Study on COVID Death Rate in Nursing Home

SD: Submitted Data PQA: Passed Quality Assurance Check RTA: Residents Total Admissions COVID-19 NB: Number of All Beds OB: Total Number of Occupied Beds CTR: Able to Test or Obtain Resources to Test All Current Residents Within Next 7 Days CTS: Able to Test or Obtain Resources to Test All Staff and/or Personnel Within Next 7 Days TR: COVID-19 Point-of-Care Tests Performed on Residents Since Last Report TS: COVID-19 Point-of-Care Tests Performed on Staff and/or Personnel Since Last Report STC: Staff Total Confirmed COVID-19 STD: Staff Total COVID-19 Deaths SN: Shortage of Nursing Staff SC: Shortage of Clinical Staff SA: Shortage of Aides SO: Shortage of Other Staff RCR: Total Resident Confirmed COVID-19 Cases Per 1,000 Residents RDR: Total Resident COVID-19 Deaths Per 1,000 Residents NRNI: Number of Residents with New Influenza NRAR: Number of Residents with Acute Respiratory Illness Symptoms Excluding COVID-19 and/or Influenza ATT: During Past Two Weeks Average Time to Receive COVID-19 Test Results from Non-Point-of-Care Tests NSRA: Number of Staff and/or Personnel with Acute Respiratory Illness Symptoms Excluding COVID-19 and/or Influenza ABHR: Alcohol-Based Hand Rub (ABHR) Available N95RS: N95 Respirator Strategy for Optimization FMS: Face Masks Strategy for Optimization EPS: Eye Protection Strategy for Optimization GS: Gowns Strategy for Optimization GLS: Gloves Strategy for Optimization

Data Exploration and cleanup

```r
# Import dataset
data <- read.csv("COVID_19_Nursing_Home_Data.csv", head=TRUE)
```

```r
# Cleanup
data[data == "" | data == " "] <- NA
# Drop subjects that did not submit data in this cycle or data did not pass
# QA check
# Drop rows where response is missing
data = data[(data$SD == "Y" | data$PQA == "Y") & !is.na(data$RDR), ]
# Drop submit data and QA check status
data <- subset(data, select = -c(SD, PQA))
# Remove subjects with only NA values
data = data[!!rowSums(!is.na(data)),]
# Use bed occupation rate instead of bed counts
data$BOC = data$OB / data$NB
# Drop bed counts
data <- subset(data, select = -c(OB, NB))
name <- names(data)

for (i in 1:length(name)) {
  col =name[i]
  if (class(data[, col]) == "character") {
    data[, col]= as.factor(data[, col])
  }
}
```

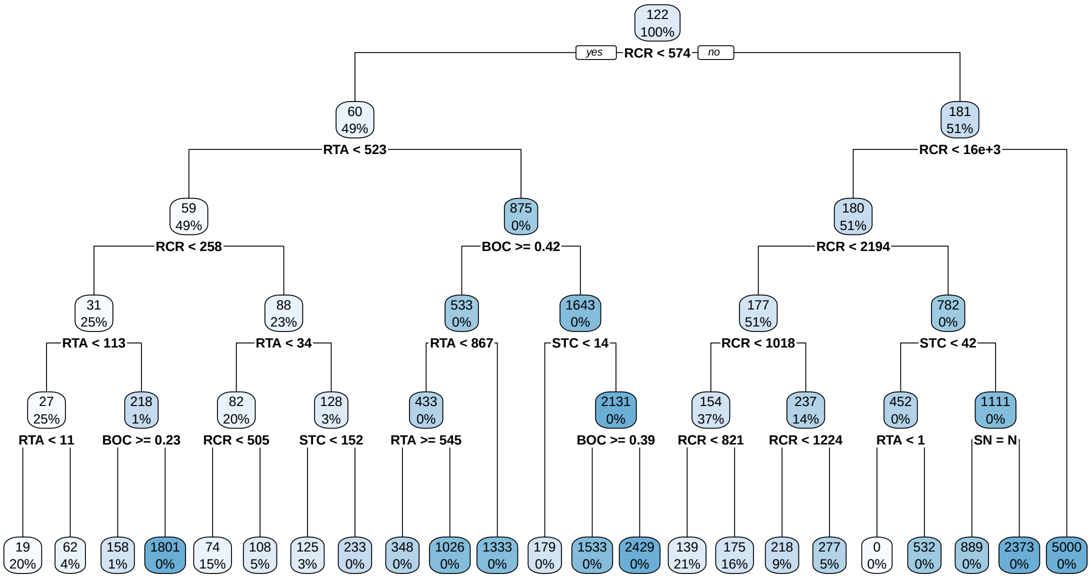
```
# Check which columns are missing
index <- names(data)
tabcol2 <- rep(NA, length(index))
for (i in 1:length(index)){
  col = index[i]
  tabcol2[i] = length(which(is.na(data[,col])))
}
index[which(tabcol2/length(data[,1])>0.05)] # missing percentage > 0.05
```

```
## [1] "TR" "TS"
```

```
index[which(tabcol2/length(data[,1])>0)] # missing data
```

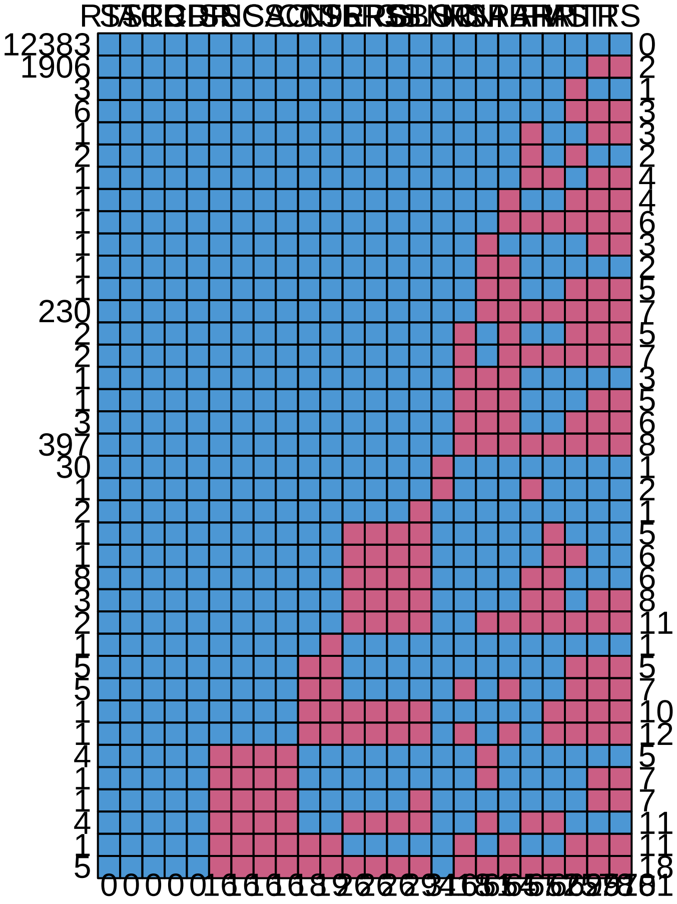
```
##  [1] "CTR"   "CTS"   "TR"    "TS"    "SN"    "SC"    "SA"    "SO"    "NRNI"
## [10] "NRAR"  "ATT"   "NSRA"  "ABHR"  "N95RS" "FMS"   "EPS"   "GS"    "GLS"
## [19] "BOC"
```

```
# Use decision tree to find significant variate that we want to keep
df <- rpart(data$RDR~ .,
            data = data,
            control = rpart.control(minsplit = 1,
                                    minbucket = 1,
                                    maxdepth = 5,
