MASTER

Using demand to prevent ageing and depreciation

Voorberg, S.

Award date:
2017

Link to publication
Using demand to prevent ageing and depreciation

S. Voorberg

September 9, 2017
Abstract

Revenue management is a broad field of models and methods that is growing in popularity. The goal of all these models is optimizing the expected revenues. In this thesis, we focus on two specific techniques within revenue management. Both techniques focus on forecasting future events based on uncertain information from history: Demand time series forecasting and dynamic pricing. The company Tech Data has asked to think of new ways to increase their revenue and inventory model to decrease depreciation costs and prevent aging. We discuss several time series forecasting methods and use them to determine the optimal method for Tech Data. Furthermore, we introduce some additional techniques such as hierarchical forecasting and regression models. Secondly, we use dynamic pricing to research methods that model the correlation between demand and price setting. The goal is to find a model that optimizes revenues and could be used by Tech Data. We combine a theoretical basis with numerical experiments on live data from Tech Data.
Acknowledgements

This thesis will be the end of an exciting period in my life. After a bachelor in Industrial Engineering, the switch to Mathematics has not always been easy. However, after three years of hard work I am fiercely proud of what I have accomplished with thanks to the people that were there through those years.

First and foremost, I would like to thank Maria for her coaching through this project. Not only were you able to appoint me the right people to work with, but through this year you have become more than just a thesis mentor. You pushed me to think of the future and to realise that this thesis is not just the finishing of a period in my life, but also the beginning of a new period. Besides Maria, I would like to thank Bert, Koo and Alessandro for their support during my research. Secondly, I would like to thank Jos and his team for his guidance in a complete new world for me. A company as Tech Data has complicated and hidden habits that you guided me through. Your enthusiasm and interest in the project were inspiring. Thank you to Yorick, Mark and Remco. Yorick, we went into this challenge together and although we did not finish at the same time, it feels as if we have completed this together. Mark and Remco, thank you for your patience. I have always felt free to ask you whatever I wanted to know during our many train journeys.

Finally, I would like to thank my lovely Anouk. You have been a great support to me. You inspire me with your always ongoing work spirit. But above all you support me with your unconditional trust in me.
# Contents

1 Introduction .......................................................... 4  
   1.1 Tech Data ....................................................... 5  
   1.2 Research ....................................................... 9  
   1.3 Overview ...................................................... 9  

2 Forecasting .......................................................... 10  
   2.1 Data research .................................................. 11  
   2.2 Precision measures .......................................... 19  
   2.3 Accuracy measures ........................................... 20  
   2.4 Forecasting methods ......................................... 21  
   2.5 Dynamic regression ......................................... 25  
   2.6 State space Models .......................................... 26  
   2.7 Spectral analysis ............................................ 29  
   2.8 Intermittent demand ........................................ 29  
   2.9 Hierarchical or grouped forecasting ....................... 31  
   2.10 Cross validation ............................................ 33  

3 Pricing ............................................................... 35  
   3.1 Deterministic problem ....................................... 36  
   3.2 Stochastic problem .......................................... 37  
   3.3 Problems with limited information ....................... 38  
   3.4 Solutions ...................................................... 40  
   3.5 Assumptions .................................................. 41  
   3.6 Tech Data ..................................................... 43  

4 Forecasting model .................................................. 45  
   4.1 Dataset ......................................................... 45  
   4.2 Data selection and cleaning ................................ 46  
   4.3 Cross Validation ............................................. 49  
   4.4 Intermittent demand ....................................... 55  
   4.5 Hierarchy ..................................................... 58
4.6 TD Forecast .................................................. 60

5 Pricing model .................................................. 62
   5.1 Single item pricing ...................................... 62
   5.2 Multi item pricing ...................................... 66

6 Conclusions and recommendations ......................... 71
   6.1 Demand forecast ........................................ 71
   6.2 Pricing ................................................... 72

A R scripts ....................................................... 74
Chapter 1

Introduction

In our current world, distribution is becoming a business of smaller getting margins and profits. Distribution is all about knowing the demand. Being able to predict demand, results in smaller warehouses, less transportation costs, faster delivery times and more optimal prices. Demand is defined as:

“The quantity of a good or service that economic agents are willing to buy at a given price.” (“Demand” 2017)

The behaviour of demand can be defined by multiple characteristics. If we look at the definition, it is important to know how many economic agents exist in the market. Furthermore, we need to know what goods or services are offered. Finally, the price is of importance on the quantity that is bought. These three parameters also have dependent variables. Examples of such variables are the number of competitors in the market, the length of the distribution channel and the economic conditions.

The field of research that is concerned with demand as parameter to make decisions is broad. Decisions in multiple directions can be made to maximize revenue based on demand knowledge.

In this thesis, we will focus on two methods that will attempt to model the demand as a function of different dependent variables.

**Time series forecasting methods:** \( \hat{y}_t = f(y_{t-1}, \ldots, y_1) \)

In this method, demand depends on historic events. Searching for patterns and trends in the historical data enables us to improve forecasts of demand.

**Dynamic pricing:** \( \hat{y}_t = f(p_{t-1}, c_{t-1}, v_{t-1}, j_{t-1}) \)

In this method, demand depends on multiple variables that jointly determine demand. The most important example is the price of the product.
Both methods have different flavours of functions with their pros and cons. The challenge is to find the right function for the right situation. Time series forecasting uses possible correlations with historical values to predict the future. This is introduced in Section 2.1. Pricing models use the correlation between demand and pricing and the estimated parameters of this correlation to predict future sales, as introduced in Chapter 3.

Before we introduce scientific theory and models for both these methods, we first have a look at the company and the problem for which these methods will be studied.

1.1 Tech Data

Founded by Edward C. Raymund, in 1974, Tech Data is a distribution company in the IT sector for over 40 years. It connects the large IT brands over the world with the resellers on smaller scale. The broad portfolio of IT products enables resellers to gain access to all their demands at one location. Besides that, Tech Data functions as a hub within the IT sector, supports resellers with financing, supply chain services, training and more value-added services. With a strong position in both North-America and in Europe, Tech Data is able to gather all of the worldwide IT brands and their solutions in one company.

In 2016, Tech Data sold products with a total value of 26.3 billion dollars. Tech Data is listed on the stock exchange of the NASDAQ with a current share value of $95 (2 May 2017). In the IT world, Tech Data is renowned for its aggressive acquisition policy. Over the last decade, more than 20 companies have been acquisitioned. This is one of the reasons for its steep growth in the market.

Within the Netherlands, Tech Data is a leading company in IT-distribution with multiple lines of business. Tech Data has its business divided into a collection of specialists. Specialist divisions such as Azlan, that is present in the enterprise IT industry, or Maverick, that is present in the Audio Video industry, create a strong platform in the IT market. Although over the years Tech Data has become a more sales focused company, its main function is still distributor.

Ageing at Tech Data

Ageing arises when products have been stocked and are not sold for a long period. This period depends on the known and expected demand for this product. For a distributor this is a problem, because it has multiple negative consequences. For example, the product will not only have to be depreciated, but in the case of disposal, extra cost have to be incurred. Furthermore, a distributor is careful with its space. Every square centimetre that is used ineffectively is a potential...
problem. Moreover, it is becoming more important to be a socially responsible company. Therefore, disposing valuable products is something companies always want to avoid.

Because of Tech Data’s function as distributor, it has stock for thousands of products with many valuable items. After several months, unsold products become aged and have to be depreciated.

The solution given in the previous paragraph is not solving the cause of the problem. Defining a better way of stock keeping will ensure less unnecessary stock and ageing. The question from Tech Data is:

Is it possible to decrease the reservation cost by improving the and if so, what are possible solutions?

It should be noted that some problems are never to be avoided. For example, the return of sold items can cause trouble if the product is rarely sold. In this thesis, we will not focus on these exceptions.

**Inventory at Tech Data**

Demand plays an important role at a company such as Tech Data. A distributor is judged by its customers on having the demanded product quickly available. This is an important reason why Tech Data is always interested in better ways of forecasting future demand for its products.
1. **Back-to-Back.** Products with little sales are not kept in stock.

2. **Aged.** Aged products are excluded from buying to ensure the old products are sold.

3. **New.** New products have no sales history and need a manual forecast.

4. **Stock.** Products with enough sales are kept in stock and future demand is anticipated in stock keeping.

5. **Safety Stock.** Products have a safety stock to ensure availability of the product. However, future demand is not anticipated.

Although it would have been interesting to include all these products in our research to see if we could find better conditions, Tech Data has a very strict international procedure that cannot be changed for a single country. Therefore, we will not include these conditions in our time series research. For this research we will use the products that are categorized as Stock.

**Demand at Tech Data**

The function of distributor makes that Tech Data has an assortment of thousands of products. A small percentage of these products is sold on a regular basis. However, the largest part of these products is sold irregularly. This causes difficult sales patterns to be forecast.
Price listing at Tech Data

Prices are an important tool in the world of distribution. Customers can be satisfied with a combination of low prices and slow distribution. Other customers are willing to pay high prices in combination with fast service. This also shows the connection between inventories and pricing.
1.2 Research

Inventory planning is always based on knowledge of demand. The demand forecasting method that Tech Data is using, seems to be too simple due to a lack of mathematical substantiation. We discuss the current forecasting method and introduce the current scientific methods on this subject. All methods will be tested on a Tech Data data set.

Furthermore, we introduce a different approach to solve the problem of predicting demand. Describing demand as a function of price and available stock makes it possible to use forecasting in a different way. Using dynamic pricing, we can adapt the price sensitivity of consumers to the amount of stock available causing more sensitivity to stock. This directly influences the amount of stock available, which might lead to less depreciation cost.

The above conclusions result in the following main question for this thesis:

*Is it possible to decrease the expected depreciation loss in a distribution company?*

Answering this question leads to the following subquestions:

1. Can the depreciation loss be decreased by improving the demand forecasting method?
2. Are there forecasting methods that can improve the current method?
3. Can the depreciation loss be decreased by influencing the demand with dynamic pricing?
4. Are there dynamic pricing models that can improve the current pricing method?
5. Does the inclusion of dynamics to the pricing model have a positive effect on the depreciation loss?

1.3 Overview

In Chapter 2 we discuss the currently available methods for forecasting. Chapter 3 is used to define the currently known dynamic pricing methods and their mathematical basis. Chapter 4 will introduce methods that we use to forecast demand at Tech Data and judge their performances. Chapter 5 is used to define a dynamic pricing model that could be used at Tech Data. Finally, chapter 6 will conclude this thesis and summarize the results.
Chapter 2

Forecasting

In this chapter we discuss current methods and models of demand forecasting from literature and their advantages and disadvantages. "Forecasting" (2016) is denoted as:

"Predicting future behaviour of a variable based upon statistical/mathematical principles applied to past and current observations of the target variable and of a limited number of explanatory variables that have been found to be in a statistically significant relationship with it. Usually the method involves specifying assumptions about the behaviour of the explanatory variables."

An important group of forecasting techniques is based on qualitative research. Qualitative research uses market knowledge and the psychology of customers. Well known methods are the Delphi method and market research (Rowe & Wright, 1999). We do not focus on these methods in this thesis because emphasis will be put on using quantitative data that is available to help forecast.

Many forecasting methods only use past and current observations, so called time series. A time series is to be defined as:

\[ \{y_t\}, \ t = 0, 1, 2..., n \]

\[ \hat{y}_t = f(y_{t-1}, ..., y_1) \]

For the demand forecasting, we use time series. However, there is one method that we discuss in Section 2.5 where explanatory variables can be included.

It is important to realise that most time series forecasting methods are parametrized. Most of the methods that are used today are parametric. In parametric approaches, first we try to fit a model, assuming the validity of the model, on the currently known data, optimizing the parameters and afterwards utilizing the model to create a forecast.
There are also non-parametric methods, reviewed by Härdle et al. (1997). Non-parametric methods allow for a general form, where parametric methods make more detailed assumptions. This general form causes the method often to be unreliable for shorter time series. In Section 2.7, one non-parametric method will be introduced.

In this chapter we focus on the introduction of models that fit the data by optimizing the parameters. However, it is impossible to decide which model will be best for each time series separately. Every method optimizes its parameters in a different way. Therefore, the goal is to find the best working method for all of the available time series. After this has been decided, the model is optimized by determining the number of parameters and their value per series. Before the introduction of these models, the data needs to be cleaned. Therefore, we will first introduce methods to do so.

2.1 Data research

Outliers

Time series often show unexpected observations. Although it is difficult to indicate values that are outliers, it is important to mention this subject. Most of our forecasting methods are based on the fact that there are repeatable movements or trending directions in the data. Having a single outlier might already disorganize the recognition of patterns, which will lead to significantly worse forecasting results. Outliers can be defined as sudden increases of sales that do not meet the expected value:

\[ y_t^* = y_t + x_t, \]

where \( x_t \) is the outlier. Much research has been done on how to remove outliers. In the book of Aggarwal (2013), specific outlier classes are introduced to find more applicable methods. Most outlier methods are based on the assumption of a distribution. Well known tests such as Dixon’s and Grubb’s test assume a normally distributed population. Both methods use a test variable based upon the distance between the value of the possible outlier and the other values. This test variable is tested using a hypothesis test.

