Project 2 - Classification

Siri Bafna

Census Income Data Analysis

Source: UCI Machine Learning Repository

Link: http://archive.ics.uci.edu/ml/datasets/Census+Income

Number of Observations: 48.8K ## Data Cleaning

Factoring character variables, cleaning NA's, reading in data

I began data cleaning by removing the column "fnlwgt" because it had no relationship to income.

Since the data had nulls as '?' instead of NA, I ran the gsub() function on the dataset, replacing '?'s with NA's. I then used the is.na() to remove any bad data from the data set. The columns with null data included workclass, occupation and native_country.

I then factored many of the variables as they were character based, not integer and ended my data cleaning by verifying my changes using str().

```
censusIncome <- read.csv("/Users/siri/Downloads/CensusIncome.csv", header=TRUE)
censusIncome <- censusIncome[,c(1,2,4,6,7,8, 9, 10, 11, 12, 13, 14, 15)]
censusIncome$workclass <- gsub("?", NA, censusIncome$workclass, fixed = TRUE);
censusIncome$native_country <- gsub("?", NA, censusIncome$native_country, fixed = TRUE
);


censusIncome$occupation <- gsub("?", NA, censusIncome$occupation, fixed = TRUE);
censusIncome <- censusIncome[!is.na(censusIncome$workclass),]
censusIncome <- censusIncome[!is.na(censusIncome$occupation),]
censusIncome <- censusIncome[!is.na(censusIncome$native_country),]


censusIncome$workclass <- as.factor(censusIncome$workclass)
censusIncome$education <- as.factor(censusIncome$education)
censusIncome$marital_status <- as.factor(censusIncome$marital_status)
censusIncome$occupation <- as.factor(censusIncome$occupation)
censusIncome$relationship <- as.factor(censusIncome$relationship)
censusIncome$race <- as.factor(censusIncome$race)
censusIncome$sex <- as.factor(censusIncome$sex)
```

```
censusIncome$native_country <- as.factor(censusIncome$native_country)

censusIncome$income_level <- as.factor(censusIncome$income_level)


str(censusIncome) # data exploration function # 1

'data.frame':   45222 obs. of  13 variables:

 $ age           : int  39 50 38 53 28 37 49 52 31 42 ...

 $ workclass     : Factor w/ 7 levels "Federal-gov",..: 6 5 3 3 3 3 3 5 3 3 ...

 $ education     : Factor w/ 16 levels "10th","11th",..: 10 10 12 2 10 13 7 12 13 10 .
..

 $ 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  : num  2174 0 0 0 0 ...

 $ capital_loss  : num  0 0 0 0 0 0 0 0 0 0 ...

 $ hours_per_week: num  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_level  : Factor w/ 2 levels "<=50K",">50K": 1 1 1 1 1 1 1 2 2 2 ...
```

Divide into train and test

```
set.seed(1234)

i <- sample(1:nrow(censusIncome), .6*nrow(censusIncome), replace=FALSE)

train <- censusIncome[i,]

test <- censusIncome[-i,]
```

