Data Mining HW3

Zeyu Yang, Blake Robert Mills

4/19/2021

#Libraries

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

## Loading required package: NLP

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## randomForest 4.6-14

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:dplyr':
##
##     combine
```

#Functions

```r
simpleCap <- function(x) {
  s <- tolower(x)
  s <- strsplit(s, " ")[[1]]
  paste(toupper(substring(s, 1,1)), substring(s, 2),
        sep="", collapse=" ")
}
```

#Files

Cleaning

```
#Gets rid of non-confirmed Covid cases in investigations
Invests <- Invests %>% filter(case_status %in% c("Confirmed", "Probable"))

#Death and Hospitalization dummy variable
Invests$DeathDummy <- ifelse(Invests$die_from_illness_ind!="Y" |
                             is.na(Invests$die_from_illness_ind)==TRUE , 0, 1)
Invests$HospDummy <- ifelse(Invests$hsptlizd_ind!="Y" |
                             is.na(Invests$hsptlizd_ind)==TRUE , 0, 1)

#Removes People not in the Jurisdiction
Verticals <- merge(Verticals, CityCounty, by.x="patient_city", by.y="City")
Verticals$specimen_collection_dt <- as.Date(Verticals$specimen_collection_dt, "%Y-%m-%d",tz="America/Ne
Verticals <- subset(Verticals, NewCounty != "Missing" & specimen_collection_dt < "2021-05-01" &
                      specimen_collection_dt >= "2020-03-01")

#Cleaning Demographic Variables
Invests$AgeNum <- as.numeric(Invests$age_calc)

#City Cleaning
Invests$CityClean <- removePunctuation(Invests$patient_city)
Invests$CityClean <- sapply(Invests$CityClean, simpleCap)
Invests <- left_join(Invests, CityCounty, by=c("CityClean" = "City"))

#Removes People out of the jurisdiction
Invests$NewCounty <- ifelse(is.na(Invests$NewCity)==TRUE, "Missing", Invests$NewCounty)
Invests <- subset(Invests, NewCounty != "Missing")

#Gender Cleaning
Invests$NewGender <- ifelse(Invests$patient_current_sex=="U", NA, Invests$patient_current_sex)

#Cleans Race Variable
Invests$NewRace <- revalue(Invests$patient_race_calc, c("American Indian or Alaska Native; White" = "Ame
                                              "Asian; Black or African American" = "Asian",
                                              "Asian; unknown" = "Asian",
                                              "Asian; White" = "Asian",
                                              "Black or African American; Native Hawaiian or Other Paci
                                              "Black or African American; unknown" = "Black or African 
                                              "Black or African American; White" = "Black or African Ame
                                              "Black or African American; White; unknown" = "Black or A
                                              "Native Hawaiian or Other Pacific Islander; White" = "Nati
                                              "not asked" = NA,
                                              "Refused to answer" = NA,
                                              "unknown" = NA,
                                              "White; unknown" = "White",
                                              "White; Other Race" = "White"))
Invests$NewRace <- ifelse(is.na(Invests$NewRace)==TRUE, "Unknown", Invests$NewRace)

#Relevels Ethnicity
Invests$ethnicity <- as.factor(Invests$ethnicity)
Invests$ethnicity <- relevel(Invests$ethnicity, ref = "Not Hispanic or Latino")
Invests$ethnicity <- ifelse(is.na(Invests$ethnicity)==TRUE, "Unknown", Invests$ethnicity)
```

```r
#Days Sick
Invests$illness_duration <- as.numeric(Invests$illness_duration)
Invests$IllnessLength <- as.Date(Invests$illness_end_dt) - as.Date(Invests$illness_onset_dt)
Invests$IllnessLength <- ifelse(Invests$symptomatic =="No", 0, Invests$IllnessLength)
```

Symptom Collapsing

```r
#Gastrointestinal Symptoms
Invests$GISymp <- ifelse(Invests$diarrhea == "Yes" |
                         Invests$abdominal_pain == "Yes",
                         1, 0)

#Ear, Nose, and Throat Symptoms
Invests$ENTSymp <- ifelse(Invests$coryza_runny_nose_ind == "Yes" |
                         Invests$loss_smell_covid == "Yes" |
                         Invests$loss_taste_smell == "Yes" |
                         Invests$sore_throat_ind == "Yes",
                         1, 0)
#Respiratory Symptoms
Invests$RespSymp <- ifelse(Invests$cough_ind == "Yes" |
                         Invests$dyspnea_ind == "Yes" |
                         Invests$wheezing_ind == "Yes",
                         1, 0)

#Systemic Symptoms
Invests$SysSymp <- ifelse(Invests$fatigue_malaise == "Yes" |
                         Invests$fever == "Yes" |
                         Invests$feverish_ind == "Yes" |
                         Invests$headache == "Yes" |
                         Invests$loss_of_appetite_ind == "Yes" |
                         Invests$nausea == "Yes",
                         1, 0)

#Muscular-Skeletal Symptoms
Invests$MuscSymp <- ifelse(Invests$chills_rigors == "Yes" |
                         Invests$myalgia == "Yes" |
                         Invests$rigors_ind == "Yes",
                         1, 0)

#Accounts for Asymptomatics
Invests$GISymp <- ifelse(is.na(Invests$GISymp)== TRUE &
                         Invests$symptomatic=="No",
                         0, Invests$GISymp)

Invests$ENTSymp <- ifelse(is.na(Invests$ENTSymp)== TRUE &
                         Invests$symptomatic=="No",
                         0, Invests$ENTSymp)

Invests$RespSymp <- ifelse(is.na(Invests$RespSymp)== TRUE &
                         Invests$symptomatic=="No",
                         0, Invests$RespSymp)

Invests$SysSymp <- ifelse(is.na(Invests$SysSymp)== TRUE &
                         Invests$symptomatic=="No",
                         0, Invests$SysSymp)