However, a time series cannot be assumed to be normally distributed as it is time-correlated. As an improvement, Chen & Liu (1993) fit an ARIMA model and include the outlier analysis in fitting the model.

In this research, we conduct a simple outlier analysis introduced by prof. dr. R.J. Hyndman. Using an old technique by Friedman (1984), we search the time series on outliers by fitting a Loess curve (Jacoby, 2000) after a STL decomposition which we discuss in Section 2.7 and looking at the quantiles of the residuals. The
Loess curve is a non-parametric method that uses local smoothing. Based on a specified neighbourhood, all inside values are assigned a weight depending on the distance to the value. The average of these weighted values gives the Loess value. Replacement is done by linear interpolation. In our case, the Loess curve is assumed to sufficiently fit the data points and thus to discover any outliers. Because this method does not make any assumptions on what model or distribution to use, it is applicable to our data. In Figures [1]a and b, we see the outlier analysis performed on a time series of a Tech Data product. Figure a is the initial time series. Figure b is the time series that has been cleaned, using the technique that was introduced. Through this chapter, we use this time series to display certain functions or graphs.

**Correlations and Stationarity**

Forecasting is mainly based on correlations and patterns in a time series. To recognize these correlations, we introduce some terms. The mean function is defined as:

\[ \mu_t = E(y_t) \]

We define the autocovariance function of a time series between time \( t \) and \( s \) as:

\[ \gamma_{t,s} = Cov(y_t, y_s) = E[(y_t - \mu_s)(y_s - \mu_t)] \]

We define the autocorrelation function of a time series as:

\[ \rho_{t,s} = Corr(y_t, y_s) = \frac{\gamma_{t,s}}{\sqrt{\gamma_{t,t}\gamma_{s,s}}} \]

Both covariance and correlation are functions indicating the dependence of two random variables. If \( \rho_{t,s} = 0 \), there is no correlation between the values, indicating that this historic term with lag \( t - s \) has no significant effect on the forecast. It is important to indicate that the autocorrelation function is dependent on all historic values up to lag \( t - s \). However, we also would like to know the direct correlation between lags \( t \) and \( s \), without all the other values in between. Therefore, we introduce the partial autocorrelation function:

\[ \phi_{kk} = Corr(y_t, y_{t-k}|y_{t-1}, y_{t-2}, \ldots, y_{t-k+1}) \]

This is the autocorrelation function without the effect of all intermediate values between \( t \) and \( k \).

We introduce the definition of stationarity to understand what the effect is on a time series.
(a) Amount of weekly sales for a product from 2013 until 2015

(b) Cleaned time series of sales of the product from 2013 until 2015

(c) Cleaned and stabilized time series of sales of the product from 2013 until 2015

Figure 1
“A time series displays stationarity if the expected value at all points in time is the same and if, additionally, the correlation between the values at two time points, \( t \) and \( t + \tau \), depends on the lag \( \tau \) but not on \( t \).” ("Stationarity", 2014)

\[
\{y_1, \ldots, y_T\} \overset{d}{=} \{y_{1+h}, \ldots, y_{T+h}\}
\]

The assumption of stationarity ensures the fact that we can predict something based on our currently known data. Knowing this, we can use the mean, variance and autocorrelation function for future values as we assume that these do not change over time.

The above stated assumption results in:

\[
\gamma_{t,s} = \gamma_{0,|t-s|}
\]

This simplifies our notation to:

\[
\gamma_h = Cov(y_t, y_{t-h}) \quad \text{and} \quad \rho_h = Corr(y_t, y_{t-h})
\]

We also would like to estimate the earlier introduced functions without knowing the model, but only having a time series. Therefore we define the sample mean:

\[
\bar{y} = \frac{1}{n} \sum_{t=1}^{n} y_t
\]

We define the sample autocovariance function, assuming we have stationarity, as:

\[
\hat{\gamma}(h) = \frac{1}{n} \sum_{t=1}^{n-h} (y_{t+h} - \bar{y})(y_t - \bar{y})
\]

We use the sample mean to estimate the real mean in the autocovariance function. Furthermore, we can only use the values with a lag of size \( h \), as we look for correlations with a lag of size \( h \). Therefore, we can only run the sum up to \( n - h \).

We define the Sample Autocorrelation function(ACF) as:

\[
\hat{\rho}(h) = \frac{\hat{\gamma}(h)}{\hat{\gamma}(0)}
\]

Finally we define the Sample Partial Autocorrelation function(PACF) recursively as:

\[
\hat{\phi}_{kk} = \frac{\rho_k - \sum_{j=1}^{k-1} \phi_{k-1,k} \rho_{k-j}}{1 - \sum_{j=1}^{k-1} \phi_{k-1,k} \rho_{k-j}}
\]

where \( \phi_{k,j} = \phi_{k-1,j} - \phi_{kk} \phi_{k-1,k-j} \)

Now that we have defined those values, we can use them to indicate which historical values need to be included in the model to make it more reliable. This approach of including certain historical lagged values is mostly used by the ARMA model in Section: 2.4
Figure 2

(a) ACF of product

(b) PACF of product

(c) Lagplot of product for lag 1 and 52
Time Series Patterns

For individual time series, modelling always begins with visualizing the data to detect certain visual patterns. For example, Figure 2 shows the dependencies on lags. Figure 2 a shows the autocorrelation function, introduced in Section 2.1, for a product. Figure 2 b shows the partial autocorrelation function. Figure 2 c shows the sales value plotted against the value of last week and the value of 52 weeks ago. The left graph shows the correlation with last week and the right graph shows the correlation for a year.

We would like to recognize patterns evolving over time. We distinguish three different patterns: Seasonality, Cycles and Trending. We define seasonality as:

“The seasonal variability of certain economic or financial factors, for example unemployment or commodity prices. In statistical analyses of time-series data, findings may need to be adjusted for seasonal variation.” (“Seasonality” 2016)

It is important that the variability has a fixed seasonal period. This could be weekly, monthly or yearly periods. In our datasets we use yearly seasonality. Cycles are also periodically defined variabilities. However, these periods have no standard seasonal length and will therefore not be picked up in the seasonality, but in the trend variable. To find patterns, we use the “Seasonal and Trend decomposition using Loess” (STL), introduced by (Cleveland et al., 1990). We define the STL Decomposition:

\[ y_t = f(s_t, t_t, e_t), \]

where \( s_t \) is the seasonal component at period \( t \), \( t_t \) is the trending/cyclic component at period \( t \) and \( e_t \) is the error component at period \( t \).

STL decomposition is a non-parametric method. The decomposition can be either additive,

\[ y_t = s_t + t_t + e_t \]

or multiplicative,

\[ y_t = s_t \times t_t \times e_t \]

or possible combinations between them. One could also apply a log transformation to turn a multiplicative relationship into additive. This decomposition method makes it easy to create seasonally adjusted data. For this decomposition, we make use of the Loess smoothing. We use the smoothed value to see if the real value is different because of seasonality. Using the seasonally adjusted data, we can apply different models to recognize if there are further dependencies that can be included in the model. Figure 3 shows the STL decomposition for the example time series.
Data transforming

Certain methods require stationarity to determine which lags should be included in the model. A method to create stationarity is differencing. A differenced series is the difference between each two observations in the original series. Differencing can also be performed seasonally. Differencing stabilizes the mean. The reason why we would like to use differencing instead of the use of an Auto Regression term, which will be introduced in Section 2.4, is that sometimes the dependency changes over time. As an Auto Regression term has a constant linear variable, we cannot take into account the time dependency. With differencing this effect can be removed.

Using for example a Kwiatkowski–Phillips–Schmidt–Shin (KPSS) test or the Dickey-Fuller test, one can test if differencing is necessary. These tests are called unit root tests. The idea of these tests is that, after fitting a trend, the series should look like white noise and not have any linear deviations. If there is too much linear deviation, differencing is needed to create stationarity. Sometimes it is necessary to test and difference twice or more for stationarity.

- **Simple differencing**
  \[ y_t = x_t - x_{t-1} \]
  
- **Seasonal differencing**
  \[ y_t = x_t - x_{t-m} \]

An extra option that could be applied is the Box-Cox transformation. A Box-Cox
transformation is defined as:

\[
\begin{align*}
    y'_t &= \log(y_t) \quad \text{if } \lambda = 0 \\
    y'_t &= (y_t^\lambda - 1)/\lambda \quad \text{if } \lambda \neq 0
\end{align*}
\]

As the assumption of a Box-Cox test is that the time series is larger than zero, we also increase the dataset with a certain level to achieve this assumption. The Box-Cox method can be used preliminary to ensure a more stable variance. Especially if the variance shows signs of dependency on the mean. Figure 1c shows the time series of product 18 adapted to a Box-Cox transformation.

**Independence testing**

After data cleaning and fitting a forecasting model, we end up with a series of residual values. However, to be sure that there is no correlation left that should be included, we want to test the series. If we can conclude that these residuals are white noise, we can conclude that all correlation have been included. There are different methods that can be used to conclude that all correlations have been included.

**ACF**

The first method is using the sample autocorrelation function on the residuals. It has been proven that for large \( n \) the ACF are approximately normally distributed: \( \{\gamma_1, \gamma_2, ..., \gamma_t\} \sim N(0, 1/n) \). So if there are more than 2 values within the first 20 ACF values outside the confidence intervals, there is probably still more correlation that should be included in the model. Figure 4 shows the ACF for the residuals of an AR(1) model. This model is introduced in Section 2.4. In this figure we see that all lags are within the bounds of the confidence interval.

**Portmanteau Test**

A second method we introduce to test for independent residuals, is making use of a Portmanteau Test. Its null hypothesis assumes an independent distribution. The statistic \( Q \) for Ljung-Box is defined as:

\[
Q_{LB} = n(n + 2) \sum_{j=1}^{h} \frac{\hat{\rho}^2(j)}{(n - j)},
\]

where \( \hat{\rho}(j) \) is the ACF as described in Section 2.1. Assuming that this \( Q \) statistic is chi-squared distributed with degrees of freedom \( h \), we can test if the hypothesis is accepted.
A somewhat similar method checking the ACF values is directly looking at the residuals. Assuming enough data is available, we can draw a histogram of the residuals or a Q/Q plot and check with the eye if the residuals seem to be normally distributed with $\mu = 0$. One can also use the Shapiro Wilkinson Test with null hypothesis: normally distributed series.

2.2 Precision measures

Precision measures indicate how precise the distribution of the random variable can be estimated, resulting in more precise forecasts. Information criteria are a statistically used measure of precision. An information criterion looks at the goodness of fit of the model put against the loss of reliability due to model complexity. This complexity is expressed in the amount of parameters that is included. An important information criterion is the Akaike’s information criterion (AIC).

$$AIC = 2k - 2 \ln(\hat{L})$$

$$\hat{L}(\hat{\theta}, \text{Model}|x) = P(x|\hat{\theta}, \text{Model}),$$
where \( k \) is the amount of free parameters used and \( \hat{L} \) is the maximum likelihood value. We penalize models with too many estimators as they lose reliability. As shown by Parzen et al. (2012), minimizing the AIC also means minimizing the Mean Squared Error for one-step-ahead forecasts. This is an important proof that connects precision measures and accuracy measures that are introduced in Section 2.3.

Other used Information criterions are the AIC for small samples (AICc) and the Bayesian information criterion (BIC):

\[
AICc = AIC + \frac{2(k + 1)(k + 2)}{n - k - 2},
\]

\[
BIC = k \ln(n) - 2 \ln(\hat{L}),
\]

where \( n \) denotes the number of observations. With the restriction \( n \gg k \). Both these measures have a larger penalty term on additional parameters and are therefore more careful with large estimation models.

### 2.3 Accuracy measures

Accuracy measures are measures of bias. Such errors will often not be noticed by increasing the dataset and optimizing the precision. However, a better model will improve accuracy. Error terms can be used to indicate how accurate a model fits a dataset. In comparing different models to one data set, one should therefore use accuracy measures to find the best model. In the past decades multiple error terms have been introduced. A much used error term is the mean absolute error (MAE).

\[
MAE = \frac{\sum_{t=1}^{n} |e_t|}{n}
\]

A setback on this error term is that it does not penalise extra for larger deviations. This is useful because we prefer small forecasting errors for each week over rare large forecasting errors. This does happen with a squared error function such as the root mean squared error. Both these functions do not take into account the size of the predictions itself. For example, if a product has average sales of 0.5 products, a constant deviation of five products is worse than it is for a product with an average sales of 50 products. For these problems the mean absolute percentage error or mean absolute scaled error can be used.

Different error terms are:

- root mean squared error (RMSE)

\[
RMSE = \sqrt{\frac{\sum_{t=1}^{n} |e_t^2|}{n}}
\]
mean absolute percentage error (MAPE)

$$MAPE = \frac{100}{n} \sum_{t=1}^{n} \left\vert \frac{e_t}{y_t} \right\vert$$

mean absolute scaled error (MASE) (Hyndman & Koehler 2006)

$$MASE = \frac{\sum_{t=1}^{n} |e_t|}{\sum_{t=2}^{n} |y_t - y_{t-1}|}$$

The MASE is an error term stating the relative performance of a method against the simple naive method (Section 2.4). Therefore, if the method scores less than one, it improves our forecasts and should be favoured over the naive method.

