Analyzing Covid data to discover factors associated with death

Erlend Kristensen, Mathias Mellemstuen, Magnus Selmer-Anderssen Kråkenes
(Dated: December 18, 2022)

In this project we analyzed Covid-19 data of people who died or did not die of Covid-19. The data set consisted of 21 features, such as whether or not someone had diabetes. We started out by using MLP and Random forest classifiers to try to predict whether someone with Covid-19 would die or not, and found Random forest to be a better classifier for this. We also extracted the feature importance using the Random forest classifier, and found age to be the most important feature. We then looked into the chance of dying based on age, and tried to fit a linear regression model to this data with as few samples as possible, and figured we only needed about 200000 samples to fully predict the death rate of Covid-19. After this we compared the death rate of the general population compared to those with diabetes, and found that those with diabetes were at larger risk of dying from Covid-19 across all age groups. With this we proved that combining data analysis with machine learning can be an important tool for predicting the health risks for a new virus outbreak and see which groups are more at risk, so we know who to prioritize for vaccines and other treatment methods.

I. INTRODUCTION

The Covid-19 pandemic has caused the death of at least 6.6 million people [World Health Organization (WHO) (VII), 2022]. Lockdowns were introduced to reduce the disease's spread, and while effective, the lockdowns have negatively impacted both the physical and mental health of the people, while also seriously hampering the economy of the world. Experts fear we now face a new wave of the disease [Folkehelseinstituttet (FHI) (VII), 2022], and there is a possibility that we will never truly rid ourselves of it and that it is something we will have to learn to live with. To prepare for such an eventuality, we want to create a tool to evaluate the major contributing factors to death due to the disease. By settling which groups are more at risk, we hope to provide improved health recommendations to the public.

We will utilize two different machine learning algorithms: The *Multilayer Perceptron* (MLP) and *Random Forest*. The two algorithms are trained with a huge covid dataset to classify the likelihood of death based on a large number of features. Once the algorithms are trained, we will analyze the models to extract the relative importance of each feature. We will then use simple regression methods to further investigate the most important contributing features.

We will introduce the necessary background and theory in section II. In section III we will explain how we have implemented the machine learning algorithms to get our results. The code itself can be accessed in section VI. The results are presented and discussed in section IV. Here we will also compare our results to previous studies. In section V we will present our conclusion and suggest further work.

II. THEORY AND METHODS

A. Data set

We will use a publicly available data set (A) in our investigation. The data is provided by the Mexican government, and it contains approximately one million samples, with up to 21 features. The majority of the features are categorical, most of which are in a yes/no fashion. In these cases, the positive case 'yes' is represented by the number 1, while the negative case 'no' is represented by the number 2. Only one feature, 'age', is numerical. However, many samples are lacking information on one or several features. In these cases, the lacking data is represented by a number between 97 and 99.

To discover which features are important in causing death, we will use the Random Forest classifier, which has a built-in method for this. A comparison of different machine learning algorithms also found that the Random Forest algorithm performed particularly well when predicting covid-19 mortality [Moulaei et al. (VII), 2022].

B. Classifiers

1. *Multilayer Perceptron*

MLP is a special case of feed forward neural networks where every layer is a fully connected layer as shown in this figure:

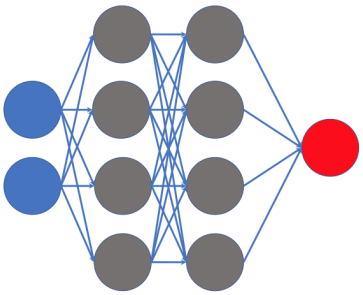

FIG. 1. A representation of what an MLP classifier could look like.

The use for MLP compared to a normal feed forward neural network is when we don't know a lot about the structure of the problem. Using fully connected layers allows us to learn the structure, rather than impose it [Alex VII]. For more on feed forward neural networks, have a look at our previous report on it found here.

2. *Random Forest*

The random forest classifier is built on the method we call *decision trees*. A decision tree starts with a simple question and follows it up with more questions to determine a yes or no answer. In our case, this question would be "will covid kill this person?", followed by questions of whether or not this person has different diseases or complications found in our data set features. Each time a question is asked, we get two different branches, one for yes and one for no. The answers to our original question are at the very end, and are called *leaf nodes*.

