OBA 455-555 Final Report
Maddie Frost, Ryan Maki, Ari Nguyen

KNN REGRESSION

```
library(tidyverse)
library(FNN)
rm(list = ls())
##SESSION
## Working Directory
salary_new1 = read_csv("salary (1).csv")
##MISSING VALUES
data_omit = na.omit(salary_new1)
salary_new = data_omit

salary_new = salary_new %>%
  select(age, workclass, education, marital.status, relationship, race, sex, hpw,
native.country.new, salary)

salary_new = salary_new %>%
  rename(salary_actual = salary) %>%
  mutate(salary_actual = factor(salary_actual, c("<=50K", ">50K")))

salary_new = salary_new %>%
  mutate(salary_greater = if_else(salary_actual == "<=50K", 1, 0))

salary_new = salary_new %>%
  mutate(education = factor(education,c("Bachelors", "Some-college", "11th", "HS-grad",
                              "Prof-school", "Assoc-acdm", "Assoc-voc", "9th",
                              "7th-8th", "12th", "Masters", "1st-4th", "10th",
                              "Doctorate", "5th-6th", "Preschool")))
salary_new = salary_new %>%
  mutate(higher_edu = if_else(education == "Bachelors", 1,
                   if_else(education == "Some-college", 1,
                   if_else(education == "Masters", 1,
                   if_else(education == "Doctorate", 1,
                   if_else(education == "Prof-school", 1,
                   if_else(education == "Assoc-acdm", 1,
                   if_else(education == "Assoc-voc", 1,0))))))))

salary_new = salary_new %>%
  mutate(workclass = factor(workclass,c("Private", "Self-emp-not-inc", "Self-emp-inc",
                              "Federal-gov", "Local-gov", "State-gov",
                              "Without-pay", "Never-worked")))
```

```
salary_new = salary_new %>%
  mutate(job_employed = if_else (workclass == "Private", 1,
                      if_else(workclass == "Self-emp-not-inc", 1,
                      if_else(workclass == "Federal-gov", 1,
                      if_else(workclass == "Local-gov", 1,
                      if_else (workclass == "State-gov", 1, 0))))))

salary_new = salary_new %>%
  mutate(marital.status=factor(marital.status,c("Married-civ-spouse", "Divorced",
                                                "Never-married", "Separated", "Widowed",
                                                "Married-spouse-absent", "Married-AF-spouse")))

salary_new = salary_new %>%
mutate(marital.status_single = if_else(marital.status == "Divorced", 1,
                                       if_else(marital.status == "Never-married", 1,
                                      if_else(marital.status == "Separated", 1,0))))

salary_new = salary_new %>%
  mutate(relationship = factor(relationship, c("Wife", "Own-child", "Husband",
                                            "Not-in-family", "Other-relative",
                                            "Unmarried")))
salary_new = salary_new %>%
  mutate(in_unit = if_else(relationship == "Wife", 1,
                   if_else(relationship == "Husband", 1,
                   if_else(relationship == "Own-child", 1,
                   if_else(relationship == "Other-relative", 1, 0)))))

salary_new = salary_new %>%
  mutate(race = factor(race, c("White", "Asian-Pac-Islander", "Amer-Indian-Eskimo",
                             "Other", "Black")))
salary_new = salary_new %>%
  mutate(race_black = if_else(race == "Black", 1, 0),
        race_white = if_else(race == "White", 1, 0),
        race_API = if_else(race == "Asian-Pac-Islander", 1, 0),
        race_AIE = if_else(race == "Amer-Indian-Eskimo", 1, 0),
        race_other = if_else(race == "Other", 1, 0))

salary_new = salary_new %>%
  mutate(sex = factor(sex, c("Female", "Male")))

salary_new = salary_new %>%
  mutate(sex_M = if_else(sex == "Male", 1,0))
```

```
salary_new = salary_new %>%
  mutate(Country = case_when(native.country.new %in% c("United-States") ~ "US",
                             native.country.new %in% c("Cambodia", "England",
                                                        "Puerto-Rico", "Canada",
                                                        "Germany", "Outlying-US(Guam-USVI-etc)",
                                                        "India", "Japan", "Greece", "South", "China",
                                                        "Cuba", "Iran", "Honduras", "Philippines",