We would like to point out that we have made use of standard error terms such as the MAE and the RMSE. They do not take into account the possibility that underestimating has other influences on the result than overestimating. Although both underestimating and overestimating have a negative effect on the Tech Data model, they are different. Underestimating will lead to lost sales, while overestimating will lead to depreciating products and higher inventory costs. However, this is not part of the research as it would complicate the forecasting too much.

### 2.4 Forecasting methods

In this section we will introduce methods that can be used to model the correlations that we discovered in Section 2.1. All functions are assuming that we are estimating a future value, indicated by the hat on all forecasting estimators.

#### Naive forecasting

We introduce the simplest forecast that exists, which is the most recently found value for this variable: a naive forecast. Mathematically stated, with $T$ the most recent value and $h$ the horizon of forecasting:

$$\hat{y}_{T+h|T} = y_T$$

It is surprising how often this is an accurate way of forecasting. However, one should try to make more use of all known information from the past. The variable might have some trend or seasonality that could be used to improve the forecasting method. Including this results in a trending naive method or seasonal naive method:

$$\hat{y}_{T+h|T} = y_T + h \times (y_T - y_{T-1})$$
\[\hat{y}_{T+h|T} = y_{T+h-km},\]

where \( m \) is the seasonal period and \( k = \left\lfloor (h - 1)/m \right\rfloor + 1. \)

Other simple methods to predict sales are the average or weighted average:

\[\hat{y}_{T+h|T} = \bar{y} = \frac{\sum_{t=1}^{T} y_t}{T}\]

\[\hat{y}_{T+h|T} = \frac{\sum_{t=1}^{T} y_t w_t}{\sum_{t=1}^{T} w_t}\]

**Exponential Smoothing**

Exponential smoothing is based on learning from the past. Using the parameters in the method, it is possible to decide how correlated the variable is with historic values or with the present. The method smooths the historical values exponentially with a factor \((1 - \alpha)\).

Simple exponential smoothing:

\[\hat{y}_{T+1|T} = \alpha y_T + \alpha(1 - \alpha)y_{T-1} + \alpha(1 - \alpha)^2y_{T-2} + \ldots\]

Component form:

\[\hat{y}_{T+h|T} = l_T\]

\[l_T = \alpha y_T + (1 - \alpha)l_{T-1},\]

\(\alpha\) and \(l_0\) are chosen by minimizing the AIC. The AIC gives the precision of this model based on the currently available data.

Exponential smoothing with trend:

\[\hat{y}_{T+h|T} = l_T + hb_T\]

\[l_T = \alpha y_T + (1 - \alpha)(l_{T-1} + b_{T-1})\]

\[b_T = \beta^*(l_T - l_{T-1}) + (1 - \beta^*)b_{T-1}\]

Exponential smoothing with damped trend:

\[\hat{y}_{T+h|T} = l_T + (\phi + \phi^2 + \ldots + \phi^h)b_T\]

\[l_T = \alpha y_T + (1 - \alpha)(l_{T-1} + \phi b_{T-1})\]

\[b_T = \beta^*(l_T - l_{T-1}) + (1 - \beta^*)\phi b_{T-1}\]

\(\alpha, \beta^*, l_0\) and \(b_0\) are chosen by minimizing the AIC.

Holt Winters method:

\[\hat{y}_{T+h|T} = l_T + hb_T + s_{T-m+h-m}\]
\[ l_T = \alpha(y_T - s_{T-m}) + (1 - \alpha)(l_{T-1} + b_{T-1}) \]
\[ b_T = \beta^*(l_T - l_{T-1}) + (1 - \beta^*)b_{T-1} \]
\[ s_T = \gamma(y_T - l_{T-1} - b_{T-1}) + (1 - \gamma)s_{T-m} \]

The parameters are optimized by minimizing the AIC. The Holt Winters method includes all possible ways of decomposition, including trend and seasonality. It is important to note that both the seasonal and trend variable can also be estimated as a multiplicative factor. However, for the trend it is advised to always use an additive estimator. An additive estimator allows for linear trend, a multiplicative estimator will allow for exponential trend.

**Auto Regression and Moving Average**

The simplest stationary time series we can construct is:
\[ y_t = c + f(Z_t, Z_{t-1}, Z_{t-2}, ..., Z_{t-q}) \]
where \( \{Z_t\} \sim WN(0, \sigma^2) \). Suppose \( y_t \) is dependent up to \( Z_{t-q} \), we call \( y_t \) \( q \)-dependent.

We introduce the Moving Average(q) term:
\[ \hat{y}_T = c + \theta_1 e_{T-1} + \theta_2 e_{T-2} + ... + \theta_q e_{T-q} \]
where \( \theta \) are the parameters that are optimized using the AIC or by looking at the ACF and \( \{e_t\} \) are the residuals. Assuming that \( \{e_t\} \sim WN(0, \sigma^2) \) is equal to assuming that \( \{y_t\} \) is a stationary process. A Moving Average (MA) term can also be discovered using the ACF and PACF. Suppose an MA(2), then the covariance is
\[ \gamma(\tau) = \begin{cases} (1 + \theta_1^2 + \theta_2^2)\sigma^2 & \text{for } \tau = 0 \\ (\theta_1 + \theta_1\theta_2)\sigma^2 & \text{for } \tau = 1 \\ \theta_2\sigma^2 & \text{for } \tau = 2 \end{cases} \]

Using the definition of ACF, \( \rho(t) > 0 \) for \( \tau = 0, 1, 2 \)

The other terms in an ARMA model are the Auto Regression(p) terms
\[ \hat{y}_T = c + \phi_1 y_{T-1} + \phi_2 y_{T-2} + ... + \phi_p y_{T-p} + e_T \]
The Auto Regression(AR) term takes historical values and adds them with a weight \( \phi \) to the forecast value. This method differs from exponential smoothing because it estimates all \( \phi \) values separately, whereas exponential smoothing only estimates
the $\alpha$ value. Using the ACF and PACF values, we can use the information to include certain specific historical terms in our model. The coefficient of

$$\phi_{hh} = \begin{cases} \phi_p & \text{if } h = p \\ 0 & \text{if } h > p \end{cases}$$

Thus, if the PACF plot shows constant peaks up to $h = 5$, we should include the AR(5) term in our model.

We refer to figure 2 a and b to see an example of the ACF and PACF. The optimal model for this series is an ARMA(1,2) model.

Combining the two models gives a combination of lagged values and lagged errors of $y_T$, resulting in an Auto Regression and Moving Average model(ARMA). For large time series databases it is very difficult to find an appropriate model by looking at the ACF and PACF for each series separately. Therefore we can use the AIC or AICc to estimate the amount of parameters needed and the values of these parameters.

It is important to realise that MA and AR terms are invertible. In the earlier mentioned references, derivations are shown to prove that each AR process can also be thought of as an infinite MA process. We do not focus on this subject as we would particularly like to focus on applying these methods.

Auto Regression Integrated Moving Average

An ARIMA model is an ARMA model with a differencing term. Because ARMA models require a stationary time series we can use differencing to achieve this condition. Differencing was covered in Section 2.1. So an ARIMA model can be described by three parameters: $(p,d,q)$. Describing the amount of MA(q) terms, AR(p) terms and differencing(d). A simple ARIMA model $(2,1,2)$ is described as follows:

$$y^d_T = y_T - y_{T-1}$$

$$y^d_T = c^d + e^d_T + \theta_1 e^d_{T-1} + \theta_2 e^d_{T-2} + \phi_1 y^d_{T-1} + \phi_2 y^d_{T-2}$$

Integer valued Auto Regression

An INAR model is a special subclass of AR-processes. INAR models do require integer values as forecasts. Although INAR is the most well known model in this class, the used method can also be applied to other models. It is also possible to create an INMA model. The method that is used to find the integer values is called ‘thinning’. A simple example is binomial thinning. For an INAR(1) method, we take $X_{T-1}$ and split this into single Bernoulli variables. For this Bernoulli distribution we assign a parameter $\alpha$ that gives the probability of success.
Summing the successful variables will always result in an integer as answer, as shown by Jung & Tremayne (2006) and Weiß (2008). Because Tech Data only sells complete products, it might be useful to study the effect of this method on product databases with thousands of products. However, adding the problem of finding correctly rounded variables causes a new problem for every forecasting method. Therefore we will not include this in our research.

2.5 Dynamic regression

Dynamic regression is a different way of forecasting variables with not only time series, but also some additional explanatory variables. Examples are marketing spend on certain products or life cycle phase. Firstly, we fit a simple regression model on the time series:

\[ y_T = \beta_0 + \beta_1 x_{1,T} + \ldots + \beta_k x_{k,T} + n_T, \]

where \( n_T \) is the residual and \( x_{1,T}, x_{2,T}, \ldots, x_{3,T} \) are known regressor variables. Secondly, the residual is used as input for an ARIMA model:

\[ n_T = c + e_T + \theta_1 e_{T-1} + \ldots + \theta_k e_{T-k} + \phi_1 n_{T-1} + \ldots + \phi_k n_{T-k} \]

Disadvantage of this method is that many regressor variables are also unknown in the future. For example, the number of unique customers buying a product might be an excellent explanatory variable on total sales of products. However, we do not know how many customers arrive in the future, resulting in the fact that this variable also has to be forecast.

Short life

Because we know that Tech Data has many products that have a short life cycle, predicting the length of this cycle can give an indication on the upcoming demand. One could use a Weibull distributed variable to model a life cycle, as the Weibull distribution is very much suited to model the life time of a product. In the articles of Aytac & Wu (2013) and Yelland (2010) there is an interesting application of this idea. Another option is to make use of the Bass model (Bass, 1969). This model has been known for a long time to model the life of a product and its characteristics as shown in publications by Kurawarwala & Matsuo (1996) and Wu et al. (2010).

Periodic Seasonality

Seasonality can arise with different time spans. There exists, for example, yearly, quarterly, weekly and even daily seasonality. Picking up small periods with long
range data is difficult using the conventional way of seasonalising as was described in Section 2.1. However, there is a different method that can be used to model these small-period seasons. Using a dynamic regression model, we can introduce Fourier terms. Fourier terms are useful for approximating a periodic function. The more terms we introduce, the more accurate the model will become, but the more parameters we have to estimate. Furthermore, it is possible to model additional seasonal patterns in a model using Fourier terms. This results in a special regression model. Each $\alpha$ and $\beta$ term have to be estimated in undermentioned formula.

$$
y_T = \alpha_1 \sin\left(\frac{2\pi T}{m}\right) + \beta_1 \cos\left(\frac{2\pi T}{m}\right) + ... + \alpha_k \sin\left(\frac{2\pi kT}{m}\right) + \beta_k \cos\left(\frac{2\pi kT}{m}\right) + n_T$$

Price

An important factor that is influencing the demand directly is the price. In a market with supply and demand, the price determines the amount of demand available. Therefore the price could be included in our regressive model. However, this method is the basis of chapter 3. Therefore we will not further discuss this method here.

2.6 State space Models

All before mentioned forecasting methods can be generalized into one framework that is called state space modelling. State space models allow for a method in which we distinguish different states. If $y_t$ is our observed value, it can be seen as a combination of underlying states such as level, trend and seasonality. The model can thus be described as:

$$y_t = w'(x_{t-1}) + \epsilon_t, \quad t = 0, 1, 2, ...$$

$$x_t = F(x_{t-1}) + g\epsilon_t, \quad t = 0, 1, 2, ...$$

where both $F$, $g$, and $w'$ are coefficients and $\{\epsilon_t\} \sim N(0, \sigma_\epsilon)$. All three coefficients are determined by the model. A state space model consists always of two basic equations: An observation equation and a transition equation. The first equation describes the connection between the observed value and her underlying states. The observed value is always assumed to be a combination of historic states and some unpredictable part $\epsilon_t$. The second equation describes the evolution of the states due to the information of time $t$, also consisting on a combination of historic states and an unpredictable term. $g$ breaks apart the $\epsilon_t$ into different errors per state. In the case of a linear innovation state space model, both equations are in linear form. Moreover, the $\epsilon_t$ can have multiple distributions in this model.
Recall that all the definitions of the exponential smoothing methods in section 2.4 were already written as state space models, with small differences in the notation. For example:

\[ \hat{y}_t = \hat{y}_{t-1} + \alpha(y_{t-1} - \hat{y}_{t-1}) \]

Which would be written as:

\[
\begin{align*}
y_t &= x_{t-1} \\
x_t &= x_{t-1} + \alpha \epsilon_t
\end{align*}
\]

Define the equations for Auto Regression(1) as

\[
\begin{align*}
y_t &= x_{t-1} \\
x_t &= \phi x_{t-1} + \nu_t
\end{align*}
\]

Define the equations for ARMA(p,q):

\[
Y_T = [\theta_{r-1} \ \theta_{r-2} \ \ldots \ \theta_0] X_t,
\]

with \( \theta \) being the parameters that were introduced with the Moving Average model.

