Katherine Wilson

COVID-19 Prediction Tool with Random Forests

Introduction

In December 2019, a strain of coronavirus broke out in Wuhan, China. Since then, COVID-19 has become a global pandemic, and nations small and large are struggling to adapt to the growing number of cases. Using open data collected from Kaggle, I develop a forecasting tool to be used at triage when new cases of COVID-19 arrive in the hospital. Because the benefits of one decision may be the costs of another, attention to forecasting accuracy and tradeoffs is given much attention.

Problem

When COVID-19 patients arrive at a hospital, decisions on how to proceed are made by the emergency room attending physician. Doctors use expertise and clinical knowledge to gauge the severity of symptoms. Based on their recommendation, patients with more likelihood of dying will receive an ICU bed or a ventilator. Patients with less likelihood will be isolated, and patients with the least risk of dying will be sent home. With precious resources like materials and isolation space in hospitals, decisions carry a lot of weight. To make matters tougher, doctors make these decisions sometimes without all of the variables that a predictive model would have the capacity to evaluate under time pressure. Can an algorithmic tool that processes the data available from accumulated COVID-19 data provide more accurate forecasts, and be used as a tool in conjunction with the doctor's expertise?

Data

Open data on Kaggle includes over 2000 observations of individuals in South Korea who attended the hospital for COVID-19 from late 2019 to early 2020.

Arguments could theoretically be made both ways on the independence of the data. On one hand, the virus affects people randomly, since it is highly contagious. More than that, the strain of the coronavirus is relatively the same. These assumptions would back up an assertion that the data is IID. On the other hand, arguments for the data not being IID point to the fact that those who are infected may infect other people in the same sample, given that they are in the same location or the same hospital. Additionally, the data at hand is not from a joint probability distribution. Instead, it is Korea's tested population, which, while substantial, has selection bias. It is not possible to know how the tested population systematically differs from the Korean population at large, so the stronger assumption here is that the data is not IID.

In order to go to a Level 2 analysis, IID data is necessary. However, random forests provide one way to proceed. Random forests address the nature of the data through sampling with replacement, random sampling of predictors, and the algorithm provides built in test data through the OOB data. Since the randomness is built into the random forest algorithm by design, then randomness is a great tool and allows us to get test data automatically. However, the confusion table only gives a sense of how well the random forest would perform at forecasting outcomes in new data. It wouldn't necessarily operate the same for new data that were not selected IID from the same joint probability distribution as the analytic sample. As noted before, it is a big assumption to make that the sample at hand represents the infected population of all of Korea, and thus using the prediction error and confusion table to forecast at triage would be specious and inappropriate. Making the previous lofty and unwarranted assumption would be the only way to generalize our findings to a broader population. Since the data are not IID, the resulting tree has limitations in that it could only be used to forecast patients' unknown outcomes when the future patient data are IID realizations from the same population that produced the training and test data.

Recoding

The original dataset contained 2119 entries of patients and 18 variables. Multiple variables contained missing values. The sex variable included 230 missing values. In order to keep the information that would be lost if list wise deletion were employed, I coded these values as "absent". Likewise, I recode the order variable with 4, the most common present order, if missing. For variables missing the age, values of 25 were imputed in missing values, since 25 was the most frequently observed age, and the provided variable does not give insight into the distribution of ages between decades. After this recoding, 2 variables remained missing from the outcome of interest, the State 3 variable. Looking into these two variables, both their State 2 values were "Lived", so I recode State 3 to released for these two observations. Limitations to this approach of recoding are that the values imputed could stray from the true values of these variables, and that something explaining the missingness could contribute to the outcomes for these observations.

In choosing the predictors for the final model, the birth year variable, which is already captured in the age variable is left out of the model. Additionally, the country variable, which only 12 values are from outside Korea is left out of the analysis data set, since, with such a small number outside of Korea, no cases may end up being sampled in the random forest. Likewise, dates of infection and symptom onset, with no knowledge of how the data were collected could be subject to much bias, and these are left out. The final random forest model includes the predictors Order, Sex, Province, Age, and confirmed date.

