Modelling COVID-19 Individual Risks in Sweden Using Spatial Information, Statistics and Machine Learning

Analysis of COVID-19 cases in sweden by training machine learning algorithms

MVEX03

Master's Thesis for the Department of Mathematical Sciences

Lukas Fu

Department of Physics

CHALMERS UNIVERSITY OF TECHNOLOGY

Gothenburg, Sweden 2024

www.chalmers.se

MASTER'S THESIS 2024

Modelling COVID-19 Individual Cases in Sweden

Analysis of COVID-19 cases in sweden using machine learning algorithm

MVEX03

Department for Mathematical Sciences

CHALMERS UNIVERSITY OF TECHNOLOGY

Gothenburg 2024

Lukas Fu
lukasfu@student.chalmers.se

Supervisors:

Ottmar Cronie
Senior Lecturer
Applied Mathematics and Statistics
Chalmers University of Technology and University of Gothenburg

Bin Han
PhD, Research Assitant
School of Public Health and Community Medicine, Sahlgrenska Academy
University of Gothenburg

CHALMERS
UNIVERSITY OF TECHNOLOGY

Modelling Spatial Relations of COVID-19 Cases in Sweden using Statistics and Machine Learning
- Analysis of Covid-19 cases in sweden using neural network algorithm
MVEX03
Lukas Fu

Supervisor: Ottmar Cronie, Bin Han
Examiner: Philip Gerlee, Department of Mathematical Sciences

Master's Thesis 2024
Department for Mathematical Sciences
Chalmers University of Technology
SE-412 96 Göteborg

Typesetting in LaTeX
Gothenburg, Sweden 2024

Preface

The project idea was proposed by Ottmar Cronie, the supervisor of this project. Many of the technical challenges I faced, especially having to adjust to a new programming language, were overcome thanks to my other supervisor, Bin Han, with whom I received plenty of assistance from. Ottmar together with Bin provided me with constant and helpful advice during the course of the project, and I am thankful that I was offered the opportunity to work with them. I also want to extend an extra word of gratitude to Martin Adiels from the University of Gothenburg, as well as the University of Gothenburg, for providing me an environment and the data to work with. Martin has a very busy schedule, so I am grateful that he could allocate time to discussing the project with me.

Modelling COVID-19 Cases in Sweden
- Analysis of Covid cases in sweden using machine learning algorithms
MVEX03
Lukas Fu
Master's Thesis 2024
Department for Mathematical Sciences
Chalmers University of Technology

Abstract

The Covid-19 pandemic was a modern time pandemic that lasted a little over two years, and caused a severe social and economical disruption on a worldwide scale. Using data consisting of individual and DeSO covariates of the population of Sweden, sourced from Statistics Sweden and the Public Health Agency of Sweden, this project aims to model individual risks of Covid-19 using machine learning algorithms, and to extract information on feature importance from the fitted models. The models tested include logistic regression, random forest, support vector machines and neural network, and Shapley values were additionally evaluated for random forest in an attempt to gain more insight into the feature relation to the prediction. The logistic regression and random forest models both resulted in feature importances consisting of a mixture of individual and DeSO features, where features such as age, level of education, and living conditions for both the DeSO and the individual, along with income and occupation of the individual, showed high importance. Support vector machines and neural network models did not produce any useful results due to computational limitations. The large size of the data set was a consistent hindrance in this project, as many issues were caused by computational costs, and many of the improvements on optimization in this project are centered around handling these costs. Further research may entail in optimizing performances of presented or alternate models, but may also expand to more thoroughly analyse the spatial and temporal dependencies of disease cases. While the results of this project might not be particularly significant on its own, this project may still provide a basis for future developments in pandemic data analysis.

Keywords: Modelling, Machine Learning, Neural Network, Logistic Regression, Random Forest, Support Vector Machine, SHAP, COVID-19

Contents

1 Introduction

The Covid-19 pandemic, also known as the coronavirus pandemic, that lasted from early 2020 to the middle of 2023 was a modern time pandemic that caused a severe social and economical disruption on a worldwide scale. The recession caused by the pandemic has also been compared to that of the Great Depression back in the 20th century [1]. It is difficult to compare the modern pandemic with larger ones throughout human history such as the Spanish Flu pandemic and the Black Death plague, as modern society is better equipped in many regards to deal with such pandemics. While the cost caused by the pandemic historically has been human lives, modern society is better equipped to prevent the loss of lives, and the costs therefore come in the form of resources spent on preventative measures. It is not known how the Covid-19 pandemic compares to historical pandemics if the preventative measures were not implemented, and therefore how impactful it was comparatively [2].

Due to how widespread the disease was early in the pandemic and how common infection and cases of death were, along with the immediate action of social restriction, many started to grow uneasy over the possible impending economic crisis. This led to panic-buying and stockpiling of various supplies, which combined with the workforce being drastically reduced due to the social restrictions caused a shortage on a large scale. This can be exemplified using the toilet paper shortage that occurred in the middle of 2020, at the beginning of the pandemic. Furthermore, social restrictions caused workforce to be reduced in all economic sectors and led to many jobs being lost [3]. Additionally, the educational sector was forced to adapt to unscheduled closure of schools and change to online teaching. It is estimated that more than 94% of students in the world were affected by this change, where the quality of education was drastically reduced as education systems were not equipped to deal with the situation [4].

More recently, the tools of artificial intelligence, machine learning and deep learning have risen in popularity for application in an increasing amount of fields, healthcare and medicine being no exception. The difference between the traditional analysis of statistical models and the modern machine learning models lie in the objective of each model. Statistical models primarily try to find and explain relations between variables and to test hypotheses, while machine learning simply tries to use the data available to maximize its own predictive capabilities. Machine learning models can therefore take into account factors that the statistical model would not consider, based on patterns in the data itself. This essentially allows the data itself to be formed into a model, which could possibly be considered as the closest representation of reality if the data is obtained from a real environment. Machine learning models are however difficult to interpret, as their design tend to not involve the user in the calculations at all, creating a black box of unknown operations. While the versatility of machine learning and deep learning models can seem desirable, the reality of an ideal model may not be as simple as picking the most well-rounded and all encompassing model available. As the American physicist and computer scientist David Hilton Wolpert stated in regard to machine learning - the "no free lunches" theorem. Even if the machine learning algorithm may seem alluring for its wide capabilities at first glance, things may not be so simple.

1.1 Purpose of project and research question

The many effects of COVID-19 has led to severe effects on a worldwide scale, and this project is conducted in the interest of learning about the pandemic and increasing knowledge for future preparedness. The purpose of this project is to explore various models and algorithms of regression and classification to find the importance of parameters (or: variables, covariates, features) in COVID-19 case-data for the country of Sweden. The

parameters include information of individuals in Sweden, such as social-demographics and personal health conditions, as well as spatial and temporal statistics in demographic statistical areas to understand how the surroundings of an individual affects the probability for COVID-19 cases. It is interesting to learn whether individual situations or surrounding conditions play a larger contributing role, or if it is some combination of the two.

The definitive goal of the project is to create a classification predictor that predicts the Covid-19 outcome based on the personal information and information of the demographic statistical area, or DeSO, of an individual, and judging through the predictor the importance of various features. The variable importance is particularly of interest, as knowing the influence of factors relating to Covid-19 cases could potentially lead to a better understanding and mitigation of future pandemics. The exact implications of the variable importance will be discussed but not explored deeply as it is not relevant to the target subject of the report. The project also does not delve into disease transmission, as the data is not sufficient for such an analysis. The classification is on any type of Covid-19 case and is therefore binary, non-covid population or Covid-19 population. The project mainly aims to try various models to see if any of the algorithms produce any reasonable results with the main objective of finding the variable importance, and future projects can continue to explore other models based on what worked or optimize the ones that end up working.

2 | Materials

The data used for this project consists roughly 10 million individuals of the Swedish population, which contains individual information such as social-demographic and personal health information and if there have been any case of Covid-19 during the pandemic period. There is also data for Demographic Statistic Areas (DeSO) for Sweden to be used for relating the Covid-19 case dependency on spatial features. The data is sourced from Statistics Sweden's register and from the Public Health Agency of Sweden's register. The Longitudinal integrated database for health insurance and labour market studies register covers sociodemographic data, and SmiNet covers the data over communicable diseases in Sweden.

2.1 Sociodemographic data

The total number of data points is 9905875, which correspond to the included individuals, and 198 features for each individual. The features in the personal information dataset include information such as gender, age, medical information, education status, living condition and Covid-19 status, among others.

Tables of all features in the data can be seen in the appendix, chapter A.

2.2 Demographic statistical areas

The statistical areas known as DeSO (demographic statistical areas in English) are known for each individual and is a part of the personal information. The DeSO data is a separate dataset that contains all the geographical areas in Sweden along with various demographical features for the areas, such as the number of individuals of a certain age range, the educational status of an individual. The number of cases that end up in ICU or death are included in the data, but not used as features.

Additionally, the data from previous spatial statistical analysis is included, such as proportions of age, work and education, or Covid-19 test case, ICU test case or death case counts of certain months of the pandemic for the DeSO.

Tables of all features in the data can be seen in the appendix, chapter A.

2.2.1 Variable descriptions

The data for the demographic statistical areas are sourced from Statistics Sweden, SCB. The following is a list of variable names from the dataset with a description. Note that each variable applies to each DeSO, meaning that variables such as total individuals refer to the total amount of individuals in that specific area.

2.2.2 Variable descriptions of spatially aggregated data

The data of each DeSO is aggregated into proportions in order to contextualize the feature values to the area. The features themselves are the same as the previous section. It is different to consider proximity to a certain number of individuals of a certain age, e.g. 50 individuals of age 20, if the population in the area is 1000, or 10000.

2.3 Combined dataset

The personal data for each individual contains a feature that refers to their DeSO location, the information of which is further expanded upon in the demographic statistical areas dataset. Each individual therefore consist of 375 features, combining the features of both datasets.

3 | Method

3.1 Algorithms and testing

The data set used for this project is large and the relationship between disease spread, and the various factors are complex. The advantage of machine learning is its capability to fit high-dimensional complex relationship, an approach which is viable due to the large amount of data.

It has however been shown that deep learning models on their own are not necessarily an improvement over simpler models [5]. Shwartz-Ziv and Armon compared in their article tree ensemble models, models comprised of multiple tree models, in their case extreme gradient boosting (XGBoost), to other proposed deep learning models claimed to outperform XGBoost [5]. Their study show that XGBoost still generally outperform these deep learning models across datasets, even on the datasets for which the deep models are proposed. Additionally, ensembles of deep learning models were tested, both in combinations and without XGBoost, as well as an ensemble of simpler non-deep models such as SVM and CatBoost. The best performing models ended up being the ensemble of deep learning models with XGBoost, followed by the simple ensemble and XGBoost alone.

There are several instances showing that deep learning models underperform relative to expectations, and that simpler models are capable of performing to a similar or even higher level comparatively. The principle of "no free lunches", meaning that no solution fits all problems, should by no means be overlooked. Therefore, in this project, while deep learning models are interesting to explore, simpler models may perform better.

3.2 Logistic regression with elastic net regularization

While linear models are well suited for regression problems, it is not adequately suited for classification as it does not output probabilities. Take a binary classification of 0 and 1 for example: a linear model fits the best hyperplane that minimizes the distance between the points and the hyperplane. The model interpolates between then points as well as extrapolates, and is in such able to output values over and under 0 and 1. A better suited method for classifications is the logistic regression model [6].

Logistic regression is a simple method for binary and linear classification problems that outputs a probability of either classification. For classification of models of three or more outcomes, a multinomial logistic regression would be used instead, or in the case of outcomes falling into a predetermined order, an ordinal logistic regression.

The logistic regression model originates from the linear regression, where through transformation using the sigmoid function, the continuous value output of is altered to a categorical value output, mapping any real valued output into the interval 0 to 1. This transformation is done by the logistic function. Given an independent input feature $\boldsymbol{x}$ of dimensions $n \times m$ and the dependent variable Y being binary $\{0, 1\}$, the multilinear function, or linear regression, is expressed as

$$z = \beta_0 + \sum_{i=1}^{N} \beta_i x. \tag{3.1}$$

Setting the input of a sigmoid function to the z of the linear regression gives a probability of predicting y as a value between 0 and 1:

$$\sigma(z) = \frac{1}{1 - e^{-z}}, \tag{3.2}$$

which tends toward 1 if z approaches ∞ and 0 if z approaches $-\infty$, and is always bounded between 0 and 1. The probabilities of each class can be measured as

$$P(y = 1) = \sigma(z), \tag{3.3}$$

$$P(y = 0) = 1 - \sigma(z). \tag{3.4}$$

Lastly, by using the odds ratio, defined as the ratio between the probability of something occurring and not occurring:

$$\frac{p(x)}{1 - p(x)} = e^z, \tag{3.5}$$

and applying the natural logarithmic function to the odds, the logistic function can be expressed as

$$p(x, \beta_0, \beta) = \frac{1}{1 + e^{-\beta_0 + \beta x}}. \tag{3.6}$$

The likelihood function of the logistic regression is written as

$$L(\beta_0, \beta) = \prod_{i=1}^{N} p(x_i)^{y_i}(1 - p(x_i))^{1-y_i}, \tag{3.7}$$

which gives the log-likelihood function expression

$$l(\beta_0, \beta) = log(L(\beta_0, \beta)) = \sum_{i=1}^{N} y_i log(p(x_i)) + (1 - y_i) log(1 - p(x_i)). \tag{3.8}$$

The parameters of the model can be estimated by minimizing the residual sum of squares (RSS) [7]:

$$(\hat{\beta_0}, \hat{\beta}) = argmin_{\beta_0, \beta_1} \frac{1}{2n} \sum_{i=1}^{n} (y_i - p(x_i, \beta_0, \beta))^2. \tag{3.9}$$

Though many regularization methods have been researched in the past two decades, the elastic net will be in focus, as described by Zou and Hastie [8][9]. The elastic net consists of a mixture of two penalties: the ridge penalty $||\beta||_2$, and the LASSO (least absolute shrinkage and selection operator) penalty $||\beta||_1$, described as

$$||\beta||_2 = \sum_{j=1}^{p} \beta^2, \text{ and } ||\beta||_1 = \sum_{j=1}^{p} |\beta_j|. \tag{3.10}$$

Minimizing the sum of the RSS and the elastic net regularization term:

$$L_\lambda(\beta) = RSS(\beta) + \lambda R(\beta) = \frac{1}{2n} \sum_{i=1}^{n} (y_i - p(x_i, \beta_0, \beta))^2 + \lambda [\frac{(1 - \alpha)}{2} ||\beta||_2 + \alpha ||\beta||_1], \tag{3.11}$$

where λ is the tuning parameter for the regularisation and and R is the regularisation function where the tuning parameter λ is applied. The variable α is the parameter for the elastic net penalty, bridging between a lasso regression ($\alpha = 1$) and a ridge regression ($\alpha = 0$).

Minimising the loss function finds the optimal solution to the log-likelihood function l, as described by the generalized linear model:

$$\min_{(\beta_0, \beta)} \frac{1}{N} \sum_{i=1}^{N} w_i (y_i, \beta_0 + x_i^T \beta) + \lambda [\frac{(1 - \alpha)}{2} ||\beta||_2 + \alpha ||\beta||_1], \tag{3.12}$$

over a grid of values of λ that covers the entire range of solutions. Additionally the weights are denoted as w_i and shows the weight of each feature involved [10].

