AALBORG UNIVERSITY

AALBORG UNIVERSITY
DENMARK

Backorder Prediction Using Machine Learning For Danish Craft Beer Breweries

by

Yuqi Li

A thesis submitted in partial fulfillment for the degree of Master of Science in Technology

in the
Global Systems Design
School of Engineering and Science

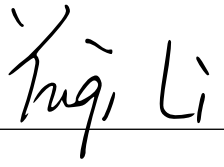
Supervised by
Lazaros Nalpantidis

September 2017

Declaration of Authorship

I, Yuqi Li, declare that this thesis titled, 'Backorder Prediction Using Machine Learning For Danish Craft Beer Breweries' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: 

Date: 02/09/2017

AALBORG UNIVERSITY

Abstract

Global Systems Design
School of Engineering and Science

Master of Science in Technology

by Yuqi Li

As a result of the tax reduction on soft drinks and beers in Denmark, the demand for craft beers had a major growth. A problem might appear to the brewers when the request of quality beers rises; the experience-based prediction could run short of ways in dealing with the situation that they cannot meet the demand. With limited production capacity, an accurate backorder forecasting can be essential for the producers. To foresee many future random orders, a more advanced prediction method can do a favor.

With a machine learning prediction approach, this report discusses how to use machine learning tools to predict future backorders based on producers' historical data. The report will give a brief introduction to Danish beer history and present the situation, explain how to process historical data, create a machine learning module and provide forecasting recommendations. A dataset of relevant data from open source was considered. The final algorithm will be tested and evaluated on this dataset. It is expected that the outcomes of this work will be of value for the future of Danish brewery.

Contents

Declaration of Authorship i

Abstract ii

List of Figures v

List of Tables vii

1 Introduction 1
- 1.1 Introduction to the project background . . . 1
  - 1.1.1 Danish craft breweries . . . 1
  - 1.1.2 Backorder and Bullwhip effect . . . 4
  - 1.1.3 Target scenario . . . 5
- 1.2 Research questions . . . 6
- 1.3 Limitations and Delimitations . . . 7

2 Theoretical background 8
- 2.1 Data issues . . . 8
  - 2.1.1 Inaccurate values . . . 8
  - 2.1.2 Missing value . . . 8
  - 2.1.3 Outlier . . . 9
- 2.2 Imbalanced dataset . . . 9
  - 2.2.1 Data sampling . . . 9
- 2.3 Dimensionality reduction . . . 9
  - 2.3.1 Principle component analysis . . . 10
- 2.4 Machine learning . . . 11
  - 2.4.1 Supervised machine learning . . . 11
  - 2.4.2 Unsupervised machine learning . . . 12
- 2.5 Naive bayes . . . 13
- 2.6 Support vector machine . . . 13
  - 2.6.1 Kernel . . . 14
- 2.7 Decision Tree . . . 15
- 2.8 K nearest neighbours . . . 15
- 2.9 Ensemble Learning . . . 16
- 2.10 Evaluation on the classifiers . . . 17
  - 2.10.1 K-fold cross validation . . . 17

2.10.2 Sensitivity and Specificity . . . 18
2.10.3 Area under the curve . . . 19

3 Literature review **20**
3.1 The background . . . 20
3.2 Dealing with Data Issues . . . 21
3.3 Issues while tuning the classifiers . . . 22
3.4 Similar works . . . 23

4 Proposed solution **25**
4.1 Introduction to the solution . . . 25
4.2 Data pre-processing . . . 27
4.2.1 Introduction to the dataset . . . 27
4.2.2 Binarization . . . 28
4.2.3 Summary statistics . . . 29
4.3 Data Visualization . . . 30
4.4 Data issues . . . 32
4.4.1 Data imbalance . . . 32
4.4.2 Missing values . . . 33
4.4.3 Data redundancy . . . 34
4.4.4 Dimentionality Reduction and Outlier detection . . . 35
4.5 Data Modeling . . . 37
4.5.1 Design of experiments . . . 37
4.5.2 Definition of training set . . . 38
4.5.3 Algorithm selection . . . 39

5 Results and Evaluation **43**
5.1 Introduction to testing set . . . 43
5.2 Results . . . 43
5.2.1 Training set one: Cleaned historical data . . . 44
5.2.2 Training set two: Balanced data . . . 46
5.3 Evaluation . . . 49

6 Conclusion and further discussion **51**

Bibliography **54**

A1. Description of the dataset **58**

A2. Coding for data preparation **59**

A3. Interview **62**

List of Figures

1.1 Number of Danish breweries from 2000 to 2015, sourced from Danske Ølentusiaster and Bryggeriforeningen (Bryggeriforeningen, known as the Danish Brewers Association, is the trade organisation and common voice for breweries and producers of soft drink in Denmark. It represents the beer and soft drink industries towards the parliament, government, media and the public regarding political issues like taxation, environment, food and health) [1] . . . 2
1.2 Danish beer market sales, from 2003 until 2015, the annual Danish beer sales remaining decreasing (chart adapted form [1]) . . . 2
1.3 The growth on new beers in Danish market from 2010-2015 . . . 3
1.4 Danish beer market geography segmentation(Adapted from marketline.com) 3
1.5 An example of bullwhip effect . . . 4

2.1 A classification problem where observations (the points) and class labels (the colours) are given and the goal is to come up with a rule for determining which class a point belongs to (one such rule is indicated by the lines). The rule can then be applied to new points [2]. . . . 11
2.2 A one-dimensional regression problem which predicts the y-values based on the x-values. The titted regression model is indicated by the black line [2]. . . . 12
2.3 A 2D clustering example. A clustering is given a dataset (here the 2D dataset shown in the left-hand pane) and has to estimate plausible divisions of the observations into clusters as indicated in the right-hand pane [2]. . . . 12
2.4 SVM hyperplane . . . 14
2.5 Illustration of a two-dimensional input space that has been partitioned into five regions using axis-aligned boundaries. θ_1 is a parameter of the model[3] . . . 15
2.6 k-nearest-neighbor classifiers applied a simulation data. The broken purple curve in the background is the Bayes decision boundary. [4] . . . 16
2.7 k-nearest-neighbor classifiers applied a simulation data. The broken purple curve in the background is the Bayes decision boundary. [4] . . . 17
2.8 An example of AUC computed while training a classifier . . . 19

4.1 Machine learning work flow [5] . . . 26
4.2 Project work flow . . . 26
4.3 A matrix of scatter plots of features combinations. Each small plot has figures beside which indicates the data range from the plotting feature . . . 30

4.4 Boxplot of each column of the data. X represents each feature, Y represents their values. 31
4.5 Mesh plot of all features from the historical data, where X represents the numbers of features, Y represents number of entries, and Z represents the value of each (X,Y). Mesh(X,Y,Z) draws a wire frame mesh with color determined by Z, so color is proportional to surface height. If X and Y are vectors, length(X) = n and length(Y) = m, where [m,n] = size(Z) [6]. 31
4.6 A pie chart of proportion of the two predicting classes 32
4.7 SVM Gaussian model accuracy plot.The accuracy is decreasing as the number of neighbour increases. Kernel scale: the smaller the better. . . . 40
4.8 K nearest neighbour model accuracy plot. The accuracy is decreasing as the number of neighbour increases. 41
4.9 Ensemble subspace KNN model accuracy plot. When the number of learners ranging from 5 to 20, KNN gets the best accuracy when the number of learners is eight, which is 83.1% 42

5.1 Scatter plot of two features of training set two after dimensionality reduction. Red indicates positive values and blue indicates negative values . 46
5.2 Confusion matrix from the traditional way to predict backorders 49

1 Interview questionnaire 1/2 63
2 Interview questionnaire 2/2 64

List of Tables

2.1 Terminology and derivations from a confusion matrix 18

4.1 Attributes from the historical data, a short explanation to them and their data types 28
4.2 Attributes from the historical data, following with summary statistics . . 29
4.3 Binary attributes from the historical data, following with numbers of true class in each attribute 29
4.4 Simple ways to detect outliers [6] 36
4.5 Results of the classifiers screening, the top three classiders are: KNN, Ensemble KNN and SVM Gaussian 39

5.1 Testing set for Training set one 43
5.2 Training set one 44
5.3 Confusion matrix for training set one with outliers 44
5.4 Sensitivity and Specificity for training set one with outliers 44
5.5 Confusion matrix value for training set one without outliers 45
5.6 Sensitivity and Specificity for training set one without outliers 45
5.7 Training set two 46
5.8 Confusion matrix for training set two with outliers, before PCA 47
5.9 Sensitivity and Specificity for training set two with outliers, before PCA . 47
5.10 Confusion matrix for training set two with outliers, after PCA 47
5.11 Sensitivity and Specificity for training set two with outliers, after PCA . . 47
5.12 Confusion matrix for training set two without outlier, before PCA 48
5.13 Sensitivity and Specificity for training set two without outlier, before PCA 48
5.14 Confusion matrix for training set two without outlier, after PCA 48
5.15 Sensitivity and Specificity for training set two without outlier, after PCA 48
5.16 Prediction performance from the traditional way to predict backorders . . 49

Chapter 1

Introduction

1.1 Introduction to the project background

1.1.1 Danish craft breweries

This section will give a brief introduction to the background of Danish craft beer brewing, which will support the following section Target scenario for a better understanding

Historically in Denmark, there were ups and downs in the brewing industry. During the 19th century, there were almost one thousand breweries, then the number reduced through closures, mergers and acquisitions, and at the dawn of the new millennium, there were only 18 breweries left in Denmark. However, from 2005 there was an explosion of new breweries opening throughout the country (see figure 1.1), soon there was hardly a spot on the map of Denmark without a local brewery [1]. Despite from brewery that had mass production on standard taste beers, craft beers are mostly considered to have more taste and better quality [7]. The definition for craft brewery is very blurry, and the concept of a micro brewery was once virtually unknown in Denmark. In America, an American craft brewery is

Small. Annual production of 6 million barrels of beer or less.

Independent. Less than 25 percent of the craft brewery is owned or controlled (or equivalent economic interest) by a beverage alcohol industry member that is not a craft brewer itself.

Traditional. A brewer that has a majority of its total beverage volume in beers, whose flavour derives from conventional or innovative brewing ingredients and their fermentation [8].

Though data shows that the beer consumption had decreased from 5.9 in 1980 to 4.0 in 2006, the business for Danish craft breweries has had a trend of growth the recent years. Figure 1.2 shows from 2003 until 2015, the annual Danish beer sales remain-

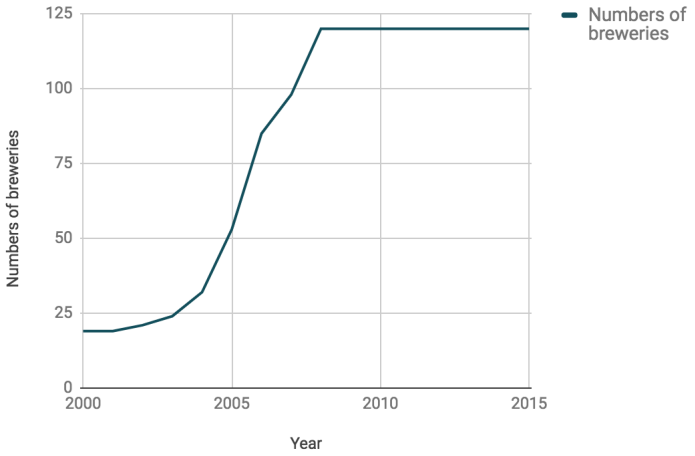


FIGURE 1.1: Number of Danish breweries from 2000 to 2015, sourced from Danske Ølentusiaster and Bryggeriforeningen (Bryggeriforeningen, known as the Danish Brewers Association, is the trade organisation and common voice for breweries and producers of soft drink in Denmark. It represents the beer and soft drink industries towards the parliament, government, media and the public regarding political issues like taxation, environment, food and health) [1]

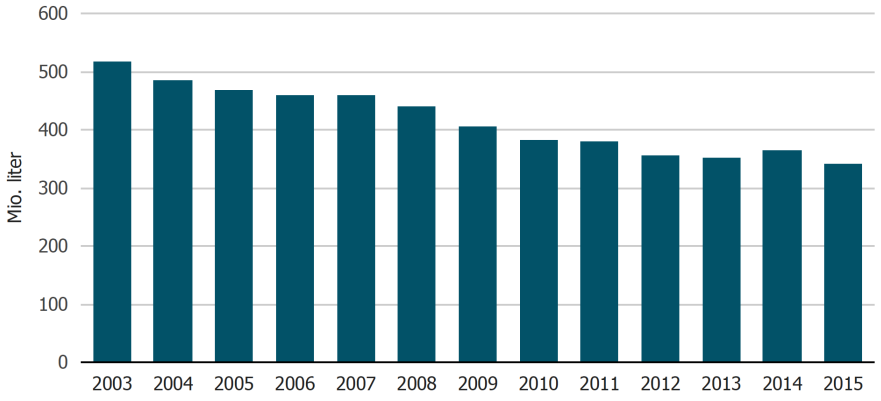


FIGURE 1.2: Danish beer market sales, from 2003 until 2015, the annual Danish beer sales remaining decreasing (chart adapted form [1])

ing decreasing. However, regardless of the decreasing sale and consumption of Danish beers, a growing interest in beer has been noticed in Denmark late years. Result from increasing demand; more craft beers appear on the market. Figure 1.3 shows some new beers appear in the Danish market from 2010-2015, by now, the number is still growing [1].

The awareness of beer quality and the applicability of beers in different contexts have increased among consumers. A good example of the growing interest is the beer organisation "Danske Ølentusiaster/Danish Beer Enthusiasts (DE)". In less than six years, more than 9000 members have joined DE making this organisation the second largest beer interest association in Europe only outnumbered by the British organisation Campaign for Real Ale (CAMRA) [7]. Danish craft beers are getting more market share,

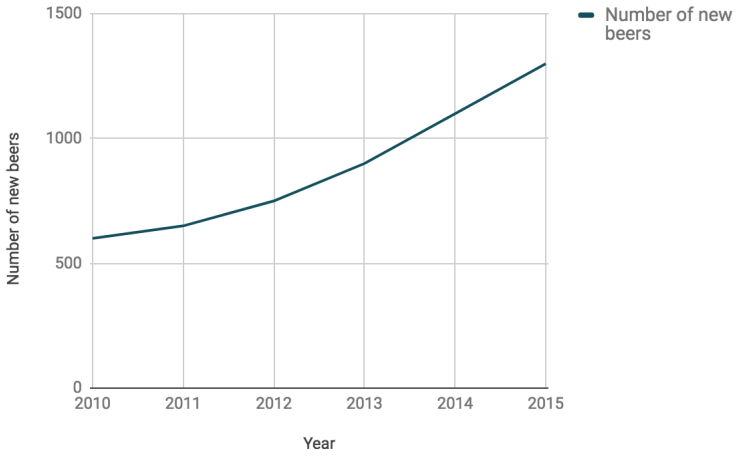


FIGURE 1.3: The growth on new beers in Danish market from 2010-2015

although the leading roles of the beer market are still big brewing companies like Carlsberg, small breweries' got a positive potential.

