ITC pt 3

Fayre-Ella Ooi

2025-04-16

```
#dataset
LookUp = read.csv(file = "/Users/fayreooi/Downloads/LookUpUpdate.csv")
LookUp = LookUp[, -c(11,12,13,14,15)]

## clean dataset

# make numbers numerical
LookUp$Customer.Count = as.numeric(gsub(",", "", LookUp$Customer.Count))

# interpreting SAIDI add a column to dataset showing average minutes
LookUp$Avg.Min.PerCustomer.2023 = LookUp$X2023.SAIDI * 60

# adding CAIDI
# LookUp$CAIDI.2023 = LookUp$X2023.SAIDI / LookUp$X2023.SAIFI

# dataset
CircuitOutage = read.csv(file = "/Users/fayreooi/Desktop/circuitWRegions.csv")
CircuitOutage$Outage.Duration..min. = as.numeric(gsub(",", "", CircuitOutage$Outage.Duration..min.))

# add information from circuit outage dataset
# we want to add the CMI from 2024

#add number of circuit outages in 2024
LookUp$Number.Outages.2024 = c(3, 4, 6, 4, 7, 3, 6, 3, 10, 2, 9, 3, 9, 1, 2, 1, 7, 2, 10, 1, 7)

## SAIDI for 2024

tapply(CircuitOutage$Outage.Duration..min., CircuitOutage$Circuit.Name, sum)
```

```
##      Adams    Alabama   Blue Jay      Dinan    Gorilla      Grand      Green
##       1537       2073        411       4994       5316        975       5514
##     Hoover  Jefferson    Johnson  Lightning    Lincoln      Logan    Magenta
##       5918       2746       1981         72       4172       6811       7597
##   Monterey     Orange     Oregon  Roosevelt    Thunder Washington     Yellow
##       1231       2521       1883       9225       2683       2937       4901
```

```
LookUp$Avg.Outage.Duration.2024 = c(840.333, 518.25, 919, 470.75, 700.1429, 979, 695.333,
                                    660.333, 759.7, 768.5, 1025, 915.333, 590.6667,
                                    411, 1341.5, 72, 845.4286, 615.5, 681.1, 975,
                                    713.4286)

# SAIFI for 2024
```

```
tapply(CircuitOutage$Customers.Affected, CircuitOutage$Circuit.Name, mean)
```

```
##      Adams    Alabama   Blue Jay      Dinan    Gorilla      Grand      Green
##   477.0000  2465.0000  1962.0000  1512.5714  1203.1111   243.0000  1683.1667
##     Hoover  Jefferson    Johnson  Lightning    Lincoln      Logan    Magenta
##   718.5714   883.0000   929.6667  1201.0000   627.1667  1412.2000  1425.1000
##   Monterey     Orange     Oregon  Roosevelt    Thunder Washington     Yellow
##   497.0000   845.6667   472.7500   246.0000  1658.0000   817.6667   746.7143
```

```
LookUp$Avg.Customers.Affected.2024 = c(846, 2465, 1684, 473 ,747, 818, 628, 930, 1426,
                                       477, 246, 883, 1204, 1962, 1658, 1201, 719,
                                       497,1413, 243, 1513)

LookUp$SAIFI.2024 = LookUp$Number.Outages.2024 / LookUp$Customer.Count

# add column of how many miles are overhead/underground

LookUp$Overhead.miles = LookUp$Circuit.Miles * (LookUp$X..Overhead/100)
LookUp$Underground.miles = LookUp$Circuit.Miles * (LookUp$X..Underground/100)

# adding cause count to Look Up data

LookUp$UG.Equipment.Failure = c(0,3,4,0,0,0,3,0,0,0,4,1,1,0,1,0,4,1,0,0,2)
LookUp$OH.Equipment.Failure = c(0,0,0,1,2,0,0,1,3,0,0,0,4,0,0,0,0,0,4,1,1)
LookUp$Third.Party = c(1,1,0,1,0,1,2,1,1,0,0,1,1,0,1,0,0,0,1,0,2)
LookUp$Weather = c(1,0,1,2,1,1,0,0,3,0,1,0,1,0,0,0,0,0,1,0,0)
LookUp$Other = c(1,0,1,0,1,0,0,0,1,0,1,0,2,0,0,1,1,1,0,0,1)
LookUp$Operation = c(0,0,0,0,2,0,1,0,1,1,2,0,0,0,0,0,1,0,2,0,0)
LookUp$Animal = c(0,0,0,0,0,1,0,0,1,0,1,0,0,1,0,0,1,0,1,0,1)
LookUp$Vegetation = c(0,0,0,0,1,0,0,1,0,1,0,1,0,0,0,0,0,0,1,0,0)

