HMM Project

CHIRITA Andrei and LENNARTZ Jan

11/18/2020

Contents

Summary

In discussing the state of the stock exchanges around the world a lot of experts use the words “bull market” or “bear market”, these words represent broad generalizations in regard to the trajectory of the market in a given moment but they also point to the fact that there may exist market regimes that describe the trajectory of the stocks and that can be inferred from the market data. As these market regimes are “hidden” behind the daily market data (or more specific behind he returns) we can use a Hidden State Markov Model in order to find them and see how well they describe the data.

In the following project we aim to find these hidden market regimes using HMM models applied on two market indexes, the S&P500 and the IWM dataset. On the first step of the project we fit a HMM model on simulated data in order to understand how the model works in general terms, how can we fit, read and interpret it and how to use the associated R functions. In the second step of the project we get the data, calculate the daily returns based on the closing value of one day and the closing value of the previous day (which is also the opening value of the analyzed day) and then we fit the models. In trying to get results as good as possible we select models based on a range of criteria, first statistical criteria like AIC, BIC or if the HMM model reached convergence and second interpretability criteria. At the end we discuss what model is best taking all the criteria into account and find a way of generalizing the model.

Simulated data

In this part we will work on simulated data in order to illustrate how the statistical models work. We will generate data from a gaussian distributions representing “bullish” or “bearish” markets and test how the model is faring in estimating in which of the states the market it is at each time step.

Installing packages and preparing the simulation

In this stage we install the packages that will be used and set the main coordinates of the simulation. We will also set the seed to make sure that the experiment will be replicable.

```
library(depmix) # for fitting HMM functions
library(quantmod) # for real financial data
library(depmixS4) # for the depmix function
library(parallel) # for parralel computations
set.seed(1) # we set the seed
Nklower <- 50
Nkupper <- 150
bullmean <- 0.1
bullvar <- 0.1
bearmean <- -0.05
bearvar <- 0.2
```

Simulating the time periods of the Markov Chain

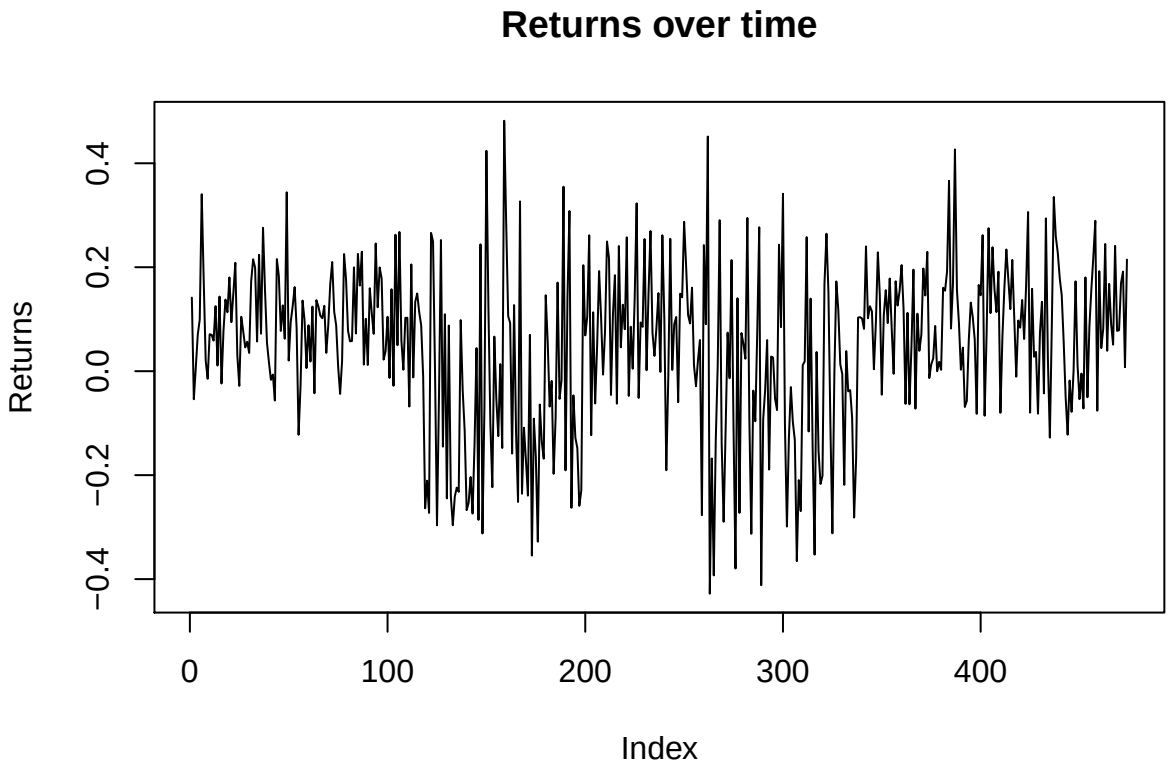
In this section we simulate the periods of the Markov Chain by putting together 5 different periods.

```
days <- replicate(5, sample(Nklower :Nkupper, 1)) # we create the daily observations

marketbull1 <- rnorm( days[1], bullmean, bullvar )
marketbear2 <- rnorm( days[2], bearmean, bearvar )
marketbull3 <- rnorm( days[3], bullmean, bullvar )
```

```
marketbear4 <- rnorm( days[4], bearmean, bearvar )
marketbull5 <- rnorm( days[5], bullmean, bullvar )
# we store the true regimes and the returns
trueregimes <-c(rep(1,days[1]), rep(2,days[2]), rep(1,days[3]),
                rep(2,days[4]), rep(1,days[5]))
returns <-c( marketbull1, marketbear2, marketbull3, marketbear4, marketbull5)
```

