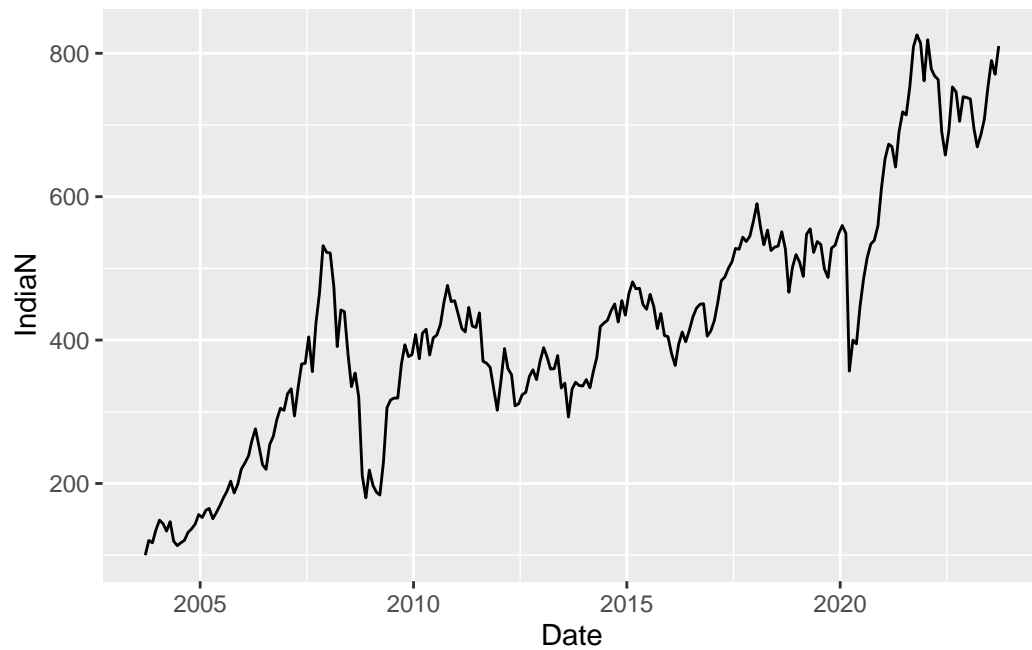
The first thing we want to get a better overveiw over is dataset one.

The normalized development of the FTSE Indicies can simply be graph it out doing the following:
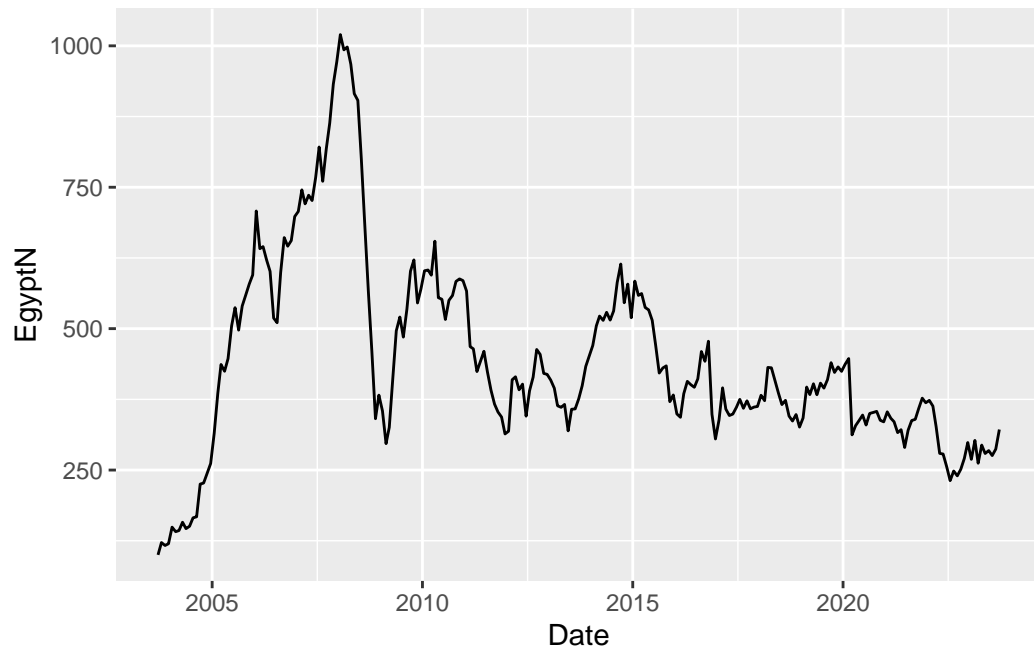
```
ggplot(
  data = normalized_trimmed,
  mapping = aes(x = Date, y = FTSEWorldN )