                                                        "Italy", "Poland", "Jamaica", "Vietnam",
                                                        "Mexico", "Portugal","Ireland",
                                                        "France", "Dominican-Republic",
                                                        "Laos", "Ecuador", "Taiwan",
                                                        "Haiti", "Columbia", "Hungary",
                                                        "Guatemala", "Nicaragua", "Scotland",
                                                        "Thailand", "Yugoslavia", "El-Salvador",
                                                        "Trinadad&Tobago", "Peru", "Hong",
                                                        "Holand-Netherlands", "NA")~ "Not-US"))%>%
  mutate(native.country.new = factor(native.country.new, c("United-States", "Cambodia",
"England",
                                                         "Puerto-Rico", "Canada", "Germany",
                                                         "Outlying-US(Guam-USVI-etc)", "India",
                                                         "Japan", "Greece", "South", "China",
                                                         "Cuba", "Iran", "Honduras", "Philippines",
                                                         "Italy", "Poland", "Jamaica", "Vietnam",
                                                         "Mexico", "Portugal", "Ireland", "France",
                                                         "Dominican-Republic", "Laos", "Ecuador",
                                                         "Taiwan", "Haiti", "Columbia", "Hungary", "Guatemala",
                                                         "Nicaragua", "Scotland", "Thailand", "Yugoslavia",
                                                         "El-Salvador", "Trinadad&Tobago", "Peru", "Hong",
                                                         "Holand-Netherlands","NA")))

salary_new = salary_new %>%
  mutate(US = if_else(Country == "US", 1, 0))

salary_new = na.omit(salary_new)

salary_new = salary_new %>%
  mutate(id = 1:nrow(salary_new))

library(FNN)
##KNN REGRESSION
##Step 1: Main Data
a1 = mean(salary_new$age)

a2 = sd(salary_new$age)
```

```
b1 = mean(salary_new$hpw)

b2 = sd(salary_new$hpw)

##STANDARDIZE
salary_new = salary_new %>%
  mutate(age_norm = (age - a1)/a2, hpw_norm = (hpw - b1)/b2)

salary_new = salary_new %>%
  mutate(race_black = if_else(race == "Black", 1, 0),
         race_white = if_else(race == "White", 1, 0),
         race_API = if_else(race == "Asian-Pac-Islander", 1, 0),
         race_AIE = if_else(race == "Amer-Indian-Eskimo", 1, 0),
         race_other = if_else(race == "Other", 1, 0),
         sex_M = if_else(sex == "Male", 1,0))

salary_new = salary_new %>%
  mutate(single = if_else(marital.status == "Divorced", 1,
        if_else(marital.status == "Never-married", 1,if_else(marital.status
        == "Separated", 1,0))))

salary_new = salary_new %>%
  mutate(job_employed = if_else (workclass == "Private", 1,
                              if_else(workclass == "Self-emp-not-inc", 1,
                                      if_else(workclass == "Federal-gov", 1,
                                              if_else(workclass == "Local-gov", 1,
                                                      if_else (workclass == "State-gov", 1, 0))))))

salary_new = salary_new %>%
  mutate(higher_edu = if_else(education == "Bachelors", 1,
                           if_else(education == "Some-college", 1,
                                   if_else(education == "Masters", 1,
                                           if_else(education == "Doctorate", 1,
                                                   if_else(education == "Prof-school", 1,
                                                           if_else(education == "Assoc-acdm", 1,
                                                                   if_else(education == "Assoc-voc", 1,0))))))))

salary_new = salary_new %>%
  mutate(salary_greater = if_else(salary_actual == "<=50K", 1, 0))

salary_new = salary_new %>%
  mutate(in_unit = if_else(relationship == "Wife", 1,
                        if_else(relationship == "Husband", 1,