\[
X_{t+1} = \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 1 \\ \phi_r & \phi_{r-1} & \phi_{r-2} & \ldots & \phi_1 \end{bmatrix} X_t + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \epsilon_{t+1,} \\ 0 \end{bmatrix}, \quad t = 0, 1, 2, \ldots
\]

Having defined and introduced some state space notation and modelling, we introduce Kalman filtering. The Kalman filter is a method based on bayesian inference. Bayesian inference uses a prior probability and likelihood to find a posterior probability. Kalman filtering has an algorithm working in two steps. First, the prediction step estimates the new states of the model. Secondly, the model is revised after a new observation. A third step that sometimes is used is the smoothing step. We introduce:

\[
z_{t|t-1} = \begin{bmatrix} y_{t|t-1} \\ x_{t|t-1} \end{bmatrix}
\]

Which results in:

\[
z_{t|t-1} = A x_{t|t-1} + b \epsilon_t
\]

Where \( A = \begin{bmatrix} w \\ F \end{bmatrix} \) and \( b = \begin{bmatrix} 1 \\ g \end{bmatrix} \) We define the following variables:

\[
\mu_{t|t-1} = E(y_t|y_{1..t-1})
\]

27
\[ \sigma_{t|t-1}^2 = Var(y_t|y_{1..t-1}) \]
\[ x_{t|t-1}^\mu = E(x_t|y_{1..t-1}) \]
\[ x_{t|t-1}^\sigma = Var(x_t|y_{1..t-1}) \]
\[ \zeta_{t|t-1} = Cov(x_t, y_t|y_{1..t-1}) \]
\[ z_{t|t-1}^\mu = E(z_t|y_{1..t-1}) \]
\[ z_{t|t-1}^\sigma = Var(z_t|y_{1..t-1}) \]

The Kalman prediction step uses the above defined variables to create a prediction for the next step.

\[ z_{t|t-1}^\mu = A x_{t-1|t-1}^\mu \]
\[ z_{t|t-1}^\sigma = A x_{t-1|t-1}^\sigma A' + \sigma_\epsilon^2 b b' \]
\[ \zeta_{t|t-1} = \begin{bmatrix} \sigma_{t|t-1}^2 \\ \zeta_{t|t-1} \end{bmatrix} \]
\[ \sigma_{t|t-1}^2 = w' x_{t-1|t-1}^\sigma w + \sigma_\epsilon^2 \]

The Kalman filter filters new information through its parameters to improve the predictions.

\[ x_{t|t}^\mu = x_{t-1|t-1}^\mu + \zeta_{t|t-1}(\sigma_{t|t-1}^2)^{-1}(y_t - \mu_{t|t-1}) \]
\[ x_{t|t}^\sigma = x_{t-1|t-1}^\sigma - \zeta_{t|t-1}\zeta_{t|t-1}'(\sigma_{t|t-1}^2)^{-1} \]

Although the methods seems complicated, it is surprising how useful it is in research on time series. The method is capable of creating confidence intervals, assuming normally distributed errors, and has a simple recursion to always create the most reliable prediction. Using methods such as Exponential smoothing and ARMA, we will include state space modelling in this research. For more extensive information on this subject one could consult [Harrison & West (1999)](Harrison%20%26%20West%20(1999)) or [Yelland (2010)](Yelland%20(2010)).
2.7 Spectral analysis

Spectral density approaches the model using a combination of frequency components. It was originally used to process signals in the information processing. Instead of prediction based on historic values, we predict a future value based on all observed oscillations in the past. We use a spectral density function to indicate the significance of certain frequencies or cycles on the complete model. In multiple references (Wise 1955) and (Broersen 2006) the connection between the ACF and the spectral density is demonstrated, given that we meet certain assumptions. This model looks a lot like the model that uses Fourier regression terms in Section 2.5. However, in this model we lose the ARIMA error term and try to capture the whole model using frequencies. Spectral density is a very different method and because it is less complete than the Fourier term regression, we will use that method. Spectral analysis can either be used parametric, using the ACF to include certain periods, or non-parametric based on a periodogram as in figure 5 (Choudhuri et al. 2004).

\[ y_T = u_{11} \sin\left(\frac{2\pi T}{m}\right) + u_{21} \cos\left(\frac{2\pi T}{m}\right) + \ldots + u_{1k} \sin\left(\frac{2\pi kT}{m}\right) + u_{2k} \cos\left(\frac{2\pi kT}{m}\right) \]

2.8 Intermittent demand

A special class of demand time series is intermittent demand. Intermittent demand has clear intervals between demand arrivals. Therefore it is much more difficult to predict what the future demand will be. However, exponential smoothing is a known working method for these sorts of time series. A different method is to model these series, forecasting the time periods between sales and the corresponding demand separately. A large setback on the intermittent methods is the fact that it forecasts a certain constant demand until the next demand arrives. In that way, there is enough stock at the moment that it arrives. However, if we consider the application of the forecast for inventory holding, it is not useful. The disadvantage of weekly ordering stock and not knowing when it will leave the distribution center is too much.

Binary forecasting

Binary forecasting focusses especially on the forecasting of ones or zeros. Where we mainly focus on the products that are possible to predict using simple times series methods as ARIMA or Exponential smoothing, this way of forecasting could very well be used in Tech Data, as more than 50% of all products can be assigned as
having intermittent demand. Predicting these sales is much more inaccurate and thus expensive. However, in literature there has been research into this subject. An important field where these series often appear is in the field of theoretical coding. (S.-I. Liu, 2001) (Morvai & Weiss, 2003)

**Croston method**

The method of Croston is a model that wants to forecast the arrival intervals and average demand separately. Let $y_t$ be our forecast, $x_t$ denote observed demand in a period, $v_t$ be the mean inter arrival time, $w_t$ be the mean demand without null sales and $q$ the current number of non sales periods.

If $x_{t-1} \neq 0$:

$$w_t = \alpha x_{t-1} + (1 - \alpha)w_{t-1}$$
$$v_t = \alpha q + (1 - \alpha)v_{t-1}$$
$$y_t = \frac{w_{t-1}}{v_{t-1}}$$

If $x_{t-1} = 0$:

$$w_t = w_{t-1}$$
$$v_t = v_{t-1}$$

*Figure 5. Periodogram*
Actually this method feels very intuitive. It is an extension on the Exponential Smoothing method, conditioning on the inter-demand times.

Other methods

Some new publications in the field of intermittent demand have opted to make use of parametric methods using bootstrapping. Furthermore, Snyder et al. (2012) and Shenstone & Hyndman (2005) mention the use of Poisson arrivals and a different distribution for the demand. This leads to possible compound Poisson distributions that are very useful to forecast. It is beyond the scope of this research to further look into this.

2.9 Hierarchical or grouped forecasting

Since a few years there is a new approach on forecasting. In these models we work with product hierarchies to include more information. Athanasopoulos et al. (2009). We still need to find a suitable model to forecast single time series. However, the complete hierarchy is optimized afterwards, revising all single time series. At first, this method mostly proved to be helpful in forecasting high level time series in a product hierarchy (Hyndman et al. 2011). However, since a few years there is also research on optimizing the complete product hierarchy of time series based on its internal build. The difference between grouped and hierarchical forecasting is that the graph of a grouped model does not lead back to one node, while that does happen with a fully hierarchical model. In stead of predicting sales on a high level, sales are predicted on a low level and those forecasts are added up to find the higher levels. This ensures lower variances, concluding in more accurate forecasts. We introduce some new notation.

\[ y_t = y_{t-1} \]

\[ y_t = y_{(X,t)} \]

\[ y_t = y_{(A,t)} \]

\[ y_t = y_{(B,t)} \]

\[ y_t = y_{(C,t)} \]

Suppose we have a hierarchy as in figure 6. For this figure, \( S \) is

\[
S = \begin{bmatrix}
1 & 1 & 1 \\
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
\end{bmatrix}
\]
Figure 6. Hierarchical tree

Let $\hat{\mathbf{y}}_t(h)$ be a vector of simple h-step forecasts for all series of $\mathbf{y}_t$.

$$\hat{\mathbf{y}}_t(h) = S P \hat{\mathbf{y}}_t(h)$$

Matrix $P$ combines the simple h-step forecasts to a revised set of forecasts $\tilde{\mathbf{y}}_t(h)$ for the complete hierarchy. For example for the bottom up approach:

$$P = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \quad SP = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Suppose

$$\hat{\mathbf{y}}_t(h) = \begin{bmatrix} 0 & 3 & 3 & 3 \end{bmatrix}$$

Then

$$\tilde{\mathbf{y}}_t(h) = \begin{bmatrix} 9 \\ 3 \\ 3 \\ 3 \end{bmatrix}$$

We define, assuming we have simple unbiased forecasts, $E[\hat{\mathbf{y}}_t(h)] = E[\mathbf{y}_t(h)]$,

$$\mathbf{\beta}_t(h) = E[\hat{\mathbf{b}}_t(h)|\mathbf{y}_1, ..., \mathbf{y}_t]$$

$$E[\hat{\mathbf{y}}_t(h)] = S \mathbf{\beta}_t(h)$$

To have unbiased revised forecasts we want the following equation to hold.

$$E[\tilde{\mathbf{y}}_t(h)] = E[\hat{\mathbf{y}}_t(h)]$$

$$SPS\mathbf{\beta}_t(h) = S\mathbf{\beta}_t(h)$$

So $SPS$ must be $S$ to be unbiased. For the bottom up approach this is valid. However, using a similar equation for the variance and estimating the covariance matrix $\Sigma$ using matrix algebra, we can derive a solution for the optimal $P$ matrix.
This way both the bottom series and the rest of the hierarchy are all equally optimized. Hyndman et al. (2016) introduce three different optimization methods to find the covariance matrix and thus optimize the combination of forecasts. The solutions are:

1. Generalized Least Squares: we use algebra to estimate the covariance matrix.
2. Weighted Least Squares: we use a diagonal weighted matrix to estimate the covariance matrix.
3. Ordinary Least Squares: we ignore the covariance matrix.

The derivations are no further included because we do not intend to derive theoretical results. We use a standard function in R developed by R.J. Hyndman that optimizes the $P$ matrix for these problems. Having found the $P$ matrix, time series can hierarchically be adapted and optimized (Athanasopoulos et al., 2009).

2.10 Cross validation

The method of cross validation is a simple technique to validate if a certain method is accurate in forecasting. A model is chosen and fit on a training dataset to determine the optimal parameters. Afterwards, the model is tested on a testing dataset. Both datasets are subsets of a single time series. Usually, we have a large dataset that is partitioned into a training and testing set. In time series analysis, we assume time dependencies. Therefore, it is not possible to test a subset by leaving a random value out. We use time series cross validation. The method of time series cross validation begins with a subset of a time series that is used to predict future values. After each forecast, the oldest time series value is added to the training set. Again, we optimize all parameters and create a new forecast. The forecasts are compared to the real data using accuracy measures that were introduced in Section 2.3. Figure 7 shows a two week horizon time series cross validation, where blue is the training set and red the two week forecast.
Figure 7. Time series cross validation
Chapter 3
Pricing

In this chapter we will discuss the method of dynamic pricing. [den Boer (2015)](den Boer (2015)) describes dynamic pricing as

“the study of determining optimal selling prices of products or services, in a setting where prices can easily and frequently be adjusted.”

Adjusting the optimal price easily and frequently is often done, as most companies perform their price setting using information systems.

It is important to make a clear distinction between dynamic pricing and revenue management as these terms are often used interchangeable. Dynamic pricing is a method focussed on the setting of prices over time. Revenue management is defined by [Sierag et al. (2015)](Sierag et al. (2015)) as “the practice of deciding which products to sell to which customers at what price under capacity constraints.” [Talluri & van Ryzin (2005)](Talluri & van Ryzin (2005)) describe it as making ‘demand-management decisions’, just as supply-chain management addresses supply decisions in a firm. In the case of revenue management the goal is increasing revenues. These are much broader definitions of which dynamic pricing can be seen as a subset. [Bitran & Caldentey (2003)](Bitran & Caldentey (2003)) give in their article a broad overview of pricing models for the complete field of revenue management both deterministic as stochastic.

In the end, both definitions study demand and the parameters that are influencing demand. However, a dynamic pricing model is able to almost continuously adapt on the known information and thus improve its model. Therefore, we could see revenue management as the search for the optimal parameters in the formula:

\[ y = f(p, g, h, c, t) \]

In the case of dynamic pricing this function would change into:

\[ y_t = f(p_t, g_t, h_t, c_t, t) \]
A revenue management model consists of four steps according to Talluri & van Ryzin (2005).

1. Data collection, collecting historical data of mainly demand.

2. Estimation and Forecasting, forecasting future data, most importantly demand.

3. Optimization, determining the optimal set of parameters to maximize revenues.

4. Control, using the derived parameters to control the systems.

For a dynamic pricing model, these four steps become a small cycle. After each time step, new information has been gained. This information changes our forecasts. New forecasts lead to different optimal parameters, resulting in a new system. The most important parameter that is determined in a pricing model is the price elasticity. It must be emphasized that two important decisions are made in every cycle. The first decision is determining the model that is assumed to model the underlying reality. The second decision is how to optimize this model with the currently known information. Equally to the forecasting models, the model is often settled and maintained and the parameters are adapted every time cycle.

