Classification

Vladimir Sokolov

10/11/2022

This notebook explores smoke detector data from Kaggle.

Load the smoke_detection_iot.csv. Simplify variables

```
df <- read.csv("smoke_detection_iot.csv")
df <- df[,c(3:14,16)]
df<-df[-c(8,9)]
str(df)
```

```
## 'data.frame':    62630 obs. of  11 variables:
##  $ Temperature.C.: num  20 20 20 20 20.1 ...
##  $ Humidity...   : num  57.4 56.7 56 55.3 54.7 ...
##  $ TVOC.ppb.     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ eCO2.ppm.     : int  400 400 400 400 400 400 400 400 400 400 ...
##  $ Raw.H2        : int  12306 12345 12374 12390 12403 12419 12432 12439 12448 12453 ...
##  $ Raw.Ethanol   : int  18520 18651 18764 18849 18921 18998 19058 19114 19155 19195 ...
##  $ Pressure.hPa. : num  940 940 940 940 940 ...
##  $ NC0.5         : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ NC1.0         : num  0 0 0 0 0 ...
##  $ NC2.5         : num  0 0 0 0 0 0 0 0 0 2.78 ...
##  $ Fire.Alarm    : int  0 0 0 0 0 0 0 0 0 0 ...
```

Check for null values.

```
sapply(df, function(x) sum(is.na(x)))
```

```
## Temperature.C.    Humidity...      TVOC.ppb.      eCO2.ppm.         Raw.H2
##              0              0              0              0              0
##    Raw.Ethanol  Pressure.hPa.          NC0.5          NC1.0          NC2.5
##              0              0              0              0              0
##     Fire.Alarm
##              0
```

#A.Divide into 80/20 train/test.

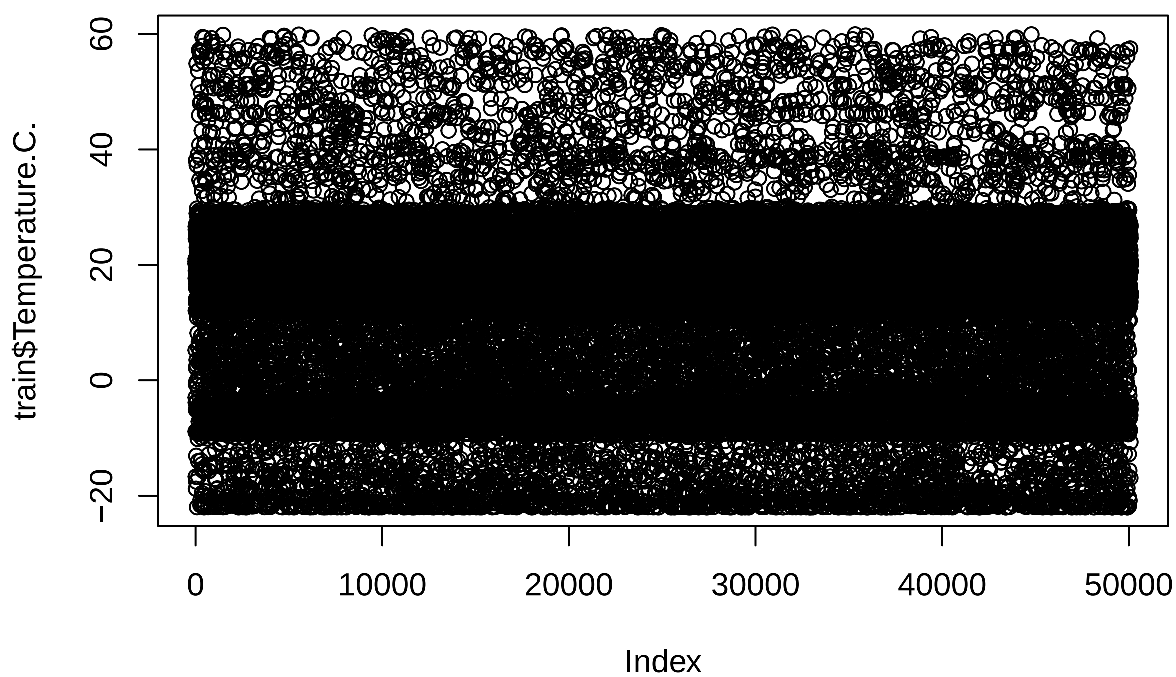
```
set.seed(12345)
i <- sample(1:nrow(df), nrow(df)*.8, replace=FALSE)
train <- df[i,]
test <- df[-i,]
```

#B.Explore data.

```
summary(train$Temperature.C.)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -22.01   10.91   20.15   15.93   25.41   59.93
```

```
plot(train$Temperature.C.)
```