                    if_else(relationship == "Own-child", 1,
                        if_else(relationship == "Other-relative", 1, 0)))))

##STEP 2: PICK STANDARDIZED
salary_new_input_norm = salary_new %>%
  select(age_norm, hpw_norm, race_black, race_white, race_API, race_AIE,
      race_other, sex_M, single, job_employed, higher_edu,
      salary_greater, in_unit)

##STEP 3: NEW DATA
newdata = as_tibble(list(age = 37, hpw = 40,race_black = 0, race_white = 1,
              race_API = 0, race_AIE = 0, race_other = 0, sex_M = 0,
              single = 0, job_employed = 1, higher_edu = 1,
              in_unit = 1, salary_greater = 1))

##STANDARDIZE
newdata = newdata %>%
  mutate(age_norm = (age - a1)/a2, hpw_norm = (hpw - b1)/b2)

salary_new = salary_new %>%
  mutate(race_black = if_else(race == "Black", 1, 0),
      race_white = if_else(race == "White", 1, 0),
      race_API = if_else(race == "Asian-Pac-Islander", 1, 0),
      race_AIE = if_else(race == "Amer-Indian-Eskimo", 1, 0),
      race_other = if_else(race == "Other", 1, 0),
      sex_M = if_else(sex == "Male", 1,0))

salary_new = salary_new %>%
  mutate(single = if_else(marital.status == "Divorced", 1,
              if_else(marital.status == "Never-married", 1,if_else(marital.status
                                == "Separated", 1,0))))
salary_new = salary_new %>%
  mutate(job_employed = if_else (workclass == "Private", 1,
                  if_else(workclass == "Self-emp-not-inc", 1,
                      if_else(workclass == "Federal-gov", 1,
                          if_else(workclass == "Local-gov", 1,
                              if_else (workclass == "State-gov", 1, 0))))))

salary_new = salary_new %>%
  mutate(higher_edu = if_else(education == "Bachelors", 1,
                if_else(education == "Some-college", 1,
                    if_else(education == "Masters", 1,
                        if_else(education == "Doctorate", 1,
                            if_else(education == "Prof-school", 1,
                               if_else(education == "Assoc-acdm", 1,
                                   if_else(education == "Assoc-voc", 1,0))))))))

salary_new = salary_new %>%
  mutate(salary_greater = if_else(salary_actual == "<=50K", 1, 0))

salary_new = salary_new %>%
  mutate(in_unit = if_else(relationship == "Wife", 1,
             if_else(relationship == "Husband", 1,
                 if_else(relationship == "Own-child", 1,
                     if_else(relationship == "Other-relative", 1, 0)))))

#STEP 4: PICK STANDARDIZED
newdata_input_norm = newdata %>%
  select(age_norm, hpw_norm, race_black, race_white, race_API, race_AIE,
      race_other, sex_M, single, job_employed, higher_edu,salary_greater,
      in_unit)
salary_new_output = salary_new$salary_greater

##STEP 5: KNN CLASSIFIER CODE
knn(salary_new_input_norm, newdata_input_norm, salary_new_output, 1)
knn(salary_new_input_norm, newdata_input_norm, salary_new_output, 2)
knn(salary_new_input_norm, newdata_input_norm, salary_new_output, 3)
knn(salary_new_input_norm, newdata_input_norm, salary_new_output, 4)
knn(salary_new_input_norm, newdata_input_norm, salary_new_output, 5)

###TRAIN & VALIDATION
rm(list = ls())
set.seed(30)
train = salary_new %>%
  sample_frac(0.7)
validation = salary_new %>%
  slice(setdiff(salary_new$id, train$id))

##STEP 1: TRAIN
a1 = mean(train$age)

a2 = sd(train$age)

b1 = mean(train$hpw)

b2 = sd(train$hpw)

train = train %>%
  mutate(age_norm = (age - a1)/a2, hpw_norm = (hpw - b1)/b2)
train = train %>%
  mutate(race_black = if_else(race == "Black", 1, 0),
      race_white = if_else(race == "White", 1, 0),
      race_API = if_else(race == "Asian-Pac-Islander", 1, 0),
      race_AIE = if_else(race == "Amer-Indian-Eskimo", 1, 0),
      race_other = if_else(race == "Other", 1, 0),
      sex_M = if_else(sex == "Male", 1,0))

salary_new = salary_new %>%
  mutate(single = if_else(marital.status == "Divorced", 1,
              if_else(marital.status == "Never-married",
              1,if_else(marital.status == "Separated", 1,0))))

salary_new = salary_new %>%
  mutate(job_employed = if_else (workclass == "Private", 1,
                  if_else(workclass == "Self-emp-not-inc", 1,
                      if_else(workclass == "Federal-gov", 1,