Target Cost Ratios

The recoded analysis data contains 32 individuals who were deceased, 1813 isolated, and 274 released from the hospital. The related proportions are 1.51% deceased, 85.60% isolated, and 12.93% released. Table 1 classifies the OOB estimates in our random forest model through a confusion matrix. The algorithm's forecasting accuracy can improve upon the previously stated forecasts at baseline that may be made by a doctor without such a tool. Additionally, the algorithm can take into account cost ratios, whereas the baseline forecasts do not take these into account. Stakeholders expressed the highest concern in releasing a patient who is the most risk of death, and hoped to make false negatives 20 times more costly than false positives. Likewise, stakeholders were next concerned about putting a patient in isolation who would actually need a ventilator. They weighted this as 10:1 false negative to false positive ratio. Other stakeholders may be concerned about too many false positives in this direction, if they

Table 1	**Deceased**	**Released**	**Isolated**	**Classification Error**
Deceased	27	0	5	0.16 (84% accuracy)
Released	10	149	115	0.46 (54% accuracy)
Isolated	105	130	1578	0.13(87% accuracy)
Forecasting Accuracy	19%	53%	92%	
randomforest(outcome ~ Sex + Age + Province + Source + Confirmed Date + Order, sampsize(21,12,26))				

are concerned about misallocating ventilator resources to people who do not need them. However, the stakeholders in this country noted that ventilators were not yet used to capacity, since many countries have been donating their old ventilators as the peak of the crisis has hit in some countries. Therefore, these stakeholders requested a 10:1 cost ratio. Finally, the stakeholders noted that the current medical systems who would use such a tool have already implemented strict shelter in place and social distancing orders, so the cost of releasing someone who was contagious, and needed to be isolated, was the least lopsided cost ratio requested, and stakeholders intended for this ratio to be 2:1.

Various cost ratios of false negatives to false positives were applied to the stratified sampling procedure with replacement. The sample of the deceased was set to 21, and after that, adjustments were made to sample the isolated and released in response. Table 1 is the confusion matrix from the OOB estimates. Because there is a large imbalance in outcome distribution (less than 2% of individuals die from the disease), then it is difficult to arrive at fully sensible cost ratios through the sampling method. Table 2 is the confusion table for deaths and released. The intent was to make a false negative death 20 more costly than a false positive. Stakeholders noted that a false negative death predicted to be a released would be the most costly decision possible, and avoided at the highest costs. The confusion table from the OOB estimates shows the random forest classification avoided all false negatives in this case. Table 3 is the confusion table for deceased and isolated. The intent was to make a false negative death 10 times as costly, and the confusion table shows a larger ratio at 20 to 1, achieved due to the high number of people isolated, and under-sampling from this parameter. Table 4 is the confusion table for the released and isolated, which was asked by stakeholders to be the situation that could be the closest to a 1:1 cost ratio. The ratio intended by stakeholders was to be a ratio of 2:1, and the actual ratio that result was around 1.2 to 1. The tradeoff makes sense, because this is the decision with the least repercussions, according to stakeholders. Stakeholders noted that failing to identify someone who should have been isolated as having been released, then they are likely to go back into their communities and be isolated anyways, with such protective measures and stay at home orders already in place.

Table 2	**Deceased**	**Released**
Deceased	27	0
Released	10	149
Intended ratio was 20:1		

Table 3	**Deceased**	**Isolated**
Deceased	27	5
Isolated	105	1578
Intended ratio was 10:1; actual is 21:1		

Table 4	**Isolated**	**Released**
Isolated	149	115
Released	130	1578
Intended ratio was 2:1; actual is 1.13:1		

Without such a tool, knowing that 1.5% of individuals die, then if a non-death is forecast, it would be correct 98% of the time and a death forecast would be correct 1.5% of the time. Our goal is to do better, because false negatives are costly, and the stakeholders (the doctors trying to save lives) will accept more false positives in order to decrease the number of false negatives. That is why the policy makers weight the cost of a false negative death so highly, their number one goal is to save lives.