In the code above after simulating the five time periods we pasted them together in the returns variable. We also created a variable that represents the true states of the Markov Chain. The result was the following sequence of returns that resembles a plot of returns that exist on some real financial markets around the world:



Creating the Hidden Markov Model

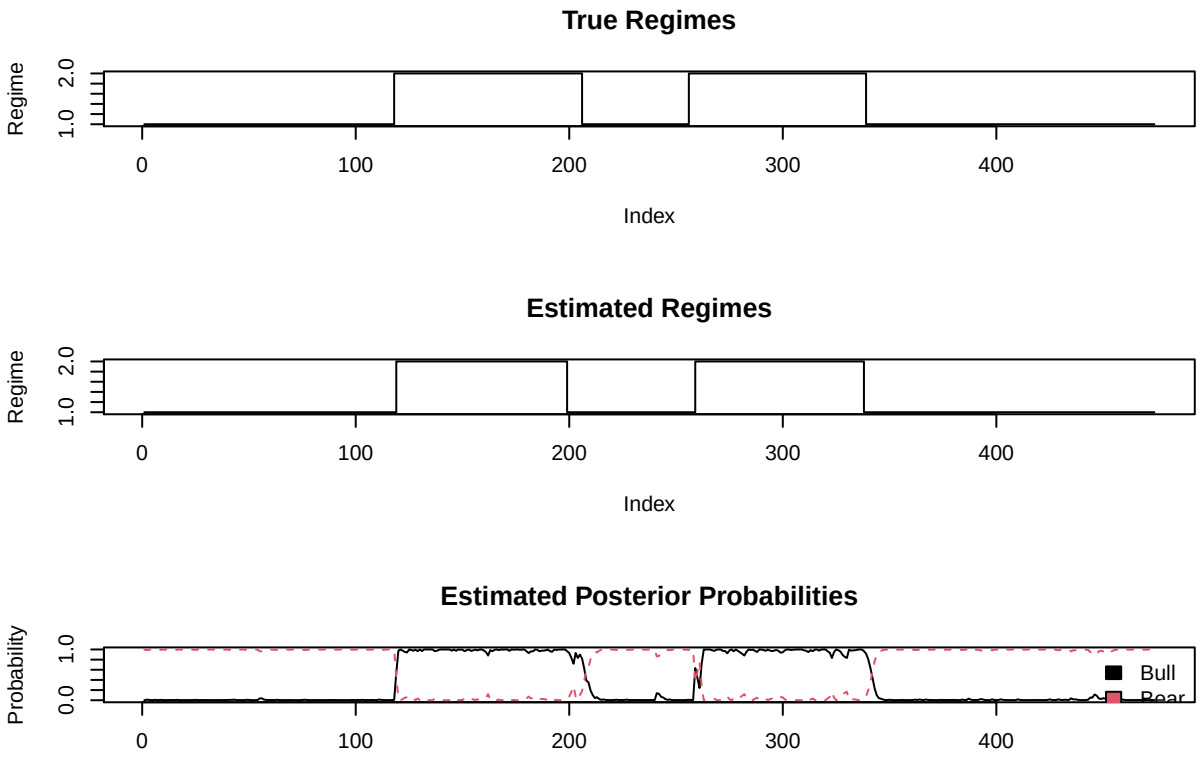
In this sub-section we will create and fit a HMM model, plot the posterior probabilities and compare them to the true states of the model.

```
hmm <- depmix(returns ~ 1, family = gaussian(), nstates = 2,
              data=data.frame(returns=returns)) # we create the HMM model
hmmfit <- fit(hmm, verbose = FALSE) # we fit the model
```

```
## converged at iteration 24 with logLik: 289.6389
```

```
postprobs <- posterior(hmmfit) # we compute the posterior probabilities
```

As can be seen from the output above the model converged after 22 iterations. The plot bellow shows that the model estimated quite well the state of the market. We must also say that the model is labeling the states in a different way than they were labeled in the "trueregimes" data so we had to re-label them.



In the end we will create a confusion matrix to see the percentage of observations wrongly classified by the model:

```
table(postprobs$state,trueregimes) # create a confusion matrix
```

```
##    trueregimes
##       1   2
##   1 303  12
##   2   0 159
```

The confusion matrix shows that most observations were classified correctly with only a few exceptions.

Real Data

The same procedure is now applied in a real data situation. The data in question is the S&P500 (GSPC) data and the IWM data set. Both data sets start from the year 2004. The S&P500 is a stock market index, representing the top 500 companies in the U.S. stock exchange. The IWM data contains stock exchange data

of a financial planning company (Intentional Wealth Management). The goal is to identify the underlying processes in order to tell when there is a change of the regime. Just like before it is assumed that there is a hidden markov chain present. Therefore, a Hidden Markov Model will be fitted and evaluated. It is not known how many states are present and there is no "ground truth", i.e. we can not tell if we detected the correct regime.

Obtaining the data

The data is retrieved with the *quantmod* library.

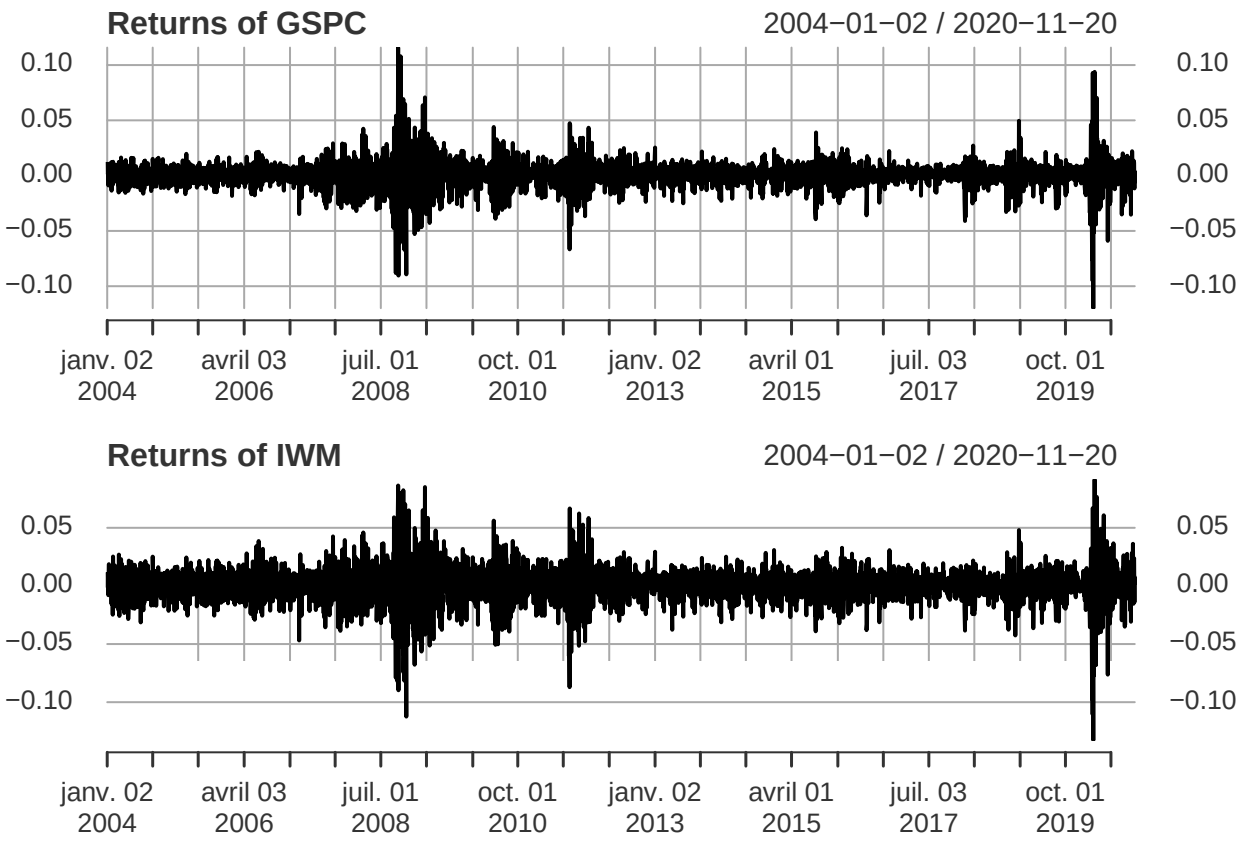
```
getSymbols("^GSPC", from="2004-01-01" )
getSymbols("IWM", from="2004-01-01" )
```

Returns

Before working with the data, the returns need to be computed because they are the observed variable we want to use to find the hidden state M. The returns can be calculated from the closing values of the stock exchange per day by the following formula: $\frac{Close_i - Close_{i-1}}{Close_{i-1}} = Return_i$. This means the return of the first day has to be skipped.

```
# GSPC Returns
Close_GSPC<-GSPC$GSPC.Close
Returns<-vector()
for( i in 2:length(Close_GSPC)){
  Returns[i]<-(as.numeric(Close_GSPC[i])-
                 as.numeric(Close_GSPC[i-1]))/as.numeric(Close_GSPC[i-1])
}
Close_GSPC$Returns<-Returns
```

If we visualize the returns we obtain the following plot:



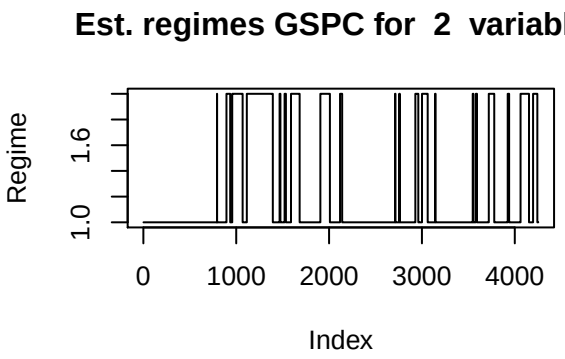
One can clearly see the financial crisis in 2008 and at the beginning of of 2020. Both data sets show the same patterns for the high peaks.

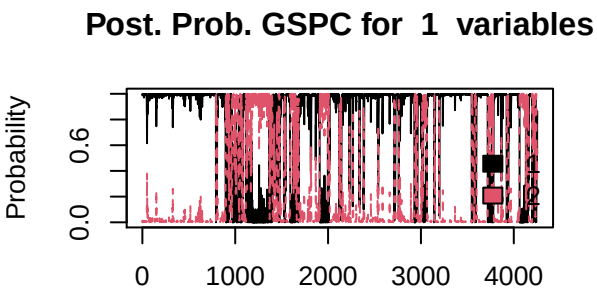
Fitting HMMs

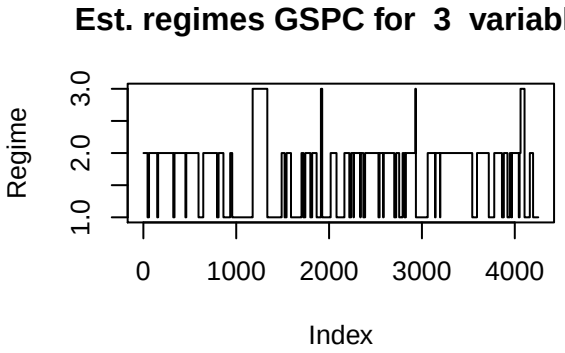
Now the EM-Algrithm will be used to fit different M-state Hidden Markov Models to each data set. We will consider only Markov Chains with two and three states. The following code was used for fitting the GSPC data Analogously, the IWM data was fitted.

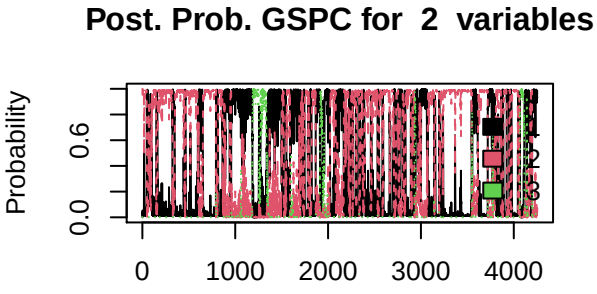
```
# GSPC
# Fit the models with 2 and 3 states
hmm_GSPC <- lapply(2:3,function(i){depmix(Returns ~ 1,
                                          family = gaussian(),
                                          nstates = i,
                                          data=data.frame(Close_GSPC$Returns))})
hmmfit_GSPC <- lapply(1:2,function(i){fit(hmm_GSPC[[i]], verbose = FALSE)})
postprobs_GSPC <- lapply(1:2,function(i){posterior(hmmfit_GSPC[[i]])})
```

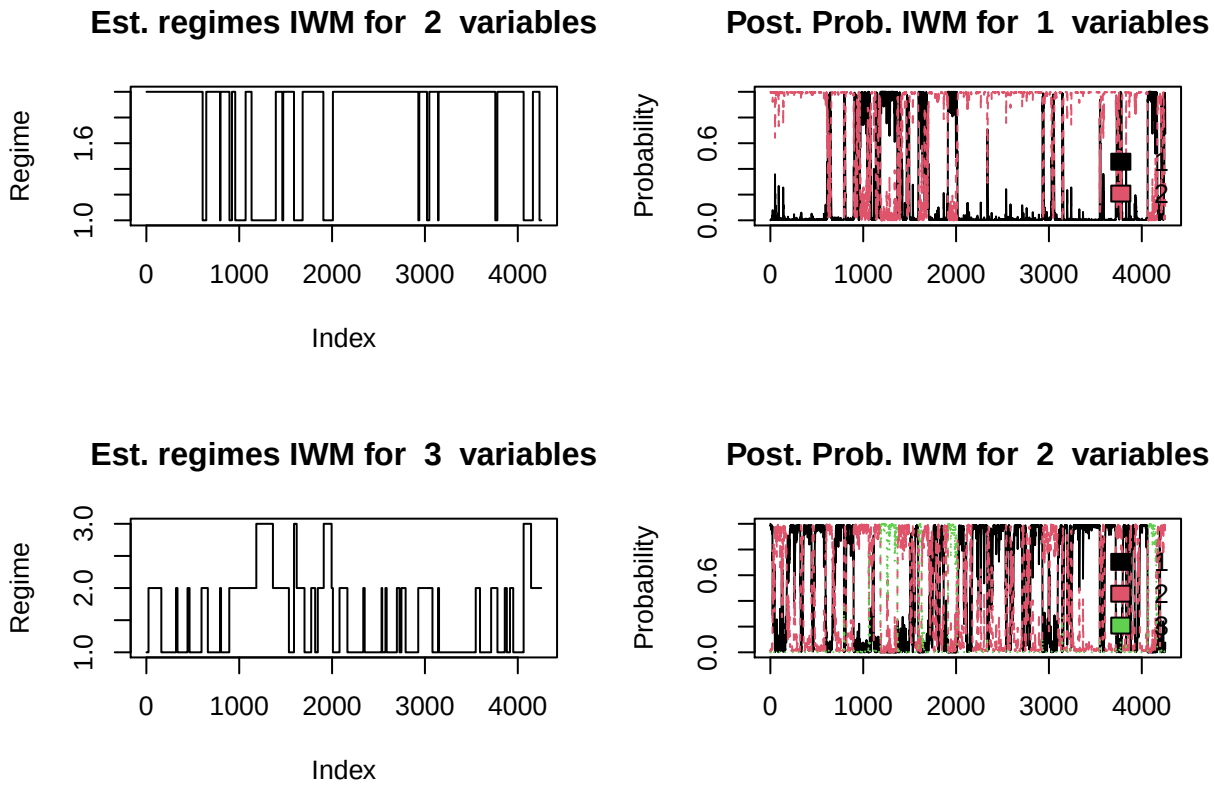
We can then plot the estimates hidden states and the posterior probabilities.


Est. regimes GSPC for 2 variables
Regime
1.0
1.6
0
1000
2000
3000
4000
Index


Post. Prob. GSPC for 1 variables
Probability
0.0
0.6
0
1000
2000
3000
4000


Est. regimes GSPC for 3 variables
Regime
1.0
2.0
3.0
0
1000
2000
3000
4000
Index


Post. Prob. GSPC for 2 variables
Probability
0.0
0.6
0
1000
2000
3000
4000



Selecting the best model

To find the best model we can look at the AIC or BIC. These are criteria which take the likelihood performance into account but penalize the number of parameters in the model. Thus, it is expected to select a model that is fitting to the data well, while it is not using too many parameters. The lower each of the criteria is the better is the model. The AIC and BIC can be obtained like this:

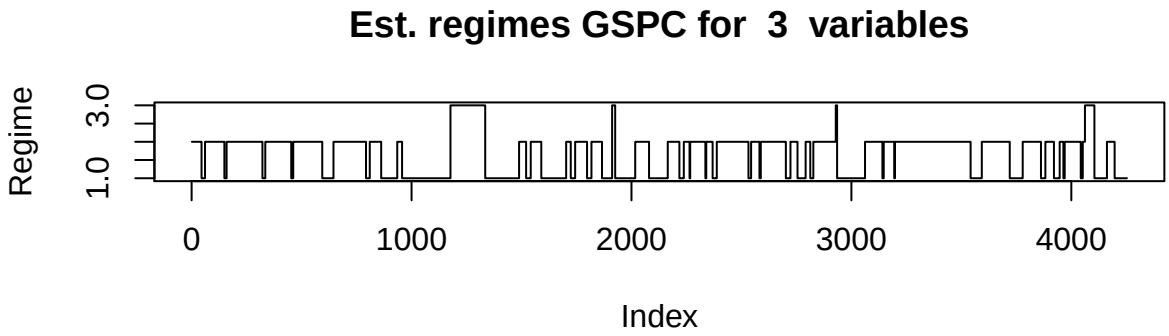
```
AIC_GSPC <- suppressWarnings(sapply(hmmfit_GSPC, AIC)) # To avoid warning messages
BIC_GSPC <- suppressWarnings(sapply(hmmfit_GSPC, BIC))
```

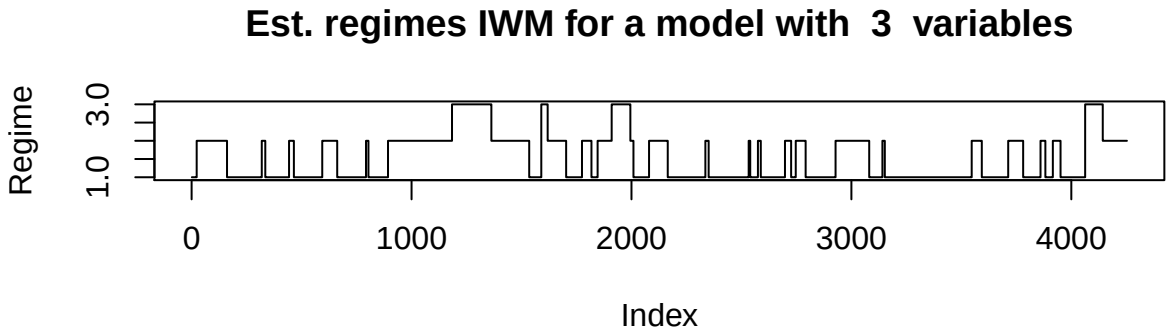
```
##           2-states  3-states
## AIC_GSPC -27696.02 -28129.05
## BIC_GSPC -27651.53 -28040.08
## AIC_IWM  -25056.98 -25355.70
## BIC_IWM  -25012.49 -25266.72
```

For both data sets the best model according to AIC and BIC is the 3 states model. However, the models are quite close. This means we could also consider each of them.

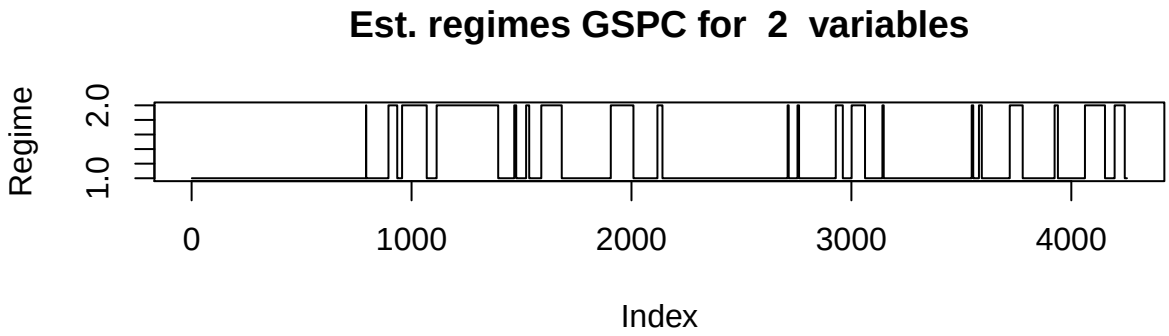
Interpreting the results of the model

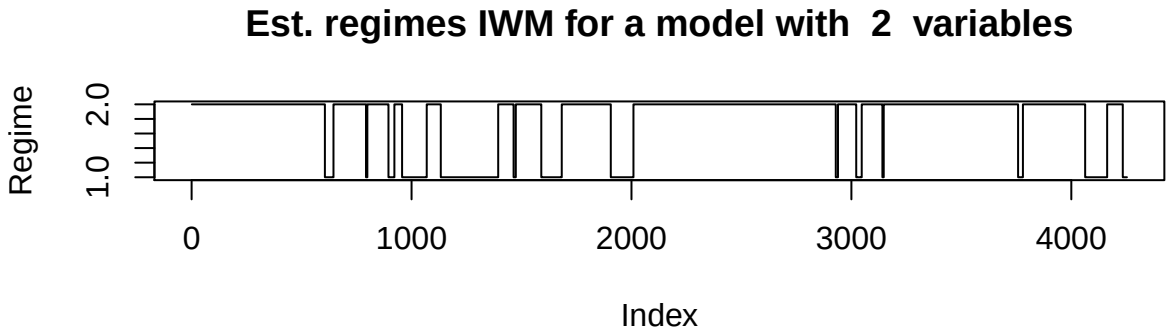
Based on the AIC and BIC criterions we decided to keep the model with three states but we will also analyze the models from the point of view of interpretability.





As can be seen from the plots the model estimates there are mainly two states with an intermediary third state happening very rarely, this may point out to the fact that in fact there are only two main market states with transition periods between them that combine some of the characteristics of both states. We must also point out to the fact that in the second part of the covered time period the model changes between states more often than in the first part which may be due to the higher market volatility of that period related to first the subprime crisis and then to the slow and sometimes shaky recovery that followed the 2008 crisis (one can recall the 2009-2010 euro crisis and the subsequent shocks related to the negotiations in regard to the fate of the debt of Greece). For the IWM dataset the three states are a little bit more balanced but it is clear that most observations have labels "2" and "3" with a smaller number of observations for the first label.





The plot above shows the predictions of the two states Hidden Markov Chain model. After a period of relatively calmness the model changes between states faster and with a higher frequency. After that there is another period of calmness related to the years around 2015 that can be explained by the improving of the economic conditions and the increase in speed of the global economy. The results are similar to both data sets (it must be noted that the labeling of the observations may differ between the two data sets, so in comparing them we looked if observations that are put under one label for one dataset are put under one label fo the other dataset notwithstanding the number assigned to the label).

For both data sets the two states HMM model captures better the variations of the market being better in spotting the periods of volatility (like the American subprime crisis of 2007-2009 or the post 2008 European debt crisis), while the three states model captures some of those effects and has lower AIC and BIC indicators the difference of these two indicators between the two states model and the three states model is extremely low. Thus, taking into account the AIC, BIC and the interpretability of the models, we would recommend using the two states Hidden Markov Models. As results for both data sets are similar we think that two states Hidden Markov Models are better for both data sets with the remark that the three states model for the IWM data is somewhat more interpretable than its GSPC counterpart.

Generalization

The chosen models are both based on three hidden states in the market. One could assume that this will be similar for other stock exchange data sets. In general it is not trivial, often practically impossible to fit a model with more states in these cases. Three states seem to explain the data quite well. However, this is only based on two data sets. To get a better overview one would have to inspect more stock exchange data sets in order to get an impression on how well it performs on them. For some data sets a two state model might still be preferable.