                          if_else(workclass == "Local-gov", 1,
                              if_else (workclass == "State-gov", 1, 0))))))

salary_new = salary_new %>%
  mutate(higher_edu = if_else(education == "Bachelors", 1,
                if_else(education == "Some-college", 1,
                    if_else(education == "Masters", 1,
                        if_else(education == "Doctorate", 1,
                            if_else(education == "Prof-school", 1,
                                if_else(education == "Assoc-acdm", 1,
                                    if_else(education == "Assoc-voc", 1,0))))))))

salary_new = salary_new %>%
  mutate(salary_greater = if_else(salary_actual == "<=50K", 1, 0))

salary_new = salary_new %>%
  mutate(in_unit = if_else(relationship == "Wife", 1,
              if_else(relationship == "Husband", 1,
                  if_else(relationship == "Own-child", 1,
                      if_else(relationship == "Other-relative", 1, 0)))))

##STEP 2: STANDARDIZE
train_input_norm = train %>%
  select(age_norm, hpw_norm, race_black, race_white, race_API,
      race_AIE, race_other,sex_M, single, job_employed, higher_edu,
      salary_greater, in_unit)
```

```
## STEP 3: VALIDATION
validation = validation %>%
 mutate(age_norm = (age - a1)/a2, hpw_norm = (hpw - b1)/b2)

validation = validation %>%
 mutate(race_black = if_else(race == "Black", 1, 0),
     race_white = if_else(race == "White", 1, 0),
     race_API = if_else(race == "Asian-Pac-Islander", 1, 0),
     race_AIE = if_else(race == "Amer-Indian-Eskimo", 1, 0),
     race_other = if_else(race == "Other", 1, 0),
     sex_M = if_else(sex == "Male",1,0))

salary_new = salary_new %>%
 mutate(single = if_else(marital.status == "Divorced", 1,
              if_else(marital.status == "Never-married", 1,if_else(marital.status
                                             == "Separated", 1,0))))
salary_new = salary_new %>%
 mutate(job_employed = if_else (workclass == "Private", 1,
                  if_else(workclass == "Self-emp-not-inc", 1,
                      if_else(workclass == "Federal-gov", 1,
                          if_else(workclass == "Local-gov", 1,
                              if_else (workclass == "State-gov", 1, 0))))))

salary_new = salary_new %>%
 mutate(higher_edu = if_else(education == "Bachelors", 1,
                 if_else(education == "Some-college", 1,
                     if_else(education == "Masters", 1,
                         if_else(education == "Doctorate", 1,
                             if_else(education == "Prof-school", 1,
                                 if_else(education == "Assoc-acdm", 1,
                                     if_else(education == "Assoc-voc", 1,0))))))))

salary_new = salary_new %>%
 mutate(salary_greater = if_else(salary_actual == "<=50K", 1, 0))

salary_new = salary_new %>%
 mutate(in_unit = if_else(relationship == "Wife", 1,
               if_else(relationship == "Husband", 1,
                   if_else(relationship == "Own-child", 1,
                       if_else(relationship == "Other-relative", 1, 0)))))

## STEP 4: STANDARDIZE
validation_input_norm = validation %>%
  select(age_norm, hpw_norm,race_black, race_white, race_API,
         race_AIE, race_other,sex_M, single, job_employed,
         higher_edu, in_unit, salary_greater)

## STEP 5: TRACK OUTPUT
train_output = train$salary_greater

##STEP 6: PREDICT FOR KNN
pred = knn(train_input_norm, validation_input_norm, train_output, 2)
validation = validation %>%
  mutate(salary_prediction = pred)
validation %>%
  select(salary_greater, salary_prediction)

##EVALUATE
# confusion matrix
validation %>%
  group_by(salary_prediction, salary_greater) %>%
  summarise(count = n()) %>%
  ungroup() %>%
  pivot_wider(names_from = salary_greater, values_from = count, values_fill = 0)

# accuracy
validation %>%
  mutate(flag = if_else(salary_greater == salary_prediction, 1, 0)) %>%
  summarise(accuracy = sum(flag)/length(flag), err_miscl_rate = 1 - accuracy)

# Sensitivity
validation %>%
  filter(salary_greater == "1") %>%
  mutate(flag = if_else(salary_greater == salary_prediction, 1, 0)) %>%
  summarise(sensitivity = sum(flag)/length(flag))

# Specificity
validation %>%
  filter(salary_greater == "1") %>%
  mutate(flag = if_else(salary_greater == salary_prediction, 1, 0)) %>%