The classification accuracy on the right side of the table shows that the random forest gets those who die classified correctly greater than 80% of the time, so a majority of these rare events are classified correctly. The forecasting accuracy, on the other hand, depends on the cost ratio. A forecast of deceased is correct 19% of the time. That may seem small, but it is a jump from our previous ability to forecast the deceased at 1.5%. Another way to look at this would be that we avoided the worst type of forecasting error, which would be to forecast someone who died as being released. Likewise, our ability to forecast the isolated and released raise to 92% and 53%, respectively, and both are improvements from baseline. This has the other effect of increasing the number of false positives of those who are forecasted to die, which was a tradeoff accepted by policy makers and stakeholders, whose previous forecasts, without an algorithm, did not take cost benefits into account.

Variable Importance Plots

Figure 1.1 shows the contribution of variables to the single outcome class of death. When the province variable is shuffled, classification accuracy for a death declines by around 35% points. Age, source, and confirmed are less important. The average classification importance, as opposed to importance for just the outcome of death, is shown in Figure 1.2. These importance measures are smaller because they now include categories that have more cases (released and isolated). Although confirmed date is now more important according to the reshuffling, the average overall classification accuracy is of less interest than the importance based on a single outcome. Moving back to Figure 4.1 then, policy makers may note that the variable of Province strongly contributes to the forecasting skill of the algorithm in forecasting death. This insight may go against what is currently accepted in the research on COVID-19. Many policy makers focus on age as associated with death in infected patients, but these plots show that province is more important in forecasting deaths, at least for the sample at hand. Of course, the variable importance plot does not show how an input is related to the response. The functional form is not revealed, and policy makers should understand that these analyses to not explain why or how province is an important predictor of death, simply that it improves forecasting accuracy.

Partial Plots

Unlike variable independence plots, the response functions in partial plots are made separately for each predictor and each outcome category. Partial plots demonstrate how each predictor is related to the response when the other predictors are held constant. Figure 2.1 is the partial response plot for the outcome of death and the predictor of age. Similar to the prevailing narrative and research, the plot confirms that chances of death increases with age, and policy makers, after converting these logits to probabilities, will find that age is strongly associated with death. The partial response plots for the released and isolated variables differ since only one outcome can occur at a time, though both show a similar downward trend with increasing age. Likewise, the partial plots picture in Figures 3.1-3.4 for the categorical variables, show the relationship between the input of categorical variables with the log odds of death. Policy makers may be interested in how death varies with a particular province, order, or source, and can likewise reframe these logit odds into probabilities to find out the association of the death outcome changes. For instance, the province of Deagu and the Daenam Hospital Cheongbo seem to be strongly associated with death, although policy makers are also advised to take sampling measures and the unrepresentative nature of the data into account when evaluating this.

Margins

Random forest algorithms aggregate the votes across trees for each outcome class in the spirit of bootstrap aggregation. When the vote percentages are large one way or the other, then the algorithm is classifying with high reliability (it may not be accurate, but it is reliable). However, when the vote percentages are nearly identical, little reliability is present in the outcome class. The maximum proportion of times that a case is classified correctly, and a maximum proportion of times it is classified incorrectly, are compared to find the margin. The larger the margin, the more confident the classification.

Figure 4.2 maps the margin for a deceased classification onto a histogram. A majority of the classifications have margins greater than 0.5. Very few are close to the 0.0 mark, and these that are sway towards the negative direction. Five votes in total were classified incorrectly, and two of these with reliability over 0.5. The margin here is lopsided in favor of the correct class, so despite the noise introduced by bagging resampling, most of the time the cases are classified correctly, and the classification is highly reliable in the valid (right) direction for the death outcome.

Since the purpose of our tool is for forecasting, then the stakeholders are most likely interested in the forecasting error, which show when projections are likely to be correct. In the same vein, looking at the margins for the other two outcomes (released and isolated), the distributions are less strongly skewed in the same way that the distribution for the outcome of death was. The algorithm is less reliable in these forecasts, which, as mentioned, is complemented by the forecasting accuracy. Since there is no line in the sand way to determine reliability, then again decision makers should be the ones to also decide on how decisive a vote should be for classification in respective outcomes.