The Danish craft beer breweries are small scaled, but their consumers are not only in Denmark. Due to market research from Markerline.com (Figure 1.4), 64.4% of the customers locate in the rest of Europe(2013), which means many breweries need to handle business and supply chain issues cross the border.

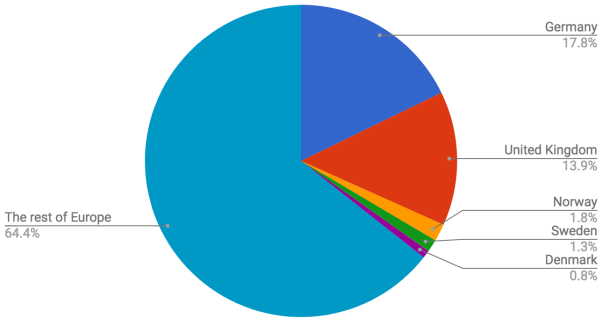


FIGURE 1.4: Danish beer market geography segmentation(Adapted from marketline.com)

1.1.2 Backorder and Bullwhip effect

The bullwhip effect causes members of the supply chain to overreact to changes in demand at the retail level. Minor demand changes at the consumer level may result in large ones at the supplier level. Bullwhip fluctuations create unstable production schedules, resulting in an expensive capacity change adjustments such as overtime, subcontracting, extra inventory, backorders, hiring and laying off of workers, equipment additions, under-utilization, longer lead times, or obsolescence of over-produced items [9].

Figure 1.5 shows an example of the interaction of stakeholders in the supply chain. An

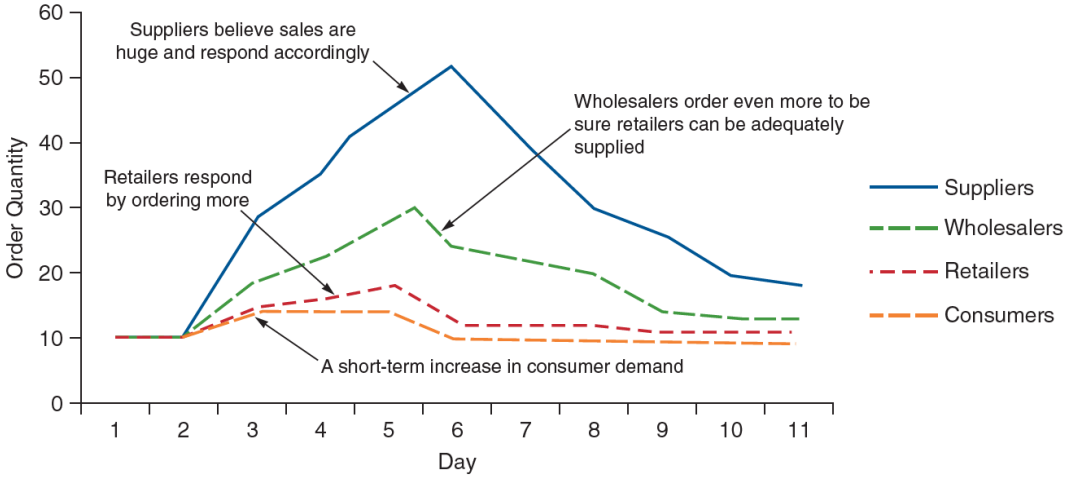


FIGURE 1.5: An example of bullwhip effect

organisation can shift demand fulfilment to other periods by allowing back orders. That is, orders are taken in one period, and deliveries promised for a later time. The success of this approach depends on how willing customers are to wait for delivery. Moreover, the costs associated with backorders can be difficult to pin down since they would include lost sales, annoyed or disappointed customers, and perhaps additional paperwork [10].

Most of the experimental studies in literature conducted by Sterman [11], Croson and Donohue [12], Croson and Donohue [13], Wu and Katok [14] as well as Cantor and Katok [15] are under backorder settings. However, backorder creates variation in orders as the replenishment quantity varies considerably in backorder cases. This variation in replenishment quantity may get reflected while placing orders. Hence, we can say that backorder is also one of the contributing factors to bullwhip effect [16].

So solving backorder problems and giving more satisfaction to customers becomes critical in the supply chain and the solution to the problem could be more precise forecasting. When a business can foresee the incoming orders, have more time to purchase and produce, the backorder will no longer be a very thorny issue.

1.1.3 Target scenario

Nowadays, craft beers produced by microbreweries have contributed more in Danish beers global turnover, which brings up the question: For small scaled breweries whose international market scale is larger than the local market, whether can they react fast enough to the market, to ensure a robust supply chain when their demand rises.

A typical Danish craft beer brewer, Amager bryhus, was interviewed by phone. From the discussion, several facts were discovered. Late years, the increasing demand of craft beers has made the company kind of passive, many times they cannot meet demand and they are actually opening a second brewery. The company is running by a scale of a few beer enthusiasts, and when they have to decide the future productions they just follow their guts based on previous experiences, it is not known how they use knowledge to predict. Details of the interview can be found in appendix A3.

Despite the talk with Amager bryhus, a few other microbreweries were also researched on. The interviews show that those breweries are producing traditionally, maybe their scale is not large enough to afford professional and advanced tools for their daily operations and sales forecasting.

As mentioned from the Danish beer brewing industry background, the demand for craft beers is rising by years. For producers, one of the primary purposes of supply chain collaboration is to improve the accuracy of forecasts [17], especially international business. In this case, when the brewery needs to collaborate with foreign distributors and retailers, many issues might appear due to the lack of accurate predictions.

In this case, considering customer satisfaction and demand, the producers need to be well prepared to ensure a robust and fast supply chain. With the help of advanced predictions, the business will have time to react before any unexpected event occurs and therefore help the producers be more intelligent on stocking and become more proactive.

1.2 Research questions

By realising the potential issue, this project is tempting to find out the best classifier for backorder predictions. Results will be compared with other classifiers in the evaluation. The research question is:

1. How can Danish brewers use machine learning techniques, train on historical data, predict backorders in their early stage of the supply chain?

2. Are machine learning predictions more efficient than the traditional forecasting?

Details of data requirements and machine learning methods will be explained, and basic terminology will be provided as well.

1.3 Limitations and Delimitations

Machine learning for predictions is a popular trend nowadays. There are great papers and works show new tools and new concept to solve outstanding issues. The work flow has many sub processes where various methods could be used to get the best forecasting performance. During literature research, many fascinating research works are showing one individual issue could be solved in another way, by another method which gives a better result. In this project, many methods could be used to address data issue for the manner of better results, but due to the time limit it is restricted to simple methods for the convenient output rather than trying out complex advanced models to get better output.
It 's hard to get data from the Danish craft beer breweries, some of them do not have stored and managed data. Possible reasons could be data security concern and time issue. So it is relevant data, which was found from open source for study, that was used in the analysis.

Based on the literature review, in order to get the best results out from limited time, restrictions were set in the project. A vast amount of solutions and methods that could have been tried out to this problem, however learning from the previous researching experience, a few of the solutions are recognized as the best to try. Resulting from he time limitation of the project duration, in this project, methods that could be processed with shorter time have been chosen for analysis.

Lacking realistic data in the business, will of course limit the results in the data pre-processing, but since it is a proof of a concept, relevant data could do the same. It can help prove the same of the idea, but of course, the hidden issues couldn't be addressed without using realistic data. So the goal of this project is not to solve a real brewery's backorder problem, but research on the methods and theories that can be used to solve such a problem.
Information about Danish beer market is an introduction to the background of this project; it cannot be viewed as a Danish market research.
Supply chain and its bullwhip effect were mentioned in the report, but there won't be broad discussion about how to solve such a problem.
The formulated problem in this project is to figure out if a product is going to be on backorder or not, which is a typical binary classification problem. Several classifiers will be used in the training, and the reason for the choice is for the convenience of the training and getting outputs.

Chapter 2

Theoretical background

This chapter will briefly introduce the tools and terminologies that were used or mentioned in the project.

2.1 Data issues

While converting a dataset to a certain format, there are some practical points to be aware. Real data are often low in quality, and careful checking a process that has become known as data cleaning, which pays off many times over [18].

2.1.1 Inaccurate values

Typographic errors in a dataset will obviously lead to incorrect values. Typographical or measurement errors in numeric values cause outliers that can be detected by graphing one variable at a time. Erroneous values often deviate significantly from the pattern that is apparent in the remaining values. Sometimes, however, inaccurate values are hard to find, particularly without specialist domain knowledge [18].

2.1.2 Missing value

Missing values are frequently indicated by out-of-range entries; perhaps a negative number (e.g., -1) in a numeric field that is normally only positive, or a 0 in a numeric field that can never normally be 0. For nominal attributes, missing values may be indicated by blanks or dashes. Sometimes different kinds of missing values are distinguished (e.g., unknown versus unrecorded versus irrelevant values) and perhaps represented by different negative integers [18].

2.1.3 Outlier

An outlier is defined as a dissimilar data point that is far from the rest of the data. It normally explains the abnormal behaviour of a feature. Outliers, being the most extreme observations, may include the sample maximum or sample minimum, or both, depending on whether they are extremely high or low. However, the sample maximum and minimum are not always outliers because they may not be unusually far from other observations.[4]

2.2 Imbalanced dataset

The class imbalance problem is one of the (relatively) new problems that emerged when machine learning matured from an embryonic science to an applied technology, amply used in the worlds of business, industry and scientific research. The class imbalance problem typically occurs when, in a classification problem, there are many more instances of some classes than others. In such cases, standard classifiers tend to be overwhelmed by the large classes and ignore the small ones. In practical applications, the ratio of the small to the large classes can be drastic such as 1 to 100, 1 to 1,000, or 1 to 10,000 (and sometimes even more) [19].

2.2.1 Data sampling

Sampling methods seem to be the dominant type of approach in the community as they straightforwardly tackle imbalanced learning. In general, the use of sampling methods in imbalanced learning consists of the modification of an imbalanced dataset by some mechanism to provide a balanced distribution. The key aspect of sampling methods is the mechanism used to sample the original dataset. Under different assumptions and with various objective considerations, various approaches have been proposed.

2.3 Dimensionality reduction

Dimensionality reduction is a way to discover a simpler representation of a very high-dimensional dataset. In machine learning and statistics, dimensionality reduction or dimension reduction is the process of reducing the number of random variables under consideration, via obtaining a set of principal variables [20]. Linear subspace analysis for feature extraction and dimensionality reduction has been studied in depth for a long time, and many methods have been proposed in the literature, including principle

component analysis (PCA) [21], linear discriminant analysis (LDA) [22], null space LDA (NLDA) etc. Though applied very successfully for pattern classification, these methods usually miss out some discriminant information while extracting relevant features for the classification task [23]. In this project, mainly principle component analysis was used for dimensionality reduction.

2.3.1 Principle component analysis

Principal component analysis, or PCA, is a technique that is widely used for applications such as dimensionality reduction, lossy data compression, feature extraction, and data visualisation [24].
There are two commonly used definitions of PCA that give rise to the same algorithm. PCA can be defined as the orthogonal projection of the data onto a lower dimensional linear space, known as the principal subspace, such that the variance of the projected data is maximised [25]. Equivalently, it can be defined as the linear projection that minimises the average projection cost, defined as the mean squared distance between the data points and their projections [26] [3].

The simplest way to transform vectors from a M dimensional space to a n <= M-dimensional space is to select an orthonormal basis v_1,...,v_n of a n-dimensional subspace V and define each b_i as the projection of x_i onto V. The general layout of the PCA algorithm is:

1. Compute the mean $m = \frac{1}{N} \sum_{i=1}^{N} x_i$
2. Subtract the mean from x_i: $X_i = x_i - m$
3. Project onto V: $b_i^T = X_i^T V$ [2]

2.4 Machine learning

How to build an intelligent machine? More than 65 years ago Alan Turing made this question the subject of his famous essay "Computing machinery and intelligence"[2] Turing proposed that instead of writing a computer program that behaves like a human from scratch, we should build a machine which initially cannot do a great many things however, which can learn from experience.

Machine learning is the implementation of Turing's idea: The study of algorithms which can learn to do interesting things. The learning is based on observed data, whether from a spreadsheet, a sensor attached to a robot or human instructions. The goal of machine learning is to improve at some task in the future based on the experience. The science of learning plays a key role in the fields of statistics, data mining and artificial intelligence, intersecting with areas of engineering and other disciplines [4]. Generally, there are two types of machine learning, supervised machine learning and unsupervised machine learning.

2.4.1 Supervised machine learning

In supervised machine learning, the task is to predict a quantity based on other quantities. It is useful to distinguish between classification and regression that fall into the category of supervised machine learning: In classification, with given observed values x and have to predict a discrete response y. Figure 2.1 shows an expample of classification.

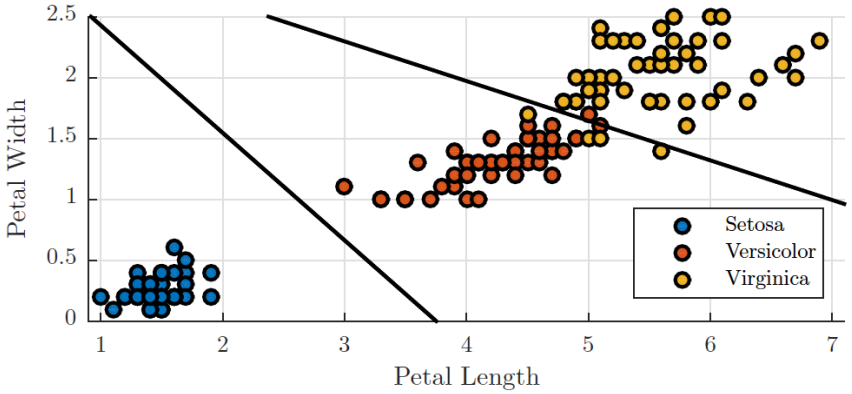


FIGURE 2.1: A classification problem where observations (the points) and class labels (the colours) are given and the goal is to come up with a rule for determining which class a point belongs to (one such rule is indicated by the lines). The rule can then be applied to new points [2].

In regression problems, we are given observed values x and have to predict a continuous response y. Figure 2.2 shows an example of regression. The distinction in output type has

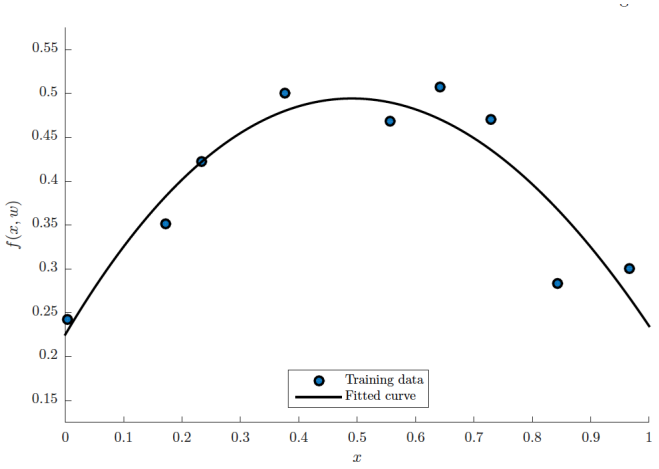


FIGURE 2.2: A one-dimensional regression problem which predicts the y-values based on the x-values. The titted regression model is indicated by the black line [2].

led to a naming convention for the prediction tasks: regression for predicting quantitative outputs, and classification for predicting qualitative outputs [4].

