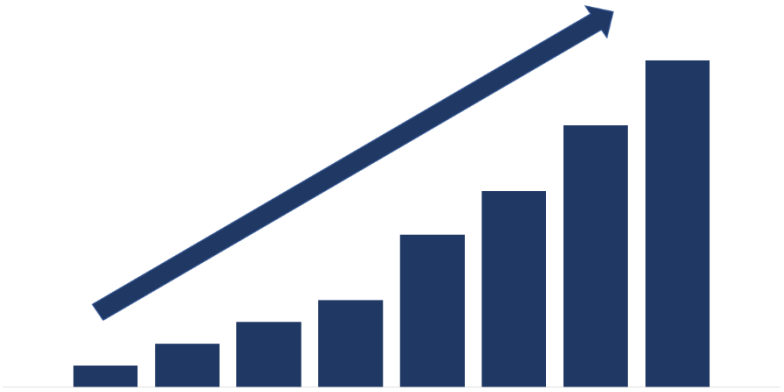


The Data Driven Model for

Earnings Forecasting

Master Thesis

MSc Finance and Investments

Copenhagen Business School

15-5-2017

Aske Buemann

Characters: 160.961

Pages: 71

Supervisor

Jeppe Christoffersen

Executive Summary

Earnings forecasting is a central topic for many financial statement users in order to predict the future performance of a company. These users can generally be grouped in 4 categories. Firstly, investors are dependent on the information to identify the correct price of a stock to implement in their trading strategies. Secondly, analysts generate and use the forecasts for providing recommendations. Thirdly, venture capitalists use future earnings to value non-listed companies. Finally, forecasting earnings is imperative for any managerial budgeting and resource allocation process.

Despite its unneglectable importance, the area of research has been dominated by the same two types of models for the past decades. Only recently, the time series and analyst forecasts have been challenged, but no accurate contender has been identified yet. One of the main challenges in this regard is the requirement for a generally applicable model in order to meet the variety of needs within the earnings forecasting community. Therefore, the ideal model is able to apply to listed and non-listed companies across a range of sizes and industries.

The thesis proposes an alternative method for earnings forecasting that it designates the *data driven model*. The model represents a way of combining the most essential advantages from the time series and analyst methodologies, whereby it integrates more information than the time series models and is less biased than the analyst forecasts. Using a larger and broader dataset than seen in previous literature, the thesis proves the superiority of the data driven model over both the time series models and analyst forecasts. In this way, the thesis contributes to the earnings forecasting literature with a new model and the underlying reasoning that utilizing financial information from comparable companies produces better forecasts than solely relying on past earnings of the same company.

The superiority of the alternative forecasting methodology has several implications for the financial statement users. Using the generally applicable data driven approach, the venture capital community is now able to generate reliable forecasts for non-listed firms that are superior to the previously used time series forecasts. At the same time, investors are able to implement better and almost costless forecasts of listed companies' earnings in their trading strategies, while managers can ease the earnings estimation in their budgeting and resource allocation processes.

Table of Contents

Table of Illustrations

1. Introduction

In a financial context, being able to correctly predict the earnings of a company has significant implications for several stakeholders. The better a forecast an investor can generate, the greater a chance to trade at a more accurate earnings' estimate and hence stock price. In this way, the investor will be able to gain an advantage over competitors and be able to implement more profitable trading strategies. Further, one of the primary tasks of an analyst is to produce forecasts of the earnings and cash flows for a range of companies, which investors can then acquire at a relatively high price. However, if the investors have a methodology at their own disposal that is as good or even better than the analyst forecasts, they could save the entire cost. In addition, there exists a significant challenge in predicting earnings for non-listed companies, where analysts are not producing forecasts. In these cases, the forecasts made are either derived from subjective discounted cash flow models or time series models. Therefore, there is a gab which concerns the vast majority of companies as only a small fraction of firms is listed on an exchange and followed by analysts. Hence, there is a need for a model to forecast earnings in the venture capital market as well. Further, earnings forecasting is a vital part of the budgeting and resource allocation process so improvements in this area can prove very useful for managers within a company as well.

The earnings forecasting literature dates back to the 1960s, where there was an observed shift in the focus of the accounting research (Peek, 1997). It moved from being very descriptive to focusing on analyzing and predicting the accounting information of corporations. This change led to an extensive search for theories that could support various techniques of predicting earnings, whereby the field of earnings forecasting was initiated. Initially, the literature was very focused on how the earnings predictions related to the market expectations (Brown, 1993). However, the literature has later directed its attention to the process itself of forecasting earnings due to the importance of the earnings forecasts in the capital markets (Peek, 1997). Since then, several models and methods for predicting earnings have been suggested, though the two methods of time series models and analyst forecasts remain the most superior. Recently, some models have tried to enter the discussion but without great success. Especially, the general applicability is a main disadvantage of the recent models and the analyst forecasts, since they mainly operate with listed firms. This paves the way for new alternative models that can improve the forecasting accuracy and applicability.

In order for one model to cover both the listed firms and the non-listed firms, it has to be created from and tested on a very broadly selected dataset. This thesis will utilize a large dataset comprised of more

than 350,000 companies including both listed and non-listed firms to ensure a wide applicability. A dataset of this magnitude has not been observed in the literature due to the literature's focus on listed firms and neglection of non-listed firms. The thesis will use its dataset to generate an alternative method for earnings forecasting, the data driven model. This alternative methodology constitutes a way of grouping similar companies in portfolios based on a broad range of metrics such as earnings, industry and total assets. For each of the portfolios of comparable companies, the average earnings development is tracked and can be applied to out-of-sample companies in order to produce forecasts of their earnings. The underlying notion of the approach is that earnings development in similar companies is a better indicator of how the company to be forecasted will develop compared to solely using the past earnings of the company itself.
The data driven approach will be compared with the currently dominant forecasting models, which is mainly comprised of time series models and analyst forecasts. The superiority of the models will be determined based on an array of analyses spanning across several horizons for both the data used and the forecast produced. At the same time, the robustness of the forecasts will be ensured by using several methods for determining the accuracy as found in the existing literature.

Thus, the structure of the thesis will be the following. First, the research questions will be outlined and elaborated. Then, the thesis will turn towards the theoretical background generated by the literature on earnings forecasting. Here, the two primary methodologies and some alternative models will be discussed along with the way of measuring accuracy of earnings forecasts. Next, the data driven forecasting method is introduced along with the data utilized in the thesis and how that differs from the data in previous literature. Thereafter, the variables used in the analyses and the findings of the model comparisons will be presented. Finally, the thesis will discuss the results and outline their contributions to the existing literature, their implications for the stakeholders of the earnings forecasting field and their limitations that further research should seek to improve.

2. Research Questions

As introduced above, the most pressing topic under earnings forecasting is to identify an accurate model that can be applicable to a large amount of companies. Throughout the thesis, the data driven model will be investigated and tested for its applicability and accuracy. The utilized dataset stems from a broad range of firms representing both listed and non-listed segments and is thus considered to reflect a general illustration of how earnings develop. Further, the data driven model is as applicable as the data used to generate it, since it primarily works as a way of structuring the data. Hence, it will be considered

as generally applicable when it is generated by this type of dataset. Thereby, the most important topic to investigate is whether the model is accurate, which leads to the first research question being formulated as:

Analyze the forecasting accuracy of the data driven model

This is an important starting point, since it would not add value to the literature to introduce an inaccurate model. However, this finding in itself is not sufficient to conclude that the data driven method is usable. In order to achieve that result it is compulsory to compare the accuracy of the model to the currently dominant models of the time series and analyst forecasts. The two sets of models are generally viewed as the most precise methodologies, whereby any model being able to surpass their accuracy can contribute significantly to the literature body. Hence, the second research question will focus on this comparison and is formulated as:

Discuss the superiority of the data driven model compared to the time series and analyst forecasts

In order to answer these two research questions, it is first necessary to identify how accuracy is measured and how the time series and analyst methodologies are defined. Thus, the next section will be concerned with the theoretical background of the earnings forecasting topic.

3. Theory

The theory section will outline the most influential theories within earnings forecasting to provide frameworks that the data driven approach can be compared to. More specifically, two time series models are chosen for further investigation, the random walk model and the AutoRegressive Integrated Moving Average (ARIMA) model, along with the methodology of analyst forecasting. The time series models have been chosen both for the ease of testing and for the relevance in the literature body on earnings forecasting. In the following, the two models' forecasting methodology will be elaborated due to their complexity. The analyst method is also chosen for its relevance in the literature, though it is more challenging to perform large scale tests due to limited data availability, whereby the thesis is only able to attain a smaller sample. A more elaborate section on how the analysts produce their forecasts is not included as a detailed account of this method is for good reasons not accessible.

Therefore, the theory section will first introduce and explain the time series models and investigate how the models produce their forecasts. Then a determination of accuracy and the measurements hereof is necessary in order to enable a discussion of which models are most accurate. Once accuracy is defined

the time series models will be discussed in comparison with analyst forecasts to determine the model superiority and conditions hereof. In addition, a section on the already proposed alternative models will be included to provide a comprehensive account of the existing forecasting methodologies.

3.1 Time Series Models

This section will introduce the time series models in questions and elaborate on how their forecasting process works. The first model to be introduced in the thesis is that of a random walk. It is one of the simpler time series models and assumes that for each time step, the model deviates from its previous value with a random increment and potentially a drift (Nau 2014). It is generally expressed as (Skovmand 2016):

$$\hat{x}_t = \mu + x_{t-1} + \epsilon_t$$

Where $\hat{x}_t$ indicates the value to be forecasted, x_{t-1} is the latest value, μ is the potential drift which is zero if there is no drift and ϵ_t is the error term at time t. In a model with no drift, the error term will be the only element that can affect the value at time t other than the previous value at time t-1. The error term is a random variable and has the property $E[\epsilon_t] = 0$ meaning that the best prediction of the term is zero due to its normal distribution with a mean of zero. Hence, the best prediction of the whole process is its previous value and the drift: $E[x_t] = \mu + x_{t-1}$.

Thus, when the random walk produces a forecast it takes the point of departure in the latest value of earnings, x_{t-1}, and adds the potential drift along with the random error term to get the first forecasted value, x_t. In order to get the next forecasted value, it takes x_t as point of departure and adds the drift and a new random error term, ϵ_{t+1}, to get x_{t+1}. It continues in this way until the desired forecasting horizon is reached.

The second model to be investigated is the ARIMA model. It has been introduced in the literature at a later stage than the random walk model but has gained popularity more recently (Brown 1993). The ARIMA model is a combination of the Autoregressive (AR) and Moving Average (MA) models, which are then "Integrated" or differentiated to reach a stationary model (Smith 2015). Stationarity is an essential aspect of statistical models and in its essence, it indicates whether the process "wanders off" into a direction. It is a very common assumption underlying many models and is necessary for most statistical analyses (Skovmand 2016). If the process to be predicted is not stationary then the variables considered, here previous earnings and future earnings, might seem to explain each other, but in reality they can be independent. Therefore, the model is differentiated until it reaches stationarity.

In its general state, the model is expressed as ARIMA(p,d,q) where *p* indicates the lags (or correlations) on earlier data points, *d* indicates the order of differencing necessary to provide a stationary process and *q* indicates the lags on earlier error terms. The ARIMA model is a more comprehensive model than the random walk, in the sense that a random walk can be expressed as a specific case of the ARIMA model (Duke 2017b): ARIMA(0,1,0). This case refers to a setting where both the AR and MA process have 0 lags and the process is differentiated once. Therefore, the ARIMA model serves as a more general framework to also include additional aspects such as correlated earnings, mean reversion, time-varying means and seasonality (Duke 2017a).

In terms of forecasting, the ARIMA model requires more data than the random walk in order to make an accurate forecast (Bradshaw et al. 2009). The general forecasting equation is written as (Duke 2017a):

$$\hat{x}_t = \mu + \phi_1 x_{t-1} + \cdots + \phi_p x_{t-p} - \theta_1 \epsilon_{t-1} - \cdots - \theta_q \epsilon_{t-q}$$

Here, $\hat{x}_t$ indicates the value to be forecasted, μ is the drift from the historical data, x_{t-1} is the previous value, ϕ_1 is the correlation coefficient with the previous value, p is the lags on the previous value, ϵ_{t-1} is the previous value of the error term, θ_1 is the correlation coefficient with the previous error term, and q is the lags on the previous error term. For the process to be stationary, the ϕ_1 must be below 1, which in turn means that the process is mean reverting. At the same time, the sign of the coefficient will indicate whether the next value will have the same sign as the previous value. To exemplify, the ARIMA(1,0,0), also referred to as the Brown-Rozeff model, can be written as:

$$\hat{x}_t = \mu + \phi_1 x_{t-1}$$

While, ARIMA(0,1,1) referred to as the Griffin-Watts model can be written as:

$$\hat{x}_t - x_{t-1} = \mu - \theta_1 \epsilon_{t-1} \leftrightarrow$$

$$\hat{x}_t = \mu + x_{t-1} - \theta_1 \epsilon_{t-1}$$

Both the ARIMA(1,0,0) and ARIMA(0,1,1) will be included in the analyses of the thesis due to their popularity in the literature (Brown 1993).

3.2 Forecasting Accuracy

The theory section will cover the determination of which models are most accurate in their earnings forecasts. Therefore, it is critical to discuss the underlying meaning and measurements of accuracy, which will be the topic of this section.

In general, accuracy in earnings forecasts is determined based on the difference between the forecasted values and the actual values. However, there are multiple ways of making that calculation. As seen in Table 1 from the appendix, the literature body uses a variety of methods, which are summarized below in Table 2.

Table 2: Accuracy Measures

Measure	Abbreviation	Calculation	Definition	Frequency
Mean Error	ME	$\frac{1}{n}\sum_{i=1}^{n} a_i - f_i$	The mean of the difference between the actual and forecasted values of the firm	24%
Mean Absolute Error	MAE	$\frac{1}{n}\sum_{i=1}^{n} \lvert a_i - f_i \rvert$	The mean of the absolute difference between the actual and forecasted values of the firm	19%
Mean Relative Error	MRE	$\frac{1}{n}\sum_{i=1}^{n} \frac{a_i - f_i}{a_i}$	The mean of the percentage difference between the actual and forecasted values of the firm	10%
Mean Absolute Percentage Error	MAPE	$\frac{1}{n}\sum_{i=1}^{n} \left\lvert \frac{a_i - f_i}{a_i} \right\rvert$	The mean of the absolute percentage difference between the actual and forecasted values of the firm	33%
Mean Squared Error	MSE	$\frac{1}{n}\sum_{i=1}^{n} (a_i - f_i)^2$	The mean of the squared difference between the actual and forecasted values of the firm	29%
Mean Squared Relative Error	MSRE	$\frac{1}{n}\sum_{i=1}^{n} \left(\frac{a_i - f_i}{a_i}\right)^2$	The mean of the squared percentage difference between the actual and forecasted values of the firm	10%

The accuracy measures can generally be divided into three groups. The first is the simple Mean Error (ME), which takes the average of differences between the forecasted and actual value. The measure also appears in a variation where the absolute differences are used instead, which is abbreviated MAE. The key benefit of this approach is its simplicity and understandable intuition given that the resulting value is quite easy to interpret, especially for the absolute version. For example, in the absolute case a resulting value of 3 means that the forecast on average deviates from the actual values with 3 units. However, it does not make any statement about how much 3 units are compared to the actual value, hence the second group of Mean Relative Errors is required. They compare the mean error or the absolute error with the actual value to identify the percentage deviation. This provides a lot more information as 3 is a relatively small deviation from an actual value of 10.000 but a rather large deviation from an actual value of 1. The relative measure also comes in an absolute version called Mean Absolute Percentage

Error (MAPE) that is quite often used as evident from the 33% of the literature utilizing this measure. Thus, it is the most utilized accuracy measure for the included literature. The final group of accuracy measures is the Mean Squared Error (MSE), which squares the difference between the actual and forecasted value. The main difference to the MAPE is that the outliers are weighted more significantly as the difference is squared, which allows for easier identification of methodologies exhibiting larger outliers. However, the desirability of promoting the outliers depends heavily on the research design. From Table 2, it is evident that a significant part of the literature has used one of the two squared accuracy measures, whereby it is determined relevant to include in the thesis as well.

The above accuracy measures will be the foundation for determining the performance of the proposed model in the thesis. Thus, when the following sections refer to the superiority or accuracy of a model, it is with reference to minimizing the absolute value of the above error measures. However, there are some general disadvantages of the measures that need to be considered when evaluating the models. First, it is very difficult to track the sign of the deviation in order to determine if a model is consistently biased downwards or upwards. It is possible to get an indication through the ME measure as it will be negative if there are most average errors below the actual values, though it is still not able to provide any meaningful picture of whether there is a consistent trend of a downwards bias. This can be partly solved by making some simple counts of the number of negative variations compared to the number of positives ones, though these would not take the magnitude of the variations into account. Secondly, accuracy can be defined in other ways than just a measure of mean variation, it could for instance be interesting to know the number of times the forecast is within an accepted range of the actual value. This could provide analyses of whether the forecasts are most often “close enough” to be valuable to investors. However, this remains a topic for further research.

3.3 Model Superiority

On an overall level, the literature body on earnings forecasting can be divided in three groups in terms of the proposed model superiority. The resulting division can be seen in Table 1 of the appendix, which outlines to which group each research paper belongs. The first group emphasizes the superiority of time series models including the ARIMA and random walk model. The second group directly opposes that view and finds that analysts are superior in their earnings forecasts. Finally, a third group attempts to contribute with alternative forecasting models as the superior methodology.

In the following sections, each of the three groups will be reviewed and discussed. For the former two, the discussion will be divided in the general advantages of the model and the resulting conditions that

the model is superior within. The latter type of model will to a larger extend focus on how the alternatives can be utilized to generate an improved and more generally applicable model.

To assist in the discussion, Table 3 has been generated to outline the key characteristics of the 3 branches in the literature. In the table, *sample size* is the number of firms considered in the models, *latest year* refers to the most recent year in the data period of the paper, *data horizon* refers to the years of data used to generate the forecasts and *forecast horizon* refers to the length of the forecast period.

Table 3: Literature Analysis

	Sample Size (Firms)	Latest Year (Average)	Latest Year (Maximum)	Data Horizon (Years)	Forecast Horizon (Years)	Articles Included
Time Series	2743	1984	2010	23	2.5	9
Analysts	288	1981	1984	10	1.0	7
Alternatives	373	1996	2009	17	2.2	7
Full Literature	1369	1986	2010	18	2.0	25

3.3.1 Time Series Models

In general, there are several arguments why time series models are more accurate than analyst forecasts. The initial set of arguments are related to the biases the analysts face when producing their forecasts of earnings.

Firstly, analysts are less accurate in their predictions of larger earnings changes, since their models tend to emphasize smooth earnings. This is especially apparent for smaller firms with a less documented earnings history and a less diversified business portfolio (Bradshaw et al. 2009). Hence, they are more prone to shocks from either economical or market factors. The random walk does not face the same challenge as it has shocks integrated in its ϵ-parameter. Due to the assumed normal distribution of the parameter with a mean of zero, major shocks are unlikely, though they can appear.

Secondly, analysts tend to be overly optimistic in their predictions of earnings (O'Brien 1988). One part of the optimism is driven by the incentive structure in place at their own institutions, where managers do not reward a "buy" and "hold" recommendation equally (Bradshaw et al. 2009). For a brokerage making a recommendation for one of their clients, a clear conflict of interest arises if the brokerage provides a "sell" recommendation, as the client might find another brokerage that can provide a more optimistic recommendation. Thus, the brokerage is in the end left with the choice of making a "buy" recommendation or losing the client, where the former is too often chosen (Brown 1993). The second part of the optimism arises from the personal relationship that analysts develop with their clients. To

attain as much information as possible, the analyst holds meetings with the client and might be invited to other excursions or events to get to know the company and its people (Bradshaw et al. 2009). Therefore, the analysts risk being too emotionally connected to the client and its business, which causes an upwards bias.

Besides the above biases, there are multiple reasons why the literature advocating analyst accuracy does not reflect the full picture (Bradshaw et al. 2009). Firstly, the data utilized is often outdated as seen in Table 3, where the latest data is from 1984 since the literature is from the 1980s and 1990s. This is naturally suboptimal to use when making statements about forecasting superiority 30 years later. In the literature arguing for time series models, Table 3 shows that the articles have data up to 2010, which provides more current findings.

Secondly, the analyst literature has embodied very small sample sizes compared to the current market listings with samples of as few as 50 firms. From Table 3, it is evident that the literature advocating time series models has much more comprehensive data samples with an average size of more than 9 times as many firms. Thus, the analyst literature risks misrepresenting the data population, which can cause severe biases in their results. The larger samples allow for a much more representative selection that even though it is not free of biases is at least reasonably expected to contain less of a bias.

Thirdly, the selection of the firms in the literature needs to be considered as mainly large firms with many analysts following are chosen (Bradshaw et al. 2009). This selection causes a bias as the size of the firms has a negative correlation with the relative accuracy of a random walk model. Hence, the literature attempts to make a general model based on data already biased against the random walk model.

Fourthly, the forecast horizon of the literature is too limited and too few studies have looked at horizons over 2 year. As seen in Table 3, the average horizon for the analyst literature is around 1 year, while the time series literature has an average of 2.5 years. This is favoring the analyst models since random walk models perform relatively better over longer forecast horizons than 1 year (Bradshaw et al. 2009). Thus, when only investigating the shorter horizons the research will see a biased picture.

Fifthly, the data horizon also differs significantly between the two branches of literature. As seen in Table 3, the analyst literature has an average of 10 years of data, while the time series literature has an average data horizon of almost 2.4 times as much, which increases the usability of the ARIMA model (Watts and Leftwich 1977 ; Bradshaw et al. 2009).

Finally, Bradshaw et al. (2009) outlines that even when the literature find statistically significant results for analyst forecasting superiority, it is often not economically significant. Hence, the actual interpretation of the results will not display an extensive superiority of the analysts. Even for the

literature that finds the analysts and time series model equally precise (Albrecht, Lookabill, and McKeown 1977) the analyst forecasts are of less value given the cost of acquiring the forecasts. Producing the time series forecasts requires significantly less resources than attaining the analyst forecast, whereby the time series models would be the better choice if the forecasts are equally precise (Elton and Gruber 1972).

Given the above arguments, the time series models are indicated to perform well under a set of general conditions. First of all, Conroy and Harris (1987) find that the longer the forecasting horizon, the more accurate the random walk forecast is compared to the analysts. This effect is confirmed by Bradshaw et al. (2009), who find that for horizons over 1 year the random walk model is more accurate than the analysts. For the ARIMA model, the same results are apparent in Pagach, Chaney, and Branson (2003). Further, Conroy and Harris (1987) find that a random walk outperforms the analyst predictions when the forecast is made in the beginning of the fiscal year, while analysts are more accurate comparatively speaking towards the end. This result arises mainly due to the negative correlation of the forecast horizon and the random walk accuracy and not the increased accuracy of the analyst forecasts. At the same time, the random walk process becomes more accurate when annual earnings are used for the forecast (Little 1962), which ties back to the random walk being superior on longer horizons. The length of the forecast plays a very significant role in many settings such as when using the earnings estimates for valuation purposes. In this setting, Allee (2008) also provides evidence of the time series forecasts being superior.

Second of all, the size of the considered firms is an essential factor to consider, where especially for smaller firms the random walk and ARIMA models are better predictors than analysts (Bradshaw et al. 2009 ; Pagach, Chaney, and Branson 2003 ; Branson, Lorek, and Pagach 1995). The advantage mainly arises from the fact that fewer analysts follow small firms, so the information available to analysts is less significant and hence provides less of an advantage. This also ties closely together with small firms having less publicly available information that the analysts can integrate in their forecasts.

Third of all, as stated above the variability in earnings is positively correlated with the random walk superiority. Pagach, Chaney, and Branson (2003) find that the ARIMA model is more precise for firms in fewer lines of business, where the lines of business serve as a proxy for the stability of earnings. The underlying assumption is that the more diversified the company is, the more stable the earnings are. This finding also corresponds to that by Bradshaw et al. (2009) for the random walk, where analysts are less superior for companies with more significantly changing earnings. The main reasoning behind is the

analyst bias to predict smooth earnings, which the random walk and ARIMA model are not condemned to.

3.3.2 Analyst Forecasts

In general, there are several arguments why analyst forecasts are more accurate than time series models. Firstly, the analysts have a *timing advantage* (Brown, Richardson, and Schwager 1987), which indicates that the analysts can integrate more and more information as the year develops. Thus, just before the actual earnings are announced the analyst will have almost as much information as the company announcing it. The time series model is not able to update throughout the year as no new actual earnings have been published, which puts it at a disadvantage.

Secondly, the analysts have a *contemporaneous advantage* (Brown, Richardson, and Schwager 1987), which means that at the date of the forecast, the analysts have more information available such as industry related metrics to compare to. The time series models only look at past earnings and for the random walk it only considers the very latest earning. Therefore, when making the forecast the analyst will have much more information available that enables a more accurate forecast.

Third, since the time series models are available to analysts, the analyst can integrate them in their forecast to increase the accuracy (O'Brien 1988). The analysts can use the time series forecast as point of departure and enhance them by adding the specific knowledge they possess. This can materialize in analysts knowing what items are transitory and which are permanent as that enables them to a much larger extend to identify the relevance of previous items in next period's earnings. An example hereof is if a company's earnings in one period are affected by a lost lawsuit, which will not continue to the next period. Then an ARIMA model with $p > 0$ will put heavy weight on the last earnings despite the transitory nature of the law suit. In this case, the analysts will be able to improve the ARIMA model by adjusting the forecast to solely rely on the permanent items.

Finally, Newbold, Zumwalt, and Kannan (1987) confirm the analysts' superiority to the ARIMA model due to their quicker adaptation to economic events. This links heavily with the last two points, but is still a separate advantage. It indicates that if an economic event such as the financial crisis has started to affect the company to a small degree in the latest earnings, then the analyst will be better able to predict the magnitude of the effect on the next period. Here, the time series model would only incorporate it to a limited degree in the next period, even though other external parameters show clear signs of the economy going into a financial crisis. Thus, the ARIMA model will not be able to integrate the additional external information as it relies solely on the past earnings.

Based on the above arguments, there exists several conditions under which the analyst forecasts are most precise. Firstly, the horizon of the forecast remains the most important condition for analysts to be superior to the time series models. The length of the forecast is negatively correlated with the analyst accuracy, so that on short horizons below a year the analyst is most precise (Pagach, Chaney, and Branson 2003 ; Bradshaw et al. 2009 ; Brown et al. 1987). Even on horizons of a year the analysts are considered more accurate Brown et al. (1987).

Secondly, when making the forecasts based on quarterly earnings, analysts also have superior accuracy (Brown 1993 ; Hopwood, Mckeown, and Newbold 1982). This also ties to the above condition of the shorter time horizon.

Thirdly, the analyst accuracy is negatively linked to the time until the actual earnings are announced, meaning that the shorter time until the announcement, the more accurate the forecast will be. The reasons hereof is the timing advantage (Brown, Richardson, and Schwager 1987).

Finally, Peek (1997) identifies the prior precision in forecasts to be the most significant factor in the companies where analysts are superior in their forecasts. Thus, the more accurate previous forecasts have been, the more accurate future forecasts will be.

3.3.3 Alternative Forecasting Methods

In addition to the time series and analyst forecast models, some alternative models have been proposed to challenge the two dominant methodologies. Thus, attempting to find a more superior model is not a new discipline, though none of the alternative models have truly gained traction and fellowship in the literature. This section will elaborate on the alternatives along with their advantages, disadvantages and applicability. In general, the alternative models can be categorized in 3 groups. The first utilizes company, industry or macroeconomic variables to forecast the company's earnings, which is the category this thesis falls into as well. The second seeks to adjust the already generated analyst and time series forecasts either by combining them or specifically adjusting for certain trends and biases. The third group introduces new statistical models similar to the time series models but with certain variations.

The first group of models seeks to utilize the available financial data on companies in a different way. The time series models only utilize the past earnings of the company and thus neglect a vast amount of financial data on the company itself, its industry and its macroeconomic environment. Bansal, Strauss, and Nasseh (2015) generate a model that utilizes 21 variables including both firm-specific and macroeconomic variables to increase the information gathered. The firm specific variables mainly evolve

around the numeric information found on the financial statements of the company such as stock return, gross margin, accounts receivables growth, sales growth, inventory and dividends. Where the economic variables utilized are factors such as S&P 500 Dividend yields, treasury bill rates, AAA bond yields and inflation. These predictors are combined with the current earnings and an error term to generate the estimate of the future earnings following the formula:

$$eps_{t+\tau}^{\tau} = \alpha + \sum_{j=0}^{l_1-1} \beta_j eps_{t-j} + \sum_{j=0}^{l_2-1} \gamma_j predictor_{i,t-j} + error_{t+\tau}^{\tau}$$

Here, the left side indicates the forecasted earnings pr. share, and the right side includes a constant, the current earnings pr. share, one or more of the 21 predictors and an error term. This model and its mathematical formulation is especially relevant as there are several similarities to the model of the thesis as explained further in the methodology section.

The model of Bansal, Strauss, and Nasseh (2015) is especially relevant under the fulfillment of 3 conditions. The first is under scenarios where a single model is not able to fully explain the data-generation process and hence replicate it. The second condition is that there exists a large number of variable that can potentially predict the dependent variable. Finally, the last condition is that variables are correlated in ways that can change over time so that the model configuration might require to be adjusted. They argue that the earnings forecasting setting fits these conditions to a large extend as there is no model fully describing the earnings changes, there are many variables both for the firms and its environment available and the central variables might change over time depending on the economic situation. In support of this argument they find that their model outperforms the time series models, especially when including several predictors in the model. Given these results, the model of the thesis will draw inspiration from their methodology.

However, a few significant disadvantages are present in the model. Firstly, Bansal, Strauss, and Nasseh (2015) uses solely listed firms with publicly available information. This mean that the firm-specific variables they attain will not be possible to get from non-listed firms, which limits the applicability of the model significantly. Secondly, the data sample only contains 30 firms whereby it is not possible to determine the representativeness of the sample. Thirdly, their dataset contains solely large firms, whereby their findings cannot be concluded applicable to smaller firms despite them being listed on an exchange. This is due to the inherent differences in the earnings development between the two types of firms driven by factors such as more unstable earnings for small firms. Thus, the approach of Bansal, Strauss, and Nasseh (2015) cannot be concluded to be generally applicable, whereby it is still necessary

to investigate if another model can produce a more broadly tested methodology with superior forecasting accuracy.