Summary

Limitations of the analysis include omitted variables. One can imagine many other possible variables as predictive of the outcome that were not collected in the dataset at hand. Additionally, the random forest cannot be extended to forecasts beyond the data at hand, since the data is not IID.

Perhaps the biggest point of pause, however, is the very high cost ratio that was achieved in the outcome of deceased and isolated (Table 3). Although the policy makers asked for a high cost ratio since doctors are hoping to save as many lives as possible, this may change in the future as more and more individuals enter the hospital, and less resources are available. As resources become more scarce, then the costs of a false positive (classifying someone as high risk who actually can just be isolated) may become costlier, and the stakeholders will actually prefer to take on more false negatives in this category. In this case, the parameters can be re-sampled at different rates to achieve the desired cost ratio.

The task was to improve upon decisions that were already being made by doctors and nurses who received COVID-19 cases in the ER. Despite the difficulty in obtaining the exact cost ratios that were requested by policy officials, the model provides forecasts that are superior to baseline practice. Additionally, the model accounts for the cost ratios that were requested by policy makers, which prior baseline practices did not take into account. Such a tool could be used in conjunction with the expertise of doctors and nurses in the field for allocating COVID-19 patients proper resources.

Appendix A

Fig 1.1: Forecasting Importance Plot for Deceased

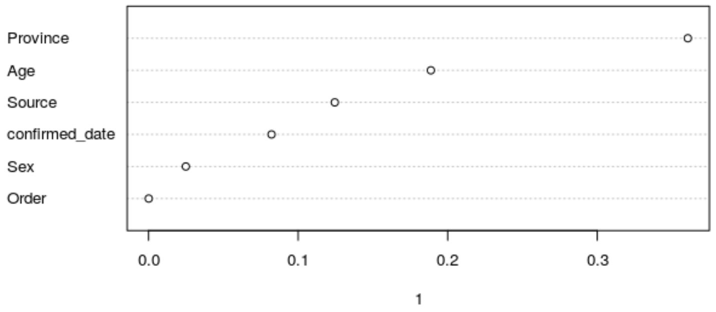


Fig 1.2: Forecasting Importance Plot Averaged for All

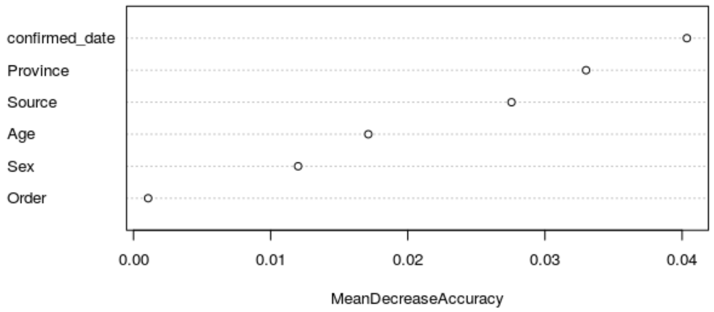


Fig 1.3: Forecasting Importance Plot for Released

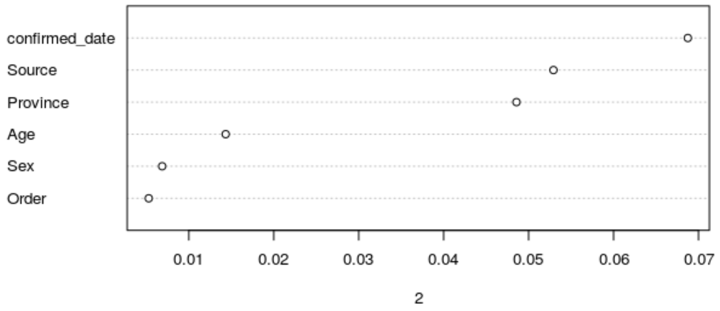


Fig 1.4: Forecasting Importance Plot for Isolated