2.4.2 Unsupervised machine learning

Apart from supervised learning is unsupervised learning. Unsupervised learning tries to solve this and similar problems where we do not have access to any "ground-truth" label information (such as the identity of the animal in the image) but we try to discover this labelling from the data alone.
Figure 2.3 shows an example of clustered data points.

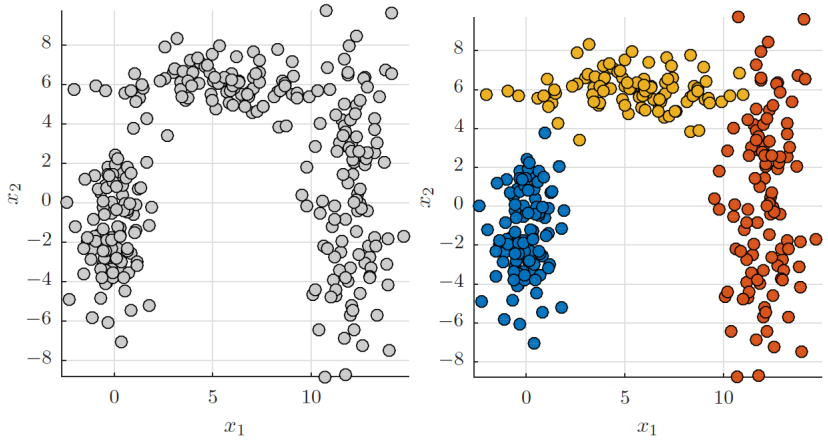


FIGURE 2.3: A 2D clustering example. A clustering is given a dataset (here the 2D dataset shown in the left-hand pane) and has to estimate plausible divisions of the observations into clusters as indicated in the right-hand pane [2].

In this project, classification method would be tries out to predict a binary target value, the backorders. Common method to deal with classification problems are: Tree based methods, nearest neighbour methods, Bayesian method and neural network.

2.5 Naive bayes

A related graphical structure arises in an approach to classification called the naive Bayes model, in which we use conditional independence assumptions to simplify the model structure. Suppose our observed variable consists of a D-dimensional vector $x = (x_1, ..., x_D)^T$, and we wish to assign observed values of x to one of K classes. Using the 1-of-K encoding scheme, we can represent these classes by a K-dimensional binary vector z. We can then define a generative model by introducing a multinomial prior$p(z|\mu)$ over the class labels, where the k^{th} component μk of μ is the prior probability of class C_k, together with a conditional distribution$p(x|z)$ for the observed vector x. The principal assumption of the naive Bayes model is that, conditioned on the class z, the distributions of the input variables $x_1, ..., x_D$ are independent. The graphical representation of this model is shown in Figure 8.24. We see that observation of z blocks the path between x_i and x_j for j $\neq$ i (because such paths are tail-to-tail at the node z) and so x_i and x_j are conditionally independent given z. If, however, we marginalise out z (so that z is unobserved) the tail-to-tail path from x_i and x_j is no longer blocked, which tells that, in general, the marginal density p(x) will not factorize with respect to the components of x [3].

2.6 Support vector machine

Support vector machines (SVMs), produces nonlinear boundaries by constructing a linear boundary in a large, transformed version of the feature space [4].
Support vector machines also have an advantage when nonlinear optimisation is involved in the training, and the objective function is convex, the solution of the optimisation problem will be relatively straightforward. The number of basis functions in the resulting models is much smaller than the number of training points, although it is usually still relatively large and typically increases along with the size of the training set [3].
Figure 2.4 shows a hyperplane divides data points from two classes on two sides of the hyperplane. The data that fall on the margin line is called support vectors.

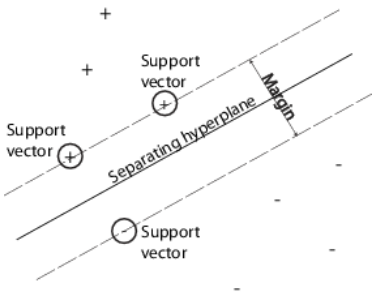


FIGURE 2.4: SVM hyperplane

As described, the support vector classifier finds linear boundaries in the input feature space. As with other linear methods, the procedure can be made more flexible by enlarging the feature space using basis expansions such as polynomials or splines. Generally linear boundaries in the enlarged space achieve better training-class separation, and translate to nonlinear boundaries in the original space. Once the basis functions$hm(x), m = 1, ..., M$are selected, the procedure is the same as before. We fit the SV classifier using input features$h(x_i) = (h1(x_i), h2(x_i), ..., h_M(x_i)), i = 1, ..., N$, and produce the (nonlinear) function $f(x) = h(x)^T\beta + \beta_0$. The classifier is $G(x) = sign(f(x))$as before.

The support vector machine classifier is an extension of this idea, where the dimension of the enlarged space is allowed to get very large, infinite in some cases. It might seem that the computations would become prohibitive. It would also seem that with sufficient basis functions, the data would be separable, and over fitting would occur [4].

2.6.1 Kernel

The selection of an appropriate kernel function is important, since the kernel function defines the transformed feature space in which the training set instances will be classified [27]. It is usual practice to estimate a range of potential settings and use cross-validation over the training set to find the best one [28].

$$K(x, x') = \langle h(x), h(x') \rangle \tag{2.1}$$

that computes inner products in the transformed space.
K should be a symmetric positive (semi-) definite function; Three popular choices for K in the SVM literature are:
dth-Degree polynomial:

$$K(x, x') = (1 + \langle x, x' \rangle)^d \tag{2.2}$$

Radial basis:

$$K(x, x') = exp(-\gamma||x - x'||^2), \tag{2.3}$$

[4]

2.7 Decision Tree

There are many simple, but widely used models that work by partitioning the input space into cuboid regions, whose edges are aligned with the axes, and then assigning a simple model (for example, a constant) to each region. These models can be viewed as a model combination method in which only one model is responsible for making predictions at any given point in input space. The process of selecting a specific model, given a new input x, can be described by a sequential decision making process corresponding to the traversal of a binary tree (one that splits into two branches at each node).
Figure 2.5 shows an example of classification tree on a two dimensional dataset.

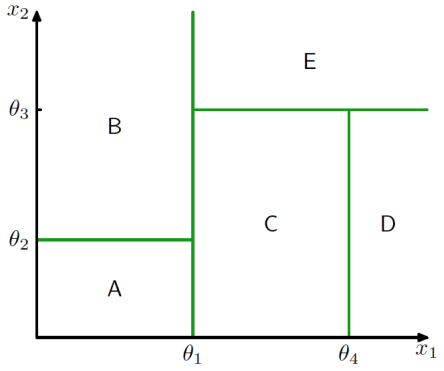


FIGURE 2.5: Illustration of a two-dimensional input space that has been partitioned into five regions using axis-aligned boundaries. θ_1 is a parameter of the model[3]

One limitation of decision trees is that the division of input space is based on hard splits in which only one model is responsible for making predictions for any given value of the input variables. The decision process can be softened by moving to a probabilistic framework for combining models.

2.8 K nearest neighbours

K nearest neighbour classifiers are memory-based and require no model to be fit. Given a query point x_0, one can find the k training points $x_(r), r = 1, ..., k$ closest in the distance

to x_0, and then classify using majority vote among the k neighbours. Ties are broken at random. To be simplified, assume that the features are real-valued, and use Euclidean distance in feature space:

$$d_{(i)} = ||x_{(i)} - x_0||. \tag{2.4}$$

Figure 2.6 and 2.7 shows the decision boundary of k-nearest-neighbor classifiers applied a simulation data when k is defined as one and fifteen.

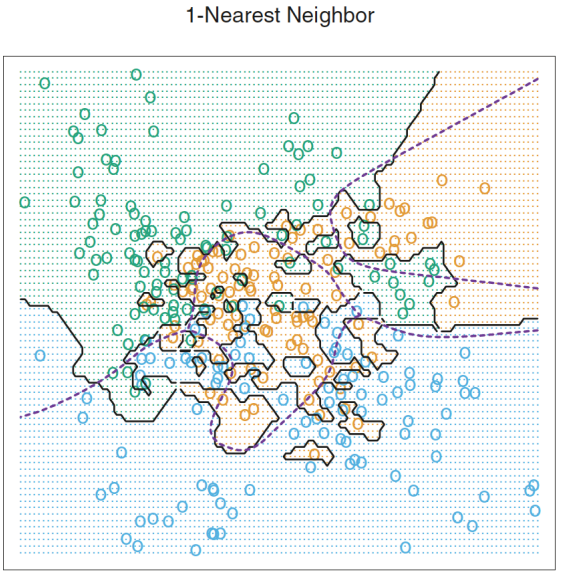


FIGURE 2.6: k-nearest-neighbor classifiers applied a simulation data. The broken purple curve in the background is the Bayes decision boundary. [4]

The decision boundary is fairly smooth compared to the lower panel, where a 1-nearest-neighbour classifier was used. There is a close relationship between nearest-neighbour moreover, prototype methods: in 1-nearest-neighbor classification, each training point is a prototype. Despite its simplicity, k-nearest-neighbors has been successful in a significant number of classification problems, including handwritten digits, satellite image scenes and EKG(electrocardiogram) patterns. It is often successful where each class has many possible prototypes, and the decision boundary is very irregular [4].

2.9 Ensemble Learning

The idea of ensemble learning is to build a prediction model by combining the strengths of a collection of simpler base models. Ensemble learning can be broken down into two tasks: developing a population of base learners from the training data, and then combining them to form the composite predictor [4].

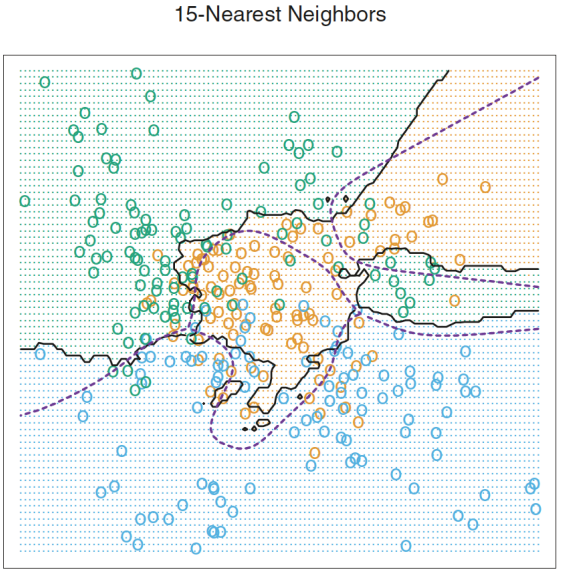


FIGURE 2.7: k-nearest-neighbor classifiers applied a simulation data. The broken purple curve in the background is the Bayes decision boundary. [4]

2.10 Evaluation on the classifiers

In this section, common practices for quantifying the performance of classifications will be provided.

2.10.1 K-fold cross validation

Cross validation is probably the simplest and most widely used method for estimating prediction error is cross-validation. This method directly estimates the expected extra-sample error $Err = E[L(Y, f(X))]$, the average generalization error when the method f(X) is applied to an independent test sample from the joint distribution of X and Y . As mentioned earlier, we might hope that cross-validation estimates the conditional error, with the training set T held fixed [4].

K-fold cross validation uses part of the available data to fit the model, and a different part to test it.

Let $k : 1, ..., N \rightarrow 1, ..., K$ be an indexing function that indicates the partition to which observation i is allocated by the randomization. Denote by $f^{-k}(x)$ the fitted function, computed with the kth part of the data removed. Then the cross-validation estimate of

the prediction error is

$$CV(f) = \frac{1}{N}\sum_{i=0}^{n} L(y_i, f^{-k(i)}(x_i)) \tag{2.5}$$

The case K = N is known as leave-one-out cross-validation. In this case (i) = i, and for the ith observation the fit is computed using all the data except the ith [4].

2.10.2 Sensitivity and Specificity

In medical classification problems, the terms sensitivity and specificity are used to characterize a rule. They are defined as follows:
Sensitivity: probability of predicting positive given true state is positive.
Specificity: probability of predicting negative given true state is negative.

Table 2.1 shows derivations of each terminology from a confusion matrix.

Terminology	Derivations
(number of) positive samples	P
(number of) negative samples	N
(number of) true positive	TP
(number of) true negative	TN
(number of) false positive	FP
(number of) false negative	FN

TABLE 2.1: Terminology and derivations from a confusion matrix

In binary classifications, a class label contains known binary values, where the predicted label contains predicted binary values. Positive samples P means the number of observations that are positive where Negative samples mean the number of observations that are predicted as negative. While comparing these two labels to compute classification accuracy, the observations that are positive and be predicted as positive as well will be recognize as true positive and on the opposite, the observations that are negative and be predicted as negative as well will be recognize as true negative. Following the same rules, the observations that are positive and be predicted as negative will be recognize as false negative while negative values get predicted as positive will be recognized as false positive.
Here listed generally used terms for classifier performance evaluation:

Sensitivity or true positive rate TPR:

$$TPR = TP/P = TP/(TP + FN) \tag{2.6}$$

Specificity (SPC) or true negative rate:

$$SPC = TN/N = TN/(TN + FP) \tag{2.7}$$

Precision or positive predictive value (PPV):

$$PPV = TP/(TP + FP) \tag{2.8}$$

Accuracy (ACC):

$$ACC = (TP + TN)/(TP + TN + FP + FN) \tag{2.9}$$

2.10.3 Area under the curve

The receiver operating characteristic curve (ROC) is a commonly used summary for assessing the tradeoff between sensitivity and specificity. It is a plot of the sensitivity versus specificity as we vary the parameters of a classification rule. The area under the curve is a commonly used quantitative summary [4].
Figure 2.8 shows a receiver operating characteristic curve and the area under the curve for a trained SVM model in this project.

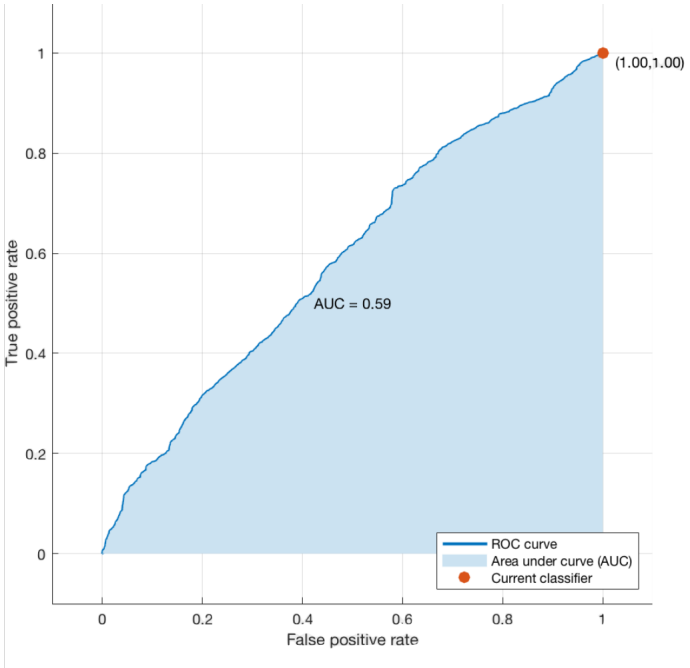


FIGURE 2.8: An example of AUC computed while training a classifier

Chapter 3

Literature review

3.1 The background

Supply chain planning is concerned with the coordination and integration of the major business activities undertaken by a company, from the procurement of raw materials to the distribution of the final products to the end users or consumers. In the changing markets nowadays, maintaining an efficient and flexible supply chain is crucial for companies, especially given the current volatilities in the business environment with constantly changing and increasing customer expectations. Various causes of uncertainty can be classified in these systems. Underestimating uncertainty and its impact can lead to planning decisions that neither safeguard an enterprise against the threats nor take advantage of the opportunities that higher levels of uncertainty provide [29].
One of the major purposes of supply chain collaboration is to improve the accuracy of forecasts. Forecasting and replenishment based on collaboration allow a company and its suppliers to coordinate decisions by exchanging complex decision-support models and strategies, hence facilitating an integration of forecasting and production schedules. Without these collaborations, companies are relegated to traditional forecasting and production scheduling [30]. As a result, the companys demand (e.g., the manufacturers demand) appears to fluctuate randomly even if the final customers demand has a predictable pattern. Forecasting the manufacturers demand under these conditions becomes a challenging task due to a well-known bullwhip effect – a result of information asymmetry [31].

