Visualizing and Forecasting Box-Office Revenues: A Case Study of the James Bond Movie Series

Vahan Petrosyan
Utah State University

Follow this and additional works at: https://digitalcommons.usu.edu/gradreports

Part of the Statistics and Probability Commons

Recommended Citation
Petrosyan, Vahan, "Visualizing and Forecasting Box-Office Revenues: A Case Study of the James Bond Movie Series" (2014). All Graduate Plan B and other Reports. 422.
https://digitalcommons.usu.edu/gradreports/422

This Thesis is brought to you for free and open access by the Graduate Studies at DigitalCommons@USU. It has been accepted for inclusion in All Graduate Plan B and other Reports by an authorized administrator of DigitalCommons@USU. For more information, please contact digitalcommons@usu.edu.
VISUALIZING AND FORECASTING BOX-OFFICE REVENUES: A CASE
STUDY OF THE JAMES BOND MOVIE SERIES

by

Vahan Petrosyan

A report submitted in partial fulfillment
of the requirements for the degree
of
MASTER OF SCIENCE
in
Statistics

Approved:

Dr. Jürgen Symanzik  Dr. Daniel C. Coster
Major Professor  Committee Member

Dr. Yan Sun
Committee Member

UTAH STATE UNIVERSITY
Logan, Utah
2014
ABSTRACT

Visualizing and Forecasting Box–Office Revenues: A Case Study of the James Bond Movie Series

by

Vahan Petrosyan, Master of Science
Utah State University, 2014

Major Professor: Dr. Jürgen Symanzik
Department: Mathematics and Statistics

This Master’s report deals with the visualization and forecasting of the box–office revenues and some related variables from the James Bond movie series. Visualization techniques such as time series plots, scatterplot matrices, dotplots, boxplots, histograms, normal quantile plots, parallel coordinates plots, heatmaps, mosaic plots, association plots, and choropleth maps are used to provide some deeper insights into the given dataset. Additionally, the results from an article published in 1997 are reproduced and extended. This article modeled the box–office revenues of the James Bond movie series. Numerous statistical models were examined to obtain the models that are closest to the original models. Then, these reproduced models are compared with newer methods such as LASSO and random forests to determine how to best forecast the box–office revenues of recent (and future) James Bond movies.
## CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>ii</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>v</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>vi</td>
</tr>
<tr>
<td>1 INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 The Importance of the Movie Industry</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Previous Research</td>
<td>2</td>
</tr>
<tr>
<td>1.3 James Bond Movies</td>
<td>3</td>
</tr>
<tr>
<td>1.4 Previous Research: James Bond Movies</td>
<td>4</td>
</tr>
<tr>
<td>1.5 Data for James Bond Movies</td>
<td>6</td>
</tr>
<tr>
<td>1.5.1 Data Sources</td>
<td>6</td>
</tr>
<tr>
<td>1.5.2 Explanatory Variables: The Economist and Baimbridge Models</td>
<td>7</td>
</tr>
<tr>
<td>1.6 Objectives</td>
<td>8</td>
</tr>
<tr>
<td>2 VISUALIZING THE ECONOMIST DATASET</td>
<td>10</td>
</tr>
<tr>
<td>2.1 Statistical Graphics</td>
<td>10</td>
</tr>
<tr>
<td>2.2 Time Series Plots</td>
<td>10</td>
</tr>
<tr>
<td>2.2.1 Kills</td>
<td>11</td>
</tr>
<tr>
<td>2.2.2 Conquests</td>
<td>12</td>
</tr>
<tr>
<td>2.2.3 Martinis</td>
<td>12</td>
</tr>
<tr>
<td>2.2.4 Bond, James Bond (BJB)</td>
<td>14</td>
</tr>
<tr>
<td>2.3 Scatterplot Matrix</td>
<td>15</td>
</tr>
<tr>
<td>2.4 Dot Plots</td>
<td>17</td>
</tr>
<tr>
<td>2.5 Box Plots</td>
<td>19</td>
</tr>
<tr>
<td>2.6 Histogram and Normal QQ Plot</td>
<td>21</td>
</tr>
<tr>
<td>2.7 Parallel Coordinates Plots</td>
<td>23</td>
</tr>
<tr>
<td>2.8 Heatmaps</td>
<td>24</td>
</tr>
<tr>
<td>2.9 Mosaic Plot</td>
<td>28</td>
</tr>
<tr>
<td>2.10 Association Plot</td>
<td>30</td>
</tr>
<tr>
<td>2.11 Choropleth Maps</td>
<td>32</td>
</tr>
<tr>
<td>2.12 Summary</td>
<td>36</td>
</tr>
<tr>
<td>3 REPLICATION OF BAIMBRIDGE’S MODEL</td>
<td>38</td>
</tr>
<tr>
<td>3.1 Reproducible Research (RR)</td>
<td>38</td>
</tr>
<tr>
<td>3.2 Replication of Baimbridge (1997)</td>
<td>39</td>
</tr>
<tr>
<td>3.2.1 First Model</td>
<td>41</td>
</tr>
<tr>
<td>3.2.2 Second Model</td>
<td>46</td>
</tr>
<tr>
<td>3.2.3 Third Model</td>
<td>49</td>
</tr>
</tbody>
</table>
### 4 PREDICTING THE BOX-OFFICE REVENUES OF THE JB MOVIE SERIES

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 Prediction Methods Overview</td>
<td>60</td>
</tr>
<tr>
<td>4.1.1 Ordinary Least Squares (OLS)</td>
<td>61</td>
</tr>
<tr>
<td>4.1.2 LASSO</td>
<td>61</td>
</tr>
<tr>
<td>4.1.3 Random Forests</td>
<td>61</td>
</tr>
<tr>
<td>4.1.4 Benchmarks</td>
<td>62</td>
</tr>
<tr>
<td>4.2 Comparison of the First Model</td>
<td>62</td>
</tr>
<tr>
<td>4.3 Comparison of the Third Model</td>
<td>64</td>
</tr>
<tr>
<td>4.4 Comparison of The Economist Model</td>
<td>66</td>
</tr>
<tr>
<td>4.5 Summary of the Model Comparison</td>
<td>68</td>
</tr>
<tr>
<td>4.6 Usage of R Packages</td>
<td>69</td>
</tr>
</tbody>
</table>

### 5 CONCLUSION AND OUTLOOK

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 Future Work</td>
<td>72</td>
</tr>
</tbody>
</table>

### APPENDICES

<table>
<thead>
<tr>
<th>Appendix</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.1 Inflation Adjusters</td>
<td>81</td>
</tr>
<tr>
<td>A.2 The Response</td>
<td>82</td>
</tr>
<tr>
<td>A.3 The Economist</td>
<td>83</td>
</tr>
<tr>
<td>A.4 The First Model</td>
<td>84</td>
</tr>
<tr>
<td>A.5 The Second Model</td>
<td>85</td>
</tr>
<tr>
<td>A.6 The Third Model</td>
<td>86</td>
</tr>
<tr>
<td>A.7 The Forth Model</td>
<td>87</td>
</tr>
<tr>
<td>B.1 R Code for Chapter 2</td>
<td>88</td>
</tr>
<tr>
<td>B.2 R Code for Chapter 3</td>
<td>109</td>
</tr>
<tr>
<td>B.3 R Code for Chapter 4</td>
<td>131</td>
</tr>
</tbody>
</table>


LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Summary table of James Bond movies. The values of BORs are in millions of dollars. Inflation adjustment year is 2014.</td>
</tr>
<tr>
<td>2</td>
<td>OLS summary results of kills, conquests, martinis, and BJB over time.</td>
</tr>
<tr>
<td>3</td>
<td>The characteristics of the “Best” replicated models. The column C&amp;0 shows the usage of the Cochrane and Orcutt technique.</td>
</tr>
<tr>
<td>4</td>
<td>Summary of the RMSE values for the test sets.</td>
</tr>
</tbody>
</table>


## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Number of JB kills per movie over time.</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>Number of Bond conquests per movie over time.</td>
<td>13</td>
</tr>
<tr>
<td>3</td>
<td>Number of martinis drunk by Bond per movie over time.</td>
<td>13</td>
</tr>
<tr>
<td>4</td>
<td>Number of “Bond, James Bond” expressions made per movie over time.</td>
<td>15</td>
</tr>
<tr>
<td>5</td>
<td>Scatterplot matrix.</td>
<td>16</td>
</tr>
<tr>
<td>6</td>
<td>Summary of averages by actor.</td>
<td>18</td>
</tr>
<tr>
<td>7</td>
<td>BORs, with respect to high, medium and small number of kills, conquests, martinis and BJB, sorted by median BOR within each category.</td>
<td>20</td>
</tr>
<tr>
<td>8</td>
<td>Box plots, showing the average inflation adjusted BOR by JB actor, sorted by median BOR within each JB actor.</td>
<td>21</td>
</tr>
<tr>
<td>9</td>
<td>Histogram and normal QQ plot for box–office and log box–office revenues.</td>
<td>22</td>
</tr>
<tr>
<td>10</td>
<td>Parallel coordinate plot of number of Bond kills, martinis, conquests, “Bond, James Bond” expression.</td>
<td>23</td>
</tr>
<tr>
<td>11</td>
<td>Heatmap plot of kills, conquests, martinis, and BJB expression by actor name and movie release date. The histogram on the top left panel shows the distribution of the data matrix.</td>
<td>25</td>
</tr>
<tr>
<td>12</td>
<td>Heatmap plot of square–root transformed kills, conquests, martinis, and BJB expression by actor name and movie release date. The histogram on the top left panel shows the distribution of the data matrix.</td>
<td>26</td>
</tr>
<tr>
<td>13</td>
<td>Mosaic plot for kills, conquests, martinis and BJB expression.</td>
<td>29</td>
</tr>
<tr>
<td>14</td>
<td>Association plot for kills, conquests, martinis, and BJB expression.</td>
<td>31</td>
</tr>
<tr>
<td>15</td>
<td>Number of Bond visits before the collapse of the USSR.</td>
<td>33</td>
</tr>
<tr>
<td>16</td>
<td>Number of Bond visits after the collapse of the USSR.</td>
<td>34</td>
</tr>
<tr>
<td>17</td>
<td>Average BOR (in millions) by country before (top panel) and after (bottom panel) the collapse of the USSR.</td>
<td>35</td>
</tr>
</tbody>
</table>
18 Summary results extracted from Baimbridge (1997), Table 1

19 The replication of the first model discussed in Baimbridge (1997). The parallel coordinates plot shows the original (in black) and 96 replicated models (in red and blue). Blue lines indicate the usage of the Cochrane and Orcutt technique. The dark red line shows the best model. The dashed line represents 0. Min = -0.21 and Max = 2.01 here.

20 OLS summary for the first model

21 Comparison of the first model discussed in Baimbridge (1997) and the best replicated model. The results of the replicated model are presented via red squares and the results of the original models are presented via blue circles.

22 The replication of the second model discussed in Baimbridge (1997). The parallel coordinates plot shows the original (in black) and 12 replicated models (in red and blue). Blue lines indicate the usage of the Cochrane and Orcutt technique. The dark red line shows the best model. The dashed line represents 0. Min = -0.30 and Max = 2.82 here.

23 OLS summary for the second model

24 Comparison of the second model discussed in Baimbridge (1997) and the best replicated model. The results of the replicated model are presented via red squares and the results of the original models are presented via blue circles.

25 The replication of the third model discussed in Baimbridge (1997). The parallel coordinates plot shows the original (in black) and 48 replicated models (in red and blue). Blue lines indicate the usage of the Cochrane and Orcutt technique. The dark blue line shows the best model. The dashed line represents 0. Min = -0.53 and Max = 4.58 here.

26 OLS summary for the third model. The variable names are different because the Cochrane and Orcutt method was adopted.

27 Comparison of the third model discussed in Baimbridge (1997) and the best replicated model. The results of the replicated model are presented via red squares and the results of the original models are presented via blue circles.
28 The replication of the fourth model discussed in Baimbridge (1997). The parallel coordinates plot shows the original (in black) and 24 replicated models (in red and blue). Blue lines indicate the usage of the Cochrane and Orcutt technique. The dark red line shows the best model. The dashed line represents 0. Min = -361 and Max = 177 here. The unit of the SSE variable is in thousands. 

29 OLS summary for the fourth model.

30 Comparison of the fourth model discussed in Baimbridge (1997) and the best replicated model. The results of the replicated model are presented via red squares and the results of the original models are presented via blue circles.

31 The second attempt to replicate the fourth model discussed in Baimbridge (1997). The parallel coordinates plot shows the original (in black) and 408 (12 × 30 + 12 × 4) replicated models (in red and blue). Blue lines indicate the usage of the Cochrane and Orcutt technique. The dark red line shows the best model. The dashed line represents 0. Min = -110 and Max = 42 here. The unit of the SSE variable is in hundreds.

32 Observed and predicted values of the log–transformed BORs (left) for the first model. The OLS of the best replicated and the Baimbridge models are shown in the top left panel. LASSO and random forests appear in the bottom left panel. The faint colored points represent the training set. Prediction results are shown with dark colored points. The dashed line in the left panel shows the average of the first 16 movies. The RMSE for the training and test sets of these models are shown in the right panel.

33 Observed and predicted values of the log–transformed BORs (left) for the third model. The OLS of the best replicated and the Baimbridge models are shown in the top left panel. LASSO and random forests appear in the bottom left panel. The faint colored points represent the training set. Prediction results are shown with dark colored points. The dashed line in the left panel shows the average of the first 16 movies. The RMSE for training and the test sets of these models are shown in the right panel.
Observed and predicted values of the log-transformed BORs (left) for The Economist model. The OLS of the best replicated and the Bench Mean 2 models are shown in the top left panel. LASSO and random forests appear in the bottom left panel. The faint colored points represent the training set. Prediction results are shown with dark colored points. The dashed line in the left panel shows the average of the first 16 movies. The RMSE for the training and the test set of these models are shown in the right panel.
CHAPTER 1
INTRODUCTION

1.1 The Importance of the Movie Industry

The movie industry is not only an influential part of the arts but it is also a vital participant of the business field. It plays an important role in the stage of the world’s economy. Specifically, in the United States, the movie industry provided over 2.2 million jobs and paid over 137 billion dollars in total wages in 2009 (Pangarker and Smit, 2013). Due to its large impact, the movie industry is an essential field to explore and study.

Forecasting box-office revenues (BORs) of a particular movie has attracted many scholars because this prediction is a difficult and challenging problem. To some analysts, “Hollywood is the land of hunch and the wild guess” (Litman and Ahn, 1998). To others, “There are no formulas for success in Hollywood” (De Vani and Walls, 1999). These ideas are mostly related to the big uncertainty of audience response to the movie before its release. Jack Valenti, president and CEO of the Motion Picture Association of America (MPAA), once mentioned that “...No one, can tell you what a movie is going to do in the marketplace ...Not until that film opens in a darkened theater, and sparks fly up between the screen and the audience can you say this film is right” (Valenti, 1978).

Often, the movie industry leaves people with an impression of a lucrative field. The images of celebrities with fancy cars and the gross revenues measured in hundreds of million dollars contribute to this impression. However, most people only pay attention to the most successful movies, which do generally make quite some profit, yet in general, this impression is not true. Vogel (2010, p. 71), mentioned that “...of
any ten major theatrical films produced, on the average, six or seven may be broadly characterized as unprofitable and one might break even . . .”. These numbers suggest that the movie industry is one of the riskiest markets in the entertainment industry, which justifies the high return rates of the successful movies. It is because of these high risks in producing movies that making an adequate budget plan and accurately predicting the revenues become very important.

1.2 Previous Research

Presumably the most important aspect of the research in the movie industry is forecasting. Forecasting BORs of a new movie is a very popular task. Scientists tried various statistical and non–statistical methods to find a better estimation of BORs. Litman (1983) was the first to develop a multiple regression model in an attempt to predict the financial success of films. Independent variables such as movie genre (science fiction, drama, action-adventure, comedy, and musical), critics’ ratings, MPAA rating (G, PG, R, and X), superstar in the cast, production costs, release company (major or independent), Academy Awards (nominations and winning in a major category), and release date (Christmas, Memorial Day, summer) were used. Litman’s model showed evidence that the variables of production costs, critics’ ratings, science fiction genre, major distributor, Christmas release, Academy Award nomination, and winning an Academy Award are all significant determinants of the success of a theatrical movie.

De Vani and Walls (1999) modeled BORs using Pareto and Lévy distributions and checked whether a movie star has any effect on the BORs. They did not find any star effect and concluded that the movie is the real star. Some researchers tried to forecast BORs of new motion pictures based on early box office data. Neelamegham and Chintagunta (1999) constructed a Bayesian model which predicted BORs across
different countries. Sharda and Delen (2006) showed that the neural networks have a better prediction rate than traditional statistical classification methods, such as discriminant analysis, multiple logistic regression, and classification and regression trees (CART). Delen et al. (2007) described a Web-based decision support system to help Hollywood managers make better decisions on important movie characteristics, such as genre, super stars, technical effects, release time, etc.

Research on predicting BORs is not limited to Hollywood movies. Some articles were published trying to predict the BORs for the Korean and Chinese movie industry. Lee and Chang (2009) predicted the BORs for the Korean movie industry using Bayesian belief network (BNN). They stated that BNNs improved the forecasting accuracy compared to artificial neural networks and decision trees. Zhang et al. (2009) used back propagation neural networks to estimate Chinese BORs. Song and Han (2013) focused on predicting the BORs for the Korean movie industry using techniques such as ordinary stepwise regression, random forests and gradient boosting.

Non-traditional methods such as extreme value theory were used to model the tails of the distribution for weekend box office returns (Bi and Giles, 2009).

1.3 James Bond Movies

All of the articles discussed in Section 1.2 were focused on movies with different genres, actors, MPAA ratings, movie directors, etc. But, movie series have very similar characteristics. Because of this, predicting the BORs for movie series will require different input variables than the ones discussed in the articles in Section 1.2. A perfect example of such a movie series to examine is the James Bond (JB) movie series. This series is based on Ian Fleming’s 14 spy stories published from 1953 to 1966. The first JB movie, Dr. No, was released in 1963 which became a blockbuster soon after the release date.
Up to now, producers created movies for all of Ian Fleming spy stories. Additionally, nine other JB movies were created that were not based on those spy stories. In this Master’s report, the findings are based on the first 22 JB movies (not including *Skyfall*) because the data were collected before the release date of *Skyfall*. There are rumors about a 24th JB movie, *Bond 24*, which supposedly will be released in November 2015. These 23 JB movies became one of the longest running and highest grossing franchises ever produced (see Table 1).

### 1.4 Previous Research: James Bond Movies

The James Bond movies and books are a research topic for scientists from different fields. The areas of research range from marketing to health care, from political science to statistics. Baimbridge (1997) used ordinary least squares (OLS) for predicting the BORs for JB franchises. Johnson et al. (2013) talked about the alcohol consumption of James Bond and the possible health consequences that could happen later. Marketing research done by Cooper et al. (2010) tried to understand the psychology of James Bond movie fans. In particular, this paper discussed the meaning of champagne and car brands and the possible influence on movie fans.

Some scientists examined the violence in the movie industry over time. For example, by analyzing JB movies, McAnally et al. (2013) hypothesized that popular movies are becoming more violent. Parallel to this MS report an article about the JB movie series was published in the Chance magazine (Derek, 2014). This article presented some visual techniques for variables kills, conquests, martinis and box–office revenues, which is the main goal of the second chapter in this MS report. Additionally, Chapter 2 will provide much more visualization techniques than in Derek (2014). The

---

1 This research only follows the “official” releases through Metro-Goldwyn-Mayer (MGM) and leaves out the other JB movies such as *Casino Royale* (1954), *Casino Royale* (1967), and *Never Say Never Again* (1983), released by CBS, Columbia Pictures, and Warner Brothers, respectively.
<table>
<thead>
<tr>
<th>#</th>
<th>Title</th>
<th>Year</th>
<th>JB Actor</th>
<th>BOR</th>
<th>BOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Dr. No</td>
<td>1963</td>
<td>Connery</td>
<td>16.07</td>
<td>157.86</td>
</tr>
<tr>
<td>2</td>
<td>From Russia, with Love</td>
<td>1964</td>
<td>Connery</td>
<td>24.80</td>
<td>222.67</td>
</tr>
<tr>
<td>3</td>
<td>Goldfinger</td>
<td>1964</td>
<td>Connery</td>
<td>51.08</td>
<td>458.62</td>
</tr>
<tr>
<td>4</td>
<td>Thunderball</td>
<td>1965</td>
<td>Connery</td>
<td>63.60</td>
<td>525.80</td>
</tr>
<tr>
<td>5</td>
<td>You Only Live Twice</td>
<td>1967</td>
<td>Connery</td>
<td>43.08</td>
<td>299.77</td>
</tr>
<tr>
<td>6</td>
<td>On Her Majesty’s Secret Service</td>
<td>1969</td>
<td>Lazenby</td>
<td>22.77</td>
<td>133.89</td>
</tr>
<tr>
<td>7</td>
<td>Diamonds Are Forever</td>
<td>1971</td>
<td>Connery</td>
<td>43.82</td>
<td>221.76</td>
</tr>
<tr>
<td>8</td>
<td>Live and Let Die</td>
<td>1973</td>
<td>Moore</td>
<td>35.38</td>
<td>166.91</td>
</tr>
<tr>
<td>9</td>
<td>The Man with the Golden Gun</td>
<td>1974</td>
<td>Moore</td>
<td>20.97</td>
<td>93.64</td>
</tr>
<tr>
<td>10</td>
<td>The Spy Who Loved Me</td>
<td>1977</td>
<td>Moore</td>
<td>46.84</td>
<td>175.39</td>
</tr>
<tr>
<td>11</td>
<td>Moonraker</td>
<td>1979</td>
<td>Moore</td>
<td>70.31</td>
<td>233.90</td>
</tr>
<tr>
<td>12</td>
<td>For Your Eyes Only</td>
<td>1981</td>
<td>Moore</td>
<td>54.81</td>
<td>164.63</td>
</tr>
<tr>
<td>13</td>
<td>Octopussy</td>
<td>1985</td>
<td>Moore</td>
<td>67.89</td>
<td>179.96</td>
</tr>
<tr>
<td>14</td>
<td>A View to a Kill</td>
<td>1987</td>
<td>Moore</td>
<td>50.33</td>
<td>118.38</td>
</tr>
<tr>
<td>15</td>
<td>The Living Daylights</td>
<td>1987</td>
<td>Dalton</td>
<td>51.19</td>
<td>109.32</td>
</tr>
<tr>
<td>16</td>
<td>License to Kill</td>
<td>1989</td>
<td>Dalton</td>
<td>34.67</td>
<td>72.92</td>
</tr>
<tr>
<td>17</td>
<td>GoldenEye</td>
<td>1995</td>
<td>Brosnan</td>
<td>106.43</td>
<td>204.30</td>
</tr>
<tr>
<td>18</td>
<td>Tomorrow Never Dies</td>
<td>1997</td>
<td>Brosnan</td>
<td>125.30</td>
<td>227.94</td>
</tr>
<tr>
<td>19</td>
<td>The World Is Not Enough</td>
<td>1999</td>
<td>Brosnan</td>
<td>126.94</td>
<td>208.65</td>
</tr>
<tr>
<td>20</td>
<td>Die Another Day</td>
<td>2002</td>
<td>Brosnan</td>
<td>160.94</td>
<td>231.30</td>
</tr>
<tr>
<td>21</td>
<td>Casino Royale</td>
<td>2006</td>
<td>Craig</td>
<td>167.45</td>
<td>213.47</td>
</tr>
<tr>
<td>22</td>
<td>Quantum of Solace</td>
<td>2008</td>
<td>Craig</td>
<td>168.37</td>
<td>195.81</td>
</tr>
<tr>
<td>23</td>
<td>Skyfall</td>
<td>2012</td>
<td>Craig</td>
<td>304.36</td>
<td>319.27</td>
</tr>
</tbody>
</table>

Table 1: Summary table of James Bond movies. The values of BORs are in millions of dollars. Inflation adjustment year is 2014.
article [The Economist (2012)] in *The Economist* summarized the average number of kills, conquests, and martinis drunk by the six different JB actors in the first 22 JB movies. This article was the initial motivation for this Master’s report.