Some disadvantages of logistic regression are similar to that of linear regression models - the expressiveness of the model is very restricted and thus struggles in predictions for more complex systems [6]. The interpretation of the output given by the weights are also not straightforward, as they are multiplicative rather than additive. Additionally, the logistic regression model suffers from complete separation, where in the presence of a feature that would perfectly separate the classes, the model would no longer be able to continue training. However, due to the addition of a penalizing factor shown in equation 3.11, complete separation do not pose an issue. Besides, if one feature completely separates the classes, a machine learning model is not necessary. Lastly, the model is only capable of handling numerical data, and thus data pre-processing would need to be conducted in order to adhere to this limitation. On the upside, the model is interpretable in the sense that an increase of a covariate influences the outcome. Furthermore, compared to other classification models that only output a final classification, logistic regression is capable of producing probabilities of the classes.

3.2.1 Cross validation and hyper-parameter optimization

Cross validation is a model validation technique used for assessing the results of an analysis based on how it generalizes to an independent data set [11]. This method is often used in prediction models where it estimates the accuracy and performance in practice by resampling and splitting the data into training and testing splits, possibly over multiple iterations. The method of partitioning the data into subsets used for training, where the data is analyzed and model is formed, and subset used for testing or validation, where the model performance is evaluated, is commonly used various algorithms. Cross validation essentially combines and averages measures of fitness in prediction to calculate a more accurate estimate of model prediction performance. This type of validation is typically referred to as the hold-out validation technique. Hold-out validation can sometimes suffer from an uneven distribution of classes in the data partitioning, which can be solved by evenly distributing the classes when partitioning.

Another type of validation is the resampling method called *k-fold cross validation*. For this method the entire data set is divided into k number of equally or mostly equally sized subsets. For the values of $j = \{1, 2, \ldots, k\}$, the j-th subset is reserved as the test data, while the rest are used for training. Validation is then done using the test data, and the algorithm repeats this process on the remaining j subsets, after which the error rates for all j are averaged to obtain the overall error rate estimate. The advantage of k-fold cross validation is its balance of bias and variance. It is also able to use the entire data set as training data, which is useful not only for data sets with low amounts of data points, but also avoids the possibility of uneven splits of data and therefore the improved values of variance.

The k-fold cross validation is utilized to find the optimal hyperparameters of the logistic model regression, specifically parameters α and λ, seen in equation 3.11, both of which are related to the penalization [10]. For each value of tested α between the values 0 and 1, a k-fold cross validation is performed. Each fold uses different values of λ and calculates their corresponding errors along with the upper and lower standard deviation. The method for calculating this error can vary, and for logistic regression, some examples are: the mean squared loss, mean absolute error and misclassification error [10]. Two values of λ are of interest: λ_{min}, corresponding to the λ resulting in the minimum error, and λ_{1se}, corresponding to the largest λ within a standard deviation of the minimum. Due to the risk of overfitting by under-penalizing, λ_{1se} is typically chosen over the minimum. The α which resulted in the lowest error, along with the corresponding λ_{min} and λ_{1se}, are then chosen as the regularization parameters.

The hold-out validation is used for the remainder of the models due to computational cost; running computationally costly models multiple times is not practical, and was not attempted for this project. This does lead to a lack of optimization of model parameters for the affected models, namely support vector machine and neural network. Ideally, a k-fold cross validation should be applied to better choose the hyperparameters of the models. The split of training to test data is 7:3 for the three aforementioned models.

3.3 Random forest

Random Forest is a machine learning algorithm for ensemble learning method for classification and regression that operates by constructing multiple decision trees to reach a single result [12], visually exemplified in figure 3.1. While decision trees can be used on their own in the decision tree learning algorithm, it has a tendency for bias and overfitting due to having low bias but high variance, meaning that it fits training points very

well but performs very poorly on new test data. By combining multiple of them together, the variation and bias are reduced, particularly when the trees are uncorrelated to each other. Ensemble learning entails a combination of models in order to obtain a better predictive performance compared to the individual models, and the concept of combining weaker models to create a strong model is called boosting.

Each tree in the random forest is comprised off of data through bootstrap aggregation or bagging, where the data is drawn from a training set with replacement. Given a training set $X = x_1, \ldots, x_n$ with responses $Y = y_1, \ldots, y_n$, bagging repeatedly (B times) selects random samples with replacement of the training set and fits the trees based on these samples. For $b = 1, \ldots, B$, the sampled data for training X_b and Y_b are then used to train a regression or classification model. The value of B can be interpreted as the number of samples or number of trees, the quantity of which varies depending on the training set. For regression, the algorithm will finalize by averaging the individual decision trees, whereas for classification the most frequent categorical variable will yield the predicted class. For an ensemble of trees $\{T_b\}_1^B$, to make a prediction at a new point x [13]:

Regression: $\frac{1}{B}\sum_{b=1}^{B} T_b(x)$

Classification: Let $\hat{C}_b(x)$ be the class prediction of the b-th random forest tree.
Then $\hat{C}_{rf}^B = $ *majority vote* $\{\hat{C}_b(x)\}_1^B$.

Bagging is especially well suited to low-bias, high-variance procedures such as trees, as individual trees can become relatively unbiased if grown sufficiently deep. Decision tree learning therefore greatly benefits from averaging, due to them being notoriously noisy. Furthermore, since each tree generated through bagging is identically distributed, the expectation of the average of B trees is the same as the expectation of the individual trees, thus the bias of the bagged trees is the same as that of the individual trees [13], leading to the only improvement coming through variance reduction. However, the averaging of trees only leads to an improved performance if the trees are uncorrelated, as training a set of trees on the same training set would lead to strongly related trees or possibly the same trees many times. In contrast to the bagging method, boosting for trees instead grows the trees in an adaptive way to remove bias, and each tree is then no longer identically distributed.

A part of the data is reserved for later use in cross validation through hold-out validation, also known as the out-of-bag sample. Usually the split for training and validation sets are roughly $7:3$ respectively. To decrease the correlation between the trees, a method called random subspace method, or feature bagging, is used. Feature bagging, similar to regular bagging, is a method where the features are randomly sampled with replacement for each learner. At each candidate split in a tree during the learning process, a random subset of the features is selected. The reason for this procedure can be observed in the ordinary bootstrap example, where if one or a few features are very strong predictors for the target output, those features will be selected in many of the trees and thus causing them to become correlated.

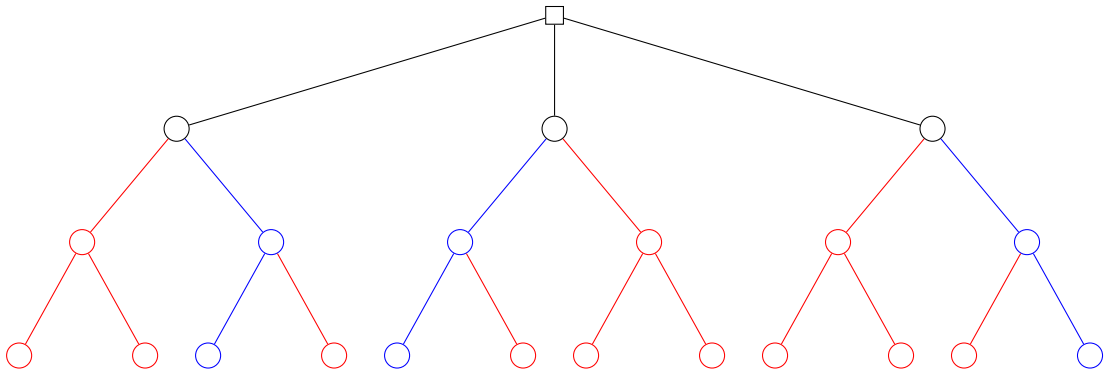

Figure 3.1: A simple example of a random forest consisting of three decision trees. Each tree is constructed with a randomly selected subset of data points and features. The chosen features for each branch is marked blue, and the final outcomes of each tree at the end of the forest will be used for classification or regression. For a regression problem, the values of the outcomes are summarized and averaged, while a majority voting is used for a classification problem.

The random forest algorithm has commonly been used as a baseline when it comes to machine learning algorithms and has seen applications in a wide variety of fields, e.g. finance and healthcare [14]. The advantages of the algorithm are its flexibility with the data format, as it can handle both regression and classification, categorical and continuous variables, automates handling of missing values in data, and it does not require normalisation of data due to using a rule-based approach. Additionally, random forests are less prone to overfitting because of the rule-based approach as well as being an ensemble of decision trees. The main drawback of the algorithm is its high computational requirement, both in terms of power and time, due to the combination of a large amount of decision trees used. However, the efficiency of the algorithm combined with the high resistance to overfitting has made this model desirable and popular in industries and research.

3.3.1 Mean decrease accuracy and Gini

The output of a random forest model can be evaluated in two ways to measure the importance of a feature: mean decrease accuracy, and mean decrease Gini or mean decrease impurity, also known as the Gini index [14]. Mean decrease accuracy of a feature is a measurement of how much accuracy the model loses by excluding it from the prediction. A feature that is highly correlated to the model would have a high value of mean decrease accuracy, as excluding it would impact the model prediction accuracy negatively. Gini importance, also known as Gini impurity, measures the gain in purity by splits of a given feature. The term purity in random forests refers to how correct a split separates classes, where maximum purity and conversely minimum impurity occurs when a split perfectly splits into two pure single class nodes. Features that tend to split nodes into pure single class nodes are more useful to the classification. However, compared to the mean decrease accuracy, the mean decrease in Gini only measures a local improvement in each split rather than the performance of the whole model and therefore less directly related to importance as well as more biased and unstable.

3.4 Support vector machines

For a binary classification problem with two classes, where the training dataset consists of input feature vectors $\boldsymbol{x}_i$ and corresponding class labels y_i, which can be written as $(\boldsymbol{x}_1, y_1) \ldots (\boldsymbol{x}_n, y_n)$, give an expression for the hyperplane as:

$$w^T \boldsymbol{x} - b = 0. \tag{3.13}$$

The vector w represents the normal vector to the hyperplane and b is the offset of the hyperplane from origin [15]. Figure 3.2 shows an example for when there are two features.

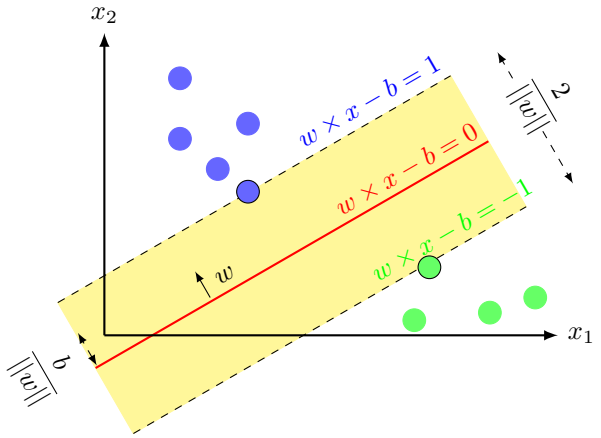

Figure 3.2: Example of a maximum-margin hyperplane for feature vector of size 2 and two classes for classification.

Hard Margin: Given a set of training data that is linearly separable, two additional hyperplanes can be selected that separate the two classes that maximize the separation distance. The region created by these two hyperplanes is the margin, and the maximum-margin hyperplane is the hyperplane the lies halfway between these two [15]. With a normalized or standardized dataset the hyperplane can be described by the equations:

$$w^T x + b = \begin{cases} 1 \text{ (data points on the boundary or above is labelled 1)} \\ -1 \text{ (data points on the boundary or below is labelled -1)} \end{cases} \tag{3.14}$$

To maximize the distance between the two hyperplanes, expressed as $\frac{2}{\|w\|}$, the Euclidian norm of the weight vector $\|w\|$ needs to be minimized. With the additional condition that the data points do not fall into the margin, the constraints can be expressed, for each i, as:

$$y_i = \begin{cases} 1 : w^T \boldsymbol{x}_i - b \geq 1 \\ -1 : w^T \boldsymbol{x}_i - b \leq -1 \end{cases} \tag{3.15}$$

Simply put, these conditions state that the data points must lie outside and on the correct side of the margin. Using this the optimization problem can be formulated as:

$$\min_{w,b} \frac{1}{2}\|w\|^2 \tag{3.16}$$

$$\text{subject to } y_i(w^T \boldsymbol{x}_i - b) \geq 1 \text{ for } i = 1, 2, 3, \ldots, n. \tag{3.17}$$

The w and b that solve this problem determines the classifier $\boldsymbol{x} \mapsto sgn(w^T \boldsymbol{x} - b)$.

Soft Margin: If the dataset is not linearly separable, the hinge-loss function is used, which is expressed as:

$$\max(0, 1 - y_1(w^T \boldsymbol{x}_i - b)), \tag{3.18}$$

which outputs 0 if the constraint is satisfied, and outputs the distance from the margin otherwise, indicating that the data point lies on the wrong side of the margin. With this the goal is to minimize:

$$\|w\| + C \left[\frac{1}{n} \sum_{i=1}^{n} \max(0, 1 - y_i(w^T \boldsymbol{x}_i - b)) \right], \tag{3.19}$$

where the parameter $C > 0$ determines the trade between increasing the size of the margin and ensuring that the data points $\boldsymbol{x}_i$ lies on the correct side of the margin [15]. The optimization problem can then be rewritten as:

$$\min_{w,b} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \zeta_i \tag{3.20}$$

$$\text{subject to } y_i(w^T \boldsymbol{x}_i - b) \geq 1 - \zeta_i \text{ and } \zeta_i \geq 0 \text{ for } i = 1, 2, 3, \ldots, n. \tag{3.21}$$

For large values of C, this will behave similar to the hard margin case with linearly separable points.

Kernel: Alternatively for non-linearly separable data, a so called kernel function can be applied to the hyperplane, allowing the algorithm to fit the maximum-margin hyperplane in a transformed feature space, thus altering non-separable problems in to separable ones. Simply put, the kernel performs complex data transformations and finds the process to separate the data based on labels and outputs [15].

In summary, support vector machines are supervised max-margin models with learning algorithms used to analyse data for classification and regression, although the algorithm is more suited towards classification. Due to its capability to manage high-dimensional data and non-linear relations, SVMs are adaptable and efficient for a variety of different applications. Additionally, as the model uses a subset of training points in the decision function, it ends up being very memory efficient. The objective of the SVM algorithm is to find the optimal hyperplane in an N-dimensional space that can separate the data-points of different classes in the feature space. In the event that the data points are not linearly separable, a kernel can be applied to the SVM in order to map the original data points into high-dimensional feature spaces in order to find the hyperplane. These kernels can be altered and customized to fit the situation to improve the performance of the model. The performance is however dependant on the choice of kernel in the event of a non-separable dataset, and the specific kernel that optimizes the model, while being important for the performance, is not necessarily easy to choose. Lastly, the SVM model only handles data of numerical representation, which means that categorical data need to be converted to a numerical one in a way that does not compromise the original meaning of the category. To this end a method called *one-hot encoding* can be used, which will be explained later as other models require similar data pre-processing.