The traditional ways of forecasting in the supply chain are Naive Forecast, Average, Moving Average, Trend, Multiple Linear Regression. Where more advanced tools could be: Neural Networks, Recurrent Neural Networks and Support Vector Machines. Techniques is of considerable value for firms, as the use of moving average, naive forecasting

or demand signal processing have been shown having an impact in the bullwhip effect [32].

It is expected that the advanced methods will outperform the more traditional ones, since

The advanced methods incorporate nonlinear models, and as such could serve as better approximations than those based on linear models;

It is expected to have a significant level of nonlinearity in demand behaviour as it exhibits complex behaviour [17].

3.2 Dealing with Data Issues

Imbalanced learning is a learning process for data representation, and information extraction with severely skewed data distribution, to develop effective decision boundaries supporting the decision-making process. Supervised learning, unsupervised learning, semi-supervised learning, or a combination of two or all of them can be involved in the learning process. The task of imbalanced learning could be implemented to regression, classification, or clustering tasks in machine learning as well [33]. Previously, some solutions to the class-imbalance problem were proposed, at both of the data and algorithmic levels. At the data level, these solutions include various forms of resampling techniques, such like random oversampling with replacement, random undersampling, directed oversampling (in which no new examples are created, but the choice of samples to replace is informed rather than random), directed undersampling (where, again, the selection of examples to eliminate is informed), oversampling with informed generation of new samples, and combinations of the above techniques. At the algorithmic level, solutions include adjusting the costs of the various classes to counter the class imbalance, adjusting the probabilistic estimate at the tree leaf, adjusting the decision threshold, and recognition-based (i.e., learning from one class) rather than discrimination-based (two class) learning [19]. Many tools and methods for dealing with data imbalance can be found in book [33], so they will not all be listed here. An outlier is defined as a data point which is very dissimilar from the rest of the data points based on some measure. A typical point like this often contains useful information on the erratic behaviour of the system described by the data. This problem typically arises in the context of very high dimensional data sets. Much of the recent work on finding outliers use methods which make implicit assumptions of the relatively low dimensionality of the data. These methods do not function quite as well when the dimensionality is high, and the data becomes sparse [34].

Reducing data dimensions for the better learning process, especially in sparsely filled high dimensional spaces has been studied for a long time. Recently, a research [35] pointed out that the significant imbalance in class cardinalities of asymmetric classification causes conventional classification techniques to yield poor accuracy (e.g., too complex learning rules which cause over fitting). Research [34], [23] and [36] provided many advanced methods to deal with outlier detection during prediction and especially with data in high dimensional space.

3.3 Issues while tuning the classifiers

There are two simple approaches to prediction: Least Squares and Nearest Neighbors. Linear models make huge assumptions about structure and yields stable but possibly inaccurate predictions. The method of k-nearest neighbours makes very mild structural assumptions: its predictions are often accurate but can be unstable [4].
In high-dimensional spaces, distance kernels are modified to emphasise some variable more than others [4].

Support vector machines (SVMs) are a new popular type of universal function approximators that are based on the structural risk minimisation principle from statistical learning theory as opposed to the empirical risk minimisation principle on which neural networks and linear regression, to name a few, are based [37]. Support vector machines project the data into a higher dimensional space and maximize the margins between classes or minimize the error margin for regression. Margins are soft, meaning that a solution can be found even if there are contradicting examples in the training set. The problem is formulated as a convex optimization with no local minima, thus providing a unique solution as opposed to back-propagation neural networks, which may have multiple local minima and, thus cannot guarantee that the global minimum error will be achieved. A complexity parameter permits the adjustment of the number of error versus the model complexity, and different kernels, such as the Radial Basis Function (RBF) the kernel can be used to permit nonlinear mapping into the higher dimensional space [17].

Support vector machines are configured to learn and predict the performance of suppliers in a supplier based company. They can forecast transaction outcomes and values of otherwise unknown attributes (e.g., timeliness, price, quality, purity, or freshness) for a prospective transaction. The Support vector machines can classify transactions or performance of suppliers as binary values, or under multi-value classifications, can as well operate on limited, possibly incomplete samples in first domains to learn about and predict other domains. Since the SVMs can use outcomes of past trans actions and

other data to predict an individual suppliers performance in future transactions, the predictions and other outputs generated by the SVMs can be used for many constancies [38].

3.4 Similar works

This section will briefly summarise the most related scientific works to this project. It provides an overview of relevant critical analysis, methods to use, restrict research areas to this topic and inspirations.

Many researchers have analysed applications of machine learning for demand forecasting in supply chain systems. As mentioned before in section 1.3, to prevent backorders, one of the ways is to improve prediction accuracy. These papers are based on researches about demand forecasting in the supply chain. Even though their research approaches are not precisely the same as backorder prediction, but the tools and methods they are using have a significant reference value to this project. In 2006, Real Carbonneau, Kevin Laframboise, Rustam Vahidov published a work called: Application of machine learning techniques for supply chain demand forecasting [17].

They compare advance prediction methods with other, more traditional ones, including naive forecasting, trend, moving average, and linear regression. Using two data sets for the experiments: one obtained from the simulated supply chain, and another one from actual Canadian Foundries orders. The findings show that even the recurrent neural networks and support vector machines show the best performance, their forecasting accuracy was not significantly better than the results from traditional models.
Advanced Prediction Methods they used: Recurrent neural networks and Support vector machines.
With the same researchers, another work called: Machine learning-Based demand forecasting in supply chains was conducted and published in 2007 [39]. The approach of this work is comparing machine learning based prediction tools with traditional predictions as well. This time, they have tried out including advanced forecasting like Neural network, recurrent neural network, SVMs and traditional forecasting like Trend, Moving Average and Exponential Smoothing, 22 tools in total.
As a group, based on ranking, the average performance of the ML techniques does not outperform the traditional approaches. However, using a support vector machine (SVM) that is trained on multiple demand series has produced the most accurate forecasts.
In 2011, Wang Guanghui conducted an analysis on Demand Forecasting of Supply Chain Based on Support Vector Regression Method [40]. This paper applies SVR to predict the demand of supply chain in real data and compared to the RBF(radial basis function)

neural network method. The result shows that SVR is superior to RBF in prediction performance. Moreover, SVR is the suitable and efficient method for demand forecasting of the supply chain.

Based on previous experience and references, this project will try out SVM tools and compare to a neural network analysis to find out the best forecasting model.

Chapter 4

Proposed solution

4.1 Introduction to the solution

As discussed in the target scenario, to avoid bullwhip effect, small brewers need to react fast and meantime avoid overreacting when the market changes. By having a more advanced forecasting too, business can identify the market more precisely where the predictions providing more evidence about what's going to happen. A relevant dataset was found to support the prediction process, which contains features from the early stage of the supply chain and has a binary target value.

Many machine learning tools and methods can perform predictions with good performance, in this project, many classification methods will be tried and evaluated, in order to get the best performance classifier and predictions. Figure 4.1 shows a general machine learning process, and it is also the work flow that this project is implicated. By understanding, discovering data issues and as well visualizing historical data, the scale and definition of the training set will become clearer.

A few standard classifiers were chosen to train the prediction models, and they were tested and evaluated. When the results are under-promising, the training set may be redesigned, some analysis may be conducting in another way, until the best performance shows up. Each classifier method was tested in many ways, and the final evaluation shows a comparisons of performance indicators, which will show the overview of each classifiers and as well the best one that can be trained under certain circumstances.

Figure 4.2 briefly shows a working diagram about the processes in this project.

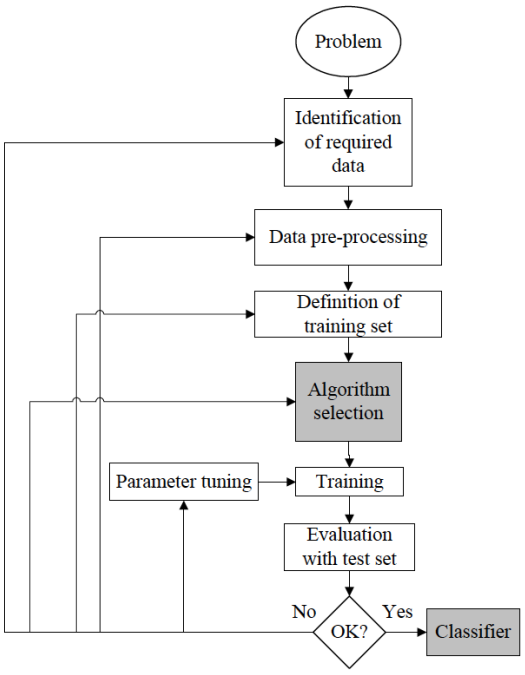


FIGURE 4.1: Machine learning work flow [5]

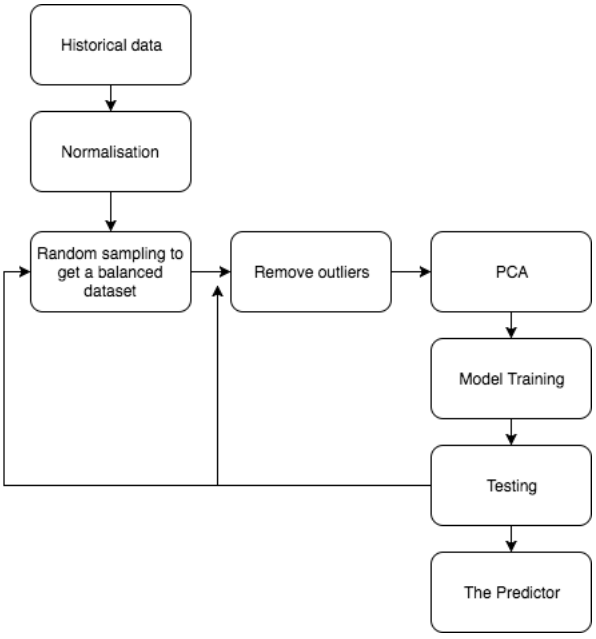


FIGURE 4.2: Project work flow

4.2 Data pre-processing

Real-world data is often incomplete, inconsistent, and/or lacking in certain behaviors or trends, and is likely to contain many errors. Data pre-processing is a proven method of resolving such issues. The goal of this stage is to transform raw data into an understandable format. In this project, it is required that all features in the training set are numeric. So this section will introduce steps for pre-processing the dataset and also give a understanding about the data contend and scale.

4.2.1 Introduction to the dataset

The dataset used in this project comes from website Kaggle, provided by an anonymous company. Details of the data can be found in appendix A1. The dataset is historical data, which contains information of their products for the 8 weeks prior to the week that is going to be predicted. The data was taken as weekly snapshots at the start of each week. In total, it consists 23 attributes with 1,915,211 entries and, 6 of which are binary, 1 notation and the other 16 are numeric. Table 4.1 shows brief explanation and data types of the 23 attributes: To note, SKU represents random products. The number of actual products is far smaller than millions, but the actual unique SKU for each product is not included as they can change. So a sequential identifier was created where each record identifies a unique product/week combination.

Among the attributes, some represents the status of transiting, like: lead time of the product (lead-time, as weeks) and amount of product in transit from source (in transit qty, units).

Some represents sales and prediction: attribute 6-10 some represents part risk and source issue: attribute 11-19.

For source performance, it represents product supplier's historical performance. Some suppliers have not been scored and have a dummy value of -99 loaded.

The part risk flag means the shortage of a part: in this project, it could be the shortage of ingredients needed for the production. And others represent inventory levels and stocking: attribute 20-23. Generally, part risk and source issue will create overdue of products, it might lead to a stock shortage. And a stock shortage may lead to the product backorder. This dataset was created on the understanding of the business and also the data availability. All the binary attributes except the target value were created as an risk indicator, based on other calculations that was not stated in this introduction. Reasons for product went on backorders could be multiple, especially when the target value is among a high dimensions space, that is where machine learning can bring values in.

	Attribute	Description	Data type
1	sku	Random ID for the product	Norminal/Discrete
2	lead time	Transit time for product (if available)	Ratio/Continuous
3	in transit qty	Amount of product in transit from source	Ratio/ Discrete
4	forecast 3 month	Forecast sales for the next 3 months	Ratio/ Discrete
5	forecast 6 month	Forecast sales for the next 6 months	Ratio/ Discrete
6	forecast 9 month	Forecast sales for the next 9 months	Ratio/ Discrete
7	sales 1 month	Sales quantity for the prior 1 month	Ratio/ Discrete
8	sales 3 month	Sales quantity for the prior 3 month	Ratio/ Discrete
9	sales 6 month	Sales quantity for the prior 6 month	Ratio/ Discrete
10	sales 9 month	Sales quantity for the prior 9 month	Ratio/ Discrete
11	perf 6 month avg	Source performance for prior 6 month	Ratio/ Discrete
12	perf 12 month avg	Source performance for prior 12 month	Ratio/ Discrete
13	pieces past due	Parts overdue from source	Ratio/ Discrete
14	deck risk	Part risk flag	Binary
15	oe constraint	Part risk flag	Binary
16	ppap risk	Part risk flag	Binary
17	stop auto buy	Part risk flag	Binary
18	rev stop	Part risk flag	Binary
19	potential issue	Source issue for part identified	Binary
20	national inv	Current inventory level for the part	Ratio/ Discrete
21	min bank	Minimum recommend amount to stock	Ratio/ Discrete
22	local bo qty	Amount of stock orders overdue	Ratio/Discrete
23	went on backorder	Product actually went on backorder	Binary

TABLE 4.1: Attributes from the historical data, a short explanation to them and their data types

4.2.2 Binarization

In statistics, the response to a yes-no question ("yes" or "no") is considered to be binary data. But in order to make the whole dataset processable in Matlab, it is necessary to have all data as numeric data type, so that all features can be involved in one matrix, which is the foundation of the data analyzing. In the dataset, there are binary attributes containing yes/no value, in order to analyse them in a matrix, it is necessary to transfer those values into numeric values. Codes are provided in appendix A2.