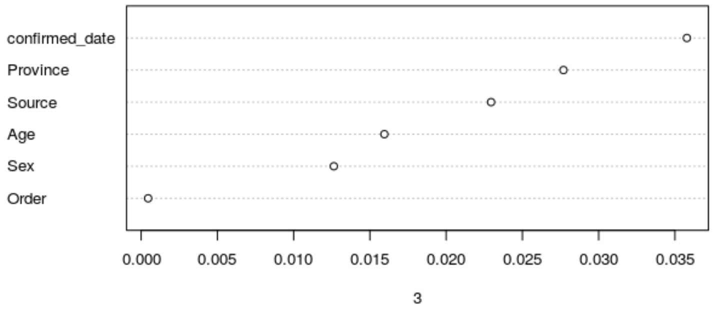


Fig 2.1: Partial Dependence Plot for Death on Age

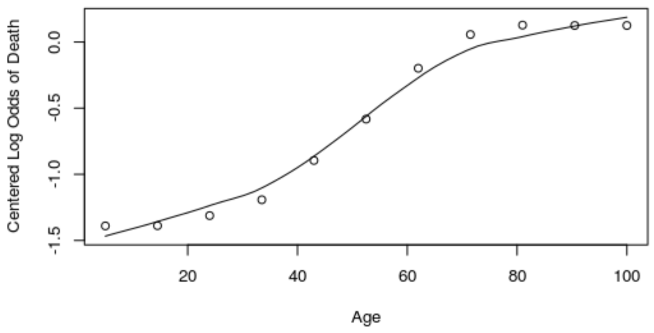


Fig 2.2: Partial Dependence Plot for Released on Age

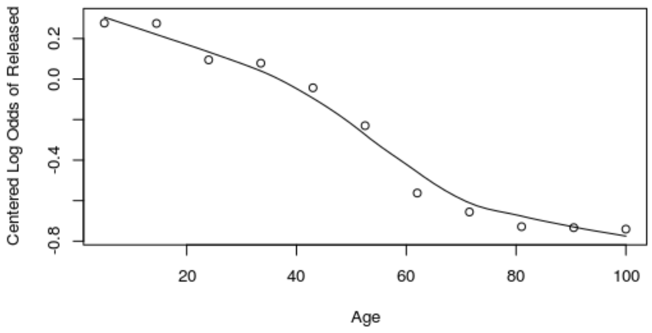


Fig 2.3: Partial Dependence Plot for Isolated on Age

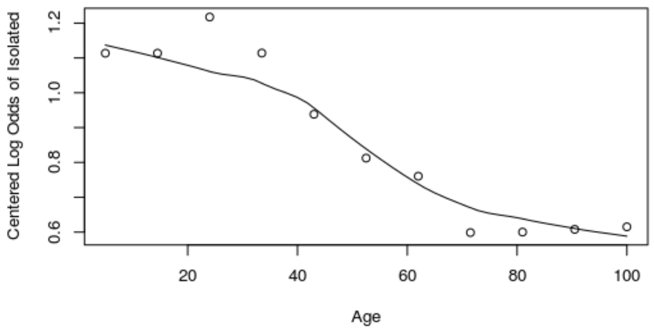


Fig 3.1: Partial Dependence Plot for Death on Sex

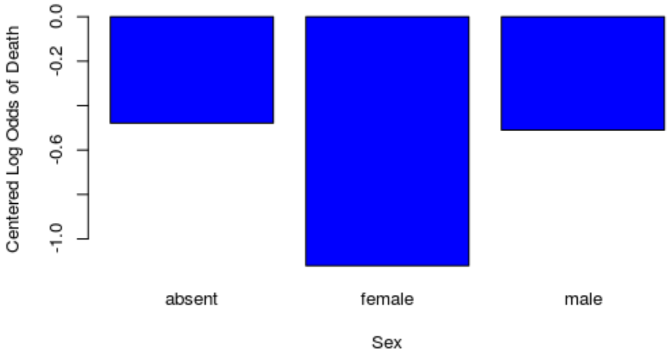


Fig 3.2: Partial Dependence Plot for Death on Order

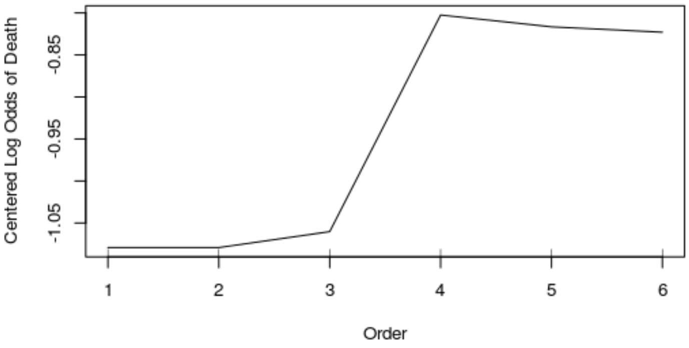


Fig 3.3: Partial Dependence Plot for Death on Province

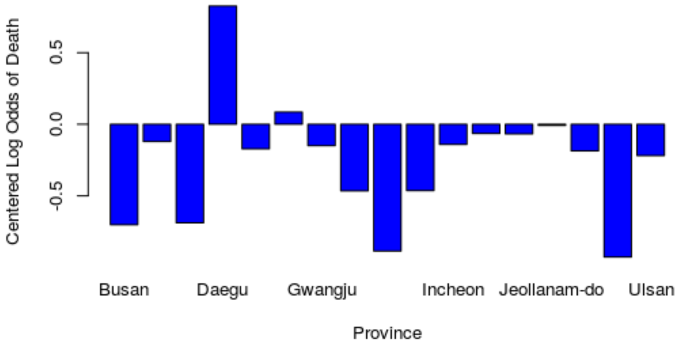


Fig 3.4: Partial Dependence Plot for Death on Source

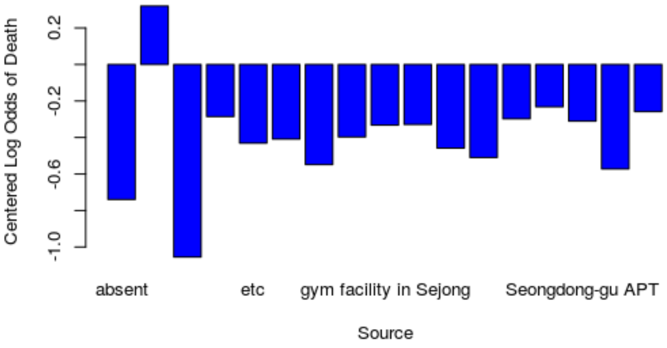


Fig 4.1: Histogram of Votes over Trees

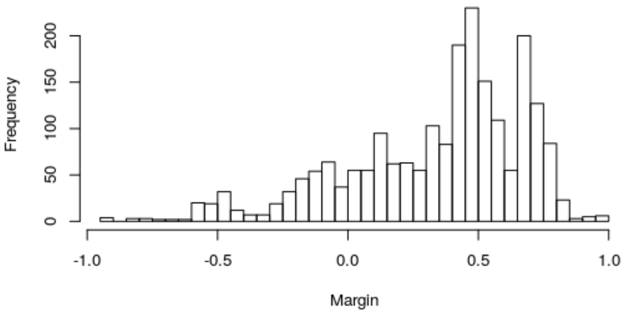


Fig 4.2: Margin for Death classification

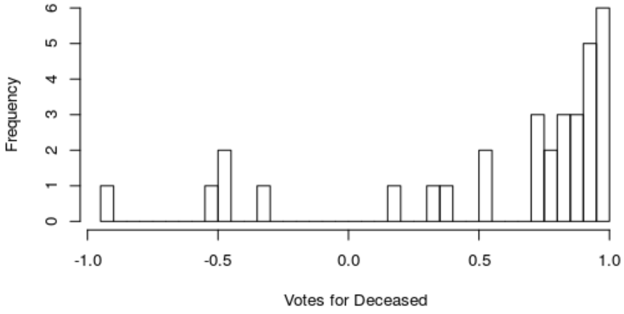


Fig 4.3: Margin for Released classification

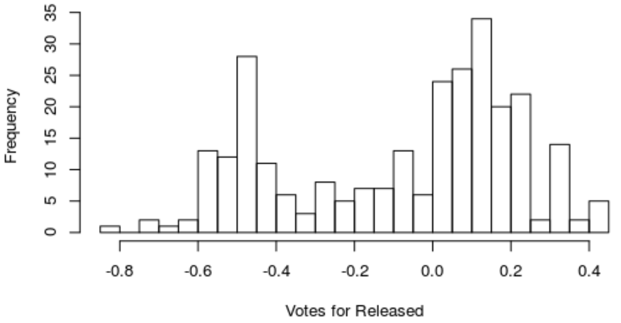
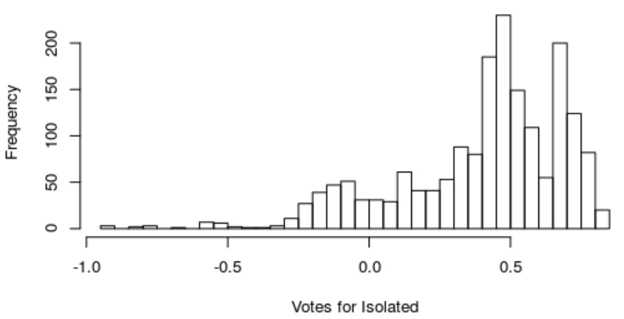