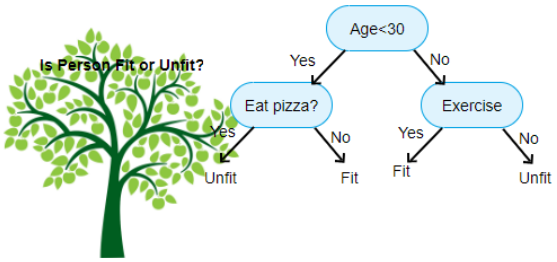

FIG. 2. A representation of what a decision tree would look like, asking the question "is this person fit?".

A random forest classifier, however, uses feature randomness to create an uncorrelated forest of decision trees. So while decision trees consider all possible features, random forest selects only a subset of these for each tree, and uses all the trees in combination to find an answer [IBM VII].

C. Evaluation

Since the task of classifying whether someone dies of covid or not is pretty difficult and complex, we will need to look at more than one form of performance measurement other than accuracy. We will therefore also look at *F1-score*, which can be given by this formula:

$$F_1 = 2 * \frac{precision * recall}{precision + recall} = \frac{2tp}{2tp + fp + fn} \tag{1}$$

where tp is true positives, fp is false positives, and fn is false negatives.

III. IMPLEMENTATION

We will use the built in classifiers in the *Scikit-learn* library, such as the MLP classifier and forest of trees classifier (Random forest). We will have to preprocess the data to fit our models, such as changing the dates of death to simply a binary 1 if dead or 0 if alive. We will first use MLP and Random forest to see how well these classifiers are able to predict whether someone dies of covid or not. Then, we will extract the most important features from the Random forest classifier. We will also use sklearns linear regression and polynomial features on our data to predict the probability of dying from covid based on age and then add in other factors later.
How the different classifiers work and how to use them can be found on these links: MLP, RandomForestClassifier, Polynomial features, Linear regression.

We will also start by looking at the *bias-variance tradeoff* for our two classification methods, as well as linear regression methods for *OLS*,*Lasso* and *Ridge* regression (more on these can be found here, but it is not essential for this report). To do this we used regression with the *Franke function*, which is a 2-dimensional function (see Appendix B), and is often used to train networks for regression problems due to its complexity.

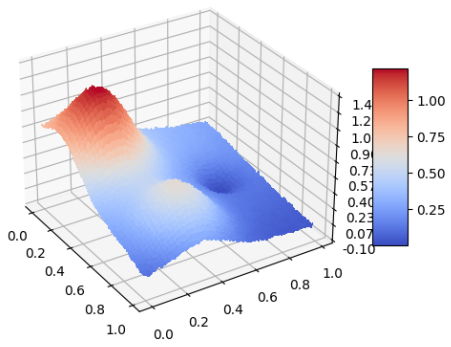

FIG. 3. Visual representation of the Franke function using $x, y \in [0, 1]$.

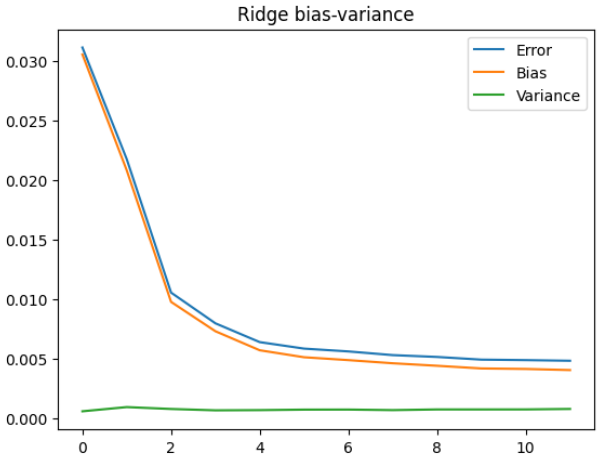

FIG. 5. Plot of bias-variance tradeoff for Ridge regression. x-axis is number of polynomials used. Made using franke function with $x, y \in [0, 1]$, so 100×100 data points.

IV. RESULTS AND DISCUSSION

Before we start analyzing the data, we want to look into the bias-variance tradeoff for our two classification models and linear regression models. This is to study the optimal balance in complexity for the models.