### 1.5 Data for James Bond Movies

#### 1.5.1 Data Sources

Probably the most important variable for examining JB movies is the response variable (US box-office revenues). This variable was collected from the Box Office Mojo [http://www.boxofficemojo.com/franchises/chart/?id=jamesbond.htm](http://www.boxofficemojo.com/franchises/chart/?id=jamesbond.htm). This website also has the inflation adjusted US box-office revenues (IAUSBOR). Two measurements of IAUSBOR were used from the Box-office Mojo website. The first one (IAUSBOR₁) was based on the webpage [http://www.boxofficemojo.com/franchises/chart/?id=jamesbond.htm](http://www.boxofficemojo.com/franchises/chart/?id=jamesbond.htm) and the second one (IAUSBOR₂) was based on the average ticket price [http://boxofficemojo.com/about/adjuster.htm](http://boxofficemojo.com/about/adjuster.htm).

Two measurements of the inflation adjuster were collected from the National Association of Theatre Owners (NATO) [http://natoonline.org/data/ticket-price/](http://natoonline.org/data/ticket-price/) and the Box–office Mojo [http://boxofficemojo.com/about/adjuster.htm](http://boxofficemojo.com/about/adjuster.htm). The numerical values of these two adjusters were positively associated and have a Pearson correlation coefficient, $r = 0.999$. Using the inflation adjuster from Box–office Mojo, the measurements of IAUSBOR are almost identical ($r = 0.99$) with the IAUSBOR at Box–office Mojo website, except for the two most successful JB movies (*Thunderball* and *Goldfinger*). Choosing the adjustment year of 2008, these two measurements gave about $100$ million difference for these two JB movies.

The consumption price index (CPI) was used to calculate the IAUSBOR. The CPI index was collected from the Bureau of Labor Statistics [Crawford and Church](http://www.bls.gov).
Using the CPI index, \( \text{IAUSBOR}_3 \) was calculated. In this research, all three measurements of the IAUSBOR will be used for the analysis in Chapter 3.

The variable PCEMOVIES was extracted from the Federal Reserve Economic Data (FRED, <http://research.stlouisfed.org/fred2/series/DLIGRG3A086NBEA>). TOTADM and RELEASES were found in the following websites: <http://www.waynesthisandthat.com/moviedata.html> and <http://www.filmsonsoff8.com/censorship/mpaa-film-numbers-52000.html>. All these variables will also be used in Chapter 3.

JB is famous for visiting different countries when accomplishing the assigned tasks. The list of countries visited by JB in a movie was found on a Wikipedia webpage (<http://en.wikipedia.org/wiki/List_of_James_Bond_film_locations>) and was verified through the <http://www.sporcle.com/games/PumpkinBomb/bondgeography> webpage. The countries JB visited in movies are not necessarily the ones where the filming took place. Two countries in this list, Republic of Isthmus and San Monique, are fictional countries and, thus, were not included into the dataset.

### 1.5.2 Explanatory Variables: The Economist and Baimbridge Models

*The Economist* article [<cite>The Economist</cite> (2012)](http://www.bls.gov/cpi/cpid1402.pdf) summarized the average number of kills, conquests, and martinis drunk per movie by all JB actors. This article didn’t provide any information about these variables for each JB movie. Fortunately, *The Economist* editor was very kind to share the data they have used for their article. That dataset contained the number of kills, conquests, and martinis for each JB movie. Additionally, it listed the number of “Bond, James Bond” (BJB) expressions per movie.
Baimbridge (1997) discussed four regression models using OLS to predict log-BORs. This paper was published in 1997, so finding the exact data used in this paper was almost impossible. Thus, instead of trying to find the exact data, the attempt was made to replicate his four models was performed using the information given in his paper. In the first model, the author used dummy variables for each JB actor. Another dummy variable, NEWBOND, indicated whether a new JB actor had appeared. The last two variables of this model were ACTREND and ACTRENSQ. These variables show the number of appearances and the square of the number of appearances, respectively, per JB actor.

The second model is described by nominations and ratings. Dummy variables for Oscar nomination (MONOSCAR) and Oscar won (WONOSCAR) were created for this model. Three other dummy variables (ONESTAR, TWOSTAR, THREESTAR) were created showing the rating of the movies (Halliwell, 1989).

In the third model, variables SEQUENCE, GAP, GAPSQ and COLDWAR were used. SEQUENCE represented the time order of the movies. The time period of each subsequent Bond movie (GAP) was entered as a quadratic function. COLDWAR was a dummy variable showing the end of the Cold War in 1989.

The last model used the following variables: deflated average ticket price (PRICE), deflated aggregate personal consumption expenditure on movies (PCEMOVIES), total number of US admissions (TOTADM), and number of releases measured by the MPAA (REALEASES). PRICESQ and PCEMOVIESSQ were the square of variables PRICE and PCEMOVIES. CPI index (Crawford and Church, 2014) was used to deflate the variables PRICE and PCEMOVIES.

1.6 Objectives

The research in this Master’s report is divided into three main parts. In Chapter
graphical summaries of the variables used in Section 1.5.2 will be given. In addition, the response variable BOR will be compared with possible explanatory variables kills, conquests, martinis, and BJB expression. To present these graphical summaries, many visualization techniques such as time series plots, dotplots, histograms, scatterplots, parallel coordinates plots, heatmaps, mosaic plots, association plots, and choropleth maps will be displayed. Using these visualization techniques, the relationship between the explanatory variables with each other as well as with the response variable will be presented.

Chapter 3 will try to replicate the four regression models discussed in Baimbridge (1997). This paper was published in 1997 and the datasets in this paper only contained the movies released before 1990s.

Chapter 4 will examine linear regression as well as machine learning methods such as lasso and random forest for predicting BORs. For each of these methods, three datasets will be used described in Sections 1.5.2 (The Economist model, the first model, and the third model). Movies released before 1990s will be considered as the training dataset and the ones after 1990s will be used in the test set. Visual comparison will be given to compare the difference between these methods.

In Chapter 5, we will summarize the findings and suggest which model and method to use.

The appendix A will include all the datasets used in the Master’s report. All the R code will be given in Appendix B.
2.1 Statistical Graphics

John Tukey introduced the term exploratory data analysis (EDA) in the late 1970s (Tukey 1977). Rather than directly starting hypothesis testing as statisticians traditionally did, he suggested to start the analysis by looking at the data first. Often, it was done by visualization methods such as histograms, boxplots, etc.

Sometimes numerical statistical summaries can be very misleading. The quartet dataset created by Anscombe (1973) showed that without visualization, completely different datasets could lead to the same numerical results. Therefore, in this Master’s report, various visualization methods were applied to data related to the James Bond (JB) movies.

All graphical results and statistical analysis were conducted in R (R Core Team 2013). Sweave (Leisch 2002) was used for documentation in order to make the results of this Master’s report fully reproducible.

2.2 Time Series Plots

In this section, time series plots (Figures 1-4) are presented to show the trend of the variables kills, conquests, martinis, and BJB expressions with respect to time, discussed in Section 1.5.2. In each of these plots, six symbols and colors are used to distinguish all six JB actors. Additionally, these plots show the linear regression line, a lowess smoother (with parameters $f = 0.5$, $\text{iter} = 3$) (Cleveland 1979), and a moving averages smoother (with parameters $q = 5$, $p_1 = \cdots = p_q = 0.2$). These smoothers and the regression line will help to see if there are some trends with these
variables over time. Each smoother and the line is given with a distinct color. Each of
these graphs has two legends, which clarify the symbol and color differences between
the six JB actors and the color difference of the smoothers and regression line.

2.2.1 Kills

Figure 1 shows the number of JB kills over time. McAnally et al. (2013) suggested
that the violence in James Bond movies has increased over time. They defined violence
as “any scene in which there was an intentional attempt by any individual to harm
another”. The figure showing the violence increase in McAnally et al. (2013) and
Figure 1 have very similar trend and hence, they have positive correlation coefficient.

In Figure 1 the regression line and the smoother suggests some positive rela-
tionship between JB kills and time ($p = 0.074$). Also, Table 2 shows a weak positive
association between JB kills and time. However, a closer look shows that JB, when
played by Brosnan, killed far more people than when played by any other JB actor.
|                      | (Intercept) | Estimate | Std. Error | t value | Pr(>|t|) |
|----------------------|-------------|----------|------------|---------|---------|
| Kills                | -319.21     | 0.16589  | 0.08806    | 1.8838  | 0.07421 |
| Kills (without Brosnan) | 2.9407     | 0.00233  | 0.08081    | 0.0289  | 0.97733 |
| Conquests            | 41.808      | -0.0199  | 0.01126    | -1.767  | 0.09253 |
| Martinis             | -119.62     | 0.06094  | 0.01729    | 3.5240  | 0.00213 |
| Martinis (without Craig) | -35.666   | 0.01842  | 0.01224    | 1.5051  | 0.14964 |
| Martinis (with Skyfall) | -86.884   | 0.04437  | 0.01741    | 2.5491  | 0.01868 |
| BJB                  | -9.006      | 0.00507  | 0.00995    | 0.5093  | 0.61613 |

Table 2: OLS summary results of kills, conquests, martinis, and BJB over time.

Ignoring Brosnan’s performance, the JB kills and time do not seem to be positively correlated.

Table 2 shows that the weak association disappears when the linear regression excludes Brosnan’s observations. Additionally, a negative association between JB kills and time can be observed when ignoring the observations before the JB actor Brosnan. Maybe a more appropriate conclusion in this case would be: the amount of violence during the JB movies played by Brosnan leads to the impression that the violence is increasing over time.

### 2.2.2 Conquests

Figure 2 shows the number of JB conquests over time. The regression line, lowess smoother, and moving averages smoother suggest some negative relationship between conquests and time. Table 2 suggests that every year the average number of conquests is decreasing by 0.02. This is only supported by weak evidence, with a p-value of 0.093.

### 2.2.3 Martinis

This Std. Error is the standard error for the coefficient of release data and is not the standard error for the intercept coefficient.
Fig. 2: Number of Bond conquests per movie over time.

Fig. 3: Number of martinis drunk by Bond per movie over time.
Figure 3 shows the number of martinis drunk over time. The smoothers in this Figure have a shape of convex parabola. It shows that the martini consumption reached its minimum in the 1970s and started to increase afterwards. Here the picture would not be so vivid if we had ignored JB actor Craig. He drank four and six martinis during the movies *Casino Royale* and *Quantum of Solace*. The average of five martinis drunk for JB actor played by Craig is far above the number of martinis drunk by the other JB actors.

The regression line in Figure 3 shows a positive relationship between martinis and time. The p-value \((p = 0.002)\) for martinis in Table 2 suggests a highly significant linear relationship as well. In the last JB movie, *Skyfall* (which is not included in the dataset), there are no martinis drunk by JB (Thomas, 2012). The linear regression model between martinis and time would still give a significant association with a p-value of 0.019, even if the martini value of zero would be used as the 23th observation for the year 2012.

Table 2 shows that after ignoring the martinis drunk played by JB actor Craig gives a non significant linear association between martinis and time \((p = 0.15)\). Similar to Section 2.2.1, more appropriate conclusion of this section would be: Craig leads to the impression that the number of martinis drunk by JB actors are increasing over time.

### 2.2.4 Bond, James Bond (BJB)

Figure 3 shows the number of BJB expressions over time. In this figure, the opposite pattern can be seen, compared to Figure 2. The smoothers have a shape of concave parabola. In other words, the BJB expressions was not popular in 1960s and 2000s and achieved its peak in the 1970s and early 1980s. The regression line suggests a small increase over time. However, the p-value in Table 2 \((p = 0.62)\) suggests that
there is no linear relationship between BJB expressions and time.

Using only the regression results in Table 2, the conclusion would be that three out of the four variables discussed in this section have some association with time. However, distinguishing the JB actors revealed that JB actor Brosnan seems to be the major cause for the increased number of JB kills over time. Similarly, JB actor Craig might be the reason for increasing number of martinis over time.

2.3 Scatterplot Matrix

A scatterplot matrix is a useful tool to present multivariate data. For the given n variables, a scatterplot matrix contains a scatterplot for all pairs of variables. Plotting all scatterplots next to each other could be beneficial for checking the linear and non–linear relationships between all pairs of variables. In this section, a scatterplot matrix is constructed for the inflation adjusted BORs, JB kills, conquests, martinis, and BJB expressions.
Figure 5 shows the scatterplot matrix for these five variables. Using the average ticket price, the 2014 inflation adjusted BOR is shown in top left corner. The variables kills, conquests, martinis and BJB are plotted on diagonal panels (from the second row to the fifth). These variables have mostly integer values, and thus, a lot of overplotting occurs. In order to avoid this overfitting, a small randomness, called jitter was added to the explanatory variables. For all pairs of scatterplots, lowess smoothing function (parameters: \( f = 2/3, \text{iter} = 3 \)) is plotted in purple.

Colors and symbols are used to distinguish the JB actors. These colors and symbols are consistent with the time series plots in Figures 4–4. Histograms are shown in the diagonal panels, showing the distributions of all variables. A rug plot, which simply draws a tick for each value, was added to each histogram to provide more information about each observation.

Figure 5 shows some positive relationship between JB kills and BOR and some negative association between BOR and BJB. A weak negative association can be observed between martinis and conquests in the JB movies.

### 2.4 Dot Plots

Several dot plots were produced to show some simple statistical averages of the JB actors. All dotplots are ordered highest (top) to the lowest (bottom). Figure 6(a) shows the number of JB movies produced by each JB actor. Connery and Moore were the most popular JB actors with 6 and 7 movies, respectively.

Figure 6(b) suggests that Connery is the most successful JB actor in terms of inflation adjusted BOR. Here the 2014 was used for inflation adjustment year and average ticket price was used as an adjustment method. The second and third successful actors are Brosnan and Craig. The order of Brosnan and Craig will change when the BOR of the last JB movie, *Skyfall*, will be included.
Fig. 6: Summary of averages by actor.
Figure 6(c) shows that JB actor Brosnan is the most violent actor by killing twice as many people in JB movies as the second most violent JB actor, Connery. Figure 6(d) implies that JB actor, played by Lazenby, has the most conquests. However, this is based only on one observation (movie). According to Figure 6(e), JB, when played by Craig, is the biggest martini drinker with an average of 5 martinis per movie. However, JB, when played by Craig, switches from martinis to beers during the most recent JB movie, *Skyfall* (Thomas, 2012). JB, played by Dalton is the second most martini drinker with less than 1.5 martinis on average. Figure 6(f) shows that the most “Bond, James Bond” expression user was Lazenby. Similar to 6(d), this is also based only on one observation (movie).

### 2.5 Box Plots

Similar to Figure 6, the 2014 inflation adjusted BOR using average ticket price as an adjustment were examined. Figure 7 shows boxplots of kills, conquests, martinis, and BJB. Each of these variables are divided into three categories. For example, the number of kills consists of the categories 0–5 kills, 5–10 kills, and more than 10 kills. All box plots were ordered from the highest to the lowest median BOR.

Figure 7(a) shows that decrease in number of kills is associated with decease in BOR. Similarly, in Figure 7(d) when the number of BJB is increasing, the BORs seem to decrease. These two relationships found in Figure 7(a), 7(d) are consistent with the results shown in scatterplot matrix in Figure 5. Even though two of the most successful JB movies, *Thunderball* and *Goldfinger*, have two and more conquests, there exists a slight negative relationship between BOR and conquests. There is no obvious relationship between BOR and martinis.

Figure 8 shows the distribution of revenues by actor. JB actor Connery has more variability than any other actor. He also has the highest BORs. JB actors Lazenby
Fig. 7: BORs, with respect to high, medium and small number of kills, conquests, martinis and BJB, sorted by median BOR within each category.
and Dalton have the lowest median BORs and the lowest number of JB movies. In this dataset, JB actor Craig has the same number of movies as Dalton. However, this dataset does not include the latest JB movie *Skyfall* and the possible future JB movie *Bond 24* where Craig will be most likely the JB actor.

Fig. 8: Box plots, showing the average inflation adjusted BOR by JB actor, sorted by median BOR within each JB actor.

### 2.6 Histogram and Normal QQ Plot

Figure 9 consists of four graphs. Figure 9(a) and Figure 9(c) show the original and log–transformed histograms of BOR. A rug plot is added to each of these histogram plots. All BORs are deflated for the year of 1962 using the average ticket price adjustment. Figure 9(b) and Figure 9(d) show the normal quantile plots of the original and log–transformed BOR. Here the log–transformation and the deflation adjustment year of 1962 were chosen because these transformations will be used frequently in the next chapter.

Figure 9(a) shows that two observations have much higher BOR than the other
Fig. 9: Histogram and normal QQ plot for box–office and log box–office revenues.
observations. These two observations represent the movies Thunderball and Goldfinger. Even after the log–transformation, these two observations are distinctly apart from the rest of the data. The QQ plots in Figure 9(b) and Figure 9(d) show that neither the original nor the transformed BOR are close to being normally distributed.

2.7 Parallel Coordinates Plots

Similar to a scatterplot matrix, the parallel coordinates plot is also a common method to present multivariate data. In order to show the multivariate data, parallel coordinates plot sacrifices the orthogonal axis by drawing axis parallel to each other. Each multivariate data point is presented by the continuous line which is simply a connection of all neighboring axis. The relationship of non–neighboring variables becomes harder to see as the gap between these variables becomes larger. The gap in this context is the number between two variables of interest. Positive linear rela-
tionship between two neighboring variables can be observed if the connection lines of observation are parallel. If the connection lines of observations mostly cross, this is an indicator of a negative association. The scale of each parallel axis does not necessarily need be the same. It can have a common scale or individual scales varying from the minimum to the maximum of that particular variable.

Figure 10 shows the parallel coordinates plot for kills, conquests, martinis and BJB variables. Similar to boxplots in Section 2.5, these variables were divided into three categories. Distinct colors were chosen to distinguish the categories of kills variable. In Figure 10, the connection lines between conquests and martinis seem to have a lot of crossing. This means that possible negative association between conquests and martinis can be observed. The same pattern can be seen in the scatterplot matrix (Figure 5). Many interactions between the variables martinis and BJB also suggest a negative association between these variables. This is also consistent with the fourth bottom panel in Figure 5. In Figure 10, the conquest variable lies between the variables kills and martinis meaning that it is hard to examine any relationship between these variables.

2.8 Heatmaps

Heatmap is a good graphical method to visualize a matrix of numbers. These numbers can be ordered using various clustering techniques. Dendrograms are used to provide more information about clusters. After the cluster analysis, the heatmap plot uses colors to represent numbers.

Figure 11 shows a heatmap plot for the variables JB kills, conquests, martinis and BJB expression, which are presented in the columns. The rows show the JB actors’ names, followed by the release dates and movie names. The values represented by the colors are described in the upper left corner of this figure. That corner also shows
Fig. 11: Heatmap plot of kills, conquests, martinis, and BJB expression by actor name and movie release date. The histogram on the top left panel shows the distribution of the data matrix.
Fig. 12: Heatmap plot of square–root transformed kills, conquests, martinis, and BJB expression by actor name and movie release date. The histogram on the top left panel shows the distribution of the data matrix.
the histogram of the data matrix in cyan. To create the dendrograms, hierarchical clustering was implemented using Euclidean distance.

The top part of Figure 11 shows clustering for JB actor Brosnan. This cluster contains all his movies except the *Goldeneye*. The dendrogram on the left shows that the movie *Goldeneye* does not belong to any cluster group. The cluster of JB actor Brosnan is mainly due to the variable kills.

Additionally, two separated clusters can be observed for JB actor Moore. The cluster on top side including the movies *The Man with the Golden Gun*, *A View to a Kill*, *Live and Let Die* has common low number of kills and low number of martinis. The cluster on bottom for the movies *Moonraker*, *The Spy Who Loved Me* and *Octopussy* has a medium number of kills, martinis, and BJB expressions. For the latter movies, there is also a time cluster, because all these three movies were released consequently in 1977, 1979 and 1983.

In contrast, there is no cluster for JB actor Connery. Not even two of the Connery’s movies are clustered together in Figure 11, which means that all of his movies have distinct characteristics. Earlier, Figure 8 showed that JB actor Connery is most successful in term of BOR, and maybe his different appearance in each movie is one of the secrets of this success.

Figure 11 also shows that the numerical values of the variable kills are much higher than the values of conquests, martinis and BJB expression. This can be observed from the top dendrogram as the variable kills is isolated. Due to these high values, the variable kills could have a dramatic effect on clustering. To reduce the effect of this variable, a square–root transformation is applied. Specifically, the upper left panel in Figure 11 shows that the variable kills vary between 4 and 25 meaning that it will take values between 2 and 5 after the square–root transformation. This new range is very similar to the range of other variables, and, hence will
reduce the effect of kills.

Figure 12 shows a heatmap plot for the variables square-root kills, conquests, martinis and BJB expression. After the transformation, more JB actor clusters can be observed. The movies *The Living Daylights* and *Licence to Kill* played by JB actor Dalton can be observed on top of this figure. Similarly, a cluster for JB actor Craig can be observed on the bottom. Similar to Figure 11 two clusters can be observed for the JB actor Moore. Even though four movies by Connery are next to each other, less clustering is observed from the dendrogram. The result found in Figure 11 does not hold for Figure 12 after the transformation, however the “isolated” movie *GoldenEye* is clustered with *Die Another Day*. The movies *Tomorrow Never Dies* and *The World is Not Enough* does not appear in the same cluster either.

2.9 Mosaic Plot

A mosaic plot (Hartigan and Kleiner 1984) is a popular visualization method to present categorical data. For the categorical data given in the two-way contingency table, the mosaic plot creates rectangles with proportional horizontal and vertical slices. The area of the rectangles is proportional to the corresponding frequency number in the contingency table. Friendly (1994) generalizes the mosaic plots from two-way to multi-way contingency table.

A mosaic plot using a four-way contingency table is shown in Figure 13. This figure uses variables BJB expression (first vertical division) kills (first horizontal division), conquests (second vertical division), and martinis (second horizontal division). The vertical bar line on the right shows standardized Pearson residuals for the given color. Note that the standardized Pearson residuals is not the only option for the vertical bar line and hence, other independence hypothesis can be tested. The p-value under the vertical bar is 0.0277 which suggest some association between the variables.
Fig. 13: Mosaic plot for kills, conquests, martinis and BJB expression.
kills, conquests, martinis and BJB at 5% significance level.

A four-way contingency table was used in this mosaic plot. Each variable consists of three categories which makes $3^4 = 81$ possible combinations. However, there are only 22 observations (movies) in the dataset meaning that most combinations will not appear in the mosaic plot. For these observations the mosaic plot will draw vertical or horizontal lines. The largest rectangle contains four observations. The area of the widest rectangle is twice less than the area of largest rectangle, and hence the widest rectangle contains two observations. The area of other rectangles are twice smaller than the area of the widest rectangle meaning that these small rectangles have only one observation.

2.10 Association Plot

The association plot (Cohen, 1980) is a useful tool to visually check the independence of several categorical variables. Meyer et al. (2006) describes the association plot as the following: “an association plot visualizes the standardized deviations of observed frequencies from those expected under a certain independence hypothesis. Each cell is represented by a rectangle that has (signed) height proportional to the residual and width proportional to the square root of the expected counts, so that the area of the box is proportional to the difference in observed and expected frequencies.”

Similar to mosaic plot in Figure 13, the vertical bar line on the right side of Figure 14 shows standardized Pearson residuals for the given color. The p-value ($p = 0.0277$) under the vertical bar suggests some association between these variables at the 5% significance level. Figure 14 also shows a very high residual on the upper left corner of the graph. This could be a possible reason of the highly significant p-value observed under the vertical bar line.
Fig. 14: Association plot for kills, conquests, martinis, and BJB expression.
2.11 Choropleth Maps

Choropleth maps assign colors and shades to the individual areas in the map. Colors correspond to a pre-defined values or a range of values. In each movie James Bonds visited several countries and visiting exotic countries became another characteristic of the JB movie series. To determine whether countries have any effect on the BOR, three choropleth maps have been created. The world map changed significantly after the collapse of the Union of Soviet Socialist Republics (USSR). Countries like Yugoslavia, Czechoslovakia and the USSR split into 19 independent countries during 1990s. Therefore, creating a single choropleth map would be problematic as the first JB movie was released in 1963. To create meaningful maps with countries that correctly show the borders at the time the movie was released, two maps were created showing the Bond visits before and after collapse of the USSR.