3.5 Neural network models

Artificial neural networks are branch of machine learning models that are inspired by the neuronal systems found in biological neural networks in animal brains. A node in this system represents an artificial neuron which models a biological neuron, connected with edges modelling the synapses. Typically, these artificial neurons are divided into layers and perform different transmutations depending on their inputs. Each neuron receives signals from neurons of the previous layers, processes the signals, and then sends the results of this process as a signal to the next layer of neurons. The signals passed over edges are real numbers, and the computation process at each neuron is calculated using a function of the sum of inputs, called the activation function. Throughout the learning process, each edge and neuron have a weight and bias respectively associated to them that are adjusted [16].

3.5.1 Feed forward neural network

In a feed forward neural network model, the signal from the input is forwarded through layers of neuron, see figure 3.3. The number of hidden layers between the input and output layers can be adjusted, and adding more layers allows for dependencies between features associated with the neurons. The connection between two layers can be represented using a weight matrix $W_m n$, between layers of size m to size n. Using the example shown in figure 3.3, the calculation of each neuron follows as:

$$h_j = F(x_i, W_{ij}, b_j) \tag{3.22}$$

$$y_k = G(h_j, W_{jk}, b_k) \tag{3.23}$$

where the term b is a bias term associated with each neuron. The functions F and G are typically referred to as activation functions, which can be either linear or non-linear. A linear activation function could be written as:

$$h_j = F(x_i, W_{ij}, b_j) = \sum_i x_i W_{ij} + b_j. \tag{3.24}$$

The most commonly used non-linear activation function is the rectified linear unit, ReLU, defined as:

$$g(z) = \max(0, z), \tag{3.25}$$

where z is the linear transformation of the signal going into the layer from the previous layer, $z = \boldsymbol{W}^T \boldsymbol{x} + b$.

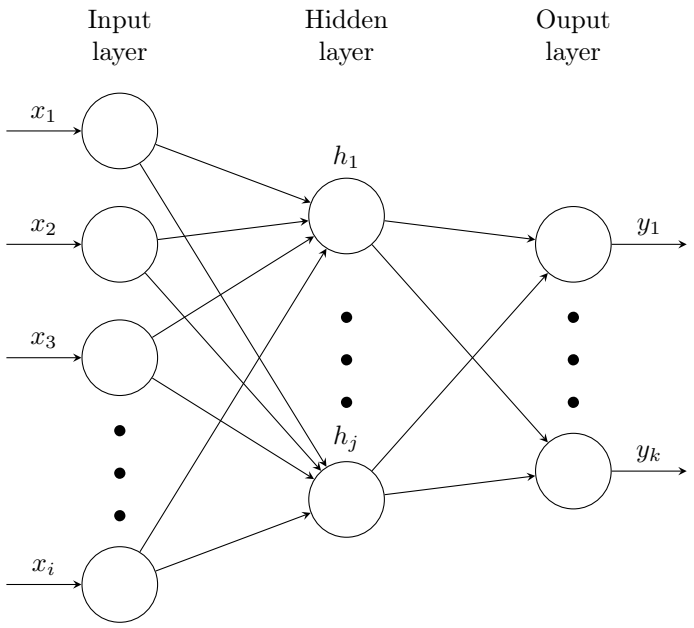

Figure 3.3: Figure of neural network consisting of an input layer x with i nodes, hidden layer h with j nodes and output layer y with k nodes. Each neuron in each layer is connected to every neuron in the next layer, and the signal passed through every edge is adjusted based on weights W_{ij} or W_{jk} and bias b_j and b_k for each node.

Through the forward propagation over the network, the signal from the input eventually reaches the output. Here, the output needs to be evaluated on its performance based on a loss function, where the model is optimized by minimizing this function. Depending on the learning paradigm, of which there are three primary methods, different actions are taken in regard to the accuracy of the output. For supervised learning, the inputs are paired with a desired output of the network for training. For unsupervised learning, the error is calculated with a cost function, a function of the data and the output of the network [16]. There are more methods for learning, such as reinforcement learning and self-learning, that will not be further elaborated upon as they are not relevant for this project.

A common choice for a loss function is the means square error loss function, which by definition can be expressed as:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (t_i - O_i)^2, \tag{3.26}$$

where t_i is the target vector, the vector of observed values of the variable being predicted, and O_i are the output of predicted values from the model. An alternative is the cross-entropy loss function, equivalent to the likelihood maximisation:

$$L = -\sum_{j=1}^{m} \sum_{k=1}^{o} y_k log f_k(x_i). \tag{3.27}$$

The summary is over the hidden layer leading up to the output layer, along with the output layer itself. Here, y_k are the output neuron values, and $f_k(x_i)$ are the class probabilities.

3.5.2 Backpropagation

In order for the network to learn from the error, a gradient estimation method called backpropagation is used to update the parameters. This method performs the update by using the output error and adjusting the weights from the output layer and towards the input layer, therefore going backwards is opposed to the previous feed forwarding of the input signal. Gradient descent learning allows for an efficient training process, and the backpropagation trains the multilayered network such that it can learn the appropriate internal representations and allow it to learn any arbitrary mapping of input and output. Note that as the gradient is used in this calculation, the activation function needs to be differentiable in the activation region. ReLU is non-differentiable in one point, so historically the logistic function is more commonly used [16].

The weight update can be described as:

$$w' = w + \delta w \text{ with weight increment } \delta w = -\eta \frac{\partial H}{\partial w}, \tag{3.28}$$

where η is the learning rate of the model and H is the energy function defined as:

$$H = \frac{1}{2} \sum_i (t_i - O_i)^2. \tag{3.29}$$

By optimizing the parameters to make the output approach the target, the overall energy of the network reduces until it reaches a minimum, which would correspond to the optimum of the network. The energy function H holds similarities to the loss function, essentially representing the error. Referring back to the example network from figure 3.3, the weight and bias increments would follow:

$$\delta w_{jk} = -\eta \frac{\partial H}{\partial w_{jk}} \tag{3.30}$$

$$\delta b_k = -\eta \frac{\partial H}{\partial w_k} \tag{3.31}$$

$$\delta w_{ij} = -\eta \frac{\partial H}{\partial w_{ij}} \tag{3.32}$$

$$\delta b_j = -\eta \frac{\partial H}{\partial b_j} \tag{3.33}$$

These increments are summarized with the current values of weight and bias to update its values before the next data point is used to for training [16].

3.5.3 Regularization

Similar to the logistic regression, regularization can be applied to neural network models in order to reduce overfitting. Neural networks are especially prone to overfitting due to the large amount of parameters, therefore a regularized alternative to the loss function can be utilized instead:

$$L_\lambda = L(\theta) + \lambda R(\theta), \tag{3.34}$$

where λ is the regularization parameter, and θ are the parameters of the neural network model. The regularization function $R(\theta)$ is similar to the regularization function described in equation 3.10, and can for example take the form:

$$R(\theta) = \sum_{i=1}^{|\theta|} \theta_i^2. \tag{3.35}$$

The parameter λ is the tuning parameter of the regularization, where a higher value results in correspondingly increased regularization. This parameter is highly important for the performance of the model - chosen too low and the regularization is insufficient and overfitting may occur, while a choice too high leads to underfitting.

3.5.4 Categorical data handling for neural networks

Similar to the SVM method and partially the logistic regression, neural networks are not capable of working with categorical data. Simply changing the categories to numerical values and working with intervals of values does not yield meaningful results during prediction as the learning process using numerical adjustments do not necessarily match the implication of the original meaning of the categorical data. The next section describes a solution to handling non-numerical data types for models that require numerical inputs, called one-hot encoding.

3.6 Data processing

For SVM and neural network models, the data would in some way need to be entirely numerically represented, while for logistic regression it is beneficial but not required. The data pre-processing would therefore need to be cleared for missing values and have non-numerical data converted to numerical data. Random forest can manage data types other than numerical and only needs the pre-processing to handle missing data.

3.6.1 One-hot encoding

The most common method for dealing with non-numerical data is called *one-hot encoding*, which can convert categorical data such as the colors blue, red and green into a binary representation: red is represented by the vector $\{1, 0, 0\}$, green by $\{0, 1, 0\}$, blue by $\{0, 0, 1\}$. This then converts the one column of category *Color* into three columns of each color, *Color.red*, *Color.green* and *Color.blue*. The categorical example can be seen in the table 3.1, and the post encoding results for the example can be seen in the follow-up table 3.2.

	Color
1	Red
2	Red
3	Blue
4	Green

Table 3.1: Categorical values of colors represented in a table. The four data points each have a color, totalling 3 different colors.

	Color.red	Color.green	Color.blue
1	1	0	0
2	1	0	0
3	0	0	1
4	0	1	0

Table 3.2: One-hot encoding of the color table 3.1. Each of the categories are separated into their own column, and whether the data point is of the category is determined by the binary representation of each column.

An alternative can be applied if the categorical data is ordinal, meaning that the categories come in an order, for example customer satisfaction: dissatisfied, neutral and satisfied. Here it is applicable to have these opinions on the same spectrum, for example on a scale of 1 to 5, where 1 represents dissatisfied and 5 for satisfied. More or less of this category of values has real meaning, compared to the nominal data described earlier with the one-hot encoding, and can therefore be converted to a numerical scale instead.

A problem that arises from this approach to encode the data from a categorical dataset to a numerical one is that every category of a feature now becomes an individual feature, meaning that a feature of 5 categories becomes 5 features. In large datasets where the number of data points can exceed hundreds of thousands or millions and where the number of categories can reach multiples of 10 and higher, the size of the dataset eventually becomes far too large. To handle this situation, being selective with which features to include can

be useful. Out of the hundreds of features, it is unlikely that a majority of these will be relevant to a high degree for the analysis. Additionally, features including a large amount of categories are unlikely to have much correlation with the outcome, especially if the classification is done with a low quantity of classes.

3.7 Feature importance details using Shapley

Random forest on its own is only capable of informing whether the feature in question is correlated or not to the classification, but how this feature relates to the classification is still unknown. Random forest essentially uses the rule based algorithm to determine a ranking of features that appear with the different classifications. For example, if age is shown to be an important feature, it is not specified if it is low or high age that are more likely candidates for covid cases. Similarly, other machine learning algorithms might not output a result that can be related to the importance of features or give an output that is hard to interpret in terms of importance. In order to more thoroughly understand the significance of features for a classification of the models, analysis using *Shapley values* can be conducted [6].

The SHAP method, Shapley Additive Explanations, uses a game theoretic approach to measure the contribution of each player, or feature, to the final outcome, also known as the payout of the game. A fair distribution of the payout gives an indication of the impact of each player by their share of the payout. Unlike the importance given by the machine learning models, the importance value of the SHAP can be positive or negative, indicating a positive or negative impact on the prediction.

3.7.1 Shapley values

Since many terms related to value will be used: feature value refers to the features x, Shapley values refers to contribution of a feature x to the prediction $\phi(val)$, and the value function $val(S)$ refers to the individual contributions of the coalition of features S. The definition of the Shapley value is the value function val of players in S. The Shapley value of a feature value is the contribution to the payout, weighted and summed over possible combinations of features values:

$$\phi_j(val) = \sum_{S \subseteq \{1,\dots,p\} \setminus j} \frac{|S|!(p-|S|-1)!}{p!}(val(S \cup j) - val(S)) \tag{3.36}$$

where S is the subset of features used for the model and p is the number of features. For a vector x of feature values of the instance to be explained and p, $val_x(S)$ is the prediction for feature values in S that are marginalized over features not included in S:

$$val_x(S) = \int \hat{f}(x_1, \dots, x_p) d\mathbb{P}_{x \notin S} - E_X(\hat{f}(X)) \tag{3.37}$$

where E is the mean effect estimate, $\hat{f}$ is the prediction model. The integration is performed over each feature outside of the subset S. To better show this, a concrete example can be used: given a model with 5 features, $\{x_1, x_2, x_3, x_4, x_5\}$, the evaluation of a prediction for a coalition S consisting of features x_2 and x_3:

$$val_x(S) = val_x(\{x_2, x_3\}) = \int_{\mathbb{R}} \int_{\mathbb{R}} \int_{\mathbb{R}} \hat{f}(X_1, x_2, x_3, X_4, X_5) d\mathbb{P}_{X_1, X_4, X_5} - E_X(\hat{f}(X)). \tag{3.38}$$

This can be compared to the linear model prediction for an instance of data x:

$$\hat{f}(x) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p, \tag{3.39}$$

which gives the contribution ϕ_j of the j-th feature on the prediction $\hat{f}(x)$:

$$\phi_j(\hat{f}) = \beta_j x_j - \beta_j E(X_j). \tag{3.40}$$

Here, x is the instance that the contributions are computed for, x_j is the feature value, $j = \{1, \ldots, p\}$, β_j is the weight of the corresponding feature j, and $E(\beta_j x_j)$ mean effect estimate for feature j. Summing the feature contributions results in the following:

$$\sum_{j=1}^{p} \phi_j(\hat{f}) = \sum_{j=1}^{p}(\beta_j x_j - \beta_j E(X_j)) = (\beta_0 + \sum_{j=1}^{p} \beta_j x_j) - (\beta_0 + \sum_{j=1}^{p} E(\beta_j x_j)) = \hat{f}(x) + E(\hat{f}(X)). \tag{3.41}$$

This is the predicted value of the data point x minus the average predicted value, which is similar to equation 3.37, where $val_x(S)$ is the prediction for feature values in coalition S.

The Shapley value satisfies the properties *efficiency*, *symmetry*, *dummy* and *additivity*, which combined together can be considered a definition of a fair payout.

Efficiency: The feature contributions need to add up to the difference of prediction for x and the average:

$$\sum_{j=1}^{p} \phi_j = \hat{f}(x) E_X(\hat{f}(X)). \tag{3.42}$$

Symmetry: Contributions from two feature values j and k should be the same if they contribute equally to all possible coalitions. Therefore, if for all $S \subseteq \{1, \ldots, p\} \setminus \{j, k\}$:

$$val(S \cup \{j\}) = val(S \cup \{k\}), \tag{3.43}$$

$$\Rightarrow \phi_j = \phi_k. \tag{3.44}$$

Dummy: A feature j that does not change the predicted value regardless of which coalition of feature values it is added to, should have Shapley value of 0. Therefore, if for all $S \subseteq \{1, \ldots, p\}$:

$$val(S \cup \{j\}) = val(S), \tag{3.45}$$

$$\Rightarrow \phi_j = 0. \tag{3.46}$$

Additivity: For a game with combined payouts $val + val^+$, the respective Shapley values follows:

$$\phi_j + \phi_j^+. \tag{3.47}$$

For example, for a random forest model, the prediction is an average of many decision trees. The property of additivity means that for a feature value, the Shapley value can be calculated for each individual tree and then average them to get the total Shapley value of the whole random forest.

3.7.2 Estimation of Shapley values

All possible coalitions of feature values are evaluated with and without the j-th feature to calculate the exact Shapley value. However, for a larger quantity of features, conducting the calculation to exactly solve the problem increases the possible coalitions exponentially. It is proposed by Strumbelj and Kononenko to approximate with Monte-Carlo Sampling [17]:

$$\hat{\phi_j} = \frac{1}{M} \sum_{m=1}^{M} (\hat{f}(x_{+j}^m) - \hat{f}(x_{-j}^m)), \tag{3.48}$$

where $\hat{f}(x_{+j}^m)$ is the prediction for x, but with a random number of feature values replaced by random values from random data points z, except for the respective value of feature j. The vector x_{-j}^m is similar to x_{+j}^m except for having values x_j^m also being taken from the sampled z.