  summarise(specificity = sum(flag)/length(flag))

##CONFUSION MATRIX
library("caret")

validation = validation %>%
  mutate(salary_greater = factor(salary_greater),
         salary_prediction = factor(salary_prediction))
m = confusionMatrix(validation$salary_prediction, validation$salary_greater)
m$overall[1]

##OPTIMAL K
options(warn = -1)
for (k in 1:14)
a = knn(train_input_norm, validation_input_norm, train_output, k)
cm = confusionMatrix(a,validation$salary_greater)
print(paste("The accuracy for k = ", k, "is", cm$overall[1]))

###CROSS VALIDATION
library("tidyverse")
library("forecast")
library("caret")
salary_new1 = read_csv("salary (1).csv")

##MISSING VALUES
data_omit = na.omit(salary_new1)
salary_new = data_omit

salary_new = salary_new %>%
  select(age, workclass, education, marital.status, relationship, race, sex, hpw,
native.country.new, salary)

salary_new = salary_new %>%
  rename(salary_actual = salary) %>%
  mutate(salary_actual = factor(salary_actual, c("<=50K", ">50K")))

salary_new = salary_new %>%
  mutate(salary_greater = if_else(salary_actual == "<=50K", 1, 0))

salary_new = salary_new %>%
  mutate(education = factor(education,c("Bachelors", "Some-college", "11th", "HS-grad",
                                        "Prof-school", "Assoc-acdm", "Assoc-voc", "9th",
                                        "7th-8th", "12th", "Masters", "1st-4th", "10th",
                                        "Doctorate", "5th-6th", "Preschool")))
salary_new = salary_new %>%
  mutate(higher_edu = if_else(education == "Bachelors", 1,
                              if_else(education == "Some-college", 1,
                                      if_else(education == "Masters", 1,
                                              if_else(education == "Doctorate", 1,
                                                      if_else(education == "Prof-school", 1,
                                                              if_else(education == "Assoc-acdm", 1,
                                   if_else(education == "Assoc-voc", 1,0))))))))

salary_new = salary_new %>%
 mutate(workclass = factor(workclass,c("Private", "Self-emp-not-inc", "Self-emp-inc",
                    "Federal-gov", "Local-gov", "State-gov",
                    "Without-pay", "Never-worked")))

salary_new = salary_new %>%
 mutate(job_employed = if_else (workclass == "Private", 1,
                 if_else(workclass == "Self-emp-not-inc", 1,
                     if_else(workclass == "Federal-gov", 1,
                         if_else(workclass == "Local-gov", 1,
                             if_else (workclass == "State-gov", 1, 0))))))

salary_new = salary_new %>%
 mutate(marital.status=factor(marital.status,c("Married-civ-spouse", "Divorced",
                        "Never-married", "Separated", "Widowed",
                        "Married-spouse-absent", "Married-AF-spouse")))

salary_new = salary_new %>%
 mutate(marital.status_single = if_else(marital.status == "Divorced", 1,
                    if_else(marital.status == "Never-married", 1,
                        if_else(marital.status == "Separated", 1,0))))

salary_new = salary_new %>%
 mutate(relationship = factor(relationship, c("Wife", "Own-child", "Husband",
                        "Not-in-family", "Other-relative",
                        "Unmarried")))
salary_new = salary_new %>%
 mutate(in_unit = if_else(relationship == "Wife", 1,
              if_else(relationship == "Husband", 1,
                  if_else(relationship == "Own-child", 1,
                      if_else(relationship == "Other-relative", 1, 0)))))

salary_new = salary_new %>%
 mutate(race = factor(race, c("White", "Asian-Pac-Islander", "Amer-Indian-Eskimo",
               "Other", "Black")))
salary_new = salary_new %>%
 mutate(race_black = if_else(race == "Black", 1, 0),
     race_white = if_else(race == "White", 1, 0),
     race_API = if_else(race == "Asian-Pac-Islander", 1, 0),
     race_AIE = if_else(race == "Amer-Indian-Eskimo", 1, 0),
     race_other = if_else(race == "Other", 1, 0))
```

```
salary_new = salary_new %>%
  mutate(sex = factor(sex, c("Female", "Male")))

salary_new = salary_new %>%
  mutate(sex_M = if_else(sex == "Male", 1,0))

salary_new = salary_new %>%
  mutate(Country = case_when(native.country.new %in% c("United-States") ~ "US",