It is important to note that the practical applicability of dynamic pricing is often proven to be difficult. To reliably estimate unknown parameters asks for much data and even large multinational companies have found out that they were not able to collect these amounts of data. A further consequence of dynamic pricing is the continuous change of prices in the market. This can cause agitation under customers and competitors in the market. Advantages of the economy are the available amounts of data through data systems and also the minimal costs of changing product prices. Companies, such as Tech Data, correct for exchange rates between Europe and North-America. That means that customers are aware of the fact that prices can always change. For these reasons, it is important to consider the added value of dynamic pricing before actually implementing it.

### 3.1 Deterministic problem

Before we specify the different assumptions that can be made in a dynamic pricing model we will introduce the simplest pricing model. Suppose a distributor of one single item $X$ with an infinite inventory in a monopoly market with myopic customers. Myopic customers are non-strategic. This means that they buy a product when they need it. The change of prices is allowed continuously. Furthermore, we
have a deterministic arrival function of \( l(p) \) customers per day, depending on price \( p \). This will result in optimizing the problem:

\[
r(p) = l(p) \cdot p
\]

Assuming the concavity of the revenue function, the solution can be found. Suppose:

\[
l(p) = a - bp
\]
\[
r(p) = ap - bp^2
\]

Then the optimal price can be found

\[
p = \frac{a}{2b}
\]

### 3.2 Stochastic problem

One of the first scientific models in the field of dynamic pricing was introduced by Gallego & van Ryzin (1994). The most important assumption in their model is that it is stochastic. They assume an arrival process that is affected by the price through function \( \lambda(p) \), which is assumed to be known. Examples of such functions are: \( \lambda(p) = ae^{-p} \) or \( \lambda(p) = a + bp \). Arriving demand is modelled as a Poisson process with parameter \( \lambda(p) \). Using this information, the revenue rate of this model is defined as:

\[
r(\lambda) = \lambda p(\lambda)
\]

This results in the following differential equation.

\[
J^*(n, t) = \sup_{\lambda} [\lambda \delta t (p(\lambda) + J^*(n - 1, t - \delta t)) + (1 - \lambda \delta t) J^*(n, t - \delta t)]
\]

Where \( n \) is the amount of available stock and \( t \) is time. In the article, for \( \lambda(p) = ae^{-p} \) an optimal solution is derived for these conditions. One of the surprising results that were found in the Gallego & van Ryzin (1994) research is that fixed pricing methods, even in stochastic situations, perform very good. A fixed pricing method is very simple and sometimes this is the best result in a situation where there is too much of uncertainty. However, in situations where there are unknown parameters it is necessary to discover the demand function by using different prices. Therefore, in such situations fixed pricing is not efficient.
3.3 Problems with limited information

In the beginning of dynamic pricing, research was done on mostly deterministic models with known parameters. However, demand is an unknown process with parameters that should be modelled stochastically. In the article of Gallego & van Ryzin (1994), demand is modelled as a stochastic arrival process. In their article, the assumption was that all parameters could be exactly determined. However, demand is a stochastic process with unknown parameters. Therefore, the goal of dynamic pricing is to develop pricing policies that learn the price-demand relation or any other relation through sales data that comes available every day. In chapter 2 many methods have been explained to estimate demand parameters. We did not include explanatory variables such as price or inventory other than through regression. Therefore, we will introduce some methods that use the correlation between price and demand. Most studies base their methods on one of the two following models.

1. Suppose the arrival of customers is now a poisson process with a certain arrival rate $\lambda$. After this arrival the customers show a certain willingness-to-pay (WtP) that is independent and identically distributed by $F(p)$. Now the expected revenue can be given as:

$$E \left[ \int_{t=0}^{T} p_t \lambda F(p_t) dt \right]$$

Farias & Van Roy (2010)

2. The second model that is also used by multiple authors is a simple linear pricing model, where $\lambda = a - bp$. The expected revenue can be given by:

$$E \left[ \int_{t=0}^{T} p_t \lambda_t dt \right]$$


For both models the objective is to maximize the expected revenues.

Bayesian Approach

Authors that use Bayesian methods to solve one of the above mentioned problems, use the same basic model. Before the calculations, an assumption is made on what demand function and probability distribution will be used. According to the Bayesian idea, this prior probability is used to determine the optimal policy. After new relevant data has become available, the updated posterior probability distribution is calculated. As assumptions can be made on what demand function
and distribution is used and finally which policies should be included in the research, this field of studies is broad. An important conclusion that has been drawn by several researchers, is that for Bayesian policies there is a positive probability that the price tends to a non-optimal position, so called incomplete learning.

As an example we describe the model of [Lobo & Boyd (2003)]. In their article they use the linear pricing model. Where $a$ and $b$ are parameters that are assumed to be normally distributed and where both parameter distributions are updated by Bayesian updating schemes. The updating in this particular article is done by Kalmar filtering. This is explained in Section 2.6. In this thesis we will not focus on Bayesian methods to analyse our data.

### Non-Bayesian approach

Other parametric approaches have also been used to find an optimal dynamic pricing policy. Most of them use maximum likelihood or linear least squares estimation to estimate the unknown parameters for the demand function. The difference with the Bayesian approach is that we do not estimate the complete distribution of the demand parameter. Therefore, a prior distribution is not necessary.

The linear least squares method is a statistically simple method that estimates the linear demand variables $a$ and $b$ by minimizing the squared residuals. This method does assume that the residuals are normally distributed with $\mu = 0$. Suppose we have available data up to time $i$, with $D_t$ the demand at time $t$. Assuming the concavity of the revenue function:

$$(\hat{a}_{i+1}, \hat{b}_{i+1}) = \arg \min_{a,b} \left\{ \sum_{t=1}^{i} (D_t - (a - bp_t)^2) \right\}$$

$$\hat{p}_{i+1} = \frac{\hat{a}_{i+1}}{2\hat{b}_{i+1}}$$

Although this method appears to be very simple and not very reliable, especially in scenarios where the demand is non-linear, this method is still used. Moreover, [Besbes & Zeevi (2015)] have proven the efficiency of this approach even in non-linear demand scenarios.

[den Boer & Zwart (2015)] use maximum likelihood estimation to estimate the optimal variables for a problem with a Bernoulli distributed demand. This method also allows differently distributed residuals. However, we introduce the likelihood function assuming the normal distribution. The likelihood function is:

$$L(\hat{a}, \hat{b}, \hat{\sigma}, p|D) = \prod_{t=1}^{i} P(D_t|p_t, \hat{a}, \hat{b}, \hat{\sigma}) = \prod_{t=1}^{i} \frac{1}{\sqrt{2\pi \hat{\sigma}^2}} e^{-\frac{(D_t - (\hat{a} - \hat{b}p_t))^2}{2\hat{\sigma}^2}}$$
The log-likelihood function is:

\[
\ln(L) = \sum_{i=1}^{n} \ln(P(D_t|p_t, \hat{a}, \hat{b}, \hat{\sigma})) = -\frac{n}{2} \ln(2\pi) - n \ln(\hat{\sigma}) - \frac{1}{2\hat{\sigma}^2} \sum_{t=1}^{i} (D_t - (\hat{a} - \hat{b}p_t))^2
\]

Knowing that for a normal distribution with \( \mu = 0 \), the maximum likelihood is equal to the least squares estimators, we could compare these two methods. However, the advantage of using maximum likelihood is the possibility of using a different demand distribution.

### Robust Optimization

Robust optimization is a different non-parametric method used by different authors to determine optimal policies. Robust Optimization is an optimization method that uses a worst case scenario. This results in certain bounds that can be optimized. However in a practical situation, such as this thesis, this method is not useful and will therefore not be further studied. (Gorissen et al., 2015) (Adida & Perakis, 2006)

### 3.4 Solutions

Different policies have been found to optimize the earned revenues by using dynamic pricing with learning. We will introduce the most important solutions.

#### Passive learning

This solution is also called certainty equivalence pricing or the myopic policy. The idea of this method is to learn by optimizing the price and assume that we find the global optimum.

Using new data a new optimal solution is determined. In the case of a linear least squares method, this means we reassess the optimal fit of a linear line through the data points and use this to adjust the price ensuring optimal revenues. This optimal solution will be adapted after each time period. The criticism on this method is that it tends to drift to a possible local optimum without having considered all feasible prices. Afterwards, the model will only confirm the local optimality of the current model and therefore revenues will be lost. Multiple researches have been performed on this issue and it has been proven that there always is a positive probability of not finding the optimal solution. In algorithm 1, we introduce a simple passive learning policy. By setting upper and lower bounds on \( p \), we prevent situations where the revenue function might temporarily become convex, causing infinite prices.
Algorithm 1 Passive Learning Policy

**Initialize.** Set initial price $p_1, p_2$. Determine $p_{max}$.

For $t \geq 2$:

**Estimate.** Determine the optimal parameters $\hat{a}_{t+1}, \hat{b}_{t+1}$ by using least linear squares.

**Update.**
- If $\frac{\hat{a}_{t+1}}{2b_{t+1}} \geq p_{max}$, set $p_{t+1} = p_{max}$
- Else if $\frac{\hat{a}_{t+1}}{2b_{t+1}} \leq p_{purchase}$, set $p_{t+1} = p_{purchase}$
- Else $p_{t+1} = \frac{\hat{a}_{t+1}}{2b_{t+1}}$

Active learning

Active learning methods will always keep in mind the trade-off between learning and earning. While the optimal revenue may point a different direction regarding the setting of the price, the active learning part of these methods will force the method to keep looking for better prices in the feasible region. Especially in the field of dynamic pricing, this is very important. As the company has no information on its clients, it is very important to learn to know them. That means one needs to discover prices that are too low and prices that are too high. [den Boer & Zwart (2013)] have researched the effect of an active learning method they call Controlled Variance pricing (CVP) compared to the Certainty Equivalence pricing method. This method uses an extra condition that creates a non-feasible region around the 'passive' optimum in case the variance of the price is not large enough. The interval is:

$$TI(t) = p_t \pm \sqrt{c[(t + 1)^\alpha - t^\alpha] \frac{t + 1}{t}}$$

The $\alpha$ and $c$ can be chosen such that the $Var(p)_t \geq c(t + 1)^{\alpha - 1}$. In their article they have proven that the parameters will converge to their true values.

3.5 Assumptions

[Talluri & van Ryzin (2005)] point out that there is a number of assumptions in a dynamic pricing model that need to be discussed.

Inventory effects

The mentioned solutions and policies are also applied in situations where we have finite inventories. Most of the research is performed on situations where there is no replenishment. In that case, it is very important to determine the starting
inventory. Fashion retailers are an example of companies that do not have the
time to replenish their items during the season.

However, there are also scenarios where replenishment is possible. The dynamic
of pricing is now not only caused by changes in the demand function but also by
the change of the marginal value of the stock. We have a pricing and inventory
problem. In the book of [Talluri & van Ryzin (2005)], the effect of stock keeping with
replenishments is also incorporated in stochastic pricing models. The problem has
two decision variables. Besides the price, we also determine the optimal inventory
level. Suppose we have an inventory of $C_t$ for the product. Furthermore, let $C^+_t$
be the after order inventory, $h_t$ the holding cost at the end of each time period
and $o_t$ ordering cost per new unit. Now:

$$J^*(n, t, C_t) = \sup_{\lambda, C^+ \geq C_t} [\lambda \delta t (p(\lambda) + J^*(n - 1, t - \delta t, C^+_t - 1) - h_t(C^+_t - 1)) + (1 - \lambda \delta t)(J^*(n, t - \delta t, C^+_t) - h_t(C^+_t)) + o(\delta t) - o_t(C^+_t - C_t)]$$

In this scenario the optimal solution, is a base-stock, posted price policy found
by [Elmaghraby & Keskinocak (2003)]. The solution actually looks like a passive
policy, apart from the situation that the inventory exceeds the optimal base stock
level. In that situation the price is discounted, the more the inventory exceeds the
base stock. In the case of finite capacity and inclusion of inventory in the learning
problem, active learning is not really necessary. Due to the fact that not only
price but also stock determines our parameters we will always learn, also if the
passive learning strategy is used as discovered by [den Boer & Zwart (2015)]. This
strategy is called endogenous learning. In section 5.1, we will introduce a model
that includes the current inventory policy used by Tech Data.

Research on jointly learning dynamic pricing and holding inventory in a learn-
ing environment is new. We found two studies into this subject by [Petruzzi &
Dada (2002) and Xu & Hopp (2005)]. The difficulty for this problem is the amount
of parameters that need to be optimized based on unknown sales data. Both au-
thors, fix some parameters and optimize their problem based on a subsection of
the unknown parameters.

**Strategic customers**

The continuous change of prices causes the possibility to strategically buy prod-
ucts. A myopic customer can be imagined as an arriving customer with a certain
willingness to pay (WtP). The customer enters the store and buys the product
dependent of its WtP. However, if we have strategic customers that arrive, the
situation changes. Certainly in an environment where customers can continuously
check the current price, we will see strategic customers that will wait as long as
the price is decreasing. Customers will attempt to predict the corporate strategy and optimize their own revenue problem. Myopic customers are easier to model and thus more used.