Another model in the first group of methodologies is presented by Reverte and Guzman (2010) who focus on circumventing the transitory nature of elements in past earnings. Hence, their methodology seeks to capture only the persistent information stored in the earnings to increase the predictability of future earnings. They do so through the variable *relative efficiency* that they define as "*the inherent ability of a firm – as compared to other similar firms – to make the most productive use of available resources*". More specifically, it is measured by the firm's ability to generate an output of revenue based on inputs from material, labor, capital and overheads. In formulating the forecasts, they compare a basic model solely including current earnings and book value to their own model where the relative efficiency is included. Their model is defined from the formula:

$$NI_{i,t+1} = \beta_0 + \beta_1 NI_{i,t} + \beta_2 BV_{i,t} + \beta_3 EFFIC_{i,t} + e'_{i,t}$$

Where the left side is the forecasted net income and the right side includes a constant, current net income, current book value, the relative efficiency and an error term. The structure of the formula is similar to that of Bansal, Strauss, and Nasseh (2015), though the main difference is its focus on the relative efficiency solely, instead of integrating other variables.

Reverte and Guzman (2010) make three important findings that the methodology of the thesis will take heavy inspiration from. The first finding is that adding another variable to the current earnings can significantly improve the forecasting accuracy, which confirms the findings of Bansal, Strauss, and Nasseh (2015). Secondly, they are able to increase the accuracy by comparing the company to be forecasted with similar companies and thereby increase the amount of information gathered. Thirdly, they make their conclusions utilizing a dataset of 1939 small and medium-sized firms, which is critical to the thesis as its dataset contains mostly smaller firms with only a small fraction being large companies. This also increases the general applicability of the results and enhances the value added.

However, the article contains some noteworthy disadvantages. First of all, the analysis solely compares the model with the relative efficiency term to the model without the relative efficiency term, whereby it does not compare to time series models or analyst forecasts. Thus, despite its two important findings it does not contribute significantly to the discussion of the time series and analyst forecasting superiority. As this discussion is the foundation of the thesis, the conclusions of the model can only be treated as having an indicatory nature for further analysis. Secondly, the model is very narrow in its view on what affects earnings. Despite finding that the relative efficiency improves the forecast, it is not able to

identify if other measures can provide a greater improvement in accuracy. Thirdly, the relative efficiency consists of rather specific information from the income statement such as revenues, personnel expenses and cost of material used, whereby it can only provide forecasts for companies where this information is available. This is a limiting factor in larger samples, since it might not be explicitly stated in the available data, for instance the publicly available data on non-listed Danish companies does not always include revenues and cost of materials used.

The second group of models takes the analyst and time series forecasts as point of departure and attempts to find adjustments or combinations of the forecasts that can increase the accuracy. One of the papers in this group is Lobo (1991) that tests 5 different combinations of analyst and time series forecasts to reach the one with the highest accuracy. The notion of combining forecasts to increase the accuracy is supported from several sources in the literature (Newbold, Zumwalt, and Kannan 1987 ; Conroy and Harris 1987 ; Kim 1996). The underlying reasoning is to circumvent the biases found in the individual models and to put more emphasis on the models that are performing better. Thus, this way of forecasting recognizes that there are differences between when the forecasting methods are superior, which it attempts to capture and combine into a generally superior model (Conroy and Harris 1987). In this setting, the traditional combination places equal weight of the individual forecasts to derive the final estimate. However, Lobo (1991) finds that this manner is the least accurate method and instead an unequal combination generated from cross-sectional data is more accurate.

Similar findings are derived by Lo and Elgers (1998) who point to 4 main adjustments that increase the accuracy. The first is to adjust for symmetric and forecastable errors in past earnings that can be avoided in future earnings predictions. The second is to combine the analyst and time series forecasts based on previous accuracy of the methods. Thirdly, they adjust for the over-optimism of the analysts. Finally, they adjust for the past general accuracy of the analysts in predicting stock returns.

Hence, the second group of alternative models serve as a set of important indicators of where the analyst and time series models are not sufficiently accurate. Thus, they indicate a clear need for a type of model that can combine the best aspects of the analyst and time series forecasts in order to circumvent the biases of the individual models. This notion will be integrated in the model of the thesis. However, the second group of models are fundamentally relying on analyst forecasts in their methodologies, whereby they can only be applied to listed firms. Thus, they are not able to contribute with a generally applicable model.

The final group of alternative models contains statistical methodologies and regressions that aim to generate better models than analysts and time series models. These models exist in many different varieties and will merely be exemplified in the thesis for the sake of completion. Ardalan (2016) provides a framework for such a model named the *Error Correction Model* (ECM), which investigates the relationship between a company's earnings and dividends. The statistical properties of the relationship between the two make it possible to use both in the forecast of earnings (Ardalan 2016), which indicates that earnings information is also stored in the dividend policy. This is an important finding as it confirms Bansal, Strauss, and Nasseh (2015) and Reverte and Guzman (2010) in that adding more information than current earnings increase the accuracy of the forecasts.

Similar investigations are performed by Jadhav, He, and Jenkins (2015) who produce 3 models, where especially two, *Linear Regression* (LR) and *Multilayer Perceptron* (MLP), produce accurate results. However, this group of alternative models suffers from 3 main disadvantages. Firstly, the number of companies included is very small, which despite the long data horizons only provides a very narrow picture. Secondly, only large and listed companies are tested, whereby the general applicability is brought into question. Finally, they do not explicitly compare to both the time series models and analysts, so their contributions to the discussion of the thesis is limited.

3.4 Theory Conclusion

When comparing the time series models of a random walk and ARIMA with analyst forecasts, the time series models have several advantages. They are less prone to subjective biases that the analysts are affected by, such as earnings smoothing, getting emotionally engaged with the company in question, and the incentive structure in place promoting "buy" recommendations over "hold" or "sell" recommendations. These biases bring the objectivity and accuracy of the analyst forecasts into questions. Further, the literature arguing for analyst superiority has several flaws that makes it prone to accepting analysts as superior. These include relatively old data, small sample sizes with mainly larger firms and short forecasting horizons. Most of these aspects directly contradict the conditions for when time series models are superior and thus create a natural bias for accepting analyst superiority.

On the other hand, the analyst forecasts have several advantages over the time series models. The two main aspects are the timing and contemporaneous advantages that allow the analysts to base their forecasts on more information both on the date of the forecast and as the year develops. Further, the analysts are able to generate the time series forecasts themselves and only update them if they have additional information that they know will have a significant effect on the forecast. This could for

instance be when the analyst has knowledge of transitory items or the effects of economic events that the time series models will not be able to integrate to a large extend.

In addition, a set of alternative models can be utilized in comparison to the analyst and time series forecasts. These models integrate several useful aspects including increased accuracy from adding more data and variables as well as from integrating information from similar companies. However, the majority of the alternative models are lacking general applicability and only few are directly compared to the analyst and time series forecasts, whereby their contribution to this discussion is limited. Thus, they are to a larger extent used to provide support of the general idea of challenging the time series and analyst models rather than actually finding a better model than the one of the thesis.

Thus, the most accurate of the existing models will arise from the time series or analyst methods. Though, given the significant arguments present for both cases, it is not a straight forward task to determine which of the two sides that contain the most influential arguments. Therefore, the deciding factor in terms of the proposed accuracy will be which of the two sets of the conditions that apply the most to the research design of the thesis. As will be elaborated in the methodology section, the data of the thesis contains more than 350.000 Danish firms with earnings data over an 18-year period from 1994 to 2012. At the same time, the firms covered are to a great extend small or medium-sized, unlisted firms that most likely have no analysts following them. The produced forecasts will be based on yearly data and have a forecast horizon of up to 8 years. These research design aspects all fall under the conditions where time series models perform best, whereby the conclusion to this theory section is that the time series models are assumed more precise for this thesis. Hence, in the analysis and discussion sections the models of a random walk and ARIMA will be the main comparative benchmarks for determining whether the models developed in the thesis are more accurate than what the literature body currently entails. The analyst forecasts will also be analyzed but only for a smaller sample of companies.

4. Methodology

The section on methodology will discuss the manner in which the results of the thesis are derived. Therefore, it will initially outline the data foundation utilized and the characteristics it entails. This will include a discussion of the variables used and the necessary formatting of the raw data. Then the section will turn towards the forecasting methods applied to generate the predictions. Finally, the method for calculating accuracy will be discussed.

4.1 Thesis Data

This section will focus on the type and extend of the data along with its applicability to the analyses in question. As the raw data file did not embody the structure required for the analyses, the reformatting process will also be elaborated upon.

The raw data file is attained through the Department of Accounting and Audit (ACC) at Copenhagen Business School, which is pulled out from the databases of Orbis (ACC 2014). In the end, Orbis attains the data from the CVR register of Erhversstyrelsen. The data consists of 352,496 firms and 2,891,384 firm years, which is 8 years pr. company on average.

For each year, 152 variables are included where 12 are characters such as CVR number and industry and 140 are numeric such as EBIT and Total Assets. The data covers listed and non-listed companies in Denmark, Greenland and the Faroe Islands in a broad range of 21 industries including both large, medium and small-sized firms. The data spans from 1994 to 2012 (ACC 2014) and attempts to capture as much data as possible by applying few restrictions on size, industry, age or financial metrics. However, to be included in the analyses the companies are required to have data on EBIT and balance (measured as total assets) for the full data horizon of the analyses in the thesis. In addition, the companies need to have data on the full forecasting horizons considered, since the produced forecasts are compared with the actual values to calculate the accuracy. The exact ranges of the data and forecast horizon vary in each analysis but cover up to 12 years of data. Further, the company needs to have an industry and an age of the most recent accounting year.

As touched upon in the theory, the dataset contrasts that of the literature as seen in Table 4 below and embodies several of the features that favor time series models (Bradshaw et al. 2009). Firstly, the data is very recent as the latest observations are from 2012 in contrast to the average of the literature being 1986. Thereby, the applicability to current companies is improved since the data includes more recent earnings development patterns that are more relevant to use in forecasting today compared to the patterns from over 30 years ago. Secondly, the sample size of 352,496 firms is large compared to the previous literature, where the average size of the literature is 1,369 companies. The more than 250 times bigger dataset allows for a much more diverse set of companies, which leads to a more generally applicable framework. Especially the fact that not only listed companies are included increases the applicability significantly as the findings of the thesis can be applied to non-listed companies. Finally, the size of the firms included in the dataset varies greatly. In the dataset of the thesis, only 319 firms or 0.1% of the firms and firm years arise from listed companies, while the remaining 99.9% of the data

stems from non-listed companies. This shapes a very clear contrast to the literature as it mainly deals with listed companies. Thus, the general applicability is increased further and enhances the relevance of the findings as the majority of companies in the world are non-listed.

Table 4: Thesis Data Structure

	Sample Size (Firms)	Latest Year (Average)	Latest Year (Maximum)	Data Horizon (Years)	Forecast Horizon (Years)
Literature	1,369	1986	2010	18	2.0
Thesis	352,496	2012	2012	2 to 10	1 to 8

Nevertheless, the data gathering methodology utilized before the dataset is obtained contains some noticeable disadvantages highlighted by ACC (2014) that could have an effect on the analyses of the thesis. Firstly, there are differences between how Orbis defines and calculates the accounting measures and firm characteristics. This can cause misleading interpretations and further calculations of the variables given the incoherence.

Secondly, the database initially had significant challenges with missing data and duplicate values. This is attempted solved through merging with additional databases from Orbis, yet the merging process has to make some simplifying assumptions. It assumed that all non-numeric variables remained the same for each year, for instance that the companies did not change industry. This can be a critical assumption as the thesis utilizes the industry as a key component in some of the analyses. Thus, if it has changed during the data period, the company might be more appropriately located in another portfolio during the analyses. However, the probability of the company changing between the 21 overall defined industries is considered very small. Further, during the merge of the databases the missing values from previous years are filled out by values from later databases. As an example, if a company is missing data on EBIT from 2011 then a database from 2013 with the value for 2011 is included to fill out the missing value. This can be misleading if the company has made changes in arrear of their accounting information, where the most correct value would have been the originally reported one from 2011. However, this issue is not viewed to have a significant impact of the analyses as making accounting changes in arrear is not common practice and happens infrequently.

Third, the variable *FOUND* is assumed to cover the founding year of the company, but it can also contain the latest year where the registered information of the company is changed. Hence, if a company changes its name, the variable FOUND will update to that year and the analyses will assume it is the founding year of the company. This can lead to significant distortions in the portfolio groupings of the analyses as they are based on a relative year to when the company is founded. As an example, a

company that after 15 years of operating changes its name will have its 16th financial year stated as the 1st year in the analyses since the FOUND variable is updated. This will skew the portfolio of the company as the financial metrics will not be comparable given its different stage in the company's life cycle. Finally, there are some companies with partial CVR numbers, which could be caused by an incorrect addition of a company. This is also indicated by the fact that despite the dataset stating it contains 352,496 firms, Danmarks Statistik (2014) only identifies 295,220 companies in Denmark in 2012. The mismatch can be partially explained by Danmarks Statistik (2014) not including the companies from the Faroe Islands and Greenland, but this cannot provide the full explanation. Hence, there are some of the CVR numbers that are concluded to be incorrect, though this has not been possible to filter out. Further, the dataset includes companies that have been terminated between 1992 and 2012, which would not be included in Danmarks Statistik (2014).

However, the outlined disadvantages in the data collection methodology are not considered to significantly affect the comparative results as all models will be affected equally. Nevertheless, they limit the number of companies to be used in the analyses, especially due to missing observations for the accounting year and EBIT. Hence, in the analysis only 59,404 companies are utilized, where 846 are the companies to be forecasted and the remaining 58,558 are used to generate the forecasts. The 846 companies are chosen randomly from the dataset to avoid selection biases.

The final step of the data preparation is to convert the data from the ACC format to having the format required for the data driven approach. This requires a two-step process, where the first conversion is to go from having a firm year for each row to having the unique CVR numbers in each row with the yearly EBIT data in the columns. The second step is to transform the dataset from having each firms' earnings categorized by calendar years to the refined data structure of having the earnings categorized by firm year. The second step is illustrated in Figure 1a and Figure 1b showing the original data structure and the refined data structure, respectively.

Figure 1a: Original Data Structure

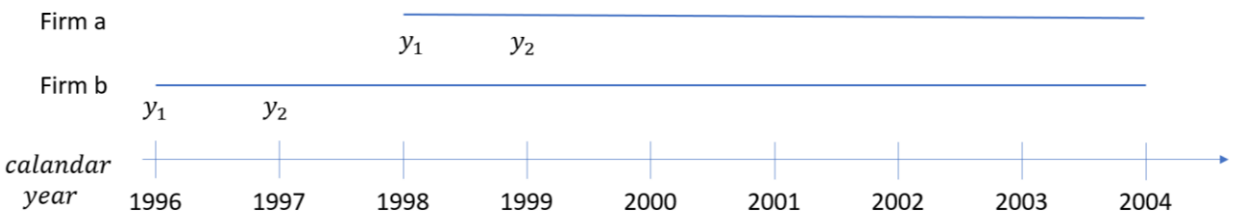
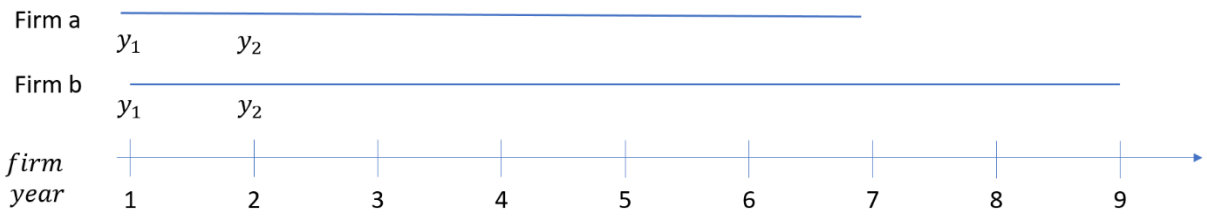


Figure 1b: Refined Data Structure



4.2 Forecasting Methodology

The forecasting methodology is essential in the thesis as this is the core way it differentiates from previous literature. Given the extensive dataset previously described, the thesis is able to generate its own forecasting technique, the data driven forecasting, through a grouping of variables. This method will be contrasted with the more generally accepted and utilized forecasting methods from the time series models of the random walk and ARIMA, in order to test which of the methodologies is superior. In addition, the data driven forecasting will be compared to a small samples of analyst forecasts to provide an indication of the superiority of the models.

Therefore, the first part will concern how the data driven forecasting is performed, the second part will concern how the time series models derive their forecasts, and the third part will outline the methodology for attaining the analyst forecasts.

4.2.1 Data Driven Forecasting

The data driven forecasting is a way of grouping similar companies based on a variety of measurable variables. The first step in the method is to identify groups or portfolios of comparable companies. Once portfolios of similar companies have been formed, the development in earnings in the portfolios can be identified for each year. Then, the development in the portfolios is utilized to predict the future development for other companies exhibiting the same characteristics. The underlying notion of the model is that similar companies will exhibit a similar development in earnings. Hence, the data driven model can be formulated in the below expression:

$$EBIT_{t+i} = EBIT_t \times \left(1 + egp_{j,1}\right) \times \left(1 + egp_{j,2}\right) \times \ldots \times \left(1 + egp_{j,i}\right)$$

Where *t* is the data horizon, *i* is the forecast horizon, *egp* is the EBIT growth of the portfolio and *j* is the variable(s) the portfolio is based on. The methodology is probably best explained through an example. The example will take its point of departure in a single company needing to be forecasted. The company has been operating for 6 years and a venture capital fund wants to produce a forecast for the next 3

years of earnings. To do so, they have identified a portfolio of comparable companies within the same industry that have similar EBIT in their 6th year of operating and which has at least 9 years of EBIT available in total. Then, the average EBIT of the portfolio of companies is attained for each year, so the yearly growth can be calculated. Here, the average growth in the portfolio from their 6th to 7th year of operating is 10%, while the next two years exhibit growth rates of -5% and 20%. At the same time, the company to be forecasted has an EBIT of 100 in its 6th year of operating.
Hence, in this example the data horizon is 6 years, the forecast horizon is 3 years, the EBIT in year 6 is 100, the variables used for forming the portfolio are EBIT&Industry and the growth rates in the portfolio for the 3 years are 10%, -5% and 20%. Then, the methodology will predict the earnings in year 9 to be:

$$EBIT_9 = EBIT_6 \times \left(1 + egp_{EBIT\&Industry,1}\right) \times \left(1 + egp_{EBIT\&Industry,2}\right) \times \left(1 + egp_{EBIT\&Industry,3}\right) \leftrightarrow$$

$$EBIT_9 = 100 \times (1 + 10\%) \times (1 - 5\%) \times (1 + 20\%) = 125.4$$

If the earnings in years 7 and 8 are desired to derive explicitly as well, this can easily be done by following the same methodology:

$$EBIT_7 = EBIT_6 \times \left(1 + egp_{EBIT\&Industry,1}\right) = 100 * (1 + 10\%) = 110$$

$$EBIT_8 = EBIT_7 \times \left(1 + egp_{EBIT\&Industry,1}\right) = 110 * (1 - 5\%) = 104.5$$

$$EBIT_9 = EBIT_8 \times \left(1 + egp_{EBIT\&Industry,1}\right) = 104.5 * (1 + 20\%) = 125.4$$

The only difference here is that three forecasts are made with a forecast horizon of 1 year for each forecast, while the data horizon is 6, 7 and 8, respectively. However, as visible the final results are the same.

More specifically, to produce the forecasts using the data driven methodology, the following approach is undertaken. The method used depends on the type of variables chosen to form the portfolios and in this context, there are two types of variables considered: *dynamic* and *static variables*. For the dataset of the thesis, the overview of variables is shown below in Table 5. A dynamic variable is defined as being dependent on the year the portfolios are formed in, whereby the grouping of portfolios will differ based on which year the grouping is performed in. An example hereof could be total assets, which changes each year whereby the division in portfolios will be different each year. The static variables are defined as being constant despite changing the year the portfolios are formed in. An example hereof could be industry, which remains the same for a company despite choosing another financial year. In addition,

age is considered as static since it remains constant for the same dataset, as it is only if another year of data is added to the dataset that the company's age is changed.

Table 5 below provides an overview of the utilized variables and their category. In the thesis, dynamic variables are EBIT, balance measured by total assets and growth in EBIT, while the static variables are industry and age. The reason for this selection is mainly driven by data availability, since these are the main variables with consistent data for a sufficient number of companies.
Further, the table shows the number of portfolios each variable is categorized in, where the dynamic variables can be divided in 5 or 10 portfolios according to the 20^{th} or 10^{th} percentiles of the data, respectively. Here, the industry and age variables are divided based on the number of industries and ages present in the dataset. For the age variable, the oldest company in the dataset is Tivoli A/S that was founded in 1843, which is 169 years before the end of the dataset in 2012. Thus, the dataset has at least one company founded in each year since 1843 except for 3 years, which brings the total number of ages present to 166.

Table 5: Variable Overview

Variable	Category	# of Portfolios	Portfolio Method
EBIT	Dynamic	5 or 10	According to each 20th or 10th percentile
Balance	Dynamic	5 only	According to each 20th percentile
Growth in EBIT	Dynamic	5 or 10	According to each 20th or 10th percentile
Industry	Static	21 industries	According to each of the 21 industries
Age	Static	166 ages	According to each of the 166 ages

For dynamic variables, the forecasting is performed in the following manner, which will also be illustrated with an example below. First, a *grouping year* is chosen to determine which year the portfolios should be formed in. There are several ways in which the grouping year can be chosen, though the arguments are strongest for choosing one of the following two options. The first option is to choose the grouping year based on the latest available firm year of the company to be forecasted, so in the example above if a company's 7^{th} year is to be forecasted then the grouping year in the dataset will be year 6. This allows for an alignment of the company life cycle, so the companies in the dataset will be in the same phase of the life cycle and hence increase the applicability of the forecast. The second option is to choose the grouping year as the latest available calendar year in the dataset. This allows for the most updated data and takes current economic conditions into account as for very old companies, their 6^{th} year could have occurred in a completely different economic climate that might not be

applicable today. However, the second method could lead to suboptimal groupings if there are large differences in the age of companies considered. In 2017, a company with an age of 50 years will most likely have completely different EBIT and Balance values than a company with an age of 6 years, whereby it will not be grouped in the same portfolio. Nevertheless, the company might have looked very similar when it was 6 years old whereby it actually could contribute with a relevant earnings development pattern. Thus, the thesis will use the first way of determining the grouping year, which is mainly driven by being able to match the phases in the company life cycle and to avoid the disadvantage of the second manner outlined above. Whether the analyses would be affected by choosing the grouping year in a different manner will be for further research to investigate.

The second step in forecasting the dynamic variables is to divide the dataset in a set of portfolios based on the variable in the grouping year, such as dividing in 5 portfolios based on the Balance for each 20^{th} percentile. Thus, it is required to determine the number of portfolios desired in the analysis. Arguments for choosing a small number of portfolios evolve around ensuring to have sufficient data in each portfolio, so forecasting is not based on undesired outliers. However, choosing a larger number of portfolios allows for a more nuanced range of potential developments of the company's earnings, which could produce more accurate groupings and hence forecasts. There is yet to be established a guideline for the minimum size of a portfolio, though given that averages are used in the next step it is advisable to maintain a decent size to avoid too significant weight placed on outliers.

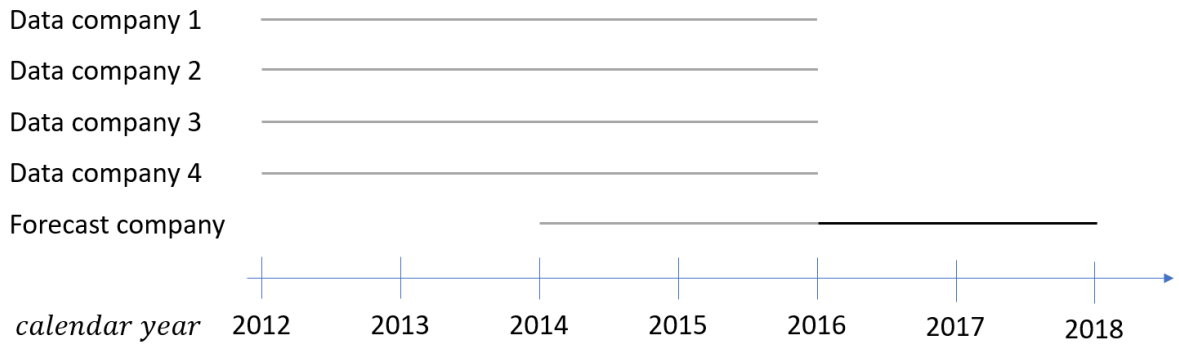
The third step is to take the average earnings for each portfolio for each forecast year in order to identify the percentage development in earnings from year to year. Thus, the result is a matrix with portfolios and firm years showing the percentage changes.

Finally, the company to be forecasted is assigned a portfolio based on the dynamic variable in its latest year to ensure the portfolio shows the most relevant development. Then the percentage development in earnings for the portfolio is applied to the latest earnings of the company to achieve a forecast of the desired horizon. The first three steps are performed once as the portfolios do not change based on the company to be forecasted, whereby for the next company in questions it is only necessary to repeat the last step and apply the relevant portfolio development to generate the forecast.

To illustrate the approach, an example is calculated in the following. Here, 4 companies represent the dataset, while 1 company is to be forecasted with a data horizon of 3 years and a forecast horizon of 2 years. The data horizons are here driven by the 4 *data companies* having an age of 5 years, while the *forecast company* has an age of 3 years. Figure 2 illustrates how a timeline for the 5 companies could look, where the data companies have data from year 2012 to 2016 and the forecast company has data

from 2014 to 2016. The forecast is then performed for 2017 and 2018, which is the firm year 4 and 5 for the forecast company.

Figure 2: Forecasting Example Timeline



The EBIT and forecasting values are show in Table 6a and 6b below, which are noted in firm years and not calendar years. To produce the forecast a dynamic variable of EBIT is chosen and it is decided to produce 2 portfolios from the data. Using the steps above, the first step in the forecasting process is choosing the grouping year. This is chosen as firm year 3 to align the companies in terms of the company life cycles. Second, the number of portfolios is chosen as 2 in order to secure that only relevant firms are compared to the forecast company. In this specific example, one could argue that there is not sufficient data to split in multiple portfolios, but for the sake of providing a simple example the data is divided in 2 portfolios. Third, Table 6b is calculated with the average earnings for each portfolio and the growth rates from year 3 to 4 and 4 to 5. Finally, the forecast company is assigned a portfolio based on the EBIT in year 3. Here, the EBIT is higher than the 50th percentile, so the forecast company is assign portfolio 1. Then, the growths in portfolio 1 of 11% and 4% from Table 6b are applied to the EBIT of the forecast company in year 3 of 90, which leads to a value of 90*(1+11%) = 100 in year 4 and 100*(1+4%) = 104 in year 5.

Table 6a: Forecasting Example, Company Data

Company	Y1	Y2	Y3	Y4	Y5	Portfolio
Data Company 1	70	80	100	120	130	1
Data Company 2	10	5	9	14	17	2
Data Company 3	100	105	107	110	110	1
Data Company 4	50	40	30	40	20	2
Forecast Company	75	80	90	***100***	***104***	1

Table 6b: Forecasting Example, Portfolio Data

Portfolio	Y1	Y2	Y3	Y4	Y5	Growth 3-4	Growth 4-5
Portfolio 1	85	93	104	115	120	11%	4%
Portfolio 2	30	23	20	27	19	38%	-31%

The above example can also be applied to the formula expressing the data driven model. Thus, the formula for the forecast of year 5 is:

$$EBIT_5 = EBIT_3 \times \left(1 + egp_{EBIT,1}\right) \times \left(1 + egp_{EBIT,2}\right) \leftrightarrow$$

$$EBIT_5 = 90 \times (1 + 11\%) \times (1 + 4\%) = 104$$

For static variables, the first two steps from the dynamic variable process are simplified significantly as the creation of portfolios is performed by combining the companies in the dataset based on their static variable such as industry, where each portfolio represents an industry. Thus, it is necessary to consider the number of portfolios arising from the static variable to ensure a reasonable balance between nuanced groupings and sufficient data points in each group. The latter is especially important to emphasize since the risk of having few observations is greater once the portfolios are not divided evenly based on percentiles, but rather based on a variable that has no guarantee of an equal division. An example hereof could be a niche industry that only few companies in the dataset belong to, which increases the risk of basing portfolio developments on outliers. Finally, the third and the fourth step are performed in the same way as under the dynamic variables in order to produce a forecast for the company in question.

When forming portfolios based on only one variable, there is a significant risk that the portfolios become too broad and lose some of their applicability. Therefore, it is relevant to test whether a combination of variables can generate a more precise prediction than single variables can. The methodology for creating multi-variable portfolios is similar to the above and falls into the following steps. Initially, if one or more dynamic variables are included the grouping year needs to be determined as discussed above. Then each company in the dataset is assigned to a portfolio for each variable included. Thus, if three variables are included with three portfolios each, a given company in the dataset will have three portfolio numbers from one to three assigned. Thereafter, an additional step is required to convert the three portfolio numbers into one overall portfolio. This is managed through a portfolio table as seen in Table 7 in the appendix, where each possible combination of the three variables is outlined and assigned a final portfolio number. As visible from the example, the number of final

portfolios is the number of portfolios in each variable multiplied with each other, which creates an exponential increase in the number of portfolios. Thus, even more caution must be applied in ensuring that each portfolio has sufficient data points to avoid relying on outliers. This is especially apparent when including static variables as outlined above, for instance if the age and industry variables are combined in the analysis the number of portfolios would be 21*166 = 3486. In this case, less than 17 companies are on average in each portfolio, which means that most likely a portion of the portfolios would be based on much fewer companies and hence be very prone to outliers. Finally, step three and four are the same as above using the final portfolios as basis of the development to generate the forecast for the companies in question.

The reason why the data driven approach to forecasting is relevant to consider is that it incorporates some of the advantages that only analysts currently possess, while still embodying many of the same advantages as the time series models contain. The main new advantage is the incorporation of information from comparable firms, so the forecast goes beyond the past earnings of the company. This will decrease the contemporaneous advantage outlined by Brown, Richardson, and Schwager (1987) as analysts go through the same exercise of gathering comparable companies and evaluating how they have developed over time in order to incorporate that in their forecasts. Any quantifiable variable that the analysts could find comparable companies with can be integrated in the data driven approach to increase the accuracy of the forecast. This is a significant change as no time series model has been able to integrate information on similar companies. Furthermore, the approach allows for integrating future stages of the company's life cycle by comparing to companies who have been through the same stages. The other main advantage is the method's ability to integrate information on the general changes in economic conditions that companies go through. This is indirectly included when taking the firm year of earnings for all companies, so that a company who started operating in 1995 can be compared with a company starting to operate in 2000 despite being in very different phases of the dot-com bubble (Stern NYU 2017). In this case, year one of earnings would be 1996 and 2001, respectively, with one being on the rise of the bubble where earnings were high on average, while the other exhibits the opposite case. Hence, by integrating the information indirectly stored in the earnings for these companies, the data driven approach can provide a picture of how earnings vary on average across economic conditions as well. This aspect is considered an advantage rather than a disadvantage since the magnitude of the dataset will ensure that the forecasts are not based on outliers, while the effect is still included.
At the same time, the method entails several of the same advantages as the time series models do, such as avoiding earnings smoothing or emotional biases in the earnings estimations that the analysts exhibit.