                             native.country.new %in% c("Cambodia", "England",
                                                       "Puerto-Rico", "Canada",
                                                       "Germany", "Outlying-US(Guam-USVI-etc)",
                                                       "India", "Japan", "Greece", "South", "China",
                                                       "Cuba", "Iran", "Honduras", "Philippines",
                                                       "Italy", "Poland", "Jamaica", "Vietnam",
                                                       "Mexico", "Portugal","Ireland",
                                                       "France", "Dominican-Republic",
                                                       "Laos", "Ecuador", "Taiwan",
                                                       "Haiti", "Columbia", "Hungary",
                                                       "Guatemala", "Nicaragua", "Scotland",
                                                       "Thailand", "Yugoslavia", "El-Salvador",
                                                       "Trinadad&Tobago", "Peru", "Hong",
                                                       "Holand-Netherlands", "NA")~ "Not-US"))%>%
  mutate(native.country.new = factor(native.country.new, c("United-States", "Cambodia",
"England",
                                                           "Puerto-Rico", "Canada", "Germany",
                                                           "Outlying-US(Guam-USVI-etc)", "India",
                                                           "Japan", "Greece", "South", "China",
                                                           "Cuba", "Iran", "Honduras", "Philippines",
                                                           "Italy", "Poland", "Jamaica", "Vietnam",
                                                           "Mexico", "Portugal", "Ireland", "France",
                                                           "Dominican-Republic", "Laos", "Ecuador",
                                                           "Taiwan", "Haiti", "Columbia", "Hungary", "Guatemala",
                                                           "Nicaragua", "Scotland", "Thailand", "Yugoslavia",
                                                           "El-Salvador", "Trinadad&Tobago", "Peru", "Hong",
                                                           "Holand-Netherlands","NA")))

salary_new = salary_new %>%
  mutate(US = if_else(Country == "US", 1, 0))

salary_new = na.omit(salary_new)

salary_new = salary_new %>%
  mutate(id = 1:nrow(salary_new))
```

```
###N ROW
set.seed(30)
train = salary_new %>%
  sample_frac(0.7)

validation = salary_new %>%
  slice(setdiff(salary_new$id, train$id))

train.lr = glm(salary_greater ~ age  + education + marital.status + workclass +
              relationship + race + sex + hpw + native.country.new, train, family = "binomial")

summary(train.lr)

validation = validation %>%
  mutate(prob_prediction = predict(train.lr, validation, type = "response"))


validation = validation %>%
  mutate(salary_prediction = if_else(prob_prediction >= 0.5, 1, 0))

validation = validation %>%
  mutate(salary_greater = factor(salary_greater, c(0,1)),
         salary_prediction = factor(salary_prediction, c(0,1)))

confusionMatrix(validation$salary_prediction, validation$salary_greater)
```

LOGISTIC REGRESSION

```
rm(list = ls())
# load tidyverse package in the current session
library("tidyverse")
library("FNN")
library("forecast")
library("caret")
### Start Here
Salary = read_csv("salary.csv")
###Set Working Directory
##Formatting Data
Salary = Salary %>%
  mutate (id = 1:nrow(Salary))

Salary = Salary %>%
  rename(salary_actual = salary)

Salary = Salary %>%
  mutate(salary_over_50 = if_else(salary_actual == "<=50K", 1, if_else(salary_actual ==
"<50K",1,0)))%>%
  mutate(Male = if_else(sex == "Male", 1,0), Female = if_else(sex == "Female",1,0))

Salary1 = Salary %>%
  select(id, age, Male, education, marital.status, relationship, race, hpw,
native.country.new, salary_over_50)

## Separate Validation and Train Data
set.seed(30)
train = Salary1 %>%
  sample_frac(0.7)
validation = Salary1 %>%
  slice(setdiff(Salary1$id, train$id))