For Shapley value estimation of a single feature value, the j-th feature of the set, requires M iterations, instance of interest x, feature index j, data matrix X and machine learning model f. For all $m = 1, \ldots, M$:

- Draw random instance z from data X

- Choose random permutation o of the feature values
- Order x: $x_1, \ldots, x_j, \ldots, x_p$
- Order z: $z_1, \ldots, z_j, \ldots, z_p$
- Construct new instance with j: $(x_{+j} = x_1, \ldots, x_{j-1}, x_j, z_{j+1}, \ldots, z_p)$
- Construct new instance without j: $(x_{+j} = x_1, \ldots, x_{j-1}, z_j, z_{j+1}, \ldots, z_p)$
- Compute marginal contribution $\phi_j^m = \hat{f}(x_{+j}) - \hat{f}(x_{-j})$
- Compute Shapley value as the average $\phi_j(x) = \frac{1}{M} \sum_{m=1}^{M} \phi_j^m$

The averaging implicitly weighs the samples with a probability distribution of the data matrix X. This procedure can then be repeated for each of the feature values to obtain Shapley values for all the features.

3.7.3 Shapley Additive Explanations

Shapley Additive Explanations, SHAP, is a method used to explain individual predictions of an instance x based on theoretical optimal Shapley values by computing the contribution of each feature in the prediction, as described by Lundberg and Lee [18]. In SHAP, the Shapley value explanations are represented as an additive feature attribution method or a linear model, specified as:

$$g(z') = \phi_0 + \sum_{j=1}^{M} \phi_j z'_j, \tag{3.49}$$

where g is the explanation model, $z' \in \{0,1\}^M$ is the coalition vector and $\phi_j \in \mathbb{R}$ is the feature attribution for feature j, where M is the amount of features. The coalition vector indicates the presence of the feature value, with 1 and 0 representing presence and absence respectively. The computation of Shapley values utilizes a similar idea, where the presence and absence of certain feature values are simulated. For an instance of interest x, the coalition vector has all elements of 1, and the equation 3.49 can be simplified to:

$$g(x') = \phi_0 + \sum_{j=1}^{M} \phi_j. \tag{3.50}$$

Shapley values are able to satisfy the properties of efficiency, symmetry, dummy and additivity, but SHAP instead describes three other desirable properties: *local accuracy*, *missingness* and *consistency* [18]. Let f be the original prediction model to be explained, and g the explanation model. Explanation models often use simplified inputs x', that map the original inputs through a mapping function $x = h_x(x')$.

Local accuracy: Similar to the Shapley value efficiency property, but using the coalition vector instead gives:

$$\hat{f}(x) = g(x') = \phi_0 + \Sigma_{j=1}^{M} \phi_j x'_j. \tag{3.51}$$

By defining $\phi_0 = E_X(\hat{f}(x))$ and setting the coalition vector x'_j to all 1's, this expression changes to the Shapley value efficiency property.

Missingness: Missingness indicates that a missing feature gets an attribution of zero:

$$x'_j = 0 \Rightarrow \phi_j = 0. \tag{3.52}$$

The coalitions x'_j are indicators of the presence of feature values, indicated by 1, and therefore the absence by 0. This property is described as a "minor book-keeping property" by Lundberg [18] that in theory could have an arbitrary Shapley value without changing the local accuracy property, since it multiplies with $x'_j = 0$.

Consistency: The consistency property states that if a model changes in a way that the marginal contribution of a feature value increases or remains unchanged without regard for other features, the Shapley value also increases or remains unchanged. Let $\hat{f}_x(z') = \hat{f}(h_x(z'))$ and z'_{-j} indicate $z'_j = 0$. For two models f and f' that satisfy:

$$\hat{f}'_x(z') - \hat{f}'_x(z'_{-j}) \geq \hat{f}_x(z') - \hat{f}_x(z'_{-j}) \tag{3.53}$$

for all inputs of $z \in \{0,1\}^M$, then:

$$\phi_j(\hat{f}', x) \geq \phi_j(\hat{f}, x). \tag{3.54}$$

From the consistency property, the Shapley value properties linearity, dummy and symmetry follow.

3.7.3.1 KernelSHAP

KernelSHAP uses an alternative to directly calculating the Shapley values using a kernel-based approach that estimates for an instance x the contributions of each feature value to the prediction, consisting of the following steps.

- Sample coalitions $z'_k \in \{0,1\}^M$, $k \in \{1,\ldots,K\}$, where 1 and 0 represents feature presence and absence respectively.
- Get prediction for each coalition z'_k by converting it to the original feature space $h_x(z'_k)$, and then applying the model $\hat{f}(h_x(z'_k))$.
- Compute the weight for each coalition z'_k using the SHAP kernel.
- Fit a weighted linear model.
- Return Shapley values ϕ_k and the coefficients from the linear model.

The sampled coalitions essentially consist of vectors of 1's and 0's in a random order, repeated K times to form the data set for the regression model, where the target for the regression model is the prediction for a coalition. The model however has not been trained on this sort of binary coalition and thus need the coalitions to be converted to the feature values. This function $h_x(z') = z$ maps 1's of the coalition to the corresponding value from instance x that needs to be explained. For tabular data, a coalition of 0 maps to another instance from the sampled data, meaning that absent data is represented by the feature value of a random feature value data.

Instance x on the left, and the corresponding values after applying function h_x on the right.

Age	Gender	Color
1	1	1

Age	Gender	Color
20	Male	Green

Instance x on the left with absent features, and the corresponding values after applying function h_x on the right, with absent features replaced by feature values from randomly sampled data instance.

Age	Gender	Color
0	1	0

Age	Gender	Color
26	Male	Blue

For tabular data, the function h_x treats the feature X_j and other features X_{-j} as independent and integrates over the marginal distribution:

$$\hat{f}(h_x(z')) = E_{X_{-j}}[\hat{f}(x)]. \tag{3.55}$$

However, sampling from the marginal distribution means that the dependence structure between features that are present and absent are ignored. This means that KernelSHAP suffers from the same problem as other permutation-based interpretation methods, as the estimation ends up putting too much weight on unlikely instances, resulting in unreliable outputs. The solution to this would be to use the conditional distribution instead, which would alter the value function and thus the game to which Shapley values are the solution

to. The interpretation of Shapley values for an absent feature value in a conditional game could still have a non-zero Shapley value, despite not being used by the model at all.

Another important concept in SHAP is weighting of the sampled instances. Small and large coalitions, corresponding to few and many 1's respectively, result in the largest weights, since it is easier to learn about individual features if they are studied in isolation. To achieve Shapley compliant weighting, the proposed expression of the SHAP kernel follows

$$\pi_x(z') = \frac{(M-1)}{\binom{M}{|z'|}|z'|(M-|z'|)}. \tag{3.56}$$

Here, M is the maximum coalition size, and $|z'|$ is the number of features present in an instance z' [18]. Using data, targets and weights, a weighted linear regression model can be constructed for the explanation model g:

$$g(z') = \phi_0 + \sum_{j=1}^{M} \phi_j z'_j. \tag{3.57}$$

The coefficients of the model, ϕ_j, are the Shapley values. This can then be further optimized by minimizing the loss function L:

$$L(\hat{f}, g, \pi_x) = \sum_{z' \in Z} [\hat{f}(h_x(z')) - g(z')]^2 \pi_x(z'), \tag{3.58}$$

where Z is the training data, and the loss function follows a sum of squared errors [6].

3.7.3.2 TreeSHAP

Another variant of SHAP for tree-based machine learning models is the TreeSHAP, used for models such as decision trees, random forests and gradient boosted trees, that functions as a faster but model specific alternative to KernelSHAP. Compared to exact KernelSHAP that has a computational complexity of $O(TL2^M)$, TreeSHAP instead has $O(TLD^2)$, where T is the number of trees, L is the maximum number of leaves in a tree, M is the maximum coalition size, and D is the maximum depth of a tree [6]. This method however ends up producing feature attributions that are unintuitive, but due to KernelSHAP in general being slow computationally, the faster implementation of TreeSHAP can be desirable.

4 Results

4.1 Random forest

The random forest algorithm produces a list of the calculated features along with their importance, where a higher score indicated a higher level of importance to the outcome. The plot shows the top ranking features along with their *mean decrease accuracy* and *mean decrease Gini*. Initially, running the model on the entire dataset resulted in many redundant variables showing high importance, such as other covid related data and some very similar or duplicate parameters left over from previous analyses. These were manually removed to allow more relevant variables to show.

A randomized subset of 2 and 3 million data points of the data was selected for the two runs of the random forest model. In general, the split of data for training and testing was set to $0.7, 0.3$. The number of trees used is recommended to be roughly between 60 and 130, so the number of trees were set to 128. The number of trees affect the .

4.1.1 Exclusion of excess features

The original dataset includes multiple features of covid indication, and as the classification is done on the feature Any_Covid, all other features of covid cases are removed. Initially, running the model fit on the entire data set resulted in many clearly redundant features showing high importance: date of positive testing, cases of ICU, cases of death from covid, and many more are similar to the feature any covid. Specifically for the random forest algorithm, inclusion of excess features interferes with the rule-based method of the model, meaning that more relevant features end up inhibited by a lack of pre-processing of the data. Additionally, since this analysis does not aim to consider the temporal aspect of the pandemic, the monthly cases of testing, ICU and deaths are removed. Lastly, all redundant features, for example the anonymous serial number, old DeSO codes and duplicate features are also removed.

4.1.2 Variable importance and error plots

Note that there are two results shown: one run where all the unrelated features have been removed, shown in figures 4.1, 4.2 and 4.5, as well as an earlier version of the run, shown in 4.3, 4.4 and 4.6, which still included the temporal data. The month-to-month cases themselves are not particularly interesting for this current analysis, but the Shapley value evaluation, shown in figures 4.5 and 4.6, showed more information on feature contributions to the output. Due to this inclusion, the other results from this earlier run will be included to give context for its SHAP plot.

The top 45 features ranked in importance based on mean decrease accuracy and mean decrease Gini are shown in the plots in figures 4.1 and 4.3, where mean decrease accuracy is in general more related to the model performance. The features ranked high in each of these measurements are mostly unique from each other. Mean decrease accuracy consists mostly of socio-demographic statistical area data variables such as age proportions, low or high education in regard to health work or not, household condition (living alone or together, with or without children). Mean decrease Gini seems to rank socio-economic data higher, such as area of living environment, income per habitant, disposable income, work income and various job related

statistics. Additionally, the variables XKOORDsw and YKOORDsw, which are the geographical coordinates of the DeSO, show an importance to covid cases through Gini.

Only the top 45 features are included in the importance plots shown in figures 4.1 and 4.3, as the total amount of features included in the random forest algorithm were over 200. The lower importance features are evidently not important to the model fit and are not showcased in this report.

An additional shown graph is the error plot for the random forest, figures 4.2 and 4.4, showing how the prediction error change over the course of the trees being calculated. The confusion matrix for both runs using the test split of the data is also provided to show the complete model predictive accuracy, showing an accuracy of 77.592% and 75.984% for the primary run and the earlier run respectively. The confusion matrices are displayed in the tables 4.1 and 4.2

y_pred_test	Non Covid population	Any covid
Non Covid population	848885	11299
Any covid	0	39125

Table 4.1: Confusion matrix for the random forest model using the allocated test split of the data subset.

y_pred_test	Non Covid population	Any covid
Non Covid population	566206	8135
Any covid	0	25739

Table 4.2: Confusion matrix for the earlier run of the random forest model using the allocated test split of the data subset.

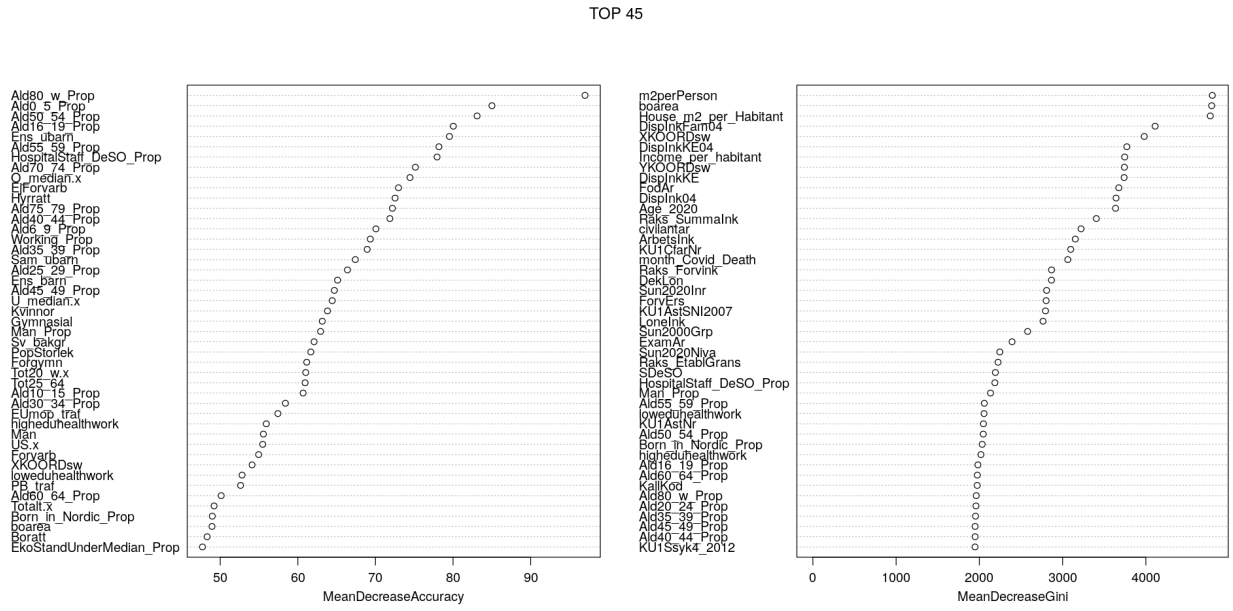

Figure 4.1: The top ranking 45 features in order for the primary result of the random forest model. The left graph is measured with mean decrease accuracy, and the right with mean decrease Gini.