Fig 4.4: Margin for Isolated classification



Appendix B

```
###import code
install.packages("car")
install.packages("tidyverse")
install.packages("randomForest")
install.packages("pdp")
install.packages("caret")
library(tidyverse)
library(randomForest)
library(pdp)
library(caret)
library(car)

###recoding
rm(list = ls())
Patient <- read.csv("PatientInfo.csv", na.strings = c("",NA))
save(Patient, file = "Patient.rdata")
attach(Patient)
ID <- as.numeric(as.character(patient_id))
Sex <- Patient$sex
ID<-as.numeric(as.character(patient_id))
Sex <- recode(sex, "NA = 'absent'")
Age<-as.numeric(as.character(recode(age,"'0s'=5;'10s'=15;'20s'=25;'30s'=35;
                              '40s'=45;'50s'=55; '60s'=65;'70s'=75; '80s'=85;
                              '90s'=95;'100s'= 100; NA = 25")))
Province <- recode(province, "NA = 'absent'")
Source <- recode(infection_case, "NA = 'absent'")
Order<- recode(infection_order, "NA = '4'")
confirmed_date <- recode(Confirmed, "NA = '2020-02-20'")
State3 <- state
State3 <- recode(State3, "NA = 'deceased'")
State2 <- as.factor(ifelse(State3 == "deceased", "Died", "Lived"))
State2 <- recode(State2, "NA = 'Lived'")
data_new <- data.frame(ID, Sex, DOB, Age, Province, Source,
                  Order, confirmed_date, State3, State2)
data_new$confirmed_date <- as.POSIXct.Date(data_new$confirmed_date)
summary(data_new)
save(data_new, file = "work1.rdata")
detach(Patient)
coronavirus_data <- data_new
coronavirus_data$State3 <- as.factor(coronavirus_data$State3)
table(coronavirus_data$State3)
coronavirus_data <- coronavirus_data%>% mutate(
  outcome = recode(State3,
                `deceased` = 1,
                    `isolated` = 3,
                    `released` = 2)) %>%
  arrange(State3)
table(coronavirus_data$State3)
coronavirus_data[[2119,11]] <- 2
coronavirus_data[[2118,11]] <- 2
coronavirus_data$outcome <- as.factor(coronavirus_data$outcome)
save(coronavirus_data, file = "coronavirus_data.Rdata")


# Random Forest
library(randomForest)
set.seed(222)
table(coronavirus_data$outcome)
class(coronavirus_data$confirmed_date)
coronavirus_data$confirmed_date <- as.Date(coronavirus_data$confirmed_date)
coronavirus_data$outcome <- as.factor(coronavirus_data$outcome)
rf <- randomForest(outcome~Sex + Age + Province + Source +
                      confirmed_date +Order, data=coronavirus_data,
                    importance = TRUE,
                    sampsize = c(21,12,26))
print(rf)

###variable importance plots
par(mfrow = c(2,2))

varImpPlot(rf, class = 1, type = 1, scale = FALSE,
           main = "Fig 1.1: Forecasting Importance Plot for Deceased")
varImpPlot(rf, type = 1, scale = FALSE,
           main = "Fig 1.2: Forecasting Importance Plot Averaged for All")
varImpPlot(rf, class = 2, type = 1, scale = FALSE,