# make KV into categories
# LookUp$KV = factor(LookUp$KV,
#                       levels = c(4, 12, 16),
#                       labels = c("Four", "Twelve", "Sixteen"))

# add TOTAL outage duration per circuit in 2024

tapply(CircuitOutage$Outage.Duration..min. , CircuitOutage$Circuit.Name, sum)
```

```
##      Adams    Alabama   Blue Jay      Dinan    Gorilla      Grand      Green
##       1537       2073        411       4994       5316        975       5514
##     Hoover  Jefferson    Johnson  Lightning    Lincoln      Logan    Magenta
##       5918       2746       1981         72       4172       6811       7597
##   Monterey     Orange     Oregon  Roosevelt    Thunder Washington     Yellow
##       1231       2521       1883       9225       2683       2937       4901
```

```
LookUp$Total.Outage.Duration = c(2521, 2073, 5514, 1883, 4901, 2937, 4172, 1981,
                                 7597, 1537,9225, 2746, 5316, 411, 2683, 71, 5918,
                                 1231, 6811,975,4994)

LookUp$SAIDI.2024 = LookUp$Total.Outage.Duration / LookUp$Customer.Count

# add average minutes per customer for 2024
LookUp$Avg.Min.PerCustomer.2024 = LookUp$SAIDI.2024 * 60
```

```r
# KV-miles
LookUp$KV.Miles = LookUp$Circuit.Miles * LookUp$KV

write.csv(LookUp, "ihateitc.csv", row.names = FALSE)
```

```r
SAIFI = LookUp[, c("Circuit.Name","X2023.SAIFI", "SAIFI.2024")]
SAIFI
```

```
##    Circuit.Name X2023.SAIFI   SAIFI.2024
## 1        Orange       0.002 0.0023094688
## 2       Alabama       0.001 0.0011604294
## 3         Green       0.002 0.0022213995
## 4        Oregon       0.002 0.0034873583
## 5        Yellow       0.005 0.0041031653
## 6    Washington       0.002 0.0022865854
## 7       Lincoln       0.004 0.0036719706
## 8       Johnson       0.001 0.0018281536
## 9       Magenta       0.006 0.0042247571
## 10        Adams       0.001 0.0012143291
## 11    Roosevelt       0.012 0.0204081633
## 12    Jefferson       0.003 0.0021216407
## 13      Gorilla       0.004 0.0034090909
## 14     Blue Jay       0.001 0.0003262643
## 15      Thunder       0.001 0.0005069708
## 16    Lightning       0.001 0.0002784740
## 17       Hoover       0.003 0.0026819923
## 18     Monterey       0.001 0.0008354219
## 19        Logan       0.002 0.0031055901
## 20        Grand       0.001 0.0020000000
## 21        Dinan       0.001 0.0026923077
```

```r
write.csv(SAIFI, "SAIFI.csv")

SAIDI = LookUp[, c("Circuit.Name", "X2023.SAIDI", "SAIDI.2024")]
SAIDI
```

```
##    Circuit.Name X2023.SAIDI  SAIDI.2024
## 1        Orange        1.68  1.94072363
## 2       Alabama        0.57  0.60139252
## 3         Green        2.40  2.04146612
## 4        Oregon        1.39  1.64167393
## 5        Yellow        2.20  2.87280188
## 6    Washington        1.79  2.23856707
## 7       Lincoln        2.40  2.55324357
## 8       Johnson        0.91  1.20719074
## 9       Magenta        3.77  3.20954795
## 10        Adams        1.08  0.93321190
## 11    Roosevelt       13.65 20.91836735
## 12    Jefferson        1.88  1.94200849
## 13      Gorilla        1.63  2.01363636
## 14     Blue Jay        0.13  0.13409462
## 15      Thunder        0.74  0.68010139
## 16    Lightning        0.02  0.01977165
## 17       Hoover        2.18  2.26743295
## 18     Monterey        0.62  0.51420217
```

```
## 19        Logan        1.75  2.11521739
## 20        Grand        1.73  1.95000000
## 21        Dinan        2.09  1.92076923
```

```
write.csv(SAIDI, "SAIDI.csv")

