STOCK MARKET STATE CLASSIFICATION AND PREDICTION USING MARKOV CHAIN

Report submitted to

Indian Institute of Technology, Kharagpur

for the award of the degree

of

Master of Technology

in Financial Engineering

by

Soumy Ladha

DEPARTMENT OF AEROSPACE/FINANCIAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY, KHARAGPUR

May 2018

DECLARATION

I certify that

a. The work contained in this report is original and has been done by me under the guidance of my supervisor.

b. The work has not been submitted to any other Institute for any degree or diploma.

c. I have followed the guidelines provided by the Institute in preparing the report.

d. I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.

Whenever I have used materials (data, theoretical analysis, figures, and text) from other sources, I have given due credit to them by citing them in the text of the report and giving their details in the references. Further, I have taken permission from the copyright owners of the sources, whenever necessary.

CERTIFICATE

This is to certify that the project titled "**Stock Market State Classification and Prediction using Markov Chain**" submitted in partial fulfilment of the requirement for the award of degree of Master of Technology in Financial Engineering, is an authentic record of the work carried out by Mr. Soumy Ladha (Roll no. 13AE3FP09) under the supervision and guidance of Professor Somesh Kumar

Soumy Ladha Date:
(13AE3FP09)

It is certified that the above statement by the candidate is correct to the best of my knowledge and belief.

....
Professor Somesh Kumar Date:
Department of Mathematics
IIT Kharagpur

ABSTRACT

Modeling non-stationarity is a major setback in traditional model building thus making the analysis of financial system complex. In this, we analyze such systems using contemporary approach such as Markov chains. We use two different methods for identifying the states of the stock market. Thus by examining the transitions in states we try to inspect instances of drastic change in it. With a multitude of states, we build a Markov chain model to forecast the likelihood of next plausible state. With this, we develop a method for understanding market in a way which is simple and coherent.

ACKNOWLEDGEMENT

I would extend my sincerest gratitude to Prof. Somesh Kumar for his consistent support and clear guidance throughout this phase in this semester at IIT Kharagpur. At last I would like to thank my friends and family for supporting emotionally through difficult times during the accomplishment of this project.

CONTENTS

Chapter 1 Introduction

Time series is one of the way to analyse complex systems. The correlation has been a powerful tool in the analysis of time series. Time series analysis is done on stationary data yet most natural processes are non-stationary. In non-stationary situations analysis becomes arduous and model robustness is lost. To overcome these challenges we need superior models which are less vulnerable to non-stationarity of data. To achieve this we endeavour to define states through which systems pass and in which they remain for short times. Success in this respect would allow to get a better discretion of the system and might even lead to methods for controlling the system in more efficient ways.

Chapter 2 Literature Review

Title of the paper	**Author/Source**	**Description**
Stock Market Prediction Using Hidden Markov Models	Aditya Gupta and Bhuwan Dhingra	This article presents the Maximum a Posteriori HMM approach for forecasting stock values for the next day given historical data.
Application of Markov Chains to Stock Trends	Kevin J. Doubleday and Julius N. Esunge	This article analyses Dow Jones Industrial Average using a discrete time stochastic model, (Markov Chain). Two models are build, where the DJIA was considered as being in a state of (1) gain or loss and (2) small, moderate, or large gain or loss.
Predicting Stock Prices	Shuchi S. Mitra and Michael J. Riggieri	In this article author uses markov chain to predict the immediate future stock prices for a given company. They applied Markov Chain calculations to the data to create a 4x4 transitional probability matrix. Using this transition matrix they found four steady states which represented the probability that a stock price for a given day would fall into one of the four states.

Chapter 3 Background Study

Markov Chain

A Markov chain is a mathematical model of a random phenomenon evolving with time in a way that the past affects the future only through the present. The “time” can be discrete (i.e. the integers), continuous (i.e. the real numbers), or, more generally, a totally ordered set. A discrete-time Markov chain is a sequence of random variables X_1, X_2, X_3 ... with the Markov property that the probability of moving to the next state depends only on the present state and not on the previous states

<u>Definition 1.1</u>: A stochastic process is defined to be an indexed collection of random variables $\{X_t\}$, where the index t runs through a given set T. The variable X_t is meant to represent a measurable characteristic, or point of interest. Thus, the stochastic process provides a mathematical representation of how the status of the physical system evolves over time.

Example of markov chain: A mouse is in a cage with two cells, 1 and 2. A mouse lives in the cage. We record position of mouse every minute. Thus the movement of mouse from one cage to another can be thought as movement from one state to another.

<u>Definition 1.2</u>: A stochastic process $\{X_t\}$ is said to have the Markovian property if $P\{X_{t+1}=j|\ X_0 = k_0, X_1 = k_1,\ldots,X_{t-1} = k_{t-1}, X_t = i\} = P\{X_{t+1}=j|\ X_t = i\}$, for t = 0,1,2… and every sequence i, j, k_0, k_1,…, k_{t-1}. This is saying that the probability of X_{t+1} being equal to j is solely dependent upon the preceding event of what X_t equals.

Transition Probabilities

Conditional probabilities for Markov Chains are called transition probabilities.

<u>Definition 1.3</u>: If Conditional probabilities are defined as

$$P_{ij} = P\{X_{t+1} = j|X_t = i\} \tag{1}$$

Markov Chain can also have n-step transition probabilities, which is the conditional probability that the process will be in state j after n-steps provided that it starts in state i at time t.

<u>Definition 1.4</u>: N-step transition probabilities are defined as the conditional probability

$$P\{X_{t+n} = j\ |X_t = i\} = P\ \{X_n = j|X_0 = i\} \text{ for all } t = 0, 1\ \ldots \tag{2}$$

Therefore, a Markov Chain is a stochastic process that states that the conditional probability of a future event relies on the present state of the process, rather than any past states, or events

Chapman-Kolmogorov Equations

Chapman-Kolmogorov Equations are used to provide a method to compute all of the n-step transition probabilities:

$$p_{ij}^{(n)} = \sum_{k=0}^{m} p_{ik}^{(m)} \cdot p_{kj}^{(n-m)} \tag{3}$$

These equations are used to point out that when we go from one state to another in n steps, the process will be in some other state after exactly m (m is less than n) states. Thus the summation is just the conditional probability that, given a starting point in one state, the process goes to the other state after m steps and then to the next state in n –m steps. Therefore, by summing up these conditional probabilities over all the possible steady states must yield. These expressions allow us to obtain the n-step probabilities from the one-step transition probabilities recursively.

Transition Matrix

The conditional probabilities for a stochastic process can be organized into an n-step transition matrix. Such a matrix is of the form

$$P = \begin{pmatrix} P_{11} & \cdots & P_{1n} \\ \vdots & \ddots & \vdots \\ P_{n1} & \cdots & P_{nn} \end{pmatrix} \tag{4}$$

A transition matrix shows the transition probability in a particular column and row as the transition from the row state to the column state. Transition probability is comprised of conditional probability and hence each row of a transition matrix must also sum to the value 1 since each row signifies a state of the overall stochastic process, and each entry within each row is a conditional probability for the process to be in that state.

State Transition Diagrams

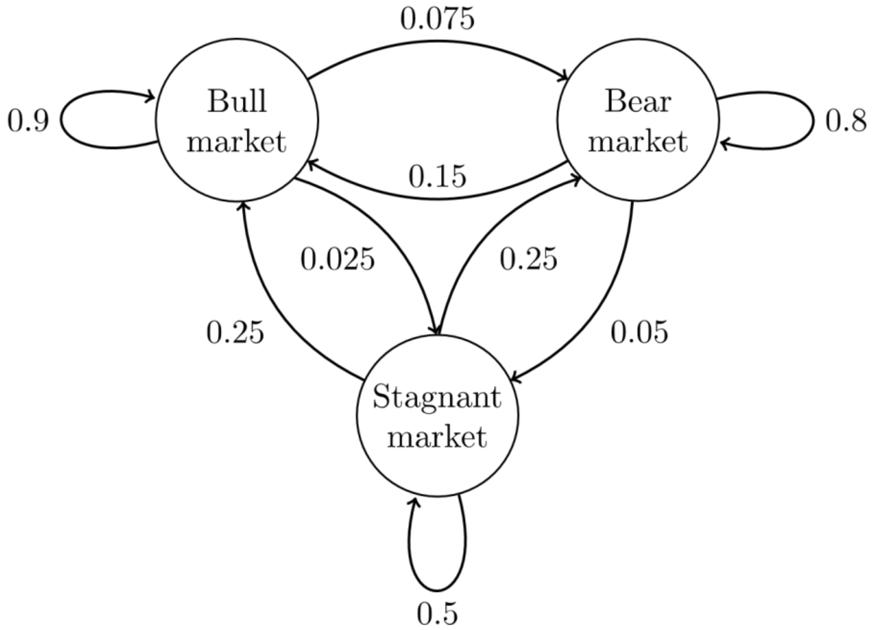

Figure 3-1 Representing State Transition Diagram

A convenient and useful method to visualize the state of Markov Chains when they have stationary transition probabilities and a finite number of states is through the use of a state transition diagram. In such diagram, each state of a Markov chain is drawn as a numbered node, and the conditional probability of moving from one state to another is drawn by connecting the nodes with an edge and labelling the edge with the numbered probability.

Categorizing States of Markov Chains

Since Markov chains are long run stochastic processes that include transitional probabilities which indicate the likelihood the process will move from one state to another, it is often necessary to categorize, or classify, the varying types of states.

Definition 1.5: A state k is said to be accessible from a state j if $(n) > 0$ for some $n \geq 0$, or simply stated, the system can eventually move from state j to state k.

Definition 1.6: If a state j is accessible from a state k and k is accessible from state j then states j and k are said to communicate with one another.

In a Markov chain, every state communicates with itself, since $P_{jj}(0) = \{X_{0=j} | X_{0=j}\} = 1$. If a state j communicates with k and k communicates with l then state j communicates with state l since different states can communicate with one another within the same system, Markov Chains can be placed into classes, which are groupings of states that only communicate with one another. If every state of a Markov chain communicates with every other state within the chain, that is if the entire Markov chain is in itself one class, then the chain is said to be irreducible.

Transient States

Definition 1.7: A state i is said to be transient if, given that we start in state i, there is a non-zero probability that we will never return to i.

Recurrent States and Absorbing State

Definition 1.7: If a stochastic process, such as a Markov chain, enters a state, and will definitely return to it, the state is said to be recurrent. Hence, recurrent states cannot be transient; however, they can be absorbing.

Definition 1.8: A state is considered to be absorbent, if after entering the state, the process will never leave the state. If for example, the state j is an absorbing state, then $P_{jj} = 1$. From the above definitions, it is apparent that when grouping the states of a Markov chain into classes, each state belonging to a class is either transient or recurrent. For an irreducible finite-state Markov chain, every state is recurrent, and for any finite-state Markov chain, all the states cannot be transient.

Periodicity and Ergodicity

Definition 1.9: A state i has period k if any return to state i must occur in multiples of k time steps. The period of a state is defined as

$$k = gcd(n > 0: Pr(X_n = i|X_0 = i) > 0) \quad (5)$$

If k = 1, then the state is said to be aperiodic. Otherwise (k > 1), the state is said to be periodic with period k. A Markov chain is aperiodic if every state is aperiodic. An irreducible Markov chain only needs one aperiodic state to imply all states are aperiodic. Every state of a bipartite graph has an even period.

Definition 1.10: A state i is said to be ergodic if it is aperiodic and positive recurrent. In other words, a state i is ergodic if it is recurrent, has a period of *1*, and has finite mean recurrence time. If all states in an irreducible Markov chain are ergodic, then the chain is said to be ergodic.

Properties of Markov Chains in the Long Run

Steady State Probabilities

After the n-step transition probabilities for a Markov chain have been calculated, the Markov chain will display the characteristic of a steady state. Meaning, that if the value of n is large enough, every row of the matrix will be the same, and such, the probability that the process is in each state does not depend on the initial state of the process. Therefore, the probability that the process will be in each state k after a certain number of transitions is a limiting probability that exists independently of the initial state. This can be defined as:

For any irreducible ergodic Markov chain, $lim_{n\to\infty} P_{ij}\ (n)$ exists and is independent of i.

$$\lim_{n\to\infty} P_{ij}\ (n) = \pi_j > 0 \quad (6)$$

Where the π_j uniquely satisfy the following steady-state equations

$$\pi_j^{(n)} = \sum_{k=0}^{m} \pi_j p_{ij} \qquad \text{For j = 0, 1, 2…m} \quad (7)$$

$$\sum_{k=0}^{m} \pi_j = 1$$

The steady state probabilities of the Markov chain are π_j. These values indicate that after a large number of transitions the probability of finding the process in a particular state such as j tends to the value of π_j which is independent of the initial state. The π_j are also known as stationary probabilities, when if the initial probability of being in state j is given by π_j for all j, then the probability of finding the process in state j at time n=1,2,… is also given by π_j, or $P\{X_n = j\} = \pi_j$

Clustering

Classifying unlabelled data into similar groups is called clustering. In a cluster data items are similar but dissimilar from another cluster.