4.2.3 Summary statistics

After the dataset become fully numeric, a Summary Statistics can be performed, ought to see more features about the set. Due to the differences between data types of attributes, summery statistics does not always make sense when applying. Here only sensible explanation is conducted. This summary statistics table 4.2 shows the results before outliers were removed. Table 4.3 shows the total true class in each binary attribute.

Attribute	Value	Mean	Median	Stand deviation	Maximum
lead time	weeks	498,2	15	29615,2	12334404
in transit qty	unit	7,9	8	6,8	52
forecast 3 month	unit	44,1	0	1342,7	489408
forecast 6 month	unit	178,1	0	5026,6	1427612
forecast 9 month	unit	345	0	9795,2	2461360
sales 1 month	unit	506,4	0	14378,9	3777304
sales 3 month	unit	55,9	0	1928,2	741774
sales 6 month	unit	175	1	5192,4	1105478
sales 9 month	unit	341,7	2	9613,2	2146625
perf 6 month avg	N/A	525,3	4	14838,6	3205172
perf 12 month avg	N/A	52,8	0	1255	313319
pieces past due	unit	0	0	0	1
national inv	unit	1	1	0,2	1
min bank	unit	0	0	0	1
local bo qty	unit	0	0	0,1	1

TABLE 4.2: Attributes from the historical data, following with summary statistics

Attribute	Number of true class
potential issue	907
deck risk	387483
oe constraint	245
ppap risk	203834
stop auto buy	1626774
rev stop	731
went on backorder	11293

TABLE 4.3: Binary attributes from the historical data, following with numbers of true class in each attribute

4.3 Data Visualization

The dataset is a combination of numeric and binary data. It contains randomly distributed missing values, negative values, and outliers as well. Figure 4.3 shows a plot matrix of all features from the historical data, which creates a matrix of sub-axes containing scatter plots of the columns of X against the columns of Y. As seen from the

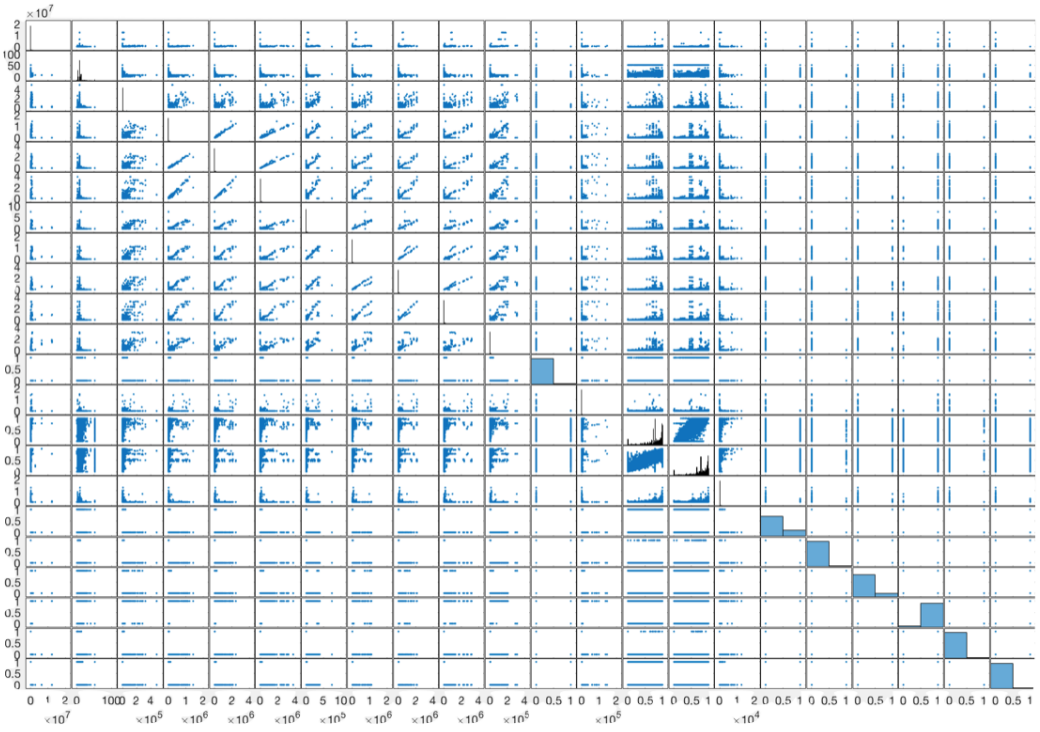


FIGURE 4.3: A matrix of scatter plots of features combinations. Each small plot has figures beside which indicates the data range from the plotting feature

plot matrix, there are not obvious relations between different features. But the reason why this is happening might be the huge difference between features. Figure 4.4and 4.5 shows an imbalance on feature scales, and also a great deal of outliers from each feature, which leads to further normalization and a clearer outlier detection.

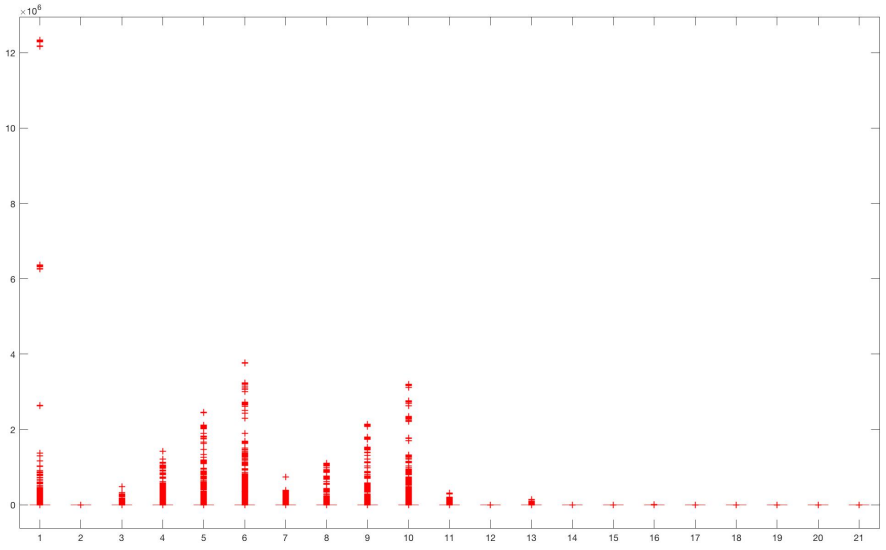


FIGURE 4.4: Boxplot of each column of the data. X represents each feature, Y represents their values.

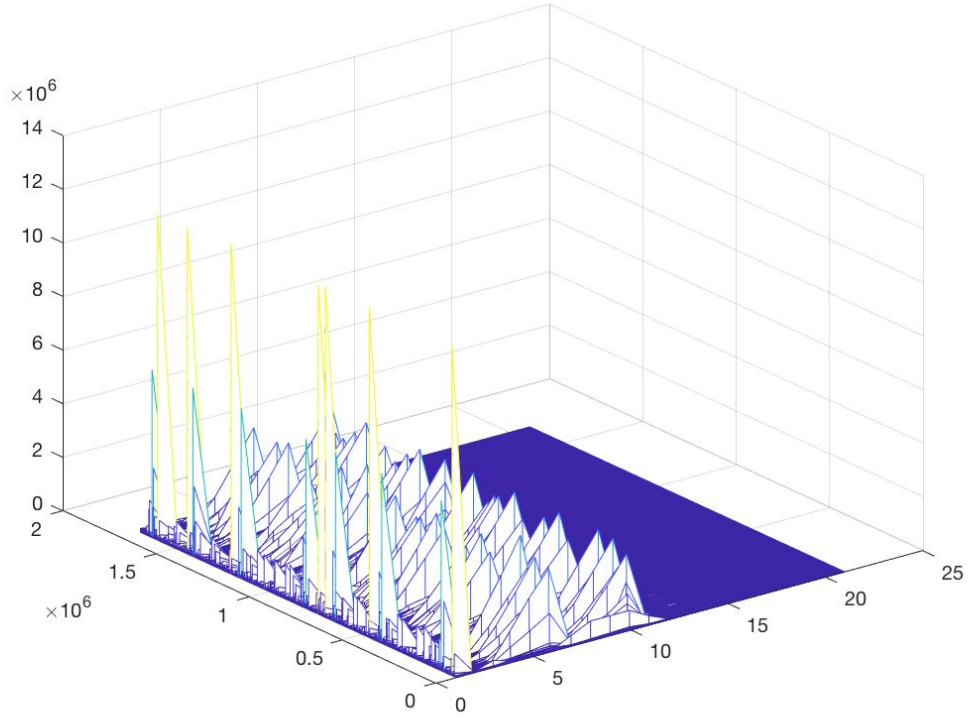


FIGURE 4.5: Mesh plot of all features from the historical data, where X represents the numbers of features, Y represents number of entries, and Z represents the value of each (X,Y). Mesh(X,Y,Z) draws a wire frame mesh with color determined by Z, so color is proportional to surface height. If X and Y are vectors, length(X) = n and length(Y) = m, where [m,n] = size(Z) [6].

4.4 Data issues

In this section all related data processing processes are addressed. For details of the coding please go to appendix A2

4.4.1 Data imbalance

Knowing from the data, 0.7% of the products actually went on backorder, which creates a big imbalance on the classes. It is a common problem for a real dataset. Figure 4.6 shows the proportion of the two predicting classes.

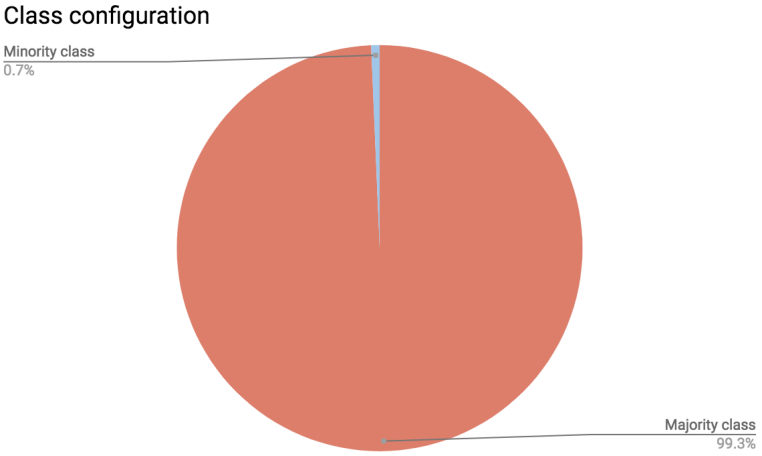


FIGURE 4.6: A pie chart of proportion of the two predicting classes

Preferably, it is required that a classifier provides a balanced degree of predictive accuracy for both the minority and majority classes on the dataset. However, in many standard learning algorithms, it is found that classifiers tend to provide a severely imbalanced degree of accuracy, with the majority class having close to 100% accuracy and the minority class having accuracy of 0 - 10%;

This low accuracy on minority class often suggests that the conventional evaluation practice of using singular assessment criteria, such as the overall accuracy or error rate, does not provide adequate information in the case of imbalanced learning.

4.4.2 Missing values

To detect missing values, including NaN values, empty entries ect, a single line of code is needed and instant results will be shown. In Matlab, the function is called "ismissing". Detect"NaN" values:
Columns 1 through 23
0 1 0 1 1 1 1 1 1 1 1 1 0 1 1 1 1 0 0 0 0 0 0
Detect missing values by using 'ismissing' function:
Columns 1 through 23
0 1 100894 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

As showed from results, column 3, lead-time has 100894 missing values, those missing values were left empty since there was no realistic data to fill in. However, most of the missing values is isolated to one particular attribute, generally there are two options to deal with this. Option one, to fill empty entries and complete the data matrix with the mean of all observed instances of the lead-time attribute. Option two, discard this attribute if it is not essential. Since the data has not been analyzed for significance yet, it is unsure about whether this feature can influence the future analysis or not, so here option one is chosen to be the solution. This is the second step making the data set fully numeric. For some reasons, when the company record data, some data spots were left blank, which are not importable in Matlab, that creates NaN(not a number) values. In this dataset, attribute 'lead time' has 100894 NaN values, but since this number is not big compare to the volume of the whole dataset, they are kept in the dataset for further cleaning. For those of values that are either NaN or negitive, it is needed to replace them with mean or median of the feature. In this project, it was replaced with mean. After this step, the dataset is fully numeric and can be processed with data analyzing by Matlab.

4.4.3 Data redundancy

Scaling before applying SVM is very important. Part 2 of Sarle's Neural Networks FAQ Sarle (1997) explains the importance of this and most of considerations also apply to SVM. The main advantage of scaling is to avoid attributes in greater numeric ranges dominating those in smaller numeric ranges. Another advantage is to avoid numerical difficulties during the calculation. Because kernel values usually depend on the inner products of feature vectors, e.g. the linear kernel and the polynomial kernel, large attribute values might cause numerical problems. We recommend linearly scaling each attribute to the range [-1; +1] or [0; 1] [41]. Many machine-learning methods such as principal component analysis and K-means are sensitive to outliers or features that reside on a different scale, and it may, therefore, be a good idea to standardise (change) features. As a result from experience, normalisation can not only help PCA perform better, but also save time during outlier detection and model training. For the manner of convenience and time saving, it is needed to perform normalisation, or feature transformation just to call it in another way. There are a few ways to scale the data, one way shows as equation 4.1, scale data into [0,1]:

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{4.1}$$

or scale data into [-1,1], shows as equation 4.2

$$X' = \frac{X - \mu}{\sigma} \tag{4.2}$$

4.4.4 Dimentionality Reduction and Outlier detection

Due to the data's high dimentionality, a reduction on the dimentionality could be useful for the classification training that happens later on. Linear subspace analysis for feature extraction and dimensionality reduction has been studied in depth for a long time and many methods have been proposed in the literature, including principle component analysis (PCA) [21], linear discriminant analysis (LDA) [22], null space LDA (NLDA) [42] etc. Dimension reduction methods are invaluable for the analysis of genomics data and other high-dimensional data. In particular, principal component analysis (PCA) and related methods reduce a large number of variables (eg, genes or genetic markers) to a small number of latent components that explain much of the variation in those variables. By keeping the major significance of the data set and have much smaller data domin saves much time during classification model training and the prediction on test set [23]. Principal component analysis can be also used as an exploratory tool, and the principal components can also be used as predictors in a statistical model for an outcome. This latter use of PCA can solve issues of overfitting, identifiability, and collinearity that arise when a predictive model is fit using the original data [36].

Outlier detection

But sometimes, though applied very successfully for pattern classification, these dimentionality reduction methods usually miss out some discriminant information while extracting relevant features for the classification task. That's due to the outliers from the dataset. Outlier detection is an important data mining task and has been widely studied in recent years [43], but it is not always easy to detect them. Table 4.4 shows simple ways to detect outliers. Among these simple methods, Mean method is faster but less robust than 'median'; Quartiles is useful when the data in A is not normally distributed; Grubbs method assumes that the data in A is normally distributed; And Gesd, is an iterative method that is similar to 'grubbs', but can perform better when there are multiple outliers masking each other.

[6] The reason why that outliers are hard to detect might be:

(a) outliers are rare and hard to collect

(b) selecting subspaces for outlier detection is a complex problem [34] and some times it will cause loss of discriminant features.