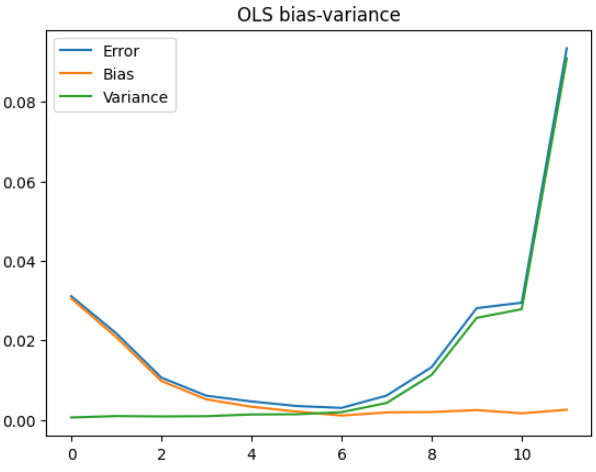

FIG. 4. Plot of bias-variance tradeoff for OLS regression. x-axis is number of polynomials used. Made using franke function with $x, y \in [0, 1]$, so 100×100 data points.

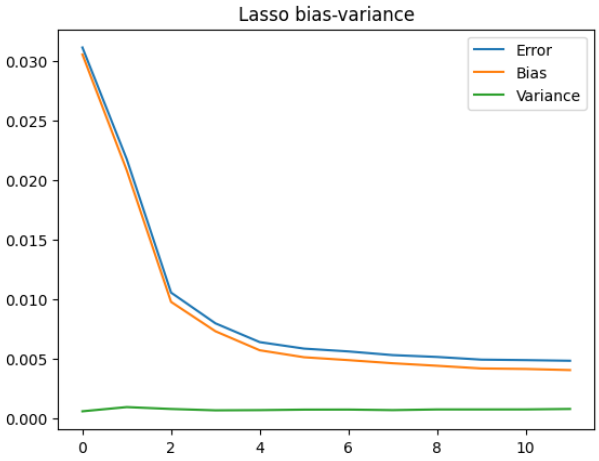

FIG. 6. Plot of bias-variance tradeoff for Lasso regression. x-axis is number of polynomials used. Made using franke function with $x, y \in [0, 1]$, so 100×100 data points.

As we can see from figures 4, 5, 6, the optimal number of polynomials for OLS is about 6, while for Ridge and Lasso, the bias-variance tradeoff is more stable, but results in a worse MSE for the optimal solution compared to OLS. This means that OLS might be best performing for certain scenarios, but Lasso and Ridge being more stable means they might generally be better choices.

We now do the same for our MLP and Random forest classifiers:

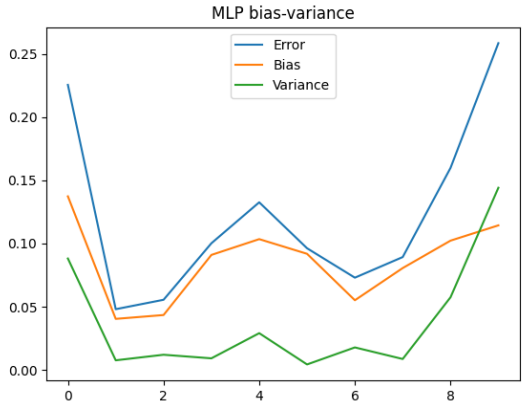

FIG. 7. Plot of bias-variance tradeoff for MLP. x-axis is number of layers used. Made using franke function with $x, y \in [0, 1]$, so 100×100 data points.

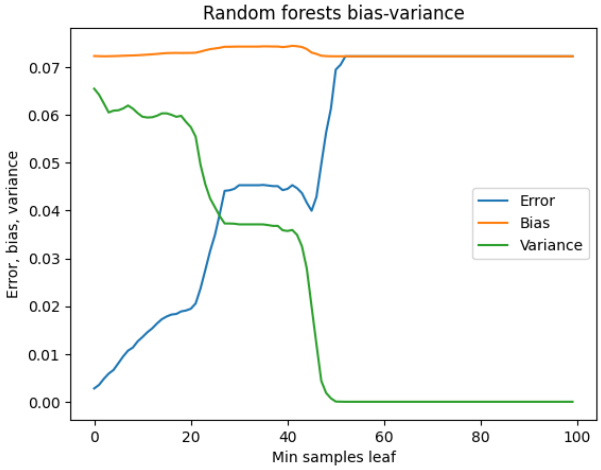

FIG. 8. Plot of bias-variance tradeoff for Random forest classifier. x-axis is minimum number of samples required to be a leaf node. Made using franke function with $x, y \in [0, 1]$, so 100×100 data points.