K-Means clustering

K-means clustering is a clustering algorithm, which is used when you have unlabelled data (i.e., data without defined categories or groups). In this algorithm we classify data into K-clusters. This is an iterative algorithm which in every step classify data to one of the nearest clusters. Data points are clustered based on feature similarity. The results of the K-means clustering algorithm are:

- The centroids of the K clusters, which can be used to label new data.
- Labels for the training data (each data point is assigned to a single cluster).

Each centroid of a cluster is a collection of feature values which define the resulting groups. By examining cluster centroid we can qualitatively understand the properties of a group.

Algorithm

The K-means clustering algorithm uses iterative refinement to produce a final result. The algorithm inputs are the number of clusters K and the data set. The data set is a collection of features for each data point. The algorithms starts with initial estimates for the K centroids, which can either be randomly generated or randomly selected from the data set. The algorithm then iterates between two steps:

1. Data assignment step: Each centroid defines one of the clusters. In this step, each data point is assigned to its nearest centroid, based on the squared Euclidean distance. More formally, if ci is the collection of centroids in set C, then each data point x is assigned to a cluster based on

$$\underset{c_i \in C}{argmin}\ dist(c_i, x)^2 \tag{8}$$

2. Centroid update step: In this step, the centroids are recomputed. This is done by taking the mean of all data points assigned to that centroid's cluster.

$$c_i = \frac{1}{S_i} \sum_{x_i \in S_i} x_i \tag{9}$$

The algorithm iterates between steps one and two until a stopping criteria is met (i.e., no data points change clusters, the sum of the distances is minimized, or some maximum number of iterations is reached).

This algorithm is guaranteed to converge to a result. The result may be a local optimum (i.e. not necessarily the best possible outcome), meaning that assessing more than one run of the algorithm with randomized starting centroids may give a better outcome.

Choosing K

The algorithm described above finds the clusters and data set labels for a particular pre-chosen K. To find the number of clusters in the data, the user needs to run the K-means clustering algorithm for a range of K values and compare the results. In general, there is no method for determining exact value of K, but an accurate estimate can be obtained using the following techniques.

One of the metrics that is commonly used to compare results across different values of K is the mean distance between data points and their cluster centroid. Since increasing the number of clusters will always reduce the distance to data points, increasing K will always decrease this metric, to the extreme of reaching zero when K is the same as the number of data points. Thus, this metric cannot be used as the sole target. Instead, mean distance to the centroid as a function of K is plotted and the "elbow point," where the rate of decrease sharply shifts, can be used to roughly determine K.

A number of other techniques exist for validating K, including cross-validation, information criteria, the information theoretic jump method, the silhouette method, and the G-means algorithm. In addition, monitoring the distribution of data points across groups provides insight into how the algorithm is splitting the data for each K.

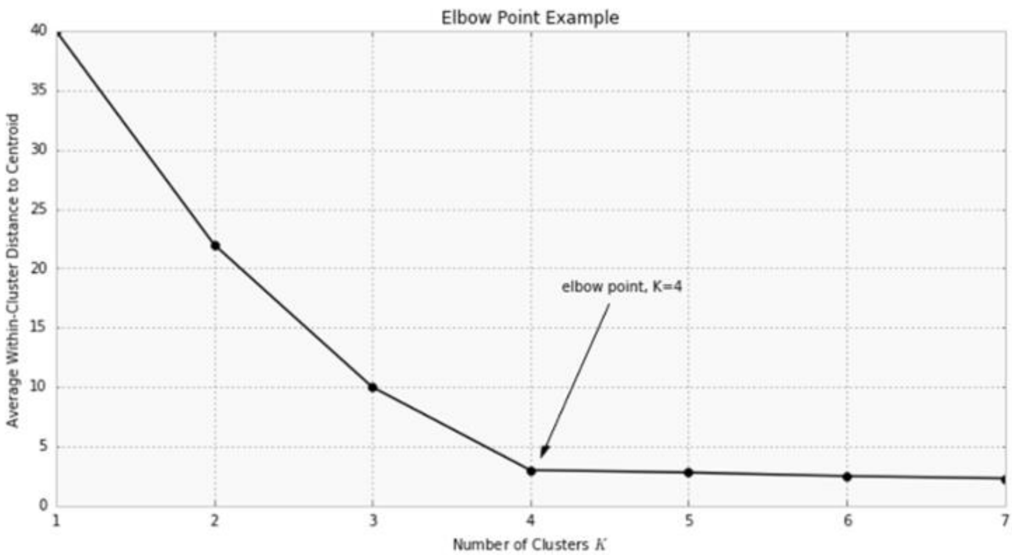

Figure 3-2 Image explains what cluster number to choose

Maximum Likelihood estimation

Given a distribution for data, the next step is to fit the model to the data. Typical probability distributions will have unknown parameters, numbers that change the shape of the distribution. The technical term for the procedure of finding the values of the unknown parameters of a probability distribution from data is estimation. During estimation we seek to find parameters that make the model fit the data. The objective function defines model fitness to data and we try to find estimators for parameters that maximize (or minimize) an objective function.

In the case of the Gaussian distribution, these parameters are called the "mean" and "standard deviation" written as mu and sigma.

Maximum Likelihood is most commonly used objective function, and the most well-understood estimation procedures. The maximum likelihood estimates are often referred to as MLEs.

The likelihood is defined as a conditional probability: P(data | model), the probability of the data given the model. Typically, the only part of the model that can change are the parameters, so the likelihood is often written as $P(X|\theta)$ where X is a data matrix, and θ is a vector containing all the parameters of the distribution. This notation makes explicit that the likelihood depends on both the data and a choice of any free parameters in the model.

We have some i.i.d., observations from a pool, say X_1, X_2 … X_n, which we refer to as a vector X. We write down the likelihood, L, which is the conditional probability of the data given the model. In the case of independent observations, we use the joint probability rule to write likelihood as:

$$L = P(X|\theta) = P(X_1|\theta)P(X_2|\theta) \dots P(X_n|\theta) = \prod_{i=1}^{n} P(X_i|\theta) \tag{10}$$

The likelihood for Gaussian data

We assume that each observation is described by the Gaussian distribution, we have two parameters, the mean, μ and standard deviation σ.

$$L = \prod_{i=1}^{n} P(X_i|\theta) = \prod_{i=1}^{n} N(X_i|\mu,\sigma) = \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(X_i-\mu)^2}{2\sigma^2}} \tag{11}$$

We derive the MLEs for the univariate Gaussian likelihood introduced above. After writing down the likelihood function, the next step is to find the maximum of this function by taking the derivatives with respect to the parameters, setting them equal to zero and solving for the maximum likelihood estimators. We take logarithm of the likelihood function to estimate parameters since it is relatively easy to solve log-likelihood in comparison to likelihood. We take logarithm because the logarithm is monotonic (it doesn't change the ranks of numbers) the maximum of the log-likelihood is also the maximum of the likelihood.

$$\begin{aligned} logL = log \prod_{i=1}^{n} N(X_i|\mu,\sigma) &= \sum_{i=1}^{n} \log N(X_i|\mu,\sigma) \\ &= \sum_{i=1}^{n} -\log\sigma - \frac{1}{2}\log(2\pi) - \frac{(X_i-\mu)^2}{2\sigma^2} \end{aligned} \tag{12}$$

To find the maximum likelihood estimate of the mean, μ, we will take derivatives with respect to μ. Using the linearity of the derivative operator, we have

$$\frac{\partial}{\partial\mu} logL = \frac{\partial}{\partial\mu}\sum_{i=1}^{n} -\log\sigma - \frac{1}{2}\log(2\pi) - \frac{(X_i-\mu)^2}{2\sigma^2}$$
$$= \sum_{i=1}^{n} -\frac{\partial}{\partial\mu}\log\sigma - \frac{1}{2}\frac{\partial}{\partial\mu}\log(2\pi) - \frac{\partial}{\partial\mu}\frac{(X_i-\mu)^2}{2\sigma^2} = 0 \tag{13}$$

Since two of the terms have no dependence on μ, their derivatives are simply zero. Taking the derivatives we get

$$\frac{\partial}{\partial\mu} logL = \sum_{i=1}^{n} -0 - 0 + \frac{2(X_i-\mu)}{2\sigma^2} = \frac{1}{\sigma^2}\sum_{i=1}^{n}(X_i-\mu) = 0, \mu = \mu_{MLE} = \frac{1}{n}\sum_{i=1}^{n} X_i$$
$$= m_X \tag{14}$$

A similar derivation is also possible for the standard deviation.

$$\frac{\partial}{\partial\sigma} logL = 0 \rightarrow \sigma_{MLE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(X_i-\mu)^2} = s_X \tag{15}$$

In general, setting the derivatives of the likelihood with respect to all the parameters to zero leads to a set of equations with as many equations and unknowns as the number of parameters and solving them will get us MLE estimates.

Chapter 4 Methodology

Dataset

The dataset for method 1 consisted of daily closing values of Sensex for the period of 2000 to 2017. The dataset for method 2 consisted of daily closing prices of following indices as well as it constituents (Stocks) for period of 2002 to 2017. Both of datasets were downloaded using Bloomberg terminal.

- Auto
- Bankex
- Capital Goods
- Consumer Durables
- Fast moving consumer goods (FMCG)
- Health Care
- Information Technology (IT)
- Metal
- Oil and Gas
- Public Sector Units
- Sensex

State Estimation

Method 1

As previously described, the objective is to identify states in the stock market and after identifying those states build a Markov chain model to predict the likelihood of next state.

Once the dataset is amassed, the first step is making a moving window having Sensex data of last 50 days. Since we needed a future forecast of the stock prices, so we build an Auto-Regressive Integrated Moving Average (ARIMA) model on the 50 days data. Then forecast the value for next day i.e. 51st day. Using that forecasted value and actual value we calculate the error corresponding to the prediction. Using moving windows we generate an array of daily error. Then we perform adf-test, in which we found data to be stationary.

```
        Augmented Dickey-Fuller Test

data:  error
Dickey-Fuller = -15.641, Lag order = 16, p-value = 0.01
alternative hypothesis: stationary

Warning message:
In adf.test (error): p-value smaller than printed p-value
```

Finally, we bin the data into 4 categories (states). Binning is done by observing histogram and removing outliers. The method of binning is explained below:

- Plot the histogram and box-plot to detect any outliers thus remove them from the analysis.
- Calculate minimum and maximum of the error array.
- Calculate the difference between Maximum and Minimum and divide it by 8, let's say this value be 'α' and also calculate the average of maximum and minimum and let this value be 'β'.
- Generate 4 intervals as follows [Minimum, 'β'-'α'), ['β'-'α', 'β'), ['β', 'β'+'α'], ('β'+'α', Maximum].
- Each of these intervals corresponds to a state, example if an error in particular day lies between [Minimum, 'β'-' α') then it in state '1' likewise to '2', '3' and '4' thus we binned the data in 4 states.

State '1' corresponds to the market being Very Low.
State '2' corresponds to the market being Low.
State '3' corresponds to the market being High.
State '4' corresponds to the market being Very high.

With above steps, we have developed a method to classify data in 4 states.