Customers.Affected = LookUp[, c("Circuit.Name","Avg.Min.PerCustomer.2024", "Avg.Min.PerCustomer.2023")]
Customers.Affected
```

```
##    Circuit.Name Avg.Min.PerCustomer.2024 Avg.Min.PerCustomer.2023
## 1        Orange               116.443418                     100.8
## 2       Alabama                36.083551                      34.2
## 3         Green               122.487967                     144.0
## 4        Oregon                98.500436                      83.4
## 5        Yellow               172.368113                     132.0
## 6    Washington               134.314024                     107.4
## 7       Lincoln               153.194614                     144.0
## 8       Johnson                72.431444                      54.6
## 9       Magenta               192.572877                     226.2
## 10        Adams                55.992714                      64.8
## 11    Roosevelt              1255.102041                     819.0
## 12    Jefferson               116.520509                     112.8
## 13      Gorilla               120.818182                      97.8
## 14     Blue Jay                 8.045677                       7.8
## 15      Thunder                40.806084                      44.4
## 16    Lightning                 1.186299                       1.2
## 17       Hoover               136.045977                     130.8
## 18     Monterey                30.852130                      37.2
## 19        Logan               126.913043                     105.0
## 20        Grand               117.000000                     103.8
## 21        Dinan               115.246154                     125.4
```

```
library(leaps)

LookUp.Numerical.1 = LookUp[sapply(LookUp, is.numeric)]
LookUp.Numerical.1 = LookUp.Numerical.1[, -c(4,5,6,7,8,9,13,17,18,19,20,21,22,23,24,25)]
LookUp.Numerical.1 = LookUp.Numerical.1[, -c(5)]
LookUp.Numerical.1 = LookUp.Numerical.1[, -c(9)]

## FOR 2024
cor(LookUp.Numerical.1, use = "complete.obs", method = "pearson")
```

```
##                              Circuit.Number         KV Customer.Count
## Circuit.Number                  1.000000000 0.09376495     0.24421509
## KV                              0.093764951 1.00000000     0.28217882
## Customer.Count                  0.244215087 0.28217882     1.00000000
## Number.Outages.2024            -0.005259274 0.32224063     0.01167555
## Avg.Customers.Affected.2024    -0.087707627 0.29755351     0.79564517
## Overhead.miles                  0.071960686 0.47760752     0.20141578
## Underground.miles               0.270615159 0.53670576     0.11173124
## UG.Equipment.Failure           -0.068649631 0.19452259     0.06777295
## KV.Miles                        0.171698715 0.75255534     0.23350441
##                              Number.Outages.2024 Avg.Customers.Affected.2024
## Circuit.Number                       -0.005259274                 -0.08770763
## KV                                    0.322240634                  0.29755351
## Customer.Count                        0.011675552                  0.79564517
```

```