As we can see in figure 7, the bias-variance tradeoff is around 4 layers, so we will use this moving on with our data set. For the Random forest classifier that we see in figure 8, the bias stayed about the same, but the variance tradeoff was at about 25 samples for each leaf. Keep in mind that since we use regression to test the bias-variance tradeoff, it means that we might get a different result than if we tested using our classification data set. But because it is such a complex set, we chose something that is more easily used for measuring performance.

We now want to analyze the different activation functions for our MLP classifier, to see how they perform on the given data set and see which hyperparameters work best. We will use a sample size of the data, consisting of 10000 randomly chosen points.

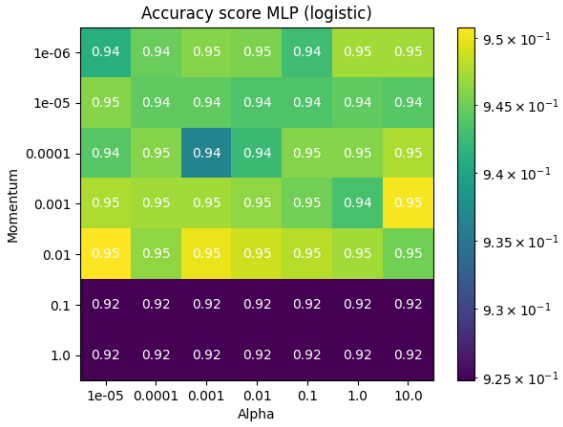

FIG. 9. Plot of accuracy made using layer sizes $[18, 15, 12, 8]$ with 500 max iterations and logistic as activation function.

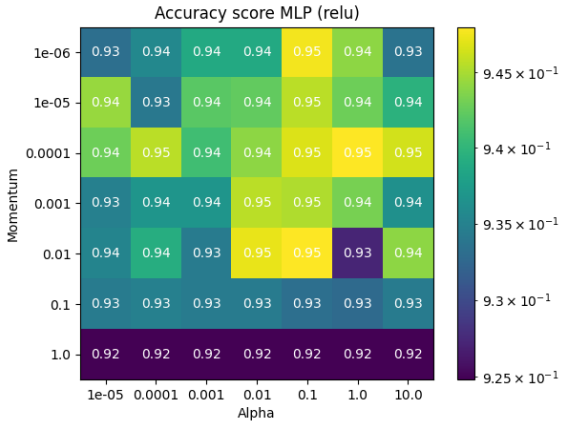

FIG. 10. Plot of accuracy made using layer sizes $[18, 15, 12, 8]$ with 500 max iterations and relu as activation function.

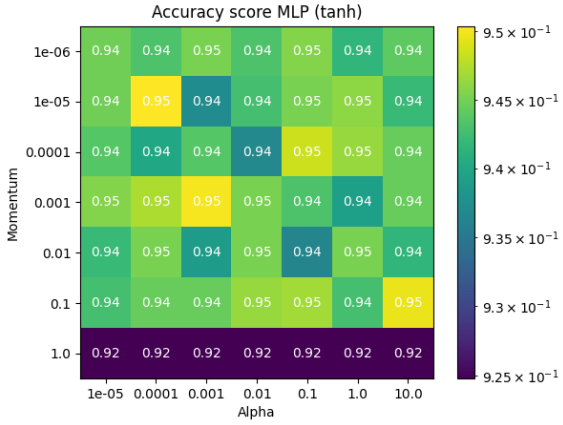

FIG. 11. Plot of accuracy made using layer sizes $[18, 15, 12, 8]$ with 500 max iterations and tanh as activation function.

Looking at figures 9, 10, and 11, we see that we can achieve an accuracy score of 95% for all three activation functions when using the best choice of hyperparameters. An accuracy score of around 95% is generally a good result. However, when we analyze our data, we see that the mortality rate of covid in the data set is 7%. This means our data set is greatly skewed, and thus, the accuracy measurement is not enough, so we have to also analyze the F1-scores.