                                    cp = 0,
                                    xval = 6))
rpart.plot(df)
```



```
# variables to consider
var_import <- names(df$variable.importance)
```

```
# Visualize missing pattern of original dataset
md.pattern(data)
```



```
##       RTA STC STD RCR RDR SN SC SA SO CTS CTR N95RS EPS GS GLS BOC NRNI NSRA
## 12383   1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    1    1
## 1906    1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    1    1
## 3       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    1    1
## 6       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    1    1
## 1       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    1    1
## 2       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    1    1
## 1       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    1    1
## 1       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    1    1
## 1       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    1    1
## 1       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    1    0
## 1       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    1    0
## 1       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    1    0
## 230     1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    1    0
## 2       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    0    1
## 2       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    0    1
## 1       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    0    0
## 1       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    0    0
## 3       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    0    0
## 397     1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   1    0    0
## 30      1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   0    1    1
## 1       1   1   1   1   1  1  1  1  1   1   1     1   1  1   1   0    1    1
## 2       1   1   1   1   1  1  1  1  1   1   1     1   1  1   0   1    1    1
## 1        1   1   1   1   1  1  1  1  1   1   1    0   0  0   0   1    1    1
## 1        1   1   1   1   1  1  1  1  1   1   1    0   0  0   0   1    1    1
## 8        1   1   1   1   1  1  1  1  1   1   1    0   0  0   0   1    1    1
## 3        1   1   1   1   1  1  1  1  1   1   1    0   0  0   0   1    1    1
## 2        1   1   1   1   1  1  1  1  1   1   1    0   0  0   0   1    1    0
## 1        1   1   1   1   1  1  1  1  1   1   0    1   1  1   1   1    1    1
## 5        1   1   1   1   1  1  1  1  1   0   0    1   1  1   1   1    1    1
## 5        1   1   1   1   1  1  1  1  1   0   0    1   1  1   1   1    0    1
## 1        1   1   1   1   1  1  1  1  1   0   0    0   0  0   0   1    1    1
## 1        1   1   1   1   1  1  1  1  1   0   0    0   0  0   0   1    0    1
## 4        1   1   1   1   1  0  0  0  0   1   1    1   1  1   1   1    1    0
## 1        1   1   1   1   1  0  0  0  0   1   1    1   1  1   1   1    1    0
## 1        1   1   1   1   1  0  0  0  0   1   1    1   1  1   0   1    1    1
## 4        1   1   1   1   1  0  0  0  0   1   1    0   0  0   0   1    1    0
## 1        1   1   1   1   1  0  0  0  0   0   0    