## Number.Outages.2024                 1.000000000                  0.05821609
## Avg.Customers.Affected.2024         0.058216086                  1.00000000
## Overhead.miles                      0.077745619                  0.16456135
## Underground.miles                   0.434788861                  0.00635884
## UG.Equipment.Failure                0.360491789                  0.10930331
## KV.Miles                            0.206690558                  0.17845602
##                              Overhead.miles Underground.miles
## Circuit.Number                   0.07196069        0.27061516
## KV                               0.47760752        0.53670576
## Customer.Count                   0.20141578        0.11173124
## Number.Outages.2024              0.07774562        0.43478886
## Avg.Customers.Affected.2024      0.16456135        0.00635884
## Overhead.miles                   1.00000000        0.38461796
## Underground.miles                0.38461796        1.00000000
## UG.Equipment.Failure            -0.05811670        0.21811838
## KV.Miles                         0.90346332        0.63795635
##                              UG.Equipment.Failure   KV.Miles
## Circuit.Number                         -0.06864963 0.17169872
## KV                                      0.19452259 0.75255534
## Customer.Count                          0.06777295 0.23350441
## Number.Outages.2024                     0.36049179 0.20669056
## Avg.Customers.Affected.2024             0.10930331 0.17845602
## Overhead.miles                         -0.05811670 0.90346332
## Underground.miles                       0.21811838 0.63795635
## UG.Equipment.Failure                    1.00000000 0.01707542
## KV.Miles                                0.01707542 1.00000000
```

```
# choosing number of variables
sub_sel = regsubsets(Number.Outages.2024 ~ ., data = LookUp.Numerical.1, nvmax = 8)
sub_res = summary(sub_sel)

summary(sub_sel)
```

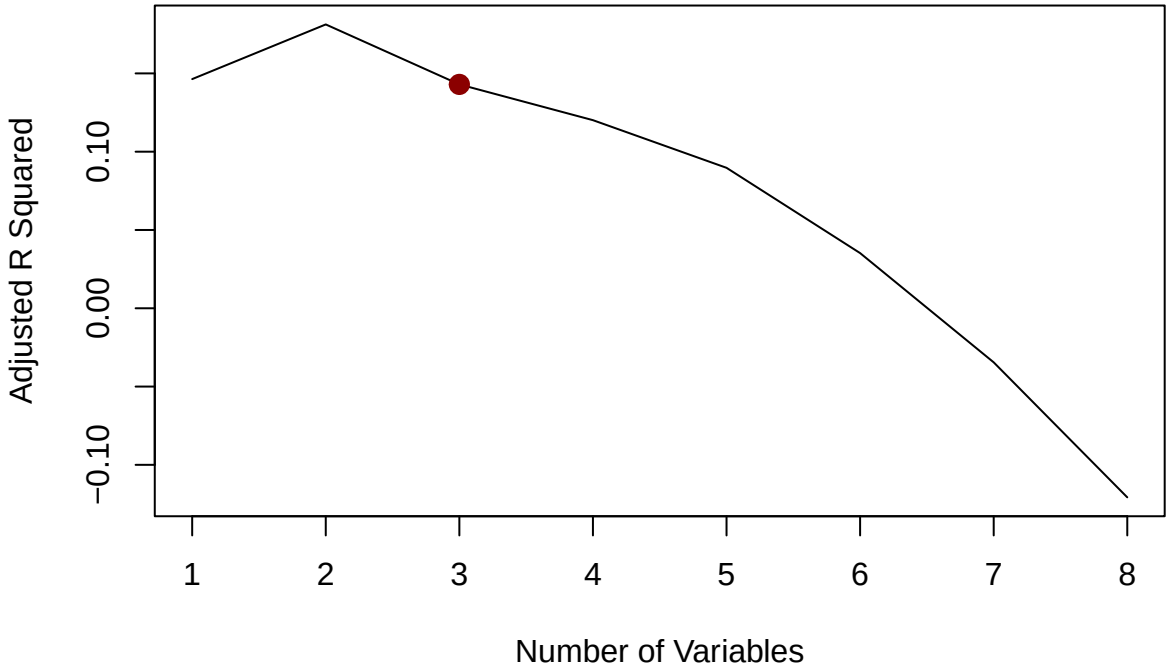
```
## Subset selection object
## Call: regsubsets.formula(Number.Outages.2024 ~ ., data = LookUp.Numerical.1,
##     nvmax = 8)
## 8 Variables  (and intercept)
##                              Forced in Forced out
## Circuit.Number                   FALSE      FALSE
## KV                               FALSE      FALSE
## Customer.Count                   FALSE      FALSE
## Avg.Customers.Affected.2024      FALSE      FALSE
## Overhead.miles                   FALSE      FALSE
## Underground.miles                FALSE      FALSE
## UG.Equipment.Failure             FALSE      FALSE
## KV.Miles                         FALSE      FALSE
## 1 subsets of each size up to 8
## Selection Algorithm: exhaustive
##          Circuit.Number KV  Customer.Count Avg.Customers.Affected.2024
## 1  ( 1 ) " "            " " " " " "            " "
## 2  ( 1 ) " "            " " " " " "            " "
## 3  ( 1 ) "*"            " " " " " "            " "
## 4  ( 1 ) " "            "*" " " " "            " "
## 5  ( 1 ) " "            "*" " " " "            " "
## 6  ( 1 ) " "            "*" "*" "*"            " "
```

```
## 7  ( 1 ) " "             "*" "*"             "*"
## 8  ( 1 ) "*"             "*" "*"             "*"
##          Overhead.miles Underground.miles UG.Equipment.Failure KV.Miles
## 1  ( 1 ) " "             "*"               " "                 " "
## 2  ( 1 ) " "             "*"               "*"                 " "
## 3  ( 1 ) " "             "*"               "*"                 " "
## 4  ( 1 ) "*"             "*"               " "                 "*"
## 5  ( 1 ) "*"             "*"               "*"                 "*"
## 6  ( 1 ) "*"             "*"               "*"                 "*"
## 7  ( 1 ) "*"             "*"               "*"                 "*"
## 8  ( 1 ) "*"             "*"               "*"                 "*"
```

```
sub_res$rsq
```

```
## [1] 0.1890414 0.2631399 0.2715196 0.2960999 0.3172774 0.3246573 0.3275492
## [8] 0.3275516
```