Figure 15 shows the number of visits to different countries in the JB movie series before the collapse of the USSR. These visits do not necessarily include all the countries that the movies were filmed at. For example, in the Die Another Day movie JB visits North Korea, but the filming did not take place in North Korea. Furthermore, European and Caribbean counties are hard to see in the world map. Therefore, the zoomed-in choropleth maps for European and Caribbean countries are displayed in the bottom panels of Figure 15.

Figure 16 shows the frequency of JB visited countries after collapse of the USSR. Figures 15 and 16 show that United States and European countries are the most popular for Bond visits. Hong Kong is another popular country, but it is not visible on these maps because of its small area. African and South American countries, Canada, and Australia are the least popular countries for JB visits.

Figure 17 shows the average BORs across the countries before and after the collapse of USSR. Here 16 movies were released before the collapse of the USSR and
Fig. 15: Number of Bond visits before the collapse of the USSR.
Fig. 16: Number of Bond visits after the collapse of the USSR.
Fig. 17: Average BOR (in millions) by country before (top panel) and after (bottom panel) the collapse of the USSR.
six movies were released after the collapse of the USSR. Except for Japan, the average BOR in Asian countries varies between $80 million to $200 million. The average BOR is mostly higher in European, and South and North American countries compared to Asian countries. The BOR seems quite evenly spread after the collapse of the USSR.

2.12 Summary

This chapter presented various visual tools to better understand the data presented in *The Economist* magazine ([The Economist, 2012](#)). In Section 2.2, the regression line showed an increasing trend of kills and martinis over time. Careful look in Figure 1 and Figure 3 showed that JB actors Brosnan and Craig could lead that increasing trend.

Section 2.3 presented a scatterplot matrix which indicated some positive relationship between BOR and kills, and some negative relationship between BOR and BJB expression. The same pattern was observed via boxplots in Section 2.5. The scatterplot matrix also showed some negative relationship between variables martinis and conquests, and between the variables martinis and BJB expression. Similar conclusion was made by using the parallel coordinates plot in Section 2.7.

Section 2.4 showed that the most violent JB actor was Brosnan with almost 20 kills per movie on average. Craig drunk on average five martinis per movie which was at least three times higher than the next JB martini drinker. As shown in 2.2 high numbers like this characterizing JB actors may change the trend of the particular variables over time.

Section 2.6 showed that neither the inflation adjusted BOR nor log–transformation of it are close to normal distribution. In particular, the two most successful movies, *Thunderball* and *Goldfinger*, can be classified as big outliers under the assumption of
normality.

Section 2.8 presented a heatmap plots where at first clusters for JB actor Brosnan were examined. This was due to the impact of kills variable as Brosnan was the most violent one when played as JB actor (Sections 2.4 and 2.2.1). The square-root transformation of kills variable reduces the impact of it which vanishing the Brosnan’s cluster. The two separated clusters for JB actor Moore stayed consistent in both of the heatmap plots. Additionally, new clusters for JB actors Danton and Craig appeared after decreasing the impact of the kills variable.

The mosaic plot in Section 2.9 showed that there is some association between the variables kills, conquests, martinis and BJB expressions. The same conclusion was derived from the association plot in Section 2.10. The choropleth maps showed that the BORs before the collapse of the USSR are slightly higher in Europe, South and North Americas than the BORs in Asia. Finally, the JB movies were more popular when JB visited the Unites States and Europe, compared to JB movies where he visited other countries.
CHAPTER 3

REPLICATION OF BAIMBRIDGE’S MODEL

3.1 Reproducible Research (RR)

During the last decade, replication of scientific findings became an important part of research. Research is often presented in condensed formats such as journal articles and slideshows where findings could be extremely hard to check and extend. For example, one difficulty can arise while trying to access the data. Without specific information about the data and its sources, the replication of scientific findings becomes a hard task. Even a small difference in the data and its transformations may cause a different output and, hence, a different conclusion.

Another difficulty in RR is the limited access to code written in various programming languages. Publicly available code makes it easier for peers to be involved in the field and to extend previous ideas. Additionally, Stodden et al. (2013) demonstrated the importance of reproducible research in computational research. Stodden (2011) challenged the researchers to share their data and code if they are confident in their research results. Fortunately, as the researchers’ awareness of RR rises, the percentage of publically available code and data are increasing over time (Stodden, 2013).

RR is an important component in this MS report as well. This chapter reproduces the results presented in Baimbridge (1997). Due to the absence of the original data, various websites were used to recollect the data. Sometimes, more than one source was found for the same variable (e.g., box-office revenue) with different outcomes values. In that case, the values from different sources should be compared and discussed
separately, although it would be advantageous if Baimbridge (1997) had preserved more reproducibility.

Recently, reproducible research with R became more and more popular (Gandrud, 2013). Leisch (2002) introduced \textit{sweave}, which integrates R and \LaTeX, creating tables and numerical outputs from R directly into \LaTeX. In this MS report, R and R \textit{sweave} were used to make this research fully reproducible.

3.2 Replication of Baimbridge (1997)

Baimbridge (1997) presented four linear regression models, which were related to the James Bond (JB) box–office revenues (BOR). Each model used a natural log transformation of the BORs to reduce the effect of highly successful movies. Additionally, a technique defined by Cochrane and Orcutt (1949) was applied to correct the first order autocorrelation between the predictors.

The data in the first model includes dummy variables for three of the JB actors (CONNERY, LAZENBY, MOORE). In this model, JB actor Dalton was omitted to prevent the problems with perfect collinearity. Hence, the intercept coefficient will represent JB actor Dalton and the coefficients of the other JB actors’ variable will be relative to Dalton. The dummy variable NEWBOND represents the appearance of a new JB actor. ACTREND and ACTRENDSQ count the appearances of each and the square of each appearances, respectively.

The second model is related to the movie award nominations and ratings. The dummy variables NOMOSCAR and WONOSCAR show whether a particular movie was nominated or won an Oscar, respectively. The dummy variables ONESTAR, TWOSTAR and THREESTAR, correspond to the movie ratings presented in Halliwell (1989). Similar to the variable DALTON in the first model, the zero star Halliwell rating was omitted to avoid perfect collinearity. This means that the inter-
cept coefficient represents the zero star movies, and the coefficients of the ONESTAR, TWOSTAR, and THREESTAR variables show the increase in BOR relative to the zero star movies.

Fig. 18: Summary results extracted from Baimbridge (1997), Table 1.

The third model used the variables SEQUENCE, GAP, GAPSQ, and COLDWAR: SEQUENCE is the order of movie releases, GAP shows the time gap between two consecutive movies, GAPSQ is the squared value of the GAP variable and COLDWAR represents the end of the COLDWAR in 1989.

The fourth model consists of the variables PRICE (average deflated ticket price), PRICESQ, PCEMOVIES (aggregate personal consumption on movies), PCEMOVIESQ, TOTADM (total number of US movie admissions) and RELEASES (number of releases measured by the MPAA). The PRICESQ and PCEMOVIESSQ are the square
of variables PRICE and PCEMOVIES. The summary results of Baimbridge (1997) are presented in Figure 18.

### 3.2.1 First Model

In Baimbridge (1997), the descriptions of some variables were vague. For example, it was not clear if the variable ACTREND starts from zero or from one. Additionally, the value of the NEWBOND for the first (Dr. No) and the seventh (Diamonds are Forever) JB movies were not specified (0 or 1). For the first movie, Dr. No the JB actor Connery was a new JB actor, but there was not any other JB actors before. For the seventh movie, Diamonds are Forever, JB actor Connery was a new JB actor compare to the last movie, but not compared to all the other movies. Furthermore, Baimbridge (1997) specified that the Cochrane and Orcutt (1949) technique was used to correct the first order autocorrelations, but he did not mention if it was used for all four models. The inflation adjustment year is not known either. Possible inflation adjustment years are 1963 (the year when the first JB movie Dr. No was released in the United States) or 1962 (the year when the first JB movie Dr. No was released in the United Kingdom).

The inflation adjustment method was also not clearly stated in Baimbridge (1997). Common methods can be based on the Consumer Price Index (CPI) or the average ticket price. Specifically, the 1962 inflation adjusted box–office revenues using the CPI index and the average ticket price can be calculated using Equations 1 and 2, respectively.

\[
Y_{1962} = \log \left( \frac{Y_x \cdot T_{1962}}{10^6 \cdot T_x} \right) 
\]

\[
Y_{1962} = \log \left( \frac{Y_x \cdot CPI_{1962}}{10^6 \cdot CPI_x} \right) 
\]

where \(Y_x\) is the unadjusted BOR for the given year \(x\), \(T_x\) is the average ticket price.
for the given year \( x \), and \( CPI_x \) is the consumption price index for the given year \( x \). Here the BORs are in millions.

The average ticket price adjuster and the box–office mojo adjuster gave nearly identical results except for the movies *Goldfinger* and *Thunderball* as discussed in Section 1.5.1. Therefore, inflation adjustment used in the box-office mojo (http://www.boxofficemojo.com/franchises/chart/?id=jamesbond.htm) was considered as well.

To obtain closer estimate found in Baimbridge (1997), all possible combinations for the variables setting described above were considered. Two possibilities for AC-TREND (starting from 0 and starting from 1), four possibilities for the NEWBOND variable (NEWBOND\(_1\) = 0, NEWBOND\(_1\) = 1, NEWBOND\(_7\) = 0, NEWBOND\(_7\) = 1), the usage of the Cochrane and Orcutt (1949) technique (whether the technique was used or not), two inflation adjustment years (1962 and 1963), and three inflation adjustment strategies (using the CPI index, average ticket price, and box–office mojo website) yield 96 different linear models. For all 96 models, linear regression coefficients were obtained. The best model was chosen by the coefficients that had the smallest sum of squared deviations (SSE) from the Baimbridge coefficients.

Figure 19 shows the parallel coordinates plot for all 96 models. The first seven columns show the regression coefficients of these models. The last column is the SSE from the Baimbridge’s coefficients. The faded blue lines represent the models in which the Cochrane and Orcutt (1949) technique was applied. From the last column we can see that the overall SSE is greater for the models using this technique than for the ones without. Therefore, it is most likely that the Cochrane and Orcutt (1949) technique was not performed on the first model.
Fig. 19: The replication of the first model discussed in Baimbridge (1997). The parallel coordinates plot shows the original (in black) and 96 replicated models (in red and blue). Blue lines indicate the usage of the Cochrane and Orcutt technique. The dark red line shows the best model. The dashed line represents 0. Min = -0.21 and Max = 2.01 here.

For some variables in this figure, it seems that only 24 out of 96 observations are visible. The BOR ratio between two adjustment years is a constant number meaning that the difference in log-transformed BORs is a constant as well. Therefore, using a log-transformed response variable with different inflation adjustments will only change the intercept coefficient in the OLS. Thus, more lines seem to be connected between the “(Intercept)” and the CONNERY variables than between the CONNERY and the LAZENBY. The best model with the smallest SSE chosen out of the 96 models has the following parameters:

- ACTREND: Starting from 1
- NEWBOND$_1 = 0$
- NEWBOND$_7 = 1$
- Cochrane and Orcutt: Not used
- Adjustment year: 1962
- Adjustment method: CPI

Figure 20 shows the OLS output based on the parameters from the best model. It would be time consuming to numerically compare the results in this figure with those in the upper left panel in Figure 18. For that reason, visualization techniques such as dot plots will be used to simplify the comparison of the results from the original and the replicated models.

Call:
```
lm(formula = logBoxOffice ~ ., data = model1Old)
```

Residuals:
```
       Min 1Q Median 3Q Max
-0.41683 -0.27473  0.01953  0.17187  0.41446
```

Coefficients:
```
                           Estimate Std. Error   t value Pr(>|t|)
(Intercept)               1.250220   0.56302  2.2210  0.0535  
CONNERY                   0.856900   0.31784  2.6960  0.0246 *  
LAZENBY                   0.607160   0.44093  1.3770  0.2018  
MOORE                     0.456770   0.31124  1.4680  0.1763  
ACTRENDSQ                 0.093950   0.03989  2.3400  0.0429 *  
NEWBOND                   0.394270   0.30596  1.2890  0.2297  
```

---

Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

Residual standard error: 0.3493 on 9 degrees of freedom
Multiple R-squared: 0.7319, Adjusted R-squared: 0.5531
F-statistic: 4.094 on 6 and 9 DF,  p-value: 0.02923

Fig. 20: OLS summary for the first model.

Figure 21 shows the OLS coefficients with the corresponding t-values and ANOVA output of the original and the best replicated models. The vertical dashed lines show the t-values used in 95% confidence intervals of the OLS coefficients. The OLS coefficients are ordered from the absolute smallest (bottom) t-value of the replicated model to the absolute highest one (top). From Figure 21, it can also be inferred that
the numerical results of these models are not exactly the same, but they are very similar.

In the past, there have been many re-releases for JB movies (http://movieposterauthenticating.com/wordpress/james-bond/james-bond-1-sheet-1980-re-releases/). Events like this change the BOR of all JB movies, which makes the exact replication of the Baimbridge model even harder.

Fig. 21: Comparison of the first model discussed in Baimbridge (1997) and the best replicated model. The results of the replicated model are presented via red squares and the results of the original models are presented via blue circles.

However, the replicated model captures most of the variation found in Baimbridge. In both of these models, the t–values suggest that the effects of CONNERY∗ ACTREND∗ and ACTRENDSEQ∗ are significant at the 5% significance level while the variables MOORE, LAZENBY, and NEWBOND are not. The original model has a slightly higher $R^2$, adjusted $R^2$, and F–statistic than the replicated one. In the replicated model, the Durbin and Watson (1971) statistic is 2.04 with a p–value of 0.32. This

\[1\text{Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1}\]
suggests that there is not a significant evidence of serial autocorrelation, which indicates that there is no need to perform the Cochrane and Orcutt (1949) technique. This result is consistent with the best replicated model described in this section. Overall, the first model was successfully replicated. However, in all 96 replicated models, including the best models, the variable LAZENBY is underestimated.

3.2.2 Second Model

Similar to the first model, there were some ambiguities with the variables and methods used in the second model. All combinations of adjustment years (1962 and 1963), adjustment methods (CPI, ticket price, box-office mojo) and Cochrane and Orcutt (1949) technique (used, not used) were considered to obtain the best replication of the second model. For all twelve models, linear regression coefficients were obtained. The best model was chosen by the coefficients that had the smallest SSE from the Baimbridge’s coefficients.

Figure 22 shows the parallel coordinates plot for the twelve models mentioned above. The first six columns show the regression coefficients of these models, and the last column is the sum of squared deviation from Baimbridge’s model. Similar to Figure 19, the best model is marked in dark red, and Baimbridge’s model is marked in black.

In this figure, only four out twelve observations can be distinguished, excluding the Baimbridge coefficients. Since the regression coefficient using different adjustment methods are similar to each other, it may give an impression of four lines instead of twelve (the twelve lines are overlapped and only four became visible). The best model chosen out of the twelve models has the following parameters:
- Cochrane and Orcutt: Not used
- Adjustment year: 1963
- Adjustment method: Average ticket price

Fig. 22: The replication of the second model discussed in Baimbridge (1997). The parallel coordinates plot shows the original (in black) and 12 replicated models (in red and blue). Blue lines indicate the usage of the Cochrane and Orcutt technique. The dark red line shows the best model. The dashed line represents 0. Min = -0.30 and Max = 2.82 here.

However, as Baimbridge likely used the same inflation adjustment for all of his models, Figure 22 also shows the estimates based on a CPI adjustment for 1962 (orange line). The OLS output using the parameters from the best model is given in Figure 23. Figure 24 shows the OLS coefficients with the corresponding t-values and ANOVA output of the original and the best replicated models. The vertical dashed lines show the t-values used in 95% confidence intervals of the OLS coefficients. The OLS
coefficients are ordered from the absolute smallest (bottom) t–value of the replicated model to the absolute highest one (top).

```
Call:
  lm(formula = logBoxOffice ~ ., data = model2Old)

Residuals:
     Min       1Q   Median       3Q      Max
-0.44592 -0.13528 -0.06188  0.14178  0.57293

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.4504     0.1814  13.505  9.54e-08 ***
ONESTAR     -0.1453     0.2447  -0.594   0.5657
TWOSTAR     0.3950     0.2593   1.523   0.1587
THREESTAR   0.4504     0.2846   1.583   0.1446
WONOSCAR    1.0387     0.2870   3.619   0.0047 **
NOMOSCAR    0.6009     0.2447   2.456   0.0339 *

---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 . 0.1 ‘ ’ 1

Residual standard error: 0.3416 on 10 degrees of freedom
Multiple R-squared: 0.7239,    Adjusted R-squared: 0.5859
F-statistic: 5.244 on 5 and 10 DF,  p-value: 0.01272
```

Fig. 23: OLS summary for the second model.

A small variation between the replicated and the original models can be observed in Figure 24. The variable ONESTAR has a negative OLS coefficient, meaning that the average BOR of one-star movies rated by Halliwell (1989) is less than the ones with no star. Figure 22 shows that this result is consistent with Baimbridge’s second model and all twelve replicated models. Similar to the first model, the replicated findings in the second model are similar to the original model. Both models have highly significant t–values for WONOSCAR**2 and marginally significant t–values for NOMOSCAR*. Additionally, none of the Halliwell rating variables has a significant effect relative to zero star movies. In the original and the best replicated models, the

---

2Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
F–statistic, $R^2$, and adjusted $R^2$ have similar values. In the replicated model, the Durbin and Watson (1971) statistic is 1.93 with a p–value of 0.41. These numbers suggest that there is not a significant evidence of serial autocorrelation, which means there is no need to apply the Cochrane and Orcutt (1949) technique. This result is also consistent with the best replicated model described in this section. Overall, the best model out of the twelve replicated models is a good replication of Baimbridge’s second model.

Fig. 24: Comparison of the second model discussed in Baimbridge (1997) and the best replicated model. The results of the replicated model are presented via red squares and the results of the original models are presented via blue circles.

3.2.3 Third Model

In order to better replicate the third model, six different response variables were considered (two adjustment years for three adjustment methods). Additionally, the replication is done with and without Cochrane and Orcutt (1949) technique. The variable GAP was described as the time gap between two movies in Baimbridge (1997). However, he did not specify how the rounding was done for the GAP. Thus,
the GAP rounded by year and the GAP rounded by years and months were both used to replicate the third model. The original paper did not specify whether the SEQUENCE variables starts from one or zero. Therefore, two types of SEQUENCE variables were used \((1, 2, \ldots, 16 \text{ and } 0, 1, \ldots, 15)\), resulting in a total of 48 replicated models. Similar to Sections 3.2.1 and 3.2.2, the linear regression coefficients were obtained for all 48 models. The best model was chosen by the coefficients that had the smallest sum of squared deviations (SSE) from the Baimbridge’s coefficients.

![Parallel coordinates plot](image)

**Fig. 25**: The replication of the third model discussed in Baimbridge (1997). The parallel coordinates plot shows the original (in black) and 48 replicated models (in red and blue). Blue lines indicate the usage of the Cochrane and Orcutt technique. The dark blue line shows the best model. The dashed line represents 0. Min = -0.53 and Max = 4.58 here.

Figure 25 shows the parallel coordinates plot for the 48 models mentioned above. The first five columns show the regression coefficients of these models. The last column is
the sum of squared deviation from Baimbridge’s model. The best model is marked in
dark blue and Baimbridge’s model is marked in black. Figure 25 indicates that the
SSE values of the blue lines are smaller than the ones of the red lines. This means
that Cochrane and Orcutt (1949) technique will give a closer estimate to the original
model and will thus be implemented in the best replicated model. The model with
the minimum SSE has the following parameters:

- GAP: Rounded by years
- SEQUENCE: Start from one
- Cochrane and Orcutt: Used
- Adjustment year: 1962
- Adjustment method: Box–office mojo

Call:
```
lm(formula = YB ~ XB - 1)
```

Residuals:

<table>
<thead>
<tr>
<th></th>
<th>Min</th>
<th>1Q</th>
<th>Median</th>
<th>3Q</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min</td>
<td>-0.7241</td>
<td>-0.1292</td>
<td>0.0000</td>
<td>0.1938</td>
<td>0.6654</td>
</tr>
</tbody>
</table>

Coefficients:

| Estimate | Std. Error | t value | Pr(>|t|) |
|----------|------------|---------|---------|
| XB(Intercept) | 4.00934 | 0.48334 | 8.295 | 8.56e-06 *** |
| XBSEQUENCE | -0.08671 | 0.04189 | -2.070 | 0.0653 . |
| XBGAP | -0.54362 | 0.48847 | -1.113 | 0.2918 |
| XBGAPSQ | 0.15037 | 0.14631 | 1.028 | 0.3283 |
| XBCOLDWAR | -0.32378 | 0.48943 | -0.662 | 0.5232 |

---

Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

Residual standard error: 0.4231 on 10 degrees of freedom
Multiple R-squared: 0.9721, Adjusted R-squared: 0.9582
F-statistic: 69.8 on 5 and 10 DF, p-value: 1.899e-07

Fig. 26: OLS summary for the third model. The variable names are different because
the Cochrane and Orcutt method was adopted.
The OLS output using the parameters from the best model is given in Figure 26. Figure 27 shows the OLS coefficients with the corresponding t-values and ANOVA output of the original and the best replicated models. The vertical dashed lines show the t–values used in 95% confidence intervals of the OLS coefficients. They are ordered from the absolute smallest (bottom) t–value of the replicated model to the absolute highest one (top).

Fig. 27: Comparison of the third model discussed in Baimbridge (1997) and the best replicated model. The results of the replicated model are presented via red squares and the results of the original models are presented via blue circles.

Figure 27 shows similar results among Baimbridge and replicated models. SEQUENCE\(^3\) is marginally significant in both the original and the replicated models. The variables GAP, GAPSQ, and COLDWAR are not significant. In the original model, the F–

\[^{3}\text{Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1}\]
also shows that the model gives a closer estimate to Baimbridge’s coefficients when using the Cochrane and Orcutt (1949) technique. Thus, using the Durbin Watson statistic is a valid adjustment for this model. However, all the replicated estimates seems to overestimated the GAP variable and underestimate the variables GAPSQ and COLDWAR.

3.2.4 Fourth Model: First Attempt

In this section, the replication of the fourth model described in Baimbridge (1997) is performed. Similar to Sections 3.2.1–3.2.3, six different response variables were considered (two adjustment years for three adjustment methods). For each response variable, a regression model is implemented with and without the Cochrane and Orcutt (1949) technique.

In Baimbridge’s fourth model, shown in Figure 18, the coefficient estimate of the TOTADM is -3.6111. The range of the response variables calculated by Equation (1) and (2) varies between two and four, the TOTADM variable should not have high variation. However, the range of the TOTADM variable varies around one billion in the collected dataset. This suggests that some type of transformation is necessary to obtain a closer coefficient of TOTADM. Therefore, a log transformation and unit adjustment to billions were used.

Figure 28 shows the parallel coordinates plot for 24 replicated models. The first seven columns show the regression coefficient of these models. The last column is the sum of squared deviation (in thousands) from Baimbridge’s model. The best model is marked in dark red and Baimbridge’s model is marked in black.
Fig. 28: The replication of the fourth model discussed in Baimbridge (1997). The parallel coordinates plot shows the original (in black) and 24 replicated models (in red and blue). Blue lines indicate the usage of the Cochrane and Orcutt technique. The dark red line shows the best model. The dashed line represents 0. Min = -361 and Max = 177 here. The unit of the SSE variable is in thousands.

Figure 28 shows that even the best replicated model did not capture the signs of some of Baimbridge’s coefficients. The Cochrane and Orcutt (1949) technique gave higher SSE results. The model with the minimum SSE has the following parameters:
- TOTADM: log transformation
- Cochrane and Orcutt: not used
- Adjustment year: 1962
- Adjustment method: CPI

The OLS output using the parameters from the best model is given in Figure 29.
Call:
lm(formula = logBoxOffice ~ ., data = model4Old)

Residuals:
    Min 1Q Median 3Q Max
-0.69468 -0.29080 -0.04265 0.20841 0.89356

Coefficients:
                              Estimate Std. Error t value Pr(>|t|)
(Intercept)              -6.110e+00  2.364e+01  -0.258  0.802
PRICE                  -6.651e+01  1.380e+02  -0.482  0.641
PRICESQ                3.252e+01  6.984e+01   0.466  0.653
PCEMOVIES              7.140e+00  1.369e+01   0.522  0.615
PCEMOVIESQ            -2.945e-01  5.797e-01  -0.508  0.624
TOTADM                5.354e-04  1.681e-03   0.318  0.757
RELEASES             -4.979e-03  4.656e-03  -1.069  0.313

Residual standard error: 0.6201 on 9 degrees of freedom
Multiple R-squared:  0.1547,    Adjusted R-squared:  -0.4088
F-statistic: 0.2745 on 6 and 9 DF,  p-value: 0.9352

Fig. 29: OLS summary for the fourth model.