Invests$MuscSymp <- ifelse(is.na(Invests$MuscSymp)== TRUE &
                         Invests$symptomatic=="No",
                         0, Invests$MuscSymp)
```

Pre-existing Conditions Cleaning

```r
#Liver Conditions
Invests$LiverCond <- ifelse(Invests$chronic_liver_dis_ind == "Yes", 1, 0)
Invests$LiverCond <- ifelse(Invests$preexisting_cond_ind == "No", 0, Invests$LiverCond)

#Lung Conditions
Invests$LungCond <- ifelse(Invests$chronic_lung_dis_ind == "Yes", 1, 0)
Invests$LungCond <- ifelse(Invests$preexisting_cond_ind == "No", 0, Invests$LungCond)

#Renal (Kidney) Conditions
Invests$RenalCond <- ifelse(Invests$chronic_renal_dis_ind == "Yes", 1, 0)
Invests$RenalCond <- ifelse(Invests$preexisting_cond_ind == "No", 0, Invests$RenalCond)

#Cardiovascular Conditions
Invests$CVCond <- ifelse(Invests$cv_disease_ind == "Yes", 1, 0)
Invests$CVCond <- ifelse(Invests$preexisting_cond_ind == "No", 0, Invests$CVCond)

#Autoimmune Conditions
Invests$ImmunoCond <- ifelse(Invests$immuno_condition_ind == "Yes", 1, 0)
Invests$ImmunoCond <- ifelse(Invests$preexisting_cond_ind == "No", 0, Invests$ImmunoCond)

#Diabetes
Invests$Diab <- ifelse(Invests$diabetes_mellitus_ind == "Yes", 1, 0)
Invests$Diab <- ifelse(Invests$preexisting_cond_ind == "No", 0, Invests$Diab)

#Neurological or Psychiatric Conditions
Invests$NeuroCond <- ifelse(Invests$NEURO_DISABLITY_IND == "Yes" |
                            Invests$psychiatric_condition == "Yes",
                            1, 0)
Invests$NeuroCond <- ifelse(Invests$preexisting_cond_ind == "No", 0, Invests$NeuroCond)

#Substance Abuse
Invests$SubAbuse <- ifelse(Invests$substance_abuse == "Yes", 1, 0)
Invests$SubAbuse <- ifelse(Invests$preexisting_cond_ind == "No", 0, Invests$SubAbuse)

#Smoker
Invests$Smoker <- ifelse(Invests$current_smoker_ind == "Yes", 1, 0)
Invests$Smoker <- ifelse(Invests$preexisting_cond_ind == "No", 0, Invests$Smoker)

#Obesity
Invests$Obese <- ifelse(Invests$obesity_ind == "Yes", 1, 0)
Invests$Obese <- ifelse(Invests$preexisting_cond_ind == "No", 0, Invests$Obese)

#Pneumonia
Invests$Pneum <- ifelse(Invests$pneumonia == "Yes", 1, 0)
```

Random Forest

```
## The following `from` values were not present in `x`: NANA
## The following `from` values were not present in `x`: NANA
```

#Verticals Sickness in the Last 3 days

```
VerticalPos <- vector()


for(i in unique(Verticals$NewCity)){
  NewV <- Verticals %>% filter(Lab_Status=="Positive", NewCity== i) %>% dplyr::count(specimen_collectio
  NewV <- NewV %>% complete(specimen_collection_dt  = seq.Date(min(na.omit(Verticals$specimen_collectio
                                                    max(na.omit(Verticals$specimen_collection_dt)),
                                                    by="day"))
  NewV$n <- replace_na(NewV$n, 0)
  NewV$Last3Days <- rollsum(NewV$n, align="right", k=3, fill=NA)
  NewV$Sick14Days <- rollsum(NewV$n, align="right", k=14, fill=NA)
  NewV$City <- i
  NewV <- subset(NewV, specimen_collection_dt >= "2020-03-16")
  VerticalPos <- rbind(VerticalPos, NewV)
}

VerticalPos <- left_join(VerticalPos, CityCounty, by= "City")
VerticalPos$PercentSick14 <- VerticalPos$Sick14Days/VerticalPos$CityPopulation
rm(NewV)

Invests$first_pos_test <- as.Date(Invests$first_pos_test, "%Y-%m-%d")
Invests <- left_join(Invests, VerticalPos, by=c("NewCity", "first_pos_test"="specimen_collection_dt"))
```

```
#Keeps the relevant variables
InvestsComplete <- Invests %>% dplyr::select(IllnessLength, HospDummy, DeathDummy, GISymp, ENTSymp, Resp
                               MuscSymp, LiverCond, LungCond, RenalCond, CVCond, ImmunoCond, Diab,
                               NeuroCond, Smoker, SubAbuse, Obese, Pneum, AgeNum, ethnicity,
                               NewIndustry, NewRace, NewGender, PercentSick14, NewCounty.y)

#gets rid of totally incomplete cases and fills the preexisting and sypmtoms
InvestsComplete <- subset(InvestsComplete, rowSums(is.na(InvestsComplete[ , 3:19])) < 16)
InvestsComplete[ , 3:19] <- InvestsComplete[ , 3:19] %>% na.fill(0)
```

Deathmodel