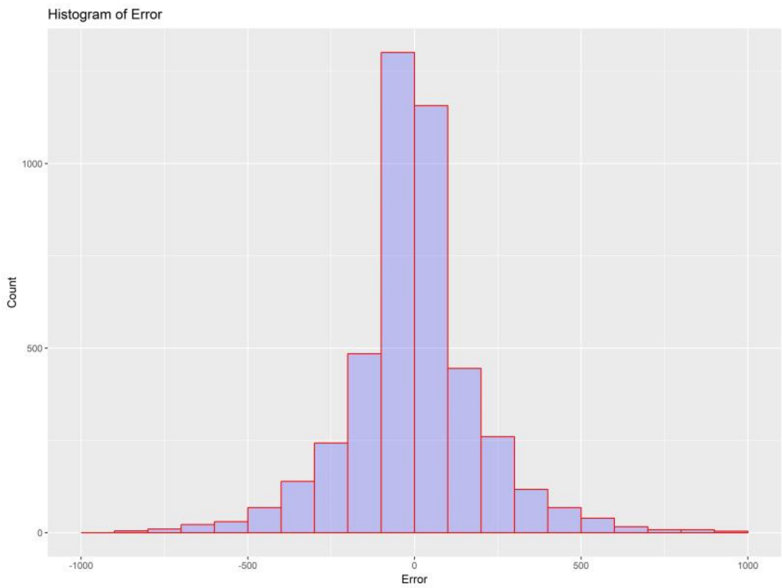

Figure 4-1 Histogram of error

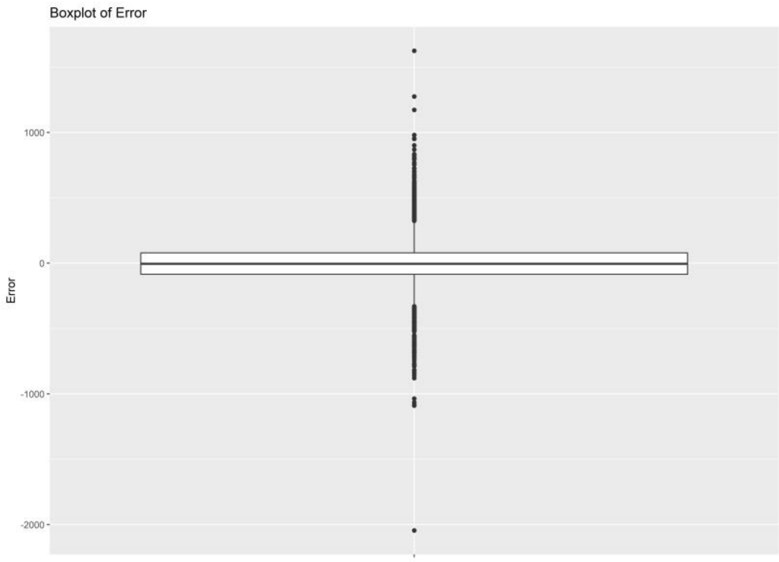

Figure 4-2 Box plot of error

Method 2

In this, we try to find states using an approach which contrasts with conventional time series analysis. Here we use indices data as they can play a major role in the estimation of financial risk. We calculate daily logarithmic return corresponding to these indices. Now using a moving window of 20 days we estimated correlation matrix among 11 variables thus we have an array of correlation matrix with the number of entries being (number of days in 2002-2017) minus 20.

Heatmap is generated using steps below:-

- Using two loops calculate norm of the difference between every element in an array
- Divide that norm by the number of columns in the correlation matrix
- We have the matrix in which 'i_{th}' row and 'j_{th}' column is the norm of the difference between 'i_{th}' element of the array and 'j_{th}' element of the array.
- Using 'Heatmap' command in 'R' generate the heatmap

Then we applied k-NN clustering algorithm. The steps of clustering are explained below:

- k-NN clustering can directly be applied using "kmeans" command in 'R' but we need to determine optimal number of clusters.
- Using loop iterate number of the cluster from 1-40, calculate error on each cluster values and plot them.
- Using elbow method as explained choose the most appropriate number of clusters i.e. 10. For further analysis set the appropriate number of clusters as 10

- Obtain cluster centers using the command in ‘R’, then classify the elements of correlation matrix array to appropriate clusters.
- We have an array in which every element is an indicator variable taking one value from 1 to 10 which denotes the state of the market at a particular time.

The same process is repeated for the collection of stocks that are used to create sectorial indices

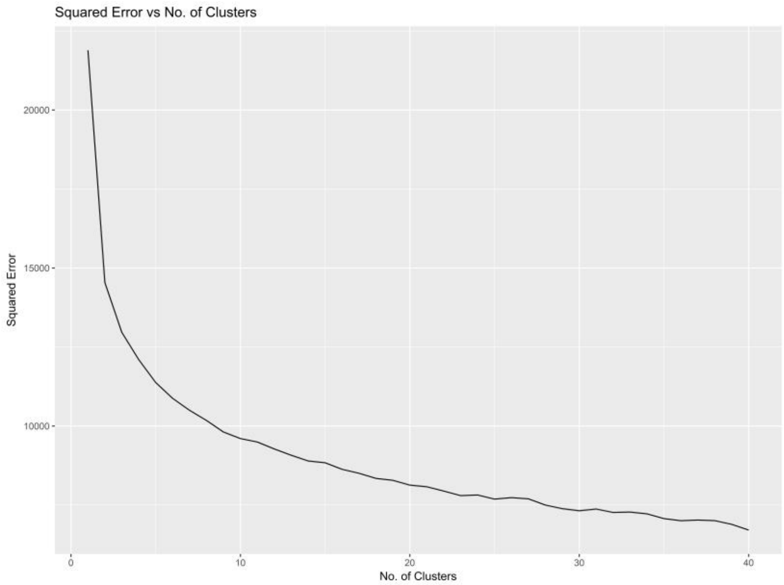

Figure 4-3 Squared error vs. Number of clusters

Markov Chain Estimation

In this, we will estimate transition probability matrix from a given sequence of states. Let us assume that it is sufficiently long sequence containing ‘n’ distinct states. The steps for estimating transition matrix as stated below:

- Since we have ‘n’ states and hence we will have ‘nxn’ transition matrix.
- In this method, we will perform likelihood maximization to get estimates for transition matrix
- For a given sequence replace every transition between states to its assumed probability e.g. Let the sequence be AABBAB, let the transition matrix be

$$\begin{pmatrix} p_{AA} & p_{AB} \\ p_{BA} & p_{BB} \end{pmatrix}$$

- For this sequence the likelihood will $p_{AA} * p_{AB} * p_{BB} * p_{BA} * p_{AB}$.
- Differentiate with respect to each unknown variables and solve the given set of equations with additional equation be (sum of probability in a row is ‘1’)
- We used this estimation method to estimate transition matrix for both State estimation method for ‘part 1’ and ‘part 2’

Chapter 5 Results

The result of estimation of transition matrix and steady state matrix are as follows:

Table 1 Transition Probability matrix using method 1

	1	2	3	4
1	0.36	0.17	0.19	0.28
2	0.20	0.27	0.33	0.20
3	0.19	0.24	0.36	0.21
4	0.31	0.16	0.17	0.36

Table 2 Stationary probability matrix using method 1

Very Low	Low	High	Very High
0.27	0.21	0.26	0.27

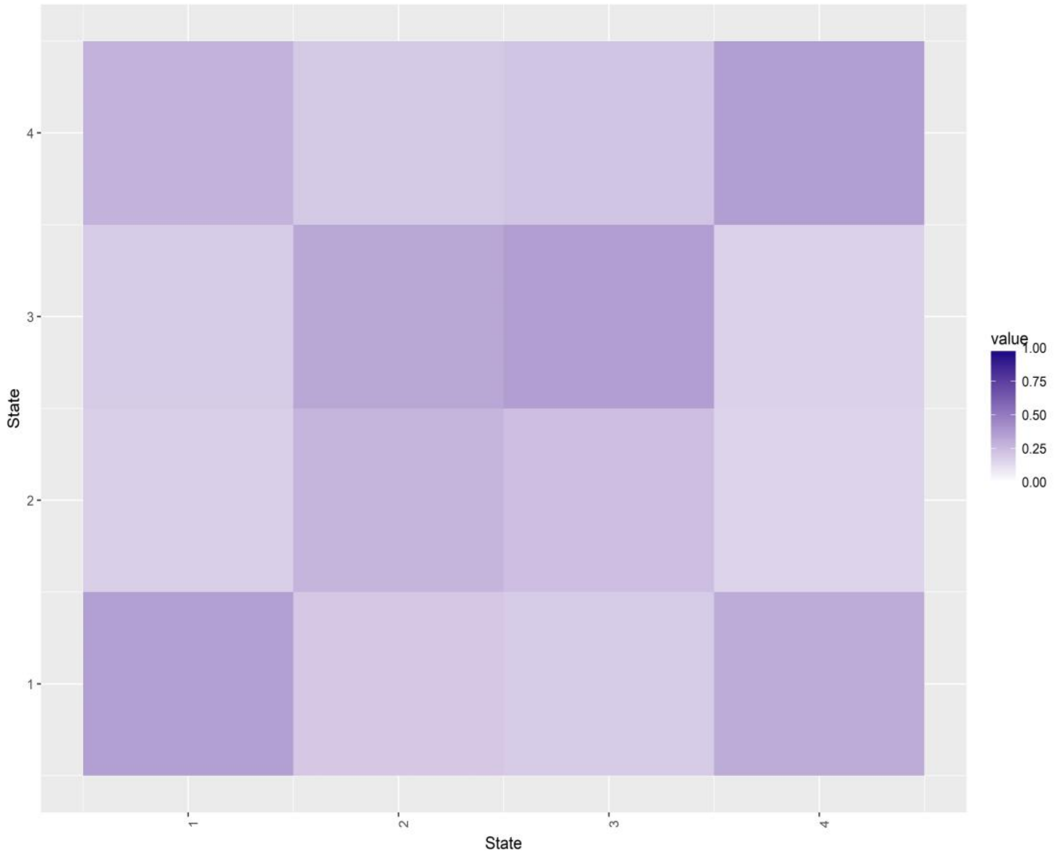

Figure 5-1 Representing Heatmap of transition probability obtained using method 1

