Disaster Relief Project Part 1

Yihnew Eshetu

2/26/2020

```
library(ISLR)
library(regclass)
library(caret)
library(MASS)
library(class)
library(pROC)
library(ROCR)
library(nnet)
```

Read in Haiti csv file

```
haiti.image = read.csv("HaitiPixels.csv")
attach(haiti.image)
```

```
set.seed(10)
data.size = nrow(haiti.image)
split.size = data.size/10
permutation = sample(data.size)
```

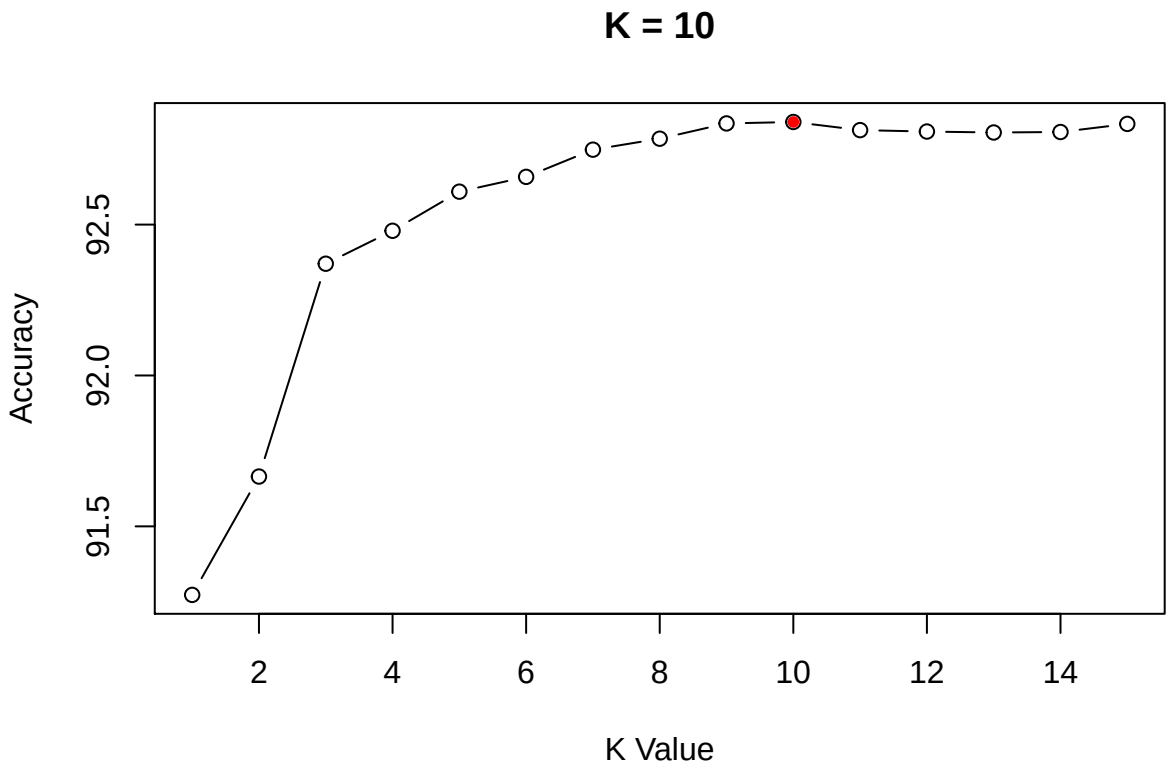
```
knn.function = function(kvalue) {
  knn.acc = 0
  for (i in 1:10) {
    haiti.image.sample = permutation[((i - 1) * split.size + 1) : (i * split.size)]

    haiti.image.test = as.data.frame(haiti.image[haiti.image.sample,])
    haiti.image.train = as.data.frame(haiti.image[-haiti.image.sample,])

    knn.pred = knn(haiti.image.train[-1], haiti.image.test[-1], cl =  haiti.image.train[,1], kvalue)
    knn.acc = knn.acc + mean(knn.pred == haiti.image.test[,1])
  }
  return((knn.acc/10)*100)
}

k.value.acc = c('1' = 0, '2' = 0, '3' = 0,
                  '4' = 0, '5' = 0, '6' = 0,
                  '7' = 0, '8' = 0, '9' = 0,
                  '10' = 0, '11' = 0, '12' = 0,
                  '13' = 0, '13' = 0, '15' = 0)

start_time <- Sys.time()
for (k in 1:15){
  k.value.acc[k] = knn.function(k)
}
end_time <- Sys.time()
end_time - start_time
```

```
## Time difference of 4.701939 mins
```

```
k.value.acc[which.max(k.value.acc)]
```

```
##       10
## 92.83997
```

```
plot(seq(1,15), k.value.acc,type = 'b', xlab = 'K Value', ylab = 'Accuracy', main = paste0('K = ', which
points(which.max(k.value.acc), k.value.acc[which.max(k.value.acc)], col ='red', pch = 20)
```




```
lda.function = function() {
  lda.acc = 0
  for (i in 1:10) {
    haiti.image.sample = permutation[((i - 1) * split.size + 1) : (i * split.size)]

    haiti.image.test = as.data.frame(haiti.image[haiti.image.sample,])
    haiti.image.train = as.data.frame(haiti.image[-haiti.image.sample,])

    lda.fit = lda(Class ~ Red + Green + Blue, data = haiti.image.train)
    lda.pred = predict(lda.fit, haiti.image.test)
    lda.acc = lda.acc + mean(lda.pred$class == haiti.image.test[,1])
  }
  return((lda.acc/10)*100)
}
start_time <- Sys.time()
lda.function()
```

```
## [1] 85.62619
```

```
end_time <- Sys.time()
end_time - start_time
```

```
## Time difference of 1.470174 secs
```

```
qda.function = function() {
  qda.acc = 0
  for (i in 1:10) {
    haiti.image.sample = permutation[((i - 1) * split.size + 1) : (i * split.size)]

    haiti.image.test = as.data.frame(haiti.image[haiti.image.sample,])
    haiti.image.train = as.data.frame(haiti.image[-haiti.image.sample,])

    qda.fit = qda(Class ~ Red + Green + Blue, data = haiti.image.train)
    qda.pred = predict(qda.fit, haiti.image.test)
    qda.acc = qda.acc + mean(qda.pred$class == haiti.image.test[,1])
  }
  return((qda.acc/10)*100)
}
start_time <- Sys.time()
qda.function()
```

```
## [1] 90.12018
```

```
end_time <- Sys.time()
end_time - start_time
```

```
## Time difference of 1.626837 secs
```

```
log.function = function() {
  log.acc = 0
  for (i in 1:10) {
    haiti.image.sample = permutation[((i - 1) * split.size + 1) : (i * split.size)]

    haiti.image$Class = factor(haiti.image$Class)