```
## [1] 170
```

```
##
## Call:
## glm(formula = as.factor(DeathDummy) ~ GISymp + ENTSymp + RespSymp +
##     SysSymp + MuscSymp + LiverCond + LungCond + RenalCond + CVCond +
##     ImmunoCond + Diab + NeuroCond + Smoker + SubAbuse + Obese +
##     Pneum + AgeNum + PercentSick14 + `NewIndustry_Agriculture or Animal Husbandry` +
##     `NewIndustry_Construction and Real Estate` + `NewIndustry_Disabled, Retired, or Unemployed` +
##     `NewIndustry_Disabled/Retired/Unemployed` + `NewIndustry_Educational Services` +
##     `NewIndustry_Food or Lodging Services` + `NewIndustry_Food Production (Meat/Poultry)` +
##     `NewIndustry_Healthcare Services` + `NewIndustry_Manufacturing or Heavy Industry` +
##     `NewIndustry_Minor or Student` + `NewIndustry_Missing or did not want to disclose` +
##     `NewIndustry_Professional Services` + `NewIndustry_Public Sector` +
##     NewIndustry_Retail + `NewIndustry_Self-employed` + `NewIndustry_Social Assistance Services` +
##     `NewIndustry_Transportation and Automotive Services` + NewIndustry_Utilities,
##     family = "binomial", data = InvestsComplete)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3863  -0.0765  -0.0375   0.0000   3.8751
##
## Coefficients: (1 not defined because of singularities)
##                                                         Estimate Std. Error
## (Intercept)                                          -5.446e+11  4.390e+13
## GISymp1                                              -7.027e-01  3.122e-01
## ENTSymp1                                             -3.477e-01  2.730e-01
## RespSymp1                                             6.481e-01  3.212e-01
## SysSymp1                                              9.798e-01  3.673e-01
## MuscSymp1                                            -5.585e-01  2.741e-01
## LiverCond1                                            8.699e-01  6.235e-01
## LungCond1                                            -2.778e-01  2.990e-01
## RenalCond1                                            8.211e-01  2.946e-01
## CVCond1                                               6.803e-01  2.647e-01
## ImmunoCond1                                          -2.467e-01  3.043e-01
## Diab1                                                -3.288e-01  2.968e-01
## NeuroCond1                                            1.588e+00  2.661e-01
## Smoker1                                               7.056e-01  6.859e-01
## SubAbuse1                                             7.620e-01  7.580e-01
## Obese1                                                8.038e-01  3.814e-01
## Pneum1                                                2.993e+00  2.782e-01
## AgeNum                                                1.414e+00  2.405e-01
## PercentSick14                                         6.694e+00  9.920e+00
## `NewIndustry_Agriculture or Animal Husbandry`         5.446e+11  4.390e+13
## `NewIndustry_Construction and Real Estate`            5.446e+11  4.390e+13
## `NewIndustry_Disabled, Retired, or Unemployed`        5.446e+11  4.390e+13
## `NewIndustry_Disabled/Retired/Unemployed`                    NA         NA
## `NewIndustry_Educational Services`                    5.446e+11  4.390e+13
## `NewIndustry_Food or Lodging Services`                5.446e+11  4.390e+13
## `NewIndustry_Food Production (Meat/Poultry)`          5.446e+11  4.390e+13
## `NewIndustry_Healthcare Services`                     5.446e+11  4.390e+13
## `NewIndustry_Manufacturing or Heavy Industry`         5.446e+11  4.390e+13
## `NewIndustry_Minor or Student`                        5.446e+11  4.390e+13
## `NewIndustry_Missing or did not want to disclose`     5.446e+11  4.390e+13
## `NewIndustry_Professional Services`                   5.446e+11  4.390e+13
## `NewIndustry_Public Sector`                           5.446e+11  4.390e+13
## NewIndustry_Retail                                    5.446e+11  4.390e+13
## `NewIndustry_Self-employed`                           5.446e+11  4.390e+13
## `NewIndustry_Social Assistance Services`              5.446e+11  4.390e+13
## `NewIndustry_Transportation and Automotive Services`  5.446e+11  4.390e+13
## NewIndustry_Utilities                                 5.446e+11  4.390e+13
##                                                      z value Pr(>|z|)
## (Intercept)                                           -0.012  0.99010
## GISymp1                                               -2.251  0.02439 *
## ENTSymp1                                              -1.274  0.20281
## RespSymp1                                              2.018  0.04362 *
## SysSymp1                                               2.667  0.00764 **
## MuscSymp1                                             -2.037  0.04162 *
## LiverCond1                                             1.395  0.16295
## LungCond1                                             -0.929  0.35277
## RenalCond1                                             2.787  0.00532 **
## CVCond1                                                2.570  0.01016 *
## ImmunoCond1                                           -0.811  0.41754
## Diab1                                                 -1.108  0.26803
## NeuroCond1                                             5.970 2.38e-09 ***
## Smoker1                                                1.029  0.30362
## SubAbuse1                                              1.005  0.31478
## Obese1                                                 2.108  0.03506 *
## Pneum1                                                10.760  < 2e-16 ***
## AgeNum                                                 5.879 4.13e-09 ***
## PercentSick14                                          0.675  0.49976
## `NewIndustry_Agriculture or Animal Husbandry`          0.012  0.99010
## `NewIndustry_Construction and Real Estate`             0.012  0.99010
## `NewIndustry_Disabled, Retired, or Unemployed`         0.012  0.99010
## `NewIndustry_Disabled/Retired/Unemployed`                 NA       NA
## `NewIndustry_Educational Services`                     0.012  0.99010
## `NewIndustry_Food or Lodging Services`                 0.012  0.99010
## `NewIndustry_Food Production (Meat/Poultry)`           0.012  0.99010
## `NewIndustry_Healthcare Services`                      0.012  0.99010
## `NewIndustry_Manufacturing or Heavy Industry`          0.012  0.99010
## `NewIndustry_Minor or Student`                         0.012  0.99010
## `NewIndustry_Missing or did not want to disclose`      0.012  0.99010
## `NewIndustry_Professional Services`                    0.012  0.99010
## `NewIndustry_Public Sector`                            0.012  0.99010
## NewIndustry_Retail                                     0.012  0.99010
## `NewIndustry_Self-employed`                            0.012  0.99010
## `NewIndustry_Social Assistance Services`               0.012  0.99010
## `NewIndustry_Transportation and Automotive Services`   0.012  0.99010
## NewIndustry_Utilities                                  0.012  0.99010
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1339.99  on 7169  degrees of freedom
## Residual deviance:  563.16  on 7134  degrees of freedom
## AIC: 635.16
##
## Number of Fisher Scoring iterations: 25
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
##
## Call:
## glm(formula = as.factor(DeathDummy) ~ GISymp + RespSymp + SysSymp +
##     MuscSymp + RenalCond + CVCond + NeuroCond + Obese + Pneum +
##     AgeNum + `NewIndustry_Healthcare Services` + `NewIndustry_Missing or did not want to disclose` +
##     NewIndustry_Retail, family = "binomial", data = InvestsComplete)
```

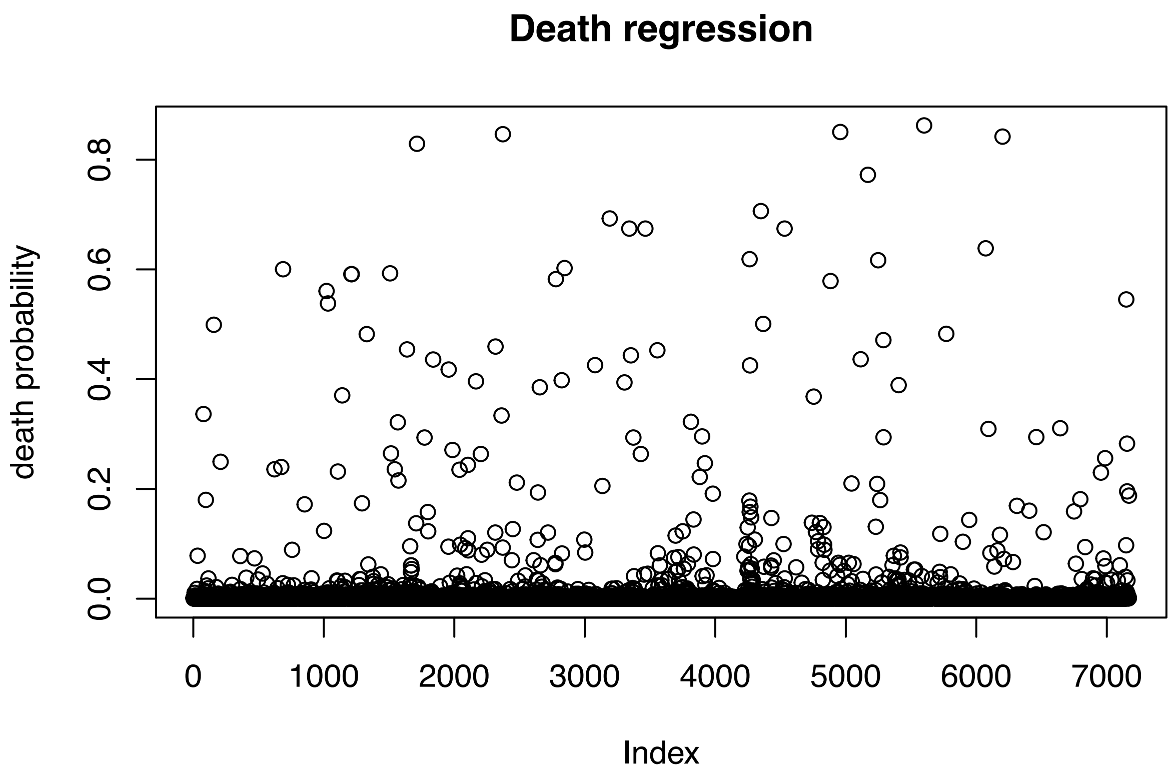
```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6021  -0.0820  -0.0405  -0.0213   4.1252
##
## Coefficients:
##                                                   Estimate Std. Error z value
## (Intercept)                                        -7.1786     0.4447 -16.143
## GISymp1                                            -0.7581     0.3020  -2.510
## RespSymp1                                           0.5705     0.3100   1.840
## SysSymp1                                            0.9500     0.3601   2.638
## MuscSymp1                                          -0.6305     0.2630  -2.398
## RenalCond1                                          0.7136     0.2709   2.634
## CVCond1                                             0.6601     0.2541   2.598
## NeuroCond1                                          1.5972     0.2501   6.385
## Obese1                                              0.5643     0.3541   1.594
## Pneum1                                              2.9818     0.2620  11.379
## AgeNum                                              1.2830     0.1866   6.877
## `NewIndustry_Healthcare Services`                 -14.8210   595.7053  -0.025
## `NewIndustry_Missing or did not want to disclose`   0.6735     0.3834   1.757
## NewIndustry_Retail                                -14.7452   873.2662  -0.017
##                                                   Pr(>|z|)
## (Intercept)                                        < 2e-16 ***
## GISymp1                                            0.01207 *
## RespSymp1                                          0.06576 .
## SysSymp1                                           0.00833 **
## MuscSymp1                                          0.01650 *
## RenalCond1                                         0.00843 **
## CVCond1                                            0.00937 **
## NeuroCond1                                        1.71e-10 ***
## Obese1                                             0.11102
## Pneum1                                             < 2e-16 ***
## AgeNum                                            6.09e-12 ***
## `NewIndustry_Healthcare Services`                  0.98015
## `NewIndustry_Missing or did not want to disclose`  0.07898 .
## NewIndustry_Retail                                 0.98653
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1339.99  on 7169  degrees of freedom
## Residual deviance:  580.33  on 7156  degrees of freedom
## AIC: 608.33
##
## Number of Fisher Scoring iterations: 19
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
## Loaded glmnet 4.1

## Loading required package: lattice

## Loading required package: ggplot2

##
## Attaching package: 'ggplot2'

## The following object is masked from 'package:randomForest':
##
##     margin

## The following object is masked from 'package:NLP':
##
##     annotate

## [1] "cross validation fold 1"
## [1] "variables used in model: RespSymp\nSysSymp\nMuscSymp\nRenalCond\nCVCond\nNeuroCond\nPneum\nAgeNu
## [1] "Probability higher than  0.091  is consider dead"
## [1] "Recall : 0.827586206896552"
## [1] "Precision : 0.452830188679245"
## [1] "cross validation fold 2"
## [1] "variables used in model: GISymp\nSysSymp\nMuscSymp\nRenalCond\nCVCond\nNeuroCond\nPneum\nAgeNum
## [1] "Probability higher than  0.041  is consider dead"
## [1] "Recall : 0.814814814814815"
## [1] "Precision : 0.366666666666667"
## [1] "cross validation fold 3"
## [1] "variables used in model: GISymp\nSysSymp\nRenalCond\nNeuroCond\nPneum\nAgeNum\nNewIndustry_Missi
## [1] "Probability higher than  0.036  is consider dead"
## [1] "Recall : 0.80952380952381"
## [1] "Precision : 0.293103448275862"
## [1] "cross validation fold 4"
## [1] "variables used in model: GISymp\nSysSymp\nMuscSymp\nCVCond\nNeuroCond\nPneum\nAgeNum"
## [1] "Probability higher than  0.017  is consider dead"
## [1] "Recall : 0.848484848484849"
## [1] "Precision : 0.325581395348837"
## [1] "cross validation fold 5"
## [1] "variables used in model: MuscSymp\nRenalCond\nNeuroCond\nObese\nPneum\nAgeNum"
## [1] "Probability higher than  0.024  is consider dead"
## [1] "Recall : 0.84"
## [1] "Precision : 0.276315789473684"

##
## Call:
## glm(formula = as.factor(DeathDummy) ~ RespSymp + MuscSymp + SysSymp +
##     RenalCond + CVCond + NeuroCond + Pneum + `NewIndustry_Missing or did not want to disclose` +
##     AgeNum, family = "binomial", data = InvestsComplete)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6017  -0.0842  -0.0451  -0.0260   4.2331
##
## Coefficients:
##                                                   Estimate Std. Error z value
## (Intercept)                                        -7.4095     0.4458 -16.621
## RespSymp1                                           0.5265     0.3078   1.711
## MuscSymp1                                          -0.7212     0.2594  -2.780
## SysSymp1                                            0.8805     0.3579   2.460
## RenalCond1                                          0.7997     0.2685   2.978
## CVCond1                                             0.6569     0.2510   2.617
## NeuroCond1                                          1.6708     0.2451   6.818
## Pneum1                                              2.9667     0.2544  11.663
## `NewIndustry_Missing or did not want to disclose`   0.7890     0.3799   2.077
## AgeNum                                              1.3678     0.1812   7.549
##                                                   Pr(>|z|)    
## (Intercept)                                         < 2e-16 ***
## RespSymp1                                           0.08716 .  
## MuscSymp1                                           0.00543 ** 
## SysSymp1                                            0.01389 *  
## RenalCond1                                          0.00290 ** 
## CVCond1                                             0.00887 ** 
## NeuroCond1                                         9.25e-12 ***
## Pneum1                                              < 2e-16 ***
## `NewIndustry_Missing or did not want to disclose`  0.03781 *  
## AgeNum                                             4.39e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1339.99  on 7169  degrees of freedom
## Residual deviance:  596.75  on 7160  degrees of freedom
## AIC: 616.75
##
## Number of Fisher Scoring iterations: 9
```

```
##    recall precision    cutoff
## 0.8148148 0.2872063 0.0140000
```



Hospitalization Model

```
#Hospitalization Regression
HospModel <- glm(as.factor(HospDummy) ~
             #Symptoms
              GISymp + ENTSymp + RespSymp + SysSymp + MuscSymp  +

             #Pre-Existing
             LiverCond + LungCond + RenalCond + CVCond + ImmunoCond + Diab + NeuroCond +
             Smoker + SubAbuse + Obese +

             #After Effects
             Pneum +

             #Demographics
             AgeNum + PercentSick14 + NewIndustry,

             # Industries patient work in



             data= InvestsComplete, family= "binomial", maxit = 100)

summary(HospModel)
```

```
##
## Call:
## glm(formula = as.factor(HospDummy) ~ GISymp + ENTSymp + RespSymp +
##     SysSymp + MuscSymp + LiverCond + LungCond + RenalCond + CVCond +
##     ImmunoCond + Diab + NeuroCond + Smoker + SubAbuse + Obese +
##     Pneum + AgeNum + PercentSick14 + NewIndustry, family = "binomial",
##     data = InvestsComplete, maxit = 100)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0231  -0.2549  -0.1605  -0.1083   3.4525
##
## Coefficients:
##                                                     Estimate Std. Error z value
## (Intercept)                                         -4.23788    0.34592 -12.251
## GISymp1                                              0.49147    0.14774   3.327
## ENTSymp1                                            -0.82830    0.15491  -5.347
## RespSymp1                                            0.61142    0.18022   3.393
## SysSymp1                                             0.39885    0.21862   1.824
## MuscSymp1                                            0.03238    0.15996   0.202
## LiverCond1                                           0.33899    0.51868   0.654
## LungCond1                                            0.16041    0.18771   0.855
## RenalCond1                                           0.27389    0.23983   1.142
## CVCond1                                              0.37628    0.16464   2.285
## ImmunoCond1                                          0.16781    0.22723   0.738
## Diab1                                                0.57357    0.15916   3.604
## NeuroCond1                                           0.09625    0.19754   0.487
## Smoker1                                             -0.86479    0.53137  -1.627
## SubAbuse1                                           -0.03122    0.62831  -0.050
## Obese1                                               0.52197    0.27643   1.888
## Pneum1                                               4.04367    0.25894  15.616
## AgeNum                                               0.78548    0.10350   7.589
## PercentSick14                                        3.25898    5.50816   0.592
## NewIndustryConstruction and Real Estate              0.57775    0.49409   1.169
## NewIndustryDisabled, Retired, or Unemployed          0.20576    0.31505   0.653
## NewIndustryEducational Services                     -0.29445    0.44273  -0.665
## NewIndustryFood or Lodging Services                  0.26372    0.53382   0.494
## NewIndustryFood Production (Meat/Poultry)            0.10276    0.36993   0.278
## NewIndustryHealthcare Services                      -0.72298    0.43671  -1.655
## NewIndustryManufacturing or Heavy Industry           0.17534    0.45046   0.389
## NewIndustryMinor or Student                         -0.09847    0.54756  -0.180
## NewIndustryMissing or did not want to disclose       0.42255    0.35795   1.180
## NewIndustryProfessional Services                    -0.56930    0.45507  -1.251
## NewIndustryPublic Sector                            -1.00128    0.63887  -1.567
## NewIndustryRetail                                   -0.25170    0.47511  -0.530
## NewIndustrySelf-employed                             0.08229    0.62115   0.132
## NewIndustrySocial Assistance Services               -0.67898    0.61832  -1.098
## NewIndustryTransportation and Automotive Services  -0.37378    0.55500  -0.673
## NewIndustryUtilities                               -12.74441  276.48158  -0.046
##                                                    Pr(>|z|)
## (Intercept)                                         < 2e-16 ***
## GISymp1                                            0.000879 ***
## ENTSymp1                                           8.94e-08 ***
## RespSymp1                                          0.000692 ***
## SysSymp1                                           0.068093 .
## MuscSymp1                                          0.839564
## LiverCond1                                         0.513393
## LungCond1                                          0.392778
## RenalCond1                                         0.253432
## CVCond1                                           0.022288 *  
## ImmunoCond1                                       0.460215    
## Diab1                                             0.000314 ***
## NeuroCond1                                        0.626083    
## Smoker1                                           0.103632    
## SubAbuse1                                         0.960367    
## Obese1                                            0.058989 .  
## Pneum1                                             < 2e-16 ***
## AgeNum                                            3.22e-14 ***
## PercentSick14                                     0.554075    
## NewIndustryConstruction and Real Estate           0.242271    
## NewIndustryDisabled, Retired, or Unemployed       0.513705    
## NewIndustryEducational Services                   0.506003    
## NewIndustryFood or Lodging Services               0.621282    
## NewIndustryFood Production (Meat/Poultry)         0.781178    
## NewIndustryHealthcare Services                    0.097826 .  
## NewIndustryManufacturing or Heavy Industry        0.697088    
## NewIndustryMinor or Student                       0.857285    
## NewIndustryMissing or did not want to disclose    0.237822    
## NewIndustryProfessional Services                  0.210923    
## NewIndustryPublic Sector                          0.117051    
## NewIndustryRetail                                 0.596277    
## NewIndustrySelf-employed                          0.894602    
## NewIndustrySocial Assistance Services             0.272161    
## NewIndustryTransportation and Automotive Services 0.500638    
## NewIndustryUtilities                              0.963235    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 3192.8  on 7169  degrees of freedom
## Residual deviance: 1926.5  on 7135  degrees of freedom
## AIC: 1996.5
## 
## Number of Fisher Scoring iterations: 15
```

```
step.HospModel <- HospModel %>% stepAIC(trace = FALSE)
summary(step.HospModel)
```

```
## 
## Call:
## glm(formula = as.factor(HospDummy) ~ GISymp + ENTSymp + RespSymp + 
##     SysSymp + RenalCond + CVCond + Diab + Smoker + Obese + Pneum + 
##     AgeNum, family = "binomial", data = InvestsComplete, maxit = 100)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -3.0018  -0.2644  -0.1644  -0.1100   3.5494  
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept) -4.18295    0.16398 -25.509  < 2e-16 ***
## GISymp1      0.49581    0.14388   3.446 0.000569 ***
## ENTSymp1    -0.86703    0.14933  -5.806 6.39e-09 ***
## RespSymp1    0.58304    0.17603   3.312 0.000926 ***
## SysSymp1     0.35353    0.20228   1.748 0.080516 .
## RenalCond1   0.42015    0.22798   1.843 0.065348 .
## CVCond1      0.42784    0.16056   2.665 0.007704 **
## Diab1        0.61529    0.15702   3.918 8.91e-05 ***
## Smoker1     -0.79458    0.50543  -1.572 0.115932
## Obese1       0.55247    0.26501   2.085 0.037099 *
## Pneum1       4.09609    0.25031  16.364  < 2e-16 ***
## AgeNum       0.92871    0.07924  11.720  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3192.8  on 7169  degrees of freedom
## Residual deviance: 1957.3  on 7158  degrees of freedom
## AIC: 1981.3
##
## Number of Fisher Scoring iterations: 7
```

```
# collinerity check
## the VIF is relatively low with around 1-2 for all variable selected
Hosp_VIF <- car::vif((step.HospModel))
Hosp_VIF <- Hosp_VIF[which(Hosp_VIF<10)]


### with cross validation
Metrics2 = matrix(NA,nrow = 5, ncol = 3)
colnames(Metrics2) <- c("Recall", "Precision", "cutoff")
Logit_out2 <- list()
for(i in seq_len(k)){

    #code that isolates the test/train data!
    is_test = test_folds == i
    is_train = !is_test
    # fitting models and counting time
    print(paste("cross validation fold", i))
    t0 <- Sys.time()
    HospModel2 <- glm(as.factor(DeathDummy) ~
             #Symptoms
             #NewIndustry +
               GISymp +ENTSymp + RespSymp + SysSymp +

             #Pre-Existing
             RenalCond + CVCond + Diab + Smoker + Obese +

             #After Effects
             Pneum +

             #Demographics
             AgeNum + PercentSick14,

             data = InvestsComplete[is_train,], family = "binomial")
    #print(paste('running Logit regression on hospitalization took', Sys.time() - t0))
    # prediction metirc

    Hmodel_coef <- coef(HospModel2)[which(summary(HospModel2)$coefficients[,4]<0.05)]

    pos <- match(names(InvestsComplete),stringr::str_remove(names(Hmodel_coef)[-1],"1"))
    x <- InvestsComplete[,which(!is.na(pos) == TRUE)]
    print(paste("variables used in model:",NLP::as.String(names(x))))


    x <- x[is_test,]
    Actual <- InvestsComplete[is_test,"HospDummy"]
    Actual <- apply(Actual,2,as.numeric)
    Y <- Hmodel_coef[1]+ apply(x,2,as.numeric) %*% Hmodel_coef[2:length(Hmodel_coef)]
    p_hat <- exp(Y)/(1+exp(Y))
    Logit_out2[[i]] <- p_hat
    ## we want recall higher than 0.8 so we have 80% of the hospitalization recognized

    Rec <- c()
    Prec <- c()
    F1 <- c()
    for (j in seq(1,1000,by = 1)){
      cutoff <- 1 - 0.001*j
      Rec[j] <- ModelMetrics::recall(Actual, p_hat, cutoff = cutoff)
      Prec[j] <- ModelMetrics::precision(Actual, p_hat, cutoff = cutoff)
      F1[j] <- 2*Rec[j]*Prec[j]/(Rec[j]+Prec[j])
    }
    ## Maximize F1 score since we care both recall and precision

    F1 <- na.fill.default(F1,0)
    Best_prec <- max(na.omit(F1))
    Best_score_ind <- Position(function(x) x == Best_prec, F1)
    cutoff <- 1 - 0.001*Best_score_ind
    Metrics2[i,1] <- Rec[Best_score_ind]
    Metrics2[i,2] <- Prec[Best_score_ind]
    Metrics2[i,3] <- cutoff

    print(paste("Probability higher than ", round(cutoff,4), " is consider hosipitalized"))
    print(paste("F1 :", F1[Best_score_ind]))
    print(paste("Recall :", Rec[Best_score_ind]))
    print(paste("Precision :", Prec[Best_score_ind]))

}
```

```
## [1] "cross validation fold 1"
## [1] "variables used in model: ENTSymp\nRenalCond\nCVCond\nPneum\nAgeNum"
## [1] "Probability higher than  0.021  is consider hosipitalized"
## [1] "F1 : 0.547619047619048"
## [1] "Recall : 0.55421686746988"
## [1] "Precision : 0.541176470588235"
## [1] "cross validation fold 2"
## [1] "variables used in model: GISymp\nENTSymp\nSysSymp\nRenalCond\nCVCond\nObese\nPneum\nAgeNum"
## [1] "Probability higher than  0.016  is consider hosipitalized"
## [1] "F1 : 0.475247524752475"
## [1] "Recall : 0.571428571428571"
## [1] "Precision : 0.406779661016949"
## [1] "cross validation fold 3"
## [1] "variables used in model: GISymp\nENTSymp\nSysSymp\nRenalCond\nCVCond\nObese\nPneum\nAgeNum"
## [1] "Probability higher than  0.068  is consider hosipitalized"
## [1] "F1 : 0.496240601503759"
## [1] "Recall : 0.392857142857143"
## [1] "Precision : 0.673469387755102"
## [1] "cross validation fold 4"
## [1] "variables used in model: GISymp\nENTSymp\nRenalCond\nCVCond\nObese\nPneum\nAgeNum"
## [1] "Probability higher than  0.034  is consider hosipitalized"
## [1] "F1 : 0.482269503546099"
## [1] "Recall : 0.404761904761905"
## [1] "Precision : 0.596491228070175"
## [1] "cross validation fold 5"
## [1] "variables used in model: GISymp\nENTSymp\nRenalCond\nSmoker\nObese\nPneum\nAgeNum"
## [1] "Probability higher than  0.017  is consider hosipitalized"
## [1] "F1 : 0.451282051282051"
## [1] "Recall : 0.523809523809524"
## [1] "Precision : 0.396396396396396"
```

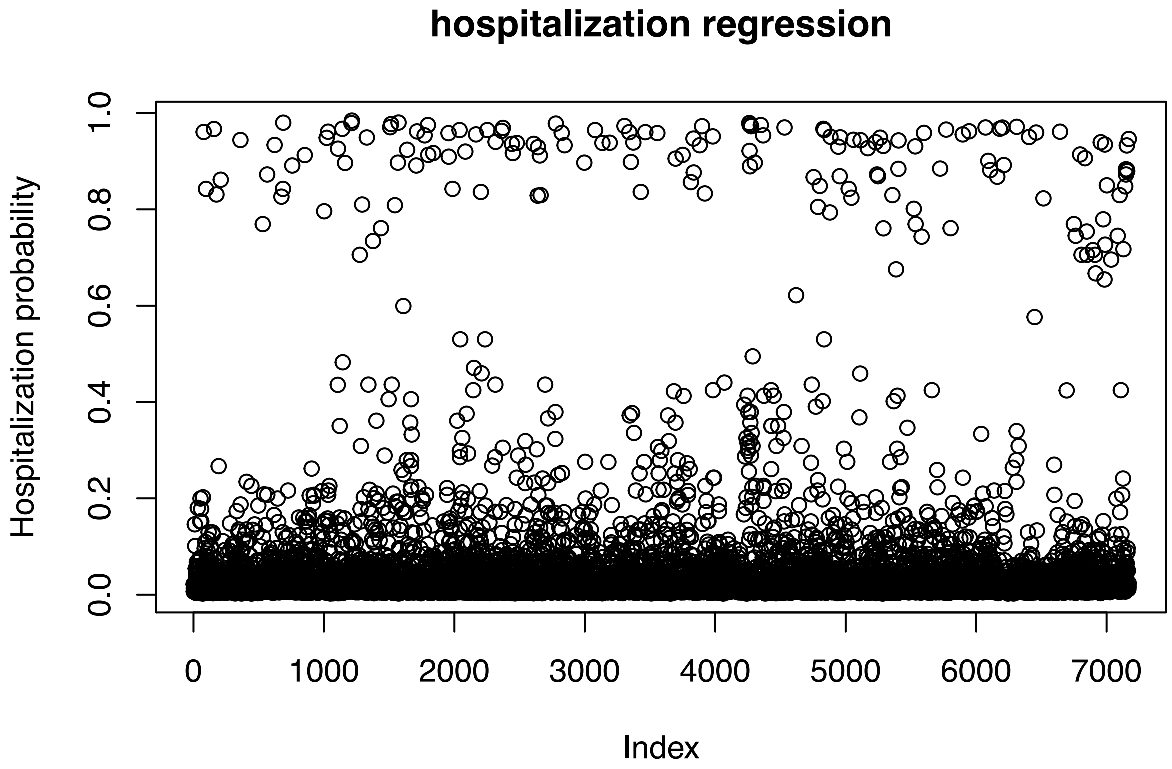
```
## best was from cross validation fold 1: "F1 =  0.547619047619048"
## [1] "variables used in model: ENTSymp\nRenalCond\nCVCond\nPneum\nAgeNum"