1   1  1   1   1    0    1
## 5        1   1   1   1   1  0  0  0  0   0   0    0   0  0   0   1    0    0
##          0   0   0   0   0 16 16 16 16  18  19   26  26 26  29  31  418  651
##        NRAR ABHR FMS ATT    TR   TS
## 12383    1    1   1   1     1    1    0
## 1906     1    1   1   1     0    0    2
## 3        1    1   1   0     1    1    1
## 6        1    1   1   0     0    0    3
## 1        1    0   1   1     0    0    3
## 2        1    0   1   0     1    1    2
## 1        1    0   0   1     0    0    4
## 1        0    1   1   0     0    0    4
## 1        0    0   0   0     0    0    6
## 1        1    1   1   1     0    0    3
## 1        0    1   1   1     1    1    2
## 1        0    1   1   0     0    0    5
## 230      0    0   0   0     0    0    7
## 2        0    1   1   0     0    0    5
## 2        0    0   0   0     0    0    7
## 1        0    1   1   1     1    1    3
## 1        0    1   1   1     0    0    5
## 3        0    1   1   0     0    0    6
## 397      0    0   0   0     0    0    8
## 30       1    1   1   1     1    1    1
## 1        1    0   1   1     1    1    2
## 2        1    1   1   1     1    1    1
## 1        1    1   0   1     1    1    5
## 1        1    1   0   0     1    1    6
## 8        1    0   0   1     1    1    6
## 3        1    0   0   1     0    0    8
## 2        0    0   0   0     0    0   11
## 1        1    1   1   1     1    1    1
## 5        1    1   1   0     0    0    5
## 5        0    1   1   0     0    0    7
## 1        1    1   0   0     0    0   10
## 1        0    1   0   0     0    0   12
## 4        1    1   1   1     1    1    5
## 1        1    1   1   1     0    0    7
## 1        1    1   1   1     0    0    7
## 4        1    0   0   1     1    1   11
## 1        0    1   1   0    0    0   11
## 5        0    0   0   0    0    0   18
##        654  657 657 669 2578 2578 9101
```

Missing Mechanism:

```
# Variable with missing values
var_miss <- index[which(tabcol2/length(data[,1])>0)]
# Only keep the significant ones we wanted to keep in previous section
var_miss <- intersect(var_miss, var_import)
# Significant variables that do not contain missing values
var_complete <- setdiff(var_import, var_miss)

# Compute missing mechanism of all variables with missing values
for (i in 1:length(var_miss)) {
  loopdata = data[var_import]
  misscol = var_miss[i]
  print(misscol)
  loopdata$R <- ifelse(is.na(data[,misscol]), 1, 0)
  loopdata[,misscol] <- NULL
  mechanism = glm(R ~ RCR + STC + RTA + STD,family = "binomial",
                data=loopdata)
  print(summary(mechanism))
}
```

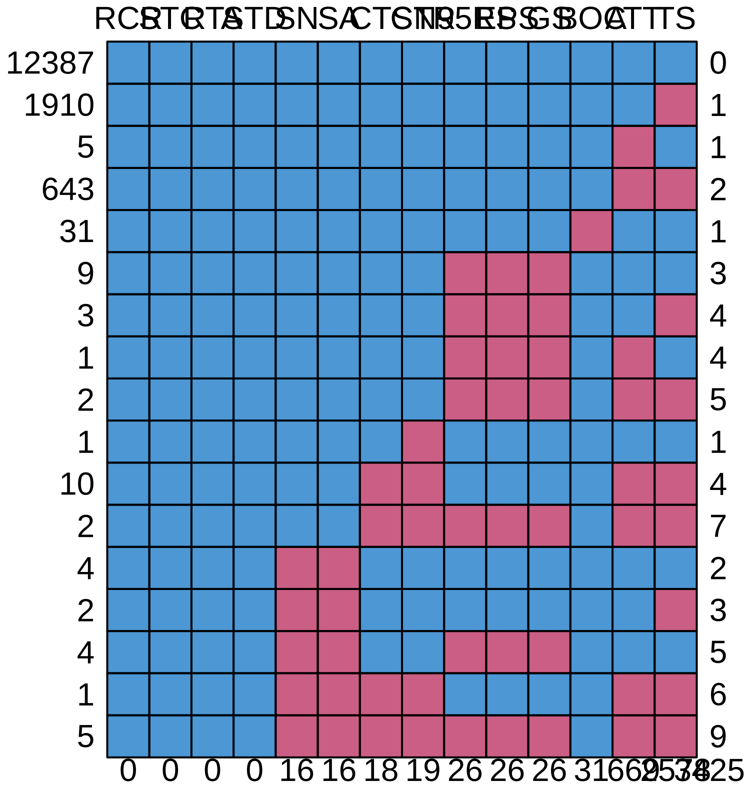
```
## [1] "CTR"
##
## Call:
## glm(formula = R ~ RCR + STC + RTA + STD, family = "binomial",
##     data = loopdata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.2893  -0.0518  -0.0480  -0.0452   3.7463
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -7.075e+00  4.000e-01 -17.689   <2e-16 ***
## RCR          6.946e-05  3.657e-04   0.190    0.849
## STC          6.868e-03  6.113e-03   1.123    0.261
## RTA          2.110e-03  2.827e-03   0.746    0.455
## STD          1.111e-01  2.119e-01   0.524    0.600
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 291.54  on 15019  degrees of freedom