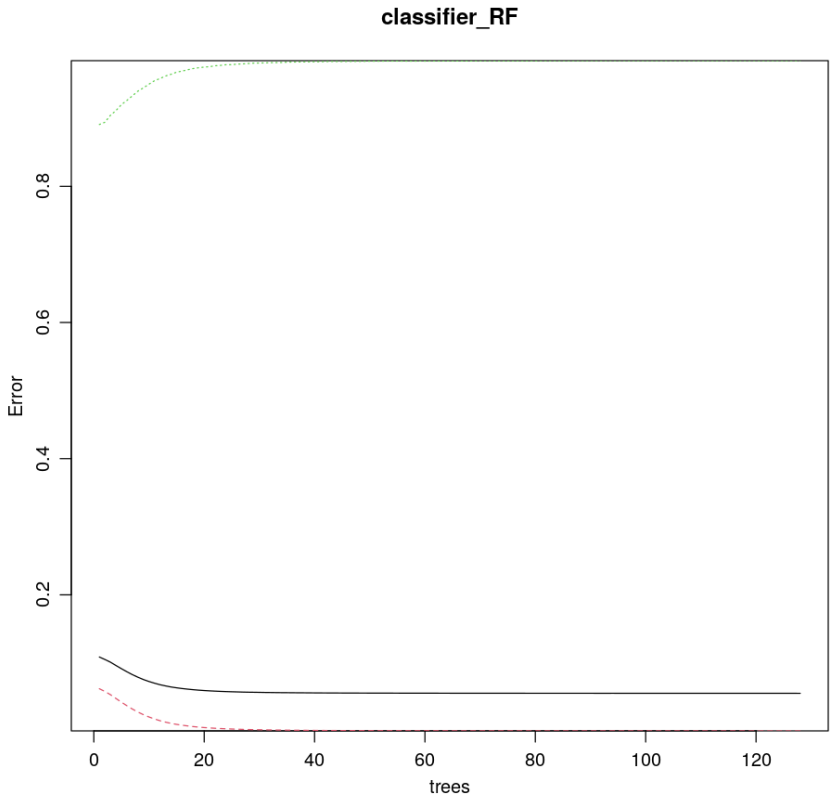

Figure 4.2: The error plot over the number of trees for the primary result of the random forest model. The two coloured dashed lines indicate the classifications of each class, currently showing two since the classification is binary. The black solid line is the out-of-bag error.

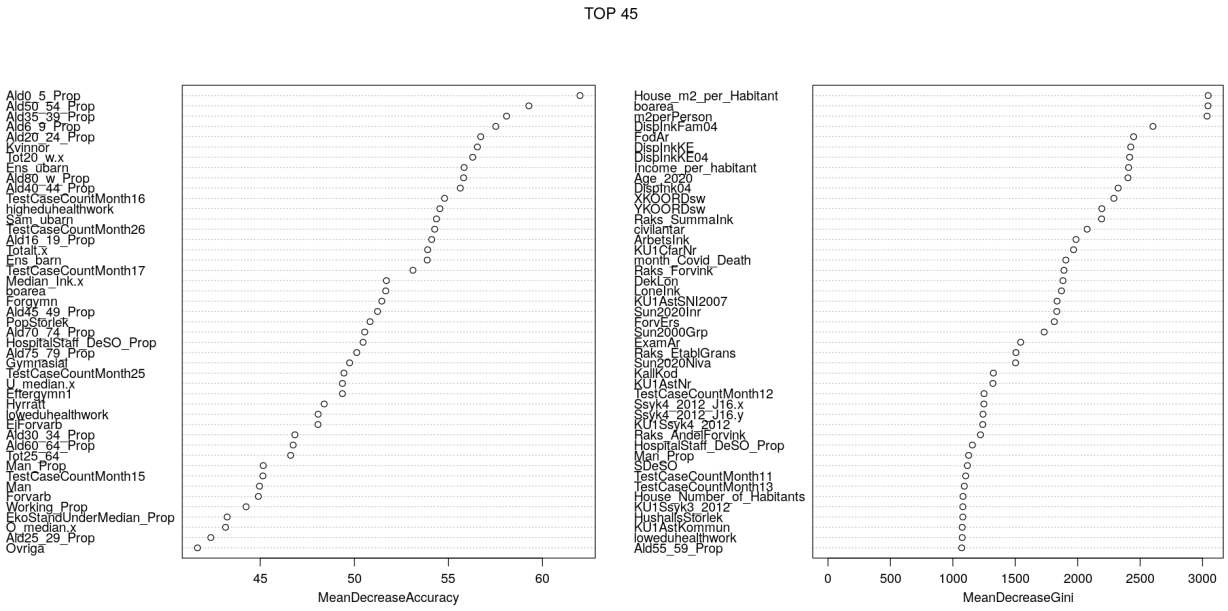

Figure 4.3: The top ranking 45 features in order for the earlier result of the random forest model. The left graph is measured with mean decrease accuracy, and the right with mean decrease Gini.

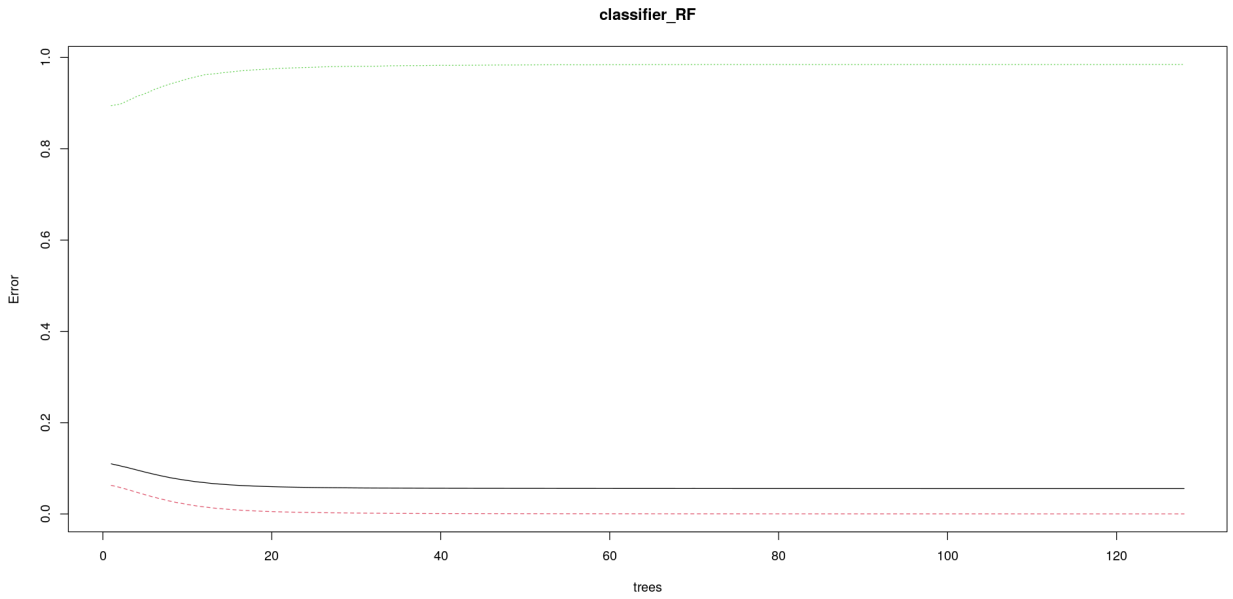

Figure 4.4: The error plot over the number of trees for the earlier result of the random forest model. The two coloured dashed lines indicate the classifications of each class, currently showing two since the classification is binary. The black solid line is the out-of-bag error.

4.1.3 Feature significance with SHAP

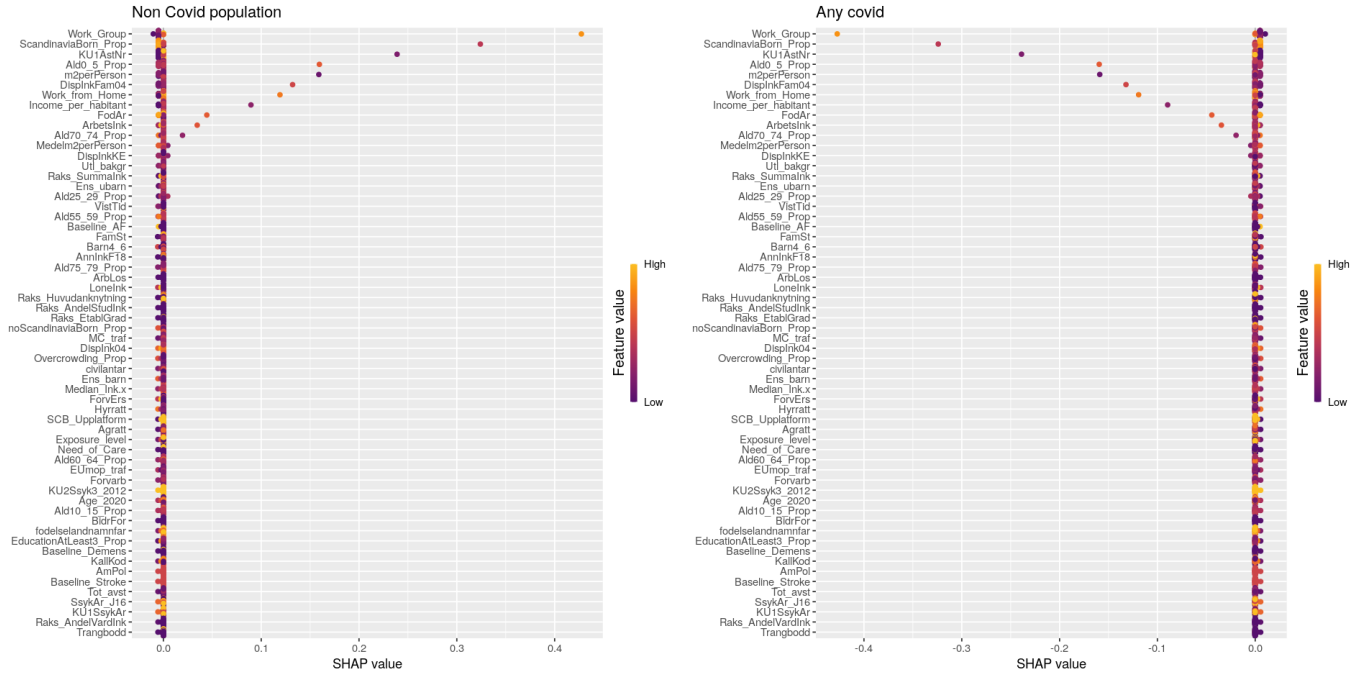

Figure 4.5: SHAP calculation of the random forest fit. The two graphs are mirrored due to the binary classing. Colour indicates the importance of the feature, while the SHAP value shows how positively or negatively the feature interacts with the outcome.

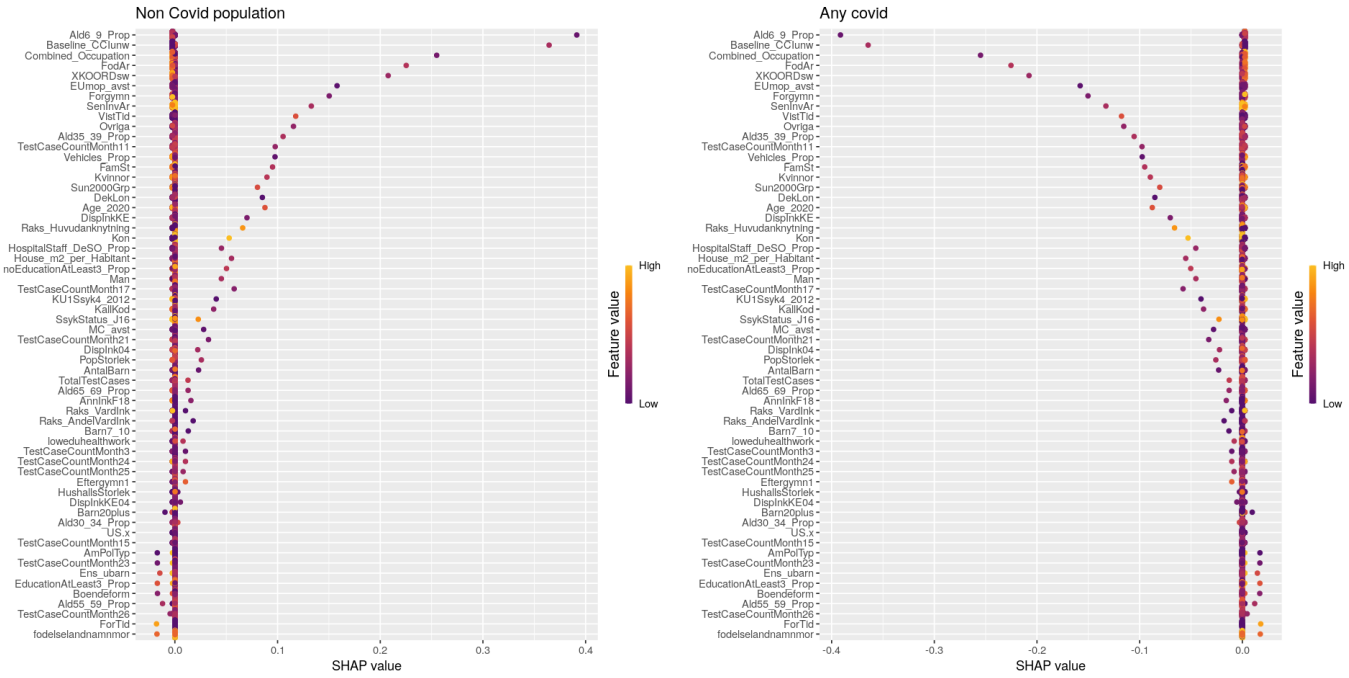

Figure 4.6: SHAP calculation of an earlier random forest fit before some unrelated features were removed from the data set before analysis. Disregard the presence of monthly cases as they are deemed irrelevant to the analysis as a whole. The two graphs are mirrored due to the binary classing. Colour indicates the importance of the feature, while the SHAP value shows how positively or negatively the feature interacts with the outcome.

4.2 Feature selection for inclusion in numerical methods

Using the results from random forest, an overview of which features are of interest can be formed and used to determine which features are worth investigating in the other methods. As a portion of the dataset is non-numerical, being either in the format of *date*, *character* or *factor*, a complete conversion using one-hot encoding was necessary. This method would create far too many columns (parameters) for the data set, as each category would need its own column.

4.2.1 Features included

The list of features included in the models excluding random forest is shown in table 4.3. Note that some of the features are aggregations of other features and as such contain similar or the same information. For example, age proportions are similar to age quantities, but the proportion is meaningful in the context of covid and exposure, as 50 individuals of a certain age means something different in a population of 100 compared to 1000. Similarly, Overcrowding_prop uses features such as living area and family conditions for its calculations, and those features are therefore indirectly included. A notable point is the combination of DeSO information and personal information of the individual being used together in the model with the intent of comparing the impact of the environment and the individuals themselves. The 6 features listed at the end of the table 4.3 are features related to personal information.