```
plot(sub_res$adjr2, xlab = "Number of Variables", ylab = "Adjusted R Squared", type = "l")
points(3, sub_res$adjr2[3], col = "darkred", cex = 2, pch = 20)
```




```
# predicting

# FORWARD
regfit_fwd = regsubsets(Number.Outages.2024 ~ . , data = LookUp.Numerical.1,
                        nvmax = 7, method = "forward")
summary(regfit_fwd)
```

```
## Subset selection object
## Call: regsubsets.formula(Number.Outages.2024 ~ ., data = LookUp.Numerical.1,
##     nvmax = 7, method = "forward")
## 8 Variables  (and intercept)
##                            Forced in Forced out
## Circuit.Number                 FALSE      FALSE
## KV                             FALSE      FALSE
```

```
## Customer.Count                  FALSE      FALSE
## Avg.Customers.Affected.2024     FALSE      FALSE
## Overhead.miles                  FALSE      FALSE
## Underground.miles               FALSE      FALSE
## UG.Equipment.Failure            FALSE      FALSE
## KV.Miles                        FALSE      FALSE
## 1 subsets of each size up to 7
## Selection Algorithm: forward
##          Circuit.Number KV  Customer.Count Avg.Customers.Affected.2024
## 1  ( 1 ) " "            " " " "            " "
## 2  ( 1 ) " "            " " " "            " "
## 3  ( 1 ) "*"            " " " "            " "
## 4  ( 1 ) "*"            "*" " "            " "
## 5  ( 1 ) "*"            "*" " "            " "
## 6  ( 1 ) "*"            "*" " "            " "
## 7  ( 1 ) "*"            "*" "*"            " "
##          Overhead.miles Underground.miles UG.Equipment.Failure KV.Miles
## 1  ( 1 ) " "            "*"               " "                  " "
## 2  ( 1 ) " "            "*"               "*"                  " "
## 3  ( 1 ) " "            "*"               "*"                  " "
## 4  ( 1 ) " "            "*"               "*"                  " "
## 5  ( 1 ) " "            "*"               "*"                  "*"
## 6  ( 1 ) "*"            "*"               "*"                  "*"
## 7  ( 1 ) "*"            "*"               "*"                  "*"
```

```
# BACKWARD
regfit_bwd = regsubsets(Number.Outages.2024 ~ . , data = LookUp.Numerical.1,
                        nvmax = 7, method = "backward")
summary(regfit_bwd)
```

```
## Subset selection object
## Call: regsubsets.formula(Number.Outages.2024 ~ ., data = LookUp.Numerical.1,
##     nvmax = 7, method = "backward")
## 8 Variables  (and intercept)
##                              Forced in Forced out
## Circuit.Number                   FALSE      FALSE
## KV                               FALSE      FALSE
## Customer.Count                   FALSE      FALSE
## Avg.Customers.Affected.2024      FALSE      FALSE
## Overhead.miles                   FALSE      FALSE
## Underground.miles                FALSE      FALSE
## UG.Equipment.Failure             FALSE      FALSE
## KV.Miles                         FALSE      FALSE
## 1 subsets of each size up to 7
## Selection Algorithm: backward
##          Circuit.Number KV  Customer.Count Avg.Customers.Affected.2024
## 1  ( 1 ) " "            " " " "            " "
## 2  ( 1 ) " "            "*" " "            " "
## 3  ( 1 ) " "            "*" " "            " "
## 4  ( 1 ) " "            "*" " "            " "
## 5  ( 1 ) " "            "*" " "            " "
## 6  ( 1 ) " "            "*" "*"            " "
## 7  ( 1 ) " "            "*" "*"            "*"
##          Overhead.miles Underground.miles UG.Equipment.Failure KV.Miles
## 1  ( 1 ) " "            "*"               " "                  " "
```