## Residual deviance: 289.30  on 15015  degrees of freedom
## AIC: 299.3
##
```

```
## Number of Fisher Scoring iterations: 9
##
## [1] "CTS"
##
## Call:
## glm(formula = R ~ RCR + STC + RTA + STD, family = "binomial",
##     data = loopdata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.2692  -0.0503  -0.0469  -0.0446   3.7463
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -7.069e+00  4.273e-01 -16.545   <2e-16 ***
## RCR          2.885e-05  4.374e-04   0.066    0.947
## STC          6.164e-03  6.505e-03   0.948    0.343
## RTA          2.349e-03  2.760e-03   0.851    0.395
## STD          6.348e-02  2.930e-01   0.217    0.828
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 278.14  on 15019  degrees of freedom
## Residual deviance: 276.40  on 15015  degrees of freedom
## AIC: 286.4
##
## Number of Fisher Scoring iterations: 9
##
## [1] "TS"
##
## Call:
## glm(formula = R ~ RCR + STC + RTA + STD, family = "binomial",
##     data = loopdata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9984  -0.6394  -0.6072  -0.5373   2.6377
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.412e+00  4.246e-02 -33.266  < 2e-16 ***
## RCR          2.227e-04  4.565e-05   4.878 1.07e-06 ***
## STC         -8.887e-03  9.949e-04  -8.932  < 2e-16 ***
## RTA          1.220e-03  4.236e-04   2.880  0.00398 **
## STD          7.429e-02  3.306e-02   2.247  0.02462 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 13773  on 15019  degrees of freedom
## Residual deviance: 13674  on 15015  degrees of freedom
## AIC: 13684
##
## Number of Fisher Scoring iterations: 4
##
## [1] "SN"
##
## Call:
## glm(formula = R ~ RCR + STC + RTA + STD, family = "binomial",
##     data = loopdata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.2111  -0.0466  -0.0456  -0.0447   3.7312
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.955e+00  4.717e-01 -14.744   <2e-16 ***
## RCR          4.323e-05  4.798e-04   0.090    0.928
## STC          1.949e-03  9.144e-03   0.213    0.831
## RTA         -8.798e-04  5.992e-03  -0.147    0.883
## STD          1.493e-01  1.780e-01   0.839    0.402
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 251.01  on 15019  degrees of freedom
## Residual deviance: 250.52  on 15015  degrees of freedom
## AIC: 260.52
##
## Number of Fisher Scoring iterations: 9
##
## [1] "SA"
##
## Call:
## glm(formula = R ~ RCR + STC + RTA + STD, family = "binomial",
##     data = loopdata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.2111  -0.0466  -0.0456  -0.0447   3.7312
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.955e+00  4.717e-01 -14.744   <2e-16 ***
## RCR          4.323e-05  4.798e-04   0.090    0.928
## STC          1.949e-03  9.144e-03   0.213    0.831
## RTA         -8.798e-04  5.992e-03  -0.147    0.883
## STD          1.493e-01  1.780e-01   0.839    0.402
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 251.01  on 15019  degrees of freedom