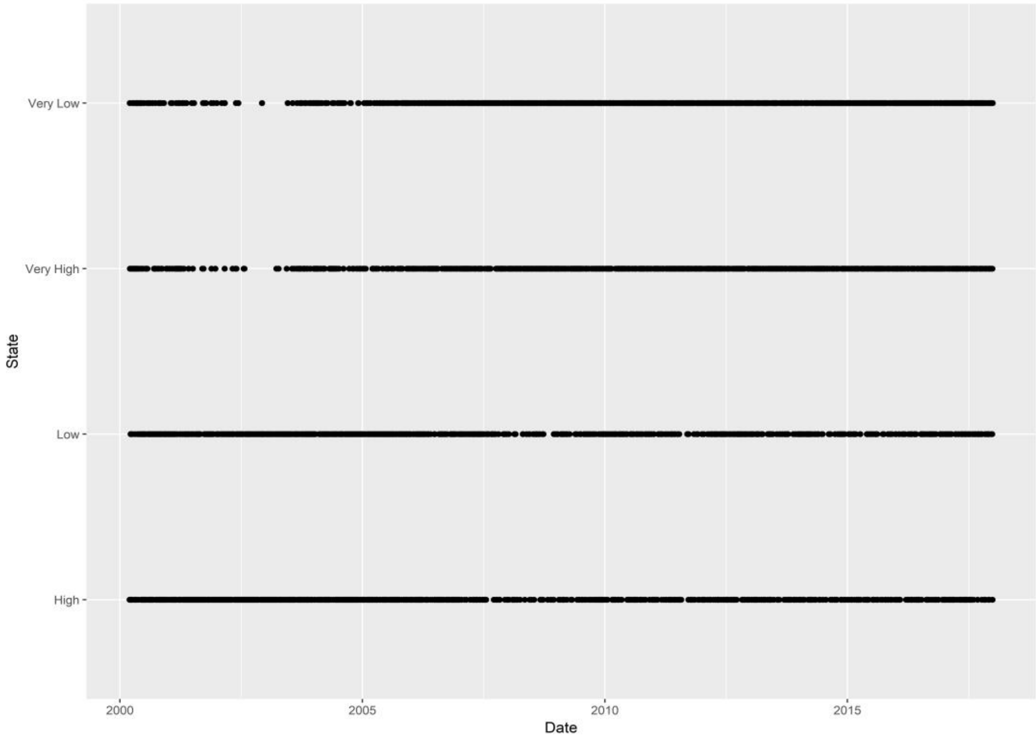

Figure 5-2 Image Shows Transition of States over time

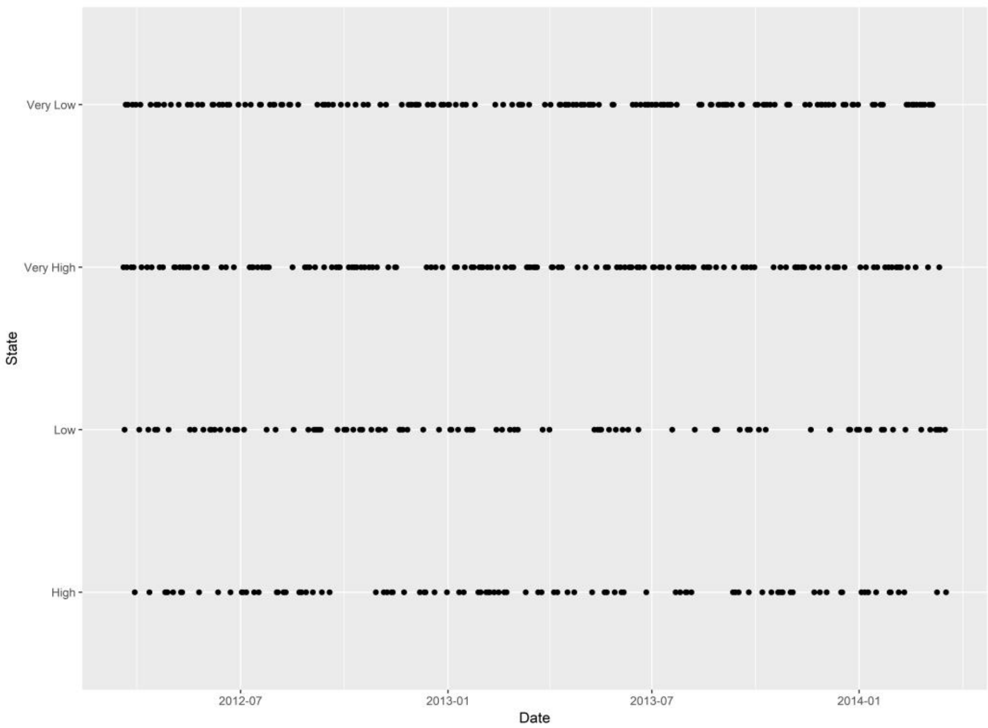

Figure 5-3 Image show states transition for period of 2012 to 2014

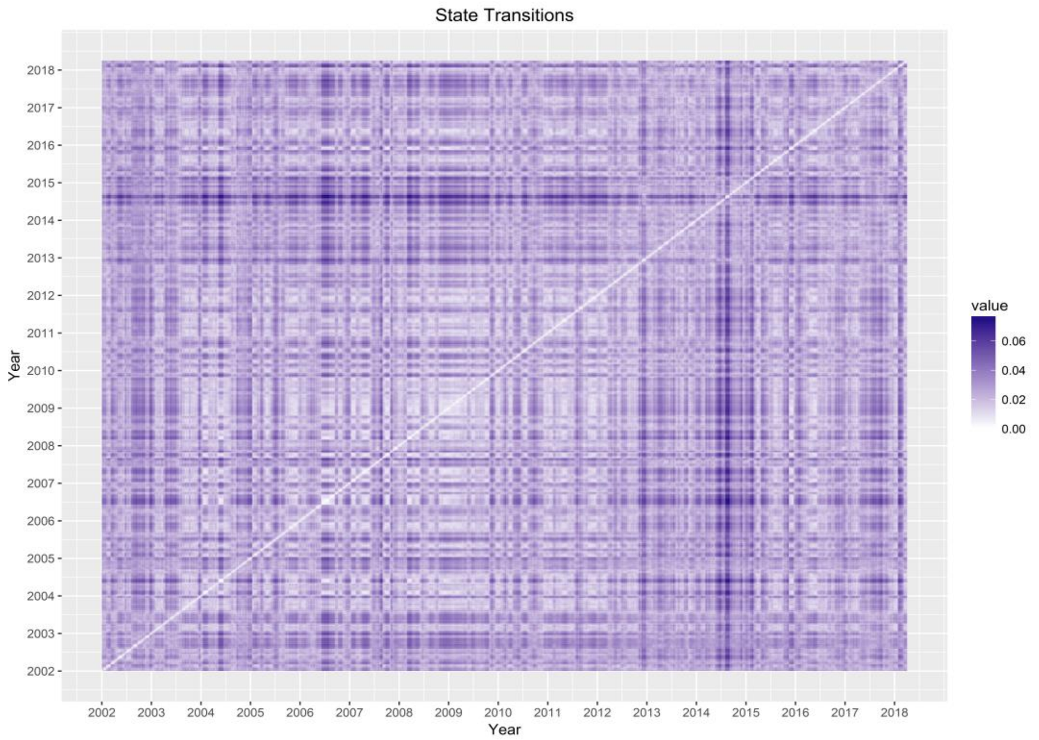

Figure 5-4 Heatmap Representing change in correlation matrix at different times

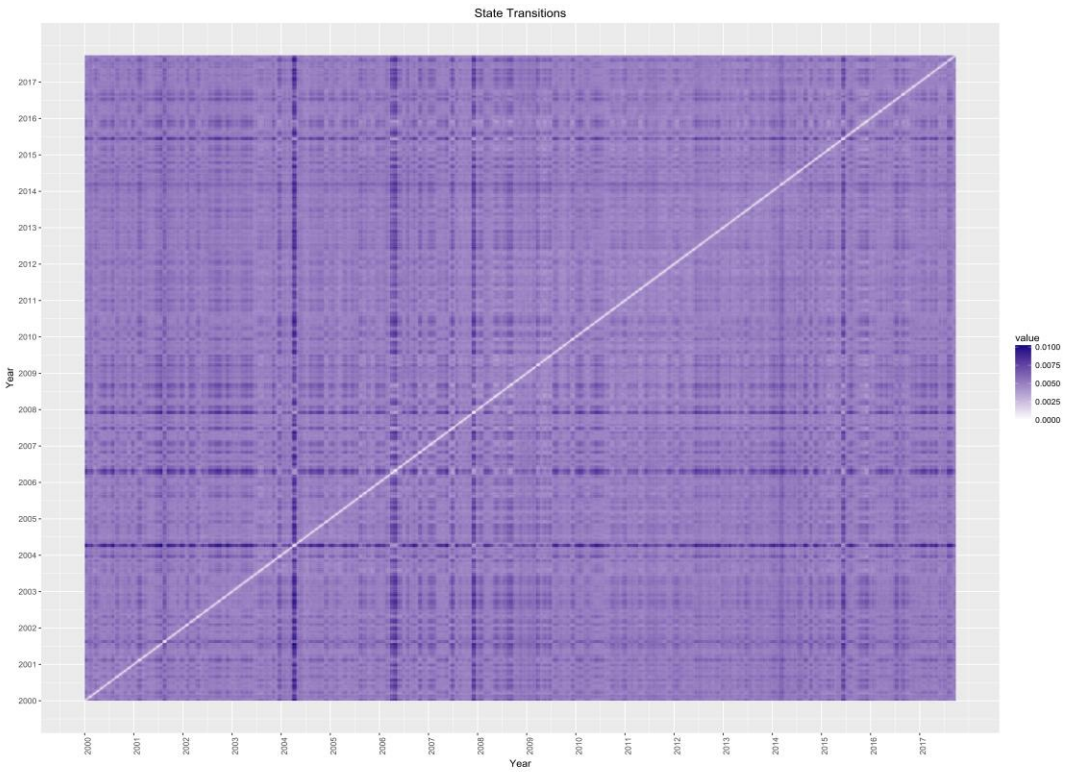

Figure 5-5 Heatmap representing change in correlation matrix (sector wise stock) at different times