## Running Logistic Regression
train.lr = glm(salary_over_50 ~ age + Male + education + marital.status +
                 relationship + race + hpw + native.country.new, train, family = "binomial")

summary(train.lr)

# predict output in the validation data
validation = validation %>%
  mutate(prob_prediction = predict(train.lr, validation, type = "response"))
validation = validation %>%
  mutate(Salary_prediction = if_else(prob_prediction >= 0.5, 1, 0))
# function to generate error/accuracy measures
# confusionMatrix(prediction variable, actual variable)
# converting the variables to factor class

validation = validation %>%
  mutate(Salary_actual = factor(salary_over_50, c(0,1)),
         Salary_prediction = factor(Salary_prediction, c(0,1)))

## confusion matrix, accuracy, sensitivity & specificity
confusionMatrix(validation$Salary_prediction,validation$Salary_actual)
#To remove all data/values
rm(list = ls())
```

CLASSIFICATION TREE

```
rm(list = ls())
library(tidyverse)
library(rpart)
library(rpart.plot)
library(caret)

salary = read_csv("salary.csv")
salary_new = salary %>%
  rename(native_country = native.country.new) %>%
  select(age, workclass, education, marital.status, relationship, race, sex, hpw, native_country, 
salary)
salary_new = salary_new %>%
  rename(salary_actual = salary) %>%
  mutate(salary_actual = factor(salary_actual, c("<=50K", ">50K")))

##Grouping variables
salary_new = salary_new %>%
  mutate(educationlvl = case_when(education %in% c("11th","9th", "7th-8th", "12th", "1st-4th", 
"10th", "5th-6th", "Preschool") ~ "preHS",
                              education %in% c("Bachelors") ~ "Undergrad",
                              education %in% c("Masters", "Prof-School", "Doctorate") ~ "Postgrad",
                              education %in% c("Assoc-acdm", "Assoc-voc") ~ "Associates",
                              education %in% c("HS-grad","Some-college") ~ "HSgraduate")) %>%
  mutate(educationlvl = factor(educationlvl, levels = c("HSgraduate","Postgrad", "Associates", 
"Undergrad", "preHS")))
salary_new = salary_new %>%
  mutate(work = case_when(workclass %in% c( "Without-pay", "Never-worked") ~ 
"notworking",
                                  workclass %in% c("Self-emp-not-inc", "Self-emp-inc") ~ "selfemployed",
                                  workclass %in% c("Federal-gov", "Local-gov", "State-gov") ~ "govwork",
                                  workclass %in% c("Private") ~ "Private")) %>%
  mutate(workclass = factor(work,c("Private", "selfemployed", "govwork", "notworking")))

salary_new = salary_new %>%
  mutate(marital_status = case_when(marital.status %in% c("Divorced", "Never-married", 
"Separated", "Widowed", "Married-spouse-absent") ~ "notmarried",
                          marital.status %in% c("Married-civ-spouse", "Married-AF-spouse") ~ "married")) 
%>%
  mutate(marital.status = factor(marital_status,c("notmarried", "married")))
salary_new = salary_new %>%
  mutate(relationship = factor(relationship, c("Wife", "Own-child", "Husband", "Not-in-family", 
"Other-relative", "Unmarried")))
salary_new = salary_new %>%
  mutate(race = factor(race, c("White", "Asian-Pac-Islander", "Amer-Indian-Eskimo", "Other",
"Black")))
salary_new = salary_new %>%
  mutate(sex = factor(sex, c("Female", "Male")))
salary_new = salary_new %>%
  mutate(native_country = factor(native_country,c("United-States", "other")))
salary_new = salary_new %>%
  mutate(id = 1:nrow(salary_new))
set.seed(30)
train = salary_new %>%
  sample_frac(0.7)
validation = salary_new %>%
  slice(setdiff(salary_new$id,train$id))
salary_new.ct = rpart(salary_actual~ age + work + educationlvl + marital_status + relationship +
race +
                  sex + hpw + native_country,
                data = train,
                method = "class", cp = 0.00001, minsplit = 10, xval = 10)
plotcp(salary_new.ct)
cp.table = as_tibble(salary_new.ct$cptable)
optimal.cp = cp.table %>%
  filter(nsplit == 10)
pruned.ct = prune(salary_new.ct, cp = optimal.cp$CP)
prp(pruned.ct, type = 1, extra = 2, under = TRUE, split.font = 2, varlen = 0)
results = predict(pruned.ct, validation, type = "class")
validation = validation %>%
  mutate(salary_prediction = results)
confusionMatrix(validation$salary_prediction,validation$salary_actual)
```