## Residual deviance: 250.52  on 15015  degrees of freedom
## AIC: 260.52
##
## Number of Fisher Scoring iterations: 9
##
## [1] "ATT"
##
## Call:
## glm(formula = R ~ RCR + STC + RTA + STD, family = "binomial",
##     data = loopdata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6413  -0.3028  -0.2900  -0.2791   2.5908
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.321e+00  7.301e-02 -45.478  < 2e-16 ***
## RCR          1.014e-04  6.606e-05   1.535  0.12488
## STC          2.179e-03  1.349e-03   1.616  0.10620
## RTA          3.505e-03  5.035e-04   6.961 3.37e-12 ***
## STD          1.216e-01  4.290e-02   2.835  0.00459 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5470.7  on 15019  degrees of freedom
## Residual deviance: 5411.4  on 15015  degrees of freedom
## AIC: 5421.4
##
## Number of Fisher Scoring iterations: 6
##
## [1] "N95RS"
##
## Call:
## glm(formula = R ~ RCR + STC + RTA + STD, family = "binomial",
##     data = loopdata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.2901  -0.0605  -0.0573  -0.0546   3.6398
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.6616163  0.3438059 -19.376   <2e-16 ***
## RCR          0.0001241  0.0002623   0.473    0.636
## STC          0.0043432  0.0061803   0.703    0.482
## RTA          0.0010027  0.0032526   0.308    0.758
## STD          0.1536400  0.1355692   1.133    0.257
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 382.63  on 15019  degrees of freedom
## Residual deviance: 380.83  on 15015  degrees of freedom
## AIC: 390.83
##
## Number of Fisher Scoring iterations: 9
##
## [1] "EPS"
##
## Call:
## glm(formula = R ~ RCR + STC + RTA + STD, family = "binomial",
##     data = loopdata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.2901  -0.0605  -0.0573  -0.0546   3.6398
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.6616163  0.3438059 -19.376   <2e-16 ***
## RCR          0.0001241  0.0002623   0.473    0.636
## STC          0.0043432  0.0061803   0.703    0.482
## RTA          0.0010027  0.0032526   0.308    0.758
## STD          0.1536400  0.1355692   1.133    0.257
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 382.63  on 15019  degrees of freedom
## Residual deviance: 380.83  on 15015  degrees of freedom
## AIC: 390.83
##
## Number of Fisher Scoring iterations: 9
##
## [1] "GS"
##
## Call:
## glm(formula = R ~ RCR + STC + RTA + STD, family = "binomial",
##     data = loopdata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.2901  -0.0605  -0.0573  -0.0546   3.6398
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.6616163  0.3438059 -19.376   <2e-16 ***
## RCR          0.0001241  0.0002623   0.473    0.636
## STC          0.0043432  0.0061803   0.703    0.482
## RTA          0.0010027  0.0032526   0.308    0.758
## STD          0.1536400  0.1355692   1.133    0.257
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 382.63  on 15019  degrees of freedom
## Residual deviance: 380.83  on 15015  degrees of freedom
## AIC: 390.83
##
## Number of Fisher Scoring iterations: 9
##
## [1] "BOC"
##
## Call:
## glm(formula = R ~ RCR + STC + RTA + STD, family = "binomial",
##     data = loopdata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.4600  -0.0666  -0.0640  -0.0608   3.7537
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.3517758  0.3155708 -20.128   <2e-16 ***