A solution that is often introduced in researches with strategic customers is so called pre-announced pricing. By announcing prices beforehand it is possible to determine the strategy of customers. An example of this way of pricing is fixed pricing. Q. Liu & Zhang (2013) have shown that price commitment in a competitive market is valuable. Tech Data sells in a market where the appearance of strategic customers seems not very plausible. Products are often sold to resellers that need stock or to large companies in projects. The customers buy products when they need it.

Multiple products

A distribution company will often sell multiple versions and models of one product from different vendors. We call these products substitutable. The price of the one product will influence the sales of the other products resulting in non-optimal results. This means that the price of all products has to be optimized either knowing the price of the other products or jointly. In section 5.2 we introduce a model that makes use of a number of company regulations at Tech Data, causing the model to optimize a group of products.

Monopoly environment

An assumption that is also important in the simple pricing model that we introduced is the monopoly market. A research direction is the utility of a market parameter in our forecasts to include certain outside influences. Den Boer (2015) introduces a new demand function with a market variable included.

It has also been discussed that assuming a monopoly in a market is automatically taking into account by using data from this market. Sudden changes in demand might not only be caused by changing willingness to pay but also by a competitor lowering its prices. However, this can be seen as an attempt to influence the willingness to pay that will therefore return in the model without actually adding a market variable.

3.6 Tech Data

Tech Data is part of a highly competitive market with many strategic purchases and highly correlated products. That means we have to be careful to suddenly implement pricing policies based on the fact that they predict higher revenues.
Batches

An interesting new field of research is the addition of arrival of customer batches (Levin et al., 2014) (Ngendakuriyo & Taboubi, 2016). Besides simple products such as laptops, Tech Data also distributes hardware for data centers. These orders are often in the order of thousands of products. It would be interesting to see if there are results that would also be helpful to Tech Data. However in this research we have focussed on relatively simple product groups, leaving the topic for further research.

Perishability

The IT market in which Tech Data is present, is changing fast. Although products have no expiration date, trends and developments cause changes. Laptops of three years old are already not meeting the standard as it is today. However, the large costs arise when products have to be depreciated. This means that the time horizon $T$ in itself also is an unknown parameter that should be estimated to reduce costs. In section 2.5 we already discussed the model of Bass to determine in which phase a product is. Models, such as Bass model, can also be used to predict the life time of products and predict $T$. 
Chapter 4

Forecasting model

In this chapter we will use several methods that were introduced in chapter 2 to forecast time series from Tech Data. Before we introduce the results of these methods, it is important to point out that forecasts are never correct. We try to approach reality, but we assume that this reality can never be completely modelled. The advantage of methods such as ETS and ARIMA, is that they have more adaptability and therefore should be able to come closer to reality. Many products from Tech Data have irregular selling patterns. That is why we focus on a product group within Tech Data with products that are sold more regularly.

4.1 Dataset

We have obtained 630 product time series that are sold in the sales group of Tech Data. The initial length of every time series consists of 216 weekly sales. In this research we will separate time series with more and less than 200 sold items. Table 2 classifies the time series based on the conditions. Time series, such as Figure 8 cannot be forecast reliably and would interfere with the errors for time series that can be forecast. For these series we use different methods such as Croston or a Compound Poisson method, as discussed in section 2.8. In section 4.4 we will discuss the time series with intermittent demand.
Data selection and cleaning

We begin with data cleaning and sorting as discussed in Section 2.1. Some products have expired during the four years of data we used. It would interfere with our models, if we would try to forecast these products outside their expiring date. Therefore, we resize the time series to their useful length. That means the series begins at its first sale and ends at its final sale. We use two different scenarios to research the forecasting methods.

1. Time series with more or less than 100 weeks of sold items
2. Time series with more or less than 30 weeks of sold items

The time series with less than respectively 100 and 30 weeks are indicated as intermittent as was discussed in Section 4.1 and studied in Section 4.4. Afterwards, the series are sorted based on two criteria.

1. Seasonality, the time series has a length of more than two years after resizing, allowing seasonal methods to be used.
2. Correlation, can be found by a Ljung-Box test.
As the seasonal series can be forecast by seasonal methods we will examine them separately to research the leverage of possible seasonal correlations. We use an STL decomposition as discussed in Section 2.1 for forecasting methods with seasonality. Furthermore, correlated time series indicate the value of forecasting methods that use this correlation and are therefore also studied separately.

The classifications lead to five different groups per scenario as shown in Table 2.

The large amount of demand that has no more than 100 sales periods over a length of 216 periods, indicates a known problem in the business-to-business sales world. Purchases are often done in batches to ensure discounts. This causes difficult and irregular sales patterns. Because most periods have no sales, the dependency on previous periods will appear to be high. That is why we have added the MASE in Section 2.3. It reveals the improvement of a method against the naive forecasting method. Furthermore, the percentage of correlated series is not high. This means that ARMA or ETS methods might not outperform simple methods such as the TD forecast over all time series.

We use the technique by Friedman (1984) to remove outliers. Therefore, it is valid to replace the outliers. It is interesting to apply a special method on these ‘outlier’ sales, such as Croston’s method. This is discussed in Section 4.4.

For single time series, we search for auto correlations. However, for 200 different time series, we use precision measures to ensure which parameters should be included in the forecasting models. We apply a BoxCox transformation on time series where it is necessary. The BoxCox transformation stabilizes the mean over time. This transformation is undone after forecasting to attain the real forecast. Furthermore, differencing is used in the ARIMA method to ensure stationarity.

Because of the multiple time series it is not possible to determine the inde-
Figure 9. Cross Validation Mean Absolute Error with seasonal and correlated data without cleaning

Data cleaning

Currently, the TD Forecast method is used without any data cleaning. We have studied if data cleaning has a significant effect on the accuracy of certain methods. Figure 9 shows the average Mean Absolute Error for 188 correlated products without data cleaning. Because the TD Forecast method has its own cleaning procedure by removing the maximum and minimum, it appears to be the best method. However, the average MAE for all methods is worse, including TD Forecast if we compare it to Figure 10. Therefore, we can conclude that the data cleaning has a positive effect on forecasting the regular demand of all products. Both the ETS and ARIMA need, as discussed, a more cleaned version of the series to recognize the correct correlations.
4.3 Cross Validation

After data cleaning, we use the data to test a list of methods that is compared using a time series cross validation as introduced in Section 2.10. The models that are tested using cross validation are:

1. Naive model as discussed in Section 2.4
2. Simple ETS as discussed in Section 2.4
3. ARIMA as discussed in Section 2.4
4. STL decomposed Naive model 2.1
5. STL decomposed ETS model
6. STL decomposed ARIMA model
7. TBATS model
8. Fourier regression as discussed in Section 2.5
9. Bayesian model
10. Tech Data current method as discussed in Section 1.1
11. Mean as discussed in Section 2.4

TBATS stands for Exponential smoothing State space model with Box-Cox transformation, ARMA errors, Trend and Seasonal components. It is a method that includes all these methods in one forecast and optimizes it using precision measures. The Bayesian method uses a simple linear trend and the Kalman filter method as discussed in Section 2.6.

We use the final 10 weeks for each product as testing set. That is the weeks at the end of the series, available after data cleaning. The limited amount of ten weeks can be explained. Firstly, the optimization takes much time for the amount of time series and methods that are included. Secondly, because of the different lengths of resized time series, the maximum of 10 weeks ensures a training set that is long enough. We separately study the methods on the classified groups of time series. This is because for each group different methods are possible to use.
Seasonal and Correlated data

Figure 10, 11 and 12 show respectively the MAE, RMSE and MASE of all methods given over the six weeks horizon for seasonal and correlated series. Table 3 indicates the average error for the six period horizon for all methods. Furthermore, we have tested all the methods on a short term forecast of two weeks to determine the optimal method. 188 seasonal and correlated products with more than 30 weeks of sales have been used to assess the methods.

The ARIMA and ETS models have a low Mean Absolute Error. As TBATS is a combination of these two models it also performs well. Furthermore, it is surprising to see the good performance of TD Forecast in series where there is significant correlation. The Root Mean Squared Error, which penalizes larger outliers, indicates the good performance of the mean function and the ARIMA and SARIMA method. As the mean is conservative and non-sensitive to large differences, it does explain the low RMSE error. The Mean Absolute Scaled Error indicates the best method is either ETS or ARIMA. Although overall the TD Forecast method does show small errors, we would advise to use the ARIMA or ETS method. This is also indicated in the Table 3 for short term forecasting. We would advise to use the ETS method. It is easier and faster to optimize. Looking at the MASE, we see only an improvement of 0.04 compared to the naive method. This again indicates the difficulty of forecasting these time series.

If we look at the methods that have a seasonal decomposition included, we can conclude that this seasonal component is not strong enough to increase the accuracy. It is possible that on the long term these methods do show better results. We see a small improvement of these methods over the time horizon. However, in this research, seasonality is not of enough significant relevance.

We perform an extra test where we only include the 96 series with more than 100 periods of sales. If these time series do indicate more correlation, we can conclude that the correlated methods have not enough information with 30 weeks of sales. Although the error terms do increase, this holds for all methods. The increase can be explained by the fact that these methods have much less non-sales periods, which makes it more difficult to forecast. This set of series has no significant changes in the errors of the individual methods compared to series with 30 or more weeks of sales. This can be seen in Figure 13. Therefore we conclude that the addition of information does not have a positive effect on finding the correlations.

It is important to note that most methods do differ less than one unit in the MAE. This indicates that the rounding of the values could have a large influence
(a) Cross Validation Mean Absolute Error with seasonal and correlated data

(b) Cross Validation Mean Absolute Error with seasonal and correlated data zoomed

*Figure 10*
Figure 11. Cross Validation Root Mean Squared Error with seasonal and correlated data

Figure 12. Cross Validation Mean Absolute Scaled Error with seasonal and correlated data
on the performance of a method. One might consider the value of an integer method, as was introduced in Section 2.4. However, in this thesis we will mainly focus on finding the optimal forecasting method. Therefore, we will not research the rounding of forecasting values.

Table 3

Methods comparison

<table>
<thead>
<tr>
<th>Method</th>
<th>MAE</th>
<th>RMSE</th>
<th>MASE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time horizon</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>Naive</td>
<td>6.59</td>
<td>5.29</td>
<td>34.77</td>
</tr>
<tr>
<td>ETS</td>
<td>5.61</td>
<td>5.30</td>
<td>28.08</td>
</tr>
<tr>
<td>ARIMA</td>
<td>5.58</td>
<td>5.29</td>
<td>26.47</td>
</tr>
<tr>
<td>SNAIVE</td>
<td>7.83</td>
<td>7.43</td>
<td>35.83</td>
</tr>
<tr>
<td>SETS</td>
<td>6.06</td>
<td>6.01</td>
<td>28.96</td>
</tr>
<tr>
<td>SARIMA</td>
<td>5.65</td>
<td>5.82</td>
<td>26.42</td>
</tr>
<tr>
<td>TBATS</td>
<td>5.77</td>
<td>5.25</td>
<td>32.01</td>
</tr>
<tr>
<td>Fourier</td>
<td>6.19</td>
<td>5.92</td>
<td>31.82</td>
</tr>
<tr>
<td>Bayesian</td>
<td>6.11</td>
<td>5.65</td>
<td>32.07</td>
</tr>
<tr>
<td>TDForecast</td>
<td>5.70</td>
<td>5.72</td>
<td>27.66</td>
</tr>
<tr>
<td>Mean</td>
<td>6.66</td>
<td>6.62</td>
<td>26.36</td>
</tr>
</tbody>
</table>

A conclusion on products with correlated lags is difficult to draw. We do conclude that seasonal decomposition is unnecessary. The time series of Tech Data do not show enough seasonal correlation over multiple years, compared to the short term weekly correlations.

There is not a single forecasting method that stands out. We would advise to use a portmanteau test to indicate correlations. If we find time-dependent time series, we do advise to incorporate a method for these series. ETS would, for this dataset, be the best advise, as it performs equally good as ARIMA but is much faster in calculations.

No seasonal correlated data

We study 132 time series with a life shorter than two year but with auto correlation. They cannot be forecast using seasonal methods. Figure 14 shows us that the MAE is almost doubled compared to the series with more than two years of data. Thus, although the seasonal methods do not give better forecasts, the value of two years of history is conducive for the forecast. In this scenario the TBATS method performs the best, not only for the MAE, but also the RMSE. Mostly in the first two weeks, the correlated methods such as ARIMA and ETS perform better than
(a) Cross Validation Mean Absolute Error with seasonal and correlated data

(b) Cross Validation Mean Absolute Error with seasonal and correlated data and more than 100 sales periods

Figure 13
the more simple methods. Again, pointing at the fact that \[ \text{simple methods} \], we would advise to use a method focused on correlation. The advantage that is achieved in the first two weeks can be valuable for such items.

Uncorrelated data

If we look at time series without significant auto correlations, we find completely different results. It is surprising that in this case, the TBATS model is showing a high accuracy in both figure 14 and table 4 as TBATS is a model based on correlation methods. In this scenario, the TD forecast method has high accuracy. The error is much higher, which is no surprise given the absence of correlation. We would advise to use the TD forecast method for these products. No other method significantly improves the forecasting, certainly not on the short term. For these products the simplicity of the TD forecast method makes it reliable.