Variable name	Variable description
Ald0_5_prop	Proportion of individuals aged 0 to 5
Ald6_9_prop	Proportion of individuals aged 6 to 9
Ald10_14_prop	Proportion of individuals aged 10 to 15
Ald15_19_prop	Proportion of individuals aged 16 to 19
Ald20_24_prop	Proportion of individuals aged 20 to 24
Ald25_29_prop	Proportion of individuals aged 25 to 29
Ald30_34_prop	Proportion of individuals aged 30 to 34
Ald35_39_prop	Proportion of individuals aged 35 to 39
Ald40_44_prop	Proportion of individuals aged 40 to 44
Ald45_49_prop	Proportion of individuals aged 45 to 49
Ald50_54_prop	Proportion of individuals aged 50 to 54
Ald55_59_prop	Proportion of individuals aged 55 to 59
Ald60_64_prop	Proportion of individuals aged 60 to 64
Ald65_69_prop	Proportion of individuals aged 65 to 69
Ald70_74_prop	Proportion of individuals aged 70 to 74
Ald75_79_prop	Proportion of individuals aged 75 to 79
Ald80_w_prop	Proportion of individuals aged 80 and higher
Overcrowding_prop	Proportion of individuals
Man_prop	Proportion on men (essentially the same as proportion of women)
noScandinaviaBorn_prop	Proportion of individuals not born in Scandinavia
noEducationAtLeast3_prop	Proportion of individuals without an education of at least 3 years at university level
Working_prop	Proportion of individuals working
EkoStandUnderMedian_prop	Proportion of economical standing under the median
Vehicles_prop	Proportion of individuals owning vehicles
loweduhealthwork	Proportion of individuals in low education health work
higheduhealthwork	Proportion of individuals in high education health work
Medelm2perPerson	Average of living area m^2 per person
XKOORDsw	x-coordinate of the DeSO
YKOORDsw	y-coordinate of the DeSO
Age_2020	Age of individuals as of 2020
Habitants_per_house	The number of habitants in the household of the individual
Income_per_habitant	Income-range category of the individual
Gender	Gender of the individual
Education_3	Individual with at least 3 years of education at university level
Trangbodd	Overcrowding of an individual

Table 4.3: Table of all features included included in the remaining models.

4.3 Generalized linear model

The fit for logistic regression utilized the full set of data, and the k-fold cross validation used for hyperparameter optimization, for $k = 10$, resulted in $\alpha = 1$, $\lambda_{min} = 0.000433565160551969$ and $\lambda_{1se} = 0.00009785611711204$. The errors in the cross validation process were calculated using mean absolute error.

The graphs below showcase a plot of coefficients and errors based on the value of lambda, as well as how the coefficients relate to the deviance, as shown in figures 4.7 and 4.8 respectively. The figure 4.7 explains the change in coefficients as the value of lambda, and thus the penalization, changes. When the model has a high penalty, the coefficients of each feature begin to drop to 0. The amount of features that remain relevant for certain increments of lambda can be seen at the top of all the figures. The error plot shows the mean absolute error as it changes over values of lambda, showing two dashed lines that represent λ_{min} and λ_{1se}, the minimum lambda where the error is minimized, and the lambda within one standard error

(essentially the error bar in the error graph) of from the minimum. Lastly, in figure 4.8, the deviance shows the goodness-of-fit statistic and how the coefficients change, describing how well the model fits to a set of

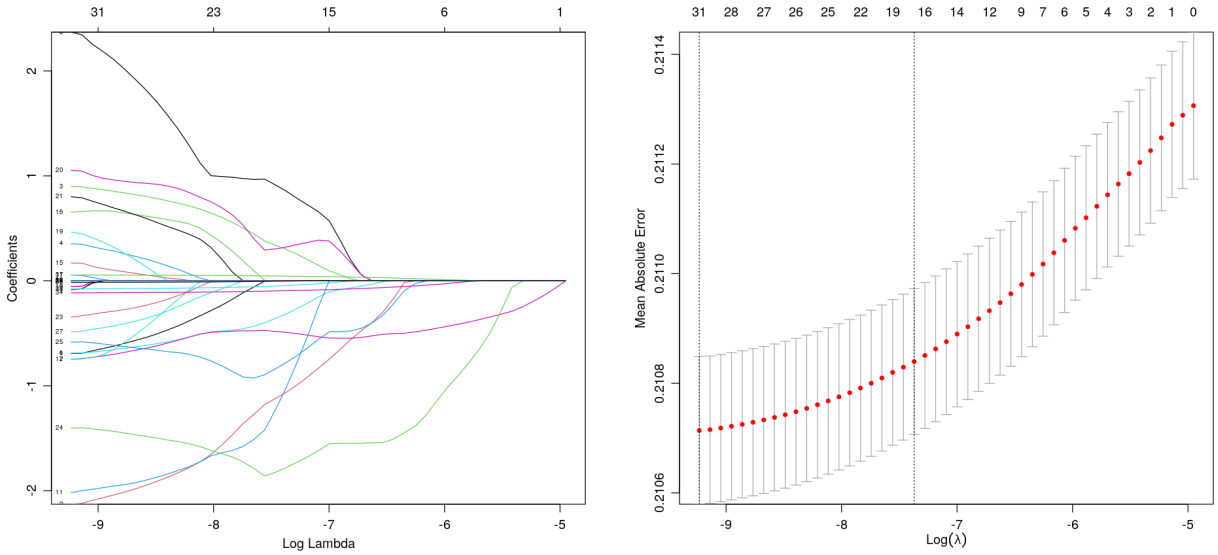

Figure 4.7: The left graph shows the variation in the coefficients of each covariate (feature) over lambda. The increasing value of lambda shows higher penalization, meaning that less important features are filtered out. The right graph shows the error plot along with the error bars over values of lambda. The two dashed lines show the minimum lambda and the lambda within 1 standard deviation from the minimum, λ_{min} and λ_{1se} respectively. On the top of the both graphs, numbers indicating how many features are still relevant are shown, decreasing as the penalization from lambda increases.

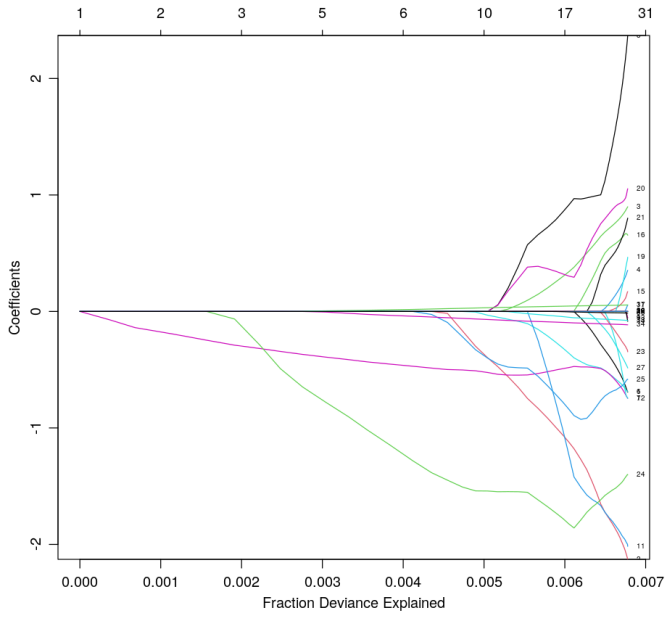

Figure 4.8: Graph that shows how the coefficients change based on the deviance, i.e how coefficient presence changes how well the model fits the data.

4.3.1 Variable importance

In tables 4.4 and 4.5, the variable importances are shown while using λ_{min} and λ_{1se} respectively. The penalization is smaller in the minimum lambda case, and therefore more variables play a role in the model, while for the higher penalization of λ_{1se}, more variables are excluded. Note that every feature except the last 6 are related to the DeSO.

Feature	Importance with λ_{min}
Overcrowded_prop	6.922694e-01
Man_prop	2.128093e+00
NonScandinavian_prop	8.988491e-01
NonHighEducation_prop	3.526945e-01
Employed_prop	0.000000e+00
EconomicStandard_prop	6.883158e-01
Vehicles_prop	7.458534e-01
HighEducatedCare	2.369465e+00
lowEducatedCare	5.417976e-02
m2PerPerson	1.398804e-03
Ald0_5_prop	2.016056e+00
Ald6_10_prop	7.450982e-01
Ald11_15_prop	5.383646e-02
Ald16_19_prop	8.461115e-02
Ald20_24_prop	1.701762e-01
Ald25_29_prop	6.543467e-01
Ald30_34_prop	5.550075e-02
Ald35_39_prop	0.000000e+00
Ald40_44_prop	4.653331e-01
Ald45_49_prop	1.053502e+00
Ald50_54_prop	8.012001e-01
Ald55_59_prop	0.000000e+00
Ald60_64_prop	3.458913e-01
Ald65_69_prop	1.402895e+00
Ald70_74_prop	5.841513e-01
Ald75_79_prop	0.000000e+00
Ald80_w_prop	4.849856e-01
XKOORDsw	7.742816e-05
YKOORDsw	6.572132e-05
Age_2020	3.417991e-03
House_number_of_habitants	5.596768e-02
Income_per_habitant	1.701130e-03
Overcrowded	7.759571e-02
Gender	1.148374e-01
Education_indiv_3	1.377670e-02

Table 4.4: The list of importances using λ_{min} for the model, $\alpha = 1$ and a 10-fold cross validation. Higher value of importance simply implies a higher feature importance.

Feature	Importance with λ_{1se}
Overcrowded_prop	0.1127223727
Man_prop	1.3585095885
NonScandinavian_prop	0.5023381780
NonHighEducation_prop	0.0000000000
Employed_prop	0.0000000000
EconomicStandard_prop	0.4566516587
Vehicles_prop	0.4761500873
HighEducatedCare	0.9754261597
lowEducatedCare	0.0000000000
m2PerPerson	0.0002328554
Ald0_5_prop	1.5762214286
Ald6_10_prop	0.0000000000
Ald11_15_prop	0.0000000000
Ald16_19_prop	0.0000000000
Ald20_24_prop	0.0000000000
Ald25_29_prop	0.1897040047
Ald30_34_prop	0.0000000000
Ald35_39_prop	0.0000000000
Ald40_44_prop	0.0000000000
Ald45_49_prop	0.5297857557
Ald50_54_prop	0.0000000000
Ald55_59_prop	0.0000000000
Ald60_64_prop	0.0000000000
Ald65_69_prop	1.7244417980
Ald70_74_prop	0.9164475648
Ald75_79_prop	0.0000000000
Ald80_w_prop	0.0000000000
XKOORDsw	0.0000802020
YKOORDsw	0.0000524011
Age_2020	0.0029723560
House_number_of_habitants	0.0491913980
Income_per_habitant	0.0006128525
Overcrowded	0.0592024242
Gender	0.1017189289
Education_indiv_3	0.0068839162

Table 4.5: The list of importances using λ_{1se} for the model, $\alpha = 1$ and a 10-fold cross validation. Higher value of importance simply implies a higher feature importance.

4.4 Support vector machine

Support vector machines provide a few values for the fitted object such as the supporting vectors, index of the support vectors from the data matrix, the corresponding coefficients times the labels and the negative intercept. The values of the SVM fit does not lead to a direct representation of feature importance.

The randomized subset of the data was 5 000 000 data points, with the kernel radial basis function and a cost of 5. The split of training to test data was set to $0.7, 0.3$.

4.4.1 Model values

The SVM model resulted in 24 758 support vectors and coefficients. The coefficients produced will not be listed as it consists of 24 758 values. The prediction accuracy was low enough to essentially be of no use.

4.5 Neural network

The neural network model is able to produce a weight matrix along with biases as a result, but the software used did not output any of these results due to the model not converging. Non-convergence also led to SHAP being unusable.

In total for the neural network model, 500 000 data points was used with a $0.7, 0.3$ split in training and testing data. Number of epochs used were 5, with a threshold of 0.05, two hidden layers of size 15 and 8.

4.6 Summary

Overall, the results produced do not show any particular preference for DeSO or individual features, and consistently relates a mix of environmental and individual features as important for Covid-19 cases. High on the list for feature importance assessment for both random forest, shown in figure 4.1 and 4.3, and logistic regression, shown in figure 4.5, are the age, level of education, and living conditions for both the DeSO and the individual. Logistic regression also included the correlation between the economic standard of a DeSO to Covid cases. Random forest, due to considering a larger amount of features, also ranks individual income and occupation highly.

The SHAP values ended up only being used for random forest and provided fairly limited insight on the feature importance. Since the number of features used for this model was large, only a portion could be included in the plot, and only the negative contributions to Covid-19 cases could be shown. Additionally, only a few features showed a significant output, so the information gained is very low. This issue seemingly stems from how data was used for the KernelSHAP computation: only a few hundred data points were used for the Shapley value calculation. For a normal size of data set with fewer features involved, this would not have posed an issue. However, since over 200 features are involved and millions of data points are used, the allocated subset for Shapley calculation were not sufficient. This difference can be seen in the two results of Shapley calculations, figures 4.5 and 4.6, where the latter calculated Shapley values on a model that used a smaller data subset for prediction, but larger subset of data for the Shapley calculation. The Shapley calculation therefore was compromised for the results in figure 4.5 due to computation cost. Ideally, a larger subset of data would have been used for the Shapley calculation, but computation cost proved to be a significant hindrance. Even though TreeSHAP was an option to reduce computation time, attempts at implementing it for random forest were unsuccessful. It is also possible to be more selective with the feature inclusion for random forest, similar to the other models. Initially the purpose of the random forest was to include as many features as possible to gauge possible features of interest, but for the purpose of optimal predictive performance and usage for SHAP value calculations, a smaller subset of features would have been necessary.

Both SVM and NN did not provide any usable results. The SVM model resulted in far too many support vectors, meaning that the model likely grossly overfitted. Due to this, using this model to compute predictions for confusion matrices or KernelSHAP became unreasonable. The NN model did not converge using the subset of data and chosen hyperparameters, and did therefore not output anything. Convergence likely would have required more data, as well as better choices of hyperparameter, but the computation cost ended up being too large of a hindrance.

5 Discussion and conclusions

5.1 Interpretation of the results

The results shown under the random forest model and the logistic regression model display a variety of features that affect the covid outcome of an individual. Interestingly, it showcases a combination of individual factors and factors based on the DeSO, meaning that the covid cases are likely to be based on both the living conditions of an individual and the area they live in. For example, both the proportion of age groups within the DeSO and the age of the individual showed significant importances in both random forest and logistic regression. On the other hand, the proportion of individuals with an education of less than 3 years at university level showed lower importance compared to the factor of an individual having an education of more than 3 years at university level. This could indicate that a higher education means that individuals are more informed and capable of protecting themselves, but lower education does not necessarily imply that they are a risk to others.

5.1.1 Significance of λ_{min} and λ_{1se} for logistic regression

For the logistic regression, two values of lambda could be considered for various operations. When optimizing the value of alpha, the minimum was decided to be used as the $1se$ variant, the lambda value within a standard error of the minimum, often resulted in trivial solutions. The issue originated from the errors calculated by the cross validation, which for this data set only varied a small amount for most α and λ even in the presented error graph, the variation in error is small. In many cases, the error bar would encompass all values of error points, meaning that λ_{1se} simply picks the largest lambda.

The reason for choosing λ_{1se} becomes more apparent when observing the feature importance tables, where increasing the penalization with a larger value of lambda filters out the unnecessary features. Possibly similar to how regularization prevents overfitting in a neural network model by stopping it once it passes some threshold in loss, or how polynomial regression becomes completely nonsensical if there is no limit to the polynomial degree, the minimum value of lambda could be too generous of a penalization. A lack of penalization leads to an overabundance of features that could potentially be relevant to the model, causing it to be unrealistically dependent on far too many variables. Therefore, the choice should, if possible, be the $1se$ variant of lambda to make sure that overfitting in this sense do not occur.