## RCR          0.0002035  0.0001879   1.083    0.279
## STC          0.0004335  0.0063275   0.069    0.945
## RTA          0.0033508  0.0017172   1.951    0.051 .
## STD         -0.9714388  0.9260504  -1.049    0.294
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 445.29  on 15019  degrees of freedom
## Residual deviance: 440.56  on 15015  degrees of freedom
## AIC: 450.56
##
## Number of Fisher Scoring iterations: 10
```

```r
# Visualization of missing pattern of dataset with only significant variables
data_import <- data[c(var_import, "RDR")]
md.pattern(data[var_import])
```



```
##       RCR STC RTA STD SN SA CTS CTR N95RS EPS GS BOC ATT   TS
## 12387   1   1   1   1  1  1   1   1     1   1  1   1   1    1    0
## 1910    1   1   1   1  1  1   1   1     1   1  1   1   1    0    1
## 5       1   1   1   1  1  1   1   1     1   1  1   1   0    1    1
## 643     1   1   1   1  1  1   1   1     1   1  1   1   0    0    2
## 31      1   1   1   1  1  1   1   1     1   1  1   0   1    1    1
## 9       1   1   1   1  1  1   1   1     0   0  0   1   1    1    3
## 3       1   1   1   1  1  1   1   1     0   0  0   1   1    0    4
## 1       1   1   1   1  1  1   1   1     0   0  0   1   0    1    4
## 2       1   1   1   1  1  1   1   1     0   0  0   1   0    0    5
## 1       1   1   1   1  1  1   1   0     1   1  1   1   1    1    1
## 10      1   1   1   1  1  1   0   0     1   1  1   1   0    0    4
## 2       1   1   1   1  1  1   0   0     0   0  0   1   0    0    7
## 4       1   1   1   1  0  0   1   1     1   1  1   1   1    1    2
## 2       1   1   1   1  0  0   1   1     1   1  1   1   1    0    3
## 4       1   1   1   1  0  0   1   1     0   0  0   1   1    1    5
## 1       1   1   1   1  0  0   0   0     1   1  1   1   0    0    6
## 5       1   1   1   1  0  0   0   0     0   0  0   1   0    0    9
##          0   0   0   0 16 16  18  19    26  26 26  31 669 2578 3425
```

MCAR Imputation

```r
# Variable MCAR, impute them with linear model
fit=glm(CTR~ RCR + STC + RTA + STD,data=data_import, family = "binomial")
data_import$impute.CTR=predict(fit,newdata=data_import)
data_import$impute.CTR[!is.na(data_import$CTR)]=
  data_import$CTR[!is.na(data_import$CTR)]
data_import$CTR = data_import$impute.CTR
data_import = subset(data_import, select = -c(impute.CTR))


fit=glm(CTS~ RCR + STC + RTA + STD,data=data_import, family = "binomial")
data_import$impute.CTS=predict(fit,newdata=data_import)
data_import$impute.CTS[!is.na(data_import$CTS)]=
  data_import$CTS[!is.na(data_import$CTS)]
data_import$CTS = data_import$impute.CTS
data_import = subset(data_import, select = -c(impute.CTS))

fit=glm(SN~ RCR + STC + RTA + STD,data=data_import, family = "binomial")
data_import$impute.SN=predict(fit,newdata=data_import)
data_import$impute.SN[!is.na(data_import$SN)]=
  data_import$SN[!is.na(data_import$SN)]
data_import$SN = data_import$impute.SN
data_import = subset(data_import, select = -c(impute.SN))

fit=glm(SA~ RCR + STC + RTA + STD,data=data_import, family = "binomial")
data_import$impute.SA=predict(fit,newdata=data_import)
data_import$impute.SA[!is.na(data_import$SA)]=
  data_import$SA[!is.na(data_import$SA)]
data_import$SA = data_import$impute.SA
data_import = subset(data_import, select = -c(impute.SA))