This problem has become a major concern when comes to machine learning in high dimension datasets. However, there are researchers trying to find ways to solve this problem, detect outliers in high dimensions dataset [23]. Research paper can be found in reference:[23] [34] [44].

Due to time issue, these advanced tools and techniques cannot be applied on the outlier detection in this project, and only Mean method was tried out for defining the training

Method	Description
median	Returns true for elements more than three scaled MAD from the median.
mean	Returns true for elements more than three standard deviations from the mean.
quartiles	Returns true for elements more than 1.5 interquartile ranges above the upper quartile or below the lower quartile.
grubbs	Applies Grubbs's test for outliers, which iteratively removes one outlier per iteration based on hypothesis testing.
gesd	Applies the generalized extreme Studentized deviate test for outliers.

TABLE 4.4: Simple ways to detect outliers [6]

set.

Note that the scale of the attributes affects the outcome of principal components analysis, and it is common practice to standardize all attributes to zero mean and unit variance first [18].

Principle component analysis

For the dimentionality reduction manner, a Principle component analysis was conducted in this project. PCA can significantly reduce the features from 22 down to 5. By reducing the dimentionality, irelevant features are excluded, features that have 95% significance are left after the reduction. PCA also helps in this way where the real/verbal meanings behind each feature are overlooked, only linear relations are left to describe how important is the feature, which will do a great help for feature selection. To note, in order to get same quantity features for training set and test set, it is better to convert two sets together during the Principle components analysis, since classifier training and prediction needs the features form two sets to be same ordered and aligned.

4.5 Data Modeling

4.5.1 Design of experiments

When data pre-processing and data cleaning give a better explanation of how the data looks like, the issues from the data for machine learning are also revealed. Outliers detection in high dimensional dataset and dataset extremely imbalanced are the core issues in this project, both of which can confuse classifiers during data training and give inaccurate results.

The variability of data issues were highly considered, so experiments were designed ought to reduce most of the negative influence that these issues can give on the classifier accuracy, so that the best classifier for backorder prediction can be found out.

For the issue of data imbalance, if use full historical data to train classifier, it may give a confusing result where the accuracy is very high but the true class precision is low. A sampled dataset with balanced class values may give a more promising result. For the issue of outliers: Since outliers can be a fraction to the accuracy of the classifiers, for comparison, the data can be trained in two manners: with outliers and without outliers. The result can show if removing outliers can lead to better accuracy.

Based on the literature review, support vector machine always give more accurate results during prediction, since SVMs allow for some degree of mis-classification of data and then repeatedly optimizes a loss function to find the required best fit model. Review also shows that SVM Gaussian as a basic classification method is widely use, so it will be trained together with other classifiers to compare results.

4.5.2 Definition of training set

A classification task usually involves separating data into training and testing sets. Each instance in the training set contains one target value" (i.e. the class labels) and several attributes" (i.e. the features or observed variables). A training set is a set of data used to discover potentially predictive relationships. A test set is a set of data used to assess the strength and utility of a predictive relationship.

To reduces data redundancy and data dimensionality, enable machine learning algorithms to operate faster and more effectively, A data normalisation and Principle component analysis were conducted. These methods can remain the data quality and main features while reducing data size. After the analysis, the newly generated features may lead to the creation of more concise and accurate classifiers. In addition, the discovery of meaningful features contributes to better comprehensibility of the produced classifier, and a better understanding of the learned concept.

Due to the imbalanced issue, use the full historical data as a training set may not give the best classification results. However, there are many tools and methods to deal with data imbalance, some of which are simple and some others are more intelligent. The sampling methods described in the Theoretical Background are designed to reduce between-class imbalance. Although research indicates that reducing between-class imbalance will also tend to reduce within-class imbalances [45], it is worth considering whether sampling methods can be used in a more direct manner to reduce within-class imbalancesand whether this is beneficial. Based on this, by extracting all data entries with minority class and random sampling data entries from majority class, a balanced dataset is created to reduce that problems that might occur during training on imbalanced data. For comparison to see if the sampling method is beneficial, two training sets were created: Training set one: a training set which includes all entries from cleaned historical data

Training set two: a training set which includes all minority class and same amount of majority class

4.5.3 Algorithm selection

The choice of which specific learning algorithm should be used is a critical step. Once preliminary testing is judged to be satisfactory, the classifier (mapping from unlabeled instances to classes) is available for routine use [5]. As mentioned in literature review, SVM and Neural networks are the best performers on predictions, but in this project it is restrict to try out different classifiers to test for the best model. Neural networks would not be considered for training. Based on the balanced training set, multiple classifiers were trained on cross validation with five folds.

Table 4.5 shows the results from the classifiers screening, where the top three classifiers (selected based on overall accuracy and area under the curve) are KNN, Ensemble KNN and SVM Gaussian.

Ranking	Classifier	Accuracy	AUC
1	KNN	73.1%	0.73
2	Ensemble KNN	72.9%	0.76
3	SVM Gaussian	63.2%	0.67
4	Decision Tree	59.9%	0.65
5	Logistic regression	55.7%	0.60
6	Linear discriminant analysis	55.7%	0.60
7	Linear SVM	49.2%	0.49

TABLE 4.5: Results of the classifiers screening, the top three classiders are: KNN, Ensemble KNN and SVM Gaussian

After the algorithm selection, the selected classifiers were trained on different kernel scales, number of neighbours and number of learners to get the best performance for that specific method, so in the chapter of results all the forecasting results are from the best performance that one model can get.

Support vector machines

As the number of features is small, one often maps data to higher dimensional spaces (i.e., using nonlinear kernels) [41].

The Gaussian kernel nonlinearly maps samples into a higher dimensional space so it, unlike the linear kernel, can handle the case when the relation between class labels and attributes is nonlinear. Furthermore, the linear kernel is a special case of RBF since the linear kernel with a penalty parameter C has the same performance as the RBF kernel with some parameters [46]. Figure 4.7 shows the prediction performance against the kernel scale. The rule discovered is that, the smaller the kernel scale is, the better the accuracy is. When the kernel scale is set as the lowest, which is 0.001, the model gets the best performance.

In the original feature space, a hyperplane cannot separate the classes, and the support vector classifier does poorly. The polynomial support vector machine makes a substantial improvement in test error rate, but is adversely affected by noise features (outliers). It is also very sensitive to the choice of kernel. At this stage, for polynomial kernel, it was chosen to use degree of 2 and 3. However, other kernels like linear kernel and polynomial kernel do not have better performance than the Gaussian kernel while outliers were not removed [4].

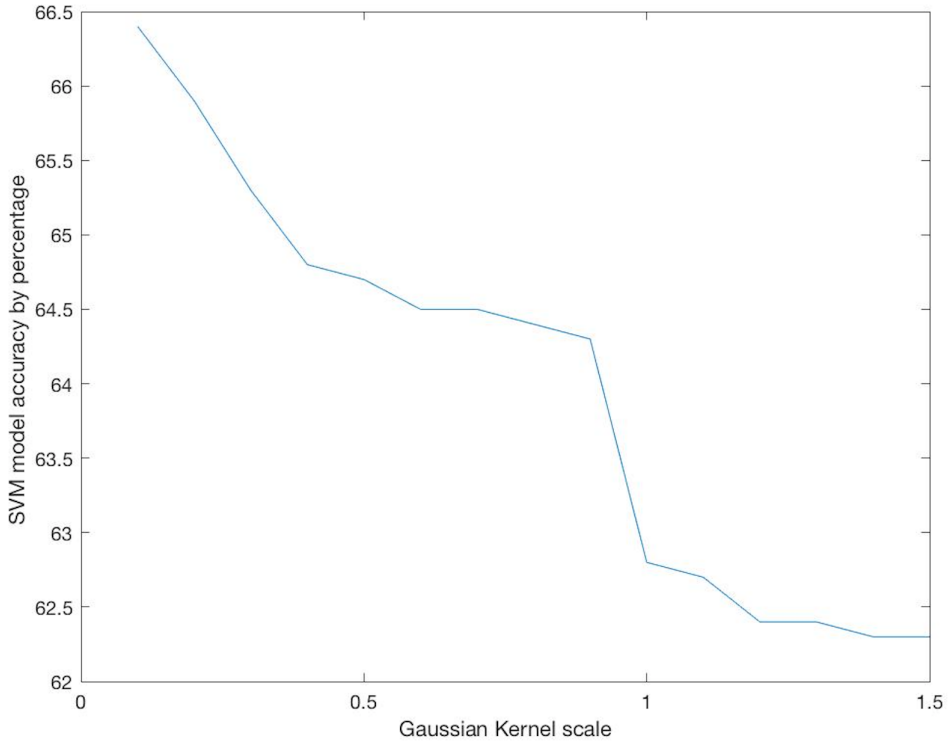


FIGURE 4.7: SVM Gaussian model accuracy plot.The accuracy is decreasing as the number of neighbour increases. Kernel scale: the smaller the better.

K nearest neighbours K nearest neighbours algorithm is one of the simplest of all machine learning algorithms. The most intuitive nearest neighbour type classifier is the classifier with k equals to one, which assigns a point x to the class of its closest neighbour in the feature space. KNN works better when the amount of features are small and are apart from each other. Figure 4.8 shows that when k=1 the classifier gets the best accuracy.

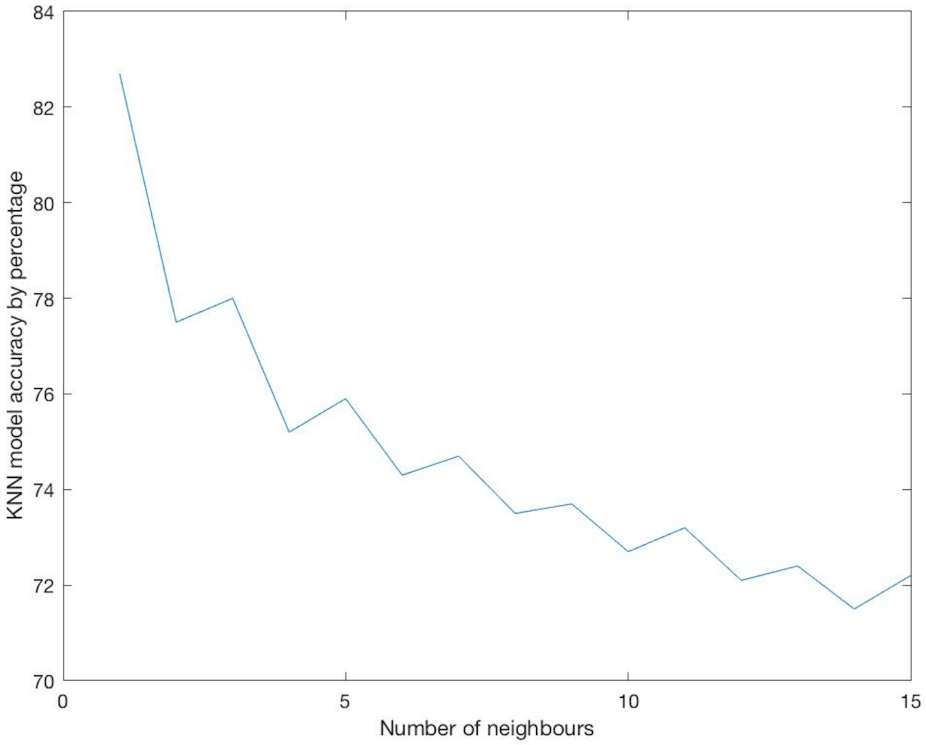


FIGURE 4.8: K nearest neighbour model accuracy plot. The accuracy is decreasing as the number of neighbour increases.

Ensemble subspace KNN

An ensemble of classifiers succeeds in improving the accuracy of the whole when the component classifiers are both diverse and accurate [47]. Figure 4.9 shows an improved model accuracy inside of an ensemble subspace KNN method. By having as least number of learners of possible, while k=8 the ensemble KNN gives the best result.

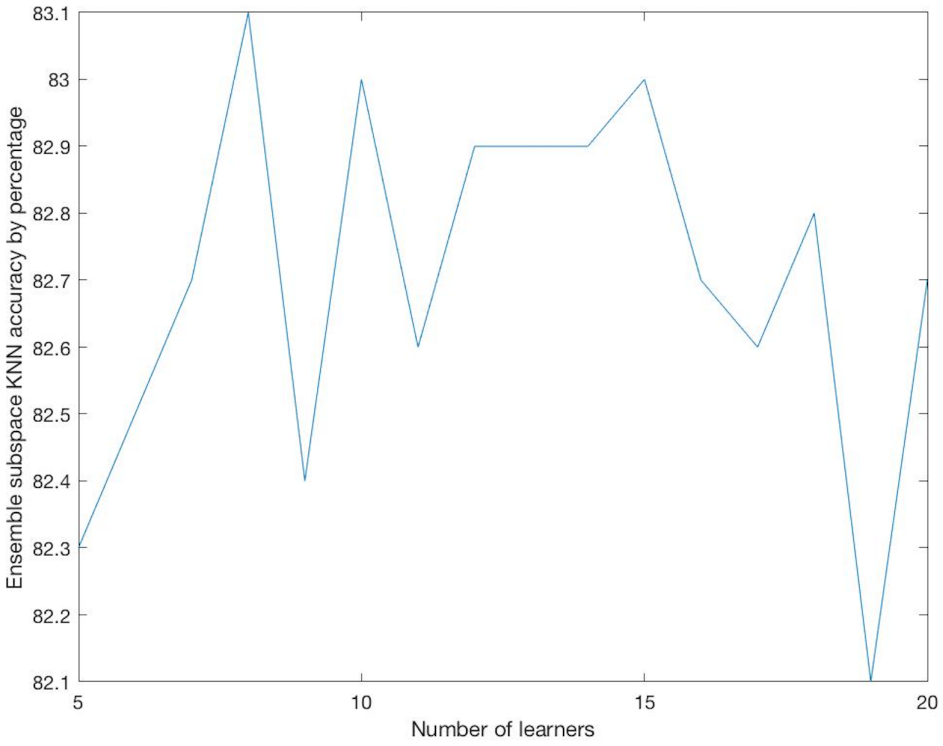


FIGURE 4.9: Ensemble subspace KNN model accuracy plot. When the number of learners ranging from 5 to 20, KNN gets the best accuracy when the number of learners is eight, which is 83.1%

Chapter 5

Results and Evaluation

5.1 Introduction to testing set

The testing set is normalized and analyzed by PCA together with different training sets for classification predictions. Table 5.1 shows the configuration of different testing sets that used during the training.

Test Set	Total entries	Number of features
Before removing outliers	227351	21
PCA before removing outliers	227351	4
PCA after removing outliers	227351	2

TABLE 5.1: Testing set for Training set one

5.2 Results

Result from the design of experiments many combinations were tested and evaluated. In this section some valuable results are presented for future evaluation and discussion.

5.2.1 Training set one: Cleaned historical data

This training set is extremely imbalanced, the results getting from the classifiers trained under this training set is unstable. Here the results are mainly for comparison with the balanced dataset.

Table 5.2 shows an overview of training set one.

Training set one	Total entries	Number of features
Before removing outliers	1687860	21
After removing outliers	1657976	21
After PCA	1657976	2

TABLE 5.2: Training set one

Table 5.3 and 5.4 shows results about confusion matrix and performance of this training set with outliers.