5.2 Model performance

Comparing the results of the models presented in this project, random forest and logistic regression showed better results and were easier to implement for the given data. The data used for the models ended up considerably lower than initially anticipated, as the code would stop running and reset during long runs. The workaround was to shorten the run time and therefore the data subset size, which likely reduced the performance of the affected models. The data used for SVM was reduced in an attempt at applying kernel SHAP to the fit, which had a long run time despite the reduced data subset size. Neural networks are known to be very computationally heavy, but the amount of training is necessary for the model to perform well. It can be seen in the results that the amount of included data points varied between the models, from a subset of millions or a subset of thousands. Using a subset of the total data set is not fully representative of the entire set. Additionally, since each individual belongs to a DeSO, and the values in the DeSO are discrete, a

lower amount of data points can lead to a high variance in the outcome and an unstable model. It is possible that an improvement in computation, whether it is a hardware upgrade or a software optimization, would allow for SVM and NN to have better results.

5.2.1 Model flexibility and computation cost

The issue that become increasingly more apparent over the course of the project was the trade-off between flexibility of the model and its computation time. Due to the size of the data set, the difference in computation time exacerbated this issue even further. Logistic regression, being the simplest and least flexible model, showed relevant results within a decent time frame of computation. Comparatively, on the complete opposite side of the spectrum, the neural network model ended up needing a full day to compute using a subset of 1% of the total data set, and was not able to converge. It is possible that the model can converge if the data used was closer to the total amount, but the computation time and cost would be unreasonable for a project of this size. The main reason that neural networks were used was its potential to capture very complex systems regardless of context, but the high amount of complexity might not be relevant or hold any scientific insight. Over-flexibility of the model might lead it to create relations where none exist, essentially overfitting, since data in reality does not always fully represent the outcome. The SVM model similarly suffered from high computational costs, especially since the kernel used was non-linear. The model produced seemed to also be computationally heavy to predict with, as the SHAP calculation could take up to a whole day to complete with a small subset of the data set.

5.3 Interpretability of the models

A very common problem of machine learning algorithms is the black box nature of such models, where many times the inputs are inserted to the model and outputs are received out on the other end and the computations inside are conducted without user influence or explicit programming. This issue is considerably more apparent with observing the difference between the results of random forest and logistic regression compared to support vector machines and neural network. Random forest and logistic regression are both able to output variable importance: random forest using mean decrease accuracy and mean decrease Gini, while logistic regression show the importance of coefficients related to each feature are connected to the output of the model. Support vector machines and neural networks however are harder to interpret in terms of the features, as the models are inherently just algorithms made for maximizing the ability to predict the outcome. The model created by the support vector machine algorithm does not have any relation to the original dataset. Each feature being represented by a dimension and the separation of the classes by the construction of a hyperplane holds no particular resemblance to the features, as the shape, size, position etc. of the hyperplane does not translate into a more or less valued feature. Usage of this model is more relevant when the classification itself is the desired outcome, but for this project where the relevance of each parameter is desired, the typical version of support vector machines are not very well suited. The output of neural networks is similarly somewhat ambiguous, where the output is essentially just the weights and biases that determine the influence that input data has on the output, but have no meaning as to how individual features directly correlate with the output.

The useful models for this project seem to have been the models with some sort of ability to evaluate the performance of the model based on the individual features. Random forest utilizes the mean decrease measurements as well as the Shapley values to determine the importance, and logistic regression uses the coefficients in combination with the penalization to see importance. The increase in penalization means that less valuable features that affect the model less are penalized and removed, leaving the important ones. Support vector machines seem to have the option to utilize a penalizing factor to achieve a similar effect. Additionally, there are alterations to the support vector machine model that can produce variable selection that were not implemented for this project [19]. The results of SVM produced in this project did point to the necessity of some form of regularization, as the results ended up significantly overfitted. This idea is the same as the regularization applied to logistic regression and neural networks in order to prevent overfitting. This would also lead to the additional need to optimize for regularization parameters, possibly leading to more issues regarding computational costs.

The Shapley values serves as a solid choice for model agnostic evaluation of the feature importance, as it uses the predictive capabilities of the model fitting to judge the payout of each feature. However, similar to the choice of model, calculating Shapley values are computationally costly, which is further exacerbated by a large and complex model fit. Tree-based algorithms have the option of TreeSHAP to reduce computational costs, but this choice also comes with drawbacks related to TreeSHAP.

5.3.1 Feature interaction and H-statistics

The results of the project only show the importance of features when directly related to the outcome of covid. This is not necessarily an accurate depiction of the features, as many factors can depend on each other. For example, an individual of high socio-economical status may be less susceptible to transmission, but only if they live in a DeSO where the living standard is high. Only when given a certain surrounding or environment would some features end up playing a larger role in covid cases.

Models are normally making predictions on the outcome based on a couple of terms: a constant term, terms for each individual feature on their own and terms for the interactions between features. A way to estimate this interaction between variables is to measure how the variation of the prediction depends on the interaction, known as the H-statistic, introduced in 2008 by Friedman and Popescu [6].

The H-statistics can be separated in to two categories: the two-way interaction that describes the interaction between two features, and the total interaction measure that describes the interaction between a feature and all other features in the model. Technically there is no limit to the number of features chosen for this analysis, the two-way and total interactions are the most interesting cases.

Calculation of the H-statistic is computationally expensive as it iterates over all data points and evaluation is of partial dependence is conducted with all data points in turn. To make matters worse, the two-way statistics (j vs. k) and total H-statistics (j vs. all) require $2n^2$ and $3n^2$ calls respectively to the predict function of the machine learning model. To save on computation, it is possible to instead use a subset of the n data points, but this comes at the cost of increasing variance and partial dependence estimates, which decreases the stability of the H-statistics.

5.3.2 Machine learning compared to statistical analysis

Machine learning primarily focuses on optimizing the performance of the model and its predictive ability, while statistical approaches instead puts the variable relations as its priority. Machine learning is therefore able to find variable relations that statistical analysis would not necessarily consider. However, this also means that whatever context is used for the analysis is lost during the learning and the results will not necessarily be relevant. As has been discussed previously, one of the main challenges of machine learning models is the interpretation of its results. Other applications where the prediction provided is the main usage of the model do not suffer from this problem. The choice of machine learning algorithms over statistical analyses could potentially be useful for certain situations where the data type or format fits the situation. In this project where the variable importances are of interest, machine learning algorithms are generally difficult to utilize fully, needing the use of Shapley values to complement the interpretability. The large size of the data set additionally posed a challenge when attempting to complete the learning due to the computational costs.

In a sense, machine learning models for application similarly as this project grants the opposite information as a statistical model: statistical models set up the variable relations and the user figures out a way to optimize the outcome, while machine learning optimizes that outcome and the user instead figures out the variable relations. For an application where the variable importances are the main focus, machine learning can offer an avenue of exploration, wherein the data and its patterns determine the model outcome

5.4 Further research

One of the ideas that were not explored in this project was changing the covid indicator variable. The variable used was Any_Covid, but as mentioned in the section for materials, many variable representing some sort of covid indication were available, such as death by covid or the number of ICU cases. It could be possible to combine those two with Any_Covid to be able to predict the level of covid instead, too. Alternate choices for

data pre-processing, such as the indicator variable as well as the variables included in the numerical models, could lead to a broader or different analysis compared to the current choices.

5.4.1 Temporal analysis

One of the use cases for analysis of covid data is for the understanding and preparation for potential future pandemic threats. A real case of disease spreading would involve a passing of time, whereas the analysis of this project only considered the overall cases of covid during the pandemic period. While the data for the temporal analysis was included in the dataset, it was not used. The provided temporal data was collected over the course of 26 months, similarly documenting cases of any covid, ICU cases and deaths by covid on each month for each DeSO. However, the other covariates in the data set do not necessarily have a similar temporal property documented, e.g. age, income and jobs could and would have changed over the time period of the pandemic. Some covariates such as year of birth and country of birth however are temporally invariant and are therefore usable in a temporal analysis. It is unlikely to be as simple as replacing the covid indicator used in this project with each of the monthly indicators instead, as many covariates in the data set are temporally static.

5.4.2 Spatial dependency

Most of the models used for this project assumes that the data points are individually independent, therefore that each point is uncorrelated with the others. Even though the feature that refers to the X and Y coordinate are used in the modelling, it is just another feature that will show whether the location of an individual's DeSO plays a role in predicting covid outcome. Spatial relations in this project is considered only when it relates to the DeSO features, especially the spatially aggregated version. This can be seen as the neighbourhood of an individual, which typically in spatial statistics are the adjacent sites of a discrete grid of points. Though every individual has this neighbourhood, it is not interactable, meaning that there is no spatial process.

One of the machine learning algorithms that are capable of taking spatial dependencies into account is the *Convolutional Neural Network*, a model typically used in image analysis where the image is pre-processed through convolution and reduced to more reasonably fit into a neural network framework. The convolution of the image functions both as a way for the neural network to work with spatially correlated information, but also to condense the parameter quantity to lower the model training time to a reasonable amount, since normally each pixel value of the image is an input for the neural network, which even at low resolutions are a large amount of pixels. This approach is however limited to the usage of a discrete field of values, essentially akin to that of an image. The data of individuals in a geographical sense does not translate particularly well into such a format of data, so this approach is unlikely to be viable for this type of analysis.

The most accurate representation of the actual data would be a marked point process, which would have points on a continuous map representing the individuals on a geographic space. This is however also not a particularly reasonable approach, as the required data would be very intrusive to the privacy of individuals due to the necessity of accurate locations of each individual or their households.

5.4.3 Model accuracy

The accuracy of the models were not in focus in this project and have only been mentioned as a side note. Due to a personal inexperience with the used programming language and a lack of access to resources in terms of library and packages, optimizing the models were not considered particularly heavily. As the models still produced relevant results, as well as the time constraint of the project, this did not pose any apparent issue. It is however important to note that the primary use case of machine learning is its predictive capabilities, and that a severely suboptimal model does not inspire confidence in the final results being accurate. Future work in this regard would be to implement these models, as well as other models, in a more optimal manner to observe whether the outcomes are similar.

Specifically, when it comes to optimizing machine learning models, running a k-fold cross validation to find hyperparameter optimums would be the most common option, similar to how it was performed with the logistic regression model. The issue here once again comes back to the heavy computational cost - running the model once is already a tall order, and running multiple of them for optimization is far worse. Some better choices for model options, as in options for the model framework in a certain program, in this case R,

could definitely have made a significant difference. An alternative to R would have been Python, due to its wide usage and development within the machine learning industry.

5.5 Conclusion

The usage of machine learning models have had widespread applications in a variety of fields of research, and this project served as a proof of concept to show whether it is also applicable in the analysis of covid data. In practice, the sheer quantity of data to process served as a major challenge, and models with high or even medium levels of computational costs were difficult to handle. Even after fitting the model, making the results interpretable was not straightforward, as the covariates of the data needed to be clearly related to the outcome of the models. It is not unreasonable to consider both support vector machines and neural networks to be viable machine learning models for this sort of analysis, but to confirm whether it produces reasonable and interpretable results for this set of data could not be verified in this project.

Random forest and logistic regression were capable of producing results that were analytically interesting, where many of the prominent important features were reasonable in the context of the data source. The combination of important feature stemming from a wider societal scope with the personal information of the individual provide possible insights to the factors playing a role in the risk of individual cases of Covid-19.

Reference

[1] Parveen S. Abbass K. Song H. Achim M. V. Naseer S., Khalid S. Covid-19 outbreak: Impact on global economy. *Front Public Health. 30;10:1009393, doi: 10.3389/fpubh.2022.1009393*, 2023.

[2] J. Feehan. Is covid-19 the worst pandemic? *Maturitas, doi: 10.1016/j.maturitas.2021.02.001*, 2021.

[3] M. et al Nicola. The socio-economic implications of the coronavirus pandemic (covid-19): A review. *Int J Surg, doi: 10.1016/j.ijsu.2020.04.018*, 2020.

[4] Wanihaya Irshad Ahmad Reshi, Khanrafiq. Covid-19 pandemic and teaching and learning: A literature review. *MORFAI JOURNAL, 2(4), 820–826. https://doi.org/10.54443/morfai.v2i4.693*, 2023.

[5] Armon A. Shwartz-Ziv, R. Tabular data: Deep learning is not all you need. *arXiv:2106.03253 ,doi: https://doi.org/10.48550/arXiv.2106.03253*, 2021.

[6] C. Molnar. *Interpretable Machine Learning*. Bookdown, 2023-08-21.

[7] K. Narasimhan B. Hastie T. Tay, J. Elastic net regularization paths for all generalized linear models. *Journal of Statistical Software, 106(1), 1–31. https://doi.org/10.18637/jss.v106.i01*, 2023.

[8] Hastie T. Zuo, H. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology, 67(2), 301-320, https://doi.org/10.1111/j.1467-9868.2005.00503.x*.

[9] Hastie T. Zuo, H. Addendum: Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology, 67(5), 768, https://doi.org/10.1111/j.1467-9868.2005.00527.x*, 2005.

[10] Qian J. Tay K. Hastie, T. An introduction to glmnet. *Stanford University*, 2023-03-27.

[11] Daniel Berrar et al. Cross-validation., 2019.

[12] L. Breiman. Random forests. *Machine Learning 45, 5–32, https://doi.org/10.1023/A:1010933404324*, 2001.

[13] Tibshirani R. Friedman J. Hastie, T. *Random Forests. In: The Elements of Statistical Learning*. Springer Series in Statistics. Springer, New York, NY. https://doi.org/10.1007/978-0-387-84858-$7_1$5, 2009.

[14] E. Biau, G. Scornet. A random forest guided tour. *arXiv:1511.05741, https://doi.org/10.48550/arXiv.1511.05741*, 2015.

[15] S. Suthaharan. *Support Vector Machine. In: Machine Learning Models and Algorithms for Big Data Classification*. Integrated Series in Information Systems, vol 36. Springer, Boston, MA. https://doi.org/10.1007/978-1-4899-7641-3_9, 2016.

[16] B. Mehlig. *Machine Learning with Neural Networks: An Introduction for Scientists and Engineers*. Cambridge University Press, 2021.

[17] Kononenko I. Štrumbelj, E. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems 41.3: 647-665*, 2014.

[18] Lee S. Lundberg, S. A unified approach to interpreting model predictions. *arXiv:1705.07874, doi: https://doi.org/10.48550/arXiv.1705.07874*, 2017.

[19] A. Rakotomamonjy. Variable selection using svm-based criteria. *Journal of Machine Learning Research 3*, 2003.