fit=glm(N95RS~ RCR + STC + RTA + STD,data=data_import, family = "binomial")
data_import$impute.N95RS=predict(fit,newdata=data_import)
data_import$impute.N95RS[!is.na(data_import$N95RS)]=
  data_import$N95RS[!is.na(data_import$N95RS)]
data_import$N95RS = data_import$impute.N95RS
data_import = subset(data_import, select = -c(impute.N95RS))

fit=multinom(EPS~ RCR + STC + RTA + STD, data = data_import)
```

```
## # weights:  18 (10 variable)
## initial  value 16472.592656
## iter  10 value 6480.042263
## iter  20 value 6086.286415
## final  value 6086.208315
## converged
```

```r
data_import$impute.EPS=predict(fit,newdata=data_import)
data_import$impute.EPS[!is.na(data_import$EPS)]=
  data_import$EPS[!is.na(data_import$EPS)]
data_import$EPS = data_import$impute.EPS
data_import = subset(data_import, select = -c(impute.EPS))

fit=multinom(GS~ RCR + STC + RTA + STD, data = data_import)
```

```
## # weights:  18 (10 variable)
## initial  value 16472.592656
## iter  10 value 4988.606278
## iter  20 value 4318.424867
## iter  30 value 4318.084893
## iter  40 value 4318.073317
## final  value 4318.073237
## converged
```

```r
data_import$impute.GS=predict(fit,newdata=data_import)
data_import$impute.GS[!is.na(data_import$GS)]=
  data_import$GS[!is.na(data_import$GS)]
data_import$GS = data_import$impute.GS
data_import = subset(data_import, select = -c(impute.GS))
```

```r
for (i in 1:length(names(data_import))) {
  col =names(data_import)[i]
  if (class(data_import[, col]) == "factor") {
    data_import[, col]= as.numeric(data_import[, col])
  }
}
```

Non-MCAR Imputation

EM algorithm

```r
set.seed(438)
data.imputed.em=amelia(data_import, m=5)
```

```
## -- Imputation 1 --
##
##   1  2  3  4  5  6  7
##
## -- Imputation 2 --
##
##   1  2  3  4  5
##
## -- Imputation 3 --
##
##   1  2  3  4  5  6  7  8
##
## -- Imputation 4 --
##
##   1  2  3  4  5  6  7  8  9
##
## -- Imputation 5 --
##
##   1  2  3  4
```

```
data.imputed.em <- data.imputed.em$imputations$imp5
```

Analysis, Correlation

```
### stage1
resp.zero <- which(data.imputed.em$RDR==0)
resp.nonzero <- which(data.imputed.em$RDR!=0)

data.imputed.em$bin.resp <- ifelse(data.imputed.em$RDR==0, 0, 1)

m.stage1 <- glm(bin.resp~RCR+STC+RTA+BOC+SN+GS+SA+TS+STD+EPS+N95RS+CTS+CTR,
                family=binomial(link="logit"),
                data=data.imputed.em)
```

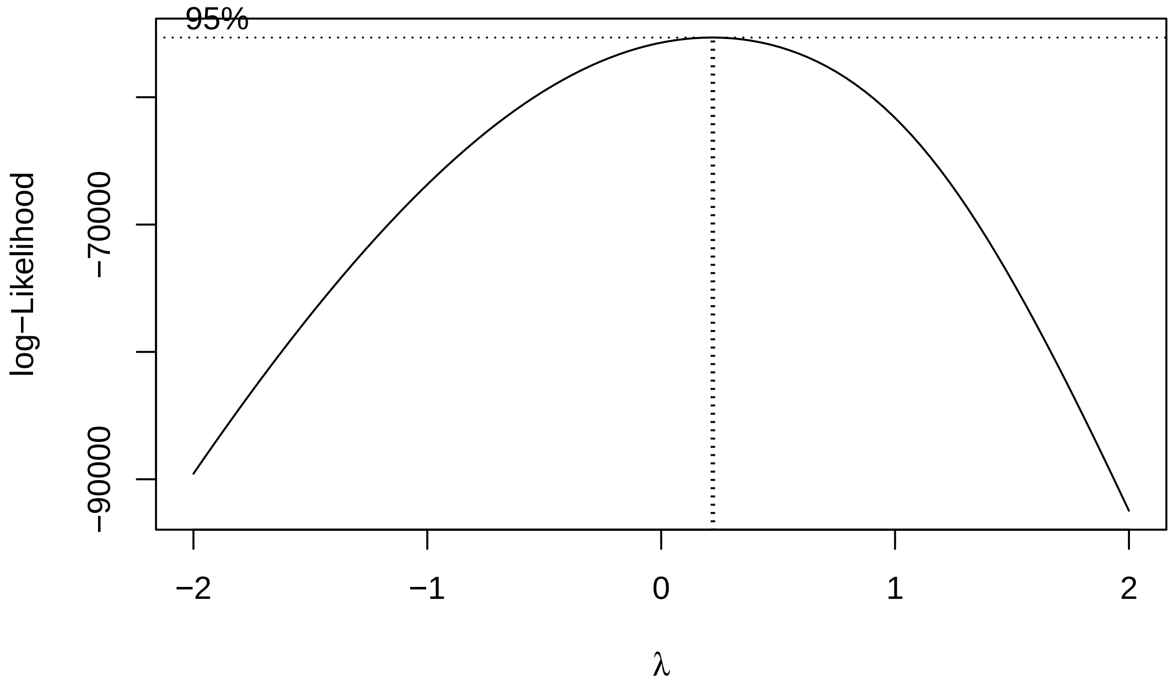
```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(m.stage1)
```

```
##
## Call:
## glm(formula = bin.resp ~ RCR + STC + RTA + BOC + SN + GS + SA +
##     TS + STD + EPS + N95RS + CTS + CTR, family = binomial(link = "logit"),
##     data = data.imputed.em)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -8.4904   0.0528   0.1966   0.4442   1.7400
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.676e+00  4.940e-01  -3.394  0.00069 ***
## RCR          4.173e-03  1.178e-04  35.435  < 2e-16 ***
## STC          3.200e-02  2.003e-03  15.978  < 2e-16 ***
## RTA          3.267e-02  2.587e-03  12.628  < 2e-16 ***
## BOC          3.004e-01  1.502e-01   2.000  0.04553 *
## SN           5.191e-02  1.216e-01   0.427  0.66950
## GS           8.360e-02  1.232e-01   0.678  0.49747
## SA          -4.055e-02  1.154e-01  -0.352  0.72519
## TS           8.594e-04  4.152e-04   2.070  0.03844 *
## STD          2.170e-01  8.093e-02   2.681  0.00734 **
## EPS         -7.885e-05  1.120e-01  -0.001  0.99944
## N95RS       -4.824e-02  9.134e-02  -0.528  0.59741
## CTS          2.761e-01  5.084e-01   0.543  0.58709
## CTR         -2.752e-01  5.011e-01  -0.549  0.58282
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 13875.9  on 15019  degrees of freedom