Further, this dataset includes very recent data with a large sample size covering a wide range of companies, which increases its applicability as previously discussed.

However, the data driven methodology also comes with some general disadvantages compared to the time series and analyst models. First, it requires significant amounts of data to generate the models as the portfolios require a certain number of firms to be relevant as elaborated upon above. This can be a critical disadvantage in replicating experiments as acquiring access to a sufficiently large pool of company data is challenging. Nevertheless, the thesis entails a large dataset and therefore does not suffer from this disadvantage. Further, for the methodology to be generally applicable, the dataset needs to contain a wide variety of firms and not just large firms that are listed on an exchange. In many settings, attaining this data proves a big challenge due to the limited public information available on non-listed companies. However, in the thesis it has been possible to attain data for both listed and non-listed companies, whereby this disadvantage does not affect the analyses significantly.
Secondly, the main difference between the data driven approach and the time series models is the integration of previous earnings of the company in question when making the forecast. This occurs either through the trend analysis of the ARIMA model or the drift term in the random walk. In the data driven method, only the latest year of the company in question is utilized as a starting point for the forecast, though all the information included in the previous years are not utilized. Thus, it does not integrate the information stored in the previous earnings, which can cause the forecast to be less accurate.
Thirdly, the data driven approach places heavy weight on the latest financial year of the company depending on which variables are considered in the portfolio groupings. However, this approach can be misleading if the latest year has a considerable number of transitory items or in any other way did not represent the company's true outlook going forward. Thereby, the company risks getting categorized incorrectly so that the expected earnings development might not be the most applicable one.
Finally, it is not straight forward to identify what a truly comparable company is. Even for companies that appear comparable such as companies of the same age in the same industry might have radically different outlooks and future earnings due to different strategies or other critical components that are difficult to quantify. Even parameters that might work well in identifying similar firms in one segment might be misleading in another. Thus, the model is forced to make the underlying assumption that in general the law of large numbers indicates that as the number of firms included increases, a mean reverting effect occurs in order for the model to be generally applicable.

In the end, the question of whether the time series models are better than the data driven method will boil down to whether past earnings of the same company is a better predictor of future earnings than the earnings of comparable companies. The results of this test will be elaborated upon during the analysis and discussion sections.

The above methodology for the data driven approach entails three key restrictions that need to be fulfilled to achieve unbiased and relevant forecasts. Firstly, to circumvent the risk of portfolios with few companies, it is essential that the dataset used is of a significant size. In this way, portfolios will statistically have a better chance of getting allocated a sufficient number of firms. Secondly, it is essential to use out of sample companies to determine the forecasting accuracy. This is especially critical for small datasets since if the company is included in the portfolio used to forecast it, the forecast will be biased towards the actual values of the company. Thus, the accuracy will appear artificially improved. Therefore, the portfolio developments are derived from a different part of the dataset than the firms used to determine the accuracy of the techniques. Finally, it is important that the same group of firms is used to determine the forecasting accuracy for each analysis in order to secure comparability. Thus, the thesis has ensured to be in compliance with all three aspects.

4.2.2 Time Series Model Forecasting

The time series models are introduced in the theory section and consist in this context of the ARIMA model and the random walk model. As their general composition and forecasting method has already been explained, this section will seek to expand on why they are relevant to compare to and what varieties of the models are utilized in the thesis.

As visible in the theory section, the majority of the literature is concerned with testing the accuracy of analyst forecasts compared to time series model forecasts. It concludes that for a dataset containing a large sample of recent data with a wide range of firms such as the one seen in the thesis, the time series models will be superior to the analysts. Hence, the time series models are the most accurate and relevant model to be used to test the data driven approach against. Further, if the data driven method proves superior to the time series models, it will then indirectly be superior to the analyst forecasts as well. This would be a major finding, especially considering the general applicability of the models due to the wide range of companies included in the data. Thus, as most companies are not listed, then whichever model proves superior for this dataset will be the most accurate model for majority of companies in general.

The analyses of the thesis will include several variations of the time series models to ensure that the findings are as profound as possible. For the ARIMA models, the statistical program of R that the thesis uses has two ways of applying the model to the data. The first one is the *auto.arima* function that takes the past earnings of the company as input and attempts to identify which ARIMA variation has the best fit. R uses the *Akaike Information Criterion* (AIC) of each potential ARIMA variation to identify which model is able to capture the tendencies in the data most appropriately (OTexts 2017). The AIC is a way of comparing the fit of statistical models, where it takes both the actual fit of the data and the generally expected fit for any dataset into account (Bozdogan 1987). Thus, R runs through the possible ARIMA models for each firm and chooses the one with the best fit, which it then uses for making a forecast. This is an efficient way of ensuring that the most fitting and accurate ARIMA model is applied. The second function is *ARIMA*, which takes a series of earnings as input along with values of p, d and q. In order to test the theory the popular models of ARIMA(0,1,1) and ARIMA(1,0,0), also called the Griffin-Watts and Brown-Rozeff models (Brown 1993), are included.

For the random walk models, R provides the function *rwf* which takes a period of actual earnings as input and then produces a forecast with a desired horizon using the random walk model. In the function, it is possible to define if R should include a drift term, so the rwf function is included both with and without a drift to ensure the broadest spectrum of the tests. In the variation without the drift, the model purely forecasts the future values as the latest actual value. However, when including the drift term the model forecasts the earnings to change each year with the drift (Hyndman 2017). These two cases exhibit very simple ways of forecasting the random walk, whereby a third random walk forecast is integrated in the analysis. This forecast utilizes the Brownian motion, where the change in earnings can be expressed as $\mathrm{dx} = \mu\,\mathrm{dt} + \sigma\,\mathrm{dW_t}$, where $\mathrm{dW_t} = \epsilon_t\sqrt{\mathrm{dt}}$ (Liang 2003). This indicates that the difference in the variable, dx, can be expressed as the mean of variable, μ, multiplied by the time increment of 1 year dt, added with the variance of the variable, σ, times an increment of the Wiener process, $\mathrm{dW_t}$. The latter is defined by a random error term at time t, ϵ_t, multiplied by the square root of the time increment, dt. The underlying notion of this type of process is that the next value will be the current value plus its mean and a random shock scaled by the variance of the variable. Thereby, it is another way of expressing the general random walk process defined in the theory section. The difference between the Brownian motion and the random walk with a drift from R is found in the shock or Wiener process that is added to the forecasted value. This is in order to take the unpredictable variations in the variable into account, which has the effect of increasing the volatility of the forecast. An argument supporting this type of model is that it overcomes the excessive earnings smoothing applied by analysts.

However, the shocks might also lead to more significant outliers. Thus, the method is included to create a broader range of models to test.

4.2.3 Analyst Forecasting

To directly compare the data driven forecasting with analyst predictions, a small dataset of forecasts is gathered. The dataset contains 30 listed companies from the dataset of the thesis, while the analyst forecasts are attained through the Bloomberg database. The companies are selected through 2 criteria. The first is that they have 10 years of data in the dataset of thesis in order to be able to generate combinations of data and forecasting horizons that are comparable to those made for the other analyses. This criterion narrows the number of companies to a large extend due to the shortcomings of the dataset as commented earlier. The second criterion is that the companies have analyst forecasts from 2007 to 2010 in Bloomberg as that period is chosen for the forecasting horizon to maximize the data availability from the dataset of the thesis. For smaller Danish stocks, this criterion is rather limiting as despite the large database of Bloomberg, it requires analysts to perform predictions of the stocks. Thus, smaller deviations in this criterion are accepted to keep the number of included companies decent, whereby 6 companies do not have a forecast for 2007 but only 2008-2010. In the end, 30 companies are left as shown in Table 8 of the appendix. This amount is not sufficient for ensuring a certainty in the results since the sample only constitutes 2% of the average sample size of the literature as seen in Table 3. However, it will still serve as an important indication of the results a larger sample could produce.

In this context, Bloomberg is chosen as provider for 2 reasons. Firstly, it is easily accessible and the historic forecasts are an integrated function. Secondly, the quality of the data and the number of analysts are high compared to other software, whereby the most accurate estimates are deemed to be on Bloomberg.

However, Bloomberg has two disadvantages worth mentioning. The first is its limitation on data for smaller Danish companies as noted earlier. The second is the drawback of not being able to identify the forecast horizon of the historical forecasts. Thus, it is not certain whether the forecast of 2008 is performed on 1-Jan-2008 so it is a one year forecast, or if it is produced before and constitute a forecast horizon of more than one year. Therefore, it is necessary to produce 2 tables showing the result of each scenario as will be elaborated upon in the analysis section.

4.3 Forecasting Accuracy

The forecasting accuracy measures and a discussion on their relevance has already been provided in the theory section. Thus, this section will elaborate on how the accuracy measures are implemented in practice and how that can affect the analyses.

The accuracy calculations work in the same way for all forecasts and evolve around the 6 included measures as displayed in Table 2 in the theory section. Once the forecasts from the previous sections have been performed, the actual values of the same companies for the forecast horizon are extracted from the data. Then the forecasted values are subtracted from the actual values to attain a matrix with the differences. Thereafter, the 6 accuracy measures are calculated for each company using the formulas in Table 2. In order to attain one set of accuracy measures for each method, the average is taken for the test firms for each forecasting method. Thus, the resulting tables contain 6 columns with accuracy measures along with 6 rows for the time series models and 13 rows with the data driven methods.

Taking the simple average of the test firms can cause a bias towards zero in the non-absolute accuracy measures of the ME and MRE. This could arise from outliers on either side offsetting each other and making it appear as if the ME or MRE is close to zero, despite the absolute value being far away from zero. Thus, these two measures will be utilized to a lesser extend as the absolute measures are deemed less prone by this bias. However, taking the absolute value will not indicate whether the forecast is generally forecasting above or below the actual values, which can prove as useful information as well.

Further, as the dataset contains many small companies with EBIT close to 0, the relative measures of MRE, MAPE and MSRE will experience a bias. This is caused by an absolute small change that can have a significant relative value. Especially for values around zero, when a company moves from an absolute small loss to a small profit or visa versa, then the relative value risks being very high and negative. The latter negative effect mainly causes a bias in the MRE measure, as MAPE and MSRE convert to positive values. However, the same issues are present in the literature (Albrecht, Lookabill, and McKeown 1977), whereby it is not deemed a severe disadvantage of this research design.

Furthermore, as the Brownian motion method utilizes random elements, it will generate a new estimate every single time it is executed. In order to take those deviations into account and to approach a more normalized solution, the Brownian motion forecast is run 1000 times and then only the average is compared to the other forecasting techniques. This relies on the assumption of mean reversion and the

law of large numbers, and could potentially suffer from the aforementioned bias when taking the means of the non-absolute measures.

5. Analysis

This section will analyze the accuracy of the data driven approach compared to the dominant models presented by the theory. To ensure as robust results as possible, this will be performed over multiple data and forecast horizons along with analyzing several accuracy measures.

The analysis section will first consider the included variables and how these are combined in the 13 data driven analyses. This will also include how the variables are correlated and how that correlation affects the analyses. Then, the data driven approach is analyzed in comparison to the time series models and the analyst forecasts. In addition, the thesis investigates other relevant results relating to which of the specific models are most accurate and how the number of portfolios and variables affect the data driven approach.

5.1 Analyses and Variables

As described in the methodology, the variables selected for the data driven analysis are EBIT, total assets, growth in EBIT, industry and age. The reason for selecting these variables mainly involves the data availability to ensure as many companies to be included in the analysis as possible. In the analysis, the 5 variables are both tested individually and in a combination with other variables which leads to a total of 13 data driven analyses. These consist of 7 analyses with individual variables, 3 analyses with two variables combined and 3 analyses with three variables combined. In addition, out of the 13 analyses 3 are tested with 10 portfolios instead of 5 in either the absolute EBIT or the EBIT growth variable. This is done to identify the impact of changing the number of portfolios.

The analyses performed are displayed below in Table 9, which shows the specific and general models used from the time series and data driven methodologies. In addition, for the data driven models it shows the number of variables, the number of portfolios the individual variables use and the resulting total number of portfolios when the variables are combined.

Table 9: Model Overview

Specific Model	General Model	# of Variables	Individual Variable Portfolios	Total Number of Portfolios
AutoArima	Time Series	NA	NA	NA
ARIMA(0,1,1)	Time Series	NA	NA	NA
ARIMA(1,0,0)	Time Series	NA	NA	NA
rwfNoDrift	Time Series	NA	NA	NA
rwfDrift	Time Series	NA	NA	NA
rwBrown	Time Series	NA	NA	NA
EBIT5	Data Driven	1	5	5
Industry	Data Driven	1	21	21
EBIT5&Industry	Data Driven	2	5 & 21	105
Age	Data Driven	1	166	166
Balance	Data Driven	1	5	5
EBIT5&Balance	Data Driven	2	5 & 5	25
Growth5	Data Driven	1	5	5
EBIT5&Growth5	Data Driven	2	5 & 5	25
EBIT5&Growth5&Balance	Data Driven	3	5 & 5 & 5	125
EBIT5&Growth5&Industry	Data Driven	3	5 & 5 & 21	525
EBIT5&Growth10&Balance	Data Driven	3	5 & 10 & 5	250
EBIT10	Data Driven	1	10	10
Growth10	Data Driven	1	10	10

In the multi-variable analyses, the underlying reasoning of how the combination of variables is chosen requires some elaboration. The overall reason for combining variables is to integrate multiple dimensions of a company to ensure as much relevant information is utilized to perform the earnings forecasts. A way of identifying how variables interact is by determining their correlations. Here it can be argued that two uncorrelated variables would span different dimensions to what makes a company and hence provide more information when combined. Therefore, Table 10 shows the correlations between the numeric variables in consideration and the one period ahead earnings, $EBIT_{t+1}$, which serves as an indication of the correlation with future earnings. The table displays several relevant indications, where the most important ones are found in the second column. Here, it is visible that the variables most correlated with the future earnings is the current earnings, $EBIT_t$, and the growth in current earnings, $Growth_t$. The balance, $Balance_t$, shows some correlation but not to as large an extend, while the age, Age_t, has almost no correlation with future earnings. The negative correlation with the growth variable primarily arises from the volatility in the variable especially given the small absolute sizes of the EBIT,

which can cause large relative changes. Further, the table displays that the growth in EBIT is uncorrelated with the current EBIT and the balance, which could indicate that integrating this in an analysis would add more information and hence improve the forecast. At the same time, the current EBIT and balance variables are positively correlated, which indicates that their developments support each other to a certain degree. However, the magnitude of the correlation estimate indicates that they do not contain the exact same information and are therefore still relevant to combine.

Table 10: Correlation Matrix

	$EBIT_{t+1}$	**$EBIT_t$**	**$Growth_t$**	**$Balance_t$**	**Age_t**
$EBIT_{t+1}$	1.00				
$EBIT_t$	0.60	1.00			
$Growth_t$	-0.27	0.00	1.00		
$Balance_t$	0.10	0.41	0.00	1.00	
Age_t	0.01	0.01	0.00	0.01	1.00

Despite the low correlations of the age variable with current EBIT, growth and balance, it will not be utilized in the multi-variable analyses. This is due to its biases caused by the previously mentioned disadvantages of the FOUND variable. As visible from Figure 3a, the number of new companies in the years of 1999 and 2000 are above 30,000, which is heavily inflated when comparing to Danmarks Statistik (2001) and Danmarks Statistik (2003) reporting values of around 18,000 for both years. In the dataset, this has the effect of 50% of the FOUND variable being located in the years of 1999 to 2003 as visible in Figure 3b. This can cause small sample biases in the remaining years, especially when combined with other variables. Thus, to secure as little bias in the analyses as possible, the age variable is only included by itself, where that analysis will mostly serve indicatory purposes rather than being directly explanatory.

Figure 3a: Number of New Companies

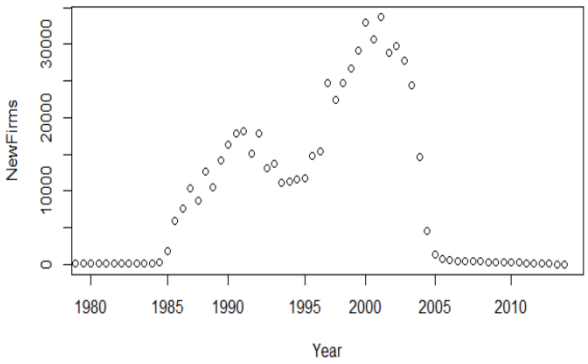
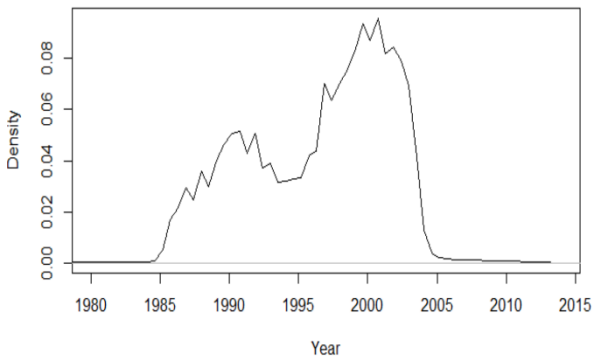


Figure 3b: FOUND Variable Distribution



In addition to the correlation matrix, the current EBIT is considered an essential aspect to estimate the future EBIT. This is apparent from the literature on time series models, where the current EBIT is the point of departure of statistical models and is hence a critical part of the forecast. Given that EBIT is included it is not found relevant to include other similar metrics from the income statement such as EBITDA or contribution margin as these would span a similar dimension and thus not add sufficiently new information. Further, the current growth in EBIT can provide valuable information on the future value of EBIT, which is apparent in a valuation context as well, where the growth rates are important predictors of the development and hence value of the company. Moreover, the balance sheet aspect of the company shows the assets generating the growth and thereby should provide an additional dimension to the company's profile. The balance sheet aspects also become important to consider when determining the value of a company both through the calculation of the cash flows and when deriving the market value of equity from the enterprise value.

Hence, the three variables of absolute EBIT, growth in EBIT and the balance measured by total assets are the main variables considered in the multi-variable analyses. Based on the above reasons, it is assumed that they will provide three different aspects to a company's current performance and future outlook and thus integrate more information in order to provide the most accurate forecasts.

5.2 Results

In this section, the results of the analyses will be discussed along with their robustness to changes in key variables. The section will be split in three, where the first two parts focus on the model superiority between the data driven and the time series forecasts both in terms of the general and specific models. The third part will then focus on the results from comparing the data driven models with the analyst forecasts.

The investigations are analyzed in three dimensions: the data horizon, the forecast horizon and the accuracy measure. The data horizon is divided in *partial*, 1-4 years, and *full*, 5-10 years, while the forecast horizon is split in *short*, 1-2 years, *medium*, 3-4 years and *long*, 5-8 years as illustrated below in Table 11.

Table 11: Data and Forecast Horizons

Years	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Data Horizon	Partial				Full					
Forecast Horizon	Short		Medium		Long					

The accuracy measures will primarily concern MAPE, MSRE and MAE. MAPE is utilized due to its superiority to biases in the other measures as previously mentioned and due to its frequent use in the theory as seen in Table 2, whereby it will be taken as the primary measure of accuracy. It is complemented by MSRE in order to identify the magnitude of the outliers in the results. Here MSRE is chosen over the more frequently utilized MSE, because the broadness of the dataset of the thesis causes large variations in the absolute values, whereby the relative values are more relevant to compare. Further, the MAE measure is included to confirm the MAPE and MSRE robustness as it is the least biased metrics in the first group of accuracy measures.

5.2.1 Time Series Comparison

One of the most important investigation of the thesis is the accuracy of the data driven approach compared to the time series models. Table 12 depicts a summary of this investigation using MAPE, while Figure 4a and 4b graphically illustrates the results. Figure 4a and 4b display the analyses of the MSRE and MAPE metrics on a partial and full data horizon, respectively. In Table 12, the time series models are represented by the AutoArima function from R, as that provides the most superior results and is by definition using the ARIMA model that fits the data best. From the data driven approach, the two most accurate models, *EBIT&Balance* and *EBIT&Growth&Balance*, are included along with the single variable analyses that they are comprised of. The EBIT&Balance analysis means the method, where companies are grouped in portfolios based on both their absolute EBIT and total asset values. The full overview with all variables is displayed in Table 13 of the appendix.

Table 12: Time Series Comparison, Summary, MAPE

Data Horizon	Partial	Full
AutoArima	4.1	3.0
EBIT	11.9	4.4
Balance	50.3	18.3
Growth	45.6	3.9
EBIT&Balance	5.4	2.4
EBIT&Growth&Balance	6.3	2.3

The table only includes the data driven variables using 5 portfolios

From Table 12, it is visible that whenever the data horizon is partial, the time series models have the lowest value and hence best accuracy. This is evident for all forecasting horizons, though the tendency is strongest when the forecasting horizon is long, where all three ARIMA models are superior to all the data driven models. The results of Table 12 can be viewed in Figure 4a, where the lowest MAPE for the

partial data horizon occurs for 3 of the time series models. This result speaks to the necessity of utilizing a vast dataset when generating the data driven estimates as less than 5 years of data is not sufficient to generate superior forecasts. The result is especially noticeable as Bradshaw et al. (2009) state that the ARIMA models require 10 years of data to be optimal. Thus, it can be derived that the data driven approach has an even greater requirement for data than the ARIMA models do.

Figure 4a: MAPE and MSRE on Partial Data Horizon

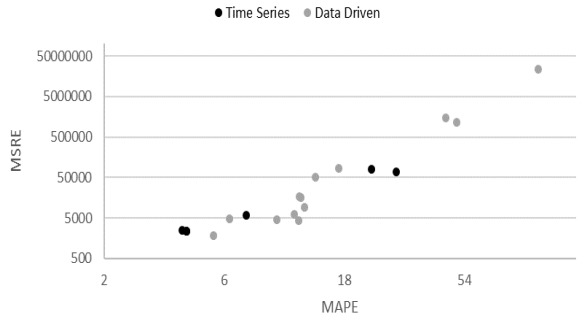
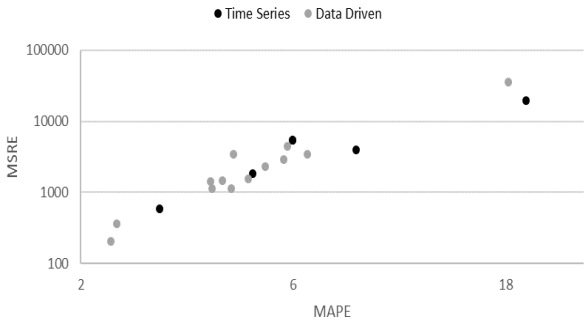
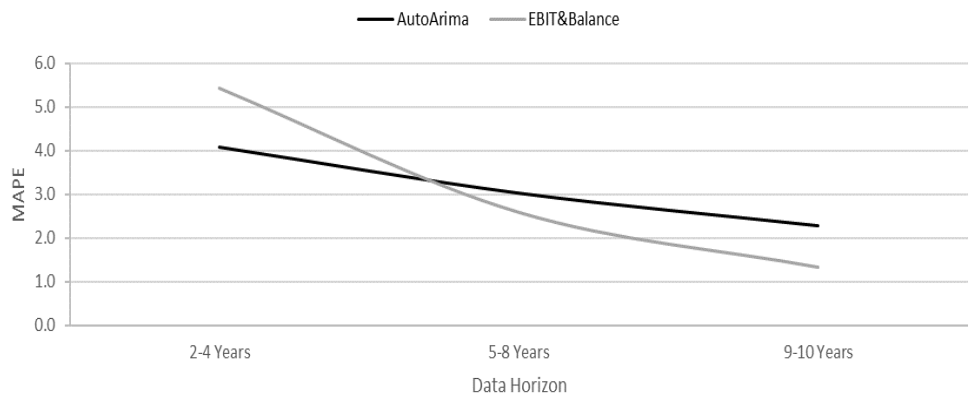


Figure 4b: MAPE and MSRE on Full Data Horizon



However, when the data horizon is 5 years and above, two of the data driven approaches become superior to the time series models. The results can be viewed in Figure 4b, where the lowest MAPE and MSRE for the full data horizon occurs for the data driven models. This is also evident for all forecasts horizons, though especially at the short term where 4 of the 5 most accurate methods are from the data driven approach. This shows that the data driven approach produces more accurate predictions than time series models on all forecast horizons as long as more than 4 years of data is utilized to generate the forecasts. The superiority on the medium and long term forecasts is especially interesting, since the accuracy of the time series models increase relative to the analysts as the forecast horizon increases (Conroy and Harris 1987). Despite this effect, the data driven models remain superior which indicates that they increase in precision at a faster rate than the time series models. This can best be illustrated by Figure 5, where it is visible that the time series models initially have the lowest MAPE, indicating a higher precision, but as the data horizon increases the data driven models increase more in precision than the time series models. From Figure 5 it is visible that the tipping point is at 5 years of data, after which the data driven models exhibit superior accuracy.

Figure 5: The Effect of the Data Horizon on Accuracy



Due to the contradiction of the findings compared to the existing literature, the robustness of the results is an essential aspect to investigate. Therefore, the analysis has been performed on 17 different combinations of data and forecast horizons and on 6 different accuracy measures, where selected results are visible in Table 13 to 15 of the appendix. The amount of combinations performed increases the robustness towards outliers caused by the data and forecasting horizons and ensures that the results are reliable and generated by a sufficient amount of data points. Furthermore, the analyses have been performed for multiple accuracy measures to secure that the results are not affected by biases in these measures. The MSRE and MAE results further confirmed the MAPE findings with the data driven approach being superior when generated from more than 4 years of data. At the same time, both measures find that the data driven approach is also superior on a short and medium forecasting horizon for a data horizon of 1-4 years. Thus, the MAPE analyses are more conservative in their findings than the other accuracy measures. The findings from the MAE and MSRE analyses are very relevant as they confirm that the forecasts are also more precise when comparing the absolute values and in terms of outliers, respectively, which is fundamental support to the argument of superiority.

5.2.2 Specific Model Comparison

In addition to the main finding, the analyses of the thesis have produced several other findings that are worth mentioning. This includes which of the specific models and variables that are most accurate, the impact on the number of portfolios in the data driven model and the impact on the number of variables in the data driven model. A summary of these findings is included below in Table 16, while the full analyses are in the appendix in Table 17 to 19.

From Table 16, it is visible that for the time series models the ARIMAs are most accuracy and that the random walks are the least precise models for the full data horizon. The slight increase in the ARIMA models in the full data horizon occurs from the two other ARIMA models analyzed and not the AutoArima function, as that decreases from a MAPE of 4.1 to 3.0 as shown in Table 12. It is further visible that of the data driven variables, the analyses including the absolute EBIT are most accurate for both data horizons. The analyses including the balance are not nearly as accurate, which is mainly driven by the analysis using only the balance variable has a high MAPE as seen in Table 12.

Table 16: Time Series Comparison, Summary Grouped, MAPE

Data Horizon	Partial	Full
Time Series	12.0	8.0
ARIMAs	4.2	5.0
RWs	19.9	11.0
Data Driven	24.2	5.5
EBITs	10.4	4.3
Balances	19.0	7.2
Growths	32.2	4.3
Industries	13.7	5.5
5 ports	14.7	3.8
10 ports	44.0	5.3
1 variables	36.4	6.8
2 variables	9.2	3.7
3 variables	10.5	4.4

For the number of portfolios in the data driven analysis, it produces more accurate results to use 5 portfolios instead of 10. This could be caused by the analyses increasing in precision as the amount of data increases, hence if each portfolio is based on less data, then the forecast is less accurate. It is further visible that analyses with more than one variable provides better forecasts than the ones with only one variable. This is confirmed in Table 18 of the appendix as the MSRE measure for single-variable analyses is substantially higher than the rest of the analyses, which indicates large outliers. Finally, Table 16 shows that the analyses with 2 variables are slightly more accurate than the ones with 3 variables for both data horizons.

5.2.3 Analyst Comparison

In addition to comparing the data driven approach to the dominant times series models, it is compared to 30 analyst forecasts to provide an indication of the superiority of the models. As touched upon in the methodology section, Bloomberg does not clearly state whether the produced forecasts are 1 year

predictions or have been produced earlier. It is assumed that the forecasts are done on a yearly basis and thus have a forecast horizon of 1 year, though it has not been possible to verify. Hence, for completion the thesis includes 2 comparisons to the analyst forecasts.

The results for the first analysis assuming 1-year forecasts are shown in Table 20 in a summary format, while the full analysis is displayed in Table 21 in the appendix. Here, the forecast horizon is 1 for all the analyses, while the data horizon differs. In Table 20 is included the analyst forecasts along with the 3 superior models from the previous analyses for consistency, which is the AutoArima, the EBIT&Balance and EBIT&Growth5&Balance. Further, the most precise time series model in this analysis, random walk with no drift, and most precise data driven model in this analysis, EBIT growth in 10 portfolios, are included as well. It is visible that for all the 4 data horizons, the data driven models are more accurate than the analyst forecasts. Especially in the 6 and 7-year data horizon, where the forecasts are up to 12 times as accurate. As visible in Table 21, the superiority applies to up to 11 out of the 13 data driven models for the two data horizon. The number decreases for the data horizons of 8 and 9 years, but still maintains a value of above 50% of the data driven models proving superior across the 4 data horizons. This is a major result despite having an indicative nature and it validates the data driven approach's superiority significantly. Further, the absolute values of the MAPE for the data driven models have decreased significantly compared to the previous analyses, which could indicate they perform better under more stable company settings.

Table 20: Analyst Forecast Comparison, Fixed Forecast Horizon

	MAPE			
Data Horizon	*6*	*7*	*8*	*9*
Forecast Horizon	*1*	*1*	*1*	*1*
Analysts	1.2	1.3	1.2	1.4
AutoArima	1.6	1.4	1.3	1.6
rwfNoDrift	0.6	1.4	1.7	1.5
EBIT&Balance	0.8	0.9	0.9	0.9
EBIT&Growth5&Balance	0.2	0.4	0.5	0.6
Growth10	0.1	0.3	0.5	0.7

In addition, it is visible from Table 20 that the time series models only produce more accurate forecasts on 1 data horizon compared to the analyst, even though the accuracies are relatively close. As the data driven methods are superior to the time series forecasts on all horizons, it further validates the previous findings of the data driven approach being superior to the time series models. Table 21 proves an even

stronger superiority as between 8 and 10 data driven models are superior to all time series models for all data horizons.

An interesting aspect in this analysis is the superiority of the growth variables with 10 portfolios in the data horizons of 6 and 7. This is especially worth noting as it is one of the least accurate variables when considering the full dataset as seen in Table 13. However, when it is solely listed companies included the growth variable is a good predictor. This could indicate that the growth in the listed companies is more stable and predictable than the full dataset including a more varied group of companies. The accuracy of the growth variable also causes the EBIT&Growth model to surpass the EBIT&Balance model in superiority. The earnings stability is confirmed by the random walk with no drift being superior to the ARIMA models in 2 out of 4 data horizons. Thus, it appears that the listed firms are better forecasted by more stable models or models that are able to capture the stability in growth.
Contradicting to the previous analysis, the growth variable shows worse results for the longer data horizons, which affects all the data driven models including the variable. This is not considered a general model result, as it is mostly attributed to the outliers in the dataset of the 30 listed companies.