           main = "Fig 1.3: Forecasting Importance Plot for Released")
varImpPlot(rf, class = 3, type = 1, scale = FALSE,
           main = "Fig 1.4: Forecasting Importance Plot for Isolated")

####Partial plots
part1<- partialPlot(rf, pred.data = coronavirus_data, x.var = "Age",
                     rug = T, which.class = 1)
part2<- partialPlot(rf, pred.data = coronavirus_data, x.var = "Age",
                     rug = T, which.class = 2)
part3<- partialPlot(rf, pred.data = coronavirus_data, x.var = "Age",
                     rug = T, which.class = 3)
par(mfrow = c(2,2))

#tranform the logs back to probablity
scatter.smooth(part1$x, part1$y, xlab = "Age",
        ylab = "Centered Log Odds of Death", main = "Fig 2.1: Partial Dependence Plot for Death on
Age")
##how the other plots change based on outcome
scatter.smooth(part2$x, part2$y, xlab = "Age",
        ylab = "Centered Log Odds of Released", main = "Fig 2.2: Partial Dependence Plot for
Released on Age")
scatter.smooth(part3$x, part3$y, xlab = "Age",
        ylab = "Centered Log Odds of Isolated", main = "Fig 2.3: Partial Dependence Plot for Isolated
on Age")

par(mfrow = c(2,2))
part2 <- partialPlot(rf, pred.data = coronavirus_data,
            x.var = Sex, reg = T, prob = T,which.class = 1 ,
             main = "Fig 3.1: Partial Dependence Plot for Death on Sex",
            xlab = "Sex", ylab = "Centered Log Odds of Death")
part2
part2 <- partialPlot(rf, pred.data = coronavirus_data,
            x.var = Order, reg = T, prob = T,which.class = 1 ,
            main = "Fig 3.2: Partial Dependence Plot for Death on Order",
            xlab = "Order", ylab = "Centered Log Odds of Death")
part2
part2 <- partialPlot(rf, pred.data = coronavirus_data,
            x.var = Province, reg = T, prob = T,which.class = 1 ,
            main = "Fig 3.3: Partial Dependence Plot for Death on Province",
            xlab = "Province", ylab = "Centered Log Odds of Death")
part2
part2 <- partialPlot(rf, pred.data = coronavirus_data,
            x.var = Source, reg = T, prob = T,which.class = 1 ,
            main = "Fig 3.4: Partial Dependence Plot for Death on Source",
            xlab = "Source", ylab = "Centered Log Odds of Death")
part2
table(coronavirus_data$Source, coronavirus_data$State3)

par(mfrow = c(2,2))
m1<- randomForest::margin(rf)
m1
hist(m1, breaks = 30, main = "Fig 4.1: Histogram of Votes over Trees", xlab = "Margin")
m2 <- subset(m1, names(m1) == 1)
m2
hist(m2, breaks = 30, main = "Fig 4.2: Margin for Death classification", xlab = "Votes for Deceased")
m2
m3 <- subset(m1, names(m1) == 2)
hist(m3,breaks = 30, main = "Fig 4.3: Margin for Released classification", xlab = "Votes for Released")
m3 <- subset(m1, names(m1) == 3)
hist(m3, breaks = 30, main = "Fig 4.4: Margin for Isolated classification", xlab = "Votes for Isolated")
```