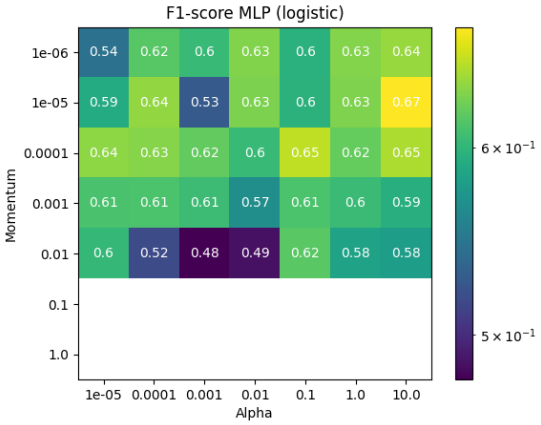

FIG. 12. Plot of F1-scores made using layer sizes $[18, 15, 12, 8]$ with 500 max iterations and logistic as activation function.

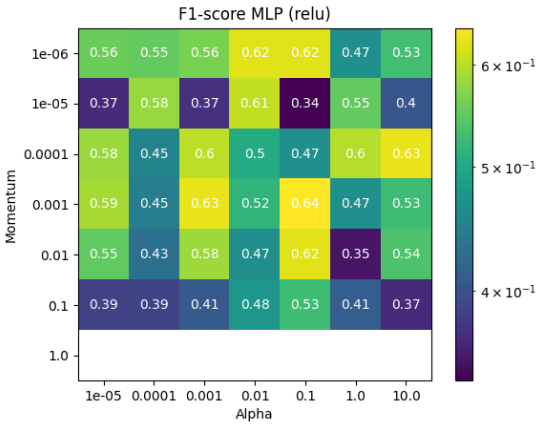

FIG. 13. Plot of F1-scores made using layer sizes $[18, 15, 12, 8]$ with 500 max iterations and relu as activation function.

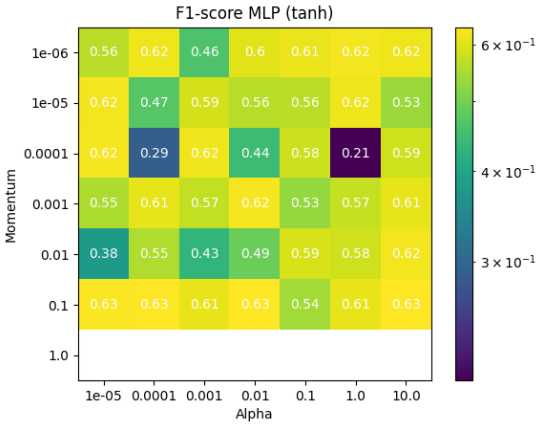

FIG. 14. Plot of F1-scores made using layer sizes $[18, 15, 12, 8]$ with 500 max iterations and tanh as activation function.

To get a good indication of which activation function performs best, we plot F1-score using the best overall hyper-parameters, with varying sample sizes:

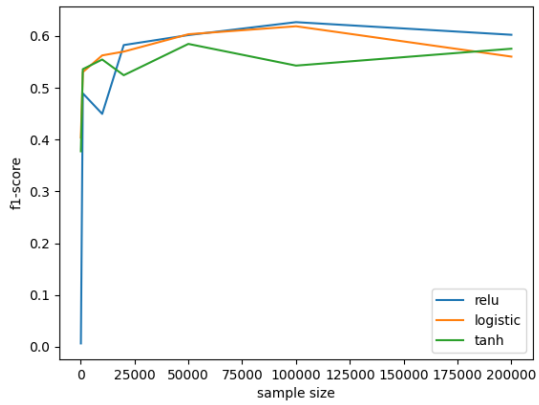

FIG. 15. Plot of F1-scores for tanh, relu and logistic activation function made using layer sizes $[18, 15, 12, 8]$ with 500 max iterations, $\alpha = 0.01$ and momentum $= 0.001$.

By analyzing the heatmaps shown in figures 12, 13, and 14, it seems as if the logistic function is the best activation function for a very low sample size, as the heatmaps were calculated using a sample size of 10000. However, when we increase the sample size, as shown in figure 15, the relu model matches the performance of the logistic model relatively quickly. With sample sizes between 25000 and 100000, these two models perform evenly, but with sample sizes higher than this, the relu model seems to be the best choice. The tanh model is generally underperforming at all sample sizes, but while the relu and logistic models are on negative trends for sample sizes larger than 100000, the tanh model is on a positive trend. Thus, there is a possibility that the tanh model is a good choice for sample sizes larger than what

we have tested.