Figure 30 shows the OLS coefficients with the corresponding t-values and ANOVA output of the original and the best replicated models. The vertical dashed lines show the t-values used in 95% confidence intervals of the OLS coefficients. The OLS coefficients are ordered from the absolute smallest (bottom) t-value of the replicated model to the absolute highest one (top).

Figure 30 shows that the variables PRICE\(^*\)\(^4\), PRICESQ\(^*\), and TOTADM\(^*\) are significant for the Baimbridge model. In the replicated model, none of the variables are significant. Moreover, the variables PRICE and PRICESQ have opposite signs in the original and the replicated models. The F-statistic is relatively small in the replicated model. Similar patterns can be observed for Durbin Watson, \(R^2\), and adjusted \(R^2\). Overall, the replication of the fourth model was not satisfactory. Therefore, a second attempt was made by checking additional models.

\(^4\)Signif. codes: 0 "***" 0.001 "**" 0.01 "*" 0.05 "." 0.1 " " 1
3.2.5 Fourth Model: Second Attempt

Baimbridge (1997) mentioned that “... movie demand will only become price sensitive once a critical level has been reached resulting in the estimation of the relative maxima.” This sentence can be understood in different ways. Therefore, the following variations of the PRICE variable are considered:

\[
NEWPRICE_i = \begin{cases} 
0, & \text{if } PRICE_i < PRICE_{cut} \\
PRICE_i, & \text{otherwise}
\end{cases}
\] (3)

\[
NEWPRICE_i = \begin{cases} 
0, & \text{if } PRICE_i > PRICE_{cut} \\
PRICE_i, & \text{otherwise}
\end{cases}
\] (4)

for \( i = 1,2,\ldots,16 \). \( PRICE_{cut} \) takes any of the 16 values of the observed movie admission price.
For the given 16 movies, these equations create 32 different variations of the PRICE variable. However, in two of these variations the variable PRICESQ becomes only a linear transformation of the variable PRICE, which gives “NA” values when predicting the regression coefficient of PRICESQ. The exclusion of these two variations result in 30 different PRICE variables to consider.

When the Cochrane and Orcutt technique is applied, the calculation of the regression coefficients for some variations of the PRICE variable took several hours. Therefore, the Cochrane and Orcutt technique was only applied to the best four models, i.e., the models with the smallest SSE, that resulted from all models without using this technique. Thus, 30 PRICE variations without the Cochrane and Orcutt technique and four PRICE variations with the Cochrane and Orcutt technique are considered. For each of the above mentioned PRICE variations, two variations of the adjustment year, three variations of the adjustment method, and two variations of the TOTADM variable are applied creating a total of 408 (34 × 2 × 3 × 2) models.

Figure 31 shows the parallel coordinates plot for these 408 replicated models. The first seven columns show the regression coefficient of these models and the last column is the sum of squared deviation (in hundreds) from Baimbridge’s model. Out of these 408 models, the model with the minimum SSE has the following parameters:
- TOTADM: log transformation
- Cochrane and Orcutt: not used
- Adjustment year: 1962
- Adjustment method: Mojo
- PRICE cutoff: \( PRICE_{cut} = 1.0096 \) with \( NEW\,PRICE_i = \begin{cases} 0, & \text{if } PRICE_i > PRICE_{cut} \\ PRICE_i, & \text{otherwise} \end{cases} \)
Fig. 31: The second attempt to replicate the fourth model discussed in Baimbridge (1997). The parallel coordinates plot shows the original (in black) and 408 $(12 \times 30 + 12 \times 4)$ replicated models (in red and blue). Blue lines indicate the usage of the Cochrane and Orcutt technique. The dark red line shows the best model. The dashed line represents 0. Min = -110 and Max = 42 here. The unit of the SSE variable is in hundreds.

While the SSE of the best model in the first attempt was close to 7000, the SSE of the best model in the second one was under 70. This is an improvement of about 100 times in terms of the SSE. However, it is still around 300 to 400 times larger than the best SSE of the first three models. Even with the second attempt, the replication of the fourth models was not successful.

3.3 Replication Summary

This chapter included some attempts to replicate the four regression models presented in Baimbridge (1997). The data from the original article were not available which made the replication a hard task. Sometimes, up to 408 models were assessed
to obtain coefficients that were most similar to those obtained by Baimbridge in his original models. The first three models were successfully replicated, capturing the regression coefficients and the t-values very closely, in contrast to the replication of the fourth model. In each of these four models, three inflation adjusters and two adjustment years were considered.

Table 3 summarizes these settings for the best models obtained when replicating Baimbridge’s four models. The CPI index with the inflation adjustment year of 1962 was used for the first model. Average ticket price adjusted to 1963 was applied for the second model. The third and the fourth models used the box-office mojo inflation adjustment method with the 1962 inflation adjustment year.

Overall, there was not a big difference between the adjustment years and methods in terms of the sum of squared deviation from the original model. The models changed more dramatically when the Cochrane and Orcutt (1949) technique was used. In the first three models, when the Durbin and Watson (1971) statistic showed a significant effect of serial autocorrelation, the Cochrane and Orcutt (1949) technique gave a better estimate of the original model. The minimum (best) SSE’s of all replicated models are shown in Table 3. It appears that the first, the second, and the third models are really close to the original models, but the fourth model is not close at all to the corresponding original model.

<table>
<thead>
<tr>
<th></th>
<th>1962</th>
<th>1963</th>
<th>C&amp;O</th>
<th>Best SSE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CPI</td>
<td>Ticket</td>
<td>Mojo</td>
<td>CPI</td>
</tr>
<tr>
<td>First Model</td>
<td>✓</td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Second Model</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Third Model</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Fourth Model</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3: The characteristics of the “Best” replicated models. The column C&O shows the usage of the Cochrane and Orcutt technique.
CHAPTER 4
PREDICTING THE BOX-OFFICE REVENUES OF THE JB MOVIE SERIES

4.1 Prediction Methods Overview

Forecasting the box–office revenues (BOR) is one of the most important aspects of research in the movie industry. Sections 1.2 and 1.4 showed that prediction of the BOR became a popular task over the last three decades. In this chapter, various methods will be used to predict the BOR of the JB movie series. The 16 JB movies that were released before 1990 are used as a training set and the six JB movies released after 1990 are used as a test set.

The datasets used in the first and the third Baimbridge (1997) model and in The Economist model are used to make the predictions (Section 1.5). The dataset from the second Baimbridge model is not considered because the last version of the Halliwell book was released in 1989. For movies released after 1989, this makes the observations of the variables ONESTAR, TWOSTAR, and THREESTAR (Section 1.5.2) impossible to find. The replication of the fourth Baimbridge model was not successful and, thus, the prediction for this dataset is not considered.

OLS (Section 4.1.1), LASSO (Section 4.1.2), and random forests (Section 4.1.3) are applied on the first and the third Baimbridge model and on The Economist model to predict the BORs. Additionally, for the first and third models Baimbridge’s OLS coefficients were used to forecast the BOR. For each model, visualization tools are used to compare the different methods and their results.
4.1.1 Ordinary Least Squares (OLS)

OLS is the most commonly used method in regression. It is easy to model and interpret. In this chapter, OLS is used to predict the BORs. For the first and third Baimbridge models we will start from the full model and will check all possible combinations of explanatory variables. The best model will be selected by the variables that will minimize the Akaike Information Criterion (AIC) (Claeskens and Hjort, 2008, p. 22). In The Economist model, the AIC criteria deletes all the variables (kills, conquests, martinis, and BJB expression). Therefore, instead of fitting the minimum AIC model, the full model with four variables will be fitted. In the first and the third Baimbridge model, the AIC criteria will not be used because the Baimbridge (1997) fitted the full models.

4.1.2 LASSO

The LASSO (Tibshirani, 1996) is a shrinkage and selection method for linear regression, which constraints the absolute sum of the regression coefficients, $\sum_j |\beta_j|$’s. In other words, the LASSO estimates can be defined as:

$$ \hat{\beta} = \text{arg min}_{\beta} \left\{ \sum_{i=1}^{N} \left( y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\} \quad \text{s.t.} \quad \sum_{j=1}^{p} |\beta_j| \leq t \quad (5) $$

where $t \geq 0$ is the tuning parameter. The LASSO will be used for The Economist, the first and the third models. Among hundred tuning parameters, the best parameter will be chosen using cross–validation accuracy rates. Predictions will be made using the best tuning parameter.

4.1.3 Random Forests

Random forests (Breiman, 2001) is a popular ensemble–learning algorithm for
classification and regression. It is one of the tree–based algorithms in which the averages of multiple trees are taken for prediction (default number of trees, $ntree = 500$). At each node of the tree, random forests takes some number of variables ($mtry$) to perform the next split (for regression, the default $mtry$ is the square–root of the number of variables). All trees are fully grown. The regression random forests are applied to all three dataset discussed in Section 4.1.2. For this analysis, $mtry = 2$ and $ntree = 5000$ will be used.

**4.1.4 Benchmarks**

Two benchmark predictors were used to predict the BORs. The first benchmark (bench mean) is simply the average BOR of the first 16 movies. This will be equivalent to an OLS model with all $\beta$’s = 0. The second benchmark (bench mean 2) predicts the next BOR based on the average of all BORs found before. This benchmark will only be used in *The Economist* model, which will substitute the Baimbridge model.

Overall, the three or four methods will be compared with each other as well as with the one or two benchmarks. The comparison will be based on the root mean squared error (RMSE) which can be calculated with the following equation:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{k} (y_i - \hat{y}_i)^2}{k}}$$

(6)

where $k$ is the number of observations in the given dataset ($k = 16$ for the training set and $k = 6$ for the test set. $y_i$s are the observed BORs and $\hat{y}_i$s are the predicted BORs for those $k$ movies.

**4.2 Comparison of the First Model**

In this section, four regression and machine learning methods (OLS, Baimbridge
OLS, LASSO, and random forests) are used to predict the BORs of the first model. For all these models, the predicted values of the training set and the test set are shown in Figure 32. The left panel of Figure 32 shows a scatterplot of the movie release date and the response variable defined in Equation (1) in Section 3.2.1. Here the 1962 CPI inflation adjuster for the response variable is chosen, so that it is consistent with the parameters of the best replicated model (OLS) in Section 3.2.1.

![Graph showing observed and predicted values of log-transformed BORs](image)

Fig. 32: Observed and predicted values of the log-transformed BORs (left) for the first model. The OLS of the best replicated and the Baimbridge models are shown in the top left panel. LASSO and random forests appear in the bottom left panel. The faint colored points represent the training set. Prediction results are shown with dark colored points. The dashed line in the left panel shows the average of the first 16 movies. The RMSE for the training and test sets of these models are shown in the right panel.
As discussed in Section 4.1.1, the model was chosen by the minimum AIC value. In this model, the AIC criteria kept only the variables CONNERY, ACTREND, and ACTRENDSQ. For the first model, the OLS and the random forests predicted really well the third and fourth movies in the test set (see Figure 32). However, the prediction of Baimbridge and LASSO are not so impressive for any movies in the test set. In this figure, the predictions from all four methods mostly underestimate the observed BORs of the test set.

The dot plot in the right panel (Figure 32) shows the RMSE of the training and the test sets. The RMSE of the bench mean gives the smallest value for the test set, even though it has the largest value for the training set. Similar patterns can be observed for LASSO. The RMSE of the training set and the test set for the OLS and random forests are closer to each other. Random forests has the smallest RMSE among the four methods described in Section 4.1 which has a slightly smaller RMSE than that of the OLS. The benchmark mean has around three to four times smaller RMSE than the other methods described in Section 4.1. The best BOR prediction for future JB movies based on the first model is simply the average of the first 16 JB movies.

4.3 Comparison of the Third Model

In this section, the same methods as in Section 4.2 are used to predict the BORs. Here, the 1963 average ticket price inflation adjuster for the response variable is chosen, so that it is consistent with the parameters of the best replicated model (OLS) in Section 3.2.3. The minimum AIC criteria removed all variables except the variable SEQUENCE. This makes the relationship between the SEQUENCE and the response variable to be linear. The negative relationship between SEQUENCE and the response variable can be observed in the top left panel in Figure 33. This
relationship does not seem perfectly linear because the x-axis shows the release date (not SEQUENCE), and the gap between release dates is not constant.

Fig. 33: Observed and predicted values of the log-transformed BORs (left) for the third model. The OLS of the best replicated and the Baimbridge models are shown in the top left panel. LASSO and random forests appear in the bottom left panel. The faint colored points represent the training set. Prediction results are shown with dark colored points. The dashed line in the left panel shows the average of the first 16 movies. The RMSE for training and the test sets of these models are shown in the right panel.

Baimbridge’s model also shows a negative relationship between time and the response variable. However, the predicted BOR is unusually high in 1995. The movie *Goldeneye* (1995) has a GAP value of six years, and consequently a GAPSQ value of 36. Thus, such a high positive value for GAPSQ increases the prediction value by
almost two. Except this observation for Baimbridge, the other predictions for the test set are underestimated.

In the bottom left panel of Figure 33 a similar relationship can be observed for the LASSO and random forests methods. The predictions of random forests stays relatively constant for the test set, which allows random forests to have a smaller RMSE than that of the Baimbridge, OLS, and LASSO methods.

The right panel of Figure 33 shows the RMSE rate for both, the training set and the test set. Here, all methods (in the test set) perform worse in the third model compared to the first one. In particular, the best method (random forests) for the third model resulted in an RMSE that is almost twice as big as the RMSE for the best method (random forests) for the first model. Similar to Section 4.2 the RMSE of the benchmark gave the smallest RMSE which is more than six times smaller than the RMSE of the random forests in the third model.

4.4 Comparison of The Economist Model

Baimbridge (1997) did not use the The Economist dataset because it was only published in 2012. Thus, as a substitute to the Baimbridge model, the second benchmark (bench mean 2) is used. In Figure 34 red asterisks were used to mark fitted and predicted values of the bench mean 2 method. As stated in Section 4.1 the minimum AIC criteria was not applied to this dataset.

For The Economist dataset, the predicted values of the training set and the test set are shown in Figure 34. The BORs for the test set seem to decrease over time. The same pattern can be observed for LASSO, but with less a extreme decreasing rate. Random forests predict extremely well for the fourth and sixth movies in the test set. The prediction of the other four movies are acceptable. The right panel of Figure 34 shows the RMSE rate for both, the training set and the test set. Again,
the RMSE of the random forests is the smallest among the methods OLS, LASSO, and random forests. Overall, the RMSEs obtained in this section are much smaller than the ones in Section 4.2 and 4.3. However, the small RMSEs in this section were still higher than the RMSE of bench mean and bench mean 2.

Fig. 34: Observed and predicted values of the log–transformed BORs (left) for The Economist model. The OLS of the best replicated and the Bench Mean 2 models are shown in the top left panel. LASSO and random forests appear in the bottom left panel. The faint colored points represent the training set. Prediction results are shown with dark colored points. The dashed line in the left panel shows the average of the first 16 movies. The RMSE for the training and the test set of these models are shown in the right panel.
4.5 Summary of the Model Comparison

In this chapter, three datasets were observed to predict the BORs of the JB movie series. For each dataset, three or four methods were applied, and the RMSE of each method was determined (see Figures 32–34). Table 4 combines the RMSE results of these three models and five or six methods including the one or two benchmarks. The last column in that table is the arithmetic average of the RMSEs calculated in the first, the third and The Economist\(^1\) models. The last row shows the arithmetic average RMSE value of the three or four methods. The Economist dataset gives the smallest RMSE values among all three datasets. The third model dataset has the worst prediction rates with the average RMSE being more than two times higher than that of the The Economist model (See Table 4). Overall, the RMSEs of random forests were smaller than those of Baimbridge, OLS, and LASSO. Table 4 summarizes the test set RMSE for three datasets and four methods. None of these models is able to beat the benchmarks suggesting that the average of the first 16 or all previously released movies are the safest predictors.

<table>
<thead>
<tr>
<th></th>
<th>Model 1</th>
<th>Model 3</th>
<th>The Economist</th>
<th>Mean Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baimbridge</td>
<td>0.586</td>
<td>0.883</td>
<td>NA</td>
<td>0.735</td>
</tr>
<tr>
<td>OLS</td>
<td>0.372</td>
<td>0.979</td>
<td>0.503</td>
<td>0.618</td>
</tr>
<tr>
<td>LASSO</td>
<td>0.640</td>
<td>0.841</td>
<td>0.343</td>
<td>0.608</td>
</tr>
<tr>
<td>Forests</td>
<td>0.357</td>
<td>0.751</td>
<td>0.242</td>
<td>0.450</td>
</tr>
<tr>
<td>Bench Mean 1</td>
<td>0.103</td>
<td>0.146</td>
<td>0.103</td>
<td>0.117</td>
</tr>
<tr>
<td>Bench Mean 2</td>
<td>NA</td>
<td>NA</td>
<td>0.099</td>
<td>0.099</td>
</tr>
<tr>
<td>Mean Model</td>
<td>0.526</td>
<td>0.864</td>
<td>0.363</td>
<td></td>
</tr>
</tbody>
</table>

Table 4: Summary of the RMSE values for the test sets.

\(^1\)For Baimbride, the average includes only the first and the third models and for the Bench Mean 2, the average includes only The Economist model
4.6 Usage of R Packages

All the statistical analysis and visualization in this Master’s report were done in R (R Core Team, 2013). Various R packages were used to produce the graphs in Chapters 2, 3, and 4. The time series plots in Section 2.2, scatterplot matrix in Section 2.3, boxplots in Section 2.5, and histograms in Section 2.6 were created by the graphics package in R (R Core Team, 2013). The package Hmisc (Harrell et al., 2014) was used to produce the dotplots in Section 2.4. Normal quantiles plots were created by the car package (Fox and Weisberg, 2011).

The R package PairViz (Hurley and Oldford, 2011) was used to generate the parallel coordinates plot in Section 2.7. For the heatmap plots in Section 2.8, the R package gplot (Warnes et al., 2013) was used. The R package vcd (Meyer et al., 2006) was used to generate the mosaic plot in Figure 13. The association plot in Figure 14 was created by using the vcd package (Meyer et al., 2006). The R package maps (Becker, Wilks, Brownrigg and Minka, 2013) was used to produce the choropleth map before the collapse of the USSR and the packages maptools (Bivand and Lewin-Koh, 2013) and mapdata (Becker, Wilks and Brownrigg, 2013) were used to produce the choropleth map after the collapse of the USSR.

All dotplots and parallel coordinates plots in Chapter 3 (Figures 19, 21, 22, 24, 25, 27, 28, 30) were produced using the lattice package in R (Sarkar, 2008). The Cochrane and Orcutt (1949) technique was done by the orcutt package in R (Spada et al., 2012).

All figures in Chapter 4 (Figures 32, 34) were based on the graphics package (R Core Team, 2013). Computations and predictions to perform random forests were obtained from the randomForest package (Liaw and Wiener, 2002). All computations for LASSO were done using the glmnet package (Friedman et al., 2010).

The RColorBrewer (Neuwirth, 2011) package was used to color the lines or points.
in the following figures: Chapter 2 – time series plots, scatterplot matrix, parallel coordinates plot, heatmaps, choropleth maps; Chapter 3 – all figures; Chapter 4 – all figures.
CHAPTER 5
CONCLUSION AND OUTLOOK

The movie industry plays a very important role in the US economy. Therefore, research related to the movie industry has become an important topic for many scientists over the last three decades. Being one of the riskiest industries, prediction of box–office revenues (BOR) can play a vital role for many movies. In this Master’s report, the visualization and prediction of the box–office revenues for the James Bond movie series was performed. In Chapter 2, various visualization techniques were presented to understand the details in the data presented in The Economist article [The Economist (2012)].

An increasing trend was observed for the variables JB kills and martinis drunk over time (Figures 1 and 3). Some positive relationship was found between the BOR and number of JB kills, and some negative relationship between BOR and BJB expressions (Figures 5 and 7). Additionally, some clustering for the JB actors Moore, Dalton, Brosnan, and Craig (Figures 11 and 12) was observed.

Chapter 3 replicated four linear regression models that were presented in Baimbridge (1997). The first three models were successfully replicated, closely capturing the regression coefficients, t–values, F–statistic, the $R^2$, adjusted $R^2$, and Durbin Watson statistic. The total sum of squared deviations (SSE) of these models from Baimbridge’s coefficients were between 0.2 and 0.3 (see Table 3). The SSE results for these three models were impressively good, compared to the SSE value of the fourth model which was 68.45.

Chapter 4 used three datasets (the first model, the third model, and The Economist model; p. 7) to predict the BORs. For all these models, the ordinary least squares,
LASSO, and random forests were applied to make predictions. The Economist dataset using random forests gave the best prediction results in terms of the error rate on the test dataset. However, this is still worse than just using the past mean to make predictions for future box-office revenues.

Overall, we found that more JB kills and less BJB expression would increase the BORs. Additionally, we found some significant indicators to predict the BOR, but this did not give us much confidence to generalize the forecasting for upcoming JB movies. After all, “. . . No one can tell you what a movie is going to do in the marketplace . . . Not until that film opens in a darkened theater, and sparks fly up between the screen and the audience can you say this film is right” (Valenti 1978).

Finally, there is a slogan that says “milk the cow as long as you can”. This seems to be applicable to movie series, in particular to the James Bond movie series.

5.1 Future Work

Future research to forecast the BOR could look into the following:
- adjust for population growth
- adjust for the number of movie theaters
- adjust for the average capacity of movie theaters
- use other regression methods
- use time series analysis; in particular, autoregressive and moving averages methods
- include inflation–adjusted production costs
- include release information (holiday weekend vs. non-holiday weekend)
- include movie director information
- combine the significant variables from the first and the third model into a new model
REFERENCES


**URL:** [http://CRAN.R-project.org/package=mapdata](http://CRAN.R-project.org/package=mapdata)


**URL:** [http://CRAN.R-project.org/package=maps](http://CRAN.R-project.org/package=maps)


**URL:** [http://CRAN.R-project.org/package=maptools](http://CRAN.R-project.org/package=maptools)


**URL:** http://chance.amstat.org/2014/04/james-bond/


**URL:** http://socserv.socsci.mcmaster.ca/jfox/Books/Companion

**URL:** [http://www.jstatsoft.org/v33/i01/](http://www.jstatsoft.org/v33/i01/)


**URL:** [http://CRAN.R-project.org/package=Hmisc](http://CRAN.R-project.org/package=Hmisc)


**URL:** [http://CRAN.R-project.org/package=PairViz](http://CRAN.R-project.org/package=PairViz)


**URL:** http://CRAN.R-project.org/doc/Rnews/


**URL:** [http://CRAN.R-project.org/package=RColorBrewer](http://CRAN.R-project.org/package=RColorBrewer)


**URL:** [http://www.R-project.org](http://www.R-project.org)


**URL:** [http://lmdvr.r-forge.r-project.org](http://lmdvr.r-forge.r-project.org)


**URL:** [http://CRAN.R-project.org/package=orcutt](http://CRAN.R-project.org/package=orcutt)


APPENDICES
## Inflation Adjusters

Here, the ticket price is in USD and the CPI is just a multiplier (i.e., unitless)

<table>
<thead>
<tr>
<th>Movie Name</th>
<th>Release Date</th>
<th>Ticket Price</th>
<th>CPI index</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dr. No</td>
<td>1963-05-01</td>
<td>0.85</td>
<td>30.6</td>
</tr>
<tr>
<td>From Russia, with Love</td>
<td>1964-04-01</td>
<td>0.93</td>
<td>31.0</td>
</tr>
<tr>
<td>Goldfinger</td>
<td>1964-12-01</td>
<td>0.93</td>
<td>31.0</td>
</tr>
<tr>
<td>Thunderball</td>
<td>1965-12-01</td>
<td>1.01</td>
<td>31.5</td>
</tr>
<tr>
<td>You Only Live Twice</td>
<td>1967-06-01</td>
<td>1.20</td>
<td>33.4</td>
</tr>
<tr>
<td>On Her Majesty’s Secret Service</td>
<td>1969-12-01</td>
<td>1.42</td>
<td>36.7</td>
</tr>
<tr>
<td>Diamonds Are Forever</td>
<td>1971-12-01</td>
<td>1.65</td>
<td>40.5</td>
</tr>
<tr>
<td>Live and Let Die</td>
<td>1973-06-01</td>
<td>1.77</td>
<td>44.4</td>
</tr>
<tr>
<td>The Man with the Golden Gun</td>
<td>1974-12-01</td>
<td>1.87</td>
<td>49.3</td>
</tr>
<tr>
<td>The Spy Who Loved Me</td>
<td>1977-07-01</td>
<td>2.23</td>
<td>60.6</td>
</tr>
<tr>
<td>Moonraker</td>
<td>1979-06-01</td>
<td>2.51</td>
<td>72.6</td>
</tr>
<tr>
<td>For Your Eyes Only</td>
<td>1981-06-01</td>
<td>2.78</td>
<td>90.9</td>
</tr>
<tr>
<td>Octopussy</td>
<td>1983-06-01</td>
<td>3.15</td>
<td>99.6</td>
</tr>
<tr>
<td>A View to a Kill</td>
<td>1985-05-01</td>
<td>3.55</td>
<td>107.6</td>
</tr>
<tr>
<td>The Living Daylights</td>
<td>1987-07-01</td>
<td>3.91</td>
<td>113.6</td>
</tr>
<tr>
<td>License to Kill</td>
<td>1989-07-01</td>
<td>3.97</td>
<td>124.0</td>
</tr>
<tr>
<td>GoldenEye</td>
<td>1995-11-01</td>
<td>4.35</td>
<td>152.4</td>
</tr>
<tr>
<td>Tomorrow Never Dies</td>
<td>1997-12-01</td>
<td>4.59</td>
<td>160.5</td>
</tr>
<tr>
<td>The World Is Not Enough</td>
<td>1999-11-01</td>
<td>5.08</td>
<td>166.6</td>
</tr>
<tr>
<td>Die Another Day</td>
<td>2002-11-02</td>
<td>5.81</td>
<td>179.9</td>
</tr>
<tr>
<td>Casino Royale</td>
<td>2006-11-06</td>
<td>6.55</td>
<td>201.6</td>
</tr>
<tr>
<td>Quantum of Solace</td>
<td>2008-11-08</td>
<td>7.18</td>
<td>215.3</td>
</tr>
</tbody>
</table>
A.2 The Response

The BOR–raw is in millions of USD and CPI-63 · · · Mojo-62 are in USD which are calculated using the Equations (1) and (2).