) +
  geom_line( mapping = aes())
```
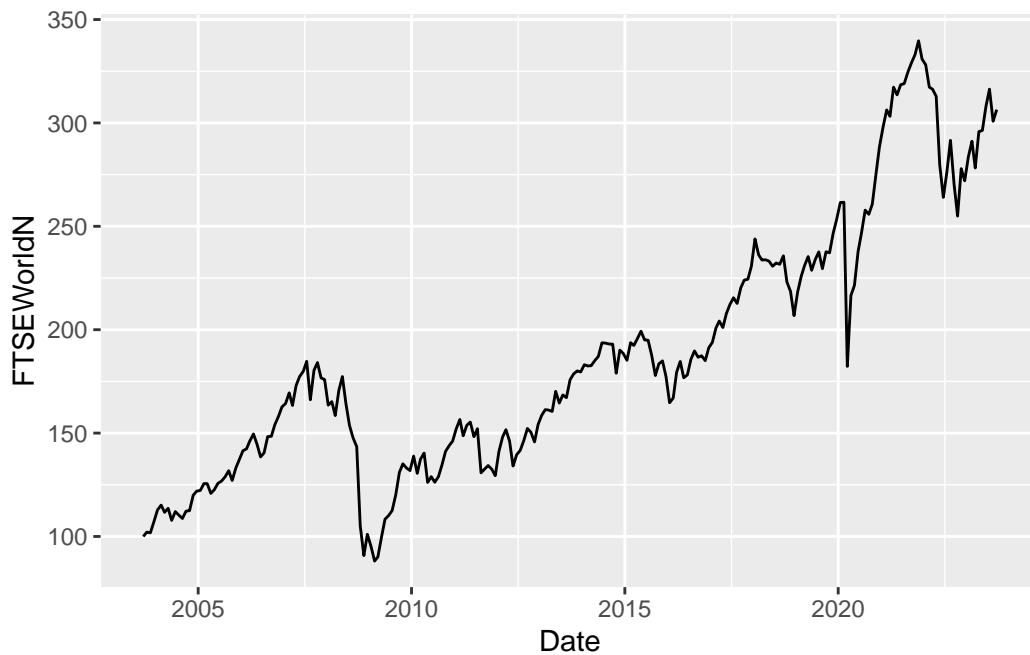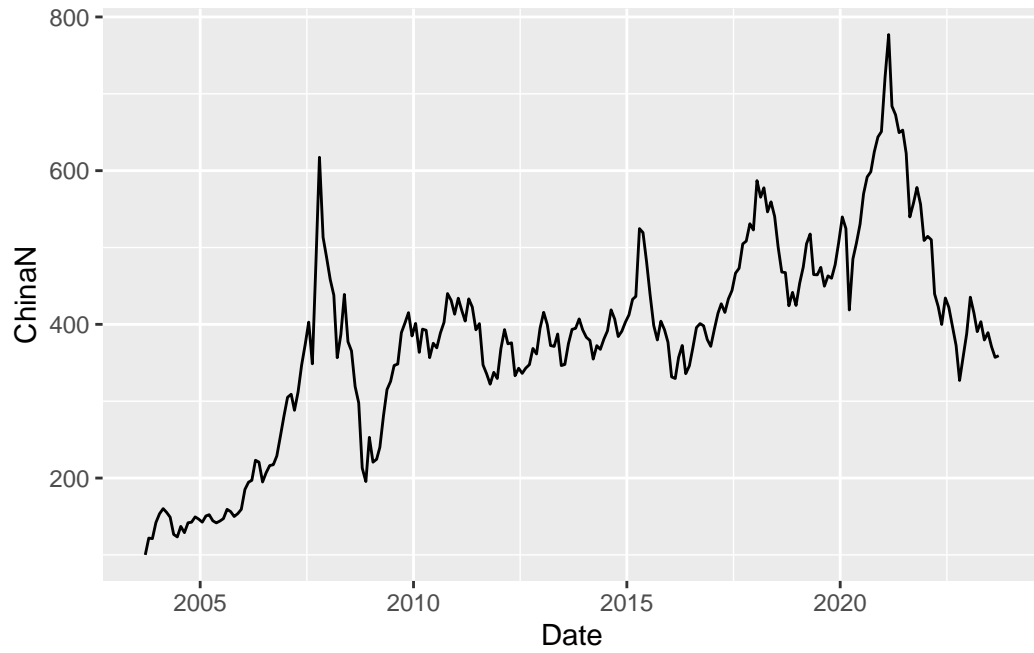
```
ggplot(
  data = normalized_trimmed,
  mapping = aes(x = Date, y = IndiaN )
) +
  geom_line( mapping = aes())
```
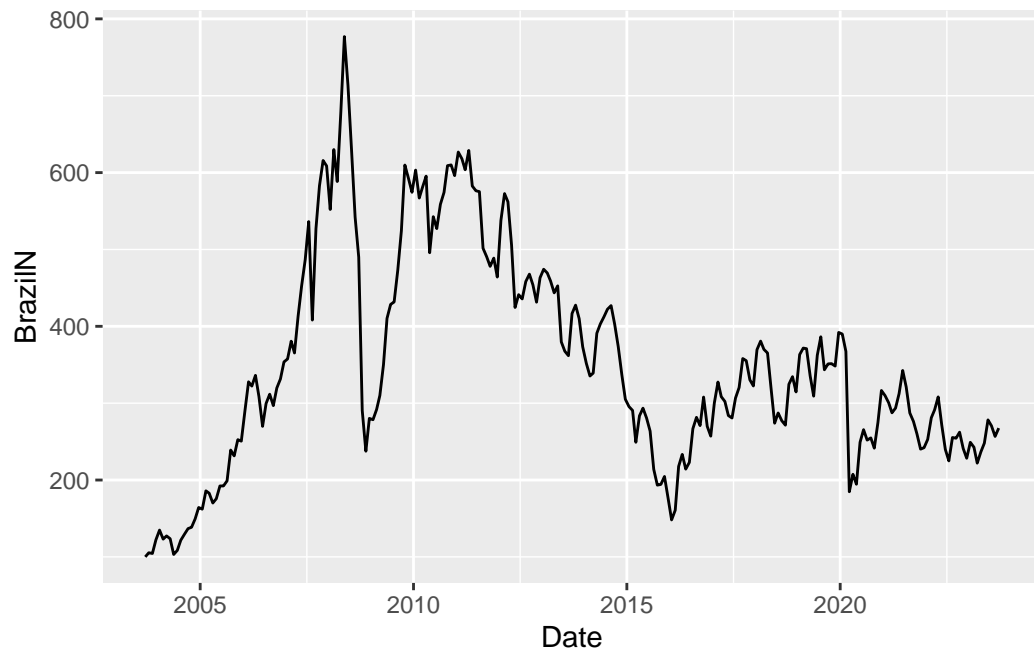


```
ggplot(
  data = normalized_trimmed,
  mapping = aes(x = Date, y = EgyptN )
) +
  geom_line( mapping = aes())
```

```
ggplot(
  data = normalized_trimmed,
  mapping = aes(x = Date, y = ChinaN )
) +
  geom_line( mapping = aes())
```

```
ggplot(
  data = normalized_trimmed,
  mapping = aes(x = Date, y = BrazilN )
) +
  geom_line( mapping = aes())
```

The returns