The F1-scores, which at most reach 0.67, indicate that our MLP classifier is not perfect for predicting whether someone dies of covid or not. However, as stated above in II A, the data set consists of 21 features, not all as relevant as the others. It is also difficult to precisely predict someone's death with "only" 21 features, seeing as there is no sure way of telling if someone will die from covid or not. So an F1-score of over 0.60 is actually pretty decent in this case.

We will now look at similar plots for our forest of trees classification method and see if this performs better on the given data set.

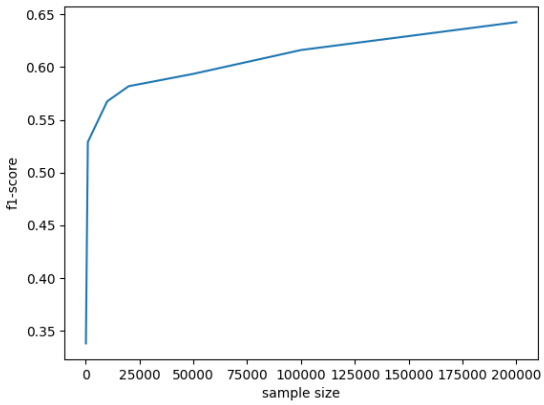

FIG. 16. Plot of F1-scores for different sample sizes for forest of trees method using scikit learns RandomForestClassifier.

If we compare figure 15 and 16, we see that the forest of trees has a slightly better outcome on the F1-scores than our MLP classifier. Also, in contrast to our MLP, it does not seem that forest of trees hits a max F1-score anytime during the plot. We tested this further using the whole data set, which resulted in a F1-score of 0.82, so it seems forest of trees does a better job at classifying using the whole data set compared to MLP. This is in accordance with the previously mentioned results achieved by Moulaei et al. [Moulaei et al. (VII), 2022].

With forest of trees, we can extract and see which features the classifier found to be most relevant:

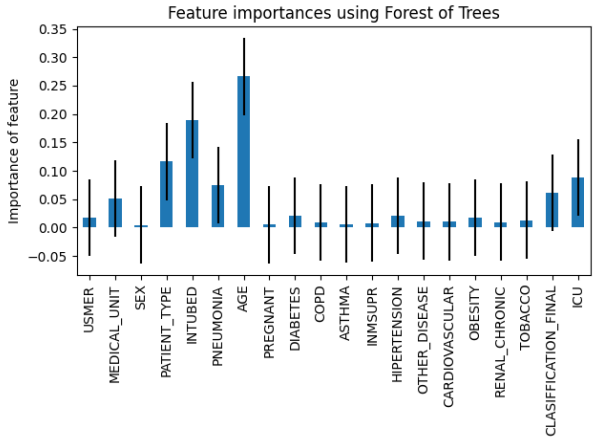

FIG. 17. Plot of feature importance for forest of trees method using scikit learns RandomForestClassifier.

In figure 17, we see that two features are dominant, age and whether or not a person was connected to a ventilator (INTUBED). Less significant, but still important features include whether a person was hospitalized or not (PATIENT_TYPE), whether a person was admitted to the intensive care unit or not (ICU), and whether or not a person had pneumonia at the time the sample was taken. None of these results are, by themselves, unexpected. People who need to be admitted to the hospital, the ICU, or a ventilator are already quite sick, and are thus more likely to die than the general case. What is interesting, however, is just how dominant the age factor is, dwarfing features such as diabetes, obesity, cardiovascular diseases, and immunosuppression (IMMSUPR), all of which are risk factors for covid-19 [FHI (VII), 2022]. This goes against what was found by Moulaei et al, who reported a higher degree of importance for ICU admission than for age, and a much closer degree of importance between the age, diabetes, hypertension, and cardiovascular disease features [Moulaei et al. (VII), 2022]. However, this difference in results is likely caused by vastly different sample sizes, as Moulaei et al. had only 1500 samples to work with. Either way, there is an agreement that age is a feature of high importance, which is also supported by other research [Flodgren et al (VII), 2020]. We will therefore further analyze this feature. We start by making a figure showing the total amount of deaths, and a figure showing the probability of death given age.