    levels(haiti.image$Class)
    haiti.image.test = haiti.image[haiti.image.sample,]
    haiti.image.train = haiti.image[-haiti.image.sample,]

    log.fit = multinom(Class ~ Red + Green + Blue, data = haiti.image.train, family = binomial)
    log.pred = predict(log.fit, type = 'class', haiti.image.test)
    log.acc = log.acc + mean(log.pred == haiti.image.test[,1])
  }
  return((log.acc/10)*100)
}
start_time <- Sys.time()
log.function()
```

```
## # weights:  25 (16 variable)
## initial  value 91604.377662
## iter  10 value 66226.318525
## iter  20 value 26064.702753
## iter  30 value 21509.378979
## iter  40 value 20995.074518
## iter  50 value 20774.201597
## final  value 20760.665601
## converged
## # weights:  25 (16 variable)
## initial  value 91604.377662
## iter  10 value 66141.917880
```

```
## iter  20 value 29261.672900
## iter  30 value 21589.317865
## iter  40 value 20964.425730
## iter  50 value 20769.914350
## final  value 20726.544511
## converged
## # weights:  25 (16 variable)
## initial  value 91604.377662
## iter  10 value 66313.954420
## iter  20 value 25492.695599
## iter  30 value 21432.147117
## iter  40 value 20989.167481
## iter  50 value 20839.630389
## final  value 20801.041489
## converged
## # weights:  25 (16 variable)
## initial  value 91604.377662
## iter  10 value 65998.486514
## iter  20 value 25054.742514
## iter  30 value 21487.613879
## iter  40 value 21032.456991
## iter  50 value 20686.018000
## final  value 20663.849430
## converged
## # weights:  25 (16 variable)
## initial  value 91604.377662
## iter  10 value 66271.563352
## iter  20 value 26059.079485
## iter  30 value 21593.449114
## iter  40 value 21186.884418
## iter  50 value 20906.029434
## final  value 20813.028262
## converged
## # weights:  25 (16 variable)
## initial  value 91604.377662
## iter  10 value 66086.261464
## iter  20 value 26130.357493
## iter  30 value 21589.830181
## iter  40 value 21153.857524
## iter  50 value 20924.235945
## final  value 20860.309138
## converged
## # weights:  25 (16 variable)
## initial  value 91604.377662
## iter  10 value 66132.168898
## iter  20 value 26020.554010
## iter  30 value 21466.147492
## iter  40 value 20905.121697
## iter  50 value 20760.572247
## final  value 20718.981207
## converged
## # weights:  25 (16 variable)
## initial  value 91604.377662
## iter  10 value 67881.661835
```

```
## iter  20 value 32006.741657
## iter  30 value 21698.559479
## iter  40 value 21220.122210
## iter  50 value 20767.500728
## final  value 20668.822604
## converged
## # weights:  25 (16 variable)
## initial  value 91604.377662
## iter  10 value 66169.475489
## iter  20 value 25634.123668
## iter  30 value 21419.122001
## iter  40 value 21012.751204
## iter  50 value 20846.905732
## final  value 20834.881928
## converged
## # weights:  25 (16 variable)
## initial  value 91604.377662
## iter  10 value 66171.536811
## iter  20 value 26026.687025
## iter  30 value 21614.353776
## iter  40 value 21107.523797
## iter  50 value 20820.398142
## final  value 20798.015381
## converged
```

```
## [1] 88.63694
```

```
end_time <- Sys.time()
end_time - start_time
```

```
## Time difference of 24.02256 secs
```