Appendix

A | Biostatistical data

A.1 Sociodemographic features

Name	Description
SDeSO	
lopnr	Anonymous serial number replacing social security number
HushallsStallning	Household status
HushallsTyp	Household type
HushallsStorlek	Household size
AntalBarn	Number of children
LGHKAT	Type of accommodation
SCB_Upplatform	States if the accommodation is provided by tenancy, condominium or ownership from the perspective of the accommodation
boarea	Living area
antrum	Number of rooms
Boendeform	Form of accommodation
FamSt	Family status
Barn0_3	Children age 0 to 3 living at home
Barn4_6	Children age 4 to 6 living at home
Barn7_10	Children age 7 to 10 living at home
Barn11_15	Children age 11 to 15 living at home
Barn16_17	Children age 16 to 17 living at home
Barn18_19	Children age 18 to 19 living at home
Barn20plus	Children age 20 or higher living at home
SenInvAr	Latest year of immigration
BeslDatum	Date of decision for basis for residence
SSYK_ArbTill	Code of occupation of worker
InreAr	Year of entry
Sun2020Niva_old	Education group 2020, old
Sun2020Inr	
Sun2020Niva	
Sun2000Grp	Education group, highest completed, SUN2000
KallKod	Source for information on highest education
ExamAr	Year of graduation
VistTid	Number of days registered in Sweden
AntalVuxnaHushallet	Number of adults in household
Trangbodd	Number of individuals living in an overcrowded environment
civilantar	Number of years in marital status

Table A.1: Table 1 of biostatistical data

Name	Description
DeSO_err	Demographic statistical areas, error
Ruta	Size of square, 250 meter square or 1000 meter square
XKOORDsw	x-coordinate
YKOORDsw	y-coordinate
Tatort	Urban area
FodAr	Year of birth
Kon	Gender
fodelselandnamn	Name of country of birth
fodelsetidfar	Birth time of father
fodelselandnamnfar	Country of birth of father
fodelsetidmor	Birth time of mother
fodelselandnamnmor	Country of birth of mother
Ssyk3_2012_J16	Occupation by SSYK2012, 3-digit level, including derived occupations for entrepeneurs
Ssyk4_2012_J16	Occupation by SSYK2012, 4-digit level, including derived occupations for entrepreneurs
SsykAr_J16	Year of occupation information, including occupations for entrepreneurs
SsykStatus_J16	Agreement of the occupation according to the november job, including derived occupations for entrepreneurs
KU1CfarNr	Workplace CFAR-number
KU1AstNr	Workplace number according to RAMS
KU1AstKommun	Workplace municipality
KU1SektorKod	Sector affiliation (company)
KU1AstSNI2007	Workplace industry, SNI2007
KU1Ssyk3_2012	Occupation according to SSYK 2012, 3 digit level, for professionals
KU1Ssyk4_2012	Occupation according to SSYK 2012, 4 digit level, for professionals
KU1SsykAr	Year of professional duties
KU1SsykKalla	Source of professional duties
KU1SsykStatus	Occupation accordance with largest source of acquisition
KU2CfarNr	Workplace CFAR-number
KU2AstNr	Workplace number according to RAMS
KU2AstKommun	Workplace municipality
KU2SektorKod	Sector affiliation (company)
KU2AstSNI2007	Workplace industry, SNI2007
KU2Ssyk3_2012	Occupation according to SSYK 2012, 3 digit level, for professionals
KU2Ssyk4_2012	Occupation according to SSYK 2012, 4 digit level, for professionals
KU2SsykAr	Year of professional duties
KU2SsykKalla	Source of professional duties
LoneInk	Gross salary
DekLon	Declared salary income
ForvErs	Erned income and work related compensation
StudMed	Student aid and help
ForLed	Sum of income prompted by parental leave

Table A.2: Table 2 of biostatistical data

Name	Description
SjukRe	Sum of income prompted by sick leave
SjukP_Bdag_MiDAS	Sickness benefit, number of gross days
SjukP_Ndag_MiDAS	Sickness benefit, number of net days
Akassa	Allowance from unemployment fund/insurance
ArbLos	Sum of income prompted by unemployment
ALosDag	Number of days in unemployment
AK14Dag	Number of days as job seeker with disabilities
ADelDag	Number of days in part-time unemployment
AmPol	Sum of income prompted by labour market policy measure
AmPolTyp	Presence of allowance in connection to labour market policy measure
ForTid	Sum of income prompted by early retirement or sickness benefit
ForTidTyp	Presence of early retirement or sickness benefit
SjukErs_Bdag_MiDAS	Sickness benefit, number of gross days
SjukErs_Ndag_MiDAS	Sickness benefit, number of net days
SjukErs_Belopp_MiDAS	Allowance amount for months with non-time limited and/or time limited sickness benefit
AldPens	Sum of income of age related retirements
SocBidrFam	Welfare for family1
SocBidrTypF	Presence of welfare for family
BostBidrFam	Housing allowance
HKapErs	Disability allowance
BidrFor	Maintenance support/advances in maintenance payment
AnnInkF18	
UnderHBidrGiv	Given alimony, simulated amount
UnderHBidrMot	Received alimony, simulated amount
DispInkKE	Disposable income per consuming unit (family)
DispInkKE04	Disposable income per consuming unit (family), according to 2004 definition
DispInk04	Disposable income (individual component)
DispInkFam04	Disposable income of family
Raks_Huduvanknytning	Main affiliation to the job market
Raks_EtablGrad	Degree of establishment
Raks_SumMan	Number of months that an individual has worked according to monthly marking on the statement of earnings and deductions
Raks_ArblosInk	Income from unemployment
Raks_forTidInk	Income from sickness or activity related allowance 6
Raks_AldPensInk	Income from age related retirement
Raks_StudInk	Income from studies
Raks_UtBidrInk	Income from labour market policy measures
Raks_VardInk	Income from parental leave or care for close relatives 6
Raks_SjukInk	Income from sickness/work injury/rehabilitation 6
Raks_EkBisInk	Individualised income from economic aid

Table A.3: Table 3 of biostatistical data

Name	Description
Raks_Forvink	Income from employment or businesses operations
Raks_SummaInk	Total income
Raks_AndelForvink	Proportion of the total income consisting of income from employment or businesse operations
Raks_AndelStudInk	Proportion of the total income consisting of income from studies
Raks_AndelUtbBidrInk	Proportion of the total income consisting of income from labour market policy measures
Raks_AndelVardInk	Proportion of the total income consisting of income from parental leave or care for close relatives 6
Raks_AndelSjukInk	Proportion of the total income consisting of income from sickness/work injury/rehabilitation 6
Raks_AndelArblosInk	Proportion of the total income consisting of income from unemployment
Raks_AndelAldPensInk	Proportion of the total income consisting of income from age related retirement
Raks_AndelForTidInk	Proportion of the total income consisting of income from sickness or activity related allowance 6
Raks_AndelEkBisInk	Proportion of the total income consisting of individualised income from economic aid
Raks_HuvInkKalla	Main source of income
MedbMan	Month of citizenship
Medblandnamn	Citizenship
pop.y	Population demarcation code
SyssStat	Employment status
YrkStalln	Occupation status
SFIBetyg	SFI grade
ArbetsInk	Labour income
studDelt	Student participation, autumn term, form of education
FoDelt	Registration at university, research education
HSDelt	Registration at university, base education
m2perPerson	Similar to "boarea"
age_2020	Age as of 2020
age_group	Age groupå
Gender	Gender
Baseline_Diabetes	Diabetes
Baseline_Hypertension	Hypertension
Baseline_AF	Atrial fibrillation
Baseline_Demens	Dementia
Baseline_COPD	Chronic obstructive pulmonary disease
Baseline_Heartfailure	Heart failure
Baseline_Cancer	Cancer
Baseline_Obesity	Obesity
Baseline_VTE	Venous thromboembolism
Baseline_MI	Myocardial infarction
Baseline_Stroke	Stroke
Born_in_Nordic	Born in nordic
House_m2_per_Habitant	Same as "boarea"
House_Number_of_Habitants	Same as "HushallsStorlek"

Table A.4: Table 4 of biostatistical data

Name	Description
Need_of_Care	In need of care or net
Education_3	Education on university level of over 3 years
Income_per_habitant	Similar to "DispInkKE"
deso_pop_dens	Population density of DeSO
Type_of_region	Rural, urban or suburban
POS_TEST	Yes or no to positive test of covid
Inpatient	Inpatient due to covid or not
ICU	ICU patient due to covid or not
Death_Covid	Death due to covid or not
Death_Covid_any_period	Death due to covid or not in any period
Death_within_30_days	Death by covid within 30 days of contraction or not
Any_death	Any death by covid
Any_death3	Any death by covid in 3 months
And_Covid	Any covid case
Covid_during_any_period	Covid case during any period
Covid_severity	Severity of covid
Covid_ICU_SIR	Standardized infection ratio of ICU cases
Covid_ICU_by_code	ICU case by code
Covid_Hosp_Level	Hospital level of covid case
Covid_Death_Levels	Death level of covid death case
Region_name	Name of region
PERIOD	Covid period of individual
PERIOD3	
MIN_POS_DAT_TOTAL	Date of total case of minimum level of covid
Place_Of_Death	Location of death case
Ssyk4_2012_J16.y	Same as Ssyk_4_2012_J16
Work_Group	Work group 1
Work_Group2	Work group 2
Combined_Occupation	Combined occupation of individual
Work_from_Home	Degree of working from home
Baseline_CCIw	Cranial Cervical Instability
Baseline_CCIunw	
Exposure_level	Level of exposure
Exposure_level_old	Level of exposure, old
DeSO_old	Same as SDeSO
month_Covid_test	Month of covid test
month_Covid_Death	Month of covid death
month_Covid_ICU	Month of covid ICU

Table A.5: Table 5 of biostatistical data

A.2 Demographic statistical areas

Population by age

Name	Description
Ald0_5	Individuals aged 0 to 5
Ald6_9	Individuals aged 6 to 9
Ald10_14	Individuals aged 10 to 15
Ald15_19	Individuals aged 16 to 19
Ald20_24	Individuals aged 20 to 24
Ald25_29	Individuals aged 25 to 29
Ald30_34	Individuals aged 30 to 34
Ald35_39	Individuals aged 35 to 39
Ald40_44	Individuals aged 40 to 44
Ald45_49	Individuals aged 45 to 49
Ald50_54	Individuals aged 50 to 54
Ald55_59	Individuals aged 55 to 59
Ald60_64	Individuals aged 60 to 64
Ald65_69	Individuals aged 65 to 69
Ald70_74	Individuals aged 70 to 74
Ald75_79	Individuals aged 75 to 79
Ald80_w	Individuals aged 80 and higher
Totalt	Total amount of individuals

Table A.6: Table of population by age.

Population by gender

Name	Description
Kvinnor	Women
Man	Men
Totalt	Total number of individuals

Table A.7: Table of population by gender.

Population by Swedish or foreign background

Name	Description
Sv_bakgr	Individuals of swedish background (includes born in Sweden with both parents born in Sweden and born in Sweden with one parent born in Sweden)
Utl_bakgr	Individuals of foreign background (includes individuals born outside of Sweden and individuals whose parents both are of foreign background)
Totalt	Total number of individuals

Table A.8: Table of population by background.

Household after household type

Name	Description
Sam_barn	Living together with children
Sam_ubarn	Living together without children
Ens_barn	Living alone with children
Ens_ubarn	Living alone without children
Ovrigt	Other household types
Totalt	Total number of individuals

Table A.9: Table of population by household type.

Population of age 20-64 by occupation

Name	Description
Forvarb	Wage labour
Ej_Forvarb	Non-wage labour
Tot_20_64	Total amount of individuals of age 20 to 64

Table A.10: Table of population by occupation.

Population of age 25-64 by education level

Name	Description
Forgymn	Pre-upper secondary high school
Gymnasial	Highschool
Eftergymn1_3	Post-high school, less than 3 years
Eftergymn3_w	Post-high school, more than 3 years
US	Missing information
Tot25_64	Total number of individuals of age 25 to 64

Table A.11: Table of population by education level.

Population of 20+ by summarized earned income

Name	Description
Median_Ink	Median income
U_median	Income under the median of the country
0_median	Income over the median of the country
Tot20_w	Total number of individuals over the age of 20

Table A.12: Table of population by income.

Population by economic standard

Name	Description
Median_ek	Median for economic standard
U_median	Economic standard under the median of the country
O_median	Economic standard over the median of the country
Tot20_w	Total number of individuals over the age of 20

Table A.13: Table of population by economic standard.

Population by form of grant

Name	Description
Agratt	Ownership
Boratt	Comunally owned
Hyrratt	Tenancy
US	Missing information
Totalt	Total number of individuals

Table A.14: Table of population by form of grant.

Vehicles owned by individuals by vehicle type and status

Name	Description
PB_traf	Private cars in traffic
PB_avst	De-registered cars
MC_traf	Motorcycles in traffic
MC_avst	De-registered motorcycles
EUmop_traf	EU-mopeds in traffic
EUmop_avst	De-registered EU-mopeds
Ovr_traf	Other vehicles in traffic
Ovr_avst	Other de-registered vehicles
Tot_traf	Total vehicles in traffic
Tot_avst	Total de-registered vehicles

Table A.15: Table of population by vehicular ownership.

Population by age proportions

Name	Description
Ald0_5_prop	Individuals aged 0 to 5
Ald6_9_prop	Proportion of individuals aged 6 to 9
Ald10_14_prop	Proportion of individuals aged 10 to 15
Ald15_19_prop	Proportion of individuals aged 16 to 19
Ald20_24_prop	Proportion of individuals aged 20 to 24
Ald25_29_prop	Proportion of individuals aged 25 to 29
Ald30_34_prop	Proportion of individuals aged 30 to 34
Ald35_39_prop	Proportion of individuals aged 35 to 39
Ald40_44_prop	Proportion of individuals aged 40 to 44
Ald45_49_prop	Proportion of individuals aged 45 to 49
Ald50_54_prop	Proportion of individuals aged 50 to 54
Ald55_59_prop	Proportion of individuals aged 55 to 59
Ald60_64_prop	Proportion of individuals aged 60 to 64
Ald65_69_prop	Proportion of individuals aged 65 to 69
Ald70_74_prop	Proportion of individuals aged 70 to 74
Ald75_79_prop	Proportion of individuals aged 75 to 79
Ald80_w_prop	Proportion of individuals aged 80 and higher

Table A.16: Table of population by age proportions.

Other features

Name	Description
PopStorlek	Population size
AntalTrangbodda	Number of individuals living in overcrowded environments
Overcrowding_Prop	Proportion of individuals living in overcrowded environment
Medelm2perPerson	Average of m^2 per person
SwedenBorn_Prop	Proportion of individuals born in sweden
higheduhealthwork	Proportion of individuals in high education health work
loweduhealthwork	Proportion of individuals in low education health work
EkoStandUnderMedian_Prop	Proportion of individuals in an economical standing under the median
Born_in_Nordic_Prop	Proportion of individuals born in nordic countries
HostpitalStaff_DeSO_Prop	Proportion of individuals working as hospital staff in a DeSO
Man_Prop	Proportion of men
Working_Prop	Proportion of individuals working
Vehicles_Prop	Proportion of individuals owning vehicles
EducationAtLeast3_Prop	Proportion of individuals having an education of at least 3 years
noEducationAtLeast3_Prop	Proportion of individuals without an education of at least 3 years
ScandinaviaBorn_Prop	Proportion of indivudals born in scandinavia
noScandinaviaBorn_Prop	Proportion of indivudals not born in scandinavia

Table A.17: Table of proportions of various features as a result of spatial aggregation.

Case counts

Name	Description
TestCaseCountMonthN	Number of test cases in the month $N = 1, 2, ..., 26$
ICUCaseCountMonthN	Number of ICU cases in the month $N = 1, 2, ..., 26$
DeathCaseCountMonth	Number of Death cases in the month $N = 1, 2, ..., 26$
TotalTestCases	Total number of test cases
TotalICUCases	Total number of ICU cases
TotalDeathCases	Total number of death cases

Table A.18: Table of case counts, ICU and death cases for each month of the pandemic, and total cases for the same categories.

Department of Mathematical Sciences
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY