Load Packages

```
library(readr)
library(MASS)
library(class)
library(caret)
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
library(e1071)
library(ISLR)
library(leaps)
library(glmnet)
```

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

```
## Loaded glmnet 4.1-1
```

```
library(readr)
```

Read the File

```
national_history_date_switch <- read_csv('national2.csv')
```

```
##
## ── Column specification ────────────────────────────────────────────────────────
## cols(
##   .default = col_double(),
##   date = col_character(),
##   deathDirection = col_character(),
##   deathDirection1 = col_character(),
##   deathDirection2 = col_character(),
##   deathDirection3 = col_character(),
##   deathDirection4 = col_character()
## )
## ℹ Use `spec()` for the full column specifications.
```

```
Covid <- national_history_date_switch
Covid[is.na(Covid)] <- 0
Covid <- Covid[, -c(1:2)]
```

```
# Split the Data
set.seed(1)
train=sample(nrow(Covid),size=0.5*nrow(Covid))
test=-(train)

dim(Covid[train,])
```

```
## [1] 210  20
```

```
dim(Covid[test,])
```

```
## [1] 210  20
```

```
training=Covid[train,]
testing=Covid[-train,]
```

BIC

```
regfit=regsubsets(deathIncrease~.,data=Covid[train,],nvmax=15)

summary(regfit)
```

```
## Subset selection object
## Call: regsubsets.formula(deathIncrease ~ ., data = Covid[train, ],
##     nvmax = 15)
## 19 Variables  (and intercept)
##                           Forced in Forced out
## deathDirectionUp              FALSE      FALSE
## deathDirection1Up             FALSE      FALSE
## deathDirection2Up             FALSE      FALSE
## deathDirection3Up             FALSE      FALSE
## deathDirection4Up             FALSE      FALSE
## inIcuCumulative               FALSE      FALSE
## inIcuCurrently                FALSE      FALSE
## hospitalizedIncrease          FALSE      FALSE
## hospitalizedCurrently         FALSE      FALSE
## hospitalizedCumulative        FALSE      FALSE
## negative                      FALSE      FALSE
## negativeIncrease              FALSE      FALSE
## onVentilatorCumulative        FALSE      FALSE
## onVentilatorCurrently         FALSE      FALSE
## positive                      FALSE      FALSE
## positiveIncrease              FALSE      FALSE
## states                        FALSE      FALSE
## totalTestResults              FALSE      FALSE
## totalTestResultsIncrease      FALSE      FALSE
## 1 subsets of each size up to 15
## Selection Algorithm: exhaustive
##           deathDirectionUp deathDirection1Up deathDirection2Up
## 1  ( 1 )  " "              " "               " "
## 2  ( 1 )  " "              " "               " "
## 3  ( 1 )  " "              " "               " "
## 4  ( 1 )  " "              " "               " "
## 5  ( 1 )  " "              "*"               " "
## 6  ( 1 )  " "              " "               " "
## 7  ( 1 )  " "              "*"               " "
## 8  ( 1 )  "*"              "*"               " "
## 9  ( 1 )  "*"              "*"               " "
## 10  ( 1 ) "*"              "*"               " "
## 11  ( 1 ) "*"              "*"               " "
## 12  ( 1 ) "*"              "*"               " "
## 13  ( 1 ) "*"              "*"               " "
## 14  ( 1 ) "*"              "*"               "*"
## 15  ( 1 ) "*"              "*"               "*"
##           deathDirection3Up deathDirection4Up inIcuCumulative inIcuCurrently
## 1  ( 1 )  " "               " "               " "             "*"
## 2  ( 1 )  " "               " "               " "             " "
## 3  ( 1 )  " "               " "               " "             " "
## 4  ( 1 )  " "               " "               " "             "*"
## 5  ( 1 )  " "               " "               " "             "*"
## 6  ( 1 )  " "               " "               " "             "*"
## 7  ( 1 )  " "               " "               " "             "*"
## 8  ( 1 )  " "               " "               " "             "*"
## 9  ( 1 )  " "               " "               "*"             " "
## 10  ( 1 ) " "               " "               "*"             " "
```

```
## 11  ( 1 ) " "                " "                "*"              " "
## 12  ( 1 ) " "                " "                "*"              " "
## 13  ( 1 ) " "                " "                "*"              " "
## 14  ( 1 ) " "                " "                "*"              " "
## 15  ( 1 ) " "                " "                "*"              " "
##           hospitalizedIncrease hospitalizedCurrently hospitalizedCumulative
## 1  ( 1 )  " "                  " "                   " "
## 2  ( 1 )  "*"                  " "                   " "
## 3  ( 1 )  "*"                  " "                   " "
## 4  ( 1 )  "*"                  " "                   " "
## 5  ( 1 )  "*"                  " "                   " "
## 6  ( 1 )  "*"                  "*"                   " "
## 7  ( 1 )  "*"                  "*"                   " "
## 8  ( 1 )  "*"                  "*"                   " "
## 9  ( 1 )  "*"                  " "                   "*"
## 10  ( 1 ) "*"                  "*"                   "*"
## 11  ( 1 ) "*"                  "*"                   "*"
## 12  ( 1 ) "*"                  "*"                   "*"
## 13  ( 1 ) "*"                  "*"                   "*"
## 14  ( 1 ) "*"                  "*"                   "*"
## 15  ( 1 ) "*"                  "*"                   "*"
##           negative negativeIncrease onVentilatorCumulative
## 1  ( 1 )  " "      " "              " "
## 2  ( 1 )  " "      " "              " "
## 3  ( 1 )  " "      " "              " "
## 4  ( 1 )  " "      " "              " "
## 5  ( 1 )  " "      " "              " "
## 6  ( 1 )  " "      " "              " "
## 7  ( 1 )  " "      " "              " "
## 8  ( 1 )  " "      " "              " "
## 9  ( 1 )  " "      " "              " "
## 10  ( 1 ) " "      " "              " "
## 11  ( 1 ) " "      " "              "*"
## 12  ( 1 ) " "      " "              "*"
## 13  ( 1 ) " "      " "              "*"
## 14  ( 1 ) " "      " "              "*"
## 15  ( 1 ) "*"      " "              "*"
##           onVentilatorCurrently positive positiveIncrease states
## 1  ( 1 )  " "                   " "      " "              " "
## 2  ( 1 )  " "                   "*"      " "              " "
## 3  ( 1 )  " "                   "*"      " "              " "
## 4  ( 1 )  " "                   "*"      " "              " "
## 5  ( 1 )  " "                   "*"      " "              " "
## 6  ( 1 )  " "                   "*"      " "              " "
## 7  ( 1 )  " "                   "*"      " "              " "
## 8  ( 1 )  " "                   "*"      " "              " "
## 9  ( 1 )  "*"                   "*"      " "              "*"
## 10  ( 1 ) "*"                   "*"      " "              "*"
## 11  ( 1 ) "*"                   "*"      " "              "*"
## 12  ( 1 ) "*"                   "*"      "*"              "*"
## 13  ( 1 ) "*"                   "*"      "*"              "*"
## 14  ( 1 ) "*"                   "*"      "*"              "*"
## 15  ( 1 ) "*"                   "*"      "*"              "*"
##           totalTestResults totalTestResultsIncrease
```

```
## 1  ( 1 )  " "              " "
## 2  ( 1 )  " "              " "
## 3  ( 1 )  "*"              " "
## 4  ( 1 )  "*"              " "
## 5  ( 1 )  "*"              " "
## 6  ( 1 )  "*"              "*"
## 7  ( 1 )  "*"              "*"
## 8  ( 1 )  "*"              "*"
## 9  ( 1 )  " "              "*"
## 10  ( 1 ) " "              "*"
## 11  ( 1 ) " "              "*"
## 12  ( 1 ) " "              "*"
## 13  ( 1 ) "*"              "*"
## 14  ( 1 ) "*"              "*"
## 15  ( 1 ) "*"              "*"
```

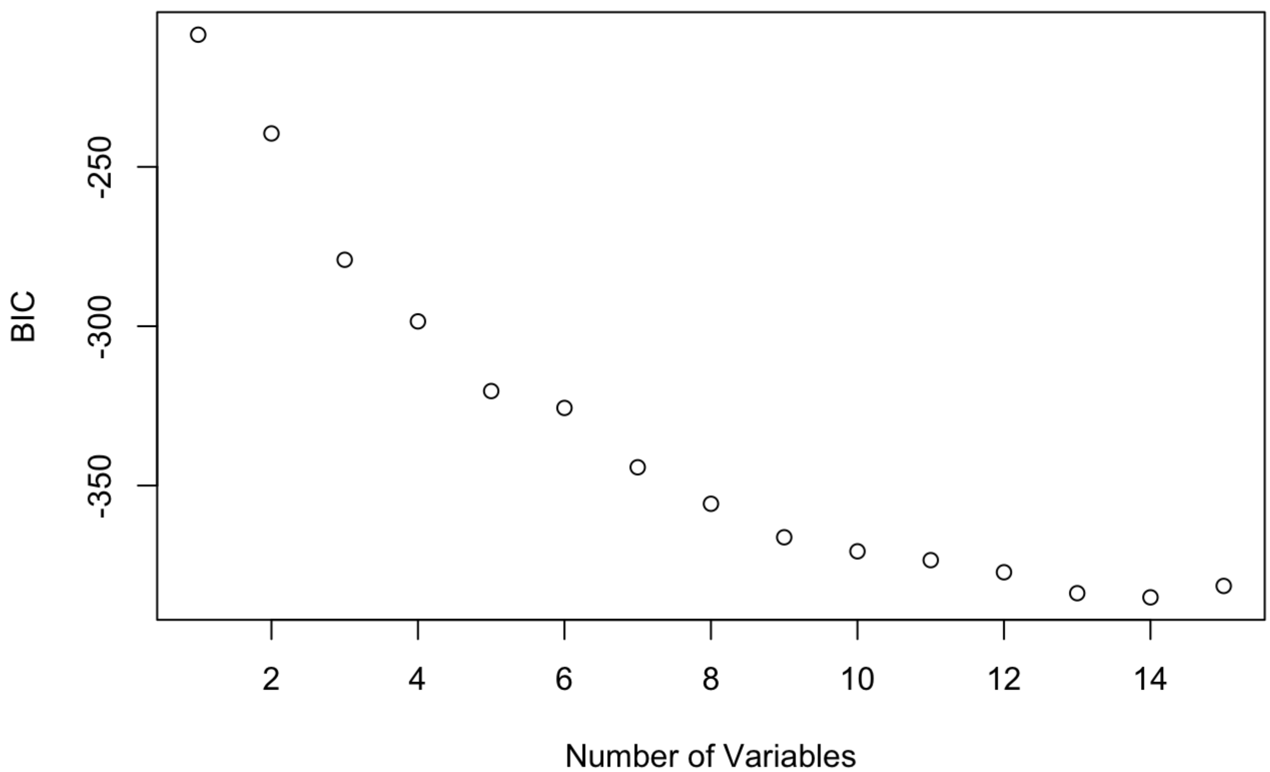
```
reg.summary=summary(regfit)
names(reg.summary)
```

```
## [1] "which"  "rsq"    "rss"    "adjr2"  "cp"     "bic"    "outmat" "obj"
```

```
reg.summary$bic
```

```
##  [1] -208.5291 -239.5195 -279.1536 -298.4901 -320.3459 -325.6457 -344.2646
##  [8] -355.7169 -366.2271 -370.6329 -373.4050 -377.2007 -383.7695 -385.0738
## [15] -381.4951
```

```
plot(reg.summary$bic,xlab="Number of Variables",ylab="BIC") ### BIC indicates the best m
odel has 9 variables
```




```
best_n=which.min(reg.summary$bic) #find the model with the lowest BIC
best_n                            # Model with 9 regressors is the best
```

```
## [1] 14
```

```
coef(regfit,id=best_n)
```

```
##             (Intercept)         deathDirectionUp        deathDirection1Up
##            4.050117e+01             4.238685e+02             2.941449e+02
##       deathDirection2Up           inIcuCumulative     hospitalizedIncrease
##            1.489491e+02            -7.279764e-01             1.065176e-01
##    hospitalizedCurrently   hospitalizedCumulative   onVentilatorCumulative
##           -2.248521e-02             1.991649e-02             2.659858e+00
##    onVentilatorCurrently                 positive         positiveIncrease
##            3.143235e-01             5.170688e-04             7.336796e-03
##                  states        totalTestResults totalTestResultsIncrease
##           -1.350200e+01            -2.883831e-05             9.402354e-04
```

Stepwise

```
model_stepwise <- train(deathIncrease ~ ., data =training,
                    method = 'glmStepAIC', direction = 'both',
                    trace = 0,
               trControl = trainControl(method = 'none', verboseIter = FALSE))

summary(model_stepwise) # This method gives 8 variables
```

```
## 
## Call:
## NULL
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -1144.3   -178.1     13.5    153.5   2010.4  
## 
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                4.050e+01  7.421e+01   0.546 0.585827    
## deathDirectionUp           4.239e+02  6.153e+01   6.889 7.56e-11 ***
## deathDirection1Up          2.941e+02  5.718e+01   5.144 6.52e-07 ***
## deathDirection2Up          1.489e+02  5.946e+01   2.505 0.013063 *  
## inIcuCumulative           -7.280e-01  8.329e-02  -8.740 1.07e-15 ***
## hospitalizedIncrease       1.065e-01  2.868e-02   3.714 0.000266 ***
## hospitalizedCurrently     -2.249e-02  4.262e-03  -5.275 3.51e-07 ***
## hospitalizedCumulative     1.992e-02  2.983e-03   6.678 2.47e-10 ***
## onVentilatorCumulative     2.660e+00  5.473e-01   4.860 2.41e-06 ***
## onVentilatorCurrently      3.143e-01  3.286e-02   9.565  < 2e-16 ***
## positive                   5.171e-04  8.565e-05   6.037 7.78e-09 ***
## positiveIncrease           7.337e-03  1.997e-03   3.674 0.000309 ***
## states                    -1.350e+01  2.823e+00  -4.784 3.39e-06 ***
## totalTestResults          -2.884e-05  8.259e-06  -3.492 0.000594 ***
## totalTestResultsIncrease  9.402e-04  2.112e-04   4.453 1.43e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for gaussian family taken to be 121822)
## 
##     Null deviance: 217767172  on 209  degrees of freedom
## Residual deviance:  23755296  on 195  degrees of freedom
## AIC: 3071.6
## 
## Number of Fisher Scoring iterations: 2
```

BIC-KNN

```
set.seed(276)
trControl=trainControl(method  = "cv",number  = 10)

knn.fit <- train(deathIncrease ~ inIcuCumulative + hospitalizedIncrease + hospitalizedCu
rrently + hospitalizedCumulative + onVentilatorCumulative + onVentilatorCurrently + posi
tive + positiveIncrease + totalTestResults, #label
                 method     = "knn", #the algorithm you select
                 tuneGrid   = expand.grid(k = 1:10), #grid for hyperparameter
                 preProcess = c("center","scale"), #standardize input data
                 trControl  = trControl,
                 metric     = "RMSE",
                 data       = training) #specify data
knn.fit
```

```
## k-Nearest Neighbors
##
## 210 samples
##   9 predictor
##
## Pre-processing: centered (9), scaled (9)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 190, 188, 188, 188, 190, 190, ...
## Resampling results across tuning parameters:
##
##   k   RMSE      Rsquared   MAE
##    1  461.6705  0.8218257  290.0006
##    2  377.8041  0.8622227  242.4938
##    3  371.7473  0.8656514  240.0481
##    4  362.5517  0.8724488  236.5830
##    5  378.2021  0.8630004  245.9442
##    6  398.7493  0.8537993  257.7555
##    7  417.2677  0.8398895  266.8665
##    8  432.6498  0.8259419  280.0156
##    9  446.6365  0.8148695  282.5174
##   10  450.4789  0.8158246  284.6154
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was k = 4.
```

```
test_pred = predict(knn.fit,newdata=testing)
mean(testing$deathIncrease-test_pred)^2
```

```
## [1] 0.140625
```

Stepwise-KNN

```
set.seed(4)
trControl=trainControl(method  = "cv",number  = 10)

knn.fit1 <- train(deathIncrease ~ inIcuCumulative +  hospitalizedIncrease + hospitalized
Currently +hospitalizedCumulative + onVentilatorCurrently + positive + positiveIncrease
 + totalTestResults, #label
                 method     = "knn", #the algorithm you select
                 tuneGrid   = expand.grid(k = 1:10), #grid for hyperparameter
                 preProcess = c("center","scale"), #standardize input data
                 trControl  = trControl,
                 metric     = "RMSE",
                 data       = training) #specify data
knn.fit1
```

```
## k-Nearest Neighbors
##
## 210 samples
##   8 predictor
##
## Pre-processing: centered (8), scaled (8)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 190, 189, 189, 187, 190, 190, ...
## Resampling results across tuning parameters:
##
##   k   RMSE      Rsquared   MAE
##    1  459.6968  0.8080517  290.4959
##    2  376.8940  0.8585859  245.9989
##    3  371.5678  0.8592665  234.7110
##    4  363.2277  0.8629891  237.1163
##    5  363.0526  0.8632598  240.4549
##    6  395.0507  0.8452910  258.7481
##    7  412.1755  0.8349530  265.5719
##    8  424.8777  0.8256935  271.2260
##    9  432.7406  0.8172640  271.2023
##   10  441.6840  0.8119456  276.7707
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was k = 5.
```

```
test_pred1 = predict(knn.fit1,newdata=testing)
mean(testing$deathIncrease-test_pred1)^2
```

```
## [1] 39.08036
```

RandomForest-KNN

```
set.seed(8)
trControl=trainControl(method  = "cv",number  = 10)

knn.fit2 <- train(deathIncrease ~ hospitalizedIncrease + totalTestResultsIncrease + onVe
ntilatorCurrently + hospitalizedCumulative + negativeIncrease + positiveIncrease + inIcu
Currently + totalTestResults, #label
                 method     = "knn", #the algorithm you select
                 tuneGrid   = expand.grid(k = 1:10), #grid for hyperparameter
                 preProcess = c("center","scale"), #standardize input data
                 trControl  = trControl,
                 metric     = "RMSE",
                 data       = training) #specify data
knn.fit2
```

```
## k-Nearest Neighbors
##
## 210 samples
##   8 predictor
##
## Pre-processing: centered (8), scaled (8)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 189, 190, 189, 186, 189, 189, ...
## Resampling results across tuning parameters:
##
##   k   RMSE      Rsquared   MAE
##    1  395.4247  0.8711760  265.9511
##    2  356.4656  0.8884258  232.8458
##    3  361.9817  0.8845384  229.8780
##    4  374.3533  0.8799502  241.4025
##    5  376.0296  0.8758882  247.8460
##    6  380.8519  0.8737664  248.9728
##    7  401.9446  0.8568702  261.7475
##    8  408.1159  0.8496633  267.9392
##    9  413.9170  0.8472182  270.7794
##   10  424.5275  0.8397886  275.2314
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was k = 2.
```

```
test_pred2 = predict(knn.fit2,newdata=testing)
mean(testing$deathIncrease-test_pred2)^2
```

```
## [1] 1292.916
```

```
# hospitalizedIncrease + totalTestResultsIncrease + onVentilatorCurrently + hospitalized
Cumulative + negativeIncrease + positiveIncrease + inIcuCurrently + totalTestResults
```

Based on KNN results, BIC model have the lowest model when K=3; Stepwise and Random Forest model has lowest test error when k = 2. BIC model has the lowest test error which is 1.286809e-05.

BIC-SVM

```
set.seed(8)

tune.out=tune(svm, deathIncrease ~ inIcuCumulative + hospitalizedIncrease + hospitalized
Currently + hospitalizedCumulative + onVentilatorCumulative + onVentilatorCurrently + po
sitive + positiveIncrease + totalTestResults ,data=training,ranges = list(epsilon = seq(
0,1,0.2), cost = 2^(2:8)))

summary(tune.out)
```

```
##
## Parameter tuning of 'svm':
##
## - sampling method: 10-fold cross validation
##
## - best parameters:
##  epsilon cost
##      0.2   16
##
## - best performance: 137517.9
##
## - Detailed performance results:
##    epsilon cost    error dispersion
## 1      0.0    4 137864.7  114345.22
## 2      0.2    4 138396.4  116433.76
## 3      0.4    4 171088.4  130308.62
## 4      0.6    4 237699.7  123857.75
## 5      0.8    4 364131.1  115462.12
## 6      1.0    4 530373.5  122138.92
## 7      0.0    8 143014.4  122625.87
## 8      0.2    8 144417.1  115249.80
## 9      0.4    8 166117.9  121109.39
## 10     0.6    8 229999.9  109848.50
## 11     0.8    8 345332.9  103761.18
## 12     1.0    8 519519.2  110913.02
## 13     0.0   16 163914.4  146347.50
## 14     0.2   16 137517.9  105384.67
## 15     0.4   16 159133.2  109729.40
## 16     0.6   16 221272.9   94767.11
## 17     0.8   16 338879.4   92121.96
## 18     1.0   16 502148.3  119634.85
## 19     0.0   32 184172.6  172289.05
## 20     0.2   32 144604.1  114190.67
## 21     0.4   32 157251.2   94648.91
## 22     0.6   32 214530.1   80588.04
## 23     0.8   32 331149.6   92424.82
## 24     1.0   32 482911.2  123592.77
## 25     0.0   64 214800.4  210998.32
## 26     0.2   64 161419.1  119219.14
## 27     0.4   64 154584.1   85387.24
## 28     0.6   64 216129.9   79716.01
## 29     0.8   64 339685.7   95599.43
## 30     1.0   64 480880.8  127660.17
## 31     0.0  128 253234.4  264474.76
## 32     0.2  128 183135.2  145427.32
## 33     0.4  128 156885.8   79377.70
## 34     0.6  128 183646.8   92642.20
## 35     0.8  128 348761.9   90402.51
## 36     1.0  128 474918.0  125913.34
## 37     0.0  256 284327.3  289913.57
## 38     0.2  256 204302.9  172602.47
## 39     0.4  256 157958.3   75439.42
## 40     0.6  256 197847.5   88709.04
```

```
## 41      0.8  256 364056.1   79558.73
## 42      1.0  256 471615.0  123662.38
```

```
test_pred=predict(tune.out$best.model,newdata=testing)
mean(testing$deathIncrease-test_pred)^2
```

```
## [1] 1675.671
```

```
### Best BIC model with SVM has epison 0 and cost 8 and test error 0.0009420781
```

Stpewise-SVM

```
set.seed(9)

tune.out1=tune(svm, deathIncrease ~ inIcuCumulative +  hospitalizedIncrease + hospitaliz
edCurrently +hospitalizedCumulative + onVentilatorCurrently + positive + positiveIncreas
e + totalTestResults ,data=training,ranges = list(epsilon = seq(0,1,0.2), cost = 2^(2:8
)))

summary(tune.out1)
```

```
##
## Parameter tuning of 'svm':
##
## - sampling method: 10-fold cross validation
##
## - best parameters:
##  epsilon cost
##        0    4
##
## - best performance: 130740.1
##
## - Detailed performance results:
##    epsilon cost    error dispersion
## 1      0.0    4 130740.1  127473.09
## 2      0.2    4 133266.7  121474.27
## 3      0.4    4 162223.7  114435.40
## 4      0.6    4 224028.2  102882.92
## 5      0.8    4 350151.1   88262.35
## 6      1.0    4 520893.3   97854.04
## 7      0.0    8 146096.9  136436.54
## 8      0.2    8 139863.0  125957.82
## 9      0.4    8 159682.5  104722.10
## 10     0.6    8 217478.3   96448.65
## 11     0.8    8 339146.7   89376.44
## 12     1.0    8 515073.8  104451.00
## 13     0.0   16 165939.0  161712.31
## 14     0.2   16 143866.3  128278.20
## 15     0.4   16 160171.2  104570.29
## 16     0.6   16 213625.8   94900.96
## 17     0.8   16 336838.9   92577.33
## 18     1.0   16 489934.4  119438.96
## 19     0.0   32 188508.5  187742.25
## 20     0.2   32 151610.0  130145.23
## 21     0.4   32 158713.1  101684.58
## 22     0.6   32 212753.8   87601.62
## 23     0.8   32 343234.1  105053.98
## 24     1.0   32 492486.0  135313.67
## 25     0.0   64 207462.9  217271.94
## 26     0.2   64 161583.6  122706.08
## 27     0.4   64 158820.6   90603.72
## 28     0.6   64 217752.2   85122.40
## 29     0.8   64 348948.3   94413.78
## 30     1.0   64 485531.9  127092.17
## 31     0.0  128 228116.7  239270.77
## 32     0.2  128 177802.5  116976.05
## 33     0.4  128 158530.2   73368.51
## 34     0.6  128 195446.5  105412.31
## 35     0.8  128 358597.8   99713.52
## 36     1.0  128 482955.7  120812.42
## 37     0.0  256 261935.1  268060.52
## 38     0.2  256 195395.2  121585.70
## 39     0.4  256 186357.2   92319.25
## 40     0.6  256 211173.4  120013.88
```

```
## 41      0.8  256 366061.4  128516.45
## 42      1.0  256 482779.3  120860.46
```

```
test_pred1=predict(tune.out1$best.model,newdata=testing)
mean(testing$deathIncrease-test_pred1)^2
```

```
## [1] 156.6271
```

```
### Best Stepwise model with SVM has epison 0 and cost 4 and test error 0.001162792
```

RandomForest-SVM

```
set.seed(11)

tune.out2=tune(svm, deathIncrease ~ hospitalizedIncrease + totalTestResultsIncrease + on
VentilatorCurrently + hospitalizedCumulative + negativeIncrease + positiveIncrease + inI
cuCurrently + totalTestResults,data=training,ranges = list(epsilon = seq(0,1,0.2), cost
 = 2^(2:8)))

summary(tune.out2)
```

```
##
## Parameter tuning of 'svm':
##
## - sampling method: 10-fold cross validation
##
## - best parameters:
##  epsilon cost
##      0.2   16
##
## - best performance: 118330.5
##
## - Detailed performance results:
##    epsilon cost    error dispersion
## 1      0.0    4 128184.6  105671.22
## 2      0.2    4 122779.9  109460.80
## 3      0.4    4 158121.1  113023.61
## 4      0.6    4 218294.6  109584.73
## 5      0.8    4 351150.6  118686.37
## 6      1.0    4 527292.6  144155.55
## 7      0.0    8 138403.8  100253.74
## 8      0.2    8 121680.7  105564.44
## 9      0.4    8 153430.0  100523.87
## 10     0.6    8 218492.5   99773.71
## 11     0.8    8 355318.3  123379.72
## 12     1.0    8 526169.7  159923.30
## 13     0.0   16 151661.3  102761.04
## 14     0.2   16 118330.5   82969.86
## 15     0.4   16 146518.1   78956.57
## 16     0.6   16 228738.1  101830.48
## 17     0.8   16 368546.2  126038.56
## 18     1.0   16 540192.1  159994.49
## 19     0.0   32 163329.9  109023.57
## 20     0.2   32 123094.8   71209.77
## 21     0.4   32 159441.5   83567.35
## 22     0.6   32 247877.0  110550.02
## 23     0.8   32 389761.6  128493.78
## 24     1.0   32 544597.6  155839.44
## 25     0.0   64 172446.6  124006.53
## 26     0.2   64 131822.4   69549.73
## 27     0.4   64 187701.2  104290.59
## 28     0.6   64 289138.4  159755.27
## 29     0.8   64 417511.5  161710.96
## 30     1.0   64 544597.6  155839.44
## 31     0.0  128 194150.7  159346.33
## 32     0.2  128 169193.8   79151.93
## 33     0.4  128 231817.8  153663.84
## 34     0.6  128 299111.6  199422.28
## 35     0.8  128 416697.2  163446.69
## 36     1.0  128 544597.6  155839.44
## 37     0.0  256 232474.0  210083.75
## 38     0.2  256 217157.9  112216.22
## 39     0.4  256 294106.1  286622.21
## 40     0.6  256 310697.9  204578.46
```

```
## 41     0.8  256 416697.2  163446.69
## 42     1.0  256 544597.6  155839.44
```

```
test_pred2=predict(tune.out2$best.model,newdata=testing)
mean(testing$deathIncrease-test_pred2)^2
```

```
## [1] 795.2648
```

```
### Best Random Forest model with SVM has epison 0 and cost 4 and test error 0.09970952
```

Based on SVM results, the BIC model has lowest test error which is 0.0009420781 and it has epison = 0, cost = 8 Later, we use Ridge & Lasso regression to control the model complexity which is selected by BIC method

BIC-Lasso/Ridge

```
# Split the Data
x.train=model.matrix(deathIncrease~inIcuCumulative + hospitalizedIncrease + hospitalized
Currently + hospitalizedCumulative + onVentilatorCumulative + onVentilatorCurrently + po
sitive + positiveIncrease + totalTestResults,Covid[train,])[,-1] #put regressors from tr
aining set into a matrix
y.train=Covid[train,]$deathIncrease #label for training set
x.test=model.matrix(deathIncrease~inIcuCumulative + hospitalizedIncrease + hospitalizedC
urrently + hospitalizedCumulative + onVentilatorCumulative + onVentilatorCurrently + pos
itive + positiveIncrease + totalTestResults,Covid[test,])[,-1] #put regressors from test
set into a matrix
y.test=Covid[test,]$deathIncrease #label for test set
```

BIC-Ridge

```
ridge.mod=glmnet(x.train,y.train,alpha=0) #build a ridge regression: alpha=0
cv.out=cv.glmnet(x.train,y.train,alpha=0) # use 10 fold cv to select shrinkage parameter
bestlam_r=cv.out$lambda.min #find the best shrinkage parameter
bestlam_r    # The lamda value
```

```
## [1] 79.30512
```

```
ridge.pred=predict(ridge.mod,s=bestlam_r,newx=x.test) #making prediction using the best
 shrinkage parameter
ridge.err=mean((ridge.pred-y.test)^2) #calculate MSE
ridge.err
```

```
## [1] 359824.4
```

```
out=glmnet(x.train,y.train,alpha=0)
predict(out,type="coefficients",s=bestlam_r)[1:9,]
```

```
##            (Intercept)        inIcuCumulative   hospitalizedIncrease 
##           4.774864e+00          -2.241493e-03           2.405351e-01 
##  hospitalizedCurrently hospitalizedCumulative onVentilatorCumulative 
##           1.596318e-03           2.673234e-04          -1.103210e-01 
##  onVentilatorCurrently               positive       positiveIncrease 
##           1.268782e-01           4.008362e-05          -1.132331e-03
```

BIC-Lasso

```
lasso.mod=glmnet(x.train,y.train,alpha=1) #build a LASSO regression
cv.out=cv.glmnet(x.train,y.train,alpha=1) # use 10 fold cv to select shrinkage parameter
bestlam_l=cv.out$lambda.min #find the best shrinkage parameter
bestlam_l
```

```
## [1] 0.08703727
```

```
lasso.pred=predict(lasso.mod,s=bestlam_l,newx=x.test) #making prediction using the best
 shrinkage parameter
lasso.err=mean((lasso.pred-y.test)^2) #calculate MSE
lasso.err
```

```
## [1] 270132.7
```

```
out=glmnet(x.train,y.train,alpha=1)
predict(out,type="coefficients",s=bestlam_l)[1:9,]
```

```
##            (Intercept)        inIcuCumulative   hospitalizedIncrease 
##          -22.962776574           -0.555649432            0.220680237 
##  hospitalizedCurrently hospitalizedCumulative onVentilatorCumulative 
##           -0.024432851            0.014433354            2.540117172 
##  onVentilatorCurrently               positive       positiveIncrease 
##            0.237051087            0.000524686            0.010961965
```

Lasso regression has lower test error 0.1969 than ridge regression 0.2744617 based on model selected with BIC. Moreover, Lasso regreesion indicates variables ‘inIcuCumulative’, ‘hospitalizedCumulative’, ‘onVentilatorCumulative’ and ‘postive’ have more explanatory power than other variables.

Elastic Net

```
Elastic.mod=glmnet(x.train,y.train,alpha=0.5) #build a ridge regression: alpha=0
cv.out=cv.glmnet(x.train,y.train,alpha=0.5) # use 10 fold cv to select shrinkage paramet
er
bestlam_e=cv.out$lambda.min #find the best shrinkage parameter
bestlam_e    # The lamda value
```

```
## [1] 0.1586102
```

```
Elastic.pred=predict(Elastic.mod,s=bestlam_e,newx=x.test) #making prediction using the b
est shrinkage parameter
Elastic.err=mean((Elastic.pred-y.test)^2) #calculate MSE
Elastic.err
```

```
## [1] 282265.5
```

```
out=glmnet(x.train,y.train,alpha=0.5)
predict(out,type="coefficients",s=bestlam_e)[1:9,]
```

```
##           (Intercept)        inIcuCumulative   hospitalizedIncrease
##         -1.751208e+01          -4.618553e-01           2.307667e-01
##  hospitalizedCurrently hospitalizedCumulative onVentilatorCumulative
##         -2.198059e-02           1.275897e-02           1.940482e+00
##  onVentilatorCurrently               positive       positiveIncrease
##          2.240305e-01           4.471337e-04           9.713347e-03
```

The test error of models with singular independent variable

```
library(randomForest)
```

```
## randomForest 4.6-14
```

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

```
##
## Attaching package: 'randomForest'
```

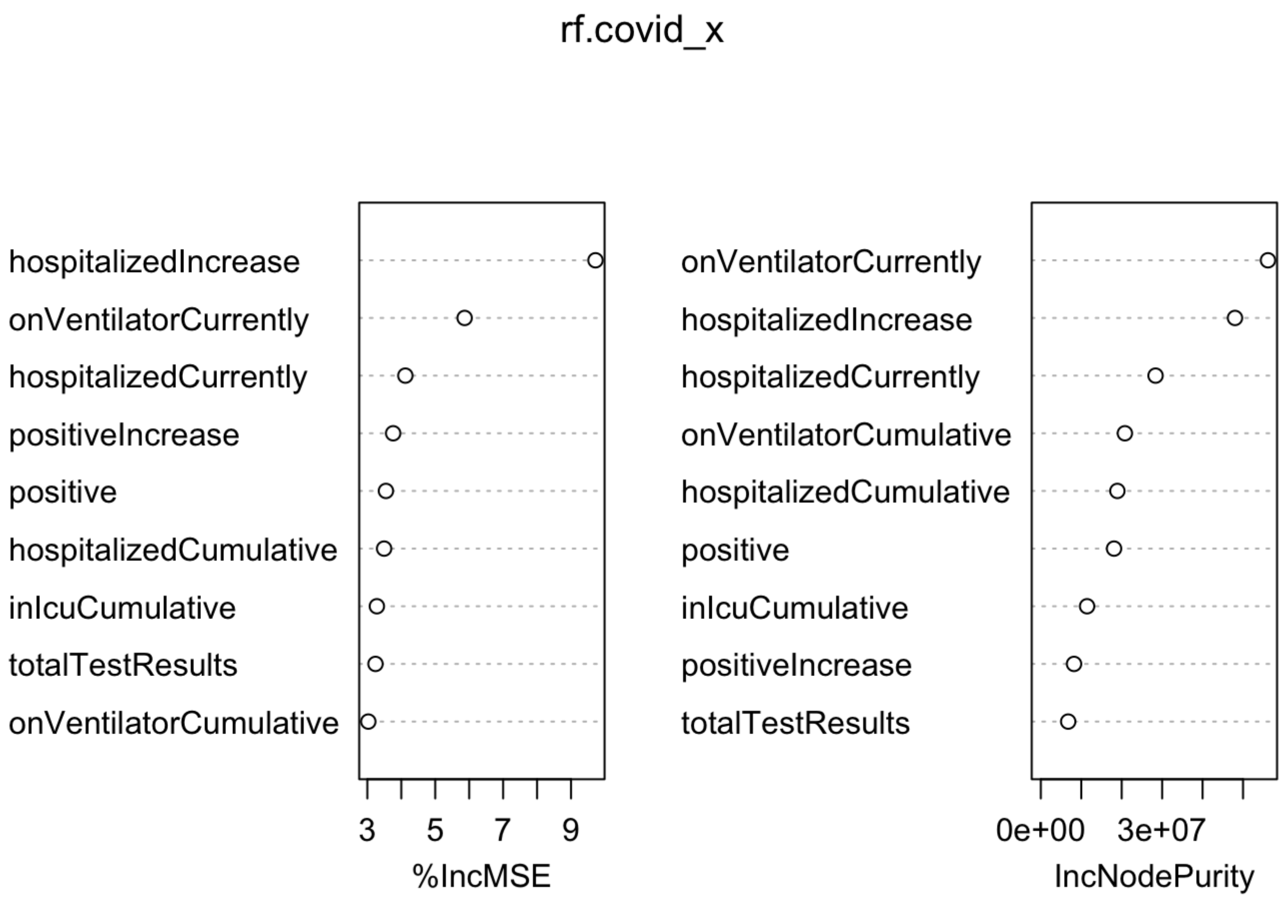
```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
rf.covid_x=randomForest(deathIncrease~inIcuCumulative + hospitalizedIncrease + hospitali
zedCurrently + hospitalizedCumulative + onVentilatorCumulative + onVentilatorCurrently +
positive + positiveIncrease + totalTestResults ,data=Covid, subset=train,mtry=5,importan
ce=TRUE,ntree=25)

importance(rf.covid_x)
```

```
##                         %IncMSE IncNodePurity
## inIcuCumulative         3.280396      11440692
## hospitalizedIncrease    9.719619      48011024
## hospitalizedCurrently   4.119324      28371601
## hospitalizedCumulative 3.493907      18928649
## onVentilatorCumulative 3.028475      20787939
## onVentilatorCurrently   5.865249      56164792
## positive                3.547225      18106988
## positiveIncrease        3.760331       8215223
## totalTestResults        3.239679       6769504
```

```
varImpPlot(rf.covid_x)
```




```
model <- lm(deathIncrease~., data = Covid)
car::vif(model)
```

```
##           deathDirection            deathDirection1            deathDirection2 
##                 1.590445                   1.503656                   1.526813 
##          deathDirection3            deathDirection4            inIcuCumulative 
##                 1.528459                   1.552630                3917.879775 
##           inIcuCurrently       hospitalizedIncrease    hospitalizedCurrently 
##               141.692596                   2.130408                  96.843462 
##   hospitalizedCumulative                   negative           negativeIncrease 
##              1234.006119                3452.370738                   4.018872 
##   onVentilatorCumulative      onVentilatorCurrently                   positive 
##              2981.453260                  60.615614                 930.373484 
##         positiveIncrease                     states           totalTestResults 
##                34.060342                   5.305762                2151.488265 
## totalTestResultsIncrease 
##                43.499864 
```