	TP	FP	TN	FN
KNN	2506	7	224740	98
SVM Gaussian	2452	5040	219707	152
Tree	2182	71	224676	422
Ensemble	2454	6	224741	150

TABLE 5.3: Confusion matrix for training set one with outliers

	TPR	PPV	SPC	ACC
KNN	0,96	1,00	1,00	1,00
SVM Gaussian	0,94	0,33	0,98	0,98
Tree	0,97	1,00	1,00	0,84
Ensemble	0,94	1,00	1,00	1,00

TABLE 5.4: Sensitivity and Specificity for training set one with outliers

Table 5.5 and 5.6 show results on the training set without outliers. Unfortunately, the results from KNN and ensemble KNN were failed to get for unknown reasons. But one of the reasons may cause the fail may due to the high computations. K-nearest-neighbour method require the entire training data set to be stored, leading to expensive computation if the data set is large. With full historical data training and testing, the model requires huge memory and time. It is not known the final accuracy of the models, but a model that costs expensive computations is not the best one to choose to use in the future.

From the results it is obvious to see that all classifiers have high accuracy rate, which is due to the extreme class imbalance.

	TP	FP	TN	FN
SVM Gaussian	0	0	224747	2604
Tree	2529	218521	6226	75

TABLE 5.5: Confusion matrix value for training set one without outliers

	TPR	PPV	SPC	ACC
Tree	0,97	0,01	0,03	0,04
SVM Gaussian	0,00	NA	1,00	0,99

TABLE 5.6: Sensitivity and Specificity for training set one without outliers

The support vector machines can predict only the majority class when they are trained by an imbalanced dataset. In this scenario, if one did not find out the imbalance issue and try to adjust kernel functions and kernel scales, it will be very confusing that the machine behaves abnormal.

Even when the outliers were removed, models still cannot get a satisfying result, this means dealing data imbalance is a primary step to do.

5.2.2 Training set two: Balanced data

A balanced dataset was created to try to find better results and more robust classifiers. All the entries with positive class were extracted out and the same amount of negative classes were randomly sampled from the rest of the historical data.

The results from training set two is also divided into two parts: a balanced dataset which includes outliers and another balanced dataset where outliers were deleted based on the mean.

Table 5.7 shows an overview of training set two.

Training set two	Total entries	Number of features
Before removing outliers	22586	21
After removing outliers	14479	21
After PCA	14479	2

TABLE 5.7: Training set two

After the dimensionality reduction, two features are left to be trained. Figure 5.1 shows a scatter plot of the two features.

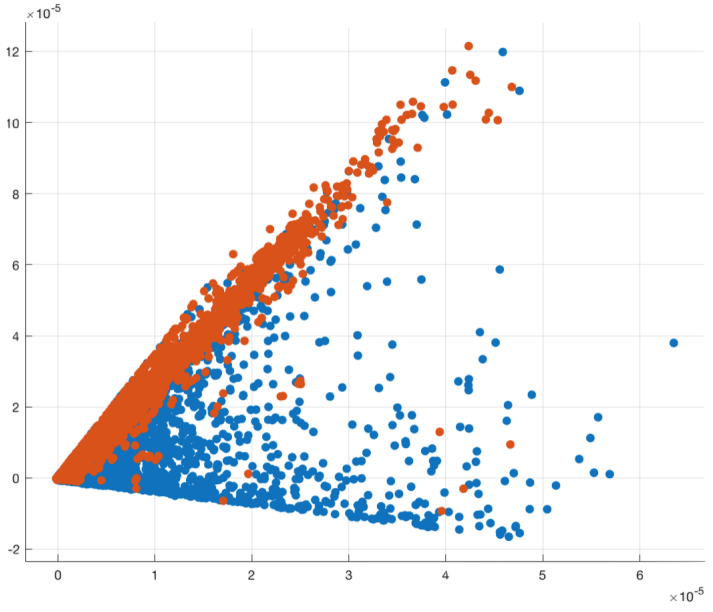


FIGURE 5.1: Scatter plot of two features of training set two after dimensionality reduction. Red indicates positive values and blue indicates negative values

When training with outliers, table 5.8 and 5.9 shows values from confusion matrix and performance based on sensitivity and sensitivity before dimensionality reduction.

Model	TP	FP	TN	FN
KNN	1483	39548	185199	1121
SVM Gaussian	1258	51710	173037	1346
Tree	2195	39968	184779	409
Ensemble	1804	44351	180396	800

TABLE 5.8: Confusion matrix for training set two with outliers, before PCA

Model	TPR	PPV	SPC	ACC	AUC
KNN	0,57	0,04	0,82	0,82	0.7
SVM Gaussian	0,48	0,02	0,77	0,77	0.63
Tree	0,84	0,05	0,82	0,82	0.83
Ensemble KNN	0,69	0,04	0,80	0,80	0.68

TABLE 5.9: Sensitivity and Specificity for training set two with outliers, before PCA

When training with outliers, table 5.10 and 5.11 shows values from confusion matrix and performance based on sensitivity and sensitivity after dimensionality reduction.

Model	TP	FP	TN	FN
KNN	435	22613	202134	2169
SVM Gaussian	35	1439	223308	2569
Tree	1143	80182	144565	1461
Ensemble	319	18708	206039	2285

TABLE 5.10: Confusion matrix for training set two with outliers, after PCA

Model	TPR	PPV	SPC	ACC	AUC
KNN	0,17	0,02	0,90	0,89	0.53
SVM Gaussian	0,01	0,02	0,99	0,98	0.5
Tree	0,44	0,01	0,64	0,64	0.54
Ensemble KNN	0,12	0,02	0,92	0,91	0.53

TABLE 5.11: Sensitivity and Specificity for training set two with outliers, after PCA

When training without outliers, table 5.12 and 5.13 shows values from confusion matrix and performance based on sensitivity and sensitivity before dimensionality reduction.

	TP	FP	TN	FN
KNN	2016	77763	146984	588
SVM Gaussian	2291	98414	126333	313
Tree	2236	71648	153099	368
Ensemble	1927	71413	153334	677

TABLE 5.12: Confusion matrix for training set two without outlier, before PCA

	TPR	PPV	SPC	ACC	AUC
KNN	0,77	0,03	0,65	0,66	0.71
SVM Gaussian	0,88	0,02	0,56	0,57	0.72
Tree	0,86	0,03	0,68	0,68	0.77
Ensemble KNN	0,74	0,03	0,68	0,68	0.75

TABLE 5.13: Sensitivity and Specificity for training set two without outlier, before PCA

When training without outliers, table 5.12 and 5.13 shows values from confusion matrix and performance based on sensitivity and sensitivity after dimensionality reduction.

	TP	FP	TN	FN
KNN	0	0	224747	2604
SVM Cubic	1940	30102	194645	664
Tree	2516	112594	112153	88
Ensemble KNN	0	0	224747	2604

TABLE 5.14: Confusion matrix for training set two without outlier, after PCA

	TPR	PPV	SPC	ACC	AUC
KNN	0,00	NA	1,00	0,99	0.5
SVM Cubic	0,75	0,06	0,87	0,86	0.81
Tree	0,97	0,02	0,50	0,50	0.73
Ensemble KNN	0,00	NA	1,00	0,99	0.5

TABLE 5.15: Sensitivity and Specificity for training set two without outlier, after PCA

From the results it is clear that classifiers behave more stable on a balanced training set. Among competing hypotheses, the one with the fewest assumptions should be selected. In this case, when the results from modelling are similar to each other, the simplest model would be chosen to be the final model. Overall, the SVM model with cubic kernel trained with reduced training set provides the best result.

5.3 Evaluation

To examine whether the results of the proposed method are better than the traditional forecasting, a comparison is given in this section.

For comparison, figure 5.2 and table 5.16 shows the result from a traditional way of predicting backorders. Details of traditional forecasting methods and equations can be found in appendix A2.

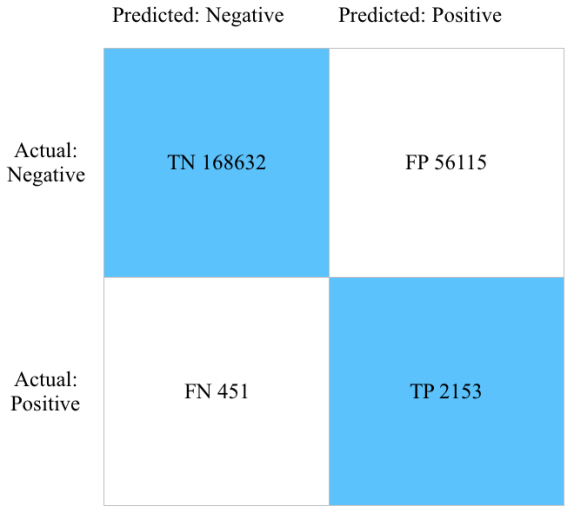


FIGURE 5.2: Confusion matrix from the traditional way to predict backorders

	TPR	PPV	SPC	ACC	AUC
Traditional method	0,83	0,04	0,75	0,75	0.79

TABLE 5.16: Prediction performance from the traditional way to predict backorders

By comparing the two results, machine learning predictions are not always the best forecasting methods compare to traditional forecasting. When training on a balanced dataset, all four methods have better overall accuracy. Throughout the trained classifiers, support vector machine is the classifier that gives the most stable and accurate predictions. Though decision tree also has outstanding results in some cases, but it sometimes behave in an extreme way that predicts only one class when feature were adjusted.

As seeing the results from the confusion matrix and sensitivity & specificity, support vector machine and K nearest neighbours method has performed well, nevertheless, support vector machine is a model that can tolerate miss-class but outliers, played better when outliers were removed. Possible reasons could be that the potential outliers

that did not detect misled the classifier to wrong results. K-Nearest Neighbors can give better results when there are only two features since it often performs very well with low dimensionalities.

While training with a balanced dataset with outliers, the advanced forecasting methods cannot compete the traditional method, which demonstrates that teh training set for machine learning models training needs to be carefully selected, cleaned and designed. Overall, the factors that influence classifiers' performance can be multiple. Feature selection, outlier detection, sampling method, kernel scale/function and maybe even more.

Comparing the two training sets, the sampled data did not help KNN get better results as it supposed to be and SVMs have had good results on both training sets. However, researchers have reported that sampling can be inefficient sometimes, for future analysis, other more advanced methods could be tried out to discover new results and rules. Some other researchers have discovered that in some cases a certain amount of outliers can help classifiers become more robust and tolerant. In this project, the relationship between classifier accuracy and data quality was not discovered.

Results from some issues from the historical data, the classifiers are not always performing well. Comparing to the results, when the outliers are removed in a balanced training set, the classifiers perform better.

Chapter 6

Conclusion and further discussion

The objective of this project is to find out whether advanced machine learning techniques give better effectiveness for backorders forecasting in the early stage of the supply chain. The results are relevant for circumstances when producers in the supply chain cannot collaborate with other stakeholders, where they have to predict backorders using their existing information. In such cases, the ability to increase forecasting accuracy will result in lower costs and higher customer satisfaction because of more on-time deliveries.

Various analyses were conducted to achieve such a goal. Data pre-processing including binarisation, summary statistics, visualisation, inaccurate values detection, data normalisation and data dimensionality reduction, provides a more understandable dataset format with data lacks fulfilled, features scaled and dataset reduced for the preference of future computation. Moreover, during data modelling, different training sets were defined to test model performance on whether imbalanced datasets, a use of cross-validation to avoid over-fitting, as well as model scale adjustments and kernel function selection have all valued the final choice of classifiers.

The answer to the research question "How can Danish brewers use machine learning techniques, train on historical data, predict backorders in their early stage of the supply chain." is stated in the report in the chapter proposed solution. From the chapter Evaluation, the support vector machine with cubic kernel gives the best results among other classifiers and as well the traditional method, which gives a positive answer to research question two: "Are machine learning predictions more efficient than the traditional forecasting".

However, more works could be done to discover rules that can result in kernels' best performance. The results from this project show, under certain circumstances, some machine learning techniques can give better prediction than that from the traditional

ways. Especially trained on the historical data that used in this project, a Support vector machine with cubic polynomial kernel gives the best prediction. By collecting more historical data or more testing data, the robustness and accuracy of the classifiers can get better.

Comparing from the overall accuracy, the cubic SVM shows much better result by providing ACC as 88% against the traditional predictor which gives 75% accuracy. However, by looking at AUC, SVM only has a slight advantage by having 81% over 79% from the traditional predictor. One important message that needs to be mentioned is that the data that the traditional prediction used to predict are included in the training set, which is a linear prediction. Overall, SVM did not provide a better true predictive rate over the competitor, which brings up the thought: whether principle component analysis has successfully selected the most significant features to turn into principle components or not. In many real world cases, when the training set has more entries or observations than the number of features, it is as crucial to select features carefully, or sometimes engineers need to create and add new features manually. Feature selection and extraction requires a thorough understanding of the business and as well the dataset. A satisfying technical solution to a business problem will involve the activities from feature design and data collection till the model evaluation and robustness testing before it becomes a general tool. In this project, the best SVM model has 6561 support vectors over 14479 observations from the dimension-reduced training set, which also shows this trained model cannot become a very general classifier to use in different companies result from the high rate of support vectors. On the other hand, it also shows the regularity of the training set is medium, which may lead to a better feature selection as well.

Support vector machines could perform very well in binary classification where an outlier detection and dimensionality reduction are conducted. However, knowledge about how to implement efficient and effective approximate training is limited, so there are also many cases that SVM cannot give a promising result. At this point there are dozens of variants of the original SVM algorithm that have been developed for various interesting situations - e.g. there are different kernels, there are different optimisation techniques, there are variants for learning with latent variables[48], for learning structured outputs [49]. There are ways of adjustments to get the best result from this classifier.

While defining a training set, it is essential to detect the outliers; one needs always to be careful to reduce the dataset since sometimes outliers can also be useful for model training. It can be tricky sometimes to define the best amount of noise in a dataset that can help the classifier perform in the best way. During the definition of a training set, especially in real cases while using real data, data preprocessing is necessary: it helps clean the data also define if the data is imbalanced. An imbalanced dataset can confuse

one person with a high overall accuracy rate, where the true positive rate could be much smaller.

In this project, many variables need to be adjusted and controlled. While dealing with missing value, outliers, to dimensionality reduction and model training, many assumptions needed to be made. Generally, among the best performers, the model with the fewest assumptions should be selected. A model that is with the least assumptions is the most robust and can be used in many scenarios. As being small and traditional is a core value of craft brewers, it is important that the prediction method be easy to use. That being said, an approach of using SVM predictions might be the best fit for this case.

Various future works can be extended from this project if given more time. First of all, many heavy computations caused from kernel adjustment can be tried out on a better computer, for a more precise algorithm screening. Second of all, more advanced predictions can be achieved. As the models discussed in this project are about predictions from the early stage in the supply chain, which is relying on the producer's existing data. When more collaboration and sharing information can appear in the business in the future, more advanced predictions can be achieved, which also requires more knowledge from not only the business but also how to collect data and choose features. Last but not least, a meeting with the brewer can be arranged to discuss more issues in the predictions from their experience, which can help with feature selection and data collection. In conclusion, it is stated that many improvements could be made based on the trained model. One-class training could also be used for the extremely imbalanced training set.

A successful backorder prediction can reduce the bullwhip effect, raise customer and partners satisfaction, and as well help brewers be more proactive in the ever changing market. By being organised and knowing the possible future demand, both operation management and supply chain management can be more efficient by the time saving and fast reactions that valued from the forecasting.

Bibliography

[1] Bryggeriforeningen. The danish beer revolution. *The Danish Breweries Association*, October 2015.

[2] Mikkel N. Schmidt Tue Herlau and Morten Mørup. *Introduction to Machine Learning and Data Mining*. Technical University of Denmark, 2016.

[3] Christopher M Bishop. *Pattern recognition and machine learning*. Springer, 2013. ISBN 0-387-31073-8.

[4] Tibshirani R. Hasdite T. and Friedman J. H. *The elements of statistical learning*. Springer, 2017. ISBN 978-0-387-84857-0.

[5] S. B. Kotsiantis. Supervised machine learning: A review of classification techniques. *Informatica*, 31(2007):249–268, 2007.

[6] MathWorks. Mathwork support. *https://se.mathworks.com/support*, 2017.

[7] M. Martens O. Mejlholm a. Beer identity in denmark. *Food Quality and Preference*, 17(2006):108–115, November 2005.

[8] American Brewers Association. www.craftbeer.com. *American Brewers Association*, Accessed on 13th Auguest, 2017.

[9] Barry Render Jay Heizer. *Operations Management,Sustainability and Supply Chain Management*. Pearson, 2014. ISBN 978-0-13-292114-5.

[10] William J. Steven. *Operations Management*. McGraw-Hill/Irwin, 2010. ISBN 978-0-07-352525-9.

[11] J.D. Sterman. Modeling managerial behaviour: Misperceptions of feedback in a dynamic decision making experiment. *Management Science*, 35(3):321–339, 1989.

[12] R. Croson and K. Donohue. Impact of pos data sharing on supply chain management: an experiment. *Production and Operations Management*, 1(12):1–12, 2005.

[13] R. Croson and K. Donohue. Upstream versus downstream information and its impact on the bullwhip effect. *System Dynamics Review*, 3(21):249 260., 2005.

[14] Katok E. Wu, D.Y. Learning, communication and bullwhip effect. *Journal of Operations Management*, 24:839–850, 2006.

[15] D.E. Cantor and E. Katok. Production smoothing in a serial supply chain: a laboratory investigation. *Transportation Research Part E*, 48:781 794., 2012.

[16] V. Madhusudanan Pillai* and T. Chinna Pamulety. Impact of backorder on supply chain performance an experimental study. *International Federation of Automatic Control*, June 2013.

[17] Rustam Vahidov Real Carbonneau, Kevin Laframboise. Application of machine learning techniques for supply chain demand forecasting. *European Journal of Operational Research*, 184(2008):1140–1154, December 2008.

[18] Mark A. Hall Ian H. Witten, Eibe Frank. *Data Mining,Practical Machine Learning Tools and Techniques*. Elsevier Inc., 2010. ISBN 978-0-12-374856-0.

[19] Aleksander Kol cz Nitesh V. Chawla, Nathalie Japkowicz. Special issue on learning from imbalanced data sets. *Sigkdd Explorations*, 6:1–6, 2004.

[20] L. K. Roweis, S. T.; Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500), 2000.

[21] M. Kirby and L. Sirovich. Application of the karhunen-loeve procedure for the characterization of human faces. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(1): 103–108, 1990.

[22] Daniel L. Swets and Juyang Weng. Using discriminant eigenfeatures for image retrieval. *Pattern Anal. Mach. Intell.*, 18(8):831–836, 1996.

[23] Rudy Setiono Huan Liu, Hiroshi Motoda and Zheng Zhao. Feature extraction for outlier detection in high-dimensional spaces. *Workshop and Conference Proceedings: The Fourth Workshop on Feature Selection in Data Mining*, 10:66–75, 2010.

[24] I. T. Jolliffe. *Principal Component Analysis*. Springer, 2002.

[25] Hotelling.H. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:417441, 1933.

[26] K. Pearson. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science, Sixth Series*, 2:559572, 1901.

[27] M. Genton. Classes of kernels for machine learning: A statistics perspective. *Journal of Machine Learning Research*, 2:299–312, 2001.

[28] S. B. Kotsiantis. Supervised machine learning: A review of classification techniques. *Informatica*, 31:249–268, July 2007.

[29] Costas D. Maranas Anshuman Gupta. Managing demand uncertainty in supply chain planning. *Computers and Chemical Engineering*, 27:1219–1227, February (2003.

[30] S. Raghunathan. Interorganizational collaborative forecasting and replenishment systems and supply chain implications. *Decision Sciences*, 30(4), 1999.

[31] Padbanaban V. Whang S. Lee, H.L. The bullwhip effect in supply chains. sloan management review. *Sloan Management Review*, Spring 1997.

[32] Disney S.M. Lambrecht M.R. Towill D.R. Dejonckheere, J. Measuring and avoiding the bullwhip effect: A control theoretic approach. *European Journal of Operational Research*, 147(3):567–590, 2003.

[33] HAIBO HE YUNQIAN MA. *Imbalanced learning: Foundations, Algorithms, and Applications*. IEEE press, 2013. ISBN 9781118074626.

[34] Philip S. Yu Charu C. Aggarwal. Outlier detection for high dimensional data. *VLDB J.*, page 211221, 2005.

[35] Lawrence O. Hall Nitesh V. Chawla, Aleksandar Lazarevic and Kevin W. Bowyer. Smoteboost: Improving prediction of the minority class in boosting. *In PKDD*, page 107119, 2003.

[36] Adam Kaplan and Eric F Lock. Prediction with dimension reduction of multiple molecular data sources for patient survival. *Cancer Informatics*, 16:1–11, 2017.

[37] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1995.

[38] MA (US); Roger Dingledine Somerville MA (US); Debra Gesimondo Stow MA (US) Nick Mathewson, Cambridge. Support vector machines for prediction and classification in supply chain management and other applications. *U.S.Patent Application Publication*, page US 2004/0034612 A1, (2004.

[39] Rustam Vahidov Real Carbonneau and Kevin Laframboise. Machine learning-based demand forecasting in supply chains. *International Journal of Intelligent Information Technologies*, 2007.

[40] Wang Guanghui. Demand forecasting of supply chain based on support vector regression method. *Procedia Engineering*, 2011.

[41] Chih-Chung Chang Chih-Wei Hsu and Chih-Jen Lin. A practical guide to support vector classification. *National Taiwan University*, 2010.

[42] Stan Z. Li Wei Liu, Yunhong Wang and Tieniu Tan. Null space approach of fisher discriminant analysis for face recognition. *In ECCV Workshop BioAW*, page 3244, 2004.

[43] Edwin M. Knorr and Raymond T. Ng. Algorithms for mining distance-based outliers in large datasets. *In VLDB*, page 392403, 1998.

[44] Subcommittee: E11.10. Standard practice for dealing with outlying observations. *ASTM International*, 2002.

[45] G. Weiss and F. Provost. Learning when training data are costly: The effect of class distribution on tree induction. *Journal of Artificial Intelligence Research*, 19: 315354, 2003.

[46] Chih-Chung Chang Chih-Wei Hsu and Chih-Jen Lin. A practical guide to support vector classification. *National Taiwan University*, 2003.

[47] Carlotta Domeniconi Bojun Yan. Nearest neighbor ensemble. *George Mason University*, 2004.

[48] Thorsten Joachims Chun-Nam John Yu. Learning structural svms with latent variables. *International Conference on Machine Learning*, 2009.

[49] Thorsten Joachims Thomas Finley. Training structural svms when exact inference is intractable. *International Conference on Machine Learning,*, 2008.

A1. Description of the dataset

The dataset used in this project comes from website Kaggle.
Kaggle is a platform for predictive modelling and analytics competitions in which companies and researchers post data and statisticians and data miners compete to produce the best models for predicting and describing the data. This crowdsourcing approach relies on the fact that there are countless strategies that can be applied to any predictive modelling task and it is impossible to know at the outset which technique or analyst will be most effective.
Link to the data: $https://www.kaggle.com/tiredgeek/predict-bo-trial$
Link to the licience: $https://creativecommons.org/licenses/by-nc-sa/4.0/$
This web-page also shows a few relevant work done with the same dataset used in this project. The works were done and uploaded by volunteers whose identity cannot be verified, therefor there works were not mentioned in the section of similar works since the works' authority is undefined.

When import this data to Matlab, one need to make sure the data output type is column vectors, which is an essential way for the attributes get distinguished into numeric and categorical data. It will do a big favor of convenience for future cleaning and analysis.

A2. Coding for data preparation

During early stage of machine learning, it is necessary to clean data to make sure as least data issues will influence the future model training. Cleaning steps include binarilsation, replace missing and negative values and convert a numeric matrix.

Data pre-processing

```
%% Binarize attributes with strings
nominal_data= went_on_backorder;
binarized_data=binarizeData(nominal_data) ;
% Merge three colomns to one
a=binarized_data;
for i=1:size(a,1)
    if a(i,2) == 1
        d(i,1) =0;
    else
        d(i,1)=1;
    end
end
%% Update the attribute
went_on_backorder =d;

%% Create a numeric matrix
Original_training_data = [national_inv,lead_time,in_transit_qty,
forecast_3_month,forecast_6_month,forecast_9_month,sales_1_month,sales_3_month,
sales_6_month,sales_9_month,min_bank,potential_issue,pieces_past_due,
perf_6_month_avg,perf_12_month_avg,local_bo_qty,deck_risk,oe_constraint,
ppap_risk,stop_auto_buy,rev_stop,went_on_backorder];

%%  Look for missing value/ empty entries
IM = ismissing(Original_training_data);
sum(IM)

%% Fill empty entries and update the attribute
C = mean(lead_time,'omitnan');
Original_training_data(:,2) = fillmissing(lead_time,'constant',C);
% Remove NaN values in the dataset
Original_training_data=rmmissing(Original_training_data);
```

Data normalization

```
%% Seperate training data and label
cleaned_trainingdata=Original_training_data(:,1:21); label=Original_training_data(:,22);
%% Training set Normalisation on scale [0,1]. *Note: scale (-1,1) can be done by using mapminmax
data=cleaned_trainingdata;
%% caculate x-x min
Y = bsxfun(@minus, data, min(data));
% caculate max(x)-min(x)
M = bsxfun(@minus, max(data), min(data));
% x'=(x-min(x))/(max-min)
ScaledTrainingData= bsxfun(@rdivide, Y, M);

%% Testingset Normalisation on scale [0,1]
data=cleaned_testset;
% caculate x-x min
Y = bsxfun(@minus, data, min(data));
% caculate max(x)-min(x)
M = bsxfun(@minus, max(data), min(data));
% x'=(x-min(x))/(max-min)
ScaledTestingData= bsxfun(@rdivide, Y, M);

%% Prepare for PCA: Convert training data and testing data
wholeset=[ScaledTrainingData;ScaledTestingData];

%% Perform PCA
[coeff,score,latent] = pca(wholeset);
[ReducedData,prinCoeffs] = principleComponents( wholeset,coeff,latent );

%% Seperate training data and test data
T=numel(ScaledTrainingData(:,1));
ReducedTrainingdata= ReducedData(1:T,:);
ReducedTestingdata= ReducedData(T+1:end,:);
%% Final trainingset for SVM
Final_trainingset=[ReducedTrainingdata,label];
```

Sampling balanced dataset

```
% data=[ScaledTrainingData,label];
%%seperating negative class value
for x=1:size(data,1)
     if b(x,22)==1
         b(x,:)=[];
     end
end
%% seperating positive class value
  for m=1:size(data,1)
    if data(m,22) ==1
        b(n,:)=data(m,:);
        n=n+1;
    else
        n=n;
        m=m+1;
    end
end
%% Sampling
k=sum(backorder(:,22));
y = datasample(no_backorder,k);
sample=[backorder;y];
S_sample=T(:,1:21); label_sample=T(:,22);

%% Normalisation with test data
data=[S_sample;S_test];
Y = bsxfun(@minus, data, min(data));
% caculate max(x)-min(x)
M = bsxfun(@minus, max(data), min(data));
% x'=(x-min(x))/(max-min)
Normal= bsxfun(@rdivide, Y, M);
k= numel(label_sample);
S_N_s=Normal(1:k,:);
S_N_t=Normal(k+1:end,:);
%% PCA
[coeff,score,latent] = pca(Normal);
[PCA_set,prinCoeffs] = principleComponents(Normal,coeff,latent );
Ptrain=PCA_set(1:k,:);
Ptest=PCA_set(k+1:end,:);
```

Traditional forecasting method

k= current inventory level - demand forecast - minimum recommended inventory level when $k > 0$, the product will be recognized as not backorder, when $k < 0$, the product will be recognized as backorder.

```
m=4;
for a=1:size(X,1)
    n= X(a,1)-X(a,m);
   if n-X(a,11)>0
       k(a,1)=0;
   else k(a,1)=1;
   end
end
```

A3. Interview

Researching questionnaire

I would like to know more about your supply chain, the ideal supply chain may follow like:

buying ingredients - producing - packing - put in stock
- transporting - in Denmark
- to other countries

So the questions will be following the stages above.

Producing beers

how much beer do you get from one batch
1250 liters
how many different beers can you make at a time (25 different ones ten times a week)
how long does it take to prepare the equipment for a new batch
cleaning : a couple hours.
where do you get your ingredients for producing beer? Name several places
80 danish 10 uk 10 germany. shipping through EU importer.
how many rare(hard/slow to get/buy) ingredients do you use in the process
brazil take 6 weeks to delivery. could be popular.
how long time does it take to produce your best selling beer
4 weeks.
how long is the expiration date on that beer
1 year.
what is the average delivering time on the raw materials for producing beer
4 weeks.
what kind of problems do you consider when your producing beers
how to produce, machines mistakes, personal mistakes.

Your Products and Storage:

how many beers do you produce per year (total number)
8000
how many kinds of beer do you produce
25

FIGURE 1: Interview questionnaire 1/2

Researching questionnaire

I would like to know more about your supply chain, the ideal supply chain may follow like:

buying ingredients - producing - packing - put in stock
- transporting - in Denmark
- to other countries

So the questions will be following the stages above.

Producing beers

how much beer do you get from one batch
1250 liters
how many different beers can you make at a time (25 different ones ten times a week)
how long does it take to prepare the equipment for a new batch
cleaning : a couple hours.
where do you get your ingredients for producing beer? Name several places
80 danish 10 uk 10 germany. shipping through EU importer.
how many rare(hard/slow to get/buy) ingredients do you use in the process
brazil take 6 weeks to delivery. could be popular.
how long time does it take to produce your best selling beer
4 weeks.
how long is the expiration date on that beer
1 year.
what is the average delivering time on the raw materials for producing beer
4 weeks.
what kind of problems do you consider when your producing beers
how to produce, machines mistakes, personal mistakes.

Your Products and Storage:

how many beers do you produce per year (total number)
8000
how many kinds of beer do you produce
25

FIGURE 2: Interview questionnaire 2/2