final_HospModel <- glm(as.factor(HospDummy) ~
             #Symptoms
             #NewIndustry +
             ENTSymp +

             #Pre-Existing
             RenalCond + CVCond +

             #After Effects
             Pneum +

             #Demographics
             AgeNum,

             data = InvestsComplete, family = "binomial")
summary(final_HospModel)
```

```
##
## Call:
## glm(formula = as.factor(HospDummy) ~ ENTSymp + RenalCond + CVCond +
##     Pneum + AgeNum, family = "binomial", data = InvestsComplete)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8015  -0.2786  -0.1820  -0.1252   3.4583
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.67343    0.11873 -30.939  < 2e-16 ***
## ENTSymp1    -0.29275    0.12786  -2.289   0.0221 *
## RenalCond1   0.54955    0.22326   2.462   0.0138 *
## CVCond1      0.62909    0.15838   3.972 7.13e-05 ***
## Pneum1       4.25526    0.24829  17.139  < 2e-16 ***
## AgeNum       0.99897    0.07617  13.115  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3192.8  on 7169  degrees of freedom
## Residual deviance: 2034.2  on 7164  degrees of freedom
## AIC: 2046.2
##
## Number of Fisher Scoring iterations: 7
```

```
pos <- match(names(InvestsComplete),stringr::str_remove(names(coef(final_HospModel))[-1],"1"))
Hosp_x <- InvestsComplete[,which(!is.na(pos) == TRUE)]

Actual <- InvestsComplete[,"HospDummy"]
Actual <- apply(Actual,2,as.numeric)
Hosp_Y <- coef(final_HospModel)[1]+ apply(Hosp_x,2,as.numeric) %*% coef(final_HospModel)[2:length(coef(
Hosp_p_hat <- exp(Hosp_Y)/(1+exp(Hosp_Y))


## cutoff
    Rec <- c()
    Prec <- c()
    F1 <- c()
    for (j in seq(1,1000,by = 1)){
      cutoff <- 1 - 0.001*j
      Rec[j] <- ModelMetrics::recall(Actual, Hosp_p_hat, cutoff = cutoff)
      Prec[j] <- ModelMetrics::precision(Actual, Hosp_p_hat, cutoff = cutoff)
      F1[j] <- 2*Rec[j]*Prec[j]/(Rec[j]+Prec[j])
    }
    ## Maximize F1 score since we care both recall and precision
    F1 <- na.fill.default(F1,0)
    Best_prec <- max(na.omit(F1))
    Best_score_ind <- Position(function(x) x == Best_prec, F1)
    cutoff <- 1 - 0.001*Best_score_ind

Hmodel_metric <- c()
Hmodel_metric["recall"] <- ModelMetrics::recall(Actual, Hosp_p_hat, cutoff = cutoff)
Hmodel_metric["precision"] <-ModelMetrics::precision(Actual, Hosp_p_hat, cutoff = cutoff)
Hmodel_metric["cutoff"] <- cutoff

Hmodel_metric
```

```
##    recall precision    cutoff
## 0.4343675 0.6127946 0.2320000
```

```
plot(Hosp_p_hat, main = "hospitalization regression", ylab = "Hospitalization probability")
```



```
#Predictions
InvestsComplete$PredHosp <- predict(final_HospModel, newdata=InvestsComplete, type="response")
InvestsComplete$PredDeath <- predict(final_DeathModel, newdata=InvestsComplete, type="response")



#Way of sorting for risk
InvestsComplete$RiskFinal <- rank((InvestsComplete$PredHosp + InvestsComplete$PredDeath)/2,
                                  ties.method = "min")
#Cuts for  Categories
P14Cuts <- max(InvestsComplete$PercentSick14)/5

#Generates Categories
InvestsComplete$AreaRisk <- cut(InvestsComplete$PercentSick14, seq(min(InvestsComplete$PercentSick14),
                                                                   max(InvestsComplete$PercentSick14)),
                                                   breaks= c(-Inf, P14Cuts, P14Cuts*2, P14Cuts*3, P14Cu
                                                   labels= c("Very Low", "Low", "Medium", "High", "Very
                                                   ordered_result = TRUE)

#Sort the patients on risk priority
InvestsComplete <- InvestsComplete[order(-InvestsComplete$RiskFinal, InvestsComplete$AreaRisk), ]
```