The analysis assuming that the forecasts in Bloomberg are made with a forecast horizon of more than 1 year is shown in summary in Table 22 and in full as part of Table 21 of the appendix. Thus, in the below it is assumed that the forecasts for 2010 are made on 1-Jan-2007, which is deemed unlikely but included for testing the robustness of the results. Hence, the data driven forecasts are also updated to have a data horizon of 6 years and then a varying forecast horizon. Generally, the same conclusions hold true as the data driven models outperform the analyst and time series forecasts. The data driven models are slightly less superior in this setting, as the MAPE values get closer to the values of the analysts. However, still around 50% of the data driven models are superior for all 4 forecast horizons, which indicates very robust results as the data driven model is generally worsened when the forecast horizon increases.

Table 22: Analyst Forecast Comparison, Fixed Data Horizon

	MAPE			
Data Horizon	*6*	*6*	*6*	*6*
Forecast Horizon	*1*	*2*	*3*	*4*
Analysts	1.2	1.3	1.2	1.4
AutoArima	1.6	2.5	3.9	3.3
rwfNoDrift	0.6	1.1	1.6	1.7
EBIT&Balance	0.8	0.9	1.0	1.0
EBIT&Growth5&Balance	0.2	0.9	1.1	1.3
Growth10	0.1	0.8	1.1	1.3

However, if the analyst forecasts are actually performed on a yearly basis, the results of Table 22 are even more significant. Then it would mean that 6 of the data driven models produce better forecasts 4 years into the future than the analysts can produce looking 1 year into the future. So even when the analysts are 3 years ahead of the data driven forecasts and know the current value in 3 years, they are still not able to beat the data driven forecasts.

However, there is one important comment to be made concerning the analyst forecasts. The accuracy measure is to some extent driven by 6 outliers in the dataset, where the analyst accuracy is significantly worsened by MAPE values of 5 to 7. Thus, if the median values are utilized instead, the results are quite different. The analysts have a median value of 0.2 for the 4 horizons, which is slightly lower than all the data driven models. Nevertheless, the thesis has throughout the entire analysis section utilized simple averages, whereby the most comparable and relevant results are the ones displayed in the tables above.

Due to the outliers in the analyst forecasts, the results of the comparison using the MSRE measure is even more in favor of the data driven approach as visible in Table 23 in the appendix. Here, the average MSRE for all horizons is 11.6 for the analysts, which is inferior to 11 out of the 13 data driven models. At the same time, 8 of the 13 data driven models have an average of less than 4.5, which is significantly less than the analyst forecasts. This confirms the results outlined above to an extensive degree.

5.3 Analysis Conclusion

The section analyzes the accuracy of the data driven approach compared to the dominant models in the earnings forecasting literature and makes several findings. One of the primary findings is that the data driven model is more accurate than the time series models for all forecast horizons when the data horizon is above 4 years. The data driven models that achieve the superiority are comprised of the variables of absolute EBIT, growth in EBIT and the balance, whereby these variables are deemed to sufficiently span the dimensions of what makes a company. Further, the data driven approach proves more accurate than the small sample of analyst forecasts for all data and forecast horizons. This is especially noteworthy as it applies to around half of the data driven models, whereby it is an even more significant superiority than compared to the time series models.

In addition, the analyses find that having fewer portfolios increases the accuracy for the general dataset due to the additional data available for each portfolio. Though for the listed firms with more stable earnings, the growth variable with 10 portfolios is superior. Finally, multi-variable analyses perform

better than single-variable analyses, which can be explained by the additional information incorporated in the forecasts.

6. Discussion

6.1 Result Deliberation

The result deliberation section will critically evaluate the results attained and compare towards the outlined research questions. This will enable a more in-depth discussion on the findings and the underlying reasoning hereof. More specifically, this section will focus on four elements where the former addresses the first research question and the latter three address the second research question. First, the size of the accuracy measures will be held towards what the literature attains to evaluate the absolute interpretation of the results. Second, it will discuss the superiority of the time series models compared to the data driven models. Third, the specific models both within the time series and data driven frameworks will be discussed along with the general composition of the data driven models. Finally, the superiority of the data driven models compared to the analyst forecasts will be elaborated upon.

6.1.1 Accuracy of the Data Driven Models

This section will seek to answer the first research question relating to the absolute accuracy of the data driven forecasts. In the main analysis of Table 13, the lowest of the MAPE measures are around a value of 2 with 1.9 being the minimum. The interpretation of this number indicates that the best forecast produced is around 200% higher or lower than the actual value, i.e. if the actual value is 1,000 then the forecasted value would be 3,000 or -1,000. In many applications of these forecast such as for valuation purposes, a potential difference of 300% in earnings and hence the value of a company is very substantial and potentially too substantial to be useful. Similar results reflect in Table 13, where the lower MAE measures are between 400 and 500 with a minimum of 413 thousand DKK. If this is assumed to be the average error in forecasting the earnings of a company, then the resulting valuation can differ with several million DKK depending on the other variables in the valuation. This difference is in itself difficult to determine if it makes the forecast irrelevant, since this absolute deviation for larger companies might be acceptable. However, when it is combined with a MAPE of 2.2 it is a significant deviation, as it indicates that the absolute value of the earnings is either 129 or 1,322 thousand DKK.
In the additional analysis comparing to the analyst forecasts, the MAPE measures are significantly lower with a minimum value of 0.13 achieved by the growth in EBIT with 10 portfolios. Further, as seen in

Table 21 in the appendix, there are numerous MAPE measures below 1, which serves as an improvement to the main analysis.

In determining the effect of the accuracy measures' magnitude, it is necessary to compare with the previous literature in order to attain an insight into the norm values of the accuracy measures. Albrecht, Lookabill, and McKeown (1977) use both MAPE and MSRE in their article to determine the accuracy of similar ARIMA and random walk models as seen in Table 1. They find MAPE results with a minimum of 0.01 and maximum of 0.8, along with MSRE results ranging between 0.07 and 4.1. In addition, Conroy and Harris (1987) find MAPE results ranging between 0.24 and 0.47 for similar ARIMA and random walk models. Thus, both findings are significantly lower than the main results of the thesis. However, this might be explained by the data horizon in the articles of 25 years and 20 years, respectively. If a best-fit trendline is added to Figure 5, then it would for the EBIT&Balance curve predict a MAPE accuracy of 0.08 and 0.04 for the 20 and 25-year data horizon, respectively. These results would be very comparable to those of Albrecht, Lookabill, and McKeown (1977) and serve as an improvement compared to Conroy and Harris (1987). In addition, as seen in the comparison to the analyst forecast the MAPE and MSRE values for listed firms are significantly lower and very similar to the findings of the literature. This is an important aspect, since the literature solely uses listed firms, so this comparison is more relevant. Moreover, it is relevant to compare the MAPE results of the alternative approach of the thesis to the results achieved in the alternative methods the literature proposes. Reverte and Guzman (2010) attain MAPE results of between 1.31 to 2.87 along with Lobo (1991) getting values around 0.6, which is more comparable with the results of the main analysis.

In summary, the sizes of the accuracy measures are not determined to be outside an acceptable range, since the data driven models produce low MAPE and MSRE results for the listed firms. Thus, the non-listed firms are indicated to be more difficult to forecast than listed firms. Nevertheless, the data driven forecasting method is concluded to be accurate in absolute terms. Moreover, the second aim of the thesis is to provide a comparison between the models, whereby the accuracy measures are in the following treated in a relative manner.

6.1.2 Superiority Compared to Time Series Models

In the theory section, 3 conditions are outlined for which the time series models would perform optimally compared to analysts. These include a positive impact on the relative superiority from an increasing forecasting horizon, decreasing size of the firms and increasing earnings variability. The latter two are fulfilled for all the analyses as the dataset includes a large set of firms where 99.9% of the

companies are not listed and the majority of the companies do not exhibit stable earnings. At the same time, several analyses are performed with a long forecasting horizon to test the first condition. Thereby, all the 3 conditions are met, which should result in superior results for the time series models and in particular the ARIMA models. This expectation is confirmed to some degree as visible in Table 24, which shows the effect of the forecast horizon on accuracy when including all data horizons. Here, the ARIMA model gets relatively better when the forecasting horizon increases compared to the data driven models. When moving from the short to the long horizon, the ARIMA model decreases by 48% in accuracy, while the two most accurate data driven models decrease with 70% and 82%.

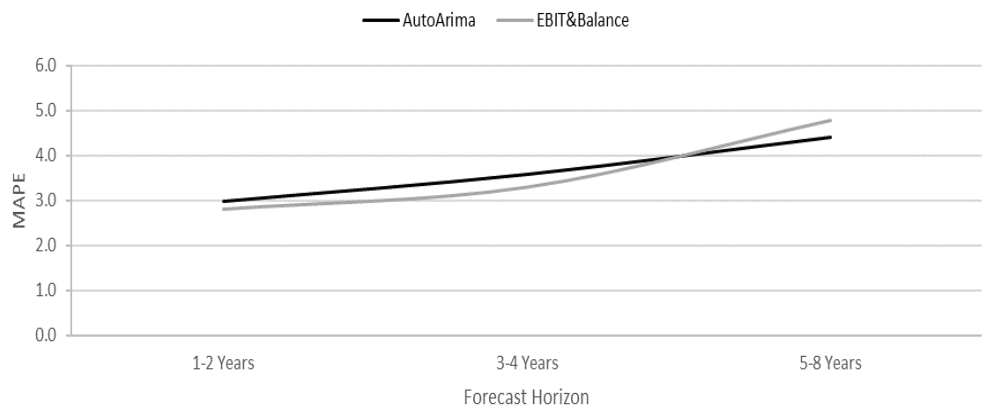
Table 24: The Effect of the Forecasting Horizon on Accuracy

Forecast Horizon	Short	Medium	Long
AutoArima	3.0	3.6	4.4
EBIT	5.7	8.2	10.4
Balance	14.8	18.5	34.4
Growth	3.6	10.1	33.3
EBIT&Balance	2.8	3.3	4.8
EBIT&Growth&Balance	2.7	3.6	5.0

However, the above results are mainly determined by the loss of accuracy of the data driven approach in the partial data horizon. As seen in Table 13 of the appendix, when more than 4 years of data are utilized to generate the forecasts, the data driven approach is superior on all forecasting horizons.

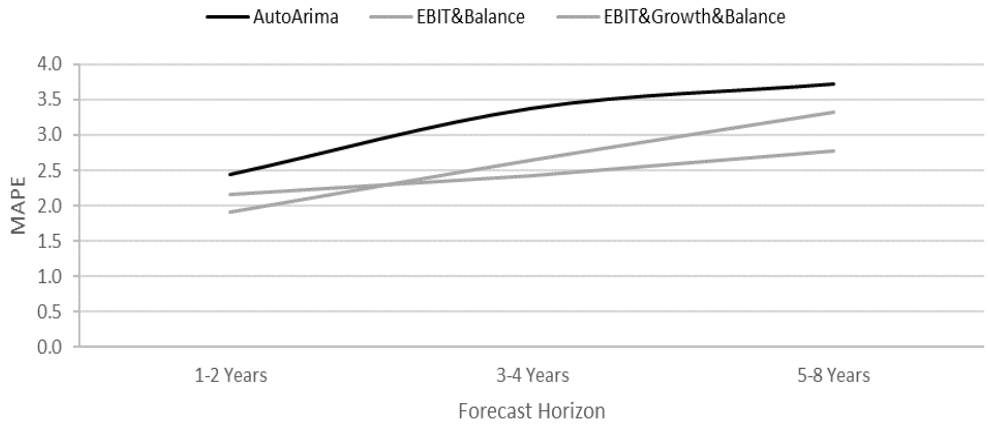
This leads to three main conclusions about the two types of models. Firstly, the data driven model exhibits similar characteristics as the ARIMA model in terms of horizons given that it increases in accuracy for an increase in the data horizon, while it decreases in accuracy for an increase in the forecasting horizon. However, what is noticeable here is that it does so even more exponentially than the ARIMA model. The increase in precision when including more data is shown in Figure 5 in the analysis section, while the decrease in precision from a longer forecasting horizon is shown below in Figure 6a.

Figure 6a: Forecasting Horizons with All Data Horizons



In both figures, it is visible that the models exhibit the same tendencies though with the data driven approach having a steeper slope. This can be explained by the integration of data points in the models. When a new year is added in the ARIMA model, it attains one additional data point to integrate in its forecast. However, when a new year is added in the data driven model, it attains as many data points as there are comparable companies, which can be several thousand new data points. Hence, the effect of the additional data is more extensive, which corresponds to the resulting accuracy in Figure 5. For the forecasting horizon, the effect on the two types of models is similar, where the slightly steeper slope is caused by the data driven models' insufficient data on the partial data horizon. Thus, it is important to note that the ARIMA superiority on forecast horizons of 5-8 years from Figure 6a is only apparent when all data horizons are included, i.e. both the results from the partial and full data horizons are included. Figure 6b shows the result using only the full data horizon where the data driven models remain more accurate for all forecasting horizons.

Figure 6b: Forecasting Horizons with Full Data Horizon



The second major conclusion is that the data driven models embody a contemporaneous advantage compared to the time series models. As outlined in the theory, the integration of information on comparable companies is a key component for analysts to gain an advantage of the time series models on the day of forecasting. This analyst advantage is decreased when comparing to the data driven model as it is built on information from comparable firms. The effect of the advantage is mostly visible in the short term as that is where the analyst incorporating the information is most accurate as seen in the theory section. Thus, the above Table 24 proves that this advantage is present as both multi-variable analyses are more accurate in the short and medium forecast horizon compared to the ARIMA model.

The third finding is that the ARIMA model is more robust towards smaller amounts of data, which contrasts the general view presented in the literature when comparing to analyst forecasts. This finding is closely connected to the first finding but constitutes a separate point of interest as it indicates that there is a lower bound for which the data driven approach cannot provide accurate forecasts. The ARIMA model is not limited in the same degree and can thus be utilized more appropriately than the data driven models for shorter data horizons. This is illustrated in Table 13 as the ARIMA model still performs decently when the data horizon is partial.

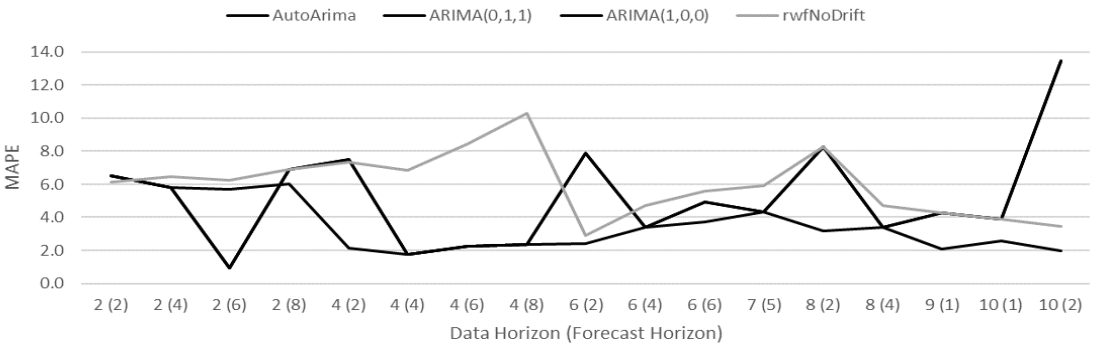
6.1.3 Superiority of Specific Models

In discussing the results of the specific time series and data driven models, the focus will first be on the time series models and then turn towards the variables of the data driven models.

In Table 13, it is visible that for almost all data and forecast horizons the ARIMA models are superior to the most accurate random walk model, the random walk with no drift. In fact, for all the 17 different

tests combining the data and forecast horizons, the random walk model is only more accurate than the ARIMA models for 1 combination as visible in Figure 7.

Figure 7: Superiority Amongst Time Series Models



In the test, where the data and forecast horizon are 2 years the random walk with no drift is 6% more accurate than the ARIMA models. This is surprising when compared to the literature stating that ARIMA models require up to 10 years of data (Bradshaw et al. 2009) and the property of the random walk models that they only require 1 year of data when they do not include a drift term. This would indicate that even for the partial data horizons, the increased data included in the ARIMA model is sufficient for it to have an advantage over the random walk. The advantage is seen to be very small in the 2-year data horizon in Figure 7, where the random walk model has similar accuracy to the ARIMA models in 3 out of 4 forecasting horizons.

Moreover, from Table 13 it is visible that amongst the random walk models the most accurate one is without a drift, while the Brownian motion is the least accurate. This shows that despite the volatile earnings in the dataset, the Brownian motion predicts too significant changes. In addition, the Brownian motion has the largest measures of MSRE of the time series models in general, as seen in Table 14. This could be caused by the instances where the model predicts a high volatility in one direction, while the earnings exhibit a volatility in the opposite direction, which would result is a significant outlier. The Brownian motion is more prone to this type of error than the random walk with no drift, which produces more smoothened forecasts.

In terms of the data driven models, the results show several relevant aspects that require additional discussion. Firstly, from Table 13 it is clear that the most accurate single variable overall is the absolute EBIT. This result is not surprising when looking at the correlations in Table 10, since the absolute EBIT has the highest correlation with the future EBIT. Despite the result being expected, it is a crucial finding

in validating the data driven model, since it heavily relies on the latest earning as outlined in the methodology section. Thus, as the latest earning is the best single-variable to predict the future earnings, it seems appropriate to place weight on it when performing the forecasts.

Secondly, the growth in the EBIT variable yields noticeable results. As seen from Table 12, the growth variable is the most accurate single variable when the data horizon is full, while it is very inaccurate for the partial horizon. This is especially evident when it is divided in 10 portfolios, where it becomes the least accurate variable. The result proves that the growth for companies in the full dataset is relatively dispersed, whereby 1 to 4 years of growth data is not sufficient to generate reliable results. However, as visible in Table 13 when the absolute EBIT is combined with the growth in EBIT the forecasts are improved both for a partial and full data horizon. Thus, the growth variable contains additional information that is not found in the absolute EBIT. This is also confirmed from the correlations in Table 10, where the growth in EBIT is completely uncorrelated to the current EBIT, while still having some correlation with the future EBIT. Nevertheless, in the analyses comparing the data driven models to the analysts, the growth variable with 10 portfolios performs very well. This finding confirms that accuracy of the growth variable forecasts is positively correlated with the stability in earnings.

Thirdly, the balance variable shows slightly contradictory results. By itself, the variable has the second least accurate forecasts in the time series comparison, only surpassed by the growth variable divided in 10 portfolios. However, as with the growth variable it significantly improves the forecasts of EBIT in the multi-variable analyses to such an extent that both of the most precise data driven models include the balance. This finding contradicts the correlation explanation as the balance variable is only slightly correlated with the future EBIT and relatively correlated with the current EBIT. If the correlation notion is the only determinant, the balance would not be able to explain the future EBIT and most of the information stored in the balance values would already be integrated in the current EBIT values. However, as it manages to improve the EBIT forecasts with around 100% it is concluded to add relevant information to the forecast, which indicates that there are other explanations than the variables' correlations determining their contributions in multi-variable analyses. Thus, additional explanations to the information added is suggested as a point of further research.

Fourthly, the industry variable provides relevant insights as well. Similar to the growth and balance variables, the industry variable by itself provides less accurate forecasts when compared to the absolute EBIT. However, when combined with the EBIT variable the multi-variable forecast is less accurate than the EBIT forecast for all combinations of data and forecast horizons, which contrasts that of the above multi-variable analyses. A similar result is seen in the multi-variable setting, where the industry variable

is added to the absolute EBIT and growth in EBIT forecast. Here, the addition of the industry variable also makes the forecasts less accurate for all combinations of data and forecast horizons. These results prove that the industry does not contain additional information that is not already stored in the EBIT and growth variables.

Another important aspect of the data driven models is their general composition. The first aspect of the composition is the number of portfolios in the analysis, where Table 16 shows that the analyses with 5 portfolios give better results compared to using 10 portfolios as more data is included. This is deemed the general result for a dataset including a broad range of companies, while smaller deviations may occur in other datasets such as seen in the analysis only including 30 listed firms.
The second aspect concerns the number of variables to include in the analyses. Table 16 shows that analyses with 2 variables perform better than those with 3 variables which goes against the logic of adding more information by adding more variables. However, in this context it is important to investigate the underlying analyses that generate Table 16, as the grouping of 3 variables contains the test using 10 portfolios for the growth variable. This skews the results as there are no tests with 10 portfolios in the grouping of 2 variables. Thus, Table 25 is created to show the effect of taking that analysis out of the 3-variable grouping.

Table 25: Multi-Variable Analyses

Data Horizon	Partial	Full
1 variables	36.4	6.8
2 variables	9.2	3.7
3 variables	10.5	4.4
3 variables wo. 10 port.	8.9	3.8

The table shows that the new 3-variable grouping is very similar to the 2-variable grouping and serves as a slight improvement on the partial data horizon. Hence, it can be concluded that including a third variable might improve the forecast slightly or at least provide similar accuracy. At the same time, the table could indicate that there is a limit to the information brought in, whereby bringing in a fourth or fifth variable might not improve the forecast to a significant degree. This effect will also be a relevant investigation of further research.

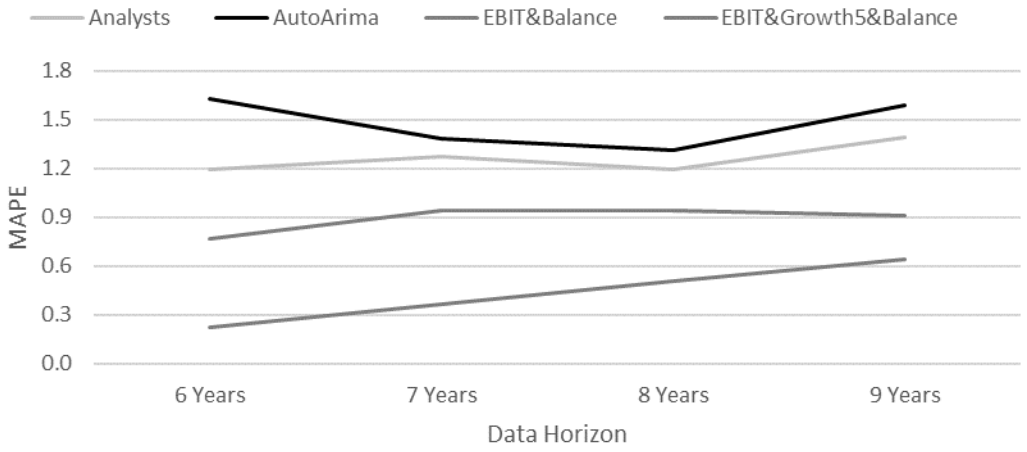
6.1.4 Superiority Compared to Analyst Models

When discussing the results from the analyst models compared to the data driven models, it is important to recall the conditions from the theory section that the analysts would perform relatively

better under. The 4 conditions are short forecast horizon, forecasting based on quarterly earnings, short time until announcement and high prior precision of the analysts. These have been accommodated to some degree as the forecast horizon and time until announcement is down to 1 year. However, the dataset only utilizes yearly data and prior precision has not been able to be integrated in the analyses. As discussed above, the data driven models exhibit many of the same characteristics as the time series models, whereby it is also relevant to include the 3 conditions for which the time series models are superior to the analyst forecasts. They are outlined above but boil down to longer time horizon, smaller firms and unstable earnings. These 3 are particularly interesting since the sample of 30 firms and corresponding forecasts exhibited short time horizons, large firms and stable earnings. Thus, despite only 2 out of 4 conditions being met for the analyst to be superior, all 3 conditions are met for the time series models to be inferior. Therefore, there are 2 conclusions that are especially relevant to highlight. In the below, the analysts are assumed to be producing 1-year forecasts as that is the most likely scenario.

First, the forecasting horizon is interesting to consider. As illustrated in Figure 8, the data driven forecasts perform significantly better than the analyst forecasts for the short forecast horizon of 1 year for all data horizons.

Figure 8: Superiority on the 1-Year Forecast Horizon

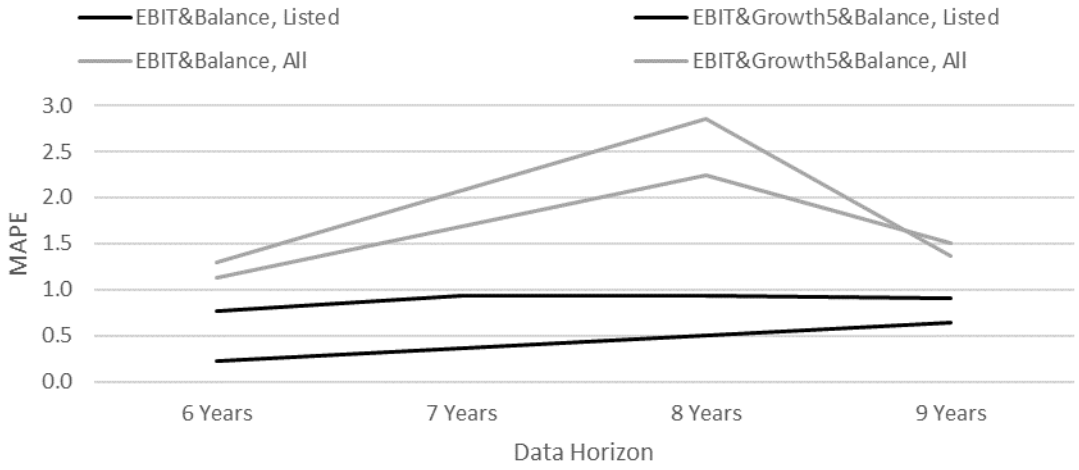


The figure shows that the time series forecasts are less accurate for the short forecast horizon of 1 year, which is as expected from the conditions outlined in the theory. However, the figure shows that the data driven forecasts do not suffer from the same disadvantage as the time series models when comparing to the analyst forecasts. This is a very interesting result as the short-term horizon is a key

advantage for the analysts over the time series forecast that they now do not possess compared to the data driven approach.

Secondly, the results above are generated using solely large, listed firms with stable earnings, which should according to the theory provide the analysts with an advantage. In addition, the forecasts from Bloomberg used up to 20 analysts in combination, which should further enhance the accuracy. This effect is seen to be sufficient for the analysts to beat the time series forecasts as expected. However, the data driven forecasting utilizes the stability of earnings to a larger degree and is even more precise for this segment than when forecasting for all types of firms in the dataset as illustrated in Figure 9.

Figure 9: The Effect of Listed Firms on Accuracy



The figure shows that the data driven forecasting is consistently more accurate when large and listed firms with stable earnings are forecasted rather than all types of firms of the dataset, where only 0.01% are listed. Thus, the analyst forecasts do not possess an advantage over the data driven approach for more stable earnings and listed firms.

However, these results can only serve as indicators of the relation between the data driven and analyst forecasting methodologies, since the sample of 30 firms is too small for more robust conclusions to be drawn. Nevertheless, the indications are very relevant and contrast the expectations from previous literature, whereby further research is encouraged to perform the same analyses on a larger data sample to verify the conclusions.

6.2 Contributions

Based on the above discussion, the thesis is able to provide two main contributions to the existing literature on the subject of earnings forecasting. The primary contribution is the notion of the data driven approach and its superiority over the time series models and analyst forecasts. The secondary contribution is to the discussion on the superiority between the ARIMA and random walk models. In this context, the thesis is able to contribute to the discussion with its finding of superiority between the two types of models.

6.2.1 Data Driven Forecasting

The main contribution of the thesis is the presentation of the data driven approach to earnings forecasting. The approach constitutes a new way of utilizing financial information on companies to produce accurate forecasts of a company's earnings. The data driven approach utilizes financial metrics from comparable companies and makes the underlying assumption that similar companies develop their earnings through similar patterns. Thus, it clearly differs from the currently accepted and utilized models, which mainly includes time series models and analyst forecasts. At the same time, it embodies several key advantages over both types of models in order to integrate the most essential information into the forecast.

Compared to the time series models it proves a more accurate way to estimate future earnings as long as more than 4 years of data are utilized to generate the model. The data driven models are up to 25.3% more accurate than the time series models and the accuracy is greater for all forecasting horizons between 1 and 8 years. Further, the most accurate data driven model only loses 28.6% in accuracy when going from the short to the long forecasting horizon compared to the best time series model losing 52.5%. However, the time series models are more accurate for less than 5 years of data, which pertains to all forecasting horizons. Nevertheless, as visible in Table 3 most researchers and forecasters have more than 4 years of data at their disposal, whereby the data driven model is deemed the superior manner for forecasting earnings.

Furthermore, the data driven models prove up to 8 times as accurate as the analyst forecasts for forecast horizons of 1 year, which should be the horizon where analysts are strongest. At the same time, the data driven method is most likely able to make 4 year forecasts that are up to 36% more accurate than the 1 year forecasts made by analysts.

As the results contradict the previous literature significantly, they have been thoroughly tested to ensure a well-documented contribution. Thus, 19 different time series and data driven methodologies

have been included and analyzed through 6 different measures of accuracy. These methodologies and accuracy measures have been tested by 17 different combinations of data and forecasting horizons. Hence, with a total of 1938 tests the results are treated as having a high robustness.

For the general applicability of the results, it is important to note the difference of the dataset compared to what is generally used in the literature. The data of the thesis includes a broad range of both listed and non-listed Danish companies, which with its 352,496 firms constitutes a 252 times larger dataset than the average of the literature. Especially the size of the dataset is expected to benefit the data driven approach as it is able to utilize a much larger part of the data in its forecasts compared to the time series models. In addition, the analysis comparing to the analyst forecasts indicated that the approach can also be utilized in forecasting earnings for listed companies as well.

6.2.2 Time Series Models

The secondary contribution of the thesis is to the discussion on the superiority amongst the time series models, since the thesis tests 6 different time series models including 3 ARIMA models and 3 random walk models. They are tested through multiple accuracy measures and horizons, whereby the thesis is able to contribute to the discussion on their superiority. For the full dataset, it finds that the 3 random walk models are inferior to one or more of the ARIMA models for 94% of all data and forecasting horizons. This clearly proves that the ARIMA model is the superior time series model for the broad dataset. However, the random walk with no drift proves more accurate when forecasting earnings for listed companies, especially when measuring on the generated outliers.

For the ARIMA models, the thesis finds the AutoArima function to be most precise for the majority of data and forecasting horizons, which contrasts the literature as it mostly uses the Griffin-Watts model of ARIMA(0,1,1) and Brown-Rozeff model of ARIMA(1,0,0). This is due to the AutoArima function automatically detecting the most fitting ARIMA model, whereby it will only choose the above two if they explain the data most accurately. Hence, the thesis encourages earnings researchers to integrate this forecasting method in further papers using ARIMA models.