4.4 Intermittent demand

\[ \text{Intermittent demand} \] We therefore have created a special function that builds a separate time series for these large sales amounts. Figure 16 shows a figure with the outlier
(a) Cross Validation Mean Absolute Error with seasonal and independent lagged data

(b) Cross Validation Mean Absolute Error with no seasonal and independent lagged data

*Figure 15*
Table 4

Methods comparison uncorrelated seasonal data

<table>
<thead>
<tr>
<th>Method</th>
<th>MAE</th>
<th>RMSE</th>
<th>MASE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Naive</td>
<td>15.63</td>
<td>88.96</td>
<td>0.95</td>
</tr>
<tr>
<td>ETS</td>
<td>20.00</td>
<td>100.58</td>
<td>0.86</td>
</tr>
<tr>
<td>ARIMA</td>
<td>20.06</td>
<td>101.28</td>
<td>0.87</td>
</tr>
<tr>
<td>SNAIVE</td>
<td>24.67</td>
<td>110.74</td>
<td>0.89</td>
</tr>
<tr>
<td>SETS</td>
<td>25.47</td>
<td>176.88</td>
<td>0.84</td>
</tr>
<tr>
<td>SARIMA</td>
<td>25.13</td>
<td>174.97</td>
<td>0.83</td>
</tr>
<tr>
<td>TBATS</td>
<td>13.30</td>
<td>67.71</td>
<td>0.83</td>
</tr>
<tr>
<td>Fourier</td>
<td>22.19</td>
<td>110.28</td>
<td>0.91</td>
</tr>
<tr>
<td>Bayesian</td>
<td>15.43</td>
<td>75.93</td>
<td>0.85</td>
</tr>
<tr>
<td>TD Forecast</td>
<td>12.58</td>
<td>69.69</td>
<td>0.82</td>
</tr>
<tr>
<td>Mean</td>
<td>20.05</td>
<td>101.28</td>
<td>0.88</td>
</tr>
</tbody>
</table>

(a) Complete series (b) Outlier series

Figure 16

values for a product. Using methods on intermittent demand such as Croston, it might be more easy to conclude what the effects are of these large deals. Besides resizing the series to the correct length, we did not perform any data cleaning on these time series as they rarely have sales periods. The time series consist of maximum 30 sales weeks. We have taken four methods:

1. Tech Data current method as discussed in Section 1.1
2. Croston as discussed in Section 2.8
3. Exponential Smoothing
4. Mean
Figure 17. Mean Absolute Error Intermittent demand

The best method is TD Forecast as we see in figure 17. With an MAE of 15 products, it is almost the same error as it is for series without outliers. However, most series consist for almost all weeks of zero sales. That means there is no possibility of having a reliable forecast for these time series.

We would advise to keep this method. Although certain methods might have a lower forecasting error than the TD forecast method, both are not effective enough to be used.

4.5 Hierarchy

A different option is hierarchical forecasting. Tech Data splits her products into product hierarchies. This sorting of time series is used as hierarchy. To be able to perform this method we use all 132 full length time series of a product hierarchy group. These time series are no further classified. It is not possible to resize the time series as they all need to have the same length to include the hierarchical correlations. We
use the four forecasting methods that are performing the best without hierarchy: Exponential smoothing, ARIMA, TD Forecast and TBATS. Because it would take too much time to perform a time series cross validation, we only forecast the final ten weeks. Unfortunately, this means that the results are only based on ten weeks from 132 different products.

In Figure 18 and Table 5 we see the MAE error for the different forecasting methods with and without hierarchical setting. Methods with an H are hierarchically optimized. The only method that improves using hierarchy is TD Forecast. Because we are not able to perform a cross validation using the hierarchical method, it is impossible to compare these results with the cross validation test. Based on the given figures, it appears that, makes that hierarchy does not create enough advantage. Besides this, the calculation time for the hierarchical forecasts is much higher. Thus, we conclude that the hierarchical forecast are not reliable enough at this moment to be used. However, Tech Data has an extensive data warehouse that can be used to improve this method. In the long future this could be a solution to more exact forecasting.
### Table 5

**Average Error for Hierarchical forecast**

<table>
<thead>
<tr>
<th>Method</th>
<th>MAE</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARIMA</td>
<td>3.88</td>
<td>7.01</td>
</tr>
<tr>
<td>ETS</td>
<td>3.30</td>
<td>6.27</td>
</tr>
<tr>
<td>TDForecast</td>
<td>3.59</td>
<td>6.61</td>
</tr>
<tr>
<td>TBATS</td>
<td>3.60</td>
<td>6.33</td>
</tr>
<tr>
<td>Single</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARIMA</td>
<td>3.65</td>
<td>6.92</td>
</tr>
<tr>
<td>ETS</td>
<td>3.28</td>
<td>6.28</td>
</tr>
<tr>
<td>TDForecast</td>
<td>4.02</td>
<td>6.99</td>
</tr>
<tr>
<td>TBATS</td>
<td>3.32</td>
<td>6.19</td>
</tr>
</tbody>
</table>

4.6 TD Forecast

We think the method of TD Forecast can still be improved. Figure 19 shows the results, where TD2 is the new version. This test includes all 630 time series that we used in this chapter.

This clearly seems an improvement on the current method. We would advise to use this adapted method in time series with not enough correlation to be forecast by the ETS method.
Figure 19. Mean Absolute Error TD Forecast
Chapter 5

Pricing model

In this chapter we will use pricing models that were discussed in Chapter 3, using sales data from Tech Data. This includes the optimization of a single product over a longer period and the optimization of a hierarchical group of products.

We do know that these models could also have been used in series where the information would have been available. However, for this research we make the assumption that there are no weeks without any sales.

5.1 Single item pricing

We work with a single item at Tech Data, with a monopolist position for this product. We know the amount of sales and the selling price for this product for a period of 190 weeks. The model is a discrete version of a dynamic pricing problem since we have weekly sales data. Figure 20 shows the data points. We do not make assumptions for the parameters of our model. Therefore, we use a learning method to learn and estimate the parameters. Although there are multiple approaches to optimize the parameters, we use the linear least squares method that was discussed in Section 3.3. The assumption for this method is that the residuals are normally distributed with $\mu = 0$. By the Central Limit Theorem, we assume that, with enough data points, this assumptions holds. We have sales data of a product. This product is sold for a period of 196 weeks. $t$ is the amount of weeks we include in our prediction. We assume a Poisson arrival process with a parameter
\[ \lambda_t(p) = a - bp_t. \] In which \( p_t \) is the price set for week \( t \). The purchase price is set at \$30. We want to optimize the expected revenue:

\[ E[p_t \Lambda_t(p_t)] \]

Making the assumption that this stochastic equation is still concave, we can simply optimize it. This assumption seems plausible, because the linear arrival function will always have a negative gradient as long as the \( b \) parameter is positive.

**Passive learning**

We use a passive learning method as discussed in algorithm 1. Based on the historic sales data, the optimal price is determined. Using the least linear squares method we find a value for \( a \) and \( b \). We work out the example for \( t = 80 \). For this scenario \( a = 997.6 \) and \( b = 21.5 \). Thus \( \lambda_{80}(p) = 997.6 - 21.5p \). As we try to optimize expected revenue,

\[ E[p_{80} \Lambda_{80}(p_{80})] = p_{80} \lambda_{80}(p_{80}) = 997.6p - 21.5p^2 \]

The optimal \( p = 23.20 \). The problem to this product is that it has sales figures in the price range between \$32 and \$50. Because we use an existing sales list, we do not know the number of sales for the product with the optimal price according to our results. Knowing that the product has a purchase price of \$30, it is not possible to set the price at \$23.
We can also use this model to optimize the profit from the product. The equation becomes:

\[(p_{80} - 30)\lambda_{80}(p_{80}) = 1642.6p - 21.5p^2\]

The optimal \(p = \$38.20\).

If we look at the optimization of the first problem in Table 6 with the 10 final weeks, it appears that the optimal sales price is \$30 as we already determined. However, the expected revenue does change, due to changing parameter values. In the model we have set the minimum price equal to our purchase price. As a result, the optimal price is \$30. For the second problem we also study the optimal price and profit. Table 7 shows the optimal price and the expected profit for every time step for the 10 final weeks. It is not surprising that this time the price is higher than \$30. Looking at the development of the optimal price, we question if there has been enough variability to ensure the optimal solution has been found. Therefore, we also use an active learning policy to study this problem. Although we include the actual sales in this figure, we must emphasize the fact that there are weeks without sales that were removed. Therefore, this figure only gives a small insight into what might be the optimal revenue.

<table>
<thead>
<tr>
<th>t</th>
<th>Sales price</th>
<th>Items sold</th>
<th>Optimal price</th>
<th>Expected revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>181</td>
<td>42.24</td>
<td>6.00</td>
<td>30.00</td>
<td>4677.92</td>
</tr>
<tr>
<td>182</td>
<td>42.24</td>
<td>14.00</td>
<td>30.00</td>
<td>4674.45</td>
</tr>
<tr>
<td>183</td>
<td>42.24</td>
<td>96.00</td>
<td>30.00</td>
<td>4671.83</td>
</tr>
<tr>
<td>184</td>
<td>42.33</td>
<td>53.00</td>
<td>30.00</td>
<td>4677.40</td>
</tr>
<tr>
<td>185</td>
<td>43.67</td>
<td>5.00</td>
<td>30.00</td>
<td>4678.66</td>
</tr>
<tr>
<td>186</td>
<td>45.47</td>
<td>1.00</td>
<td>30.00</td>
<td>4678.44</td>
</tr>
<tr>
<td>187</td>
<td>42.55</td>
<td>1.00</td>
<td>30.00</td>
<td>4679.34</td>
</tr>
<tr>
<td>188</td>
<td>42.55</td>
<td>10.00</td>
<td>30.00</td>
<td>4676.43</td>
</tr>
<tr>
<td>189</td>
<td>34.51</td>
<td>334.00</td>
<td>30.00</td>
<td>4674.28</td>
</tr>
<tr>
<td>190</td>
<td>34.51</td>
<td>333.00</td>
<td>30.00</td>
<td>4800.77</td>
</tr>
</tbody>
</table>

**Active learning**

The active learning method introduced by Den Boer and Zwart, Controlled Variance Pricing, is used to ensure that we are optimizing the global optimum. This means that we use the same linear least squares method as in the passive learning model. However, after finding the optimal price, we determine the variance. If the
Table 7

Profit optimization with passive learning policy

<table>
<thead>
<tr>
<th>t</th>
<th>Sales price</th>
<th>Optimal price</th>
<th>Sales</th>
<th>Expected sales</th>
<th>Profit</th>
<th>Expected profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>181</td>
<td>42.24</td>
<td>38.26</td>
<td>6.00</td>
<td>40.40</td>
<td>73.44</td>
<td>644.00</td>
</tr>
<tr>
<td>182</td>
<td>42.24</td>
<td>38.25</td>
<td>14.00</td>
<td>40.20</td>
<td>171.36</td>
<td>642.60</td>
</tr>
<tr>
<td>183</td>
<td>42.24</td>
<td>38.24</td>
<td>96.00</td>
<td>40.05</td>
<td>1175.04</td>
<td>641.53</td>
</tr>
<tr>
<td>184</td>
<td>42.33</td>
<td>38.26</td>
<td>53.00</td>
<td>39.49</td>
<td>653.64</td>
<td>643.79</td>
</tr>
<tr>
<td>185</td>
<td>43.67</td>
<td>38.26</td>
<td>5.00</td>
<td>26.95</td>
<td>68.35</td>
<td>644.33</td>
</tr>
<tr>
<td>186</td>
<td>45.47</td>
<td>38.25</td>
<td>1.00</td>
<td>9.80</td>
<td>15.47</td>
<td>643.59</td>
</tr>
<tr>
<td>187</td>
<td>42.55</td>
<td>38.25</td>
<td>1.00</td>
<td>37.33</td>
<td>12.55</td>
<td>643.37</td>
</tr>
<tr>
<td>188</td>
<td>42.55</td>
<td>38.24</td>
<td>10.00</td>
<td>37.13</td>
<td>125.50</td>
<td>641.99</td>
</tr>
<tr>
<td>189</td>
<td>34.51</td>
<td>38.23</td>
<td>334.00</td>
<td>113.10</td>
<td>1506.34</td>
<td>640.96</td>
</tr>
<tr>
<td>190</td>
<td>34.51</td>
<td>38.21</td>
<td>333.00</td>
<td>116.10</td>
<td>1501.83</td>
<td>657.28</td>
</tr>
</tbody>
</table>

variance is not large enough we create a bound around the optimal price ensuring enough variation. The formula for this variation is discussed in Section 3.4. The algorithm for this method can be found in the paper by [den Boer & Zwart (2013)].