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>16.07</td>
<td>2.777</td>
<td>2.764</td>
<td>2.777</td>
<td>2.583</td>
<td>2.777</td>
<td>2.583</td>
</tr>
<tr>
<td>2</td>
<td>24.80</td>
<td>3.198</td>
<td>3.185</td>
<td>3.121</td>
<td>2.927</td>
<td>3.121</td>
<td>2.927</td>
</tr>
<tr>
<td>3</td>
<td>51.08</td>
<td>3.920</td>
<td>3.907</td>
<td>3.843</td>
<td>3.649</td>
<td>4.032</td>
<td>3.838</td>
</tr>
<tr>
<td>4</td>
<td>63.60</td>
<td>4.124</td>
<td>4.110</td>
<td>3.980</td>
<td>3.786</td>
<td>4.152</td>
<td>3.958</td>
</tr>
<tr>
<td>5</td>
<td>43.08</td>
<td>3.676</td>
<td>3.662</td>
<td>3.418</td>
<td>3.224</td>
<td>3.418</td>
<td>3.224</td>
</tr>
<tr>
<td>6</td>
<td>22.77</td>
<td>2.944</td>
<td>2.931</td>
<td>2.612</td>
<td>2.418</td>
<td>2.612</td>
<td>2.418</td>
</tr>
<tr>
<td>7</td>
<td>43.82</td>
<td>3.500</td>
<td>3.487</td>
<td>3.117</td>
<td>2.923</td>
<td>3.117</td>
<td>2.923</td>
</tr>
<tr>
<td>8</td>
<td>35.38</td>
<td>3.194</td>
<td>3.181</td>
<td>2.833</td>
<td>2.638</td>
<td>2.833</td>
<td>2.638</td>
</tr>
<tr>
<td>9</td>
<td>20.97</td>
<td>2.566</td>
<td>2.553</td>
<td>2.255</td>
<td>2.061</td>
<td>2.255</td>
<td>2.061</td>
</tr>
<tr>
<td>10</td>
<td>46.84</td>
<td>3.163</td>
<td>3.150</td>
<td>2.882</td>
<td>2.688</td>
<td>2.882</td>
<td>2.688</td>
</tr>
<tr>
<td>11</td>
<td>70.31</td>
<td>3.389</td>
<td>3.376</td>
<td>3.170</td>
<td>2.976</td>
<td>3.170</td>
<td>2.976</td>
</tr>
<tr>
<td>12</td>
<td>54.81</td>
<td>2.915</td>
<td>2.902</td>
<td>2.819</td>
<td>2.625</td>
<td>2.819</td>
<td>2.625</td>
</tr>
<tr>
<td>13</td>
<td>67.89</td>
<td>3.038</td>
<td>3.025</td>
<td>2.908</td>
<td>2.714</td>
<td>2.908</td>
<td>2.714</td>
</tr>
<tr>
<td>14</td>
<td>50.33</td>
<td>2.661</td>
<td>2.648</td>
<td>2.489</td>
<td>2.295</td>
<td>2.489</td>
<td>2.295</td>
</tr>
<tr>
<td>15</td>
<td>51.19</td>
<td>2.624</td>
<td>2.611</td>
<td>2.409</td>
<td>2.215</td>
<td>2.409</td>
<td>2.215</td>
</tr>
<tr>
<td>16</td>
<td>34.67</td>
<td>2.147</td>
<td>2.133</td>
<td>2.005</td>
<td>1.810</td>
<td>2.005</td>
<td>1.810</td>
</tr>
<tr>
<td>17</td>
<td>106.43</td>
<td>3.062</td>
<td>3.049</td>
<td>3.035</td>
<td>2.841</td>
<td>3.032</td>
<td>2.838</td>
</tr>
<tr>
<td>18</td>
<td>125.30</td>
<td>3.173</td>
<td>3.160</td>
<td>3.144</td>
<td>2.950</td>
<td>3.130</td>
<td>2.936</td>
</tr>
<tr>
<td>19</td>
<td>126.94</td>
<td>3.149</td>
<td>3.136</td>
<td>3.056</td>
<td>2.862</td>
<td>3.050</td>
<td>2.856</td>
</tr>
<tr>
<td>20</td>
<td>160.94</td>
<td>3.310</td>
<td>3.296</td>
<td>3.159</td>
<td>2.965</td>
<td>3.155</td>
<td>2.961</td>
</tr>
<tr>
<td>21</td>
<td>167.45</td>
<td>3.235</td>
<td>3.222</td>
<td>3.079</td>
<td>2.885</td>
<td>3.073</td>
<td>2.879</td>
</tr>
<tr>
<td>22</td>
<td>168.37</td>
<td>3.175</td>
<td>3.162</td>
<td>2.992</td>
<td>2.798</td>
<td>2.992</td>
<td>2.798</td>
</tr>
</tbody>
</table>
## A.3 The Economist

<table>
<thead>
<tr>
<th></th>
<th>Kills</th>
<th>Conquests</th>
<th>Martinis</th>
<th>BJB Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>0</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>0</td>
<td>14</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>1</td>
<td>17</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>1</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>0</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>0</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>1</td>
<td>11</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>3</td>
<td>1</td>
<td>12</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>0</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>13</td>
<td>2</td>
<td>1</td>
<td>12</td>
<td>1</td>
</tr>
<tr>
<td>14</td>
<td>4</td>
<td>0</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>15</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>16</td>
<td>2</td>
<td>1</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>17</td>
<td>2</td>
<td>1</td>
<td>26</td>
<td>1</td>
</tr>
<tr>
<td>18</td>
<td>2</td>
<td>1</td>
<td>17</td>
<td>1</td>
</tr>
<tr>
<td>19</td>
<td>3</td>
<td>1</td>
<td>17</td>
<td>2</td>
</tr>
<tr>
<td>20</td>
<td>2</td>
<td>2</td>
<td>16</td>
<td>1</td>
</tr>
<tr>
<td>21</td>
<td>2</td>
<td>4</td>
<td>11</td>
<td>1</td>
</tr>
<tr>
<td>22</td>
<td>1</td>
<td>6</td>
<td>6</td>
<td>0</td>
</tr>
</tbody>
</table>
### A.4 The First Model

<table>
<thead>
<tr>
<th></th>
<th>CONNERY</th>
<th>LAZENBY</th>
<th>MOORE</th>
<th>ACTREND</th>
<th>ACTRENSQ</th>
<th>NEWBOND</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>25</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>36</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>4</td>
<td>16</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>5</td>
<td>25</td>
<td>0</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>6</td>
<td>36</td>
<td>0</td>
</tr>
<tr>
<td>14</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>7</td>
<td>49</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>16</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>17</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>18</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>19</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>16</td>
<td>0</td>
</tr>
<tr>
<td>21</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>22</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>0</td>
</tr>
</tbody>
</table>
### A.5 The Second Model

<table>
<thead>
<tr>
<th></th>
<th>ONESTAR</th>
<th>TWOSTAR</th>
<th>THREESTAR</th>
<th>WONOSCAR</th>
<th>NOMOSCAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>14</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
### A.6 The Third Model

<table>
<thead>
<tr>
<th>SEQUENCE</th>
<th>GAP</th>
<th>GAPSQ</th>
<th>COLDWAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>13</td>
<td>13</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>14</td>
<td>14</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>15</td>
<td>15</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>17</td>
<td>17</td>
<td>6</td>
<td>36</td>
</tr>
<tr>
<td>18</td>
<td>18</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>19</td>
<td>19</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>20</td>
<td>20</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>21</td>
<td>21</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>22</td>
<td>22</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>
A.7 The Forth Model

The PRICE is in USD.

<table>
<thead>
<tr>
<th></th>
<th>PRICE</th>
<th>PRICESQ</th>
<th>PCEMOVIES</th>
<th>PCEMOVIESQ</th>
<th>TOTADM</th>
<th>RELEASES</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.85</td>
<td>0.722</td>
<td>10.415</td>
<td>108.5</td>
<td>1092</td>
<td>112</td>
</tr>
<tr>
<td>2</td>
<td>0.92</td>
<td>0.843</td>
<td>11.131</td>
<td>123.9</td>
<td>1020</td>
<td>125</td>
</tr>
<tr>
<td>3</td>
<td>0.92</td>
<td>0.843</td>
<td>11.131</td>
<td>123.9</td>
<td>1020</td>
<td>125</td>
</tr>
<tr>
<td>4</td>
<td>0.98</td>
<td>0.963</td>
<td>11.774</td>
<td>138.6</td>
<td>1288</td>
<td>130</td>
</tr>
<tr>
<td>5</td>
<td>1.10</td>
<td>1.209</td>
<td>12.869</td>
<td>165.6</td>
<td>1200</td>
<td>133</td>
</tr>
<tr>
<td>6</td>
<td>1.18</td>
<td>1.402</td>
<td>13.906</td>
<td>193.4</td>
<td>1270</td>
<td>87</td>
</tr>
<tr>
<td>7</td>
<td>1.25</td>
<td>1.554</td>
<td>14.628</td>
<td>214.0</td>
<td>820</td>
<td>98</td>
</tr>
<tr>
<td>8</td>
<td>1.22</td>
<td>1.488</td>
<td>14.272</td>
<td>203.7</td>
<td>865</td>
<td>69</td>
</tr>
<tr>
<td>9</td>
<td>1.16</td>
<td>1.347</td>
<td>13.696</td>
<td>187.6</td>
<td>1005</td>
<td>67</td>
</tr>
<tr>
<td>10</td>
<td>1.13</td>
<td>1.268</td>
<td>13.112</td>
<td>171.9</td>
<td>1060</td>
<td>74</td>
</tr>
<tr>
<td>11</td>
<td>1.06</td>
<td>1.119</td>
<td>12.442</td>
<td>154.8</td>
<td>1118</td>
<td>100</td>
</tr>
<tr>
<td>12</td>
<td>0.94</td>
<td>0.876</td>
<td>10.885</td>
<td>118.5</td>
<td>1055</td>
<td>101</td>
</tr>
<tr>
<td>13</td>
<td>0.97</td>
<td>0.937</td>
<td>11.137</td>
<td>124.0</td>
<td>1173</td>
<td>119</td>
</tr>
<tr>
<td>14</td>
<td>1.01</td>
<td>1.019</td>
<td>11.470</td>
<td>131.6</td>
<td>1056</td>
<td>144</td>
</tr>
<tr>
<td>15</td>
<td>1.05</td>
<td>1.109</td>
<td>11.790</td>
<td>139.0</td>
<td>1089</td>
<td>208</td>
</tr>
<tr>
<td>16</td>
<td>0.98</td>
<td>0.960</td>
<td>12.106</td>
<td>146.5</td>
<td>1133</td>
<td>211</td>
</tr>
</tbody>
</table>
B.1 R Code for Chapter 2

library(car)
setwd("C:/Users/Vahan/Desktop/Masters/MS Project/Report")
BondData <- read.csv("MS_project_Data2.csv")
movieDate <- as.Date(BondData[, 13])

#model 1
CONNERY <- ifelse(as.character(BondData$Actors.Name) == "Connery", 1, 0)
LAZENBY <- ifelse(as.character(BondData$Actors.Name) == "Lazenby", 1, 0)
MOORE <- ifelse(as.character(BondData$Actors.Name) == "Moore", 1, 0)
DALTON <- ifelse(as.character(BondData$Actors.Name) == "Dalton", 1, 0)
BROSNAN <- ifelse(as.character(BondData$Actors.Name) == "Brosnan", 1, 0)
CRAIG <- ifelse(as.character(BondData$Actors.Name) == "Craig", 1, 0)

ACTREND <- rep(1, 22)
NEWBOND <- rep(1, 22)
k <- 1
for (i in 2:22) {
  k <- ifelse(BondData$Actors.Name[i] == BondData$Actors.Name[i-1], k + 1, 1)
  ACTREND[i] <- k
  if (BondData$Actors.Name[i] == BondData$Actors.Name[i-1]) {
    NEWBOND[i] <- 0
  }
}
ACTREND[7] <- 6
ACTRENDSQ <- ACTREND ^ 2
# model 2

ing <- read.csv("model 2.csv")
ONESTAR <- ifelse(rating$Halliwell == 1, 1, 0)
TWOSTAR <- ifelse(rating$Halliwell == 2, 1, 0)
THREESTAR <- ifelse(rating$Halliwell == 3, 1, 0)
NOMOSCAR <- rating$nom
WONOSCAR <- rating$win

#model 3

SEQUENCE <- 1:22
movieYear <- as.numeric(substring(movieDate, 1, 4))
movieMonth <- as.numeric(substring(movieDate, 6, 7))
movieYear.month <- movieYear + movieMonth / 12
GAP <- c(0, diff(movieYear))
GAPSQ <- GAP ^ 2
COLDWAR <- ifelse(movieYear.month < 1989, 0, 1)
GAP1 <- c(0, diff(movieYear))
GAPSQ1 <- GAP1 ^ 2

# model 4

CPIindex <- read.csv("CPIindex.csv")
CPI.year <- CPIindex[, 1]
movieYear <- as.numeric(substring(movieDate, 1, 4))
CPImovie <- sapply(1:22, function(x)
    as.numeric(as.character(CPIindex[which(CPI.year == movieYear[x]), 2])))
# PRICE <- BondData$Average.Ticket.Price * 30.2 / CPImovie
PRICESQ <- PRICE ^ 2
PCEindex <- read.csv("PCEindex.csv")
PCE.year <- as.numeric(substring(PCEindex[, 1], 1, 4))
moviePCE <- sapply(1:22, function(x)
    as.numeric(as.character(PCEindex[which(PCE.year == movieYear[x]), 2])))

PCEMOVIES <- moviePCE * CPImovie[1] / CPImovie
# PCEMOVIES <- moviePCE * 10.043 / CPImovie

PCEMOVIESQ <- PCEMOVIES ^ 2

totalAdmission <- read.csv("movie admission.csv")
totYear <- totalAdmission[, 1]
TOTADM <- sapply(1:22, function(x)
    as.numeric(as.character(totalAdmission[which(totYear == movieYear[x]), 2])))

numReleases <- read.csv("releases.csv")[-1, ]
tableReleases <- table(numReleases$YEAR)
RELEASES <- sapply(1:22, function(x)
    tableReleases[which(as.numeric(names(tableReleases)) == movieYear[x])])

########################################################
# pdf("timePlot1.pdf", height = 5, width = 9)
library(RColorBrewer)
par(mar = c(4, 4, 0.5, 0.5))
plot(movieDate, BondData$Bond.kills, xlim = c(-3200, 15000), ylim = c(0, 30),
    pch = c(8, 4:0)[BondData$Actors.Name], cex = 1.3, cex.axis = 1.3, cex.lab = 1.3,
    col = brewer.pal(6,"Dark2")[BondData$Actors.Name],
    xlab = "Release Date", ylab = "Number of JB kills")

lm1 <- lm(BondData$Bond.kills ~ movieDate)
line.col <- brewer.pal(3, "Accent")
yhat <- lm1$coef[1] + as.numeric(movieDate) * lm1$coef[2]
lines(lowess(movieDate, yhat), col = line.col[1], xlim = c(-3200, 15000),
    lwd = 1.5)
```r
lines(lowess(movieDate, BondData$Bond.kills, f = 0.5), col = line.col[2],
       xlim = c(-3200, 15000), lwd = 1.5)

ma5 <- rep(1, 5) / 5
kills.ma5 <- filter(BondData$Bond.kills, ma5)
lines(movieDate, y = kills.ma5, type = "l", col = line.col[3], lwd = 1.5)

legend(x = 12500, y = 30, title = "Actors",
       c("Connery", "Lazenby", "Moore", "Dalton", "Brosnan", "Craig"),
       pch = c(4, 1, 0, 2, 8, 3),
       col = c("#D95F02", "#66A61E", "#E6AB02", "#E7298A", "#1B9E77", "#7570B3"),
       horiz = F)

legend("topleft", inset=.05,
       c("Regression Line", "Lowess Smoothing", "MA Smoothing"),
       lty = c(1, 1, 1), col = brewer.pal(3, "Accent"), lwd = "2"
)

# dev.off()

# pdf("timePlot2.pdf", height = 5, width = 9)
par(mar = c(4, 4, 0.5, 0.5))
plot(movieDate, BondData$Conquests, xlim = c(-3200, 15000), ylim = c(0, 6),
     pch = c(8, 4:0)[BondData$Actors.Name], cex = 1.3, cex.axis = 1.3, cex.lab = 1.3,
     col = brewer.pal(6, "Dark2")[BondData$Actors.Name],
     xlab = "Release Date", ylab = "Number of conquests")

lm1 <- lm(BondData$Conquests ~ movieDate)
line.col <- brewer.pal(3, "Accent")
```
lines(lowess(movieDate, yhat), col = line.col[1], xlim = c(-3200, 15000), lwd = 1.5)
lines(lowess(movieDate,BondData$Conquests, f = 0.5), col = line.col[2], xlim = c(-3200, 18000), lwd = 1.5)

ma5 <- rep(1, 5) / 5
kills.ma5 <- filter(BondData$Conquests, ma5)
lines(movieDate, y = kills.ma5, type = "l", col = line.col[3], lwd = 1.5)

legend(x = 12500, y = 6, title = "Actors",
c("Connery", "Lazenby", "Moore", "Dalton", "Brosnan", "Craig"),
pch = c(4, 1, 0, 2, 8, 3),
col = c("#D95F02", "#66A61E", "#E6AB02", "#E7298A", "#1B9E77", "#7570B3"),
horiz = F)

legend("topleft",inset=.05,
c("Regression Line", "Lowess Smoothing", "MA Smoothing"),
lty = c(1, 1, 1), col = brewer.pal(3, "Accent"), lwd = "2"
)
# dev.off()

########################################################
# pdf("timePlot3.pdf", height = 5, width = 9)
par(mar = c(4, 4, 0.5, 0.5))
plot(movieDate, BondData$Martinis, xlim = c(-3200, 15000), ylim = c(0, 6),
pch = c(8, 4:0)[BondData$Actors.Name], cex = 1.3, cex.axis = 1.3, cex.lab = 1.3,
col= brewer.pal(6, "Dark2")[BondData$Actors.Name],
xlab = "Release Date", ylab = "Number of martinis")
lm1 <- lm(BondData$Martinis ~ movieDate)
line.col <- brewer.pal(3, "Accent")
yhat <- lm1$coef[1] + as.numeric(movieDate) * lm1$coef[2]
lines(lowess(movieDate, yhat), col = line.col[1], xlim = c(-3200, 15000),
  lwd = 1.5)
lines(lowess(movieDate, BondData$Martinis, f = 0.5), col = line.col[2],
  xlim = c(-3200, 15000), lwd = 1.5)

ma5 <- rep(1, 5) / 5
kills.ma5 <- filter(BondData$Martinis, ma5)
lines(movieDate, y = kills.ma5, type = "l", col = line.col[3], lwd = 1.5 )

legend(x = 8000, y = 6, title = "Actors",
  c("Connery", "Lazenby", "Moore", "Dalton", "Brosnan", "Craig"),
  pch = c(4, 1, 0, 2, 8, 3),
  col = c( "#D95F02", "#66A61E", "#E6AB02", "#E7298A", "#1B9E77", "#7570B3"),
  horiz = F)

legend("topleft", inset=.05,
  c("Regression Line", "Lowess Smoothing", "MA Smoothing"),
  lty = c(1, 1, 1), col = brewer.pal(3, "Accent"), lwd = 2"
)
# dev.off()
col = brewer.pal(6, "Dark2")[BondData$Actors.Name],

xlab = "Release Date", ylab = "Number of BJB", yaxt = "n")

axis(2, at=0:3, labels = 0:3, cex.axis = 1.3)

lm1 <- lm(BondData$Bond.James.Bond ~ movieDate)

line.col <- brewer.pal(3, "Accent")

yhat <- lm1$coef[1] + as.numeric(movieDate) * lm1$coef[2]

lines(lowess(movieDate, yhat), col = line.col[1], xlim = c(-3200, 15000),

lwd = 1.5)

lines(lowess(movieDate,BondData$Bond.James.Bond, f = 0.5), col = line.col[2],

xlim = c(-3200, 15000), lwd = 1.5)

ma5 <- rep(1, 5) / 5

kills.ma5 <- filter(BondData$Bond.James.Bond, ma5)

lines(movieDate, y = kills.ma5, type = "l", col = line.col[3], lwd = 1.5 )

legend(x = 12500, y = 3, title = "Actors",

c("Connery", "Lazenby", "Moore", "Dalton", "Brosnan", "Craig"),

pch = c(4, 1, 0, 2, 8, 3),

col = c("#D95F02", "#66A61E", "#E6AB02", "#E7298A", "#1B9E77", "#7570B3"),

horiz = F)

legend("topleft",inset=.05,

c("Regression Line", "Lowess Smoothing", "MA Smoothing"),

lty = c(1, 1, 1), col = brewer.pal(3, "Accent"), lwd = "2"
)

# dev.off()

## put histograms on the diagonal

panel.hist <- function(x, ...)
usr <- par("usr"); on.exit(par(usr))
par(usr = c(usr[1:2], 0, 1.5))
if (max(x) < 3) {
  h <- hist(x + 0.5, plot = FALSE, breaks = -1:3)
}
if (max(x) < 5 & max(x) > 3) {
  h <- hist(x + 0.5, plot = FALSE, breaks = 0:5)
}
if (max(x) < 10 & max(x) > 5) {
  h <- hist(x + 0.5, plot = FALSE, breaks = 0:7)
}
if (max(x) < 30 & max(x) > 10) {
  h <- hist(x + 0.5, plot = FALSE, breaks = seq(0, 30, 6))
}
if (max(x) > 30) {
  h <- hist(x + 0.5, plot = FALSE)
}

if (max(x) > 30) {
  breaks <- (h$breaks + 25 * 10 ^ 6); nB <- length(breaks)
} else {
  breaks <- (h$breaks - 0.5); nB <- length(breaks)
}
y <- h$counts; y <- y / max(y)
rect(breaks[-nB], 0, breaks[-1], y, ...)
rug(x)
BondData.jitter <- cbind(BondData[,c(7, 11, 9, 10, 12)])
BondData.jitter[, 1] <- BondData.jitter[, 1] * 8.35 / 7.94
BondData.jitter[, 2] <- jitter(BondData.jitter[, 2], amount = 0.1)
BondData.jitter[, 3] <- jitter(BondData.jitter[, 3], amount = 0.1)
BondData.jitter[, 4] <- jitter(BondData.jitter[, 4], amount = 0.1)
BondData.jitter[, 5] <- jitter(BondData.jitter[, 5], amount = 0.1)
names(BondData.jitter) <- c("BOR", "Kills", "Conquests", "Martinis", "BJB")