Amongst the random walk models, the thesis can contribute with its finding of the most accurate forecasting method being the simple random walk model without a drift. This model is superior for all data and forecasting horizons to the random walk model with a drift and the model generated by a Brownian motion. Especially the Brownian motion forecasts are found very inaccurate for this type of data and cannot be recommended for use by earnings researchers. Thus, for both listed and non-listed

firms, the thesis suggests to utilize the random walk without a drift if random walk models are to be used.

6.3 Implications

The findings of the thesis have several implications for the research community and financial statement users, which can be categorized into two main groups. The first one concerns the data driven models' superiority over time series models and what that superiority means for the relevant stakeholders. The second one builds on top of the first one to compare the data driven models to the analyst models in order to determine the superiority and its implications.

6.3.1 Data Driven Superiority over Time Series Models

One of the primary finding of the thesis is the superiority of the data driven method over the time series models. This is a key finding as it introduces a new branch of models to be used in the literature and by stakeholders interested in forecasting company earnings. As this finding contrasts the previous literature, it has several implications for the earnings forecasting community that are necessary to elaborate upon.

First of all, as touched upon in the methodology the core implication of the findings is that earnings from comparable companies is a better predictor of future earnings than past earnings of the company. This proves that the information stored in the financial parameters of other companies who have been through a similar phase in their company life cycle exceeds the information from the past earnings.

Secondly, the findings imply that the research community previously using time series models to predict earnings will be better off by using the data driven approach instead. This will increase the forecasting accuracy and provide a more precise picture of the expected outlook of a company. At the same time, the data driven models allow for a broader range of tests as researchers can integrate any measurable variable in the forecasts.

Thirdly, the results are relevant for financial statement users who utilize the time series models either instead of analyst forecast or in combination with them. They are now able to use a more precise measure to the forecast the earnings and validate the analyst accuracy.

6.3.2 Data Driven Superiority over Analyst Forecasts

The end goal of introducing the data driven approach is to test if it is a superior forecasting methodology compared to all the dominant models. The main analysis of the thesis is comparing the data driven

approach with the time series models as discussed above, while the secondary analysis is comparing with analyst forecasts. From the latter, it is apparent that the data driven approach is superior to the analyst forecasts for all data and forecast horizons. However, since the analysis used a small sample size, it can only contribute with indicatory results as robust conclusions cannot be directly drawn from it. Therefore, the thesis also uses a more indirect way of comparing the data driven approach with the analysts. In order to make the indirect comparison, it is necessary to refer to the findings of the theory section. There it concluded that in the setting where the dataset includes small firms with no analysts following and where the forecasts are on yearly data for a forecast horizon of more than 1 year, the time series models are superior to the analyst forecasts. As all of these conditions apply to the data of the thesis, the time series models are expected to be superior to the analysts' forecasts, if they had been included in the main analysis. Thus, as the data driven models are superior to the time series models, they are indirectly determined to be superior to the analyst forecasts for this type of data as well. This finding carries multiple implications as will be outlined in the following paragraphs.

First, this is a very important finding for non-listed firms. It is especially interesting for these firms as the potential for improving the forecasts is the greatest and since the number of firms in this category is far larger than the category of listed firms. The finding of the thesis makes it possible for researchers and other stakeholders to produce accurate forecasts for these companies despite lack of analyst projections. This enables an immense number of companies to be generally available for further research and analyses, which can improve the research results as the amount of data increases. Thus, broader and more generally applicable research can be performed.
The finding further enables a considerably easier valuation process of non-listed firms as potential investors only need a few key variables of the company in order to forecast their expected earnings and apply a discounted cash flow model to attain the value. Thus, the findings are also relevant for the venture capital community along with other pre-IPO investors.

Second, the findings also have important implications for listed firms. Here, the data driven models can be utilized to increase the coverage on the companies and to assess the accuracy of the analyst forecasts made, which is important information for several stakeholders. For traders and investors, the new approach can contribute with a way to determine if the stocks are correctly priced by using the earnings forecasts to derive the value of the company. For current analysts, it provides a way of determining the accuracy of their own forecasts and suggestions for improving these. For potential new

analysts, they can determine if the stock is properly covered or if they can benefit sufficiently from providing coverage for it.

Thirdly, the data driven approach is costless to utilize once the main dataset is attained given that some basic information such as current earnings and total assets is available for the companies to be forecasted. Thus, there are large cost saving opportunities present for researchers and earnings forecast users, who would otherwise have purchased access to analyst forecasts. At the same time, this serves as an easier and more accurate way for managers to perform earnings forecasts for the budgeting and resource allocation processes.

7. Limitations

The findings of the thesis have some limitations which should be paid attention towards in replicating analyses and hence require a more elaborate discussion. The limitations can be grouped into 4 main categories. The first is concerning the country specificity of the dataset to only include Danish companies. The second is the dataset's composition in terms of listed and non-listed companies and how that can have a limiting effect on the applicability of the findings. The third is the general requirement for a vast amount of data in order to generate the data driven models. Finally, the limitation of how to identify comparable firms will be touched upon.

One of the main limitations of the dataset is its sole inclusion of Danish companies. This can potentially serve as a challenge in producing a generally applicable framework, since the findings can only be tested for the market in Denmark. The effect of the limitation is that the model might not apply to other markets since companies in other countries could have different development patterns caused by regional contexts. There could also exist a different correlation between the listed and non-listed firms, which would further decrease the applicability of the findings. To circumvent this limitation, it is necessary to produce the same analyses with international datasets containing multiple countries. Thus, further research is encouraged to test if the same results are applicable in other countries.

One of the main benefits of the dataset is its general applicability given the broad range of company profiles included. Though, if the model is to be used for large listed companies, the inclusion of 319 listed firms might not be sufficient to generate accurate results. This limitation arise as the Danish stock market is very limited compared to other countries such as USA and UK, whereby more listed companies could not be integrated. The effect of the limitation is that listed companies could exhibit different earnings developments and variability, which cannot be predicted with the current dataset. Thus, to

meet this limitation more listed companies might need to be included in order for the data driven models to accurately predict their earnings. This could be a natural extension once the dataset is also expanded geographically as outlined above.

As touched upon earlier, the data driven model has a noticeable limitation in its requirement for data to be able to generate accurate forecasts. This is visible in the less accurate forecasts for variables split in 10 portfolios rather than 5 caused by less data being present in each portfolio. Further, it is confirmed by the less accurate results of the data driven approach compared to the time series models, when a data horizon of less than 5 years is utilized. Thus, to solve this limitation the dataset needs to be relatively extensive, which causes replicating analyses to be more complicated. In this sense, the Danish system is very beneficial as most of the required information is publicly stored in the official company register, Det Centrale Virksomhedsregister (CVR). Hence, attaining the same information for other countries to satisfy the first two limitations might be challenging.

The final limitation is how to identify the best approach of finding comparable companies and more specifically how to identify the variables that make them comparable. As described earlier, the correlation between variables and future earnings is able to provide some identification of relevant variables. However, it does not provide the full picture as shown by the information contributions by the balance variable despite its low correlation with the future EBIT and high correlation with the current EBIT. The effect of this limitation can be an incorrect grouping of companies, whereby a company risks being grouped in a portfolio with non-comparable companies, which would decrease the forecasting accuracy of the model. Thereby, further research can enhance this area of the data driven methodology by investigating additional ways of identifying comparability.

8. Conclusion

This thesis seeks to introduce a new methodology to the topic of earnings forecasting that is superior to the existing models. Besides being superior in accuracy, the model should have a broader applicability to properly apply to the less covered areas of earnings forecasting such as non-listed companies. The way the thesis proposes to meet this challenge is through the data driven model for earnings forecasting. Firstly, the model is concluded to be accurate in absolute terms. This is determined by investigating the absolute sizes of the accuracy measures, which lie within an acceptable range of findings from earlier research. Secondly, it is compared to the dominant models of the time series and analyst forecasts to identify if the model can contribute to the topic of earnings forecasting. In the literature section, it is

concluded that the two dominant models perform best under two different sets of conditions. Thus, the thesis creates two main parts of the analysis where each is tailored to meet these conditions to ensure the data driven model is compared to the best version of the dominant methods. From the analyses performed, it is apparent that the data driven model is more accurate than the time series models as long as more than 4 years of data is utilized to generate the forecasts and that it is more accurate than the analyst forecasts for all horizons.

In comparison with the time series models, the data driven approach exhibits many of the same characteristics, but achieves superiority by integrating more information and by embodying a contemporaneous advantage. For the specific models, the data driven approach generates the best results using a combination of the EBIT and total assets to form portfolios. Further, the data driven analyses using multiple variables are more accurate.

In comparison with the analyst forecasts, the superiority is identified in two ways. The first is through the small sample of 30 forecasts, where the data driven method is more accurate for all horizons. The second is indirectly through the data driven models being superior to the time series models, which for this type of data are superior to the analyst forecasts, whereby the data driven model is superior to the analysts.

Hence, the thesis has two main contributions to the literature. The first is the introduction of the data driven approach and its superiority to the time series and analyst forecasts. This means that earnings development in similar companies is a better indicator of how the company to be forecasted will develop compared to solely using the past earnings of the company itself. The second contribution is the finding that the ARIMA model is generally superior to the random walk forecasts. Both findings have been thoroughly tested for their robustness through 19 different models, 6 accuracy measures and 17 different combinations of data and forecast horizons, which leads to a total of 1938 tests.

Further, the findings have two main implications for the stakeholders of the earnings forecasting topic. Given the superiority of the data driven method compared to the time series models, it enables stakeholders of the venture capital community to make accurate forecasts for non-listed firms and thereby ease the valuation process significantly. At the same time, the data driven model is superior to analyst forecasts, whereby it can be applied to listed firms as well to provide an almost costless way of accurately predicting future earnings. Thereby, the model can also be of use to the managerial budgeting and resource allocation processes.

Nevertheless, the thesis is contingent on some limitations worth mentioning. The first two limitations relate to the dataset, which is bound to the Danish market and contains relatively few listed companies. Thus, the model's degree of international applicability remains a question for further research. The latter two limitations surround the data driven model's methodology and especially its requirement for data along with its way of identifying comparable companies. Therefore, the thesis encourages further research on this type of model to identify the feasibility of replication and to contribute to the method for identification of comparable firms.

9. References

ACC, Instituttet for Revision og Regnskab. 2014. "Notat Om ACC Databasen."

Albrecht, W. Steve, Larry L. Lookabill, and James C. McKeown. 1977. "The Time-Series Properties of Annual Earnings." *Journal of Accounting Research*.

Allee, Kristian Dietrich. 2008. "Estimating Cost of Equity Capital with Time-Series Forecasts of Earnings."

Ardalan, Roya K. 2016. "Improving Earnings and Dididends Forecasts Using Cointegration Analysis." *International Journal of Business*.

Bansal, Naresh, Jack Strauss, and Alireza Nasseh. 2015. "Can We Consistently Forecast a Firm's Earnings? Using Combination Forecast Methods to Predict the EPS of Dow Firms." *J Econ Finan*.

Bozdogan, Hamparsum. 1987. "Model Selection and Akaike's Information Criterion (AIC): The General Theory and Its Analytical Extensions." *Psychometrika*.

Bradshaw, Mark T., Michael S. Drake, James N. Myers, and Linda A. Myers. 2009. "A Re-Examination of Analysts' Superiority over Time-Series Forecasts."

Branson, Bruce C., Kenneth S. Lorek, and Donald P. Pagach. 1995. "Evidence on the Superiority of Analysts' Quarterly Earnings Forecasts for Small Capitalization Firms." *Decision Sciences*.

Brown, Lawrence D. 1993. "Earnings Forecasting Research: Its Implications for Capital Markets Research." *International Journal of Forecasting*.

Brown, Lawrence D., Robert L. Hagerman, Paul A. Griffin, and Mark E. Zmijewski. 1987. "Security Analyst Superiority Relative to Univariate Time-Series Models in Forecasting Quarterly Earnings." *Journal of Accounting and Economics*.

Brown, Lawrence D., Gordon D. Richardson, and Steven J. Schwager. 1987. "An Information Interpretation of Financial Analyst Superiority in Forecasting Earnings." *Journal of Accounting Research*.

Conroy, Robert, and Robert Harris. 1987. "Consensus Forecasts of Corporate Earnings: Analysts' Forecasts and Time Series Methods." *Management Science*.

Danmarks Statistik. 2001. "Nye Virksomheder 1999." *Nyt Fra Danmarks Statistik*.

———. 2003. "Tre Ud Af Fire Nye Virksomheder Går Ned." http://www.dst.dk/da/Statistik/bagtal/2003/2003-10-22-Virk.

———. 2014. "Færre Store Virksomheder I Den Private Sektor." https://www.dst.dk/pukora/epub/Nyt/2014/NR376.pdf.

Duke. 2017a. "Introduction to ARIMA: Nonseasonal Models." February 19. http://people.duke.edu/~rnau/411arim.htm#arima100.

———. 2017b. "Random Walk Model." February 19. https://people.duke.edu/~rnau/411rand.htm.

Elton, Edwin J., and Martin J. Gruber. 1972. "Earnings Estimates and the Accuracy of Expectational Data." *Management Science*.

Hopwood, William S., James C. Mckeown, and Paul Newbold. 1982. “The Additional Information Content of Quarterly Earnings Reports: Intertemporal Disaggregation.” *Journal of Accounting Research*.
Hyndman, Rob. 2017. “Rwf.” https://www.rdocumentation.org/packages/forecast/versions/7.1/topics/rwf.
Jadhav, Swati, Hongmei He, and Karl W. Jenkins. 2015. “Prediction of Earnings per Share for Industry.” *Proceedings of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*.
Kim, Young H. 1996. “Empirical Evaluation of Univariate Time-Series Methods and Combination of Forecasts for Earnings per Share.”
Liang, Yi-Yu. 2003. “Demand Modeling with the Geometric Brownian Motion Process.”
Little, Ian M. 1962. “Higgledy Piggledy Growth.” *Oxford Bulletin of Economics and Statistics*.
Lo, May H., and Pieter T. Elgers. 1998. “Alternative Adjustments to Analysts’ Earnings Forecasts: Relative and Complementary Performance.” *The Financial Review*.
Lobo, Gerald J. 1991. “Alternative Methods of Combining Security Analysts’ and Statistical Forecasts of Annual Corporate Earnings.” *International Journal of Forecasting*.
Nau, Robert. 2014. “Notes on the Random Walk Model.”
Newbold, Paul, J. Kenton Zumwalt, and Srinivasan Kannan. 1987. “Combining Forecasts to Improve Earnings per Share Prediction: An Examination of Electric Utilities.” *1nttmutiom.I Journal of Forecasting*.
O’Brien, Patricia C. 1988. “Analysts’ Forecasts as Earnings Expectations.” *Journal of Accounting and Economics*.
OTexts. 2017. “8.7 ARIMA Modelling in R.” https://www.otexts.org/fpp/8/7.
Pagach, Donald P., Barbara A. Chaney, and Bruce C. Branson. 2003. “A Note On Earnings Forecast Source Superiority.” *The Journal of Applied Business Research*.
Peek, Erik. 1997. “Earnings Forecasting Research: An Overview and Critique.”
Reverte, Carmelo, and Isidoro Guzman. 2010. “The Predictive Ability of Relative Efficiency for Future Earnings: An Application Using Data Envelopment Analysis to Spanish SMEs.” *Applied Economics*.
Skovmand, David. 2016. “Fundamentals of Time Series Analysis.”
Smith, M J de. 2015. *Statistical Analysis Handbook*.
Stern NYU. 2017. “Market Bubbles.” http://pages.stern.nyu.edu/~adamodar/New_Home_Page/invfables/bubbles.htm.
Watts, Ross L., and Richard W. Leftwich. 1977. “The Time Series of Annual Accounting Earnings.” *Journal of Accounting Research*.

10. Appendix

10.1 Tables

Table 1: Literature Overview

Paper	Sample Size and Data Horizon	Models Tested	Forecast Horizon	Superiority	Accuracy Measures
Allee (2008)	12,682 firms from 1970 to 2010	ARIMA(0,2,2)	NA, explains historical data rather than forecasting	Time Series models are superior	ME
Elton and Gruber (1972)	180 firms from 1962 to 1967	MA, 5 other TS models	1-3 years	Time Series models are at least as good as analysts	MSE
Albrecht, Lookabill, and McKeown (1977)	49 firms from 1947 to 1975	Random Walk, ARIMA(0,0,0), ARIMA(0,0,1), ARIMA(0,0,2), ARIMA(0,0,3), ARIMA(0,1,1), ARIMA(0,1,2), ARIMA(0,2,1), ARIMA(0,2,2), ARIMA(1,0,0), ARIMA(1,0,1), ARIMA(1,1,0), ARIMA(1,2,0), ARIMA(1,2,1), ARIMA(2,0,0), ARIMA(2,1,1), ARIMA(2,2,0), ARIMA(2,2,1)	3 years	Random Walk is superior	MRE, MAPE, MSRE
Little (1962)	522 firms from 1951 to 1959	Random Walk	3-5 years	Random Walk is superior	ME
Watts and Leftwich (1977)	32 firms from 1908 to 1974	Random Walk, ARIMA(1,0,0), ARIMA(0,1,1), ARIMA(0,1,2) , ARIMA(0,1,4)	1-3 years	Random Walk is superior	MAE
Bradshaw et al. (2009)	10,140 firms from 1983 to 2007	Random Walk	1-36 months	Random Walk is superior for horizons above 1 year	ME, MAPE
Branson, Lorek, and Pagach (1995)	223 firms from 1988 to 1989	ARIMA(0,1,1)	1 quarter	Random Walk is superior for smaller firms with few	MAPE

				analysts following	
Pagach, Chaney, and Branson (2003)	250 firms from 1979 to 1989	ARIMA(1,0,0) without drift	1-8 quarters	ARIMA is superior for smaller firms with high earnings variability and at longer time horizons	MAPE
Conroy and Harris (1987)	600 firms from 1963 to 1983	Random Walk, Simple Average, ARIMA(0,1,1), ARIMA(0,2,2), ARIMA(0,3,3)	1-9 months	Random Walk is superior for longer horizons. Analysts are superior closer to the year end.	MAPE
Brown (1993)	NA, no data considered	Random Walk, ARIMA(0,1,1), ARIMA(1,0,0)	1-4 quarters	Analysts are superior to time series models. Random Walk is superior to ARIMA.	ME
Brown et al. (1987)	233 firms from 1975 to 1980	3 TS models	1-3 quarters	Analysts are superior	ME
Brown, Richardson, and Schwager (1987)	702 firms from 1977 to 1982	Random Walk	1-6 quarters	Analysts are superior	MSE
O'Brien (1988)	184 firms from 1975 to 1982	ARIMA(1,0,0)	5 days to 1 year	Analysts are superior	MAE
Peek (1997)	NA, no data considered	Time Series models in general	NA, no data considered	Analysts are superior	MRE, MAE
Hopwood, Mckeown, and Newbold (1982)	258 firms from 1974 to 1978	Random Walk, 7 other TS models	1-4 quarters	Analysts are superior on quarterly earnings	MAE
Newbold, Zumwalt, and Kannan (1987)	65 firms from 1962 to 1984	ARIMA(0,1,1), ARIMA(1,0,0) with drift	1 year	Analysts are superior to ARIMA, but combination is superior to both	MSE

Bansal, Strauss, and Nasseh (2015)	30 firms from 1970 to 2009	AR, variable driven model	4 to 8 quarters	Variable driven forecasts are superior	MSE
Reverte and Guzman (2010)	1939 firms from 1999 to 2004	Relative efficiency model	1 year	Variable driven forecasts are superior	MAPE
Lobo (1991)	96 firms from 1961 to 1983	ARIMA(0,1,1)	6 years	Combinations of forecasts are superior	MAPE, MSRE
Kim (1996)	30 firms, period is not available	ARIMA(0,1,1), ARIMA(1,0,0), random walk	1 year	Combinations of forecasts are superior	NA
Lo and Elgers (1998)	511 firms from 1976 to 1989	Analyst forecast only	1 year	Combinations of forecasts are superior	MSE
Jadhav, He, and Jenkins (2015)	6 firms, period is not available	LR, RBF and MLP	NA	Statistical models are superior	NA
Ardalan (2016)	2 indices for a period of 27 years	Error Correction Model (ECM)	NA	Statistical models are superior	MSE

Table 7: Portfolio Example

Variable 1	Variable 2	Variable 3	Final Portfolio
1	1	1	1
2	1	1	2
3	1	1	3
1	2	1	4
2	2	1	5
3	2	1	6
1	3	1	7
2	3	1	8
3	3	1	9
1	1	2	10
2	1	2	11
3	1	2	12
1	2	2	13
2	2	2	14
3	2	2	15
1	3	2	16
2	3	2	17
3	3	2	18
1	1	3	19

2	1	3	20
3	1	3	21
1	2	3	22
2	2	3	23
3	2	3	24
1	3	3	25
2	3	3	26
3	3	3	27

Table 8: Analyst Companies

Company Name	CVR	Forecast Horizon
COLOPLAST A/S	DK69749917	4 years
IC GROUP A/S	DK62816414	4 years
CARLSBERG A/S	DK61056416	4 years
FLSMIDTH & CO. A/S	DK58180912	4 years
H. LUNDBECK A/S	DK56759913	4 years
ROCKWOOL INTERNATIONAL A/S	DK54879415	4 years
ROYAL UNIBREW A/S	DK41956712	4 years
BANG & OLUFSEN A/S	DK41257911	4 years
GN STORE NORD A/S	DK24257843	4 years
NOVO NORDISK A/S	DK24256790	4 years
A.P. MØLLER - MÆRSK A/S	DK22756214	4 years
BAVARIAN NORDIC A/S	DK16271187	4 years
VESTAS WIND SYSTEMS A/S	DK10403782	4 years
NOVOZYMES A/S	DK10007127	4 years
DSV A/S	DK58233528	4 years
NETOP SOLUTIONS A/S	DK16221503	4 years
RTX A/S	DK17002147	4 years
TK DEVELOPMENT A/S	DK24256782	4 years
Santa Fe Group A/S	DK26041716	4 years
BOCONCEPT HOLDING A/S	DK34018413	4 years
NORTH MEDIA A/S	DK66590119	4 years
SJÆLSØ GRUPPEN A/S	DK89801915	4 years
NEUROSEARCH A/S	DK12546106	4 years
SOLAR A/S	DK15908416	4 years
SANISTÅL A/S	DK42997811	3 years
MONBERG & THORSEN A/S	DK12617917	3 years
NORDICOM A/S	DK12932502	3 years
COLUMBUS A/S	DK13228345	3 years
JEUDAN A/S	DK14246045	3 years
TORM A/S	DK22460218	3 years

Table 13: Time Series Comparison, Full, MAPE

	MAPE							
Data horizon	*Partial*	*Partial*	*Partial*	*Full*	*Full*	*Full*	*Partial*	*Full*
Forecast horizon	*Short*	*Medium*	*Long*	*Short*	*Medium*	*Long*	*All*	*All*
AutoArima	4.3	3.8	4.1	2.4	3.4	3.7	4.1	3.0
ARIMA(0,1,1)	7.0	3.8	3.1	7.6	3.4	4.9	4.3	6.0
ARIMA(1,0,0)	7.0	3.8	3.1	7.6	3.4	4.9	4.3	6.0
rwfNoDrift	6.7	6.7	8.0	4.6	4.7	5.6	7.3	4.9
rwfDrift	13.5	19.4	29.9	6.3	9.2	13.0	23.2	8.3
rwBrown	19.6	23.7	36.4	14.7	23.4	31.2	29.1	20.0
EBIT5	11.2	12.0	12.2	3.5	4.4	6.8	11.9	4.4
Industry	19.0	9.3	20.2	5.3	6.0	8.8	17.2	6.4
EBIT5&Industry	12.8	12.9	12.2	3.8	4.9	8.0	12.5	4.8
Age	12.1	12.7	11.8	4.4	3.4	4.0	12.1	4.2
Balance	19.4	18.3	81.6	13.0	18.6	9.6	50.3	18.3
EBIT5&Balance	5.1	4.0	6.3	1.9	2.6	3.3	5.4	2.4
Growth5	6.4	15.8	80.0	2.4	4.3	8.9	45.6	3.9
EBIT5&Growth5	7.9	8.7	11.1	3.2	4.1	6.0	9.7	3.9
EBIT5&Growth5&Balance	4.2	4.8	8.1	2.2	2.4	2.8	6.3	2.3
EBIT5&Growth5&Industry	9.1	11.6	12.5	4.1	5.2	7.9	11.4	5.2
EBIT5&Growth10&Balance	10.0	13.6	15.9	4.3	9.6	5.5	13.8	5.8
EBIT10	10.1	10.9	13.4	6.2	4.4	6.0	11.9	5.7
Growth10	8.3	292.0	62.2	2.3	7.7	8.1	106.1	4.4

Table 13: Time Series Comparison, Full, MAPE (Continued)

	MAPE			
Data horizon	*All*	*All*	*All*	*All*
Forecast horizon	*Short*	*Medium*	*Long*	*All*
AutoArima	3.0	3.6	4.4	3.5
ARIMA(0,1,1)	7.4	3.6	3.9	5.2
ARIMA(1,0,0)	7.4	3.6	3.9	5.2
rwfNoDrift	5.2	5.7	6.6	6.0
rwfDrift	8.4	14.3	22.4	15.3
rwBrown	16.1	23.6	32.4	24.2
EBIT5	5.7	8.2	10.4	7.9
Industry	9.2	7.6	16.0	11.5
EBIT5&Industry	6.3	8.9	10.5	8.4
Age	6.6	8.1	8.8	7.9

Balance	14.8	18.5	34.4	33.3
EBIT5&Balance	2.8	3.3	4.8	3.8
Growth5	3.6	10.1	33.3	23.5
EBIT5&Growth5	4.5	6.4	8.6	6.7
EBIT5&Growth5&Balance	2.7	3.6	5.0	4.2
EBIT5&Growth5&Industry	5.5	8.4	10.9	8.1
EBIT5&Growth10&Balance	6.0	11.6	9.3	9.6
EBIT10	7.3	7.6	9.6	8.6
Growth10	4.0	149.8	42.8	52.3

Table 14: Time Series Comparison, Full, MSRE

	MSRE							
Data horizon	*Partial*	*Partial*	*Partial*	*Full*	*Full*	*Full*	*Partial*	*Full*
Forecast horizon	*Short*	*Medium*	*Long*	*Short*	*Medium*	*Long*	*All*	*All*
AutoArima	3866	2361	1941	243	683	946	2527	591
ARIMA(0,1,1)	5287	2361	982	8732	683	1859	2403	5408
ARIMA(1,0,0)	5287	2361	982	8732	683	1859	2403	5408
rwfNoDrift	4879	4174	7076	2194	711	1973	5801	1844
rwfDrift	11881	70657	118567	3042	2571	9402	79918	3970
rwBrown	39744	41731	96436	8813	22445	50456	68587	19685
EBIT5	5207	4305	4028	1094	471	2128	4392	1127
Industry	170641	8950	78508	3008	1374	7248	84152	3452
EBIT5&Industry	9755	18146	4559	1324	721	4197	9255	1535
Age	15373	17325	14969	2031	374	989	15659	1461
Balance	48082	31650	2218807	29088	9487	5583	1129337	35572
EBIT5&Balance	1146	502	2748	362	203	387	1786	366
Growth5	4626	64470	2970614	766	1161	5308	1502581	1432
EBIT5&Growth5	2960	2647	6368	1177	554	1935	4586	1124
EBIT5&Growth5&Balance	843	911	8585	273	153	106	4731	207
EBIT5&Growth5&Industry	5430	5407	6791	1680	1726	4247	6105	2342
EBIT5&Growth10&Balance	7413	33082	79268	1299	16485	528	49758	4502
EBIT10	9502	11059	23387	4290	572	2062	16834	2917
Growth10	5453	95189334	901264	1342	10319	2854	24249329	3452

Table 15: Time Series Comparison, Full, MAE

	MAE							
Data horizon	*Partial*	*Partial*	*Partial*	*Full*	*Full*	*Full*	*Partial*	*Full*
Forecast horizon	*Short*	*Medium*	*Long*	*Short*	*Medium*	*Long*	*All*	*All*
AutoArima	1280	1640	1204	910	956	794	1332	908
ARIMA(0,1,1)	1362	1640	797	1165	956	808	1149	1051
ARIMA(1,0,0)	1362	1640	797	1165	956	808	1149	1051
rwfNoDrift	1120	1459	1201	955	983	806	1245	947
rwfDrift	1545	2150	2734	1041	1264	1232	2290	1153
rwBrown	3511	5681	4499	2304	2685	2701	4548	2482
EBIT5	926	1054	1045	726	844	817	1018	780
Industry	1519	1363	1857	1512	1116	1112	1649	1372
EBIT5&Industry	941	1040	1029	821	886	845	1010	848
Age	1172	1385	1294	828	845	715	1286	822
Balance	2916	3270	7519	2454	3860	1756	5306	3353
EBIT5&Balance	1120	1163	1026	782	847	660	1084	779
Growth5	833	1306	3183	528	833	1148	2126	700
EBIT5&Growth5	722	898	981	500	711	867	895	610
EBIT5&Growth5&Balance	656	973	1128	413	649	440	971	473
EBIT5&Growth5&Industry	825	1035	1151	611	796	1031	1041	737
EBIT5&Growth10&Balance	937	1306	1710	783	744	549	1416	720
EBIT10	1220	1596	1461	1070	880	835	1434	983
Growth10	2302	36261	3540	511	950	1110	11411	707

Table 17: Time Series Comparison, Grouped Full, MAPE

	MAPE					
Data horizon	*Partial*	*Partial*	*Partial*	*Full*	*Full*	*Full*
Forecast horizon	*Short*	*Medium*	*Long*	*Short*	*Medium*	*Long*
Time Series	9.7	10.2	14.1	7.2	7.9	10.6
ARIMAs	6.1	3.8	3.4	5.9	3.4	4.5
RWs	13.3	16.6	24.8	8.5	12.4	16.6
Data Driven	10.4	32.8	26.7	4.3	6.0	6.6
EBITs	8.8	9.8	11.5	3.6	4.7	5.8
Balances	9.7	10.2	28.0	5.3	8.3	5.3
Growths	7.6	57.7	31.6	3.1	5.5	6.5
Industries	13.6	11.3	15.0	4.4	5.3	8.2
5 portfolios	8.1	10.0	20.4	3.0	4.0	6.2
10 portfolios	9.4	105.5	30.5	4.3	7.2	6.5
1 variables	12.3	53.0	40.2	5.3	7.0	7.5