We will only study the case where we optimize the profit. Because the optimal price of the revenue problem was outside the feasible region, Controlled Variance pricing will have no effect on it. Table [8] shows the results of the first 10 weeks. We see a different pattern in the beginning of this model. Especially, in weeks 10 and 11 we see a different pattern for both policies. However, because we use the same dataset for both models, in the long run the models will converge to the same optimal price as we see in figure [21]. To be sure that this method would have more effect than passive learning, which already has been proven, we should apply both methods on a complete group of products, where we actually adapt the prices. During the time of this research, it was not possible to weekly change the price of some products at Tech Data. Therefore, we work with historical data.

For now, the conclusion is that active learning does appear more reliable as it searches for the optimum with more attention for learning. However, for products with a short life time it is necessary to quickly gain revenues. The active learning method might, in that case, be too much focused on learning in stead of earning. Further research is needed to ensure the value of active learning for Tech Data.

Inventory pricing

We include an inventory parameter in the problem. The inventory cost will be part of the cost function causing a trade-off between revenue making and inventory costs. As this will cause endogenous learning, we will not use an active learning
Table 8

Profit optimization with Active and Passive learning

<table>
<thead>
<tr>
<th>t</th>
<th>Sales price</th>
<th>Items sold</th>
<th>Price Active</th>
<th>Price Passive</th>
<th>Profit Active</th>
<th>Profit Passive</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>45.24</td>
<td>1.00</td>
<td>39.24</td>
<td>39.24</td>
<td>79.05</td>
<td>79.05</td>
</tr>
<tr>
<td>4</td>
<td>46.27</td>
<td>2.00</td>
<td>40.23</td>
<td>39.78</td>
<td>25.25</td>
<td>25.30</td>
</tr>
<tr>
<td>5</td>
<td>45.24</td>
<td>1.00</td>
<td>39.36</td>
<td>39.76</td>
<td>35.64</td>
<td>35.70</td>
</tr>
<tr>
<td>6</td>
<td>45.24</td>
<td>3.00</td>
<td>40.15</td>
<td>40.07</td>
<td>28.10</td>
<td>28.10</td>
</tr>
<tr>
<td>7</td>
<td>46.13</td>
<td>7.00</td>
<td>39.43</td>
<td>39.59</td>
<td>42.82</td>
<td>42.84</td>
</tr>
<tr>
<td>8</td>
<td>46.98</td>
<td>4.00</td>
<td>40.10</td>
<td>40.06</td>
<td>56.55</td>
<td>56.55</td>
</tr>
<tr>
<td>9</td>
<td>47.02</td>
<td>38.00</td>
<td>41.02</td>
<td>41.02</td>
<td>50.92</td>
<td>50.92</td>
</tr>
<tr>
<td>10</td>
<td>46.90</td>
<td>35.00</td>
<td>45.00</td>
<td>50.00</td>
<td>60.25</td>
<td>240.68</td>
</tr>
<tr>
<td>11</td>
<td>46.45</td>
<td>16.00</td>
<td>45.00</td>
<td>50.00</td>
<td>77.24</td>
<td>374.72</td>
</tr>
</tbody>
</table>

method. We assume a service level of 0.95 is required. Furthermore, $h$, a variable that determines the holding cost for a unit per year, is set to 0.05 and the ordering cost for new products is neglected. For this model we assume an expected lead time of one week, $E[L] = 1$, with a standard deviation of one week, $Var[L] = 1$.

The results for this problem can be found in table 9. What we can conclude from these results, is that the product we use for these methods is not expensive enough to cause drastic changes. Products with a higher value than $30 might be more affected by this model. However, the effect is visible. Thus, we can conclude that it is useful to optimize prices based on inventory information. Tech Data should also focus on dynamic pricing with perishable items. Products that perish cause sudden losses that could have been prevented if the pricing had been better. Including this in the model could lead to severe cost savings.

5.2 Multi item pricing

Considering a hierarchy with $n \in \mathbb{N}$ products and product $k = 1...n$. These $n$ products are sold during a fixed life time with sales per week. In this model we use a dataset of 571 different products with 104 weeks of data. Based on the weeks with sales up until week $t$, we determine the linear demand function using the least squares method. This only happens with products that have actual sales of more than ten weeks, to ensure the validity of the function.
Algorithm 2 Inventory pricing Policy

Initialize. Set initial price $p_1, p_2$. Determine $p_{\text{max}}$, $h$, service level $sl$, starting inventory $q_1$, $E[L]$ and $Var[L]$. 

For $t \geq 2$:

Estimate. Determine the optimal parameters $\hat{a}_{t+1}, \hat{b}_{t+1}$ by using Least Linear Squares.

Update. If $\frac{\hat{a}_{t+1}}{2\hat{b}_{t+1}} \geq p_{\text{max}}$, set $p_{t+1} = p_{\text{max}}$

Elseif $\frac{\hat{a}_{t+1}}{2\hat{b}_{t+1}} \leq p_{\text{purchase}}$, set $p_{t+1} = p_{\text{purchase}}$

Else $p_{t+1} = \frac{\hat{a}_{t+1}}{2\hat{b}_{t+1}}$

Inventory. Calculate the $E[D]$ and $Var[D]$.

$SS = z_{sl} \sqrt{(E[L] * \sigma_L^2)(E[D] * \sigma_D^2)}$

$ROL = E[L] * E[D] + SS$

If $q_t \leq ROL$, $q_t = ROL$.

Figure 21. Optimal price according to active learning policy
Table 9

*Profit optimization with Inventory and Passive learning*

<table>
<thead>
<tr>
<th>t</th>
<th>Price Inventory</th>
<th>Price Passive</th>
<th>Profit Inventory</th>
<th>Profit Passive</th>
</tr>
</thead>
<tbody>
<tr>
<td>180</td>
<td>38.26</td>
<td>38.27</td>
<td>641.55</td>
<td>645.60</td>
</tr>
<tr>
<td>181</td>
<td>38.25</td>
<td>38.26</td>
<td>639.97</td>
<td>644.00</td>
</tr>
<tr>
<td>182</td>
<td>38.23</td>
<td>38.25</td>
<td>638.69</td>
<td>642.60</td>
</tr>
<tr>
<td>183</td>
<td>38.22</td>
<td>38.24</td>
<td>637.87</td>
<td>641.53</td>
</tr>
<tr>
<td>184</td>
<td>38.24</td>
<td>38.26</td>
<td>642.51</td>
<td>643.79</td>
</tr>
<tr>
<td>185</td>
<td>38.25</td>
<td>38.26</td>
<td>641.81</td>
<td>644.33</td>
</tr>
<tr>
<td>186</td>
<td>38.24</td>
<td>38.25</td>
<td>639.70</td>
<td>643.59</td>
</tr>
<tr>
<td>187</td>
<td>38.24</td>
<td>38.25</td>
<td>639.37</td>
<td>643.37</td>
</tr>
<tr>
<td>188</td>
<td>38.22</td>
<td>38.24</td>
<td>637.99</td>
<td>641.99</td>
</tr>
<tr>
<td>189</td>
<td>38.21</td>
<td>38.23</td>
<td>637.23</td>
<td>640.96</td>
</tr>
<tr>
<td>190</td>
<td>38.20</td>
<td>38.21</td>
<td>662.97</td>
<td>657.28</td>
</tr>
</tbody>
</table>

Because the AC price is a given parameter in this model we optimize the percentual layer dynamically. In this model we cannot assume the concavity of the linear functions for all products. However, because we see this problem as an optimization problem over all products it is not necessary for all functions to be concave. We use the same bounds as in Algorithm to ensure the price will be within a normal range. The advantage of this multi-product optimization is that the correlation between the products is included in the model. If a percentage increase of a certain product causes the reduce of sales for a different product, this will be incurred in the next optimization.

\[
(\hat{a}_{k,i+1}, \hat{b}_{k,i+1}) = \underset{a,b}{\arg \min} \left\{ \sum_{t=1}^{i} D_{k,t} - (a - b \frac{AC_{k,t}}{1 - p})^2 \right\}
\]

\[
p_{i+1} = \underset{p}{\arg \max} \left\{ \sum_{k=1}^{n} ((\hat{a}_{k,i+1} - \hat{b}_{k,i+1}) \frac{AC_{k,i}}{1 - p} (\frac{AC_{k,i}}{1 - p} - AC_{k,i})) \right\}
\]

We use a function in R that uses a combination of golden section search and parabolic interpolation to find the optimal value in the specified interval. Figures 22 and 23 show the optimal percentage per week and the expected revenue. This expected revenue does not only increase because of the optimization. After more weeks, there are more products that meet the conditions. They are added to the optimization model.
Figure 22. Optimal mark up percentage for a multi product passive learning policy

We think it would be very useful to use this method for certain product groups within Tech Data. Preferably, this group would be smaller consumer products as they have a more constant and continuous sales rate.
Figure 23. Expected Revenue for a multi product passive learning policy
Chapter 6

Conclusions and recommendations

In this thesis we have conducted research on two methods to improve the revenue management in the company Tech Data and thus decrease the expected depreciation on products. Both methods have indicated possibilities for Tech Data to improve their company methods. In this chapter we summarize the conclusions from both methods and also recommend possible implementations. Furthermore, we point out what directions can be researched to further improve the results from this thesis.

6.1 Demand forecast

The simplicity of the TD forecast method has shown its value in this research. We have invested a lot of time in finding methods that have a mathematical basis to be more reliable in forecasting. However, the simplicity of this particular method makes it reliable for all sorts of time series. During this research we did find methods that have smaller error terms and were mathematically more substantiated for certain sets of products. However, the amount of difficult time series with intermittent demand or little correlations are too much. Still there are some recommendations we would like to provide Tech Data.

Although the TD Forecast method performed very well in most of the time series, we have shown that it can be improved. We would advise to remove \[ \text{maximum and the minimum.} \]

Even though it is a small percentage of all products, there are time series with enough correlation to be forecast by a method such as ARIMA or ETS. It is exactly these products where most revenues can be made, and therefore it is important to
have an accurate forecast, resulting in enough stock.

The value of data cleaning has been demonstrated. The Mean Absolute Error was higher for every method we have studied without cleaning. Therefore, we would advise to use a simple data cleaning before forecasting. The introduced method of outlier correction is very robust and simple. It is very easy to implement and already will lead to significant forecasting improvements.

Seasonality and Hierarchical methods were both not strong enough to create significantly better forecasts. We mostly base this on the irregular demand for many products. This irregular demand makes that there are no clear patterns, especially on the longer term. The only scenario in which hierarchy could be useful, is to be used for uncorrelated series for which we advise to use TD forecast.

The forecasting of intermittent demand is still too unreliable to be practically used. Until now, Tech Data supplies its customers for these products with [89x729]. We advise the company to keep this strategy.

Further research in this direction should be focussed on handling big data. The difficulty for companies, such as Tech Data, is finding opportunities from all these large data sets. However, with the size of Tech Data, models such as hierarchical forecasting and neural networks are useful to study.

Furthermore, although we have researched the optimal forecasting methods, it is just as important to determine which time series should be judged as either intermittent, or uncorrelated demand or correlated demand. This can be further researched within a company as Tech Data.

6.2 Pricing

Dynamic pricing is a completely new path for Tech Data. Introducing new things in a running company is never easy and that is why we had to conduct research on existing data sets. However, the positive effect of dynamic pricing has been shown in the different methods that we have tested using the data. Although the method worked perfectly for the products we used in our study, there are many products at Tech Data for which this model is not feasible. Before Tech Data would think of using dynamic pricing, it should be studied for which products it is valuable. This also holds for the inclusion of inventory. The effect of holding cost seemed negligible. However, there might be a set of products where it is very effective to include the holding cost in the revenue problem.

The value of active learning has been difficult to prove in this studies. However, we think that for multi-product problems where there are much more variables to be optimized, it is not necessary to use an additional condition focussed on learning.

The method that has proven to be very applicable for Tech Data, is the multi-
item method in which we optimize complete product groups in stead of single items. This method is based on the current way of pricing in Tech Data and could therefore be more easily implemented. Furthermore, the effects of optimizing such a large group are easier to control.

Further research should be done in what methods are useful for Tech Data besides linear least squares. The disadvantage of linear least squares is the assumption of normally distributed residuals. For maximum likelihood this is not necessary. However, in a small test we conducted, using maximum likelihood, it has proven difficult to improve the results.

Secondly, research should be done in the direction of perishability. The items of Tech Data lose much of their value at the moment that the vendor introduces a new model. That means that we hope to predict the ‘life time’ of a product to determine the price and inventory levels.

Thirdly, batches should be included in these models. Currently, there is only little and young research in this direction, but the value of a working model would be attractive to Tech Data as a distributor often gets demand in batches. Finally, we think there could be a new field of research that attempts to connect the two different fields that were discussed in this thesis. One could build a model in which the demand is estimated on price and afterwards the residuals are tested for autocorrelations. If we think of the model in Section 2.5, it might be able to begin with a demand function based on prices with the residuals being used in an ARIMA model.
Appendix A

R scripts

During the practical research in chapters 4 and 5 we used the statistical program R to use the data from Tech Data and research the different methods. The scripts that have been used are available on the following link.

https://www.dropbox.com/sh/8j9b1x4ymq9qku/AACBLkYhNKhVWbRmXmg1oAPSa?dl=0

The datasets that were used in chapters 4 and 5 are still available for the authors. In consultation with Tech Data, the datasets can be used to reproduce the results.
References