## Residual deviance:  8632.1  on 15006  degrees of freedom
## AIC: 8660.1
##
## Number of Fisher Scoring iterations: 8
```

```
### stage2
#### Box-cox transformation
bc <- boxcox(lm(RDR~RCR+STC+RTA+BOC+SN+GS+SA+TS+STD+EPS+N95RS+CTS+CTR,
                data=data.imputed.em[resp.nonzero,]))
```



```
lambda <- bc$x[which.max(bc$y)]

m.stage2 <- glm((RDR^lambda-1)/lambda~RCR+STC+RTA+BOC+SN+GS+SA+TS+STD+EPS+N95RS+CTS+CTR,
                data=data.imputed.em[resp.nonzero,])
summary(m.stage2)
```

```
##
## Call:
## glm(formula = (RDR^lambda - 1)/lambda ~ RCR + STC + RTA + BOC +
##     SN + GS + SA + TS + STD + EPS + N95RS + CTS + CTR, data = data.imputed.em[resp.nonzero,
##     ])
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -34.900   -1.392    0.102    1.376   14.111
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.3371737  0.3458534  18.323  < 2e-16 ***
## RCR          0.0026112  0.0000462  56.518  < 2e-16 ***
## STC          0.0052874  0.0007866   6.722 1.87e-11 ***
## RTA          0.0078496  0.0003809  20.609  < 2e-16 ***
## BOC         -1.8101680  0.1144047 -15.822  < 2e-16 ***
## SN          -0.1485758  0.0916255  -1.622  0.10492
## GS           0.0087401  0.0898843   0.097  0.92254
## SA           0.2340079  0.0880236   2.658  0.00786 **
## TS          -0.0014927  0.0002382  -6.268 3.79e-10 ***
## STD          0.2139176  0.0317822   6.731 1.76e-11 ***
## EPS          0.0666069  0.0841256   0.792  0.42852
## N95RS       -0.0731614  0.0675805  -1.083  0.27902
## CTS          0.3493270  0.3306940   1.056  0.29083
## CTR          0.1436744  0.3256719   0.441  0.65910
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for gaussian family taken to be 4.648419)
##
##     Null deviance: 82335  on 12408  degrees of freedom
## Residual deviance: 57617  on 12395  degrees of freedom
## AIC: 54298
##
## Number of Fisher Scoring iterations: 2
```

```
# stage 1
library(knitr)
table1 <- data.frame(rbind(
  c("RCR",4.173e-03,  1.178e-04),
  c("STC",3.200e-02,  2.003e-03),
  c("RTA",3.267e-02,  2.587e-03),
  c("BOC",3.004e-01,  1.502e-01),
  c("TS",8.594e-04,  4.152e-04),
  c("STD",2.170e-01,  8.093e-02)))
colnames(table1) <- c("Predictor", "Estimates", "Standard Error")
kable(table1)
```

Predictor	Estimates	Standard Error
RCR	0.004173	0.0001178
STC	0.032	0.002003
RTA	0.03267	0.002587
BOC	0.3004	0.1502
TS	0.0008594	0.0004152
STD	0.217	0.08093

```
# stage 2
library(knitr)
table2 <- data.frame(rbind(
  c("RCR",0.0026112,  0.0000462),
  c("STC",0.0052874,  0.0007866),
  c("RTA",0.0078496,  0.0003809),
  c("BOC",-1.8101680,  0.1144047),
  c("SA",0.2340079,  0.0880236),
  c("TS",-0.0014927,  0.0002382),
  c("STD",0.2139176,  0.0317822)))
colnames(table2) <- c("Predictor", "Estimates", "Standard Error")
kable(table2)
```

Predictor	Estimates	Standard Error
RCR	0.0026112	4.62e-05
STC	0.0052874	0.0007866
RTA	0.0078496	0.0003809
BOC	-1.810168	0.1144047
SA	0.2340079	0.0880236
TS	-0.0014927	0.0002382
STD	0.2139176	0.0317822