2 variables	8.6	8.5	9.9	2.9	3.9	5.8
3 variables	7.8	10.0	12.2	3.5	5.7	5.4

Table 18: Time Series Comparison, Grouped Full, MSRE

	MSRE					
Data horizon	*Partial*	*Partial*	*Partial*	*Full*	*Full*	*Full*
Forecast horizon	*Short*	*Medium*	*Long*	*Short*	*Medium*	*Long*
Time Series	11824	20607	37664	5293	4629	11083
ARIMAs	4813	2361	1301	5902	683	1555
RWs	18835	38854	74026	4683	8575	20611
Data Driven	22033	7337522	486146	3672	3354	2890
EBITs	5282	9508	16967	1437	2610	1949
Balances	14371	16537	577352	7755	6582	1651
Growths	4454	15882642	662148	1090	5066	2496
Industries	61942	10835	29953	2004	1273	5231
5 portfolios	4281	13770	429099	954	712	2615
10 portfolios	7456	31744492	334640	2310	9125	1814
1 variables	36983	13618156	887368	5946	3394	3739
2 variables	4620	7099	4558	954	492	2173
3 variables	4562	13134	31548	1084	6121	1627

Table 19: Time Series Comparison, Grouped Full, MAE

	MAE					
Data horizon	*Partial*	*Partial*	*Partial*	*Full*	*Full*	*Full*
Forecast horizon	*Short*	*Medium*	*Long*	*Short*	*Medium*	*Long*
Time Series	1697	2369	1872	1257	1300	1192
ARIMAs	1335	1640	932	1080	956	803
RWs	2059	3097	2811	1433	1644	1580
Data Driven	1238	4050	2071	888	1074	914
EBITs	918	1133	1191	713	795	755
Balances	1407	1678	2846	1108	1525	851
Growths	1046	6963	1949	558	780	857
Industries	1095	1146	1346	981	933	996
5 portfolios	861	1067	1363	626	795	830
10 portfolios	1486	13055	2237	788	858	831
1 variables	1555	6605	2843	1090	1333	1071
2 variables	928	1034	1012	701	815	790
3 variables	806	1105	1330	602	730	673

Table 21: Analyst Forecast Comparison, Full, MAPE

	MAPE								
Data Horizon	*6*	*6*	*6*	*6*	*6*	*7*	*8*	*9*	*All*
Forecast Horizon	*1*	*2*	*3*	*4*	*1*	*1*	*1*	*1*	*All*
Analysts	1.2	1.3	1.2	1.4	1.2	1.3	1.2	1.4	1.3
AutoArima	1.6	2.5	3.9	3.3	1.6	1.4	1.3	1.6	2.2
ARIMA(0,1,1)	1.6	1.1	3.9	3.3	1.6	1.6	1.3	1.6	2.0
ARIMA(1,0,0)	1.6	1.1	3.9	3.3	1.6	1.6	1.3	1.6	2.0
rwfNoDrift	0.6	1.1	1.6	1.7	0.6	1.4	1.7	1.5	1.3
rwfDrift	2.2	3.9	6.5	5.6	2.2	2.9	3.8	1.7	3.6
rwBrown	4.6	6.2	10.5	9.5	4.6	5.7	7.1	2.1	6.3
EBITabs5	0.5	1.0	1.3	1.4	0.5	1.2	1.2	0.8	1.0
Industry	0.8	1.1	1.8	1.7	0.8	1.4	1.9	1.7	1.4
EBIT&Industry	0.6	1.0	1.3	1.4	0.6	1.1	1.0	1.0	1.0
Age	0.6	0.8	0.8	0.9	0.6	1.0	1.0	2.0	0.9
Balance	1.6	1.9	3.3	3.6	1.6	1.4	4.1	1.9	2.4
EBIT&Balance	0.8	0.9	1.0	1.0	0.8	0.9	0.9	0.9	0.9
Growth5	0.2	0.9	1.2	1.4	0.2	0.8	0.5	0.8	0.8
EBIT&Growth	0.2	0.9	1.2	1.3	0.2	0.4	0.6	0.9	0.7
EBIT&Growth5&Balance	0.2	0.9	1.1	1.3	0.2	0.4	0.5	0.6	0.7
EBIT&Growth&Industry	0.2	0.8	1.2	1.4	0.2	0.4	1.4	0.6	0.8
EBIT&Growth10&Balance	1.4	1.6	2.2	2.6	1.4	1.4	2.0	2.8	1.9
EBITabs10	0.9	1.2	1.6	1.6	0.9	1.4	2.3	2.5	1.6
Growth10	0.1	0.8	1.1	1.3	0.1	0.3	0.5	0.7	0.6

Table 23: Analyst Forecast Comparison, Full, MSRE

	MSRE								
Data Horizon	*6*	*6*	*6*	*6*	*6*	*7*	*8*	*9*	*All*
Forecast Horizon	*1*	*2*	*3*	*4*	*1*	*1*	*1*	*1*	*All*
Analysts	6.6	6.1	4.8	28.7	6.6	6.1	4.8	28.7	11.6
AutoArima	29.5	80.1	251.9	191.2	29.5	6.7	6.8	11.1	75.9
ARIMA(0,1,1)	29.5	5.1	251.9	191.2	29.5	9.3	6.8	11.1	66.8
ARIMA(1,0,0)	29.5	5.1	251.9	191.2	29.5	9.3	6.8	11.1	66.8
rwfNoDrift	0.9	5.1	12.8	12.3	0.9	7.0	10.4	9.8	7.4
rwfDrift	84.8	239.8	839.3	635.4	84.8	75.8	142.4	13.6	264.5
rwBrown	576.3	912.5	2662.8	1894.4	576.3	540.5	1052.5	16.0	1028.9
EBITabs5	0.4	3.5	7.6	7.6	0.4	5.5	4.1	1.6	3.8
Industry	1.5	3.9	16.8	15.7	1.5	6.5	13.5	12.4	9.0

EBIT&Industry	0.7	2.6	8.7	9.8	0.7	4.1	2.4	3.1	4.0
Age	0.4	1.1	0.9	1.1	0.4	2.2	1.9	18.4	3.3
Balance	2.8	4.5	30.1	31.9	2.8	2.3	38.7	5.4	14.8
EBIT&Balance	0.6	0.9	2.5	2.4	0.6	1.1	1.4	1.0	1.3
Growth5	0.1	4.6	6.1	7.3	0.1	3.2	0.9	2.4	3.1
EBIT&Growth	0.1	4.4	5.3	5.6	0.1	0.6	1.8	8.1	3.3
EBIT&Growth5&Balance	0.1	4.2	5.1	5.3	0.1	0.2	1.4	2.7	2.4
EBIT&Growth&Industry	0.1	4.0	6.1	7.8	0.1	0.6	14.9	1.9	4.4
EBIT&Growth10&Balance	2.7	4.1	23.5	24.8	2.7	3.7	11.7	18.9	11.5
EBITabs10	3.1	5.3	9.7	9.6	3.1	7.3	28.0	33.7	12.5
Growth10	0.0	4.2	4.8	7.4	0.0	0.5	1.8	2.4	2.6

10.2 R Code

```
rm(list=ls())
setwd("~/CBS Courses Master/Master Thesis/Data")

library(scales)
library(plyr)
library(haven)
library(forecast)
data <- read_sas("dst_jeppe.sas7bdat")

#variable distribution
FoundNum = as.matrix(Found[complete.cases(Found),])
dFound = density(FoundNum)
plot(dFound,col="black",xlab="Year",xlim=c(1980,2014)) #main="FOUND Distribution"
NewFirms = dFound$y*352496
years = round(c(dFound$x[450],dFound$x[460],dFound$x[468],dFound$x[478],dFound$x[486],
          dFound$x[495],dFound$x[505]),0)
yearsVec = c(450,460,468,478,486,495,505)
plot(NewFirms,xaxt = "n",xlab="Year",xlim=c(450,512)) # main="No. of New Companies"
axis(1, at=yearsVec, labels=years)

#-------- 1.1.2 Data - Forecating Setup ----------
#Part to be updated every time
fullDataLength = 1500000
dataLength = fullDataLength
dataLengthTest = 0

start = dataLengthTest + 1
end = start + dataLength - 1
#end = 2891384

dataStart = 1   #first year of the data used to generate model - HAS TO BE 1 !!!
dataEnd = 9     #last year of the data used to generate model
horizon = 1     #forecast horizon after dataEnd
inputYear = dataEnd   #Year for grouping the portfolios = last data year

dataEndCol = dataEnd-dataStart+1
horizonStartCol = dataEndCol+1
horizonEndCol = dataEndCol+horizon
```

```
#Same part every time
smallData = data[start:end,]
firms = length(t(unique(smallData[,1])))

CVR = smallData[,1]
Industry = smallData[,5]
CF = smallData[,14] #in thousands DKK
EBIT = smallData[,40] #in thousands DKK
Revenue = smallData[,45] #in thousands DKK
Balance = smallData[,49] #in thousands DKK (Total Assets)
Found = smallData[,54]
Aryear = smallData[,55]
Year = Aryear - Found
Emps = smallData[,56]

#Year cleanup, removes negative values and values above 40 years
for (i in 1:dataLength) {
  if(is.na(as.numeric(Year[i,]))){
  } else if(as.numeric(Year[i,])<0){
    Year[i,] = NA
  } else if(as.numeric(Year[i,])>39){
    Year[i,] = NA
  }
}

IndNo = vector("numeric",length = dataLength)
Growth = vector("numeric",length = dataLength)

relSmallData = cbind(CVR,Industry,CF,EBIT,Found,Aryear,Year,Balance,Emps,IndNo,Growth)
relSmallData[1:(dataLength-1),11] = round(relSmallData[2:dataLength,4]/relSmallData[1:(dataLength-1),4],4)-1

Growth = as.matrix(relSmallData[,11])

relSmallDataUnique = matrix(nrow = firms, ncol = 5) #CVR, Idnustry x2, Found x2
relSmallDataUnique[,1] = t(unique(CVR))
colnames(relSmallDataUnique) <- c("CVR","Industry","Found","IndustryNo","Age")

CFdata = matrix(nrow = firms, ncol = 41)
CFdata[,1] = t(unique(CVR))

for (i in 1:dataLength) {
  CFval = as.numeric(EBIT[i,]) #using EBIT instead of CF
  CVRval = as.character(CVR[i,])
  Yearval = as.numeric(Year[i,])

  rowNum = which(CFdata[,1]==CVRval)
  CFdata[rowNum,Yearval+2]=CFval

  relSmallDataUnique[rowNum,2] = as.character(Industry[i,])
  relSmallDataUnique[rowNum,3] = as.numeric(Found[i,])

  if(is.na(relSmallDataUnique[rowNum,3])){
    relSmallDataUnique[rowNum,5] = NA
  } else {relSmallDataUnique[rowNum,5] = 2014-as.numeric(relSmallDataUnique[rowNum,3])}
}                               #needs to be 2014 for forecast to work

colnames(CFdata) <- c("CVR",0:39) #first column is CVR, second is year 0
```

```
Balancedata = matrix(nrow = firms, ncol = 41)
Balancedata[,1] = t(unique(CVR))

for (i in 1:dataLength) {
  Balval = as.numeric(Balance[i,])
  CVRval = as.character(CVR[i,])
  Yearval = as.numeric(Year[i,])

  rowNum = which(Balancedata[,1]==CVRval)
  Balancedata[rowNum,Yearval+2]=Balval
}

#generating the accuracy measurment errors matrix
globalErrors = matrix(data=0,nrow=19,ncol=6) # given 13 analyses and 6 stat models
colnames(globalErrors) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")

test1 = "EBITabs5"
test2 = "Industry"
test3 = "EBIT&Industry"
test4 = "Age"
test5 = "Balance"
test6 = "EBIT&Balance"
test7 = "Growth5"
test8 = "EBIT&Growth"
test9 = "EBIT&Growth5&Balance"
test10 = "EBIT&Growth&Industry"
test11 = "EBIT&Growth10&Balance"
test12 = "EBITabs10"
test13 = "Growth10"

tests = c(test1,test2,test3,test4,test5,test6,test7,test8,test9,test10,test11,test12,test13)
models = c("AutoArima","InputArima1","InputArima2","rwfNoDrift","rwfDrift","rwBrown")
rownames(globalErrors) <- c(models,tests)

#Generating the list of ages
AgeUniqueLenght = length(unique(relSmallDataUnique[,5]))
AgeList = matrix(NA,ncol = 1, nrow = AgeUniqueLenght)
AgeList[,1] = unique(relSmallDataUnique[,5])

#Generating the list of industries
Industries = unique(data[,5]) #takes all unique values from master data file for NACED
IndLength = length(t(Industries))
IndType = cbind(Industries,NA)

for(i in 1:IndLength){
  if(is.na(as.numeric(IndType[i,1]))){
    IndType[i,2]="character"
  } else{IndType[i,2]="numeric"}
} #will produce warnings, but works

IndustryList = IndType[IndType[,2] == "character",]
IndUniqueLenght = length(IndustryList[,1])
IndustryNo = c(1:IndUniqueLenght)
IndustryList = cbind(IndustryList,IndustryNo)

for(i in 1:firms){
  if(is.na(as.numeric(relSmallDataUnique[i,2]))){
    rowVal = which(IndustryList[,1]==relSmallDataUnique[i,2])
```

```
   relSmallDataUnique[i,4] = IndustryList[rowVal,3]
  } else{
   relSmallDataUnique[i,4] = 1
  }
} #will produce warnings, but works

for(i in 1:dataLength){
  row = which(relSmallDataUnique[,1]==relSmallData[i,1])
  relSmallData[i,10] = relSmallDataUnique[row,4]
}

#-------- 1.1.3 Data - Test Setup --------
startRow = 1
endRow = dataLengthTest

#DataTest = data[startRow:endRow,]
require(readxl)
testData <- read_excel("Bloomberg Data.xlsx",sheet = "RexportNew", col_names=TRUE)
dataLengthTest = length(t(testData[,1]))
DataTest = testData

firmsTest = length(t(unique(DataTest[,1])))

CVR.t = DataTest[,1]
Industry.t = DataTest[,5]
EBIT.t = DataTest[,40] #in thousands DKK
Balance.t = DataTest[,49] #in thousands DKK (Total Assets)
Found.t = DataTest[,54]
Aryear.t = DataTest[,55]
Year.t = Aryear.t - Found.t
Emps.t = DataTest[,56]

#Year cleanup, removes negative values and values above 40 years
for (i in 1:dataLengthTest) {
  if(is.na(as.numeric(Year.t[i,]))){
  } else if(as.numeric(Year.t[i,])<0){
    Year.t[i,] = NA
  } else if(as.numeric(Year.t[i,])>39){
    Year.t[i,] = NA
  }
}

IndNo.t = vector("numeric",length = dataLengthTest)
Growth.t = vector("numeric",length = dataLengthTest)
relDataTest = cbind(CVR.t,Industry.t,EBIT.t,Found.t,Aryear.t,Year.t,Balance.t,Emps.t,IndNo.t,Growth.t)
relDataTest[1:(dataLengthTest-1),10] = round(relDataTest[2:dataLengthTest,3]/relDataTest[1:(dataLengthTest-1),3],4)-1
relDataTestUnique = matrix(nrow = firmsTest, ncol = 5) #CVR, Idnustry x2, Found x2
relDataTestUnique[,1] = t(unique(CVR.t))
colnames(relDataTestUnique) <- c("CVR","Industry","Found","IndustryNo","Age")
CFdataTest = matrix(nrow = firmsTest, ncol = 41)
CFdataTest[,1] = t(unique(CVR.t))

for (i in 1:dataLengthTest) {
  CFval = as.numeric(EBIT.t[i,])
  CVRval = as.character(CVR.t[i,])
  Yearval = as.numeric(Year.t[i,])
  rowNum = which(CFdataTest[,1]==CVRval)
  CFdataTest[rowNum,Yearval+2]=CFval
```

```
  relDataTestUnique[rowNum,2] = as.character(Industry.t[i,])
  relDataTestUnique[rowNum,3] = as.numeric(Found.t[i,])
  if(is.na(relDataTestUnique[rowNum,3])){
    relDataTestUnique[rowNum,5] = NA
  } else {relDataTestUnique[rowNum,5] = 2014-as.numeric(relDataTestUnique[rowNum,3])}
}

colnames(CFdataTest) <- c("CVR",0:39)

for(i in 1:firmsTest){
  if(is.na(as.numeric(relDataTestUnique[i,2]))){
    rowVal = which(IndustryList[,1]==relDataTestUnique[i,2])
    relDataTestUnique[i,4] = IndustryList[rowVal,3]
  } else{
    relDataTestUnique[i,4] = 1
  }
}
#only run this part when going to the analyses, not before
if(IndustryList[1,1]==""){ #IMPORTANT TO KEEP
  IndustryList[1,1]=" "
}

for(i in 1:dataLengthTest){
  row = which(relDataTestUnique[,1]==relDataTest[i,1])
  relDataTest[i,9] = relDataTestUnique[row,4]
}

BalancedataTest = matrix(nrow = firmsTest, ncol = 41)
BalancedataTest[,1] = t(unique(CVR.t))
for (i in 1:dataLengthTest) {
  Balval = as.numeric(Balance.t[i,])
  CVRval = as.character(CVR.t[i,])
  Yearval = as.numeric(Year.t[i,])
  rowNum = which(BalancedataTest[,1]==CVRval)
  BalancedataTest[rowNum,Yearval+2]=Balval
}
GrowthdataTest = matrix(nrow = firmsTest, ncol = 41)
GrowthdataTest[,1] = t(unique(CVR.t))
for (i in 1:dataLengthTest) {
  Growval = as.numeric(relDataTest[i,10])
  CVRval = as.character(CVR.t[i,])
  Yearval = as.numeric(Year.t[i,])

  rowNum = which(GrowthdataTest[,1]==CVRval)
  GrowthdataTest[rowNum,Yearval+2]=Growval
}

EBITInputYear = CFdataTest[,inputYear+2]
BalInputYear = BalancedataTest[,inputYear+2]
#EmpInputYear = EmpdataTest[,inputYear+2]
GrowInputYear = GrowthdataTest[,inputYear+2]
IndInput = relDataTestUnique[,4]
AgeInput = relDataTestUnique[,5]
PortMatrix = matrix(nrow = firmsTest,ncol = 8) # 8 analyses so far + 2 + (emps removed)
colnames(PortMatrix) <- c("Bal","EBIT","Growth","EBIT&Ind","EBIT&Bal","EBIT&Growth",
                          "EBIT&Grow&Ind","EBIT&Grow&Bal")

CFdataTestNum = cbind(CFdataTest[,2:41],EBITInputYear,BalInputYear,GrowInputYear,
```

```
                IndInput,AgeInput,PortMatrix) # needs to be from Y0
CFdataTestComp = CFdataTestNum[complete.cases(CFdataTestNum[,(dataStart+1)]),]
for (i in (2+dataStart):(horizonEndCol+1)) {
 CFdataTestComp = CFdataTestComp[complete.cases(CFdataTestComp[,i]),]
}
for (i in 41:45) {
 CFdataTestComp = CFdataTestComp[complete.cases(CFdataTestComp[,i]),]
}
CFdataTestComp[is.na(CFdataTestComp)] = 0
class(CFdataTestComp) <- "numeric"

#-------- 1.2.1 Data Overview - General ----------
testYears = 15
dataOverviewFullForecast = matrix(0,nrow = testYears,ncol = 2)
dataOverviewFullForecast[,1] = 1:testYears
colnames(dataOverviewFullForecast) = c("DataYears","HorizonFirms")

for(j in 1:testYears){
 CFdataComp = CFdata[,3:41] # removes year 0
 class(CFdataComp) <- "numeric"
 inputyear = j
 years = horizonEndCol+inputyear-1
 dataOverview = matrix(0,nrow = years,ncol = 2)
 colnames(dataOverview) = c("DataYears","Firms")
 for (i in inputyear:years) { #Year 0 is excluded as there are so few data points there
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i]),]
  dataOverview[i,1] = i
  dataOverview[i,2] = length(CFdataComp[,1])
 }
 dataOverviewFullForecast[j,2] = dataOverview[years,2]
}
testYears = 15
dataOverviewFullTest = matrix(0,nrow = testYears,ncol = 2)
dataOverviewFullTest[,1] = 1:testYears
colnames(dataOverviewFullTest) = c("DataYears","HorizonFirms")
for(j in 1:testYears){
 CFdataComp = CFdataTest[,3:41] # removes year 0
 class(CFdataComp) <- "numeric"

 inputyear = j
 years = horizonEndCol+inputyear-1
 dataOverview = matrix(0,nrow = years,ncol = 2)
 colnames(dataOverview) = c("DataYears","Firms")
 for (i in inputyear:years) { #Year 0 is excluded as there are so few data points there
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i]),]
  dataOverview[i,1] = i
  dataOverview[i,2] = length(CFdataComp[,1])
 }
 dataOverviewFullTest[j,2] = dataOverview[years,2]
}
#-------- 2.1.1 portfolio development - EBIT5 --------
portfolios = 5  #no. of EBIT groupings
CFdataComp = CFdata[complete.cases(CFdata[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
 CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
firmsNo = length(CFdataComp[,1])
firmsList = CFdataComp[,1]
```

```
quantiles = quantile(as.numeric(CFdataComp[,inputYear+2]), c(.2, .4, .6, .8, 1))
quantilMatrix = matrix(c(quantiles,5:1),nrow=5,ncol=2)
portfolioNo = vector(mode="numeric", firmsNo)
CFdataComp = cbind(CFdataComp,portfolioNo)
for (i in 1:firmsNo) {
EBITinputYear = as.numeric(CFdataComp[i,inputYear+2])

  if(EBITinputYear<quantilMatrix[1,1]){
    CFdataComp[i,42]=5
  } else if(EBITinputYear<quantilMatrix[2,1]){
    CFdataComp[i,42]=4
  } else if(EBITinputYear<quantilMatrix[3,1]){
    CFdataComp[i,42]=3
  } else if(EBITinputYear<quantilMatrix[4,1]){
    CFdataComp[i,42]=2
  } else{
    CFdataComp[i,42]=1
  }
}

CFdataNum=CFdataComp[,2:42] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
CFdataPortSum = matrix(0,nrow = portfolios,ncol = 40) #year 0 to 39 of data, no CVRs
for (i in 1:firmsNo) {
  portNum = CFdataNum[i,41]
  CFdataPortSum[portNum,]=CFdataNum[i,1:40]+CFdataPortSum[portNum,]
}
CFdataPortAvg = matrix(0,nrow = portfolios,ncol = 40) #year 0 to 39 of data, no CVRs
PortSize = vector(mode = "numeric",length = portfolios)
for (i in 1:portfolios) {
  PortSize[i] = sum(CFdataNum[,41] == i)
}
CFdataPortAvg = CFdataPortSum[,(dataStart+1):(horizonEndCol+1)]/PortSize
#forecasting
firmsNo = length(CFdataTestComp[,1])

CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecastEBIT1 = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataTestComp[,(dataStart+1):(dataEnd+horizon+1)]
CFforecastEBIT1[,1:dataEnd] = CFdataTestComp[,(dataStart+1):(dataEnd+1)]
CFdataPortPer = CFdataPortAvg[,2:(horizonEndCol)]/CFdataPortAvg[,1:(horizonEndCol-1)]-1
for(i in 1:firmsNo){
  inputEBIT = CFforecastEBIT1[i,(dataEndCol)]
  if(inputEBIT<quantilMatrix[1,1]){
    CFdataTestComp[i,47]=5
  } else if(inputEBIT<quantilMatrix[2,1]){
    CFdataTestComp[i,47]=4
  } else if(inputEBIT<quantilMatrix[3,1]){
    CFdataTestComp[i,47]=3
  } else if(inputEBIT<quantilMatrix[4,1]){
    CFdataTestComp[i,47]=2
  } else{
    CFdataTestComp[i,47]=1
  }
  inputPort = CFdataTestComp[i,47]
  for(j in 1:horizon){
    CFforecastEBIT1[i,(dataEndCol+j)] = round(CFforecastEBIT1[i,(dataEndCol+j-1)]*
```

```
                        (1+CFdataPortPer[inputPort,(dataEndCol+j-1)]),2)
  }
}
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastEBIT1[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF7 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF7) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF7[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF7[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF7[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF7[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF7[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
  localErrorsF7[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
  globalErrors[7,i] = round(mean(localErrorsF7[!rowSums(!is.finite(localErrorsF7)),i]),4)
#-------- 2.1.2 portfolio development - EBIT10 --------
portfolios = 10  #no. of EBIT groupings
CFdataComp = CFdata[complete.cases(CFdata[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}

firmsNo = length(CFdataComp[,1])
firmsList = CFdataComp[,1]
quantiles = quantile(as.numeric(CFdataComp[,inputYear+2]), c(.1,.2,.3,.4,.5,.6,.7,.8,.9,1))
quantilMatrix = matrix(c(quantiles,portfolios:1),nrow=portfolios,ncol=2)
portfolioNo = vector(mode="numeric", firmsNo)
CFdataComp = cbind(CFdataComp,portfolioNo)
for (i in 1:firmsNo) {
  EBITinputYear = as.numeric(CFdataComp[i,inputYear+2])
  if(EBITinputYear<quantilMatrix[1,1]){
    CFdataComp[i,42]=10
  } else if(EBITinputYear<quantilMatrix[2,1]){
    CFdataComp[i,42]=9
  } else if(EBITinputYear<quantilMatrix[3,1]){
    CFdataComp[i,42]=8
  } else if(EBITinputYear<quantilMatrix[4,1]){
    CFdataComp[i,42]=7
  } else if(EBITinputYear<quantilMatrix[5,1]){
    CFdataComp[i,42]=6
  } else if(EBITinputYear<quantilMatrix[6,1]){
    CFdataComp[i,42]=5
  } else if(EBITinputYear<quantilMatrix[7,1]){
    CFdataComp[i,42]=4
  } else if(EBITinputYear<quantilMatrix[8,1]){
    CFdataComp[i,42]=3
  } else if(EBITinputYear<quantilMatrix[9,1]){
    CFdataComp[i,42]=2
  } else{
    CFdataComp[i,42]=1
  }
}

CFdataNum=CFdataComp[,2:42] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
```

```
CFdataPortSum = matrix(0,nrow = portfolios,ncol = 40) #year 0 to 39 of data, no CVRs
for (i in 1:firmsNo) {
  portNum = CFdataNum[i,41]
  CFdataPortSum[portNum,]=CFdataNum[i,1:40]+CFdataPortSum[portNum,]
}
CFdataPortAvg = matrix(0,nrow = portfolios,ncol = 40) #year 0 to 39 of data, no CVRs
PortSize = vector(mode = "numeric",length = portfolios)

for (i in 1:portfolios) {
  PortSize[i] = sum(CFdataNum[,41] == i)
}
CFdataPortAvg = CFdataPortSum[,(dataStart+1):(horizonEndCol+1)]/PortSize
#forecasting
firmsNo = length(CFdataTestComp[,1])
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecastEBIT2 = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataTestComp[,(dataStart+1):(dataEnd+horizon+1)]
CFforecastEBIT2[,1:dataEnd] = CFdataTestComp[,(dataStart+1):(dataEnd+1)]
CFdataPortPer = CFdataPortAvg[,2:(horizonEndCol)]/CFdataPortAvg[,1:(horizonEndCol-1)]-1
for(i in 1:firmsNo){
  inputEBIT = CFforecastEBIT2[i,(dataEndCol)]
  if(inputEBIT<quantilMatrix[1,1]){
    CFdataTestComp[i,42]=10
  } else if(inputEBIT<quantilMatrix[2,1]){
    CFdataTestComp[i,42]=9
  } else if(inputEBIT<quantilMatrix[3,1]){
    CFdataTestComp[i,42]=8
  } else if(inputEBIT<quantilMatrix[4,1]){
    CFdataTestComp[i,42]=7
  } else if(inputEBIT<quantilMatrix[5,1]){
    CFdataTestComp[i,42]=6
  } else if(inputEBIT<quantilMatrix[6,1]){
    CFdataTestComp[i,42]=5
  } else if(inputEBIT<quantilMatrix[7,1]){
    CFdataTestComp[i,42]=4
  } else if(inputEBIT<quantilMatrix[8,1]){
    CFdataTestComp[i,42]=3
  } else if(inputEBIT<quantilMatrix[9,1]){
    CFdataTestComp[i,42]=2
  } else{
    CFdataTestComp[i,42]=1
  }
  inputPort = CFdataTestComp[i,47]
  for(j in 1:horizon){
    CFforecastEBIT2[i,(dataEndCol+j)] = round(CFforecastEBIT2[i,(dataEndCol+j-1)]*
                                (1+CFdataPortPer[inputPort,(dataEndCol+j-1)]),2)
  }
}
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastEBIT2[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF18 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF18) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF18[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF18[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF18[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF18[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF18[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
```

```
 localErrorsF18[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
 globalErrors[18,i] = round(mean(localErrorsF18[!rowSums(!is.finite(localErrorsF18)),i]),4)
}
#-------- 2.1.3 portfolio development - Industry --------
portfolios = IndUniqueLenght
CFdataCalc = cbind(CFdata,relSmallDataUnique[,4]) #4 because it takes industry no.
CFdataComp = CFdataCalc[complete.cases(CFdataCalc[,dataStart+2]),] #plus 2 because CVR and year 0

for(i in (dataStart+1):(dataEnd+horizon)){
 CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
firmsNo = length(CFdataComp[,1])
firmsList = CFdataComp[,1]
CFdataNum=CFdataComp[,2:42] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
CFdataPortSum = matrix(0,nrow = portfolios,ncol = 40) #year 0 to 39 of data, no CVRs
for (i in 1:firmsNo) {
 portNum = CFdataNum[i,41]
 CFdataPortSum[portNum,]=CFdataNum[i,1:40]+CFdataPortSum[portNum,]
}
CFdataPortAvg = matrix(0,nrow = portfolios,ncol = 40) #year 0 to 39 of data, no CVRs
PortSize = vector(mode = "numeric",length = portfolios)
for (i in 1:portfolios) {
 PortSize[i] = sum(CFdataNum[,41] == i)
}
CFdataPortAvg = CFdataPortSum[,(dataStart+1):(horizonEndCol+1)]/PortSize
#forecasting
firmsNo = length(CFdataTestComp[,1])
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecastInd = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataTestComp[,(dataStart+1):(dataEnd+horizon+1)]
CFforecastInd[,1:dataEnd] = CFdataTestComp[,(dataStart+1):(dataEnd+1)]
CFdataPortPer = CFdataPortAvg[,2:(horizonEndCol)]/CFdataPortAvg[,1:(horizonEndCol-1)]-1
for(i in 1:firmsNo){
 inputEBIT = CFforecastInd[i,(dataEndCol)]
 inputPort = CFdataTestComp[i,44]

 for(j in 1:horizon){
  CFforecastInd[i,(dataEndCol+j)] = round(CFforecastInd[i,(dataEndCol+j-1)]*
                          (1+CFdataPortPer[inputPort,(dataEndCol+j-1)]),2)
 }
}
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastInd[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF8 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF8) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
 localErrorsF8[i,1] = round(mean(CFdifference[i,]),2) # Error
 localErrorsF8[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
 localErrorsF8[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
 localErrorsF8[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
 localErrorsF8[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
 localErrorsF8[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
```

```
  globalErrors[8,i] = round(mean(localErrorsF8[!rowSums(!is.finite(localErrorsF8)),i]),4)
}
#-------- 2.1.4 portfolio development - EBIT&Industry --------
portfoliosEBIT = 5  #no. of EBIT groupings
portfoliosInd = IndUniqueLenght
portfoliosTotal = portfoliosEBIT*portfoliosInd
CFdataCalc = cbind(CFdata,relSmallDataUnique[,4]) #4 because it takes industry no.
CFdataComp = CFdataCalc[complete.cases(CFdataCalc[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
firmsNo = length(CFdataComp[,1])
firmsList = CFdataComp[,1]
quantiles = quantile(as.numeric(CFdataComp[,inputYear+2]), c(.2, .4, .6, .8, 1))
quantilMatrix = matrix(c(quantiles,5:1),nrow=5,ncol=2)
portfolioNo = vector(mode="numeric", firmsNo)
CFdataComp = cbind(CFdataComp,portfolioNo
for (i in 1:firmsNo) {
  EBITinputYear = as.numeric(CFdataComp[i,inputYear+2])
  if(EBITinputYear<quantilMatrix[1,1]){
    CFdataComp[i,43]=5
  } else if(EBITinputYear<quantilMatrix[2,1]){
    CFdataComp[i,43]=4
  } else if(EBITinputYear<quantilMatrix[3,1]){
    CFdataComp[i,43]=3
  } else if(EBITinputYear<quantilMatrix[4,1]){
    CFdataComp[i,43]=2
  } else{
    CFdataComp[i,43]=1
  }
}
CFdataNum=CFdataComp[,2:43] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
portfolioList = matrix(0,nrow=portfoliosTotal,ncol=4) #2 cols for each EBIT and Ind
portfolioList[,1] = IndustryNo
for(i in 1:portfoliosTotal){
  portfolioList[i,2] = sum(portfolioList[1:i,1]==portfolioList[i,1])
  portfolioList[i,3] = IndustryList[which(IndustryList[,3]==portfolioList[i,1]),1]
  portfolioList[i,4] = quantilMatrix[which(quantilMatrix[,2]==portfolioList[i,2]),1]
}
CFdataNum = cbind(CFdataNum,portfolioNo)
for(i in 1:firmsNo){
  CFdataNum[i,43] = which(portfolioList[,1]==CFdataNum[i,41] &
                          portfolioList[,2]==CFdataNum[i,42])
}
CFdataPortSum = matrix(0,nrow = portfoliosTotal,ncol = 40) #year 0 to 39 of data, no CVRs
for (i in 1:firmsNo) {
  portNum = CFdataNum[i,43]
  CFdataPortSum[portNum,]=CFdataNum[i,1:40]+CFdataPortSum[portNum,]
}
PortSize = vector(mode = "numeric",length = portfoliosTotal)
for (i in 1:portfoliosTotal) {
  PortSize[i] = sum(CFdataNum[,43] == i)
}
CFdataPortAvg = CFdataPortSum[,(dataStart+1):(horizonEndCol+1)]/PortSize
#forecasting
firmsNo = length(CFdataTestComp[,1])
```

```
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecastEBITind = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataTestComp[,(dataStart+1):(dataEnd+horizon+1)]
CFforecastEBITind[,1:dataEnd] = CFdataTestComp[,(dataStart+1):(dataEnd+1)]
CFdataPortPer = CFdataPortAvg[,2:(horizonEndCol)]/CFdataPortAvg[,1:(horizonEndCol-1)]-1
for(i in 1:firmsNo){
  inputEBIT = CFforecastEBITind[i,(dataEndCol)]
  CFdataTestComp[i,49] = which(portfolioList[,1]==CFdataTestComp[i,44] &
                         portfolioList[,2]==CFdataTestComp[i,47])
  inputPort = CFdataTestComp[i,49]
  for(j in 1:horizon){
    CFforecastEBITind[i,(dataEndCol+j)] = round(CFforecastEBITind[i,(dataEndCol+j-1)]*
                                (1+CFdataPortPer[inputPort,(dataEndCol+j-1)]),2)
  }
}
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastEBITind[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF9 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF9) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF9[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF9[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF9[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF9[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF9[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
  localErrorsF9[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
  globalErrors[9,i] = round(mean(localErrorsF9[!rowSums(!is.finite(localErrorsF9)),i]),4)
}

#-------- 2.1.5 portfolio development - Age --------
portfolios = AgeUniqueLengh
CFdataCalc = cbind(CFdata,relSmallDataUnique[,5]) #5 because it takes age
CFdataComp = CFdataCalc[complete.cases(CFdataCalc[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
CFdataComp = CFdataComp[complete.cases(CFdataComp[,42]),] #removes where age is NA
firmsNo = length(CFdataComp[,1])
firmsList = CFdataComp[,1]
CFdataNum = CFdataComp[,2:42] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
CFdataPortSum = matrix(0,nrow = portfolios,ncol = 40) #year 0 to 39 of data, no CVRs
for (i in 1:firmsNo) {
  portNum = CFdataNum[i,41]
  CFdataPortSum[portNum,]=CFdataNum[i,1:40]+CFdataPortSum[portNum,]
}
CFdataPortAvg = matrix(0,nrow = portfolios,ncol = 40) #year 0 to 39 of data, no CVRs
PortSize = vector(mode = "numeric",length = portfolios)
for (i in 1:portfolios) {
  PortSize[i] = sum(CFdataNum[,41] == i)
}
CFdataPortAvg = CFdataPortSum[,(dataStart+1):(horizonEndCol+1)]/PortSize
#forecasting
firmsNo = length(CFdataTestComp[,1])
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
```

```
CFforecastAge = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))

CFactual = CFdataTestComp[,(dataStart+1):(dataEnd+horizon+1)]
CFforecastAge[,1:dataEnd] = CFdataTestComp[,(dataStart+1):(dataEnd+1)]
CFdataPortPer = CFdataPortAvg[,2:(horizonEndCol)]/CFdataPortAvg[,1:(horizonEndCol-1)]-1
for(i in 1:firmsNo){
 inputEBIT = CFforecastAge[i,(dataEndCol)]
 inputPort = CFdataTestComp[i,45]
 for(j in 1:horizon){
  CFforecastAge[i,(dataEndCol+j)] = round(CFforecastAge[i,(dataEndCol+j-1)]*
                    (1+CFdataPortPer[inputPort,(dataEndCol+j-1)]),2)
 }
}
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastAge[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF10 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF10) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
 localErrorsF10[i,1] = round(mean(CFdifference[i,]),2) # Error
 localErrorsF10[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
 localErrorsF10[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
 localErrorsF10[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
 localErrorsF10[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
 localErrorsF10[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
 globalErrors[10,i] = round(mean(localErrorsF10[!rowSums(!is.finite(localErrorsF10)),i]),4)
}
#-------- 2.1.6 portfolio development - Balance --------
portfolios = 5
quantilesBal = quantile(as.numeric(Balancedata[,inputYear+2]), c(.2, .4, .6, .8, 1), na.rm = TRUE)
quantilMatrixBal = matrix(c(quantilesBal,5:1),nrow=5,ncol=2)
portfolioNo = vector(mode="numeric", firms)
CFdataCalc = cbind(CFdata,portfolioNo)
for (i in 1:firms) {
 BalInputYear = as.numeric(Balancedata[i,inputYear+2])
 if(is.na(BalInputYear)){
  CFdataCalc[i,42]=NA
 } else if(BalInputYear<quantilMatrixBal[1,1]){
  CFdataCalc[i,42]=5
 } else if(BalInputYear<quantilMatrixBal[2,1]){
  CFdataCalc[i,42]=4
 } else if(BalInputYear<quantilMatrixBal[3,1]){
  CFdataCalc[i,42]=3
 } else if(BalInputYear<quantilMatrixBal[4,1]){
  CFdataCalc[i,42]=2
 } else{
  CFdataCalc[i,42]=1
 }
}
CFdataComp = CFdataCalc[complete.cases(CFdataCalc[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
 CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
CFdataComp = CFdataComp[complete.cases(CFdataComp[,42]),] #removing NA rows from portfolioNo
firmsNo = length(CFdataComp[,1])
firmsList = CFdataComp[,1]
CFdataNum=CFdataComp[,2:42] #Turn the data into only numbers
```

```
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
CFdataPortSum = matrix(0,nrow = portfolios,ncol = 40) #year 0 to 39 of data, no CVRs
for (i in 1:firmsNo) {
  portNum = CFdataNum[i,41]
  CFdataPortSum[portNum,]=CFdataNum[i,1:40]+CFdataPortSum[portNum,]
}
CFdataPortAvg = matrix(0,nrow = portfolios,ncol = 40) #year 0 to 39 of data, no CVRs
PortSize = vector(mode = "numeric",length = portfolios)

for (i in 1:portfolios) {
  PortSize[i] = sum(CFdataNum[,41] == i)
}
CFdataPortAvg = CFdataPortSum[,(dataStart+1):(horizonEndCol+1)]/PortSize
#forecasting
firmsNo = length(CFdataTestComp[,1])
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecastBal = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataTestComp[,(dataStart+1):(dataEnd+horizon+1)]
CFforecastBal[,1:dataEnd] = CFdataTestComp[,(dataStart+1):(dataEnd+1)]
CFdataPortPer = CFdataPortAvg[,2:(horizonEndCol)]/CFdataPortAvg[,1:(horizonEndCol-1)]-1
for(i in 1:firmsNo){
  inputEBIT = CFforecastBal[i,(dataEndCol)]
  inputBal = CFdataTestComp[i,42]
  if(inputBal<quantilMatrixBal[1,1]){
    CFdataTestComp[i,46]=5
  } else if(inputBal<quantilMatrixBal[2,1]){
    CFdataTestComp[i,46]=4
  } else if(inputBal<quantilMatrixBal[3,1]){
    CFdataTestComp[i,46]=3
  } else if(inputBal<quantilMatrixBal[4,1]){
    CFdataTestComp[i,46]=2
  } else{
    CFdataTestComp[i,46]=1
  }
  inputPort = CFdataTestComp[i,46]
  for(j in 1:horizon){
    CFforecastBal[i,(dataEndCol+j)] = round(CFforecastBal[i,(dataEndCol+j-1)]*
                                (1+CFdataPortPer[inputPort,(dataEndCol+j-1)]),2)
  }
}
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastBal[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF11 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF11) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF11[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF11[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF11[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF11[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF11[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
  localErrorsF11[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
  globalErrors[11,i] = round(mean(localErrorsF11[!rowSums(!is.finite(localErrorsF11)),i]),4)
}
#-------- 2.1.7 portfolio development - EBIT&Balance --------
portfoliosEBIT = 5  #no. of EBIT groupings
```

```
portfoliosBal = 5   #no. of Balance groupings
portfoliosTotal = portfoliosEBIT*portfoliosBal
quantilesBal = quantile(as.numeric(Balancedata[,inputYear+2]), c(.2, .4, .6, .8, 1), na.rm = TRUE)
quantilMatrixBal = matrix(c(quantilesBal,5:1),nrow=5,ncol=2)
portfolioNo = vector(mode="numeric", firms)
CFdataCalc = cbind(CFdata,portfolioNo)
for (i in 1:firms) {
  BalInputYear = as.numeric(Balancedata[i,inputYear+2])
  if(is.na(BalInputYear)){
    CFdataCalc[i,42]=NA
  } else if(BalInputYear<quantilMatrixBal[1,1]){
    CFdataCalc[i,42]=5
  } else if(BalInputYear<quantilMatrixBal[2,1]){
    CFdataCalc[i,42]=4
  } else if(BalInputYear<quantilMatrixBal[3,1]){
    CFdataCalc[i,42]=3
  } else if(BalInputYear<quantilMatrixBal[4,1]){
    CFdataCalc[i,42]=2
  } else{
    CFdataCalc[i,42]=1
  }
}
CFdataComp = CFdataCalc[complete.cases(CFdataCalc[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
CFdataComp = CFdataComp[complete.cases(CFdataComp[,42]),] #removing NA rows from BalportfolioNo
firmsNo = length(CFdataComp[,1])
firmsList = CFdataComp[,1]
quantilesEBIT = quantile(as.numeric(CFdataComp[,inputYear+2]), c(.2, .4, .6, .8, 1))
quantilMatrixEBIT = matrix(c(quantilesEBIT,5:1),nrow=5,ncol=2)
portfolioNo = vector(mode="numeric", firmsNo)
CFdataComp = cbind(CFdataComp,portfolioNo)
for (i in 1:firmsNo) {
  EBITinputYear = as.numeric(CFdataComp[i,inputYear+2])
  if(EBITinputYear<quantilMatrixEBIT[1,1]){
    CFdataComp[i,43]=5
  } else if(EBITinputYear<quantilMatrixEBIT[2,1]){
    CFdataComp[i,43]=4
  } else if(EBITinputYear<quantilMatrixEBIT[3,1]){
    CFdataComp[i,43]=3
  } else if(EBITinputYear<quantilMatrixEBIT[4,1]){
    CFdataComp[i,43]=2
  } else{
    CFdataComp[i,43]=1
  }
}
CFdataNum=CFdataComp[,2:43] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
portfolioList = matrix(0,nrow=portfoliosTotal,ncol=4) #2 cols for each EBIT and Ind
portfolioList[,1] = quantilMatrixBal[length(quantilMatrixBal[,2]):1,2]
for(i in 1:portfoliosTotal){
  portfolioList[i,2] = sum(portfolioList[1:i,1]==portfolioList[i,1])
  portfolioList[i,3] = quantilMatrixBal[which(quantilMatrixBal[,2]==portfolioList[i,1]),1]
  portfolioList[i,4] = quantilMatrixEBIT[which(quantilMatrixEBIT[,2]==portfolioList[i,2]),1]
}
CFdataNum = cbind(CFdataNum,portfolioNo)
```

```
for(i in 1:firmsNo){
  CFdataNum[i,43] = which(portfolioList[,1]==CFdataNum[i,41] &
                    portfolioList[,2]==CFdataNum[i,42])
}
CFdataPortSum = matrix(0,nrow = portfoliosTotal,ncol = 40) #year 0 to 39 of data, no CVRs
for (i in 1:firmsNo) {
  portNum = CFdataNum[i,43]
  CFdataPortSum[portNum,]=CFdataNum[i,1:40]+CFdataPortSum[portNum,]
}
PortSize = vector(mode = "numeric",length = portfoliosTotal)
for (i in 1:portfoliosTotal) {
  PortSize[i] = sum(CFdataNum[,43] == i)
}
CFdataPortAvg = CFdataPortSum[,(dataStart+1):(horizonEndCol+1)]/PortSize
#forecasting
firmsNo = length(CFdataTestComp[,1])
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecastEBITbal = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataTestComp[,(dataStart+1):(dataEnd+horizon+1)]
CFforecastEBITbal[,1:dataEnd] = CFdataTestComp[,(dataStart+1):(dataEnd+1)]
CFdataPortPer = CFdataPortAvg[,2:(horizonEndCol)]/CFdataPortAvg[,1:(horizonEndCol-1)]-1
for(i in 1:firmsNo){
  inputEBIT = CFforecastEBITbal[i,(dataEndCol)]
  CFdataTestComp[i,50] = which(portfolioList[,1]==CFdataTestComp[i,46] &
                         portfolioList[,2]==CFdataTestComp[i,47])
  inputPort = CFdataTestComp[i,50]
  for(j in 1:horizon){
    CFforecastEBITbal[i,(dataEndCol+j)] = round(CFforecastEBITbal[i,(dataEndCol+j-1)]*
                                  (1+CFdataPortPer[inputPort,(dataEndCol+j-1)]),2)
  }
}
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastEBITbal[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF12 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF12) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")

for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF12[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF12[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF12[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF12[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF12[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
  localErrorsF12[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
  globalErrors[12,i] = round(mean(localErrorsF12[!rowSums(!is.finite(localErrorsF12)),i]),4)
}
#-------- 2.1.8 portfolio development - Growth10 --------
portfolios = 10
CFdataComp = CFdata[complete.cases(CFdata[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
CFdataNum = CFdataComp[,2:41] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
growthData = (CFdataNum[,2:40]/CFdataNum[,1:39]-1)
firmsNo = length(CFdataNum[,1])
```

```
cols = length(growthData[1,])
for(i in 1:firmsNo){
 for(j in 1:cols){
  if( growthData[i,j]=="Inf" | growthData[i,j]=="-Inf"){
   growthData[i,j]=NA
  }
 }
}
quantilesGrow = quantile(as.numeric(growthData[,inputYear+1]),c(.1,.2,.3,.4,.5,.6,.7,.8,.9,1), na.rm = TRUE)
quantilMatrixGrow = matrix(c(quantilesGrow,portfolios:1),nrow=portfolios,ncol=2)
portfolioNo = vector(mode="numeric", firmsNo)
CFdataNum = cbind(CFdataNum,portfolioNo)
for (i in 1:firmsNo) {
 growthInputYear = as.numeric(growthData[i,inputYear+1]) #+1 because there is year zero
 if(is.na(growthInputYear)){
  CFdataNum[i,41]=NA
 } else if(growthInputYear<quantilMatrixGrow[1,1]){
  CFdataNum[i,41]=10
 } else if(growthInputYear<quantilMatrixGrow[2,1]){
  CFdataNum[i,41]=9
 } else if(growthInputYear<quantilMatrixGrow[3,1]){
  CFdataNum[i,41]=8
 } else if(growthInputYear<quantilMatrixGrow[4,1]){
  CFdataNum[i,41]=7
 } else if(growthInputYear<quantilMatrixGrow[5,1]){
  CFdataNum[i,41]=6
 } else if(growthInputYear<quantilMatrixGrow[6,1]){
  CFdataNum[i,41]=5
 } else if(growthInputYear<quantilMatrixGrow[7,1]){
  CFdataNum[i,41]=4
 } else if(growthInputYear<quantilMatrixGrow[8,1]){
  CFdataNum[i,41]=3
 } else if(growthInputYear<quantilMatrixGrow[9,1]){
  CFdataNum[i,41]=2
 } else{
  CFdataNum[i,41]=1
 }
}
CFdataNum = CFdataNum[complete.cases(CFdataNum[,41]),] #removing NA rows from portfolioNo
firmsNo = length(CFdataNum[,1])
CFdataPortSum = matrix(0,nrow = portfolios,ncol = 40) #year 0 to 39 of data, no CVRs
for (i in 1:firmsNo) {
 portNum = CFdataNum[i,41]
 CFdataPortSum[portNum,]=CFdataNum[i,1:40]+CFdataPortSum[portNum,]
}
CFdataPortAvg = matrix(0,nrow = portfolios,ncol = 40) #year 0 to 39 of data, no CVRs
PortSize = vector(mode = "numeric",length = portfolios)
for (i in 1:portfolios) {
 PortSize[i] = sum(CFdataNum[,41] == i)
}
CFdataPortAvg = CFdataPortSum[,(dataStart+1):(horizonEndCol+1)]/PortSize
#forecasting
firmsNo = length(CFdataTestComp[,1])
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecastGrow2 = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataTestComp[,(dataStart+1):(dataEnd+horizon+1)]
CFforecastGrow2[,1:dataEnd] = CFdataTestComp[,(dataStart+1):(dataEnd+1)]
CFdataPortPer = CFdataPortAvg[,2:(horizonEndCol)]/CFdataPortAvg[,1:(horizonEndCol-1)]-1
```

```
for(i in 1:firmsNo){
  inputEBIT = CFforecastGrow2[i,(dataEndCol)]
  inputGrowth = CFdataTestComp[i,43]
  if(inputGrowth<quantilMatrixGrow[1,1]){
    CFdataTestComp[i,48]=10
  } else if(inputGrowth<quantilMatrixGrow[2,1]){
    CFdataTestComp[i,48]=9
  } else if(inputGrowth<quantilMatrixGrow[3,1]){
    CFdataTestComp[i,48]=8
  } else if(inputGrowth<quantilMatrixGrow[4,1]){
    CFdataTestComp[i,48]=7
  } else if(inputGrowth<quantilMatrixGrow[5,1]){
    CFdataTestComp[i,48]=6
  } else if(inputGrowth<quantilMatrixGrow[6,1]){
    CFdataTestComp[i,48]=5
  } else if(inputGrowth<quantilMatrixGrow[7,1]){
    CFdataTestComp[i,48]=4
  } else if(inputGrowth<quantilMatrixGrow[8,1]){
    CFdataTestComp[i,48]=3
  } else if(inputGrowth<quantilMatrixGrow[9,1]){
    CFdataTestComp[i,48]=2
  } else{
    CFdataTestComp[i,48]=1
  }
  inputPort = CFdataTestComp[i,48]
  for(j in 1:horizon){
    CFforecastGrow2[i,(dataEndCol+j)] = round(CFforecastGrow2[i,(dataEndCol+j-1)]*
                                (1+CFdataPortPer[inputPort,(dataEndCol+j-1)]),2)
  }
}
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastGrow2[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF19 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF19) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF19[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF19[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF19[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF19[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF19[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
  localErrorsF19[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
  globalErrors[19,i] = round(mean(localErrorsF19[!rowSums(!is.finite(localErrorsF19)),i]),4)
}
#-------- 2.1.9 portfolio development - Growth5 --------
portfolios = 5
CFdataComp = CFdata[complete.cases(CFdata[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
CFdataNum = CFdataComp[,2:41] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
growthData = (CFdataNum[,2:40]/CFdataNum[,1:39]-1)
firmsNo = length(CFdataNum[,1])
cols = length(growthData[1,])
for(i in 1:firmsNo){
```

```
 for(j in 1:cols){
  if( growthData[i,j]=="Inf" || growthData[i,j]=="-Inf"){
   growthData[i,j]=NA
  }
 }
}
quantilesGrow = quantile(as.numeric(growthData[,inputYear+1]), c(.2, .4, .6, .8, 1), na.rm = TRUE)
quantilMatrixGrow = matrix(c(quantilesGrow,5:1),nrow=5,ncol=2) #+1 because there is year zero
portfolioNo = vector(mode="numeric", firmsNo)
CFdataNum = cbind(CFdataNum,portfolioNo)
for (i in 1:firmsNo) {
 growthInputYear = as.numeric(growthData[i,inputYear+1]) #+1 because there is year zero
 if(is.na(growthInputYear)){
  CFdataNum[i,41]=NA
 } else if(growthInputYear<quantilMatrixGrow[1,1]){
  CFdataNum[i,41]=5
 } else if(growthInputYear<quantilMatrixGrow[2,1]){
  CFdataNum[i,41]=4
 } else if(growthInputYear<quantilMatrixGrow[3,1]){
  CFdataNum[i,41]=3
 } else if(growthInputYear<quantilMatrixGrow[4,1]){
  CFdataNum[i,41]=2
 } else{
  CFdataNum[i,41]=1
 }
}
CFdataNum = CFdataNum[complete.cases(CFdataNum[,41]),] #removing NA rows from portfolioNo
firmsNo = length(CFdataNum[,1])
CFdataPortSum = matrix(0,nrow = portfolios,ncol = 40) #year 0 to 39 of data, no CVRs
for (i in 1:firmsNo) {
 portNum = CFdataNum[i,41]
 CFdataPortSum[portNum,]=CFdataNum[i,1:40]+CFdataPortSum[portNum,]
}
CFdataPortAvg = matrix(0,nrow = portfolios,ncol = 40) #year 0 to 39 of data, no CVRs
PortSize = vector(mode = "numeric",length = portfolios)
for (i in 1:portfolios) {
 PortSize[i] = sum(CFdataNum[,41] == i)
}
CFdataPortAvg = CFdataPortSum[,(dataStart+1):(horizonEndCol+1)]/PortSize
#forecasting
firmsNo = length(CFdataTestComp[,1])
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecastGrow = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataTestComp[,(dataStart+1):(dataEnd+horizon+1)]
CFforecastGrow[,1:dataEnd] = CFdataTestComp[,(dataStart+1):(dataEnd+1)]
CFdataPortPer = CFdataPortAvg[,2:(horizonEndCol)]/CFdataPortAvg[,1:(horizonEndCol-1)]-1
for(i in 1:firmsNo){
 inputEBIT = CFforecastGrow[i,(dataEndCol)]
 inputGrowth = CFdataTestComp[i,43]
 if(inputGrowth<quantilMatrixGrow[1,1]){
  CFdataTestComp[i,48]=5
 } else if(inputGrowth<quantilMatrixGrow[2,1]){
  CFdataTestComp[i,48]=4
 } else if(inputGrowth<quantilMatrixGrow[3,1]){
  CFdataTestComp[i,48]=3
 } else if(inputGrowth<quantilMatrixGrow[4,1]){
  CFdataTestComp[i,48]=2
 } else{
```

```
   CFdataTestComp[i,48]=1
  }
  inputPort = CFdataTestComp[i,48]
  for(j in 1:horizon){
   CFforecastGrow[i,(dataEndCol+j)] = round(CFforecastGrow[i,(dataEndCol+j-1)]*
                                (1+CFdataPortPer[inputPort,(dataEndCol+j-1)]),2)
  }
}
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastGrow[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF13 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF13) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF13[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF13[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF13[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF13[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF13[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
  localErrorsF13[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
  globalErrors[13,i] = round(mean(localErrorsF13[!rowSums(!is.finite(localErrorsF13)),i]),4)
}
#-------- 2.1.10 portfolio development - EBIT&Growth --------
portfoliosEBIT = 5  #no. of EBIT groupings
portfoliosGrowth = 5   #no. of Balance groupings
portfoliosTotal = portfoliosEBIT*portfoliosGrowth
CFdataComp = CFdata[complete.cases(CFdata[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
CFdataNum = CFdataComp[,2:41] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
growthData = (CFdataNum[,2:40]/CFdataNum[,1:39]-1)
firmsNo = length(CFdataNum[,1])
cols = length(growthData[1,])
for(i in 1:firmsNo){
  for(j in 1:cols){
    if( growthData[i,j]=="Inf" || growthData[i,j]=="-Inf"){
      growthData[i,j]=NA
    }
  }
}
quantilesGrow = quantile(as.numeric(growthData[,inputYear+1]), c(.2, .4, .6, .8, 1), na.rm = TRUE)
quantilMatrixGrow = matrix(c(quantilesGrow,5:1),nrow=5,ncol=2)
portfolioNo = vector(mode="numeric", firmsNo)
CFdataNum = cbind(CFdataNum,portfolioNo)
for (i in 1:firmsNo) {
  growthInputYear = as.numeric(growthData[i,inputYear+1])
  if(is.na(growthInputYear)){
    CFdataNum[i,41]=NA
  } else if(growthInputYear<quantilMatrixGrow[1,1]){
    CFdataNum[i,41]=5
  } else if(growthInputYear<quantilMatrixGrow[2,1]){
    CFdataNum[i,41]=4
  } else if(growthInputYear<quantilMatrixGrow[3,1]){
    CFdataNum[i,41]=3
```

```
 } else if(growthInputYear<quantilMatrixGrow[4,1]){
  CFdataNum[i,41]=2
 } else{
  CFdataNum[i,41]=1
 }
}
firmsNo = length(CFdataNum[,1])
quantilesEBIT = quantile(as.numeric(CFdataNum[,inputYear+1]), c(.2, .4, .6, .8, 1))
quantilMatrixEBIT = matrix(c(quantilesEBIT,5:1),nrow=5,ncol=2)
portfolioNo = vector(mode="numeric", firmsNo)
CFdataNum = cbind(CFdataNum,portfolioNo)
for (i in 1:firmsNo) {
 EBITinputYear = as.numeric(CFdataNum[i,inputYear+1])
 if(EBITinputYear<quantilMatrixEBIT[1,1]){
  CFdataNum[i,42]=5
 } else if(EBITinputYear<quantilMatrixEBIT[2,1]){
  CFdataNum[i,42]=4
 } else if(EBITinputYear<quantilMatrixEBIT[3,1]){
  CFdataNum[i,42]=3
 } else if(EBITinputYear<quantilMatrixEBIT[4,1]){
  CFdataNum[i,42]=2
 } else{
  CFdataNum[i,42]=1
 }
}
portfolioList = matrix(0,nrow=portfoliosTotal,ncol=4) #2 cols for each EBIT and Ind
portfolioList[,1] = quantilMatrixGrow[length(quantilMatrixGrow[,2]):1,2]
for(i in 1:portfoliosTotal){
 portfolioList[i,2] = sum(portfolioList[1:i,1]==portfolioList[i,1])
 portfolioList[i,3] = quantilMatrixGrow[which(quantilMatrixGrow[,2]==portfolioList[i,1]),1]
 portfolioList[i,4] = quantilMatrixEBIT[which(quantilMatrixEBIT[,2]==portfolioList[i,2]),1]
}
CFdataNum = cbind(CFdataNum,portfolioNo)
CFdataNum = CFdataNum[complete.cases(CFdataNum[,41]),] #removing NA rows from portfolioNo
firmsNo = length(CFdataNum[,1])
for(i in 1:firmsNo){
 CFdataNum[i,43] = which(portfolioList[,1]==CFdataNum[i,41] &
                  portfolioList[,2]==CFdataNum[i,42])
}
CFdataPortSum = matrix(0,nrow = portfoliosTotal,ncol = 40) #year 0 to 39 of data, no CVRs
for (i in 1:firmsNo) {
 portNum = CFdataNum[i,43]
 CFdataPortSum[portNum,]=CFdataNum[i,1:40]+CFdataPortSum[portNum,]
}
PortSize = vector(mode = "numeric",length = portfoliosTotal)
for (i in 1:portfoliosTotal) {
 PortSize[i] = sum(CFdataNum[,43] == i)
}
CFdataPortAvg = CFdataPortSum[,(dataStart+1):(horizonEndCol+1)]/PortSize
#forecasting
firmsNo = length(CFdataTestComp[,1])
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecastEBITgrow = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataTestComp[,(dataStart+1):(dataEnd+horizon+1)]
CFforecastEBITgrow[,1:dataEnd] = CFdataTestComp[,(dataStart+1):(dataEnd+1)]
CFdataPortPer = CFdataPortAvg[,2:(horizonEndCol)]/CFdataPortAvg[,1:(horizonEndCol-1)]-1
for(i in 1:firmsNo){
 inputEBIT = CFforecastEBITgrow[i,(dataEndCol)]
```

```
  CFdataTestComp[i,51] = which(portfolioList[,1]==CFdataTestComp[i,48] &
                         portfolioList[,2]==CFdataTestComp[i,47])
  inputPort = CFdataTestComp[i,51]
  for(j in 1:horizon){
    CFforecastEBITgrow[i,(dataEndCol+j)] = round(CFforecastEBITgrow[i,(dataEndCol+j-1)]*
                                  (1+CFdataPortPer[inputPort,(dataEndCol+j-1)]),2)
  }
}
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastEBITgrow[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF14 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF14) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF14[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF14[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF14[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF14[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF14[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
  localErrorsF14[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
  globalErrors[14,i] = round(mean(localErrorsF14[!rowSums(!is.finite(localErrorsF14)),i]),4)
}
#-------- 2.1.11 portfolio development - EBIT&Growth5&Balance ----------
portfoliosEBIT = 5  #no. of EBIT groupings
portfoliosBal = 5   #no. of Balance groupings
portfoliosGrowth = 5 #no. of growth groupings
portfoliosTotal = portfoliosEBIT*portfoliosBal*portfoliosGrowth
quantilesBal = quantile(as.numeric(Balancedata[,inputYear+2]), c(.2, .4, .6, .8, 1), na.rm = TRUE)
quantilMatrixBal = matrix(c(quantilesBal,5:1),nrow=5,ncol=2)
portfolioNo = vector(mode="numeric", firms)
CFdataCalc = cbind(CFdata,portfolioNo)
for (i in 1:firms) {
  BalInputYear = as.numeric(Balancedata[i,inputYear+2])
  if(is.na(BalInputYear)){
    CFdataCalc[i,42]=NA
  } else if(BalInputYear<quantilMatrixBal[1,1]){
    CFdataCalc[i,42]=5
  } else if(BalInputYear<quantilMatrixBal[2,1]){
    CFdataCalc[i,42]=4
  } else if(BalInputYear<quantilMatrixBal[3,1]){
    CFdataCalc[i,42]=3
  } else if(BalInputYear<quantilMatrixBal[4,1]){
    CFdataCalc[i,42]=2
  } else{
    CFdataCalc[i,42]=1
  }
}
CFdataComp = CFdataCalc[complete.cases(CFdataCalc[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
CFdataComp = CFdataComp[complete.cases(CFdataComp[,42]),] #removing NA rows from BalportfolioNo
CFdataNum = CFdataComp[,2:42] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
growthData = (CFdataNum[,2:40]/CFdataNum[,1:39]-1)
```

```
firmsNo = length(CFdataNum[,1])
cols = length(growthData[1,])
for(i in 1:firmsNo){
 for(j in 1:cols){
  if( growthData[i,j]=="Inf" || growthData[i,j]=="-Inf"){
   growthData[i,j]=NA
  }
 }
}
quantilesGrow = quantile(as.numeric(growthData[,inputYear+1]), c(.2, .4, .6, .8, 1), na.rm = TRUE)
quantilMatrixGrow = matrix(c(quantilesGrow,5:1),nrow=5,ncol=2)
portfolioNo = vector(mode="numeric", firmsNo)
CFdataNum = cbind(CFdataNum,portfolioNo)
for (i in 1:firmsNo) {
 growthInputYear = as.numeric(growthData[i,inputYear+1])
 if(is.na(growthInputYear)){
  CFdataNum[i,42]=NA
 } else if(growthInputYear<quantilMatrixGrow[1,1]){
  CFdataNum[i,42]=5
 } else if(growthInputYear<quantilMatrixGrow[2,1]){
  CFdataNum[i,42]=4
 } else if(growthInputYear<quantilMatrixGrow[3,1]){
  CFdataNum[i,42]=3
 } else if(growthInputYear<quantilMatrixGrow[4,1]){
  CFdataNum[i,42]=2
 } else{
  CFdataNum[i,42]=1
 }
}
quantilesEBIT = quantile(as.numeric(CFdataNum[,inputYear+1]), c(.2, .4, .6, .8, 1))
quantilMatrixEBIT = matrix(c(quantilesEBIT,5:1),nrow=5,ncol=2)
portfolioNo = vector(mode="numeric", firmsNo)
CFdataNum = cbind(CFdataNum,portfolioNo)
for (i in 1:firmsNo) {
 EBITinputYear = as.numeric(CFdataNum[i,inputYear+1])
 if(EBITinputYear<quantilMatrixEBIT[1,1]){
  CFdataNum[i,43]=5
 } else if(EBITinputYear<quantilMatrixEBIT[2,1]){
  CFdataNum[i,43]=4
 } else if(EBITinputYear<quantilMatrixEBIT[3,1]){
  CFdataNum[i,43]=3
 } else if(EBITinputYear<quantilMatrixEBIT[4,1]){
  CFdataNum[i,43]=2
 } else{
  CFdataNum[i,43]=1
 }
}
portfolioList = matrix(0,nrow=portfoliosTotal,ncol=6) #2 cols for each EBIT, Bal and growth
portfolioList[,1] = quantilMatrixBal[length(quantilMatrixBal[,2]):1,2]
growthPorts = matrix(data=0,nrow = (portfoliosTotal/portfoliosGrowth),ncol = portfoliosGrowth)
growthPorts[,1] = 1
growthVector = growthPorts[,1]
for(i in 2:portfoliosGrowth){
 growthPorts[,i] = i
 growthVector = c(growthVector,growthPorts[,i])
}
for(i in 1:portfoliosTotal){
 portfolioList[i,2] = growthVector[i]
```

```
  portfolioList[i,3] = sum(portfolioList[1:i,1]==portfolioList[i,1] &
                           portfolioList[1:i,2]==portfolioList[i,2])

  portfolioList[i,4] = quantilMatrixBal[which(quantilMatrixGrow[,2]==portfolioList[i,1]),1]
  portfolioList[i,5] = quantilMatrixGrow[which(quantilMatrixGrow[,2]==portfolioList[i,2]),1]
  portfolioList[i,6] = quantilMatrixEBIT[which(quantilMatrixGrow[,2]==portfolioList[i,3]),1]
}
CFdataNum = cbind(CFdataNum,portfolioNo)
CFdataNum = CFdataNum[complete.cases(CFdataNum[,42]),] #removing NA rows from portfolioNo
firmsNo = length(CFdataNum[,1])
for(i in 1:firmsNo){
  CFdataNum[i,44] = which(portfolioList[,1]==CFdataNum[i,41] &
                          portfolioList[,2]==CFdataNum[i,42] &
                          portfolioList[,3]==CFdataNum[i,43])
}
CFdataPortSum = matrix(0,nrow = portfoliosTotal,ncol = 40) #year 0 to 39 of data, no CVRs
for (i in 1:firmsNo) {
  portNum = CFdataNum[i,44]
  CFdataPortSum[portNum,]=CFdataNum[i,1:40]+CFdataPortSum[portNum,]
}
PortSize = vector(mode = "numeric",length = portfoliosTotal)
for (i in 1:portfoliosTotal) {
  PortSize[i] = sum(CFdataNum[,44] == i)
}
CFdataPortAvg = CFdataPortSum[,(dataStart+1):(horizonEndCol+1)]/PortSize
#forecasting
firmsNo = length(CFdataTestComp[,1])
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecastEBITbalGrow = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataTestComp[,(dataStart+1):(dataEnd+horizon+1)]
CFforecastEBITbalGrow[,1:dataEnd] = CFdataTestComp[,(dataStart+1):(dataEnd+1)]
CFdataPortPer = CFdataPortAvg[,2:(horizonEndCol)]/CFdataPortAvg[,1:(horizonEndCol-1)]-1
for(i in 1:firmsNo){
  inputEBIT = CFforecastEBITbalGrow[i,(dataEndCol)]
  CFdataTestComp[i,53] = which(portfolioList[,1]==CFdataTestComp[i,46] &
                           portfolioList[,2]==CFdataTestComp[i,48]&
                           portfolioList[,3]==CFdataTestComp[i,47])

  inputPort = CFdataTestComp[i,53]
  for(j in 1:horizon){
    CFforecastEBITbalGrow[i,(dataEndCol+j)] = round(CFforecastEBITbalGrow[i,(dataEndCol+j-1)]*
                                      (1+CFdataPortPer[inputPort,(dataEndCol+j-1)]),2)
  }
}
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastEBITbalGrow[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF15 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF15) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF15[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF15[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF15[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF15[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF15[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
  localErrorsF15[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
  globalErrors[15,i] = round(mean(localErrorsF15[!rowSums(!is.finite(localErrorsF15)),i]),4)
```

```
}
#-------- 2.1.12 portfolio development - EBIT&Growth&Industry ----------
portfoliosEBIT = 5  #no. of EBIT groupings
portfoliosInd = IndUniqueLenght
portfoliosGrowth = 5 #no. of growth groupings
portfoliosTotal = portfoliosEBIT*portfoliosInd*portfoliosGrowth
CFdataCalc = cbind(CFdata,relSmallDataUnique[,4]) #4 because it takes industry no.
CFdataComp = CFdataCalc[complete.cases(CFdataCalc[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
CFdataNum = CFdataComp[,2:42] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
growthData = (CFdataNum[,2:40]/CFdataNum[,1:39]-1)
firmsNo = length(CFdataNum[,1])
cols = length(growthData[1,])
for(i in 1:firmsNo){
  for(j in 1:cols){
    if( growthData[i,j]=="Inf" || growthData[i,j]=="-Inf"){
      growthData[i,j]=NA
    }
  }
}
quantilesGrow = quantile(as.numeric(growthData[,inputYear+1]), c(.2, .4, .6, .8, 1), na.rm = TRUE)
quantilMatrixGrow = matrix(c(quantilesGrow,5:1),nrow=5,ncol=2)
portfolioNo = vector(mode="numeric", firmsNo)
CFdataNum = cbind(CFdataNum,portfolioNo)
for (i in 1:firmsNo) {
  growthInputYear = as.numeric(growthData[i,inputYear+1])
  if(is.na(growthInputYear)){
    CFdataNum[i,42]=NA
  } else if(growthInputYear<quantilMatrixGrow[1,1]){
    CFdataNum[i,42]=5
  } else if(growthInputYear<quantilMatrixGrow[2,1]){
    CFdataNum[i,42]=4
  } else if(growthInputYear<quantilMatrixGrow[3,1]){
    CFdataNum[i,42]=3
  } else if(growthInputYear<quantilMatrixGrow[4,1]){
    CFdataNum[i,42]=2
  } else{
    CFdataNum[i,42]=1
  }
}
firmsNo = length(CFdataNum[,1])
quantilesEBIT = quantile(as.numeric(CFdataNum[,inputYear+1]), c(.2, .4, .6, .8, 1))
quantilMatrixEBIT = matrix(c(quantilesEBIT,5:1),nrow=5,ncol=2)
portfolioNo = vector(mode="numeric", firmsNo)
CFdataNum = cbind(CFdataNum,portfolioNo)
for (i in 1:firmsNo) {
  EBITinputYear = as.numeric(CFdataNum[i,inputYear+1])
  if(EBITinputYear<quantilMatrixEBIT[1,1]){
    CFdataNum[i,43]=5
  } else if(EBITinputYear<quantilMatrixEBIT[2,1]){
    CFdataNum[i,43]=4
  } else if(EBITinputYear<quantilMatrixEBIT[3,1]){
    CFdataNum[i,43]=3
  } else if(EBITinputYear<quantilMatrixEBIT[4,1]){
```

```
   CFdataNum[i,43]=2
  } else{
   CFdataNum[i,43]=1
  }
}
portfolioList = matrix(0,nrow=portfoliosTotal,ncol=6) #2 cols for each EBIT, Ind and growth
portfolioList[,1] = IndustryNo
growthPorts = matrix(data=0,nrow = (portfoliosTotal/portfoliosGrowth),ncol = portfoliosGrowth)
growthPorts[,1] = 1
growthVector = growthPorts[,1]
for(i in 2:portfoliosGrowth){
 growthPorts[,i] = i
 growthVector = c(growthVector,growthPorts[,i])
}
for(i in 1:portfoliosTotal){
 portfolioList[i,2] = growthVector[i]
 portfolioList[i,3] = sum(portfolioList[1:i,1]==portfolioList[i,1] &
                          portfolioList[1:i,2]==portfolioList[i,2])

 portfolioList[i,4] = IndustryList[which(IndustryList[,3]==portfolioList[i,1]),1]
 portfolioList[i,5] = quantilMatrixGrow[which(quantilMatrixGrow[,2]==portfolioList[i,2]),1]
 portfolioList[i,6] = quantilMatrixEBIT[which(quantilMatrixGrow[,2]==portfolioList[i,3]),1]
}

CFdataNum = cbind(CFdataNum,portfolioNo)
CFdataNum = CFdataNum[complete.cases(CFdataNum[,42]),] #removing NA rows from portfolioNo
firmsNo = length(CFdataNum[,1])
for(i in 1:firmsNo){
 CFdataNum[i,44] = which(portfolioList[,1]==CFdataNum[i,41] &
                         portfolioList[,2]==CFdataNum[i,42] &
                         portfolioList[,3]==CFdataNum[i,43])
}
CFdataPortSum = matrix(0,nrow = portfoliosTotal,ncol = 40) #year 0 to 39 of data, no CVRs
for (i in 1:firmsNo) {
 portNum = CFdataNum[i,44]
 CFdataPortSum[portNum,]=CFdataNum[i,1:40]+CFdataPortSum[portNum,]
}
PortSize = vector(mode = "numeric",length = portfoliosTotal)
for (i in 1:portfoliosTotal) {
 PortSize[i] = sum(CFdataNum[,44] == i)
}
CFdataPortAvg = CFdataPortSum[,(dataStart+1):(horizonEndCol+1)]/PortSize
#forecasting
firmsNo = length(CFdataTestComp[,1])
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecastEBITindGrow = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataTestComp[,(dataStart+1):(dataEnd+horizon+1)]
CFforecastEBITindGrow[,1:dataEnd] = CFdataTestComp[,(dataStart+1):(dataEnd+1)]
CFdataPortPer = CFdataPortAvg[,2:(horizonEndCol)]/CFdataPortAvg[,1:(horizonEndCol-1)]-1
for(i in 1:firmsNo){
 inputEBIT = CFforecastEBITindGrow[i,(dataEndCol)]
 CFdataTestComp[i,52] = which(portfolioList[,1]==CFdataTestComp[i,44] &
                        portfolioList[,2]==CFdataTestComp[i,48]&
                        portfolioList[,3]==CFdataTestComp[i,47])

 inputPort = CFdataTestComp[i,52]
 for(j in 1:horizon){
  CFforecastEBITindGrow[i,(dataEndCol+j)] = round(CFforecastEBITindGrow[i,(dataEndCol+j-1)]*
```

```
                                (1+CFdataPortPer[inputPort,(dataEndCol+j-1)]),2)
  }
}
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastEBITindGrow[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF16 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF16) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF16[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF16[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF16[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF16[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF16[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
  localErrorsF16[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
  globalErrors[16,i] = round(mean(localErrorsF16[!rowSums(!is.finite(localErrorsF16)),i]),4)
}
#-------- 2.1.13 portfolio development - EBIT&Growth10&Balance ----------
portfoliosEBIT = 5     #no. of EBIT groupings
portfoliosBal = 5      #no. of Balance groupings
portfoliosGrowth = 10  #no. of growth groupings
portfoliosTotal = portfoliosEBIT*portfoliosBal*portfoliosGrowth
quantilesBal = quantile(as.numeric(Balancedata[,inputYear+2]), c(.2, .4, .6, .8, 1), na.rm = TRUE)
quantilMatrixBal = matrix(c(quantilesBal,5:1),nrow=5,ncol=2)
portfolioNo = vector(mode="numeric", firms)
CFdataCalc = cbind(CFdata,portfolioNo)
for (i in 1:firms) {
  BalInputYear = as.numeric(Balancedata[i,inputYear+2])
  if(is.na(BalInputYear)){
    CFdataCalc[i,42]=NA
  } else if(BalInputYear<quantilMatrixBal[1,1]){
    CFdataCalc[i,42]=5
  } else if(BalInputYear<quantilMatrixBal[2,1]){
    CFdataCalc[i,42]=4
  } else if(BalInputYear<quantilMatrixBal[3,1]){
    CFdataCalc[i,42]=3
  } else if(BalInputYear<quantilMatrixBal[4,1]){
    CFdataCalc[i,42]=2
  } else{
    CFdataCalc[i,42]=1
  }
}
CFdataComp = CFdataCalc[complete.cases(CFdataCalc[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
CFdataComp = CFdataComp[complete.cases(CFdataComp[,42]),] #removing NA rows from BalportfolioNo
CFdataNum = CFdataComp[,2:42] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
growthData = (CFdataNum[,2:40]/CFdataNum[,1:39]-1)
firmsNo = length(CFdataNum[,1])
cols = length(growthData[1,])
for(i in 1:firmsNo){
  for(j in 1:cols){
    if(growthData[i,j]=="Inf" || growthData[i,j]=="-Inf"){
      growthData[i,j]=NA
```

```
   }
  }
}
quantilesGrow = quantile(as.numeric(growthData[,inputYear+1]),c(.1,.2,.3,.4,.5,.6,.7,.8,.9,1), na.rm = TRUE)
quantilMatrixGrow = matrix(c(quantilesGrow,portfoliosGrowth:1),nrow=portfoliosGrowth,ncol=2)
portfolioNo = vector(mode="numeric", firmsNo)
CFdataNum = cbind(CFdataNum,portfolioNo)
for (i in 1:firmsNo) {
 growthInputYear = as.numeric(growthData[i,inputYear+1]) #+1 because there is year zero
 if(is.na(growthInputYear)){
  CFdataNum[i,42]=NA
 } else if(growthInputYear<quantilMatrixGrow[1,1]){
  CFdataNum[i,42]=10
 } else if(growthInputYear<quantilMatrixGrow[2,1]){
  CFdataNum[i,42]=9
 } else if(growthInputYear<quantilMatrixGrow[3,1]){
  CFdataNum[i,42]=8
 } else if(growthInputYear<quantilMatrixGrow[4,1]){
  CFdataNum[i,42]=7
 } else if(growthInputYear<quantilMatrixGrow[5,1]){
  CFdataNum[i,42]=6
 } else if(growthInputYear<quantilMatrixGrow[6,1]){
  CFdataNum[i,42]=5
 } else if(growthInputYear<quantilMatrixGrow[7,1]){
  CFdataNum[i,42]=4
 } else if(growthInputYear<quantilMatrixGrow[8,1]){
  CFdataNum[i,42]=3
 } else if(growthInputYear<quantilMatrixGrow[9,1]){
  CFdataNum[i,42]=2
 } else{
  CFdataNum[i,42]=1
 }
}
quantilesEBIT = quantile(as.numeric(CFdataNum[,inputYear+1]), c(.2, .4, .6, .8, 1))
quantilMatrixEBIT = matrix(c(quantilesEBIT,5:1),nrow=5,ncol=2)
portfolioNo = vector(mode="numeric", firmsNo)
CFdataNum = cbind(CFdataNum,portfolioNo)
for (i in 1:firmsNo) {
 EBITinputYear = as.numeric(CFdataNum[i,inputYear+1])
 if(EBITinputYear<quantilMatrixEBIT[1,1]){
  CFdataNum[i,43]=5
 } else if(EBITinputYear<quantilMatrixEBIT[2,1]){
  CFdataNum[i,43]=4
 } else if(EBITinputYear<quantilMatrixEBIT[3,1]){
  CFdataNum[i,43]=3
 } else if(EBITinputYear<quantilMatrixEBIT[4,1]){
  CFdataNum[i,43]=2
 } else{
  CFdataNum[i,43]=1
 }
}
portfolioList = matrix(0,nrow=portfoliosTotal,ncol=6) #2 cols for each EBIT, Bal and growth
portfolioList[,1] = quantilMatrixBal[length(quantilMatrixBal[,2]):1,2]
growthPorts = matrix(data=0,nrow = (portfoliosTotal/portfoliosGrowth),ncol = portfoliosGrowth)
growthPorts[,1] = 1
growthVector = growthPorts[,1]
for(i in 2:portfoliosGrowth){
 growthPorts[,i] = i
```

```
  growthVector = c(growthVector,growthPorts[,i])
}
for(i in 1:portfoliosTotal){
  portfolioList[i,2] = growthVector[i]
  portfolioList[i,3] = sum(portfolioList[1:i,1]==portfolioList[i,1] &
                           portfolioList[1:i,2]==portfolioList[i,2])

  portfolioList[i,4] = quantilMatrixBal[which(quantilMatrixBal[,2]==portfolioList[i,1]),1]
  portfolioList[i,5] = quantilMatrixGrow[which(quantilMatrixGrow[,2]==portfolioList[i,2]),1]
  portfolioList[i,6] = quantilMatrixEBIT[which(quantilMatrixEBIT[,2]==portfolioList[i,3]),1]
}
CFdataNum = cbind(CFdataNum,portfolioNo)
CFdataNum = CFdataNum[complete.cases(CFdataNum[,42]),] #removing NA rows from portfolioNo
firmsNo = length(CFdataNum[,1])
for(i in 1:firmsNo){
  CFdataNum[i,44] = which(portfolioList[,1]==CFdataNum[i,41] &
                          portfolioList[,2]==CFdataNum[i,42] &
                          portfolioList[,3]==CFdataNum[i,43])
}
CFdataPortSum = matrix(0,nrow = portfoliosTotal,ncol = 40) #year 0 to 39 of data, no CVRs
for (i in 1:firmsNo) {
  portNum = CFdataNum[i,44]
  CFdataPortSum[portNum,]=CFdataNum[i,1:40]+CFdataPortSum[portNum,]
}
PortSize = vector(mode = "numeric",length = portfoliosTotal)
for (i in 1:portfoliosTotal) {
  PortSize[i] = sum(CFdataNum[,44] == i)
}
CFdataPortAvg = CFdataPortSum[,(dataStart+1):(horizonEndCol+1)]/PortSize
#forecasting
firmsNo = length(CFdataTestComp[,1])
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecastEBITbalGrow2 = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataTestComp[,(dataStart+1):(dataEnd+horizon+1)]
CFforecastEBITbalGrow2[,1:dataEnd] = CFdataTestComp[,(dataStart+1):(dataEnd+1)]
CFdataPortPer = CFdataPortAvg[,2:(horizonEndCol)]/CFdataPortAvg[,1:(horizonEndCol-1)]-1
for(i in 1:firmsNo){
  inputEBIT = CFforecastEBITbalGrow2[i,(dataEndCol)]

  CFdataTestComp[i,53] = which(portfolioList[,1]==CFdataTestComp[i,46] &
                         portfolioList[,2]==CFdataTestComp[i,48]&
                         portfolioList[,3]==CFdataTestComp[i,47])
  inputPort = CFdataTestComp[i,53]
  for(j in 1:horizon){
    CFforecastEBITbalGrow2[i,(dataEndCol+j)] = round(CFforecastEBITbalGrow2[i,(dataEndCol+j-1)]*
                                 (1+CFdataPortPer[inputPort,(dataEndCol+j-1)]),2)
  }
}
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastEBITbalGrow2[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF17 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF17) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF17[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF17[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF17[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF17[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF17[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
```

```
  localErrorsF17[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
  globalErrors[17,i] = round(mean(localErrorsF17[!rowSums(!is.finite(localErrorsF17)),i]),4)
}
#-------- 3.1.1 ARIMA - Best fit --------
CFdataComp = CFdataTest[complete.cases(CFdataTest[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
firmsNo = length(CFdataComp[,1])
firmsList = CFdataComp[,1]
CFdataNum=CFdataComp[,2:41] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecast = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataNum[,(dataStart+1):(dataEnd+horizon+1)]
CFforecast[,1:dataEnd] = CFdataNum[,(dataStart+1):(dataEnd+1)]
for(i in 1:firmsNo){
  tsData = ts(CFactual[i,1:dataEnd])
  model = auto.arima(tsData)
  fullForecast = forecast(model,h=horizon)
  tsForecast = fullForecast$mean
  CFforecast[i,(dataEnd+1):(dataEnd+horizon)] = round(tsForecast)
}
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecast[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF1 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF1) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF1[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF1[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF1[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF1[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF1[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
  localErrorsF1[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
  globalErrors[1,i] = round(mean(localErrorsF1[!rowSums(!is.finite(localErrorsF1)),i]),4)
}
#-------- 3.1.2 ARIMA - Input --------
#ARIMA model 1
p1 = 0   #AR order
d1 = 1   #degree of differencing
q1 = 1   #MA order
#ARIMA model 2
p2 = 1   #AR order
d2 = 0   #degree of differencing
q2 = 0   #MA order
CFdataComp = CFdataTest[complete.cases(CFdataTest[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
firmsNo = length(CFdataComp[,1])
firmsList = CFdataComp[,1]
CFdataNum=CFdataComp[,2:41] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
```

```
class(CFdataNum) <- "numeric"
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecast1 = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecast2 = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataNum[,(dataStart+1):(dataEnd+horizon+1)]
CFforecast1[,1:dataEnd] = CFdataNum[,(dataStart+1):(dataEnd+1)]
CFforecast2[,1:dataEnd] = CFdataNum[,(dataStart+1):(dataEnd+1)]
for(i in 1:firmsNo){
  tsData = ts(CFactual[i,1:dataEnd])
  model1 = Arima(tsData,c(p1, d1, q1),method="ML")
  fullForecast1 = forecast(model1,h=horizon)
  tsForecast1 = fullForecast1$mean
  CFforecast1[i,(dataEnd+1):(dataEnd+horizon)] = round(tsForecast1)
  model2 = Arima(tsData,c(p2, d2, q2),method="ML")
  fullForecast2 = forecast(model2,h=horizon)
  tsForecast2 = fullForecast2$mean
  CFforecast2[i,(dataEnd+1):(dataEnd+horizon)] = round(tsForecast2)
}
#model input 1
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastRWND[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF2 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF2) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF2[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF2[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF2[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF2[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF2[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
  localErrorsF2[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
  globalErrors[2,i] = round(mean(localErrorsF2[!rowSums(!is.finite(localErrorsF2)),i]),4)
}
#model input 2
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastRWND[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF3 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF3) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF3[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF3[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF3[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF3[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF3[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
  localErrorsF3[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
  globalErrors[3,i] = round(mean(localErrorsF3[!rowSums(!is.finite(localErrorsF3)),i]),4)
}
#-------- 3.2.3 Random Walk - rwf --------
CFdataComp = CFdataTest[complete.cases(CFdataTest[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
  CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
firmsNo = length(CFdataComp[,1])
firmsList = CFdataComp[,1]
CFdataNum=CFdataComp[,2:41] #Turn the data into only numbers
```

```
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecastRWND = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFforecastRWD = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataNum[,(dataStart+1):(dataEnd+horizon+1)]
CFforecastRWND[,1:dataEnd] = CFdataNum[,(dataStart+1):(dataEnd+1)]
CFforecastRWD[,1:dataEnd] = CFdataNum[,(dataStart+1):(dataEnd+1)]
for(i in 1:firmsNo){
 tsData = ts(CFactual[i,1:dataEnd])
 fullForecast1 = rwf(tsData,h=horizon,drift=FALSE)
 tsForecast1 = fullForecast1$mean
 CFforecastRWND[i,(dataEnd+1):(dataEnd+horizon)] = round(tsForecast1)
 fullForecast2 = rwf(tsData,h=horizon,drift=TRUE)
 tsForecast2 = fullForecast2$mean
 CFforecastRWD[i,(dataEnd+1):(dataEnd+horizon)] = round(tsForecast2)
}
#without drift accuracy
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastRWND[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)]
localErrorsF4 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF4) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
 localErrorsF4[i,1] = round(mean(CFdifference[i,]),2) # Error
 localErrorsF4[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
 localErrorsF4[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
 localErrorsF4[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
 localErrorsF4[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
 localErrorsF4[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
 globalErrors[4,i] = round(mean(localErrorsF4[!rowSums(!is.finite(localErrorsF4)),i]),4)
}
#with drift accuracy
CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
CFdifference = as.matrix(CFactualComp - CFforecastRWD[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
localErrorsF5 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
colnames(localErrorsF5) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
for(i in 1:firmsNo){ # removes the observations that are INF or NAN
 localErrorsF5[i,1] = round(mean(CFdifference[i,]),2) # Error
 localErrorsF5[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
 localErrorsF5[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
 localErrorsF5[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
 localErrorsF5[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
 localErrorsF5[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
}
for(i in 1:6){
 globalErrors[5,i] = round(mean(localErrorsF5[!rowSums(!is.finite(localErrorsF5)),i]),4)
}
#-------- 3.2.4 Random Walk - Brownian --------
runs = 1000
dt = 1 #1 year is the delta
BMerrorMatrix = matrix(data = 0,nrow = runs, ncol = 6)
colnames(BMerrorMatrix) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
CFdataComp = CFdataTest[complete.cases(CFdataTest[,dataStart+2]),] #plus 2 because CVR and year 0
for(i in (dataStart+1):(dataEnd+horizon)){
 CFdataComp = CFdataComp[complete.cases(CFdataComp[,i+2]),] #plus 2 because CVR and year 0
}
```

```
firmsNo = length(CFdataComp[,1])
firmsList = CFdataComp[,1]
CFdataNum=CFdataComp[,2:41] #Turn the data into only numbers
CFdataNum[is.na(CFdataNum)] = 0
class(CFdataNum) <- "numeric"
CFactual = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
CFactual = CFdataNum[,(dataStart+1):(dataEnd+horizon+1)]
for(n in 1:runs){
 CFforecastRWBM = matrix(data=0,nrow = firmsNo,ncol = (dataEnd-(dataStart-1)+horizon))
 CFforecastRWBM[,1:dataEnd] = CFdataNum[,(dataStart+1):(dataEnd+1)]

 for(i in 1:firmsNo){
  tsData = ts(CFactual[i,1:dataEnd])
  s2 = sd(tsData)
  mean = mean(tsData)
  for(j in 1:horizon){
   x = CFforecastRWBM[i,dataEnd+j-1]
   eta = rnorm(1,0,sqrt(s2))
   dw = eta*sqrt(dt)
   dx = mean * dt + sqrt(s2) * dw
   CFforecastRWBM[i,dataEnd+j] = round(x+dx)
  }
 }
 CFactualComp = as.matrix(CFactual[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
 CFdifference = as.matrix(CFactualComp - CFforecastRWBM[,(dataEnd-dataStart+2):(horizon+dataEnd-dataStart+1)])
 localErrorsF6 = matrix(data=0,nrow = firmsNo,ncol = 6) #as there are 6 errors to test
 colnames(localErrorsF6) <- c("ME","MAE","MRE","MAPE","MSE","MSRE")
 for(i in 1:firmsNo){ # removes the observations that are INF or NAN
  localErrorsF6[i,1] = round(mean(CFdifference[i,]),2) # Error
  localErrorsF6[i,2] = round(mean(abs(CFdifference[i,])),2) # Abs Error
  localErrorsF6[i,3] = round(mean(CFdifference[i,]/CFactualComp[i,]),2) # Rel Error
  localErrorsF6[i,4] = round(mean(abs(CFdifference[i,]/CFactualComp[i,])),2) # Abs Rel Error
  localErrorsF6[i,5] = round(mean(CFdifference[i,]^2),2) # Squared Error
  localErrorsF6[i,6] = round(mean((CFdifference[i,]/CFactualComp[i,])^2),2) # Squared Rel Error
 }
 for(i in 1:6){
  BMerrorMatrix[n,i] = mean(localErrorsF6[!rowSums(!is.finite(localErrorsF6)),i])
 }
}
for(i in 1:6){
 globalErrors[6,i] = round(mean(BMerrorMatrix[,i]),4)
}
```