# pdf("scatterPlot1.pdf", height = 12, width = 12)
pairs(BondData.jitter, cex=2, cex.axis = 2, panel = panel.smooth,
   col = brewer.pal(6,"Dark2")[BondData$Actors.Name],
   pch = c(8, 4, 3, 2, 1, 0)[BondData$Actors.Name],
   diag.panel = panel.hist, cex.labels = 2, font.labels = 2,
   col.smooth = brewer.pal(3, "Accent")[2])
# dev.off()

#################################

library(Hmisc)
counts.actor <- table(BondData$Actors.Name)
counts.actor <- sort(counts.actor, decreasing = F)
avg.kills <- tapply(BondData$Bond.kills, BondData$Actors.Name, mean)
avg.kills <- sort(avg.kills, decreasing = F)
avg.mart <- tapply(BondData$Martinis, BondData$Actors.Name, mean)
avg.mart <- sort(avg.mart, decreasing = F)
avg.conq <- tapply(BondData$Conquests, BondData$Actors.Name, mean)
avg.conq <- sort(avg.conq, decreasing = F)
avg.bjb <- tapply(BondData$Bond.James.Bond, BondData$Actors.Name, mean)
avg.bjb <- sort(avg.bjb, decreasing = F)
avg.bor <- tapply(BondData$Vahan.adjusted * 8.35 / 7.94,
   BondData$Actors.Name, mean)
avg.bor <- sort(avg.bor, decreasing = F)
#
# pdf("dotPlot1.pdf", height = 9, width = 6)
par(mfrow = c(3, 2), mar = c(3, 5, 1.5, 0.5))
dotchart3(counts.actor, main = "(a) Number of movies", lty = 2,
cex.lab = 0.8, cex = 1, cex.main = 0.8)
dotchart3(avg.bor / 10 ^ 6, main = "(b) Average BOR in millions", lty = 2,
cex.lab = 0.8, cex = 1, cex.main = 0.8)
dotchart3(avg.kills, main = "(c) Average number of JB kills", lty = 2,
cex.lab = 0.8, cex = 1, xlim = c(4, 20), cex.main = 0.8)
dotchart3(avg.conq, main="(d) Average number of conquests", lty = 2,
cex.lab = 0.8, cex = 1, cex.main = 0.8)
dotchart3(avg.mart, main = "(e) Average number of martinis", lty = 2,
cex.lab = 0.8, cex = 1, cex.main = 0.8)
dotchart3(avg.bjb, main="(f) Average number of BJB", lty = 2,
cex.lab = 0.8, cex = 1, cex.main = 0.8)
#
# dev.off()

bymedian.actor <- with(BondData, reorder(BondData$Actors.Name,
-BondData$Mojo.Adjusted, median))
killShort <- ifelse(BondData$Bond.kills <= 5, "0-5",
                  ifelse(BondData$Bond.kills > 5 & BondData$Bond.kills <= 10,
                         "6-10", ">10"))
martiniShort <- ifelse(BondData$Martinis >= 2, ">1",
                      BondData$Martinis)
conqShort <- ifelse(BondData$Conquests >= 3, ">2",
                   BondData$Conquests)

bondDataShort <- cbind(BondData$Vahan.adjusted, BondData$Bond.James.Bond,
                        killShort, martiniShort, conqShort)
bymedian.kills <- with(as.data.frame(bondDataShort),
reorder(bondDataShort[, 3],
   -as.numeric(bondDataShort[, 1]), median))
bymedian.conquests <- with(as.data.frame(bondDataShort),
   reorder(bondDataShort[, 5],
   -as.numeric(bondDataShort[, 1]), median))
bymedian.martini <- with(as.data.frame(bondDataShort),
   reorder(bondDataShort[, 4],
   -as.numeric(bondDataShort[, 1]), median))
bymedian.bjb <- with(as.data.frame(bondDataShort),
   reorder(bondDataShort[, 2],
   as.numeric(bondDataShort[, 1]), median))

# pdf("boxPlot1.pdf", height = 6, width = 8)
par(mfrow = c(2, 2), mar = c(2, 4, 2, 1))
borMojo2014 <- BondData$Mojo.Adjusted / 10 ^ 6 * 8.35 / 7.94
boxplot(borMojo2014 ~ bymedian.kills,
   ylab = "BORs (in Millions)", ylim = c(50, 650),
   main = "(a) Number of JB kills", cex.lab = 1.2, cex.main = 1.2)

boxplot(borMojo2014 ~ bymedian.conquests,
   ylab = "BORs (in Millions)", ylim = c(50, 650),
   main = "(b) Number of conquests", cex.lab = 1.2, cex.main = 1.2)

boxplot(borMojo2014 ~ bymedian.martini,
   ylab = "BORs (in Millions)", ylim = c(50, 650),
   main = "(c) Number of martinis", cex.lab = 1.2, cex.main = 1.2)

boxplot(borMojo2014 ~ bymedian.bjb,
   ylab = "BORs (in Millions)", ylim = c(50, 650),
   main = "(d) Number of BJB", cex.lab = 1.2, cex.main = 1.2)
# dev.off()
library(car)

library(PairViz)
library(RColorBrewer)
par(mfrow = c(1, 1))
par.coord <- BondData[, 9:12]
```r
par.coord <- par.coord[, c(1, 2, 4, 3)]
colnames(par.coord) <- c("Conquests", "Martinis", "BJB", "Kills")
par.coord$Kills <- ifelse(par.coord$Kills <= 5, 1, 
                     ifelse(par.coord$Kills > 5 & par.coord$Kills <= 10, 2, 3))
par.coord$Martinis <- ifelse(par.coord$Martinis == 2, 2, 
                           par.coord$Martinis)
par.coord$Conquests <- ifelse(par.coord$Conquests >= 3, 3, 
                              par.coord$Conquests)
par.coord <- par.coord[order(par.coord[, 3]),]
par.coord <- par.coord[order(par.coord[, 2]),]
par.coord <- par.coord[order(par.coord[, 1]),]
par.coord <- par.coord[order(par.coord[, 4]),]

cols <- brewer.pal(3,"Set1")
cols <- paste(cols, 80,sep="")
cols <- cols[as.numeric(as.factor(par.coord[, 4]))]

ds <- factor_spreadout(par.coord)

rownames(ds$bars$Conquests) <- c("1", "2", ">2")
rownames(ds$bars$Martinis) <- c("0", "1", ">1")
rownames(ds$bars$Kills) <- c("0-5", "6-10", ">10")
# pdf("parCoordPlot1", height = 4, width = 7)
catpcp(ds$data,col = cols, lwd = 15,pcpbars = ds$bars, mar = c(2, 0.5, 0.5, 0.5),
        pcpbars.labels = TRUE, main = "", order = c(4, 1, 2, 3))
# dev.off()

library(heatmap.plus)
bond.mat <- t(as.matrix(cbind(BondData[, 9:12])))
```
movieNchar <- nchar(as.character(BondData[, 14]))
movieYearHeat <- substr(BondData[, 14], movieNchar - 5, movieNchar)
movieNameHeat <- substr(BondData[, 14], 1, movieNchar - 7)
# movieBOR2014 <- round(BondData[, 5] / BondData[, 6] * 8.35 / 10^6, 1)
colnames(bond.mat) <- paste(BondData$Actors.Name, movieYearHeat,
                           movieNameHeat)
rownames(bond.mat) <- c("Conquests", "Martinis", "Kills", "BJB")
# heatmap.plus(bond.mat, margins = c(17, 8))
library(gplots)

# pdf("heatMap1.pdf", height = 10, width = 10)
breaks = c(-0.5:6.5, seq(11.5, 26.5, by = 5))
col = brewer.pal(11, "PuOr")
heatmap.2(as.matrix(t(bond.mat)), dendrogram = "both",
          trace="none", margin = c(9, 32),
          lwid = c(0.5, 2), keysize = 1,
          cexRow = 1.7,
          breaks = breaks, col = col)
# dev.off()

bond.mat <- t(as.matrix(cbind(BondData[, c(9, 10, 12)],
                         sqrt(BondData[, 11]))))
colnames(bond.mat) <- paste(BondData$Actors.Name, movieYearHeat,
                         movieNameHeat)
rownames(bond.mat) <- c("Conquests", "Martinis", "BJB", "Kills")
breaks = -0.5:6.5
col = brewer.pal(7, "PuOr")
# pdf("heatMap2.pdf", height = 10, width = 10)
heatmap.2(as.matrix(t(bond.mat)), dendrogram = "both",
          trace = "none", margin = c(9, 32),
          lwid = c(0.5, 2), keysize = 1,
          cexRow = 1.7,
          breaks = breaks, col = col)
# dev.off()

library(vcd)
par(mfrow = c(1, 1))
mosaic.plot <- BondData[, 9:12]
mosaic.plot <- mosaic.plot[, c(1, 2, 4, 3)]
colnames(mosaic.plot) <- c("Conquests", "Martinis", "BJB", "Kills")

mosaic.plot$Kills <- ifelse(mosaic.plot$Kills <= 5, "0-5",
                            ifelse(mosaic.plot$Kills > 5 & mosaic.plot$Kills <= 10, "6-10", 
                                   ">10")
                           )
mosaic.plot$Martinis <- ifelse(mosaic.plot$Martinis >= 2, 
                               ">1", mosaic.plot$Martinis)
mosaic.plot$Conquests <- ifelse(mosaic.plot$Conquests >= 3, 
                                ">2", mosaic.plot$Conquests)

table.mosaic <- table(mosaic.plot[, 2:4])
table.mosaic <- table.mosaic[, , c(1, 3, 2)]
table.mosaic <- table.mosaic[, c(4, 1, 2, 3)]
table.mosaic <- table.mosaic[c(3, 1, 2), c(2, 3, 1), c(1, 3, 2), ]

# mosaicplot(table.mosaic, color = TRUE, shade = T, main = "Mosaic Plot")
mosaic(table.mosaic, color = TRUE, shade = T)
assoc(table.mosaic, shade = TRUE)
########################################################

library(maps)
library(mapdata)
library(maptools)
library(RColorBrewer)

data(wrld_simpl)
country.current <- wrld_simpl$NAME
world.map <- map("world2Hires", plot = FALSE)
country.past <- world.map$names
visit <- read.csv("Countries Visited.csv")
contry.clr <- brewer.pal(7, "YlOrBr")

visit.ussr <- visit[, 1:16]
visit.ussr <- as.factor(as.vector(t(visit.ussr)))
visit.ussr <- as.data.frame(table(visit.ussr))
visit.ussr <- visit.ussr[-1,]
layout(matrix(c(1, 2, 3), 3, 1, byrow = T),
       widths = c(5, 5), heights = c(10, 10, 10))
par(mar = c(0, 0, 0, 0))
map()
for (i in 1:nrow(visit.ussr)) {

map("world", visit.ussr[i, 1], fill = T, 
    col = contry.clr[visit.ussr[i, 2]], add = T)
}
map.axes()

legend(x = -170, y = 0, 
    c("1", "2", "3", "4", "5", "6", "7"), 
    fill = brewer.pal(7, "YlOrBr"), horiz = F)

map("world", ylim = c(7, 35), xlim = c(-92, -60))
for (i in 1:nrow(visit.ussr)) {
    map("world", visit.ussr[i, 1], fill = T, 
        col = contry.clr[visit.ussr[i, 2]], add = T)
}
map.axes()

map("world", ylim = c(30, 75), xlim = c(-25, 50))
for (i in 1:nrow(visit.ussr)) {
    map("world", visit.ussr[i, 1], fill = T, 
        col = contry.clr[visit.ussr[i, 2]], add = T)
}
map.axes()

########################################################

col.map <- rep("#FFFFFF", length(wrld_simpl$NAME ))

visit.ussr.post <- visit[, 17:22]
visit.ussr.post <- as.factor(as.vector(t(visit.ussr.post )))
visit.ussr.post <- as.data.frame(table(visit.ussr.post))
visit.ussr.post <- visit.ussr.post[-1,]
contry.clr <- brewer.pal(7, "YlOrBr")
par(mar = c(0, 0, 0, 0))
for (i in 1:nrow(visit.ussr.post)) {
  position <- which(wrld_simpl$NAME == as.character(visit.ussr.post[i, 1]))
  col.map[position] = contry.clr[visit.ussr.post[i, 2]]
}

layout(matrix(c(1, 1, 2, 3), 2, 2, byrow = T),
       widths = c(5, 5), heights = c(10, 10))
par(mar = c(2, 2, 0.3, 0.3))
plot(wrld_simpl, col = col.map, axes = F)
map.axes()
legend(x = -200, y = 40,
       c("1", "2", "3", "4", "5", "6", "7"),
       fill = brewer.pal(7, "YlOrBr"), horiz = F)

plot(wrld_simpl, col = col.map, axes = F,
     ylim = c(15, 32), xlim = c(-92,-60))
map.axes()

plot(wrld_simpl, col = col.map,
     ylim = c(30, 65), xlim = c(-20, 50))
map.axes()

########################################################

visit <- read.csv("Countries Visited.csv")
visit.ussr <- visit[, 1:16]
visit.ussr <- as.factor(as.vector(t(visit.ussr)))

bo.adj <- rep(0, 16 * 8)
bo.adj[(0:7) * 16 + 1:16] <- BondData$Vahan.adjusted[1:16]

rev.country <- cbind(as.character(visit.ussr), bo.adj)
mean.country <- tapply(as.numeric(rev.country[, 2]), rev.country[, 1], mean)
mean.country <- mean.country[-1] / 10 ^ 6
breaks <- c(80, 100, 120, 160, 200, 240, 330)
m.class <- cut(mean.country, breaks)
m.col <- ifelse(mean.country <= breaks[2], brewer.pal(5, "YlGn")[1],
    ifelse(mean.country > breaks[2] & mean.country <= breaks[3],
        brewer.pal(6, "YlGn")[2],
        ifelse(mean.country > breaks[3] & mean.country <= breaks[4],
            brewer.pal(6, "YlGn")[3],
            ifelse(mean.country > breaks[4] & mean.country <= breaks[5],
                brewer.pal(6, "YlGn")[4],
                ifelse(mean.country > breaks[5] & mean.country <= breaks[6],
                    brewer.pal(6, "YlGn")[5],
                    ifelse(mean.country > breaks[6] & mean.country <= breaks[7],
                        brewer.pal(6, "YlGn")[6],
                        ""))))))

par(mfrow = c(2, 1), mar = c(0, 0, 0, 0))
map()
for (i in 1:length(mean.country)) {
    map("world", names(mean.country)[i], fill = T,
        col = m.col[i], add = T)
}
legend(x = -170, y = -10, legend = levels(m.class), fill = brewer.pal(6, "YlGn"))
visit <- read.csv("Countries Visited.csv")
visit.ussr.post <- visit[, 17:22]
visit.ussr.post <- as.factor(as.vector(t(visit.ussr.post)))

bo.adj <- rep(0, 6 * 8)

rev.country <- cbind(as.character(visit.ussr.post), bo.adj)
mean.country <- tapply(as.numeric(rev.country[, 2]), rev.country[, 1], mean)
mean.country <- mean.country[-1] / 10 ^ 6
m.col <- ifelse(mean.country <= breaks[2], brewer.pal(5, "YlGn")[1],
    ifelse(mean.country > breaks[2] & mean.country <= breaks[3],
        brewer.pal(6, "YlGn")[2],
        ifelse(mean.country > breaks[3] & mean.country <= breaks[4],
            brewer.pal(6, "YlGn")[3],
            ifelse(mean.country > breaks[4] & mean.country <= breaks[5],
                brewer.pal(6, "YlGn")[4],
                ifelse(mean.country > breaks[5] & mean.country <= breaks[6],
                    brewer.pal(6, "YlGn")[5],
                    ifelse(mean.country > breaks[6] & mean.country <= breaks[7],
                        brewer.pal(6, "YlGn")[6],
                        ""))))))

col.map <- rep("#FFFFFF", length(wrld_simpl$NAME ))
contry.clr <- brewer.pal(5, "YlGn")

for (i in 1:length(mean.country)) {
    position <- which(wrld_simpl$NAME == names(mean.country)[i])
    col.map[position] = m.col[i]
}
par(mar = c(0, 0, 0, 0))
plot(wrld_simpl, col = col.map, axes = F)

legend(x = -170, y = -10, legend = levels(m.class), fill = brewer.pal(6, "YlGn"))

*******************************************************************************
B.2 R Code for Chapter 3

setwd("C:/Users/Vahan/Desktop/Masters/MS Project/Report")

ReplicateSummary <- function(NEWBOND1 = 1, NEWBOND7 = 1) {

  BondData <- read.csv("MS_project_Data2.csv")
  movieDate <- as.Date(BondData[, 13])

  #model 1
  CONNERY <- ifelse(as.character(BondData$Actors.Name) == "Connery",
                   1, 0)[1:16]
  LAZENBY <- ifelse(as.character(BondData$Actors.Name) == "Lazenby",
                   1, 0)[1:16]
  MOORE <- ifelse(as.character(BondData$Actors.Name) == "Moore",
                   1, 0)[1:16]
  DALTON <- ifelse(as.character(BondData$Actors.Name) == "Dalton",
                   1, 0)[1:16]
  BROSNAN <- ifelse(as.character(BondData$Actors.Name) == "Brosnan",
                   1, 0)[1:16]
  CRAIG <- ifelse(as.character(BondData$Actors.Name) == "Craig",
                 1, 0)[1:16]

  ACTREND <- rep(1, 22)[1:16]
  NEWBOND <- rep(1, 22)[1:16]
  k <- 1
  for (i in 2:16) {
    k <- ifelse(BondData$Actors.Name[i] == BondData$Actors.Name[i-1],
                 k + 1, 1)
    ACTREND[i] <- k
    if (BondData$Actors.Name[i] == BondData$Actors.Name[i-1]) {
      NEWBOND[i] <- 0
    }
  }
}
ACTREND[7] <- 6
ACTRENSQ <- ACTRENDSQ ^ 2
NEWBOND[1] <- NEWBOND1
NEWBOND[7] <- NEWBOND7

movieYear <- as.numeric(substring(movieDate, 1, 4))
avgTicket <- BondData$Average.Ticket.Price
CPIindex <- read.csv("CPIindex.csv")
CPI <- sapply(1:22, function(x)
  CPIindex$CPI[movieYear[x] == CPIindex$yearCPI])
mojoAdj <- read.csv("mojoAdj.csv")

CPIadj63 <- BondData$Mojo.Unadjusted * CPI[1] / CPI
CPIadj62 <- BondData$Mojo.Unadjusted * 30.2 / CPI

Ticketadj63 <- BondData$Mojo.Unadjusted * avgTicket[1] / avgTicket
Ticketadj62 <- BondData$Mojo.Unadjusted * 0.7 / avgTicket

respData <- log(cbind(mojoAdj[, 2:3], CPIadj63, CPIadj62,
  Ticketadj63, Ticketadj62) / 10 ^ 6)

modelX <- as.data.frame(cbind(CONNERY, LAZENBY, MOORE, ACTREND,
  ACTRENSQ, NEWBOND))[1:16, ]

library(orcutt)
summaryCoef1 <- NULL
summaryorcutt1 <- NULL
for (j in 1:6) {
  lmout <- (lm(respData[j][1:16, ] ~ CONNERY + LAZENBY + MOORE +
             ACTREND + ACTRENSQ + NEWBOND))
  summaryCoef1[[j]] <- (lm(respData[j][1:16, ] ~ CONNERY +
                         LAZENBY + MOORE +
                         ACTREND + ACTRENSQ + NEWBOND))$coef
  summaryorcutt1[[j]] <- (cochrane.orcutt(lmout)$Cochrane.Orcutt)$coef[, 1]
}

ACTREND <- ACTREND - 1
ACTRENSQ <- ACTREND ^ 2

summaryCoef0 <- NULL
summaryorcutt0 <- NULL
for (j in 1:6) {
  lmout <- (lm(respData[j][1:16, ] ~ CONNERY + LAZENBY + MOORE +
               ACTREND + ACTRENSQ + NEWBOND))
  summaryCoef0[[j]] <- (lm(respData[j][1:16, ] ~ CONNERY +
                           LAZENBY + MOORE +
                           ACTREND + ACTRENSQ + NEWBOND))$coef
  summaryorcutt0[[j]] <- (cochrane.orcutt(lmout)$Cochrane.Orcutt)$coef[, 1]
}

ActrendStart1 <- t(matrix(unlist(summaryCoef1), nrow = 7))
rownames(ActrendStart1) <- names(respData)
colnames(ActrendStart1) <- names((lm(respData[1][1:16, ] ~ CONNERY +
LAZENBY + MOORE +
ACTREND + ACTRENSQ + NEWBOND)$coef)

ActrendStart1Orcutt <- t(matrix(unlist(summaryorcutt1), nrow = 7))
rownames(ActrendStart1Orcutt) <- names(respData)
colnames(ActrendStart1Orcutt) <- names((lm(respData[1][1:16,] ~ CONNERY +
LAZENBY + MOORE +
ACTREND + ACTRENSQ +
NEWBOND))$coef)

ActrendStart0 <- t(matrix(unlist(summaryCoef0), nrow = 7))
rownames(ActrendStart0) <- names(respData)
colnames(ActrendStart0) <- names((lm(respData[1][1:16,] ~ CONNERY +
LAZENBY + MOORE +
ACTREND + ACTRENSQ + NEWBOND))$coef)

ActrendStart0Orcutt <- t(matrix(unlist(summaryorcutt0), nrow = 7))
rownames(ActrendStart0Orcutt) <- names(respData)
colnames(ActrendStart0Orcutt) <- names((lm(respData[1][1:16,] ~ CONNERY +
LAZENBY + MOORE +
ACTREND + ACTRENSQ +
NEWBOND))$coef)

TotalSummary <- list(ActrendStart0 = ActrendStart0,
ActrendStart0Orcutt = ActrendStart0Orcutt,
ActrendStart1 = ActrendStart1,
ActrendStart1Orcutt = ActrendStart1Orcutt

baimCoef <- c(1.179, 1.08, 0.9056, 0.333, 0.7835, -0.0908, 0.7807)

modelSSE <- sapply(1:4, function(x)
   (t(t(TotalSummary[[x]]) - baimCoef)) ^ 2
rownames(modelSSE) <- names(respData)
colnames(modelSSE) <- c("ActrendStart0", "ActrendStart0Orcutt",
   "ActrendStart1", "ActrendStart1Orcutt")

return(list(TotalSummary = TotalSummary, modelSSE = modelSSE))

NewBond11 <- ReplicateSummary(1, 1)$TotalSummary
NewBond11SSE <- matrix(ReplicateSummary(1, 1)$modelSSE,
   ncol = 1)
NewBond10 <- ReplicateSummary(1, 0)$TotalSummary
NewBond10SSE <- matrix(ReplicateSummary(1, 0)$modelSSE,
   ncol = 1)
NewBond01 <- ReplicateSummary(0, 1)$TotalSummary
NewBond01SSE <- matrix(ReplicateSummary(0, 1)$modelSSE,
   ncol = 1)
NewBond00 <- ReplicateSummary(0, 0)$TotalSummary
NewBond00SSE <- matrix(ReplicateSummary(0, 0)$modelSSE,
ncol = 1)

parCoordData11 <- cbind(rbind(NewBond11[[1]], NewBond11[[2]],
                        NewBond11[[3]], NewBond11[[4]]), NewBond11SSE)
parCoordData10 <- cbind(rbind(NewBond10[[1]], NewBond10[[2]],
                        NewBond10[[3]], NewBond10[[4]]), NewBond10SSE)
parCoordData01 <- cbind(rbind(NewBond01[[1]], NewBond01[[2]],
                        NewBond01[[3]], NewBond01[[4]]), NewBond01SSE)
parCoordData00 <- cbind(rbind(NewBond00[[1]], NewBond00[[2]],
                        NewBond00[[3]], NewBond00[[4]]), NewBond00SSE)

parCoordData <- rbind(parCoordData11, parCoordData10,
                       parCoordData01, parCoordData00,
                       c(1.179, 1.08, 0.9056, 0.333, 0.7835,
                         -0.0908, 0.7807, 0))

colnames(parCoordData)[8] <- "SSE"
library(RColorBrewer)
clr <- brewer.pal(3, "Pastel1")
plotClr <- rep(c(clr[1], clr[1], clr[2], clr[2], clr[3], clr[3]), 16)
plotClr <- rep(rep(c(clr[1], clr[2]), each = 6), 8)

plotClr[97] <- "#000000"
plotClr[64] <- "red"
panel.myplot <- function(..., common.scale) {
  panel.parallel(..., common.scale = TRUE)
  panel.abline(h = -min(parCoordData) / diff(range(parCoordData)), lty = 2)
}

pdf("repParCoord1.pdf", height = 5, width = 8)
parallelplot(~ parCoordData[c(1:63, 65:96, 64, 97), ],
col = plotClr[c(1:63, 65:96, 64, 97)],
lwd = c(rep(2, 95), 3, 3),
horizontal.axis = FALSE,
panel = panel.myplot,
scales = list(y = list(lim = c(-0.05, 1.05))),
var.label = T)
dev.off()

##########################################################

CPIindex <- read.csv("CPIindex.csv")
CPI <- sapply(1:22, function(x)
    CPIindex$CPI[movieYear[x] == CPIindex$yearCPI])
CPIadj62 <- BondData$Mojo.Unadjusted * 30.2 / CPI
logBoxOffice <- log(CPIadj62 / 10 ^ 6)

model1 <- as.data.frame(cbind(logBoxOffice, CONNERY, LAZENBY,
    MOORE, DALTON, BROSnan, CRAIG,
    ACTRENDS, ACTRENDSQ, NEWBOND))
model1Old <- model1[1:16, -(5:7)]
model1New <- model1[, -5]

lmOld1 <- lm(logBoxOffice ~ ., model1Old)
Model1Coef <- lmOld1$coefficients
Model1T <- summary(lmOld1)$coefficients[, 3]
Model1F <- c(summary(lmOld1)$r.squared, summary(lmOld1)$adj.r.squared,
    summary(lmOld1)$fstatistic[1], durbinWatsonTest(lmOld1)$dw)
names(Model1F) <- c("R Squared", "Adj. R Squared",
    "F Statistic", "Durbin Watson")
BaimModel1Coef <- c(1.179, 1.08, 0.9056, 0.333, 0.7835, -0.0908, 0.7807)
BaimModel1T <- c(1.76, 3.231, 1.763, 1.097, 2.121, -2.135, 1.79)
BaimModel1F <- c(0.77, 0.61, 4.99, 2.22)
summary(lmOld1)

##########################################################

library(lattice)
library(RColorBrewer)

pdf("repDotplot1.pdf", height = 5, width = 12)
plot1 <- dotplot(reorder(names(Model1T), abs(Model1T)) ~ Model1Coef + BaimModel1Coef | "Coefficient",
col = brewer.pal(3, "Set1"), pch = 0:1, xlab = ",",
scales = list(y = list(cex = 1.2), x = list(cex = 1.2)))
plot2 <- dotplot(reorder(names(Model1T), abs(Model1T)) ~ Model1T + BaimModel1T | "t-value",
col = brewer.pal(3, "Set1"), pch = 0:1, xlab = ",",
panel = function(...) {
  panel.abline(v = qt(0.025, nrow(model1)), lty = 2)
  panel.abline(v = qt(0.975, nrow(model1)), lty = 2)
  panel.dotplot(...)
},
scales = list(y = list(cex = 1.2), x = list(cex = 1.2)))
plot3 <- dotplot(reorder(names(Model1F), abs(Model1F)) ~ Model1F + BaimModel1F | "ANOVA",
col = brewer.pal(3, "Set1"), pch = 0:1, xlab = ",",
scales = list(y = list(cex = 1.2), x = list(cex = 1.2)))

print(plot1, position = c(0, 0, 0.36, 1), more = TRUE)
print(plot2, position = c(0.32, 0, 0.68, 1), more = TRUE)
print(plot3, position = c(0.64, 0, 1., 1))
dev.off()

##########################################################
# model 2

```r
rating <- read.csv("model 2.csv")
ONESTAR <- ifelse(rating$Halliwell == 1, 1, 0)[1:16]
TWOSTAR <- ifelse(rating$Halliwell == 2, 1, 0)[1:16]
THREESTAR <- ifelse(rating$Halliwell == 3, 1, 0)[1:16]
NOMOSCAR <- rating$nom[1:16]
WONOSCAR <- rating$win[1:16]

movieYear <- as.numeric(substring(movieDate, 1, 4))
avgTicket <- BondData$Average.Ticket.Price
CPIindex <- read.csv("CPIindex.csv")
CPI <- sapply(1:22, function(x)
    CPIindex$CPI[movieYear[x] == CPIindex$yearCPI])
mojoAdj <- read.csv("mojoAdj.csv")

CPIadj63 <- BondData$Mojo.Unadjusted * CPI[1] / CPI
CPIadj62 <- BondData$Mojo.Unadjusted * 30.2 / CPI
Ticketadj63 <- BondData$Mojo.Unadjusted * avgTicket[1] / avgTicket
Ticketadj62 <- BondData$Mojo.Unadjusted * 0.7 / avgTicket

respData <- log(cbind(mojoAdj[, 2:3], CPIadj63, CPIadj62,
    Ticketadj63, Ticketadj62) / 10 ^ 6)

library(orcutt)
```
summaryCoef1 <- NULL
summaryorcutt1 <- NULL
for (j in 1:6) {
  lmout <- (lm(respData[j][1:16, ] ~ ONESTAR + TWOSTAR + THREESTAR +
               WONOSCAR + NOMOSCAR))
  summaryCoef1[[j]] <- (lm(respData[j][1:16, ] ~ ONESTAR +
                         TWOSTAR + THREESTAR +
                         WONOSCAR + NOMOSCAR))$coef
  summaryorcutt1[[j]] <- (cochrane.orcutt(lmout)$Cochrane.Orcutt)$coef[, 1]
}

ActrendStart1 <- t(matrix(unlist(summaryCoef1), nrow = 6))
rownames(ActrendStart1) <- names(respData)
colnames(ActrendStart1) <- names((lm(respData[1][1:16, ] ~ ONESTAR +
                                   TWOSTAR + THREESTAR +
                                   WONOSCAR + NOMOSCAR))$coef)

ActrendStart1Orcutt <- t(matrix(unlist(summaryorcutt1), nrow = 6))
rownames(ActrendStart1Orcutt) <- names(respData)
colnames(ActrendStart1Orcutt) <- names((lm(respData[1][1:16, ] ~ ONESTAR +
                                         TWOSTAR + THREESTAR +
                                         WONOSCAR + NOMOSCAR))$coef)

BaimModel2Coef <- c(2.8169, -0.30032, 0.25894, 0.45973, 1.1211, 0.45023)
parCoordData <- rbind(ActrendStart1, ActrendStart1Orcutt, BaimModel2Coef)

SSE <- NULL
for (i in 1:12) {
  SSE[i] <- sum((parCoordData[i, ] - parCoordData[13, ]) ^ 2)
}
parCoordData <- cbind(parCoordData, c(SSE, 0))
colnames(parCoordData)[7] <- "SSE"
clr <- brewer.pal(3, "Pastel1")
plotClr <- rep(c(clr[1], clr[2]), each = 6)

panel.myplot <- function(..., common.scale) {
  panel.parallel(..., common.scale = TRUE)
}

clr <- brewer.pal(3, "Pastel1")
plotClr <- rep(rep(c(clr[1], clr[2]), each = 6), 1)

plotClr[13] <- "#000000"
plotClr[5] <- "red"
plotClr[4] <- "orange"
panel.myplot <- function(..., common.scale) {
  panel.parallel(..., common.scale = TRUE)
  panel.abline(h = -min(parCoordData) / diff(range(parCoordData)), lty = 2)
}

pdf("repParCoord2.pdf", height = 5, width = 8)
parallelplot( ~ parCoordData[c(1:3, 6:12, 4, 5, 13)],
  col = plotClr[c(1:3, 6:12, 4, 5, 13)],
  lwd = c(rep(1, 10), 2, 2, 2),
  horizontal.axis = FALSE,
  panel = panel.myplot,
  scales = list(y = list(lim = c(-0.05, 1.05))),
  var.label = T)
dev.off()
avgTicket <- BondData$Average.Ticket.Price

logBoxOffice <- log(10 ^ (-6) * BondData$Mojo.Unadjusted *
    avgTicket[1] / avgTicket)[1:16]

model2 <- as.data.frame(cbind(logBoxOffice, ONESTAR, TWOSTAR,
    THREESTAR, WONOSCAR, NOMOSCAR))

model2Old <- model2[1:16,]
model2New <- model2[,]

lmOld2 <- lm(logBoxOffice ~ ., model2Old)
Model2Coef <- lmOld2$coefficients
Model2T <- summary(lmOld2)$coefficients[, 3]
Model2F <- c(summary(lmOld2)$r.squared, summary(lmOld2)$adj.r.squared,
    summary(lmOld2)$fstatistic[1], durbinWatsonTest(lmOld2)$dw)

names(Model2F) <- c("R Squared", "Adj. R Squared",
    "F Statistic", "Durbin Watson")

BaimModel2Coef <- c(2.8169, -0.30032, 0.25894, 0.45973, 1.1211, 0.45023)
BaimModel2T <- c(15.51, -1.43, 1.03, 1.551, 4.105, 2.124)
BaimModel2F <- c(0.78, 0.38, 7.28, 2.07)

summary(lmOld2)

#########################################################################

pdf("repDotplot2.pdf", height = 5, width = 12)

plot1 <- dotplot(reorder(names(Model2T), abs(Model2T)) ~ Model2Coef + BaimModel2Coef | "Coefficient",
    col = brewer.pal(3, "Set1"), pch = 0:1, xlab = "",
    scales = list(y = list(cex = 1.2), x = list(cex = 1.2)))
plot2 <- dotplot(reorder(names(Model2T), abs(Model2T)) ~ Model2T + BaimModel2T | "t-value", 
col = brewer.pal(3, "Set1"), pch = 0:1, xlab = "", 
panel = function(...) {
  panel.abline(v = qt(0.025, nrow(model2)), lty = 2)
  panel.abline(v = qt(0.975, nrow(model2)), lty = 2)
  panel.dotplot(...)
}, 
scales = list(y = list(cex = 1.2), x = list(cex = 1.2)))

plot3 <- dotplot(reorder(names(Model2F), abs(Model2F)) ~ Model2F + BaimModel2F | "ANOVA", 
col = brewer.pal(3, "Set1"), pch = 0:1, xlab = "", 
scales = list(y = list(cex = 1.2), x = list(cex = 1.2)))

print(plot1, position = c(0, 0, 0.36, 1), more = TRUE)
print(plot2, position = c(0.32, 0, 0.68, 1), more = TRUE)
print(plot3, position = c(0.64, 0, 1, 1))
dev.off()

#********************

# model 3
SEQUENCE <- 1:16
SEQUENCE <- 0:15
movieYear <- as.numeric(substring(movieDate, 1, 4))[1:16]
movieMonth <- as.numeric(substring(movieDate, 6, 7))[1:16]
movieYear.month <- movieYear + movieMonth / 12
GAP <- c(0, diff(movieYear))[1:16]
GAPSQ <- GAP ^ 2
COLDWAR <- ifelse(movieYear.month < 1989, 0, 1)[1:16]
# GAP <- c(0, diff(movieYear.month)) [1:16]
# GAPSQ <- GAP ^ 2

ReplicateSummary3 <- function(GAP, SEQUENCE) {
  GAPSQ <- GAP ^ 2
  library(orcutt)
  summaryCoef1 <- NULL
  summaryorcutt1 <- NULL
  for (j in 1:6) {
    lmout <- (lm(respData[j][1:16, ] ~ SEQUENCE + GAP + GAPSQ +
        COLDWAR))
    summaryCoef1[[j]] <- (lm(respData[j][1:16, ] ~ SEQUENCE + GAP + GAPSQ +
        COLDWAR))$coef
    summaryorcutt1[[j]] <- (cochrane.orcutt(lmout)$Cochrane.Orcutt)$coef[, 1]
  }
  return(rbind(t(matrix(unlist(summaryCoef1), nrow = 5)),
                t(matrix(unlist(summaryorcutt1), nrow = 5))))
}

BaimModel3Coef <- c(4.3296, -0.094893, -0.36739, 0.10431, -0.1099)
parCoordData <- rbind(ReplicateSummary3(1:16, c(0, diff(movieYear)))[1:16]),
  ReplicateSummary3(1:16, c(0, diff(movieYear.month)))[1:16]),
  ReplicateSummary3(0:15, c(0, diff(movieYear)))[1:16]),
  ReplicateSummary3(0:15, c(0, diff(movieYear.month)))[1:16]),
  BaimModel3Coef)
SSE <- NULL
for (i in 1:48) {
    SSE[i] <- sum((parCoordData[i, ] - parCoordData[49, ])^2)
}
parCoordData <- cbind(parCoordData, c(SSE, 0))
colnames(parCoordData) <- c("(intercept)", "SEQUENCE",
    "GAP", "GAPSQ", "COLDWAR", "SSE")

clr <- brewer.pal(3, "Pastel1")
plotClr <- rep(rep(c(clr[1], clr[2]), each = 6), 8)
plotClr[49] <- "#000000"
plotClr[8] <- "blue"
plotClr[10] <- "orange"
panel.myplot <- function(..., common.scale) {
    panel.parallel(..., common.scale = TRUE)
    panel.abline(h = -min(parCoordData) / diff(range(parCoordData)), lty = 2)
}

pdf("repParCoord3.pdf", height = 5, width = 8)
parallelplot(~ parCoordData[c(1:7, 9, 11:48, 10, 8, 49), ],
    col = plotClr[c(1:7, 9, 11:48, 10, 8, 49)],
    lwd = c(rep(1, 46), 2, 2, 2),
    horizontal.axis = FALSE,
    panel = panel.myplot,
    scales = list(y = list(ylim = c(-0.05, 1.05))))
dev.off()
library(orcutt)

logBoxOffice <- log(1 / 10 ^ 6 * read.csv("mojoAdj.csv")[, 3])

model3 <- as.data.frame(cbind(logBoxOffice, SEQUENCE, GAP, GAPSQ, COLDWAR))

model3Old <- model3[1:16, ]
model3New <- model3[, ]

lmOld3 <- lm(logBoxOffice ~ ., model3Old)

lmOld3C <- cochrane.orcutt(lmOld3)$Cochrane.Orcutt

Model3Coef <- lmOld3C$coef[, 1]
names(Model3Coef) <- substr(names(Model3Coef), 3, 13)

Model3T <- lmOld3C$coef[, 3]
names(Model3T) <- substr(names(Model3T), 3, 13)

Model3F <- c(summary(lmOld3)$r.squared, summary(lmOld3)$adj.r.squared,
summary(lmOld3)$fstatistic[1], durbinWatsonTest(lmOld3)$dw)
names(Model3F) <- c("R Squared", "Adj. R Squared",
"F Statistic", "Durbin Watson")

BaimModel3Coef <- c(4.3296, -0.094893, -0.36739, 0.10431, -0.1099)

BaimModel3T <- c(3.906, -2.213, -0.311, 0.346, -0.218)

BaimModel3F <- c(0.49, 0.3, 2.64, 1.83)

cochrane.orcutt(lmOld3)$Cochrane.Orcutt

########################################################################

pdf("repDotplot3.pdf", height = 5, width = 12)

plot1 <- dotplot(reorder(names(Model3T), abs(Model3T)) ~ Model3Coef + BaimModel3Coef | "Coefficient",
col = brewer.pal(3, "Set1"), pch = 0:1, xlab = "",
  ylab = "Coefficient", offset = 0.2, cex.axis = 0.7, par.settings = list(axis.las = 1, axis.ticks = TRUE))
scales = list(y = list(cex = 1.2), x = list(cex = 1.2))

plot2 <- dotplot(reorder(names(Model3T), abs(Model3T)) ~ Model3T + BaimModel3T | "t-value",
      col = brewer.pal(3, "Set1"), pch = 0:1, xlab = "",
      panel = function(...) {
        panel.abline(v = qt(0.025, nrow(model3)), lty = 2)
        panel.abline(v = qt(0.975, nrow(model3)), lty = 2)
        panel.dotplot(...)
      },
      scales = list(y = list(cex = 1.2), x = list(cex = 1.2)))

plot3 <- dotplot(reorder(names(Model3F), abs(Model3F)) ~ Model3F + BaimModel3F | "ANOVA",
      col = brewer.pal(3, "Set1"), pch = 0:1, xlab = "",
      scales = list(y = list(cex = 1.2), x = list(cex = 1.2)))

print(plot1, position = c(0, 0, 0.36, 1), more = TRUE)
print(plot2, position = c(0.32, 0, 0.68, 1), more = TRUE)
print(plot3, position = c(0.64, 0, 1, 1))
dev.off()

########################################################################

CPIindex <- read.csv("CPIindex.csv")
CPI.year <- CPIindex[, 1]
movieYear <- as.numeric(substring(movieDate, 1, 4))
CPImovie <- sapply(1:22, function(x)
  as.numeric(as.character(CPIindex[which(CPI.year == movieYear[x]), 2])))

# PRICE <- BondData$Average.Ticket.Price * 30.2 / CPImovie
PRICESQ <- PRICE ^ 2

PCEindex <- read.csv("PCEindex.csv")
PCE.year <- as.numeric(substring(PCEindex[, 1], 1, 4))
moviePCE <- sapply(1:22, function(x)
    as.numeric(as.character(PCEindex[which(PCE.year == movieYear[x]), 2])))

PCEMOVIES <- moviePCE * CPImovie[1] / CPImovie
# PCEMOVIES <- moviePCE * 10.043 / CPImovie
PCEMOVIESQ <- PCEMOVIES ^ 2

totalAdmission <- read.csv("movie admission.csv")
totYear <- totalAdmission[, 1]
TOTADM <- sapply(1:22, function(x)
    as.numeric(as.character(totalAdmission[which(totYear == movieYear[x]), 2])))

numReleases <- read.csv("releases.csv")[-1, ]
tableReleases <- table(numReleases$YEAR)
RELEASES <- sapply(1:22, function(x)
    tableReleases[which(as.numeric(names(tableReleases)) == movieYear[x])])

model4 <- cbind(log(CPIadj62 / 10 ^ 6), PRICE, PRICESQ, PCEMOVIES, PCEMOVIESQ,
    TOTADM / 10 ^ 3, RELEASES)[1:16, ]
colnames(model4)[c(1, 6)] <- c("CPIadj62", "TOTADM")
TOTADM1 = log(TOTADM * 10 ^ 6)
TOTADM2 = TOTADM / 10 ^ 3

lm(respData[, 1] ~ PRICE + PRICESQ + PCEMOVIES +
    PCEMOVIESQ + TOTADM2 + RELEASES)
summaryCoef1 <- NULL
summaryorcutt1 <- NULL
summaryCoef2 <- NULL
summaryorcutt2 <- NULL
for (j in 1:6) {
    lmout1 <- lm(respData[, j][1:16] ~ PRICE[1:16] +
                  PRICESQ[1:16] + PCEMOVIES[1:16] +
    summaryCoef1[[j]] <- (lm(respData[, j][1:16] ~ PRICE[1:16] +
                              PRICESQ[1:16] + PCEMOVIES[1:16] +
                              PCEMOVIESQ[1:16] + TOTADM1[1:16] +
                              RELEASES[1:16]))$coef
    summaryorcutt1[[j]] <- (cochrane.orcutt(lmout1)$Cochrane.Orcutt)$coef[, 1]

    lmout2 <- lm(respData[, j][1:16] ~ PRICE[1:16] +
                  PRICESQ[1:16] + PCEMOVIES[1:16] +
    summaryCoef2[[j]] <- (lm(respData[, j][1:16] ~ PRICE[1:16] +
                              PRICESQ[1:16] + PCEMOVIES[1:16] +
                              PCEMOVIESQ[1:16] + TOTADM2[1:16] +
                              RELEASES[1:16]))$coef
    summaryorcutt2[[j]] <- (cochrane.orcutt(lmout2)$Cochrane.Orcutt)$coef[, 1]
}
BaimModel4Coef <- c(-22.368, 11.178, -1.8922, 6.332, -0.7743, -3.6111, 0.0046659)

parCoordData <- rbind(t(matrix(unlist(summaryCoef1), nrow = 7)),
                      t(matrix(unlist(summaryorcutt1), nrow = 7)),
                      t(matrix(unlist(summaryCoef2), nrow = 7)),
                      t(matrix(unlist(summaryorcutt2), nrow = 7)),
                      BaimModel4Coef)

SSE <- NULL
for (i in 1:24) {
  SSE[i] <- sum((parCoordData[, i] - BaimModel4Coef) ^ 2)
}

parCoordData <- cbind(parCoordData, c(SSE / 1000, 0))

colnames(parCoordData) <- c("(Intercept)", "PRICE", "PRICESQ", "PCEMOVIES", "PCEMOVIESQ", "TOTADM", "RELEASES", "SSE/1000")

# BaimModel4Coef %*% c(1, 1.2, 1.44, 4, 16, 0, 200)
clr <- brewer.pal(3, "Pastel1")
plotClr <- rep(rep(c(clr[1], clr[2]), each = 6), 2)

panel.myplot <- function(..., common.scale) {
  panel.parallel(..., common.scale = TRUE)
  panel.abline(h = -min(parCoordData) / diff(range(parCoordData)), lty = 2)
}

plotClr[25] <- "#000000"
plotClr[4] <- "red"

pdf("repParCoord4.pdf", height = 5, width = 8)

parallelplot(~ parCoordData[c(1:3, 5:24, 4, 25), ],
            col = plotClr[c(1:3, 5:24, 4, 25)],
            lwd = c(rep(1, 23), 1.5, 1.5),
            horizontal.axis = FALSE,
            panel = panel.myplot,
            scales = list(y = list(lim = c(-0.05, 1.05))),
            var.label = T)

dev.off()

##########################################################

logBoxOffice <- log(1 / 10 ^ 6 * BondData$Mojo.Unadjusted * 30.2 / CPI)
model4 <- as.data.frame(cbind(logBoxOffice, PRICE, PRICESQ,
                              PCEMOVIES, PCEMOVIESQ, TOTADM,
                              as.numeric(RELEASES)))

colnames(model4)[7] <- "RELEASES"

model4Old <- model4[1:16, ]
model4New <- model4[, ]

lm0ld4 <- lm(logBoxOffice ~ ., model4Old)
Model4Coef <- lm0ld4$coefficients
Model4T <- summary(lm0ld4)$coefficients[, 3]

Model4F <- c(summary(lm0ld4)$r.squared, summary(lm0ld4)$adj.r.squared,
              summary(lm0ld4)$fstatistic[1], durbinWatsonTest(lm0ld4)$dw)

names(Model4F) <- c("R Squared", "Adj. R Squared",
             "F Statistic", "Durbin Watson")

BaimModel4Coef <- c(-22.368, 11.178, -1.8922, 6.332, -0.7743, -3.6111,
                    0.004666)

BaimModel4T <- c(-2.165, 2.816, -2.939, 1.082, -0.922, 2.24, -1.138)

BaimModel4F <- c(0.77, 0.62, 5.06, 2.54)
summary(lmOld4)

#########################################################

pdf("repDotplot4.pdf", height = 5, width = 12)

plot1 <- dotplot(reorder(names(Model4T), abs(Model4T)) ~ Model4Coef + BaimModel4Coef | "Coefficient",
                     col = brewer.pal(3, "Set1"), pch = 0:1, xlab = "",
                     scales = list(y = list(cex = 1.2), x = list(cex = 1.2)))

plot2 <- dotplot(reorder(names(Model4T), abs(Model4T)) ~ Model4T + BaimModel4T | "t-value",
                     col = brewer.pal(3, "Set1"), pch = 0:1, xlab = "",
                     panel = function(...) {
                        panel.abline(v = qt(0.025, nrow(model4)), lty = 2)
                        panel.abline(v = qt(0.975, nrow(model4)), lty = 2)
                        panel.dotplot(...)
                   },
                     scales = list(y = list(cex = 1.2), x = list(cex = 1.2)))

plot3 <- dotplot(reorder(names(Model3F), abs(Model3F)) ~ Model4F + BaimModel4F | "ANOVA",
                     col = brewer.pal(3, "Set1"), pch = 0:1, xlab = "",
                     scales = list(y = list(cex = 1.2), x = list(cex = 1.2)))

print(plot1, position = c(0, 0, 0.36, 1), more = TRUE)
print(plot2, position = c(0.32, 0, 0.68, 1), more = TRUE)
print(plot3, position = c(0.64, 0, 1, 1))
dev.off()

