project.r

ngage

2022-11-18

```
##########################################
#                                        #
# Fall 2022 Data Mining -- Group Project #
#                                        #
# Team:                                  #
#   Tania Cuff                           #
#   Nathan Gage                          #
#   Timothy Lee                          #
#   Elizabeth McPherson                  #
#                                        #
##########################################

# Exploratory Data Analysis

columnnames <- c(
  "age",
  "workclass",
  "fnlwgt",
  "education",
  "education-num",
  "marital-status",
  "occupation",
  "relationship",
  "race",
  "sex",
  "capital-gain",
  "capital-loss",
  "hours-per-week",
  "native-country",
  "income"
)

# reading the data
train <-
  read.csv("adult.data",
           header = FALSE,
           na.strings = c("?", " ?", "NA"),)
test <-
  read.csv(
    "adult.test",
    header = FALSE,
    skip = 1,
```

```
    na.strings = c("?", " ?", "NA"),
  )

# naming columns
names(train) <- columnnames
names(test) <- columnnames

# converting char columns to factors
train <- as.data.frame(unclass(train), stringsAsFactors = T)
test <- as.data.frame(unclass(test), stringsAsFactors = T)

# needed because test is missing one country to create factors from
allcountries <-
  unique(union(train$native.country, test$native.country))
test$native.country <-
  factor(test$native.country, levels = allcountries)

str(train)
```

```
## 'data.frame':    32561 obs. of  15 variables:
##  $ age           : int  39 50 38 53 28 37 49 52 31 42 ...
##  $ workclass     : Factor w/ 8 levels " Federal-gov",..: 7 6 4 4 4 4 4 6 4 4 ...
##  $ fnlwgt        : int  77516 83311 215646 234721 338409 284582 160187 209642 45781 159449 ...
##  $ education     : Factor w/ 16 levels " 10th"," 11th",..: 10 10 12 2 10 13 7 12 13 10 ...
##  $ education.num : int  13 13 9 7 13 14 5 9 14 13 ...
##  $ marital.status: Factor w/ 7 levels " Divorced"," Married-AF-spouse",..: 5 3 1 3 3 3 4 3 5 3 ...
##  $ occupation    : Factor w/ 14 levels " Adm-clerical",..: 1 4 6 6 10 4 8 4 10 4 ...
##  $ relationship  : Factor w/ 6 levels " Husband"," Not-in-family",..: 2 1 2 1 6 6 2 1 2 1 ...
##  $ race          : Factor w/ 5 levels " Amer-Indian-Eskimo",..: 5 5 5 3 3 5 3 5 5 5 ...
##  $ sex           : Factor w/ 2 levels " Female"," Male": 2 2 2 2 1 1 1 2 1 2 ...
##  $ capital.gain  : int  2174 0 0 0 0 0 0 0 14084 5178 ...
##  $ capital.loss  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ hours.per.week: int  40 13 40 40 40 40 16 45 50 40 ...
##  $ native.country: Factor w/ 41 levels " Cambodia"," Canada",..: 39 39 39 39 5 39 23 39 39 39 ...
##  $ income        : Factor w/ 2 levels " <=50K"," >50K": 1 1 1 1 1 1 1 2 2 2 ...
```

```
str(test)
```

```
## 'data.frame':    16281 obs. of  15 variables:
##  $ age           : int  25 38 28 44 18 34 29 63 24 55 ...
##  $ workclass     : Factor w/ 8 levels " Federal-gov",..: 4 4 2 4 NA 4 NA 6 4 4 ...
##  $ fnlwgt        : int  226802 89814 336951 160323 103497 198693 227026 104626 369667 104996 ...
##  $ education     : Factor w/ 16 levels " 10th"," 11th",..: 2 12 8 16 16 1 12 15 16 6 ...
##  $ education.num : int  7 9 12 10 10 6 9 15 10 4 ...
##  $ marital.status: Factor w/ 7 levels " Divorced"," Married-AF-spouse",..: 5 3 3 3 5 5 5 3 5 3 ...
##  $ occupation    : Factor w/ 14 levels " Adm-clerical",..: 7 5 11 7 NA 8 NA 10 8 3 ...
##  $ relationship  : Factor w/ 6 levels " Husband"," Not-in-family",..: 4 1 1 1 4 2 5 1 5 1 ...
##  $ race          : Factor w/ 5 levels " Amer-Indian-Eskimo",..: 3 5 5 3 5 5 3 5 5 5 ...
##  $ sex           : Factor w/ 2 levels " Female"," Male": 2 2 2 2 1 2 2 2 1 2 ...
##  $ capital.gain  : int  0 0 0 7688 0 0 0 3103 0 0 ...
##  $ capital.loss  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ hours.per.week: int  40 50 40 40 30 30 40 32 40 10 ...
##  $ native.country: Factor w/ 41 levels " United-States",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ income        : Factor w/ 2 levels " <=50K."," >50K.": 1 1 2 2 1 1 1 2 1 1 ...
```

```
# VERIFY DATA QUALITY

# missing values / invalid:
colSums(is.na(train))
```

```
##            age      workclass         fnlwgt      education  education.num
##              0           1836              0              0              0
## marital.status     occupation   relationship           race            sex
##              0           1843              0              0              0
##   capital.gain   capital.loss hours.per.week native.country         income
##              0              0              0            583              0
```

```
train <- na.omit(train)
colSums(is.na(train))
```

```
##            age      workclass         fnlwgt      education  education.num
##              0              0              0              0              0
## marital.status     occupation   relationship           race            sex
##              0              0              0              0              0
##   capital.gain   capital.loss hours.per.week native.country         income
##              0              0              0              0              0
```

```
colSums(is.na(test))
```

```
##            age      workclass         fnlwgt      education  education.num
##              0            963              0              0              0
## marital.status     occupation   relationship           race            sex
##              0            966              0              0              0
##   capital.gain   capital.loss hours.per.week native.country         income
##              0              0              0            274              0
```

```
test <- na.omit(test)
colSums(is.na(test))
```

```
##            age      workclass         fnlwgt      education  education.num
##              0              0              0              0              0
## marital.status     occupation   relationship           race            sex
##              0              0              0              0              0
##   capital.gain   capital.loss hours.per.week native.country         income
##              0              0              0              0              0
```

```
# checking for outliers:

# no outliers in age
summary(train$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   17.00   28.00   37.00   38.44   47.00   90.00
```

```
summary(test$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   17.00   28.00   37.00   38.77   48.00   90.00
```

```
# seems like train & test have some rows where capital.gain == 99999
# this is a suspiciously high & specific value so we will omit it
summary(train$capital.gain)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
```

```
##       0       0       0    1092       0   99999
```

```
summary(test$capital.gain)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0       0       0    1120       0   99999
```

```
nrow(train[train$capital.gain == 99999, ])
```

```
## [1] 148
```

```
nrow(test[test$capital.gain == 99999, ])
```

```
## [1] 81
```

```
train <- train[!(train$capital.gain == 99999), ]
test <- test[!(test$capital.gain == 99999), ]

# the values are now far more reasonable
summary(train$capital.gain)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0     0.0     0.0   604.3     0.0 41310.0
```

```
summary(test$capital.gain)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0     0.0     0.0   585.6     0.0 41310.0
```

```
# no outliers in cap loss
summary(train$capital.loss)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    0.00    0.00   88.81    0.00 4356.00
```

```
summary(test$capital.loss)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    0.00    0.00   89.52    0.00 3770.00
```

```
# all values are > 0 and < 7 * 24 = 168
summary(train$hours.per.week)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   40.00   40.00   40.89   45.00   99.00
```

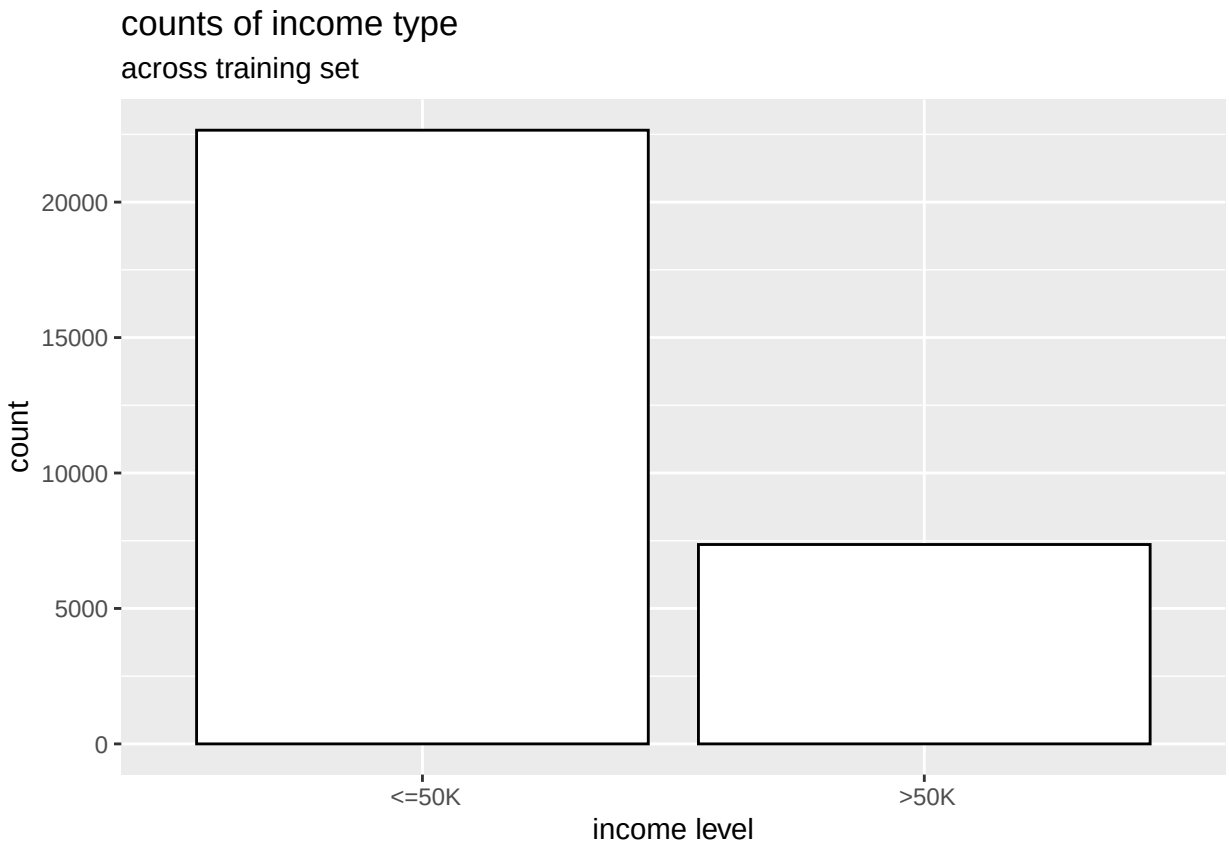
```
summary(test$hours.per.week)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00   40.00   40.00   40.89   45.00   99.00
```

```
# drop education.num (same as our education factor)
train <- train[, !names(train) %in% c("education.num")]
test <- test[, !names(test) %in% c("education.num")]

# GIVE SIMPLE AND APPROPRIATE STATISTICS

# we already performed some of this in the data clean up,
# however, we will take a look at the finalized data now:
summary(train)
```

```
##       age                    workclass         fnlwgt
##  Min.   :17.0    Private         :22208   Min.   :  13769
##  1st Qu.:28.0    Self-emp-not-inc: 2470   1st Qu.: 117606
##  Median :37.0    Local-gov       : 2062   Median : 178440
##  Mean   :38.4    State-gov       : 1278   Mean   : 189776
##  3rd Qu.:47.0    Self-emp-inc    : 1040   3rd Qu.: 237642
##  Max.   :90.0    Federal-gov     :  942   Max.   :1484705
##                 (Other)          :   14
##          education                   marital.status            occupation
##   HS-grad     :9818    Divorced             : 4203    Craft-repair   :4022
##   Some-college:6667    Married-AF-spouse    :   21    Prof-specialty :3973
##   Bachelors   :5007    Married-civ-spouse   :13943    Exec-managerial:3954
##   Masters     :1610    Married-spouse-absent:  369    Adm-clerical   :3715
##   Assoc-voc   :1306    Never-married        : 9715    Sales          :3560
##   11th        :1048    Separated            :  937    Other-service  :3210
##  (Other)      :4558    Widowed              :  826   (Other)         :7580
##           relationship                    race           sex
##   Husband       :12350    Amer-Indian-Eskimo:  286    Female: 9762
##   Not-in-family : 7706    Asian-Pac-Islander:  888    Male  :20252
##   Other-relative:  889    Black             : 2811
##   Own-child     : 4464    Other             :  229
##   Unmarried     : 3208    White             :25800
##   Wife          : 1397
##
##   capital.gain      capital.loss     hours.per.week         native.country
##  Min.   :    0.0   Min.   :   0.00   Min.   : 1.00    United-States:27365
##  1st Qu.:    0.0   1st Qu.:   0.00   1st Qu.:40.00    Mexico       :  609
##  Median :    0.0   Median :   0.00   Median :40.00    Philippines  :  187
##  Mean   :  604.3   Mean   :  88.81   Mean   :40.89    Germany      :  128
##  3rd Qu.:    0.0   3rd Qu.:   0.00   3rd Qu.:45.00    Puerto-Rico  :  109
##  Max.   :41310.0   Max.   :4356.00   Max.   :99.00    Canada       :  106
##                                                      (Other)      : 1510
##     income
##   <=50K:22654
##   >50K : 7360
##
##
##
##
##
```

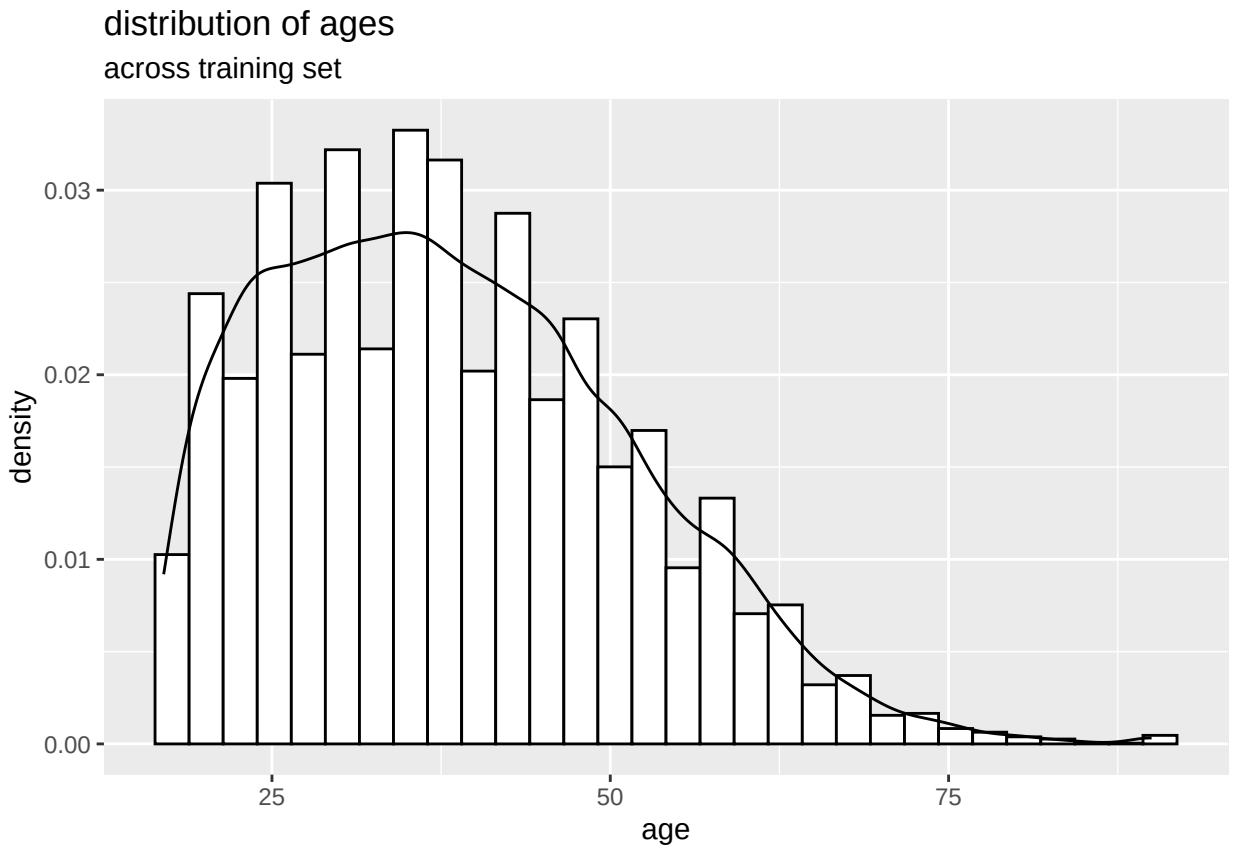
```
# VISUALIZE SOME OF THE FEATURES
library(ggplot2)

ggplot(train, aes(x = factor(income))) +
  geom_bar(color = "black", fill = "white") +
  labs(title = "counts of income type",
       subtitle = "across training set",
       x = "income level",
  )
```



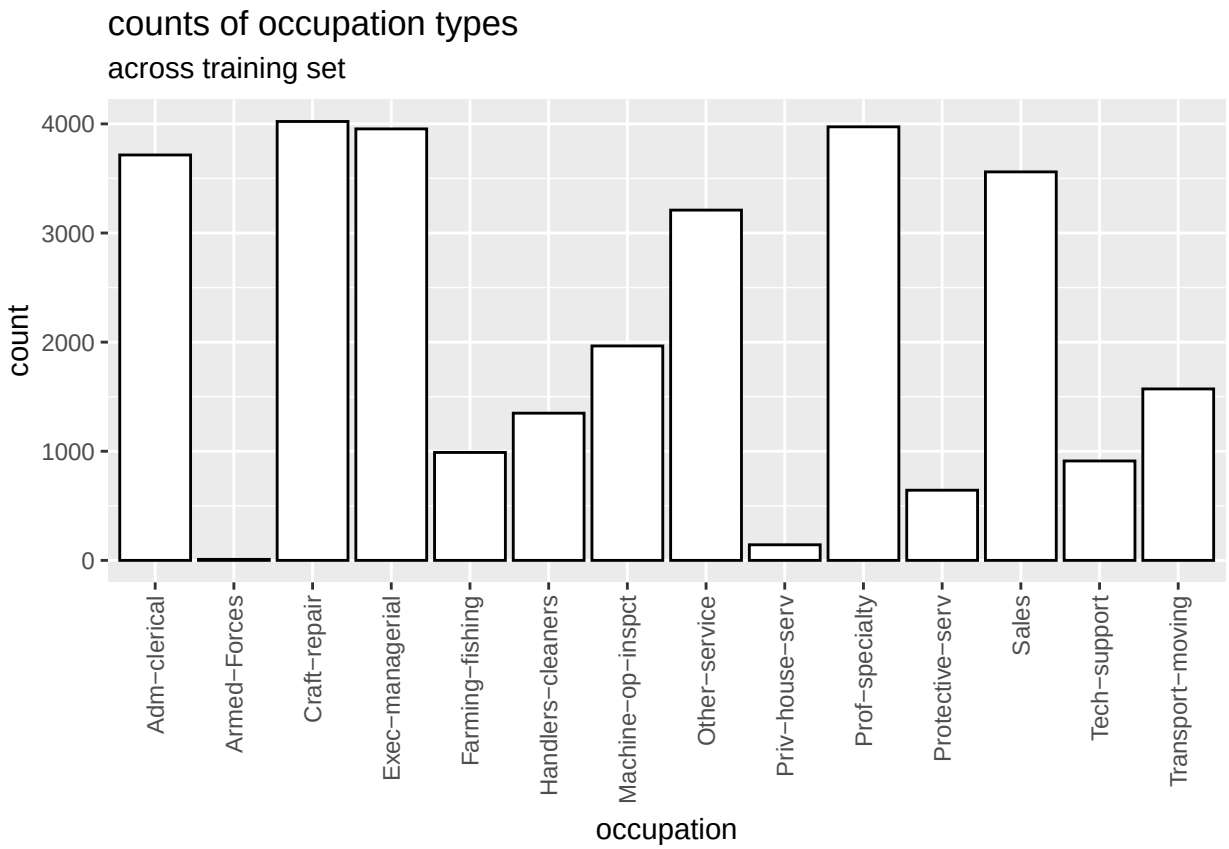

```
# distribution of age
ggplot(train, aes(x = age)) +
  geom_histogram(aes(y = ..density..), color = "black", fill = "white") +
  geom_density(alpha = .2) +
  labs(title = "distribution of ages",
       subtitle = "across training set",
       x = "age",
  )
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(density)` instead.
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

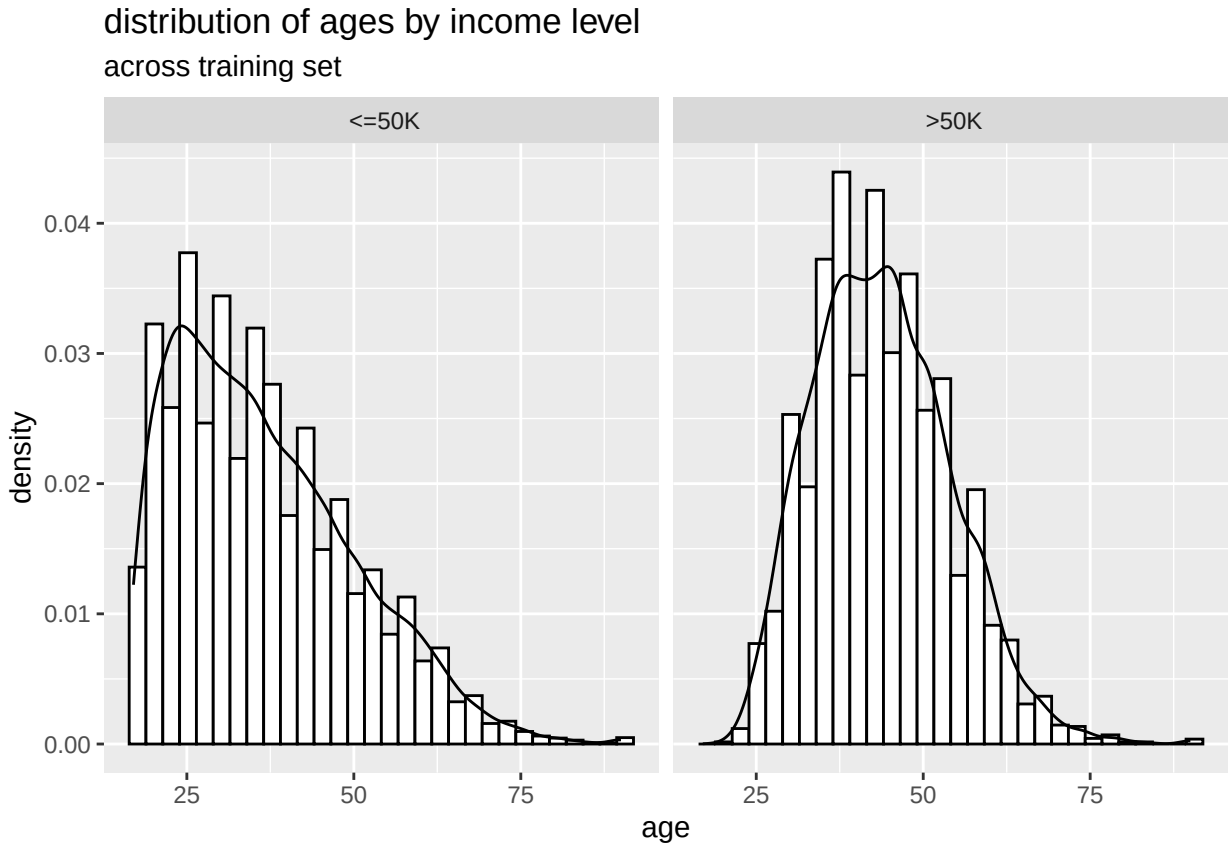



```
# let's take a look at the different types of jobs:
ggplot(train, aes(x = factor(occupation))) +
  geom_bar(color = "black", fill = "white") +
  labs(title = "counts of occupation types",
       subtitle = "across training set",
       x = "occupation") +
  theme(axis.text.x = element_text(
    angle = 90,
    vjust = 0.5,
    hjust = 1
  ))
```



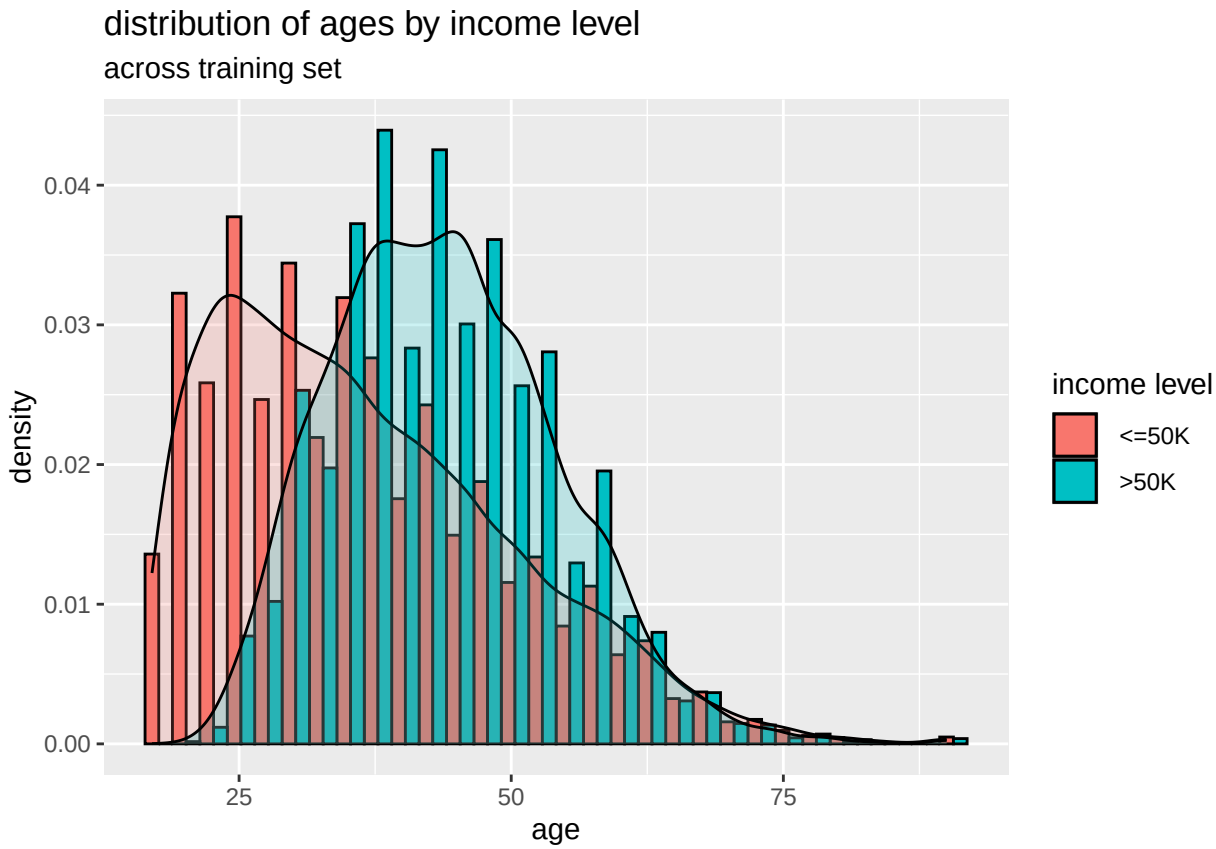

```
# EXPLORE RELATIONSHIPS

# first, we're curious how age affects the income status
ggplot(train, aes(x = age)) +
  geom_histogram(aes(y = ..density..), color = "black", fill = "white") +
  geom_density(alpha = .2) +
  facet_grid(~ income) +
  labs(title = "distribution of ages by income level",
       subtitle = "across training set",
       x = "age",
  )
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



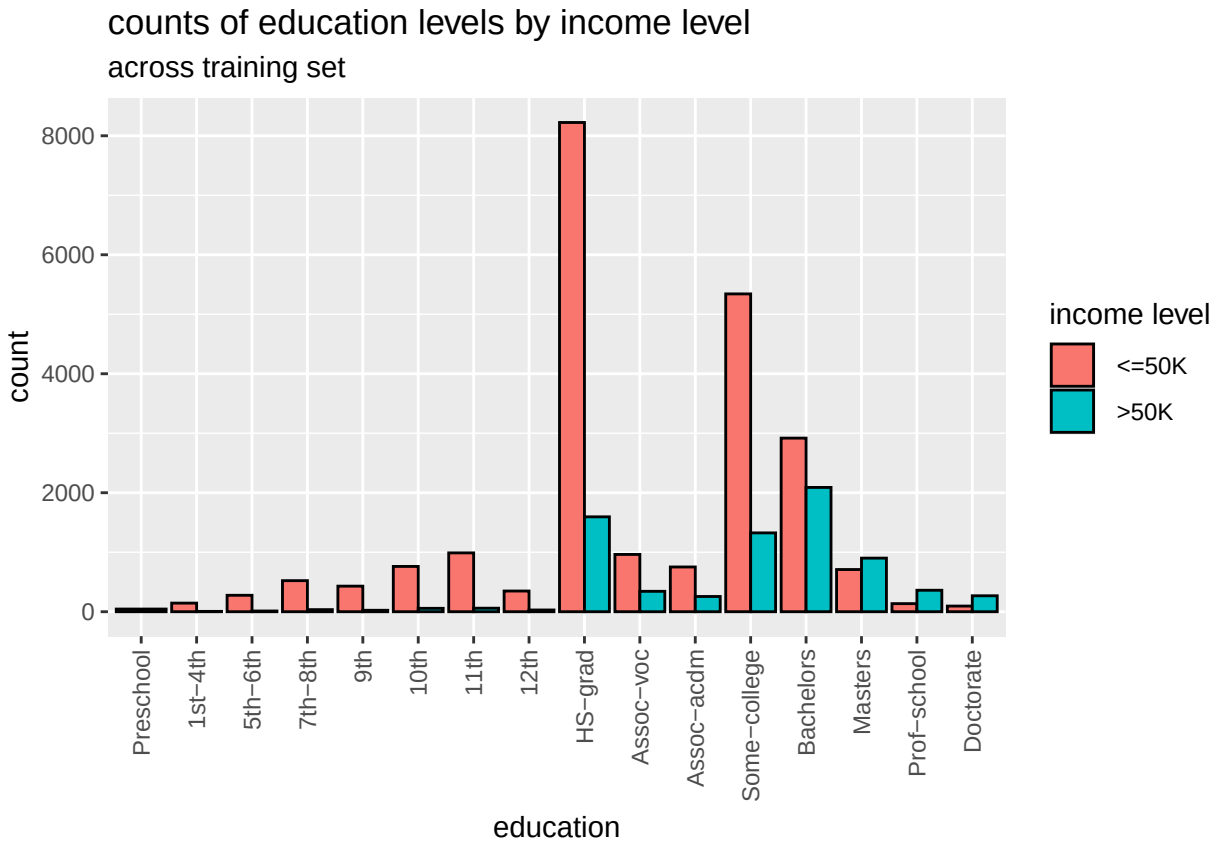

```
ggplot(train, aes(
  x = age,
  group = factor(income),
  fill = factor(income)
)) +
  geom_histogram(aes(y = ..density..), position = "dodge", color = "black") +
  geom_density(alpha = .2) +
  labs(
    title = "distribution of ages by income level",
    subtitle = "across training set",
    x = "age",
    fill = "income level"
  )
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

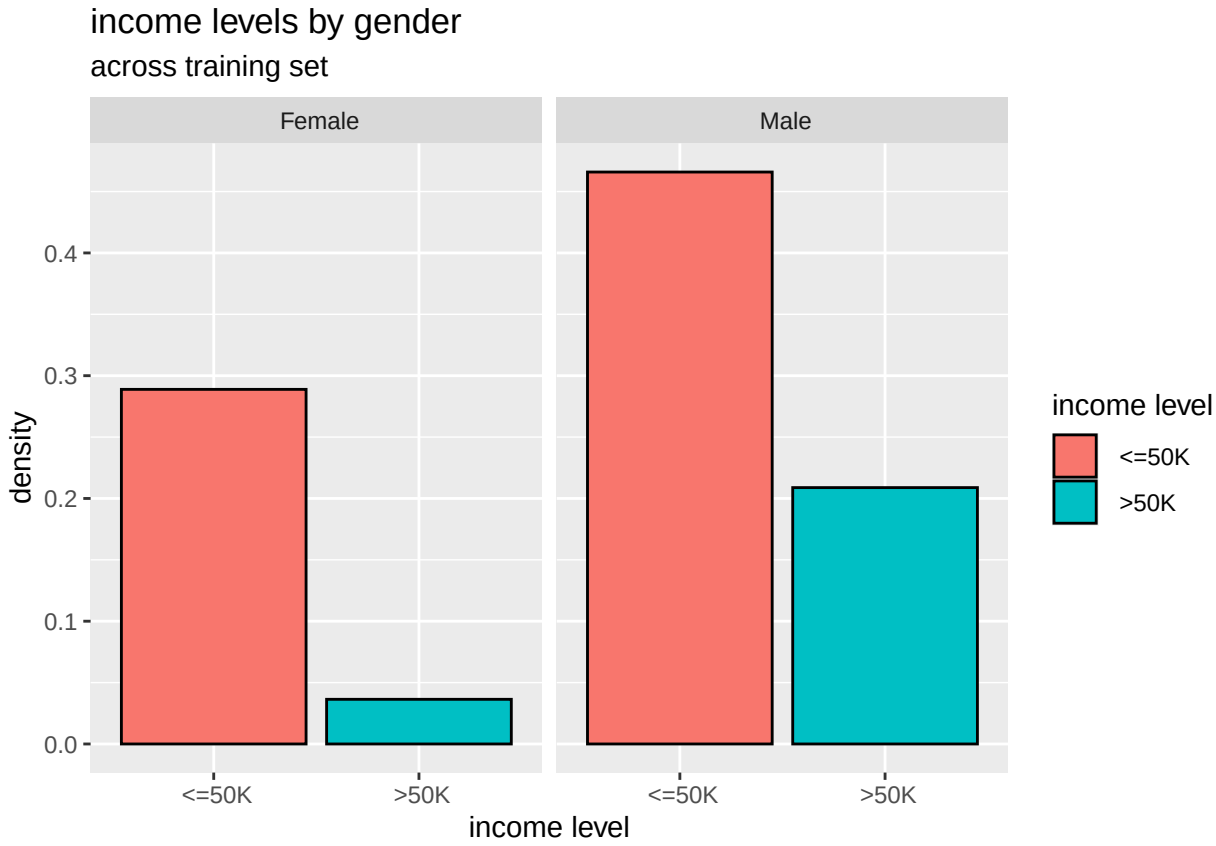



```
# for our own sake, does education level affect income?
ggplot(train, aes(
  x = factor(
    education,
    levels = c(
      " Preschool",
      " 1st-4th",
      " 5th-6th",
      " 7th-8th",
      " 9th",
      " 10th",
      " 11th",
      " 12th",
      " HS-grad",
      " Assoc-voc",
      " Assoc-acdm",
      " Some-college",
      " Bachelors",
      " Masters",
      " Prof-school",
      " Doctorate"
    )
  ),
  group = factor(income),
  fill = factor(income),
)) +
  geom_bar(position = "dodge",
```

```
          color = "black") +
  labs(
    title = "counts of education levels by income level",
    subtitle = "across training set",
    x = "education",
    fill = "income level"
  ) +
  theme(axis.text.x = element_text(
    angle = 90,
    vjust = 0.5,
    hjust = 1
  ))
```




```
# and finally we are interested in seeing the relationship between hours per week and income
ggplot(train, aes(x = factor(income), fill = factor(income))) +
  geom_bar(aes(y = (..count.. / sum(..count..))), color = "black") +
  facet_grid(~ sex) +
  labs(
    title = "income levels by gender",
    subtitle = "across training set",
    x = "income level",
    y = "density",
    fill = "income level"
  )
```




```
# CLASSIFICATION

# model 1: LOGISTIC REGRESSION
logmodel <- glm(income ~ ., data = train, family = "binomial")
logmodel.preds <- predict(logmodel, test, type = "response")
logmodel.confusion <-
  table(test$income, ifelse(logmodel.preds < 0.5, 0, 1))
logmodel.accuracy <-
  (logmodel.confusion[1] + logmodel.confusion[4]) / sum(logmodel.confusion)
logmodel.confusion
```

```
##
##                0     1
##    <=50K. 10530   830
##    >50K.   1465  2154
```

```
logmodel.accuracy
```

```
## [1] 0.8467855
```

```
# model 2: RANDOM FOREST
library(randomForest)
```

```
## randomForest 4.7-1.1
## Type rfNews() to see new features/changes/bug fixes.
##
## Attaching package: 'randomForest'
##
## The following object is masked from 'package:ggplot2':
```

```
##
##     margin
```

```
rfmodel <- randomForest(income ~ ., data = train)
rfmodel.preds <- predict(rfmodel, test)
rfmodel.confusion <- table(test$income, rfmodel.preds)
rfmodel.accuracy <-
  (rfmodel.confusion[1] + rfmodel.confusion[4]) / sum(rfmodel.confusion)
rfmodel.confusion
```

```
##          rfmodel.preds
##            <=50K  >50K
##    <=50K.   9657  1703
##    >50K.     862  2757
```

```
rfmodel.accuracy
```

```
## [1] 0.8287603
```

```
# model 3: SVM
library('e1071')
svmmodel <-
  svm(
    income ~ .,
    data = train,
    type = "C-classification",
    kernel = "linear",
    scale = T
  )
svmmodel.preds <- predict(svmmodel, test)
svmmodel.confusion <- table(test$income, svmmodel.preds)
svmmodel.accuracy <-
  (svmmodel.confusion[1] + svmmodel.confusion[4]) / sum(svmmodel.confusion)
svmmodel.confusion
```

```
##          svmmodel.preds
##            <=50K  >50K
##    <=50K.  10751   609
##    >50K.    1734  1885
```

```
svmmodel.accuracy
```

```
## [1] 0.843581
```

```
svmmodel2 <-
  svm(
    income ~ .,
    data = train,
    type = "C-classification",
    kernel = "polynomial",
    scale = T
  )
svmmodel2.preds <- predict(svmmodel2, test)
svmmodel2.confusion <- table(test$income, svmmodel2.preds)
svmmodel2.accuracy <-
  (svmmodel2.confusion[1] + svmmodel2.confusion[4]) / sum(svmmodel2.confusion)
svmmodel2.confusion
```

```
##          svmmodel2.preds
##            <=50K  >50K
##    <=50K.  11282    78
##    >50K.    3055   564
```

```
svmmodel2.accuracy
```

```
## [1] 0.7908405
```

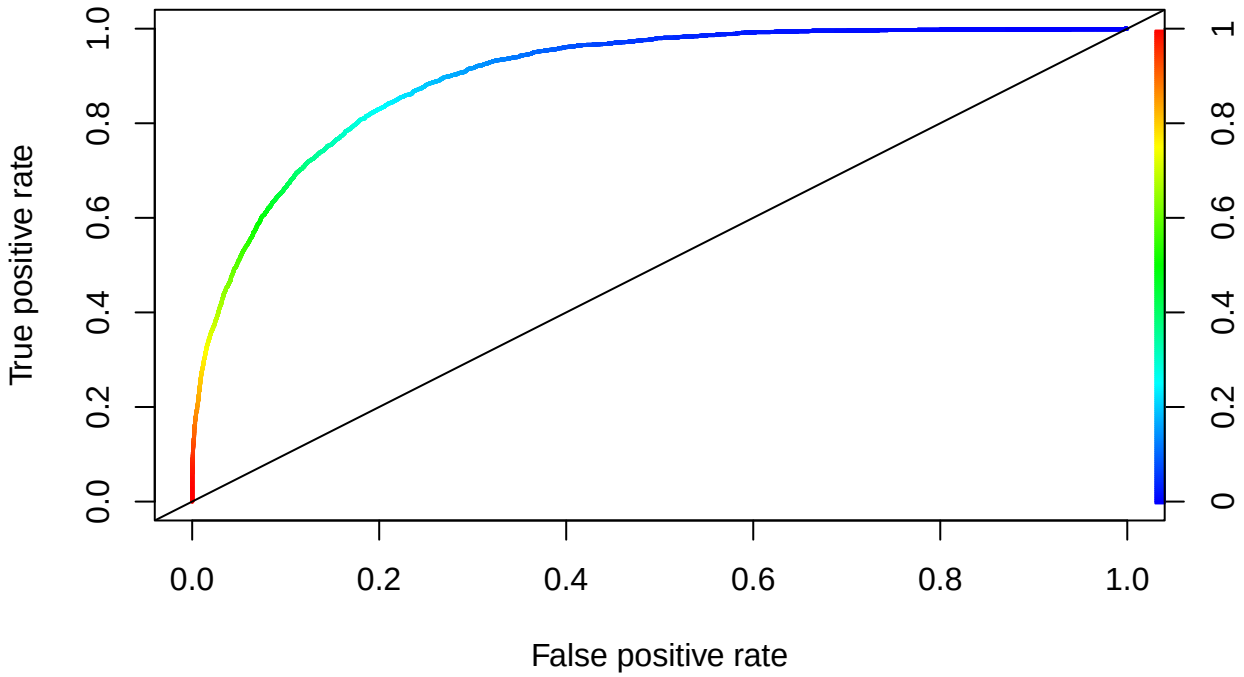
```
svmmodel3 <-
  svm(
    income ~ .,
    data = train,
    type = "C-classification",
    kernel = "sigmoid",
    scale = T
  )
svmmodel3.preds <- predict(svmmodel3, test)
svmmodel3.confusion <- table(test$income, svmmodel3.preds)
svmmodel3.accuracy <-
  (svmmodel3.confusion[1] + svmmodel3.confusion[4]) / sum(svmmodel3.confusion)
svmmodel3.confusion
```

```
##          svmmodel3.preds
##            <=50K  >50K
##    <=50K.  10718   642
##    >50K.    1725  1894
```

```
svmmodel3.accuracy
```

```
## [1] 0.8419788
```

```
# final model is logistic regression model with 84.67855% accuracy
# honorable mentions: SVM w/ linear kernel & SVM with sigmoid kernel

# creative work: ROC analysis
library(ROCR)
pred <- prediction(logmodel.preds, test$income)
roc <- performance(pred, "tpr", "fpr")
plot(roc, colorize = T, lwd = 2)
abline(a = 0, b = 1)
```




```
# this is a good ROC, with a solid balance between FPR and TPR.
# given that we have imbalanced classes, this means we are acheiving
# good performance despite class imbalance.
```