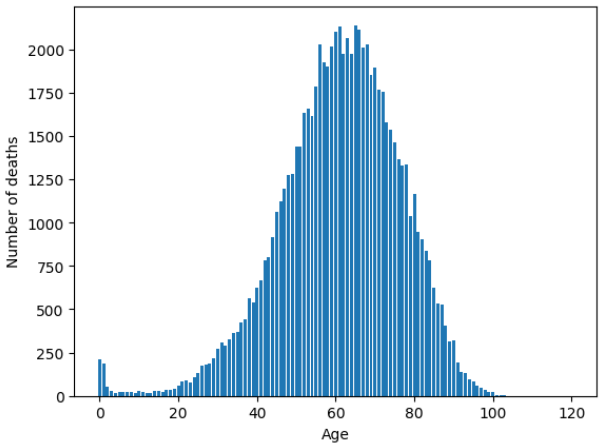

FIG. 18. Plot showing the total number of deaths at all ages.

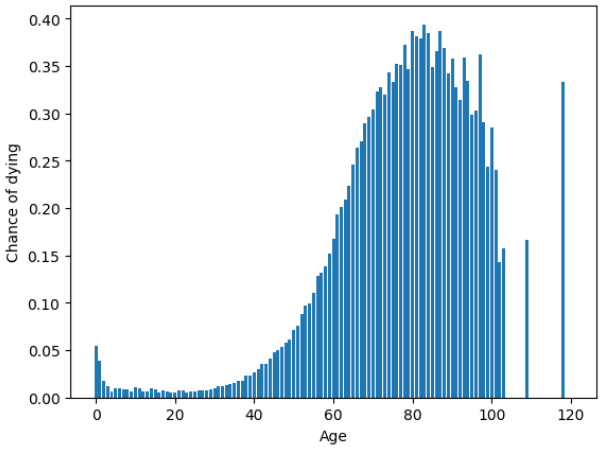

FIG. 19. Plot showing the probability of death given age.

In figure 19, we see that the probability of dying is almost 40% around age 80. The reason for such a high probability could be that most of the cases reported in this data set are for those who got serious symptoms, and not just those with mild or non-existing symptoms. However, the data set still shows that even though most deaths are centered around age groups of 60, as shown in figure 18, the probability of dying is about twice as high for those around 80 years old. The data also shows how around age 40, the death rate starts to exponentially increase, which means those above 40 are at a higher risk than the rest of the population.
This analysis gives us a great indication as to how much risk there is behind covid, and which age groups to focus on. To further study this, we will use linear regression models to see how few samples we could have and still find a probability curve good enough to predict the death rate of covid.

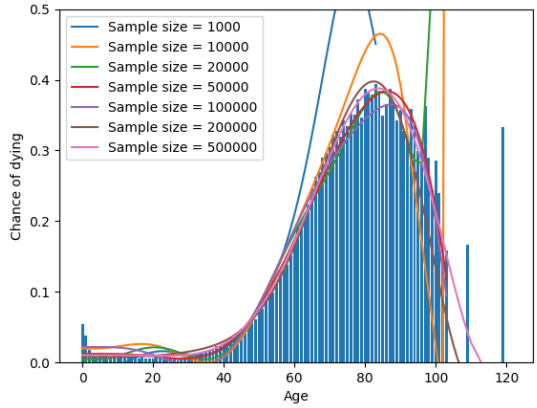

FIG. 20. Plot showing the probability of death for all ages using linear regression for multiple sample sizes. The samples were random as to make it as realistic as possible.

As we see in figure 20, the plots start to really improve and fit the data set at around 50000 samples. To further investigate, we plot the mean squared error (MSE) using bootstrapping.

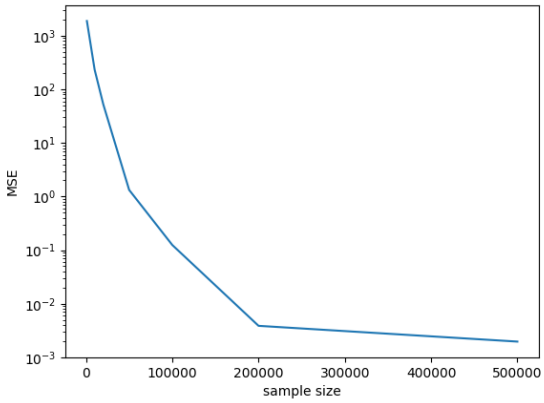

FIG. 21. Plot of MSE for our linear regression model using different random samples of different sample sizes.