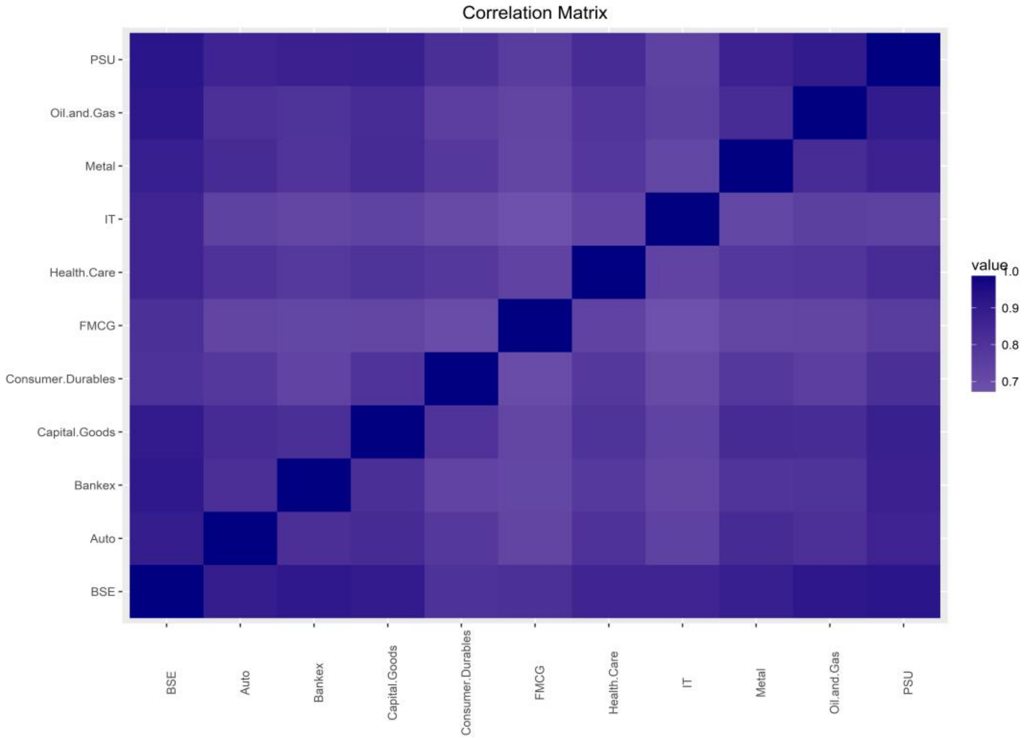

Figure 5-6 Correlation Matrix for State 1

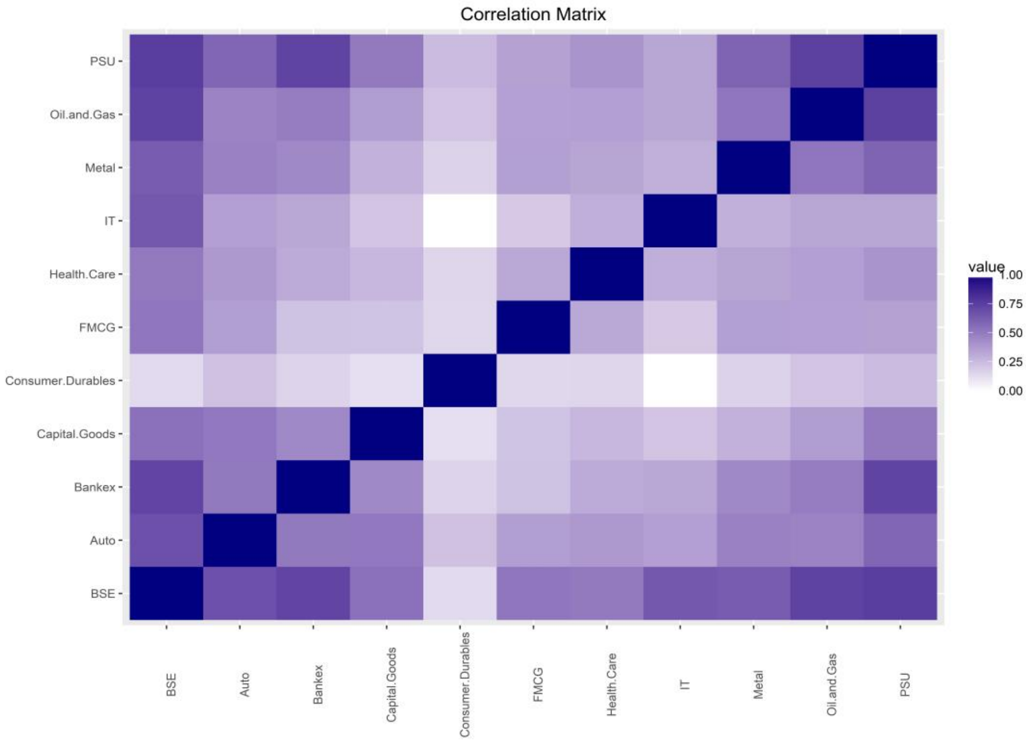

Figure 5-7 Correlation Matrix for State 2

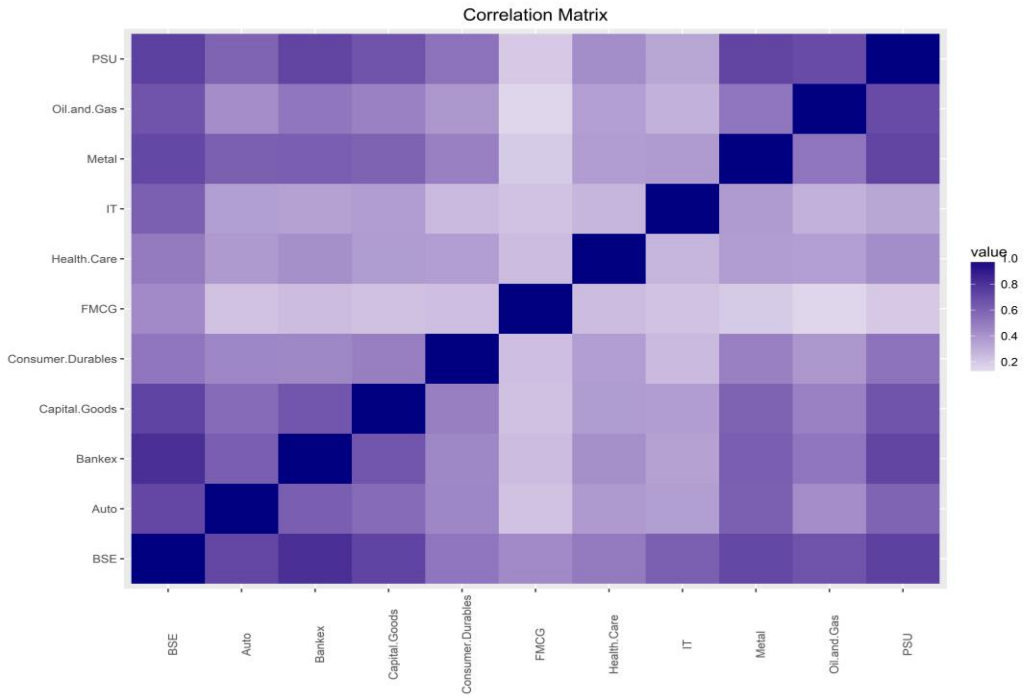

Figure 5-8 Correlation Matrix for State 3

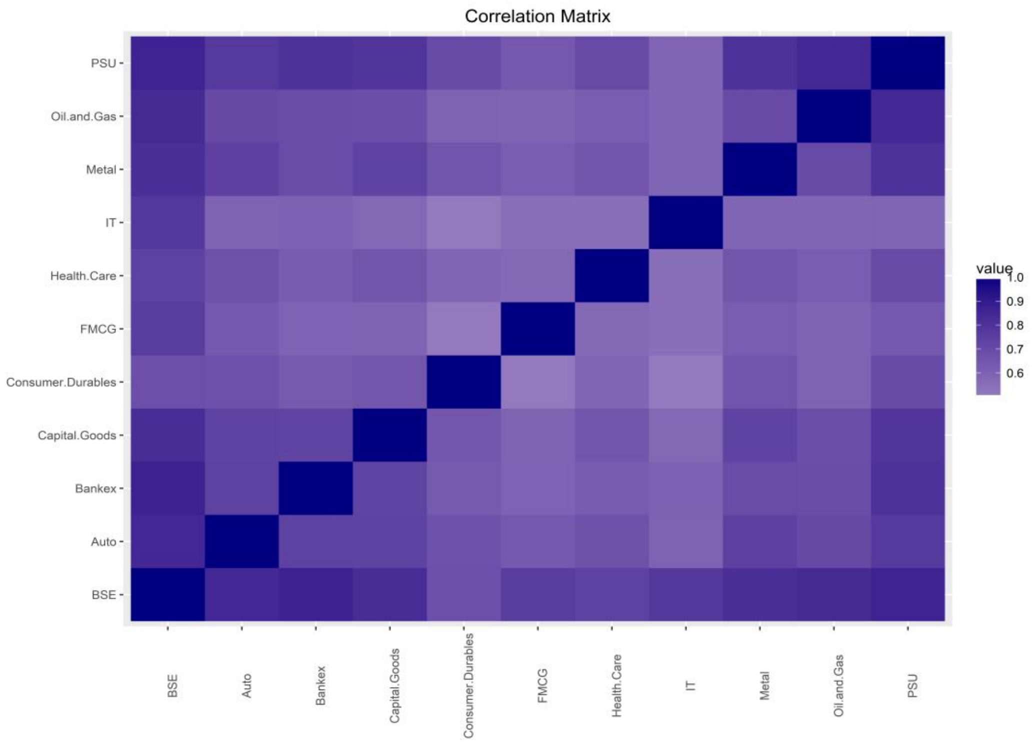

Figure 5-9 Correlation Matrix for State 4

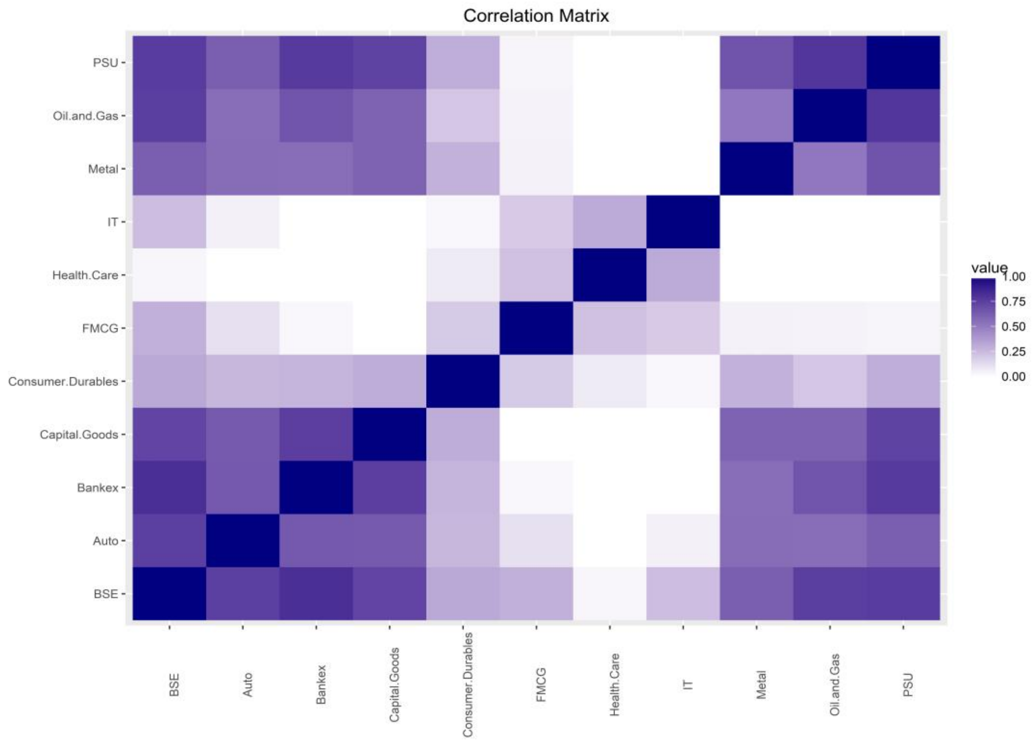

Figure 5-10 Correlation Matrix for State 5