#########################################################
B.3 R Code for Chapter 4

```r
logBoxOffice <- log(CPIadj62 / 10 ^ 6)
lmOld1 <- lm(logBoxOffice ~ ., model1Old[, -c(3, 4, 7)])

xTest1 <- model1[17:22, -c(1, 5:7)]
yTest1 <- logBoxOffice[17:22]
yTrain1 <- logBoxOffice[1:16]
xTrain1 <- model1[1:16, -c(1, 5:7)]

xTrainLM1 <- model1[1:16, -c(1, 3:7, 10)]
xTestLM1 <- model1[17:22, -c(1, 3:7, 10)]

yhatTest1 <- as.matrix(cbind(1, xTestLM1)) %*% lmOld1$coefficients
yhatTrain1 <- as.matrix(cbind(1, xTrainLM1)) %*% lmOld1$coefficients
yhatTestBaim1 <- as.matrix(cbind(1, xTest1)) %*% BaimModel1Coef
yhatTrainBaim1 <- as.matrix(cbind(1, xTrain1)) %*% BaimModel1Coef

rmseTrain1 <- sqrt(mean(resid(lmOld1) ^ 2))
rmseTest1 <- sqrt(mean((yTest1 - yhatTest1) ^ 2))
rmseBaimTrain1 <- sqrt(mean((yTrain1 - yhatTrainBaim1) ^ 2))
rmseBaimTest1 <- sqrt(mean((yTest1 - yhatTestBaim1) ^ 2))

rmseBench1Test <- sqrt(mean((yTest1 - mean(logBoxOffice[1:16])) ^ 2))
rmseBench1Train <- sqrt(mean((yTrain1 - mean(logBoxOffice[1:16])) ^ 2))

# Lasso: Model 1
library(glmnet)
grid = 10 ^ seq(10, -2, length = 100)
set.seed(1)
lassoMod = glmnet(as.matrix(xTrain1[1:16, ]), logBoxOffice[1:16],
```
alpha = 1, lambda = grid, thresh = 1e-12)

set.seed(1)

cvOut = cv.glmnet(as.matrix(xTrain1[1:16, ]), logBoxOffice[1:16], 
                  alpha = 1)

bestlam = cvOut$lambda.min

lassoTest1 <- predict(lassoMod, s = bestlam, 
                       newx = as.matrix(xTest1))

lassoTrain1 <- predict(lassoMod, s = bestlam, 
                        newx = as.matrix(xTrain1))

rmseLassoTest1 <- sqrt(mean(((yTest1 - lassoTest1) ^ 2)))

rmseLassoTrain1 <- sqrt(mean(((yTrain1 - lassoTrain1) ^ 2)))

# Random Forest Model 1

library(randomForest)

set.seed(1)

rf1 <- randomForest(x = xTrain1[1:16, ], y = logBoxOffice[1:16], 
                     mtry = 2, ntree=5000)

predRf1Test = predict(rf1, newdata = xTest1, type = "class")

predRf1Train = predict(rf1, newdata = xTrain1, type = "class")

rmseTest1Rf <- sqrt(mean(((yTest1 - predRf1Test) ^ 2)))

rmseTrain1Rf <- sqrt(mean(((logBoxOffice[1:16] - predRf1Train) ^ 2)))

rmseAll <- c(rmseTrain1, rmseBaimTrain1, rmseBench1Train, 
             rmseLassoTrain1, rmseTrain1Rf, rmseBaimTest1, 
             rmseBench1Test, rmseLassoTest1, rmseTest1Rf)

rmseNames <- c("OLS", "Baimbridge", "Benchmark", "Lasso", "Forest", 
                "OLS", "Baimbridge", "Benchmark", "Lasso", "Forest")

rmseType <- c(rep("Train", 5), rep("Test", 5))

rmseDot <- as.data.frame(cbind(rmseNames, rmseAll, rmseType))
rmseDot$pch <- c(3, 0, 4, 4, 2, 2, 3, 1, 1, 0)
clr <- brewer.pal(6, "Paired")
rmseDot$color <- c(clr[3], clr[5], "black", clr[4], clr[6])
ordRmseDot <- rmseDot[order(rmseDot$rmseAll), ]
ordRmseDot$rmseType <- factor(ordRmseDot$rmseType)
dotVariable <- as.numeric(as.character(ordRmseDot$rmseAll))
ordRmseDot$rmseType <- factor(ordRmseDot$rmseType,
levels(ordRmseDot$rmseType)[2:1])
pdf("rmseModel1.pdf", height = 7, width = 9)
layout(matrix(c(1, 2, 3, 3), 2, 2, byrow = F),
widths = c(6, 4), heights = c(1, 1))
par(mar = c(4, 4.2, 1, 0))
clr <- brewer.pal(6, "Paired")
plot(movieYear[1:16], logBoxOffice[1:16], ylim = c(1.5, 5.3), pch = 19,
xlim = c(1960, 2010), col = clr[1], cex.axis = 1.3, cex.lab = 1.3,
xlab = "Release Date", ylab = "BOR (1962, CPI)"
lines(movieYear[1:16], yhatTrain1, type = "p", pch = 4, col = clr[3])
lines(movieYear[1:16], yhatTrainBaim1, type = "p",
pch = 3, col = clr[5])
lines(x = movieYear[17:22], y = yTest1, type = "p",
pch = 19, col = clr[2])
lines(movieYear[17:22], yhatTest1, type = "p", col = clr[4], pch = 4)
lines(x = movieYear[17:22], y = yhatTestBaim1, type = "p",
pch = 3, col = clr[6])
text(x = 1989, y = 3.3, labels = "Bench Mean")
legend("topright", inset = 0.03,
c("Train Observed", "Train OLS", "Train Baimbridge",
"Test Observed", "Test OLS", "Test Baimbridge"),
pch = c(19, 4, 3, 19, 4, 3),
col = c(clr[1], clr[3], clr[5], clr[2], clr[4], clr[6]))
abline(h = mean(logBoxOffice[1:16]), lty = 2)

plot(movieYear[1:16], logBoxOffice[1:16], ylim = c(1.5, 5.3), pch = 19,
xlim = c(1960, 2010), col = clr[1], cex.axis = 1.3, cex.lab = 1.3,
xlab = "Release Date", ylab = "BOR (1962, CPI)"
lines(movieYear[1:16], lassoTrain1, type = "p", pch = 0, col = clr[3])
lines(movieYear[1:16], predRf1Train, type = "p",
pch = 2, col = clr[5])
lines(x = movieYear[17:22], y = yTest1, type = "p",
pch = 19, col = clr[2])
lines(movieYear[17:22], lassoTest1, type = "p", col = clr[4], pch = 0)
lines(x = movieYear[17:22], y = predRf1Test, type = "p",
pch = 2, col = clr[6])
text(x = 1989, y = 3.3, labels = "Bench Mean")

legend("topright", inset = 0.03,
c("Train Observed", "Train Lasso", "Train Forest",
"Test Observed", "Test Lasso", "Test Forest"),
pch = c(19, 0, 2, 19, 0, 2),
col = c(clr[1], clr[3], clr[5], clr[2], clr[4], clr[6]))
abline(h = mean(logBoxOffice[1:16]), lty = 2)

par(mar = c(4, 4, 1, 1))
dotchart(dotVariable, labels = ordRmseDot$rmseNames, lty = 0,
cex = 1, groups = ordRmseDot$rmseType, xlim = c(0, 1.2),
xlab = "Root Mean Square Error", gcolor = "black",
lcolor = "black",
pch = ordRmseDot$pch)
logBoxOffice <- log(1 / 10 ^ 6 * read.csv("mojoAdj.csv")[, 3])

lmOld3 <- lm(logBoxOffice ~ ., model3Old[, -(3:5)])
xTest3 <- model3[17:22, -1]
yTest3 <- logBoxOffice[17:22]
yTrain3 <- logBoxOffice[1:16]
xTrain3 <- model3[1:16, -1]

xTrainLM3 <- model3[1:16, -c(1, 3:5)]
xTestLM3 <- model3[17:22, -c(1, 3:5)]

yhatTest3 <- as.matrix(cbind(1, xTestLM3)) %*% lmOld3$coefficients
yhatTrain3 <- as.matrix(cbind(1, xTrainLM3)) %*% lmOld3$coefficients
yhatTestBaim3 <- as.matrix(cbind(1, xTest3)) %*% BaimModel3Coef
yhatTrainBaim3 <- as.matrix(cbind(1, xTrain3)) %*% BaimModel3Coef

rmseTrain3 <- sqrt(mean(resid(lmOld3) ^ 2))
rmseTest3 <- sqrt(mean((yTest3 - yhatTest3) ^ 2))
rmseBaimTrain3 <- sqrt(mean((yTrain3 - yhatTrainBaim3) ^ 2))
rmseBaimTest3 <- sqrt(mean((yTest3 - yhatTestBaim3) ^ 2))

rmseBench3Test <- sqrt(mean((yTest3 - mean(logBoxOffice[1:16])) ^ 2))
rmseBench3Train <- sqrt(mean((yTrain3 - mean(logBoxOffice[1:16])) ^ 2))

# Lasso: Model 3

library(glmnet)

grid = 10 ^ seq(10, -2, length = 100)
set.seed(1)
lassoMod = glmnet(as.matrix(xTrain3[1:16, ]), logBoxOffice[1:16],
                 alpha = 1, lambda = grid, thresh = 1e-12)
set.seed(1)
cvOut = cv.glmnet(as.matrix(xTrain3[1:16, ]), logBoxOffice[1:16],
                  alpha = 1)
bestlam = cvOut$lambda.min
lassoTest3 <- predict(lassoMod, s = bestlam,
                      newx = as.matrix(xTest3))
lassoTrain3 <- predict(lassoMod, s = bestlam,
                       newx = as.matrix(xTrain3))
rmseLassoTest3 <- sqrt(mean((yTest3 - lassoTest3) ^ 2))
rmseLassoTrain3 <- sqrt(mean((yTrain3 - lassoTrain3) ^ 2))

# Random Forest Model 3
library(randomForest)
set.seed(1)
rf3 <- randomForest(x = xTrain3[1:16, ], y = logBoxOffice[1:16],
                    mtry = 2, ntree=5000)
predRf3Test = predict(rf3, newdata = xTest3, type = "class" )
predRf3Train = predict(rf3, newdata = xTrain3, type = "class" )
rmseTest3Rf <- sqrt(mean((yTest3 - predRf3Test) ^ 2))
rmseTrain3Rf <- sqrt(mean((logBoxOffice[1:16] - predRf3Train) ^ 2))

rmseAll <- c(rmseTrain3, rmseBaimTrain3, rmseBench3Train, rmseLassoTrain3, rmseTrain3Rf,
             rmseTest3, rmseBaimTest3, rmseBench3Test, rmseLassoTest3, rmseTest3Rf)
rmseNames <- c("OLS", "Baimbridge", "Bench Mean", "Lasso", "Forest",
               "OLS", "Baimbridge", "Bench Mean", "Lasso", "Forest")
rmseType <- c(rep("Train", 5), rep("Test", 5))
rmseDot <- as.data.frame(cbind(rmseNames, rmseAll, rmseType))
rmseDot$pch <- c(0, 2, 4, 3, 2, 3, 1, 1, 0, 4)
clr <- brewer.pal(6, "Paired")
rmseDot$color <- c(clr[3], clr[5], "black", clr[4], clr[6])
ordRmseDot <- rmseDot[order(rmseDot$rmseAll), ]
ordRmseDot$rmseType <- factor(ordRmseDot$rmseType)
dotVariable <- as.numeric(as.character(ordRmseDot$rmseAll))
ordRmseDot$rmseType <- factor(ordRmseDot$rmseType,
    levels(ordRmseDot$rmseType)[2:1])

pdf("rmseModel3.pdf", height = 7, width = 9)
layout(matrix(c(1, 2, 3, 3), 2, 2, byrow = F),
    widths = c(6, 4), heights = c(1, 1))
par(mar = c(4, 4.2, 1, 0))
clr <- brewer.pal(6, "Paired")
plot(movieYear[1:16], logBoxOffice[1:16], ylim = c(1.5, 5.3), pch = 19,
    xlim = c(1960, 2010), col = clr[1], cex.axis = 1.3, cex.lab = 1.3,
    xlab = "Release Date", ylab = "BOR (1963, Ticket)"
lines(movieYear[1:16], yhatTrain3, type = "p", pch = 4, col = clr[3])
lines(movieYear[1:16], yhatTrainBaim3, type = "p",
    pch = 3, col = clr[5])
lines(x = movieYear[17:22], y = yTest3, type = "p",
    pch = 19, col = clr[2])
lines(movieYear[17:22], yhatTest3, type = "p", col = clr[4], pch = 4)
lines(x = movieYear[17:22], y = yhatTestBaim3, type = "p",
    pch = 3, col = clr[6])
# lines(x = movieYear[2:22], y = maModel, type = "o", lty = 2)
text(x = 1989, y = 2.95, labels = "Bench Mean")

legend(x = 1975, y = 5.2,
c("Train Observed", "Train OLS", "Train Baimbridge",
"Test Observed", "Test OLS", "Test Baimbridge"),
pch = c(19, 4, 3, 19, 4, 3),
col = c(clr[1], clr[3], clr[5], clr[2], clr[4], clr[6]))
abline(h = mean(logBoxOffice[1:16]), lty = 2)

plot(movieYear[1:16], logBoxOffice[1:16], ylim = c(1.5, 5.3), pch = 19,
xlim = c(1960, 2010), col = clr[1], cex.axis = 1.3, cex.lab = 1.3,
xlab = "Release Date", ylab = "BOR (1963, Ticket)")
lines(movieYear[1:16], lassoTrain3, type = "p", pch = 0, col = clr[3])
lines(movieYear[1:16], predRf3Train, type = "p",
pch = 2, col = clr[5])

lines(x = movieYear[17:22], y = yTest3, type = "p",
pch = 19, col = clr[2])
lines(movieYear[17:22], lassoTest3, type = "p", col = clr[4], pch = 0)
lines(x = movieYear[17:22], y = predRf3Test, type = "p",
pch = 2, col = clr[6])
# lines(x = movieYear[2:22], y = maModel, type = "o", lty = 2)
text(x = 1989, y = 2.95, labels = "Bench Mean")

legend("topright", inset = 0.03,
c("Train Observed", "Train Lasso", "Train Forest",
"Test Observed", "Test Lasso", "Test Forest"),
pch = c(19, 0, 2, 19, 0, 2),
col = c(clr[1], clr[3], clr[5], clr[2], clr[4], clr[6]))
abline(h = mean(logBoxOffice[1:16]), lty = 2)

par(mar = c(4, 4, 1, 1))
dotchart(dotVariable, labels = ordRmseDot$rmseNames, lty = 0,
cex = 1, groups = ordRmseDot$rmseType, xlim = c(0, 1.2),
xlab = "Root Mean Square Error", gcolor = "black",
lcolor = "black",
pch = ordRmseDot$pch)

dev.off()

########################################################################

Bond.data <- read.csv("MS_project_Data2.csv")
movie.Date <- as.Date(Bond.data[,13])

theEconDataX <- Bond.data[, 9:12]
logBoxOffice <- log(CPIadj62 / 10 ^ 6)

theEconData <- cbind(logBoxOffice, theEconDataX)

lmTheEcon <- lm(logBoxOffice[1:16] ~ ., theEconData[1:16,])
yhatTestE <- as.matrix(cbind(1, theEconDataX[17:22,])) %*% lmTheEcon$coefficients
yhatTrainE <- as.matrix(cbind(1, theEconDataX[1:16,])) %*% lmTheEcon$coefficients

rmseTrainE <- sqrt(mean(resid(lmTheEcon) ^ 2))
rmseTestE <- sqrt(mean((yhatTestE - logBoxOffice[17:22]) ^ 2))
rmseTrainE <- sqrt(mean((yhatTrainE - logBoxOffice[1:16]) ^ 2))

# Lasso The Economist
set.seed(1)
lassoMod = glmnet(as.matrix(theEconDataX[1:16,]), logBoxOffice[1:16],
alpha = 1, lambda = grid, thresh = 1e-12)

set.seed(1)
cvOut = cv.glmnet(as.matrix(theEconDataX[1:16, ]), logBoxOffice[1:16],
                   alpha = 1)
bestlam = cvOut$lambda.min
lassoTestE <- predict(lassoMod, s = bestlam,
                       newx = as.matrix(theEconDataX[17:22, ]))
lassoTrainE <- predict(lassoMod, s = bestlam,
                       newx = as.matrix(theEconDataX[1:16, ]))

rmseLassoTestE <- sqrt(mean((yTest1 - lassoTestE) ^ 2))
rmseLassoTrainE <- sqrt(mean((yTrain1 - lassoTrainE) ^ 2))

# Random Forest The Economist
set.seed(1)
rfE <- randomForest(x = theEconDataX[1:16, ], y = logBoxOffice[1:16],
                     mtry = 2, ntree=5000)
predRfE = predict(rfE, newdata = theEconDataX[17:22, ], type = "class")
predRfETrain = predict(rfE, newdata = theEconDataX[1:16, ], type = "class")

rmseTestRfE <- sqrt(mean((logBoxOffice[17:22] - predRfE) ^ 2))
rmseTrainRfE <- sqrt(mean((logBoxOffice[1:16] - predRfETrain) ^ 2))

# Benchmark Moving Averages
maModel <- sapply(1:21, function(x) mean(logBoxOffice[1:x]))
rmseMaTrain <- sqrt(mean((model1[2:16, 1] - maModel[1:15]) ^ 2))
rmseMaTest <- sqrt(mean((LogBoxOffice[17:22] - maModel[16:21]) ^ 2))
pdf("rmseModelE.pdf", height = 7, width = 9)
layout(matrix(c(1, 2, 3, 3), 2, 2, byrow = F),

alpha = 1, lambda = grid, thresh = 1e-12)

set.seed(1)
cvOut = cv.glmnet(as.matrix(theEconDataX[1:16, ]), logBoxOffice[1:16],
                   alpha = 1)
bestlam = cvOut$lambda.min
lassoTestE <- predict(lassoMod, s = bestlam,
                       newx = as.matrix(theEconDataX[17:22, ]))
lassoTrainE <- predict(lassoMod, s = bestlam,
                       newx = as.matrix(theEconDataX[1:16, ]))

rmseLassoTestE <- sqrt(mean((yTest1 - lassoTestE) ^ 2))
rmseLassoTrainE <- sqrt(mean((yTrain1 - lassoTrainE) ^ 2))

# Random Forest The Economist
set.seed(1)
rfE <- randomForest(x = theEconDataX[1:16, ], y = logBoxOffice[1:16],
                     mtry = 2, ntree=5000)
predRfE = predict(rfE, newdata = theEconDataX[17:22, ], type = "class")
predRfETrain = predict(rfE, newdata = theEconDataX[1:16, ], type = "class")

rmseTestRfE <- sqrt(mean((logBoxOffice[17:22] - predRfE) ^ 2))
rmseTrainRfE <- sqrt(mean((logBoxOffice[1:16] - predRfETrain) ^ 2))

# Benchmark Moving Averages
maModel <- sapply(1:21, function(x) mean(logBoxOffice[1:x]))
rmseMaTrain <- sqrt(mean((model1[2:16, 1] - maModel[1:15]) ^ 2))
rmseMaTest <- sqrt(mean((LogBoxOffice[17:22] - maModel[16:21]) ^ 2))
pdf("rmseModelE.pdf", height = 7, width = 9)
layout(matrix(c(1, 2, 3, 3), 2, 2, byrow = F),
```r
widths = c(6, 4), heights = c(1, 1))
par(mar = c(4, 4.2, 1, 0))

clr <- brewer.pal(6, "Paired")
plot(movieYear[1:16], logBoxOffice[1:16], ylim = c(1.5, 5.3), pch = 19,
     xlim = c(1960, 2010), col = clr[1], cex.axis = 1.3, cex.lab = 1.3,
     xlab = "Release Date", ylab = "BOR (1962, CPI)"
lines(movieYear[1:16], yhatTrainE, type = "p", pch = 4, col = clr[3])
lines(movieYear[2:16], maModel[1:15], type = "p",
      pch = 8, col = clr[5])
lines(x = movieYear[17:22], y = logBoxOffice[17:22], type = "p",
      pch = 19, col = clr[2])
lines(movieYear[17:22], yhatTestE, type = "p", col = clr[4], pch = 4)
lines(x = movieYear[17:22], y = maModel[16:21], type = "p",
      pch = 8, col = clr[6])
# lines(x = movieYear[2:22], y = maModel, type = "o", lty = 2)
text(x = 1989, y = 3.35, labels = "Bench Mean")

legend("topright", inset = 0.03,
      c("Train Observed", "Train OLS", "Train Bench MA",
         "Test Observed", "Test OLS", "Test Bench MA"),
      pch = c(19, 4, 8, 19, 4, 8),
      col = c(clr[1], clr[3], clr[5], clr[2], clr[4], clr[6]))
abline(h = mean(logBoxOffice[1:16]), lty = 2)

plot(movieYear[1:16], logBoxOffice[1:16], ylim = c(1.5, 5.3), pch = 19,
     xlim = c(1960, 2010), col = clr[1], cex.axis = 1.3, cex.lab = 1.3,
     xlab = "Release Date", ylab = "BOR (1962, CPI)"
lines(movieYear[1:16], lassoTrainE, type = "p", pch = 0, col = clr[3])
```

lines(movieYear[1:16], predRfETrain, type = "p",
    pch = 2, col = clr[5])

lines(x = movieYear[17:22], y = logBoxOffice[17:22], type = "p",
    pch = 19, col = clr[2])
lines(movieYear[17:22], lassoTestE, type = "p", col = clr[4], pch = 0)
lines(x = movieYear[17:22], y = predRfE, type = "p",
    pch = 2, col = clr[6])
# lines(x = movieYear[2:22], y = maModel, type = "o", lty = 2)
text(x = 1989, y = 3.35, labels = "Bench Mean")

legend("topright", inset = 0.03,
    c("Train Observed", "Train Lasso", "Train Forest",
        "Test Observed", "Test Lasso", "Test Forest"),
    pch = c(19, 0, 2, 19, 0, 2),
    col = c(clr[1], clr[3], clr[5], clr[2], clr[4], clr[6]))
abline(h = mean(logBoxOffice[1:16]), lty = 2)

par(mar = c(4, 4, 1, 1))

rmseAll <- c(rmseTrainE, rmseMaTrain, rmseBench3Train, rmseLassoTrainE, rmseTrainRfE,
    rmseTestE, rmseMaTest, rmseBench3Test, rmseLassoTestE, rmseTestRfE)
rmseNames <- c("OLS", "Bench MA", "Bench Mean", "Lasso", "Forest",
    "OLS", "Bench MA", "Bench Mean", "Lasso", "Forest")
rmseType <- c(rep("Train", 5), rep("Test", 5))
rmseDot <- as.data.frame(cbind(rmseNames, rmseAll, rmseType))
rmseDot$pch <- c(2, 8, 1, 4, 0, 0, 8, 1, 4, 2)
clr <- brewer.pal(6, "Paired")
rmseDot$color <- c(clr[3], clr[5], "black", clr[4], clr[6])
ordRmseDot <- rmseDot[order(rmseDot$rmseAll), ]
ordRmseDot$rmseType <- factor(ordRmseDot$rmseType)
dotVariable <- as.numeric(as.character(ordRmseDot$rmseAll))
ordRmseDot$rmseType <- factor(ordRmseDot$rmseType,
    levels(ordRmseDot$rmseType)[2:1])

dotchart(dotVariable, labels = ordRmseDot$rmseNames, lty = 0,
    cex = 1, groups = ordRmseDot$rmseType, xlim = c(0, 1.2),
    xlab = "Root Mean Square Error", gcolor = "black",
    lcolor = "black",
    pch = ordRmseDot$pch)
dev.off()

*****************************************************************************