In figure 21 we see that at about 200000 samples, our model is able to fit the data set more than well enough. This shows us that even though the data set consists of 1000000 different cases, we only need about 200000 to be able to accurately predict the death rate of covid based on age. For future viruses, this would be very useful to predict how dangerous the virus is, and also which age groups should be considered a risk group.
After looking into the death rate based on age, we want to combine this to examine how an underlying condition contributes to the overall death rate. Specifically, we will look at diabetes as, while not being a disease of the lungs, it is still classified as a risk group. Also, if we look at figure 17, we see that diabetes has a higher

importance than asthma, which is a respiratory disease, in our random forest model. Therefore, we made a plot of the death rate based on age for those with diabetes, and compared it to the overall death rate based on age:

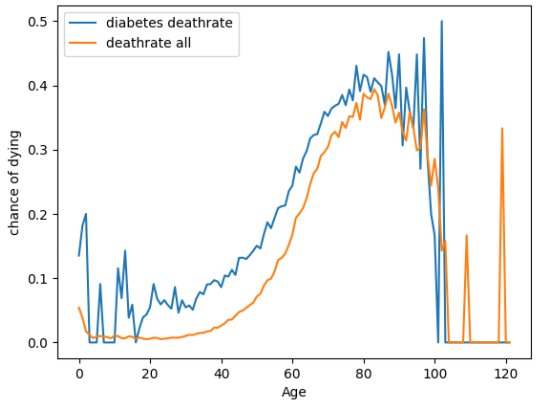

FIG. 22. Plot of chance of dying (where 1 is 100 %) for every age, comparing those with diabetes to the general population.

In figure 22, we see that the curve for diabetes cases follows a very similar trend to that of the general population, but is generally a few percentage points higher. The distance between the two curves also seems to decrease somewhat as age increases. This indicates to us that people with diabetes generally are at a higher risk from covid than those without, but that diabetes is a more serious risk factor for younger age groups.

V. CONCLUSION

We have now tested how well classifiers such as MLP and Random forest perform when predicting whether or not a person dies of covid or not. This proved to be not as effective, seeing as covid is a disease, meaning there is likely more behind whether or not a person dies than what can be represented by the 21 features we used. However, the models were able to produce somewhat decent results, and by using feature importance from Random forest, we were able to see more in-depth which factors play a bigger role in the disease. We analyzed the death rate based on age and found out that we only need about 200000 cases (samples) to be able to predict the death rate of covid well. Also adding in the death rate of those with diabetes gave us an indication that those with diabetes were indeed more at risk than the rest of the population. Using the information we gathered from this report could prove to be useful in future virus outbreaks to give us an early indication of how dangerous the virus is and who is more at risk and should be prioritized for vaccines etc. For future work, we could go more in-depth on the death rate of multiple factors and not just age and diabetes which would give us even more information on the disease.

VI. CODE

The code used in this project can be found in this Github repository.

VII. REFERENCES

Alex Feedforward Neural Networks and Multilayer Perceptrons, (BOOSTEDML, 2020).
Flodgren et al. COVID-19 and risk factors for severe disease - a rapid review, 2nd update, (FHI, 2020).
Folkehelseinstituttet Råd og informasjon til risikogrupper og pårørende, (FHI, 2022).
Folkehelseinstituttet Covid-19, influensa og andre luftveisinfeksjoner: Rapport - uke 49, (FHI, 2022).
IBM Feedforward Neural Networks and Multilayer Perceptrons, (IBM, 2020).
Moulaei, K et al. Comparing machine learning algorithms for predicting COVID-19 mortality, (BMC Medical Informatics and Decision Making, 2022).
World Health Organization WHO Coronavirus (COVID-19) Dashboard. Accessed 28. November 2022.

Appendix A: Dataset

The dataset was downloaded from: Covid-19 Dataset

Appendix B: Numerical Notation

Franke function:

$$\begin{aligned} f(x,y) = &\frac{3}{4}exp\left(\frac{(9x-2)^2}{4} - \frac{(9y-2)^2}{4}\right) \\ &+ \frac{3}{4}exp\left(-\frac{(9x+1)^2}{49} - \frac{(9y+1)}{10}\right) \\ &+ \frac{1}{2}exp\left(-\frac{(9x-7)^2}{4} - \frac{(9y-3)^2}{4}\right) \\ &- \frac{1}{5}exp\left(-(9x-4)^2 - (9y-7)^2\right) \end{aligned}$$