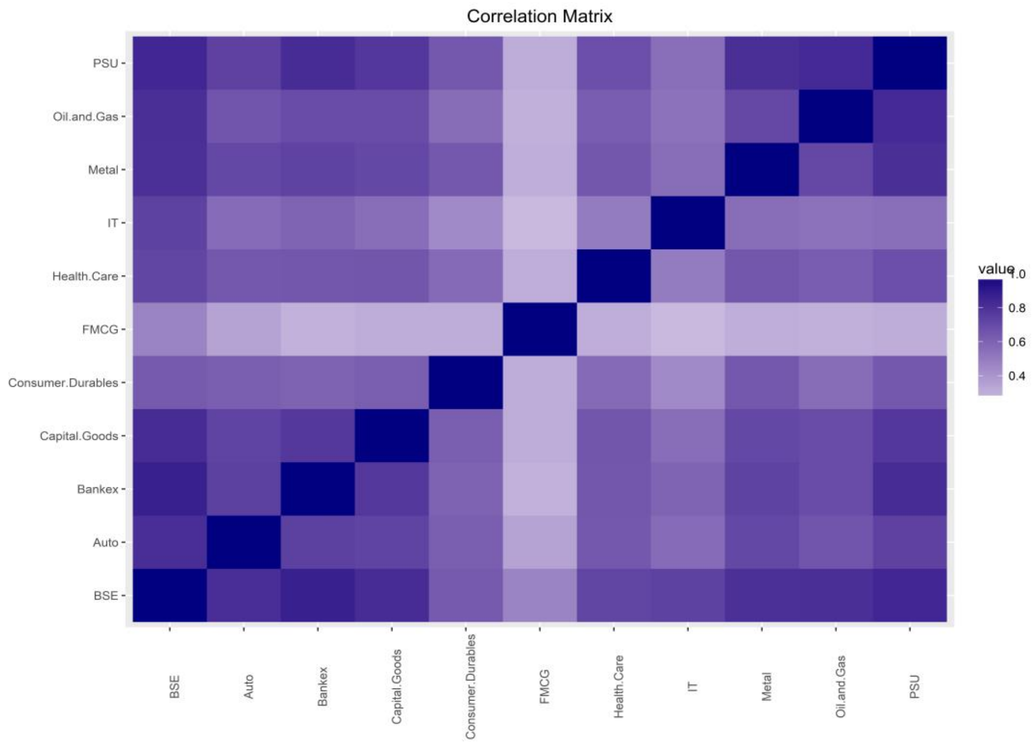

Figure 5-11 Correlation Matrix for State 6

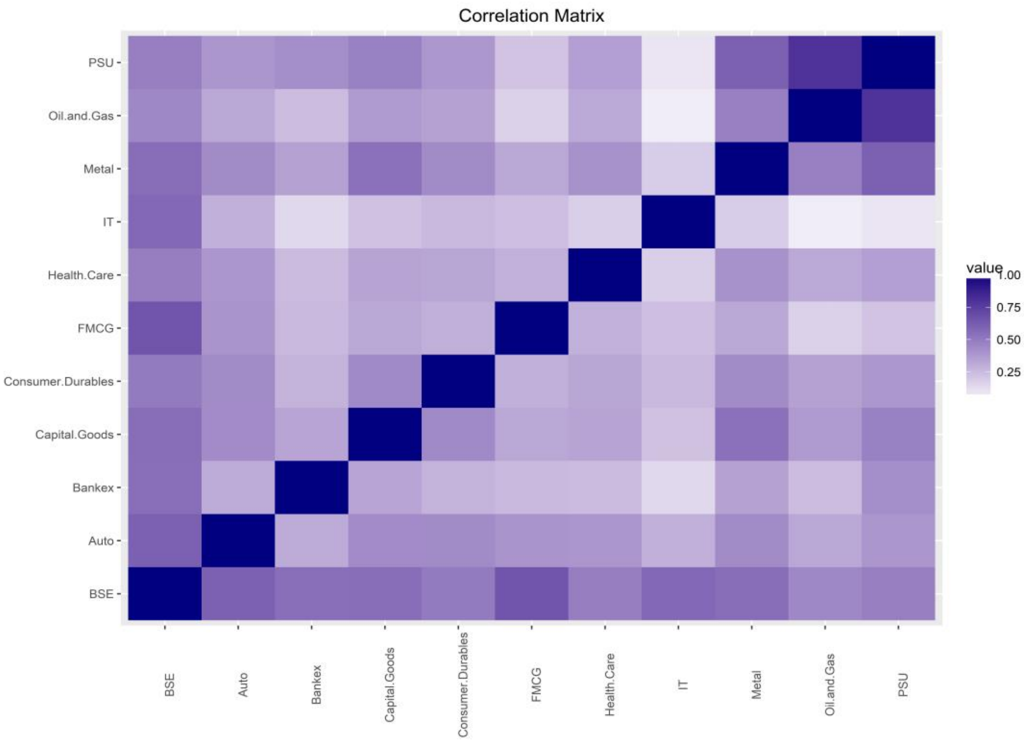

Figure 5-12 Correlation Matrix for State 7

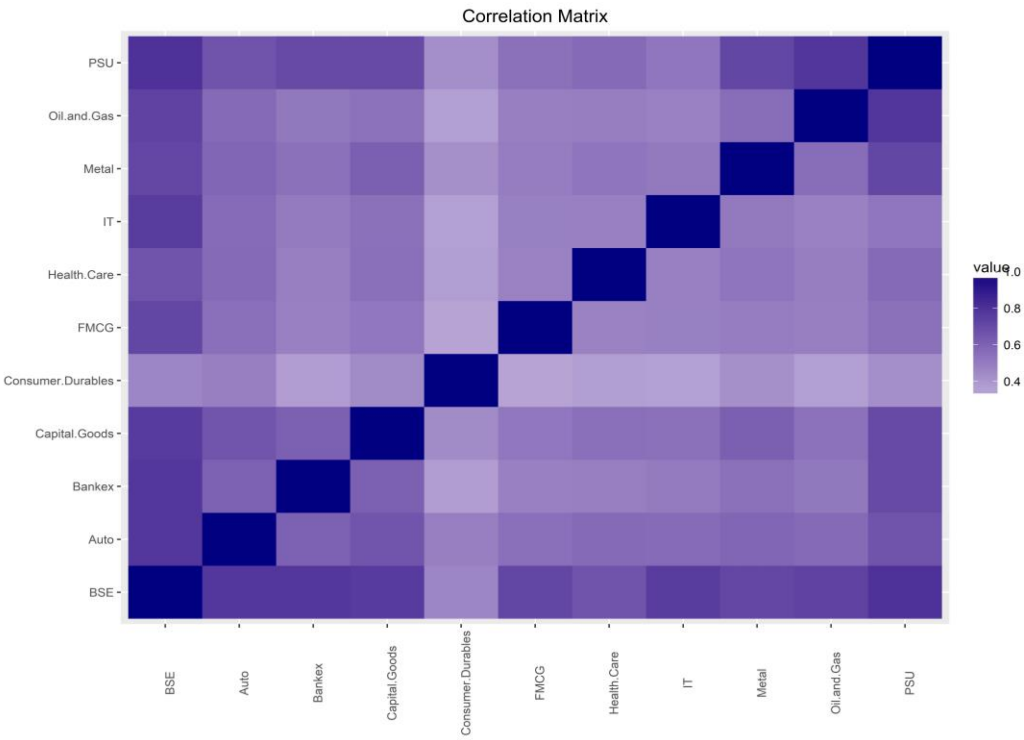

Figure 5-13 Correlation Matrix for State 8

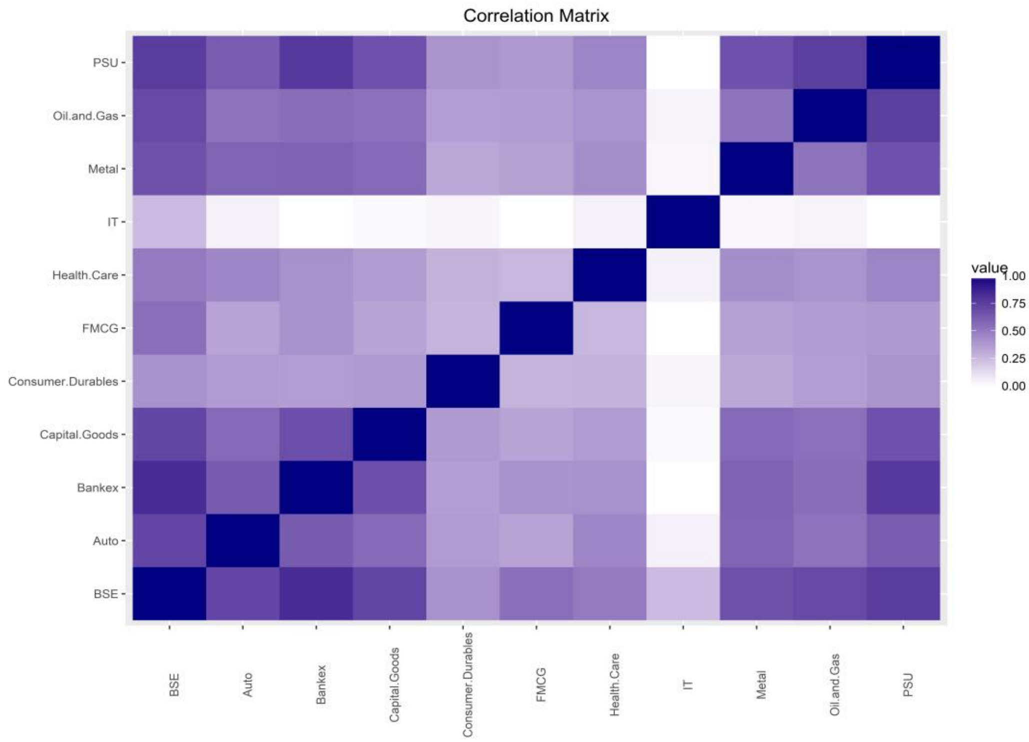

Figure 5-14 Correlation Matrix for State 9

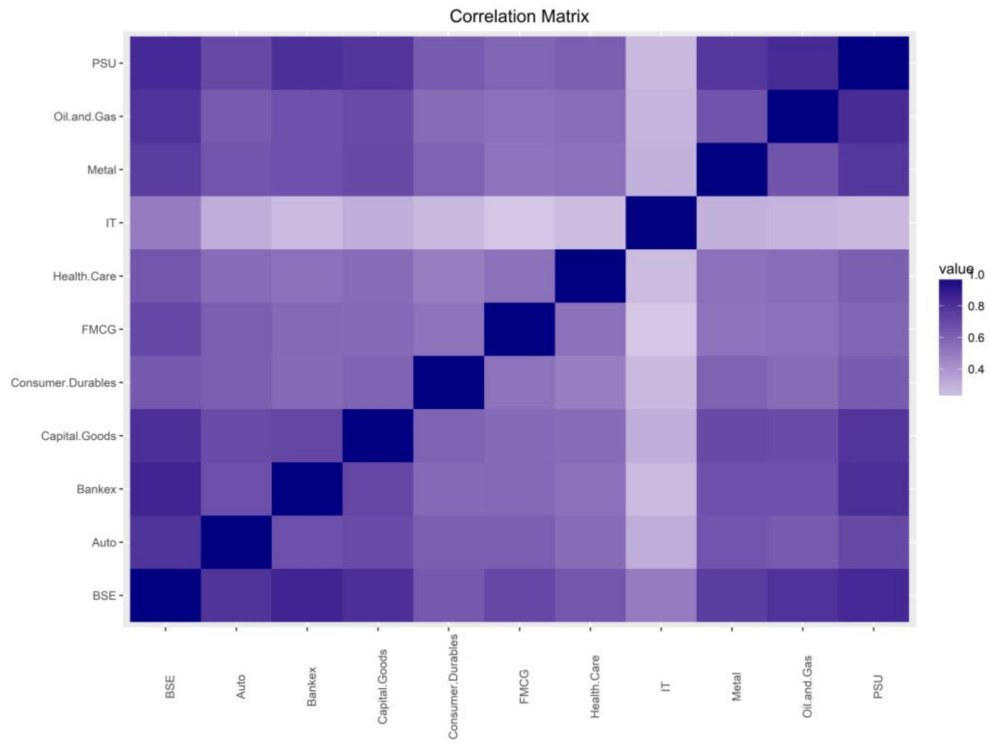

Figure 5-15 Correlation Matrix for State 10

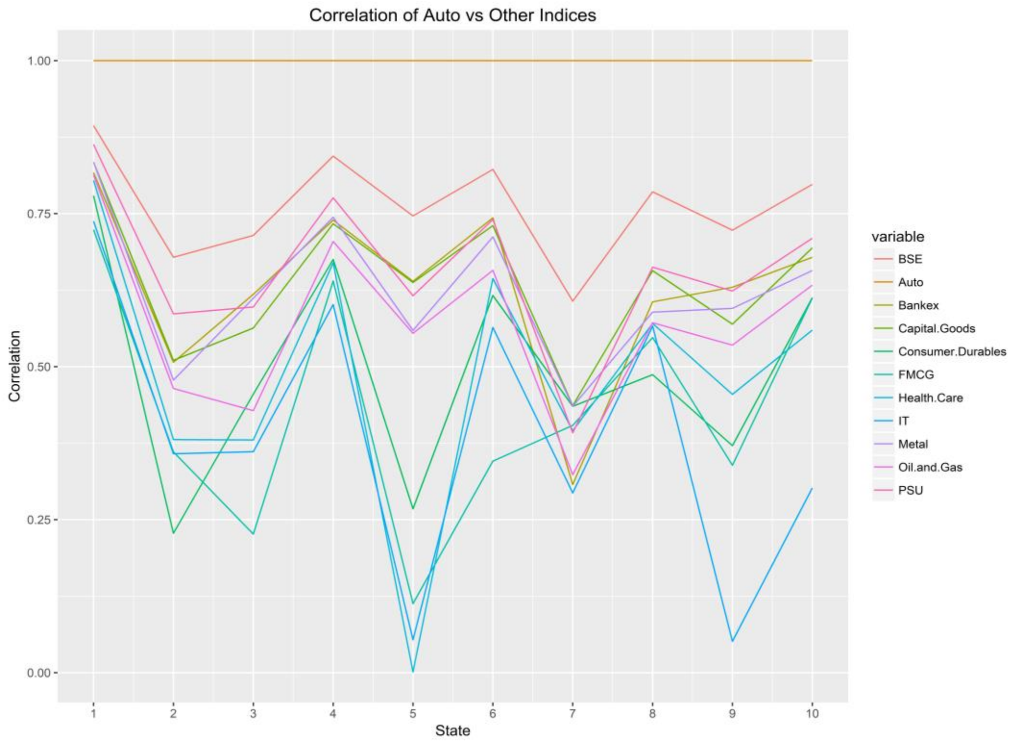

Figure 5-16 Correlation of Auto vs. other indices with change in states

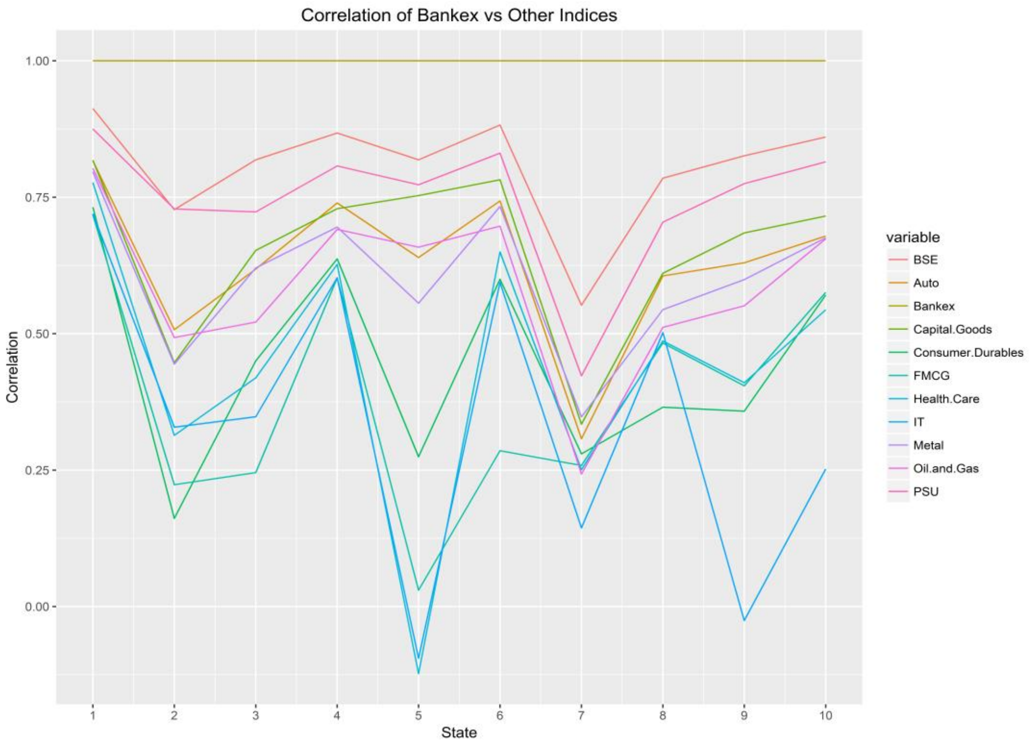

Figure 5-17 Correlation of Bankex vs. other indices with change in states

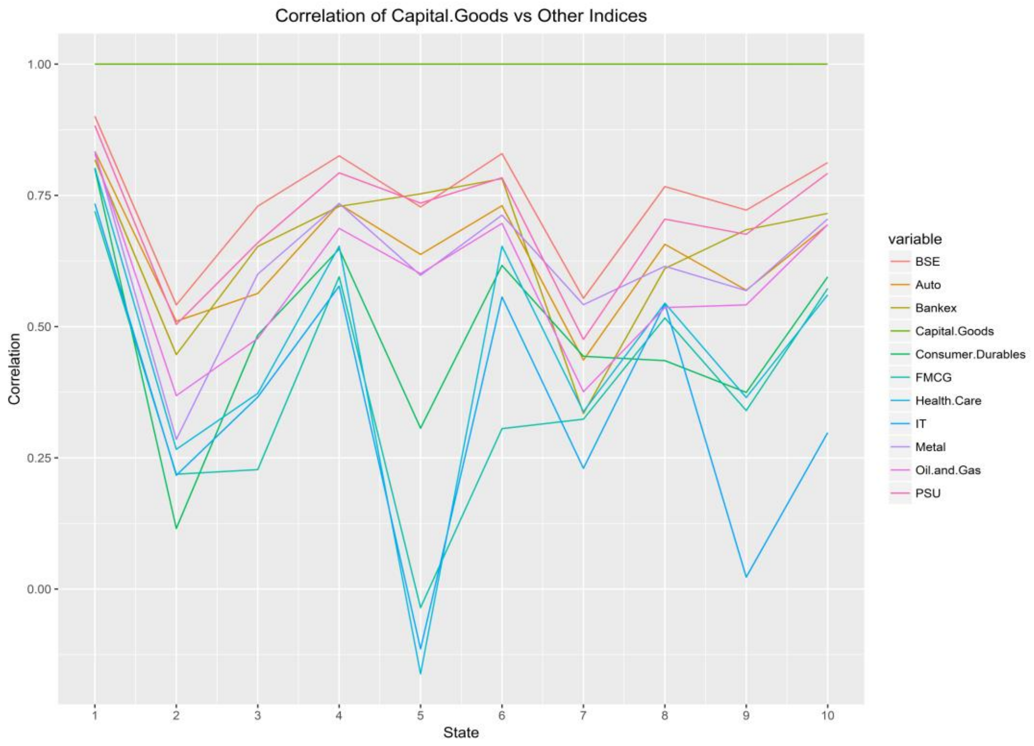

Figure 5-18 Correlation of Capital Goods vs. other indices with change in states

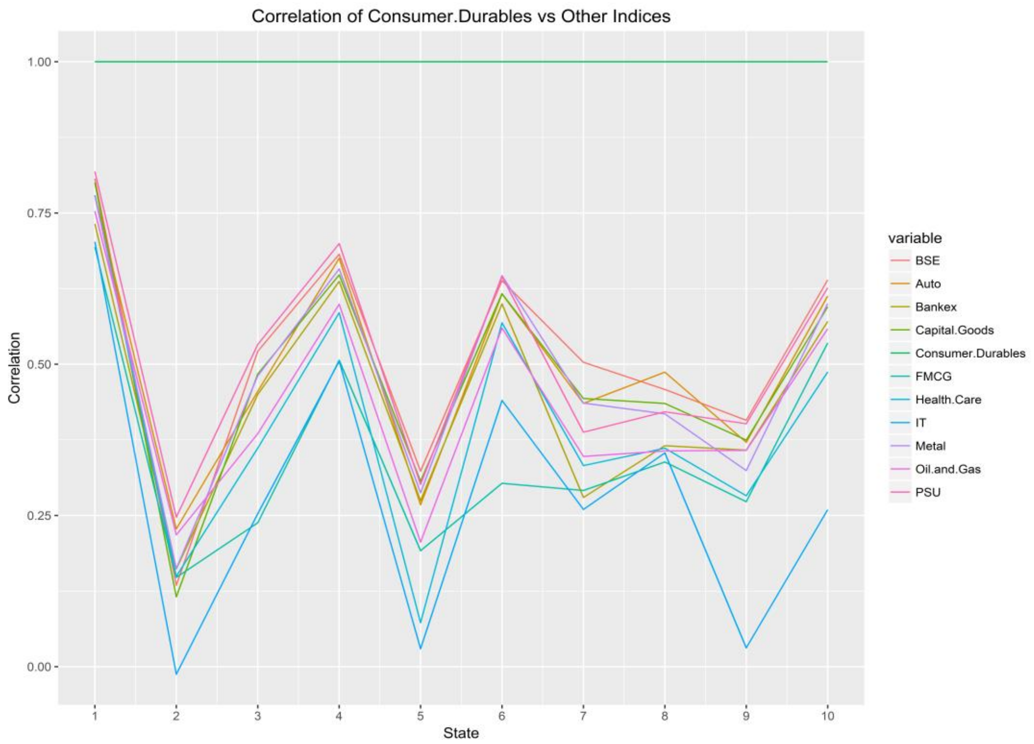

Figure 5-19 Correlation of Consumer Durables vs. other indices with change in states

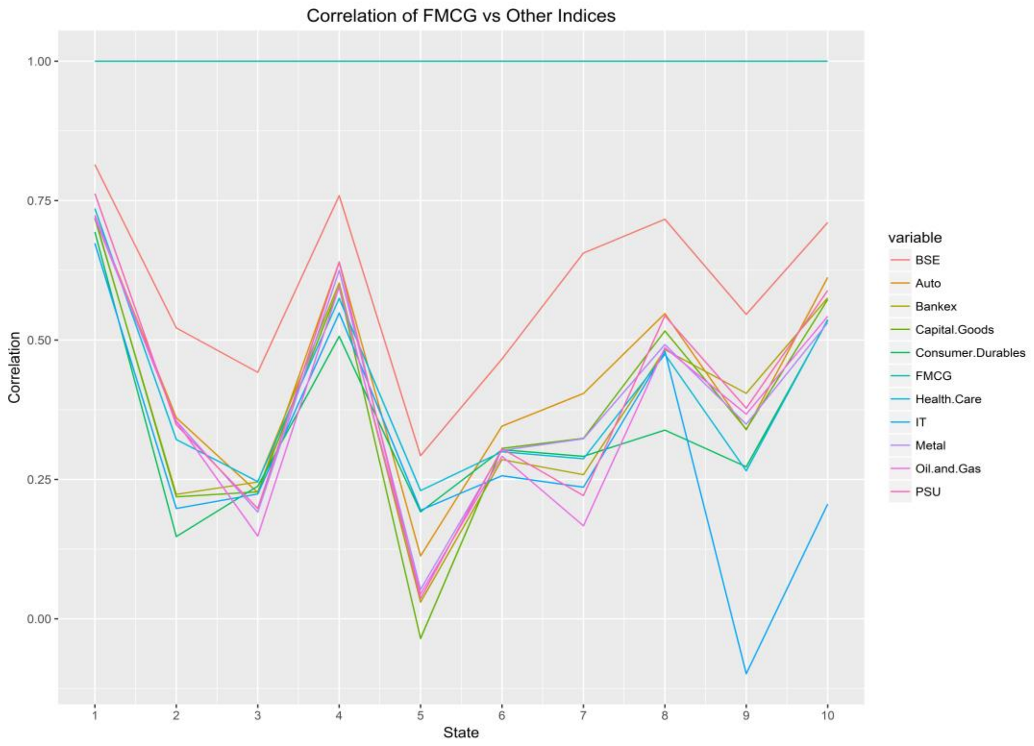

Figure 5-20 Correlation of FMCG vs. other indices with change in states

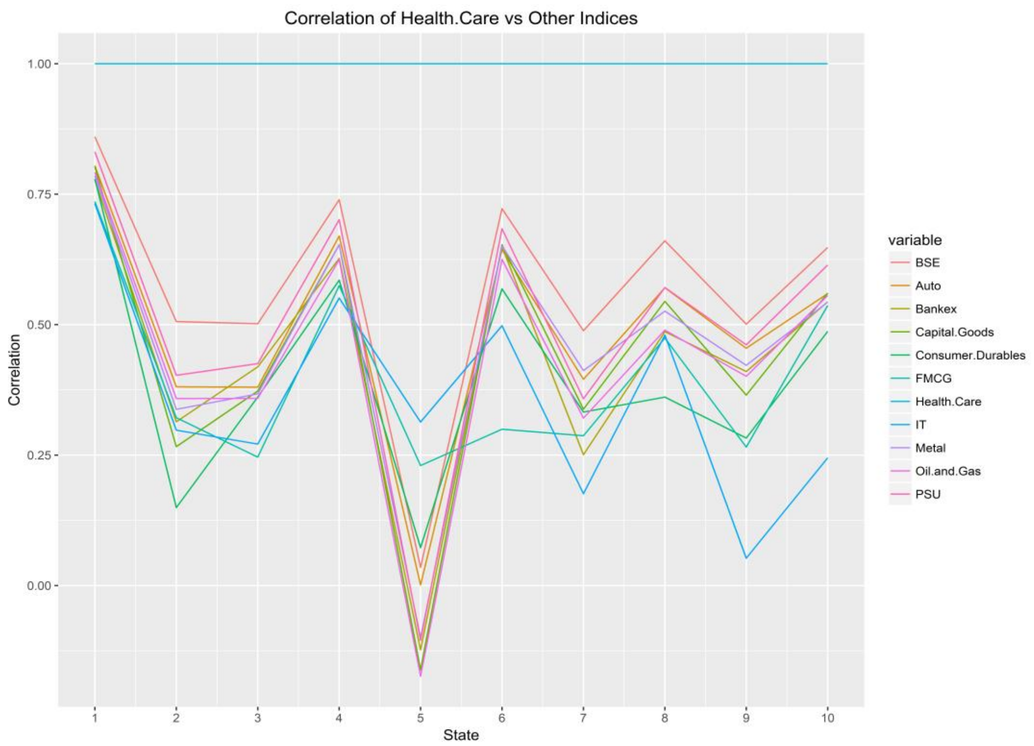

Figure 5-21 Correlation of Health Care vs. other indices with change in states

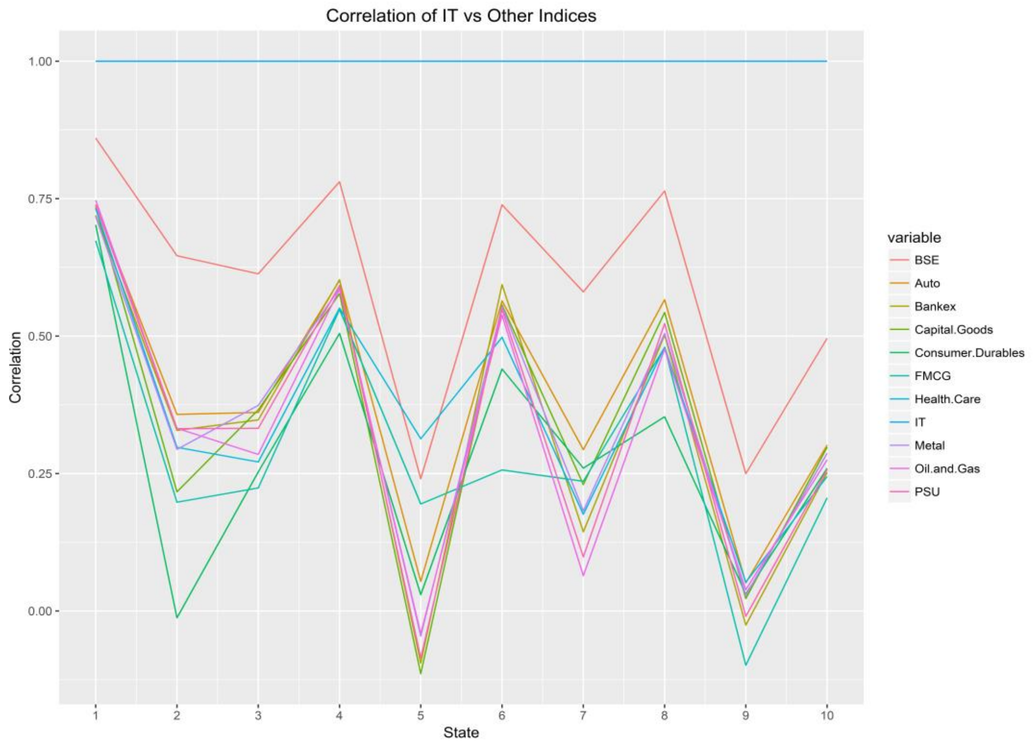

Figure 5-22 Correlation of IT vs. other indices with change in states

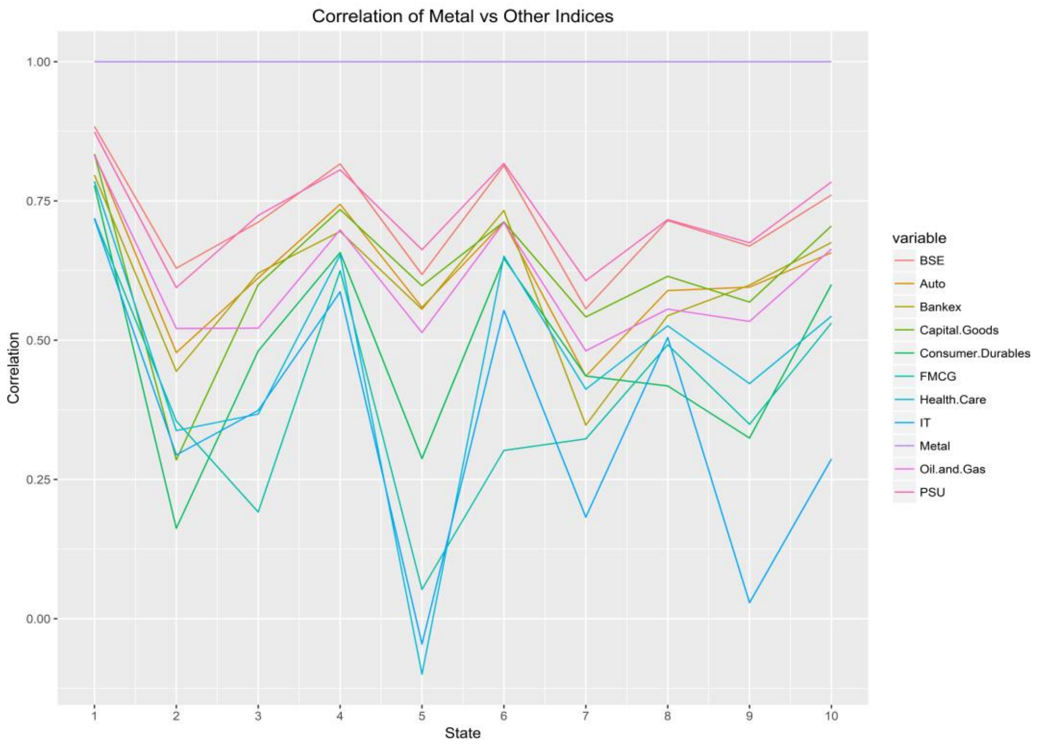

Figure 5-23 Correlation of Metal vs. other indices with change in states

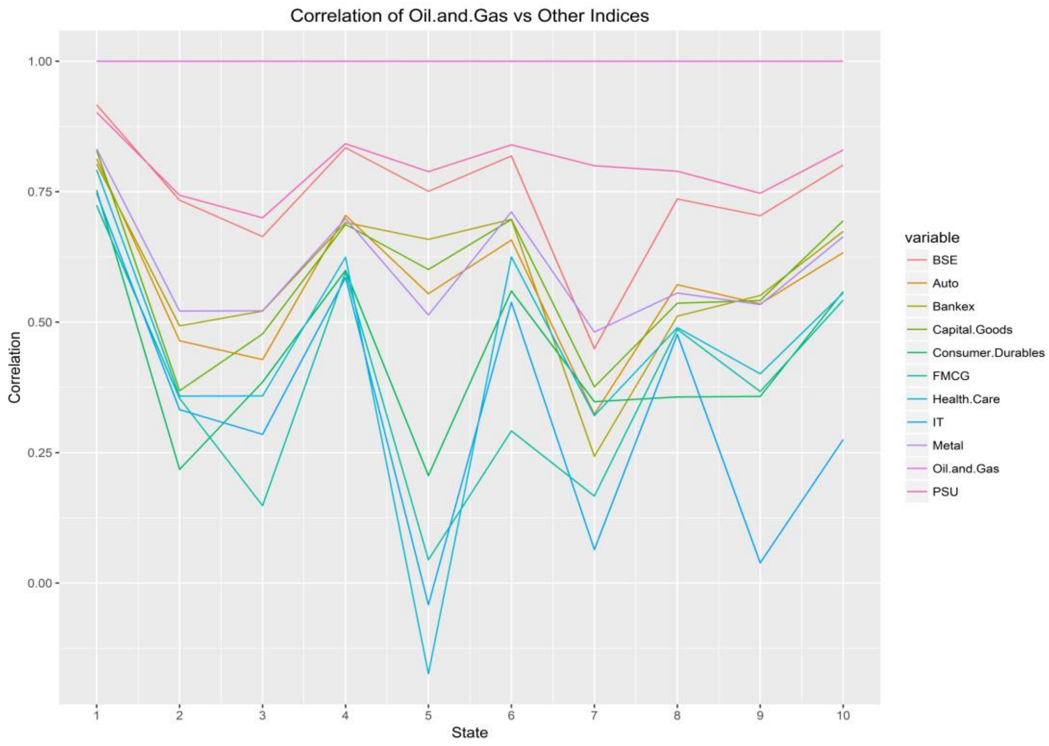

Figure 5-24 Correlation of Oil and Gas vs. other indices with change in states

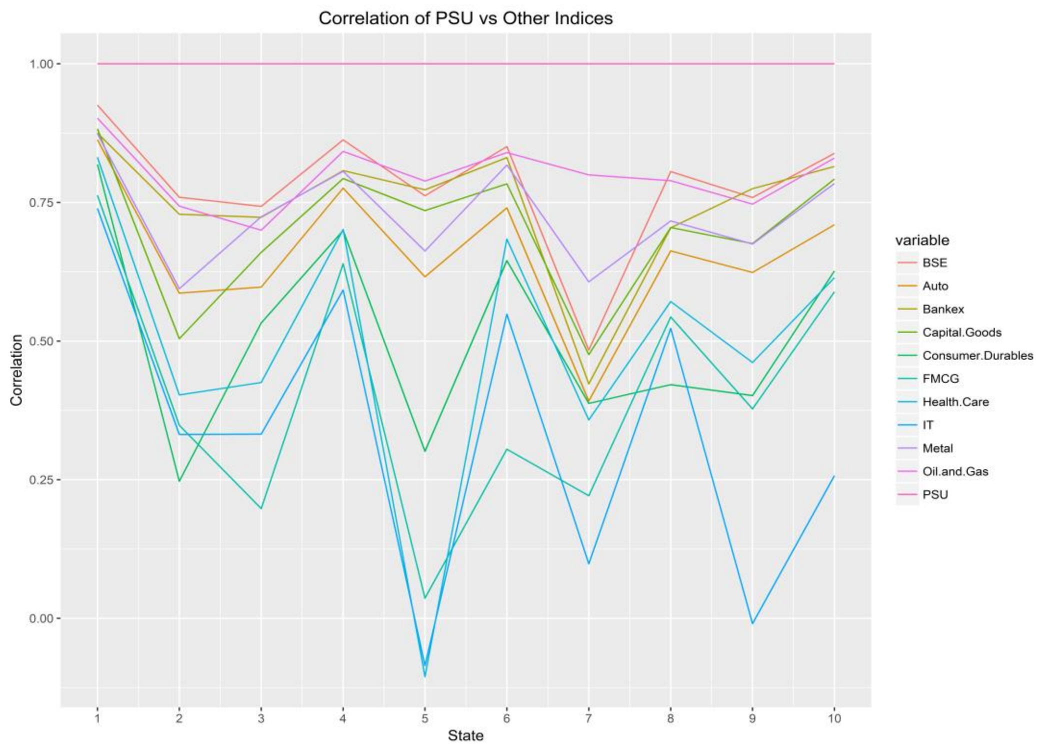

Figure 5-25 Correlation of PSU vs. other indices with change in states

Table 3 Transition probability matrix using method 2

	1	2	3	4	5	6	7	8	9	10
1	0.875	0.000	0.000	0.038	0.038	0.003	0.020	0.000	0.015	0.013
2	0.000	0.812	0.038	0.028	0.035	0.006	0.000	0.082	0.000	0.000
3	0.002	0.021	0.814	0.031	0.027	0.000	0.023	0.062	0.000	0.021
4	0.036	0.018	0.047	0.825	0.020	0.002	0.000	0.052	0.000	0.000
5	0.024	0.031	0.024	0.020	0.844	0.000	0.026	0.000	0.013	0.018
6	0.000	0.002	0.000	0.002	0.000	0.912	0.000	0.084	0.000	0.000
7	0.042	0.000	0.031	0.004	0.027	0.000	0.862	0.004	0.008	0.023
8	0.000	0.040	0.041	0.040	0.002	0.050	0.000	0.827	0.000	0.000
9	0.035	0.000	0.000	0.000	0.035	0.000	0.006	0.000	0.919	0.006
10	0.016	0.000	0.031	0.006	0.028	0.000	0.013	0.003	0.000	0.903

Table 4 Stationary probability using method 2

1	2	3	4	5	6	7	8	9	10
0.051	0.118	0.074	0.041	0.139	0.180	0.122	0.055	0.119	0.102

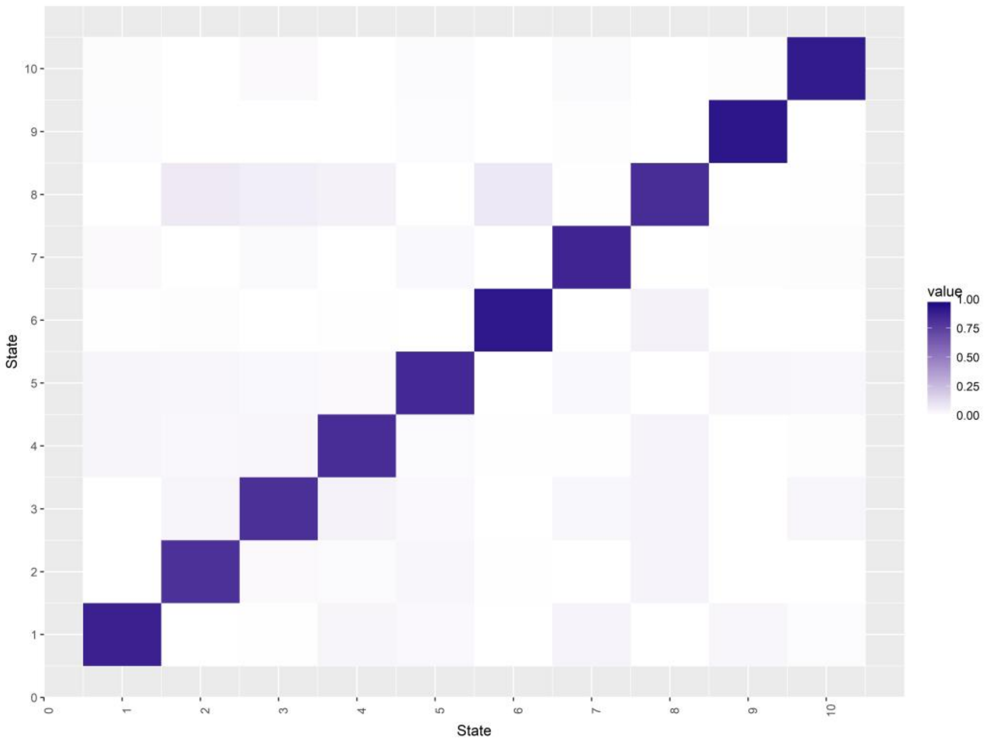

Figure 5-26 Heatmap of Transition matrix

Chapter 6 Conclusions

In the first part in which we estimated states using error, we can see that market surely moves in states but states are not distinguishable by simple models. This can be concluded from the fact that the steady probabilities are similar and also not very different terms of numerical value.

The second conclusion is the presence of market inertia, we can see that transition from the same state to another state is less likely in compared to transition to the same state. The first method of state estimation gave a very rough idea about the market states and thus it was not sufficient to conclude results.

In the second part we estimated states by applying k-NN clustering on an array of correlation matrices also we generated heatmap which tells us how the correlation has changed with respect to correlation at other time. We can see the dark-lines pattern on that map, which tells points of drastic changes. These can act as an indicator for financial crises.

In state ‘1’ we see most of the sector are moderately correlated. This state can be thought as resting state of the market. In state ‘2’ and state ‘8’ we see that Consumer durables are uncorrelated with other states which means that market of durable goods is stable and is not expected to change with others. The state ‘3’ similar to state ‘2’ in which we see FMCG is uncorrelated with others which generally means continuous supply and consumption of fast moving consumer goods. The state ‘4’ is similar to state ‘1’ but in this state, the market is less correlated to Health Care, Durables, FMCG and IT. State ‘5’ is like the counterpart of state ‘4’, it exists as a solution to clustering but does not provide significant information about the market and economy. State ‘6’ is similar to state ‘3’ but a major difference in IT. State ‘7’ represents a market in which major sectorial indices are uncorrelated. This is favourable as an effect of one cannot be felt by another. State ‘9’ and State ‘10’ are states in which IT is highly uncorrelated. These are aroused mostly from dotcom bubble of 2002 and well as high technological advancement in tech sector of 2017

An important conclusion can be seen from heatmap of transition matrix is the presence of market inertia, we see that market is more likely to be in the same state than to make the transition to another state. With this transition probability and State models, traders and investors can design strategies based on the current states.

References

Robert G Gallager. "Discrete Stochastic Process". Springer, Boston, MA. (1996)

Michael C. Münnix, Takashi Shimada, Rudi Schäfer, Francois Leyvraz, Thomas H. Seligman, Thomas Guhr & H. Eugene Stanley. Identifying States of a Financial Market. *Scientific Reports* volume2, Article number: 644. (2012)

Faber, V. Clustering and the continuous k-means algorithm. Los Alamos Science 22, 138–144. (1994)

Tastan, H. Estimating time-varying conditional correlations between stock and foreign exchange markets. Physica A 360, 445–458. (2006)

Pelletier, D. Regime switching for dynamic correlations. Journal of Econometrics 131, 445–473. (2006)