```
## 2  ( 1 ) " "           "*"               " "                 " "
## 3  ( 1 ) " "           "*"               " "                 "*"
## 4  ( 1 ) "*"           "*"               " "                 "*"
## 5  ( 1 ) "*"           "*"               "*"                 "*"
## 6  ( 1 ) "*"           "*"               "*"                 "*"
## 7  ( 1 ) "*"           "*"               "*"                 "*"
```

```
# selecting model based on a criteria (BIC)
num_bic_fwd = which.min( summary(regfit_fwd)$bic)
num_bic_bwd = which.min( summary(regfit_bwd)$bic)
num_bic_best = which.min( summary(sub_sel)$bic)

paste("forward stepwise selection:", num_bic_fwd)
```

```
## [1] "forward stepwise selection: 1"
```

```
paste("forward stepwise selection:", num_bic_bwd)
```

```
## [1] "forward stepwise selection: 1"
```

```
paste("forward stepwise selection:", num_bic_best)
```

```
## [1] "forward stepwise selection: 1"
```

```
coef(regfit_fwd, num_bic_fwd)
```

```
##       (Intercept) Underground.miles
##        3.05083366        0.09588135
```

```
coef(regfit_bwd, num_bic_bwd)
```

```
##       (Intercept) Underground.miles
##        3.05083366        0.09588135
```

```
## saidi, saifi, and underground miles are the most relevant.
```

```
set.seed(1220)
train = sample(c(TRUE,FALSE), nrow(LookUp.Numerical.1), replace = TRUE)
test = (!train)

regfit_best = regsubsets(Number.Outages.2024 ~ ., data = LookUp.Numerical.1[train, ],
                         nvmax = 8)

test_mat = model.matrix(Number.Outages.2024 ~ ., data = LookUp.Numerical.1[test, ])
head(test_mat)
```

```
##    (Intercept) Circuit.Number KV Customer.Count Avg.Customers.Affected.2024
## 3            1              3 12           2701                         1684
## 4            1              4 16           1147                          473
## 7            1              7 16           1634                          628
## 8            1              8 16           1641                          930
## 11           1             11 12            441                          246
## 12           1             12 12           1414                          883
##    Overhead.miles Underground.miles UG.Equipment.Failure KV.Miles
## 3           14.30             11.70                    4      312
## 4           85.85             15.15                    0     1616
## 7           61.75             33.25                    3     1520
## 8           38.40             25.60                    0     1024
## 11          23.04              8.96                    4      384
```

```
## 12          19.36              2.64                     1      264
```

```
p = ncol(test_mat) - 1
val_errors = numeric(p)

for (i in 1:p){
  coef_i <- coef(regfit_best, id = i)
  pred <- test_mat[, names(coef_i)] %*% coef_i
  y_test <- LookUp.Numerical.1$Number.Outages.2024[test]
  val_errors[i] <- mean((y_test - pred)^2)
}

val_errors
```

```
## [1]   17.46314   75.76733   55.30024   65.08319  147.73045  143.82215 1104.70410
## [8] 1256.32442
```

```
which.min(val_errors)
```

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

```
coef(regfit_best, 6)
```

```
##          (Intercept)       Circuit.Number                   KV
##         -21.21692557           0.64491526           0.95511197
##       Overhead.miles    Underground.miles UG.Equipment.Failure
##           0.57422353           0.61512305           2.19718887
##             KV.Miles
##          -0.03162675
```