#C.Regressions Linear Regression

```
lm1 <- lm(Fire.Alarm ~., data=train)
summary(lm1)
```

```
##
## Call:
## lm(formula = Fire.Alarm ~ ., data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.51359 -0.09061  0.05031  0.16666  1.00714
##
## Coefficients:
##                   Estimate Std. Error  t value Pr(>|t|)
## (Intercept)      1.008e+02  1.509e+00   66.806   <2e-16 ***
## Temperature.C. -3.444e-03  9.707e-05  -35.478   <2e-16 ***
```

```
## Humidity...     1.518e-02  2.696e-04   56.308   <2e-16 ***
## TVOC.ppb.      -2.993e-05  3.481e-07  -86.000   <2e-16 ***
## eCO2.ppm.       2.096e-05  1.173e-06   17.861   <2e-16 ***
## Raw.H2          6.415e-04  8.453e-06   75.891   <2e-16 ***
## Raw.Ethanol    -7.461e-04  3.521e-06 -211.887   <2e-16 ***
## Pressure.hPa.  -1.005e-01  1.641e-03  -61.217   <2e-16 ***
## NC0.5          -8.113e-01  8.370e-02   -9.694   <2e-16 ***
## NC1.0           5.401e+00  5.571e-01    9.694   <2e-16 ***
## NC2.5          -8.754e+00  9.031e-01   -9.694   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2977 on 50093 degrees of freedom
## Multiple R-squared:  0.5652, Adjusted R-squared:  0.5651
## F-statistic:  6511 on 10 and 50093 DF,  p-value: < 2.2e-16
```

```r
pred1 <- predict(lm1, newdata = test)
cor_lm1 <- cor(pred1, test$Fire.Alarm)
mse_lm1 <- mean((pred1-test$Fire.Alarm)^2)
print(paste("cor1=",cor_lm1))
```

```
## [1] "cor1= 0.760397709942583"
```

```r
print(paste("mse1=",mse_lm1))
```

```
## [1] "mse1= 0.086322708311021"
```

kNN Regression

```r
library(caret)
```

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

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

```r
fit <- knnreg(train[,1:10], train[,11],k=3)
pred2 <- predict(fit, test[1:10])
cor_knn1 <- cor(pred2, test$Fire.Alarm)
mse_knn1 <- mean((pred2 - test$Fire.Alarm)^2)
print(paste("cor2=", cor_knn1))
```

```
## [1] "cor2= 0.999761574023808"
```

```r
print(paste("mse2=", mse_knn1))
```

```
## [1] "mse2= 9.75748221477105e-05"
```

Decision tree regression

```r
library(tree)
tree1 <- tree(Fire.Alarm ~., data=train)
summary(tree1)
```

```
##
## Regression tree:
## tree(formula = Fire.Alarm ~ ., data = train)
## Variables actually used in tree construction:
## [1] "Pressure.hPa."  "TVOC.ppb."      "NC0.5"          "Temperature.C."
## Number of terminal nodes:  6
## Residual mean deviance:  0.006362 = 318.7 / 50100
## Distribution of residuals:
##       Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
## -0.9981000 -0.0002182  0.0018920  0.0000000  0.0018920  0.9998000
```

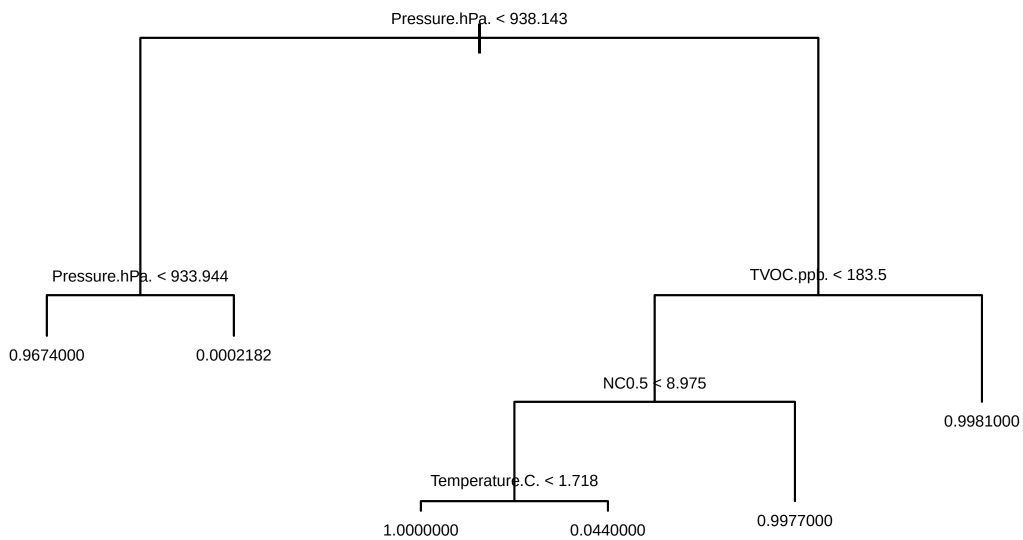
```r
pred3 <- predict(tree1, newdata=test)
cor_tree <- cor(pred3, test$Fire.Alarm)
mse_tree <- mean((pred3 - test$Fire.Alarm)^2)
print(paste("cor3=", cor_tree))
```

```
## [1] "cor3= 0.985649579620211"
```

```r
print(paste("mse3=", mse_tree))
```

```
## [1] "mse3= 0.00583144627210529"
```

```r
plot(tree1)
text(tree1, cex=0.5, pretty=0)
```



Comparison

```
print(paste("corlr=",cor_lm1))
```

```
## [1] "corlr= 0.760397709942583"
```

```
print(paste("corknn=", cor_knn1))
```

```
## [1] "corknn= 0.999761574023808"
```

```
print(paste("cortree=", cor_tree))
```

```
## [1] "cortree= 0.985649579620211"
```

#D.Analysis knn performed the best followed by decision trees and then linear regression. This is probably due to the data set being easily separated into fire detected or not which was favored by them.