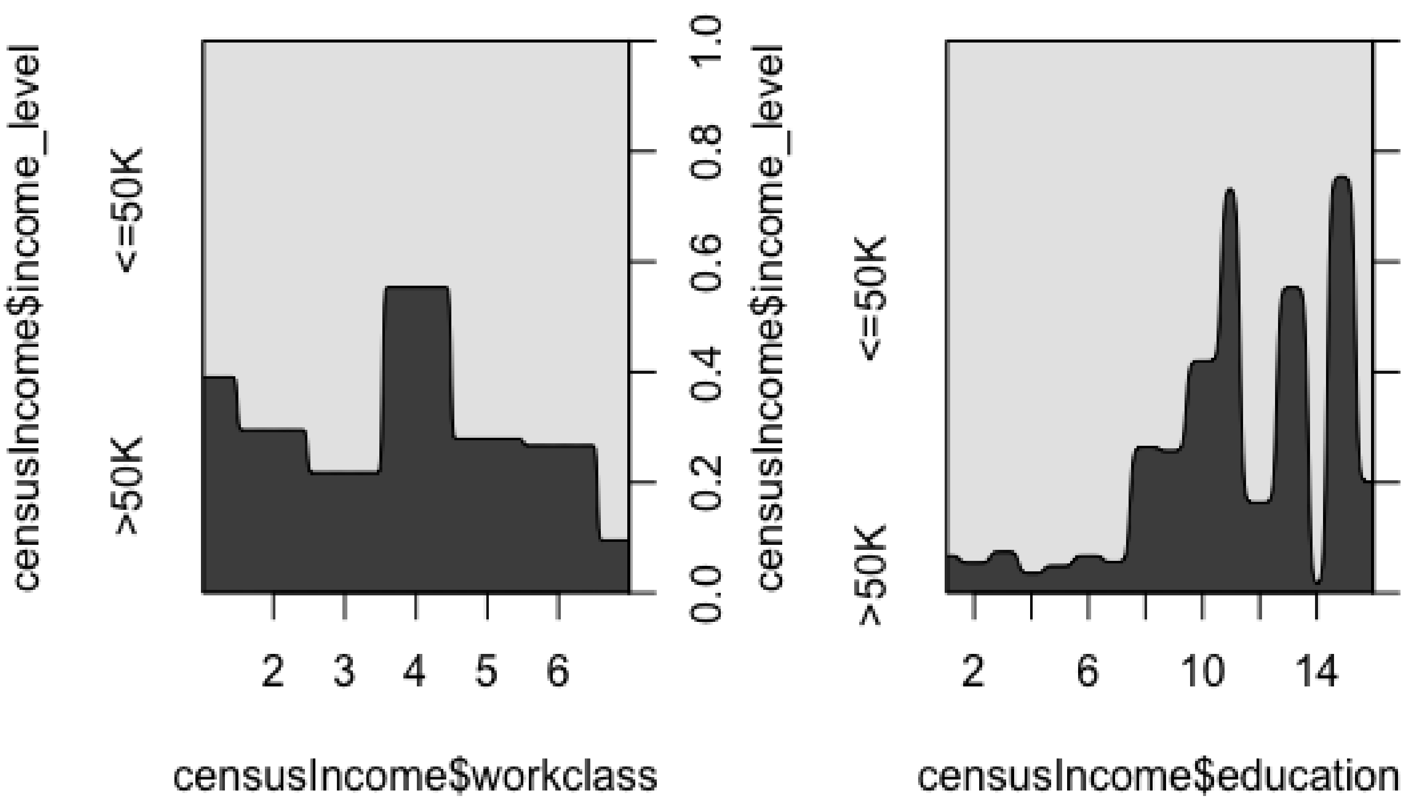
Data Exploration

```
# data exploration function # 2

per_no_capital_gain <- sum(censusIncome$capital_gain==0)/length(censusIncome$capital_g
ain)

print("Percentage of Instances Without Capital Gain")

[1] "Percentage of Instances Without Capital Gain"

print(per_no_capital_gain)

[1] 0.9161912
```

```
print("Division of Income based on Sex")
[1] "Division of Income based on Sex"
table(censusIncome$sex) # data exploration function # 3

Female   Male
 14695  30527
```

Plots For Data Exploration

```
par(mfrow=c(1, 2))
cdplot(censusIncome$income_level~censusIncome$workclass)
cdplot(censusIncome$income_level~censusIncome$education)
```



As shown below in all of my models, my feature consisted of all the columns except native_country. This is because native_country contains more than 32 levels and therefore is unable to be modeled with. All my other features were chosen because they were obviously related to income (workclass, education, occupation, race, sex, etc) and all contribute to it, as known theoretically.

Logistic Regression - Algorithm # 1

```
glm1 <- glm(income_level~.-native_country, data=train, family=binomial)
glm.fit: fitted probabilities numerically 0 or 1 occurred
summary(glm1) # data exploration function # 4

Call:
glm(formula = income_level ~ . - native_country, family = binomial,
    data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-5.1878  -0.5103  -0.1879  -0.0255   3.6292

Coefficients:
                                      Estimate Std. Error z value
(Intercept)                         -6.839e+00  4.398e-01 -15.551
age                                  2.601e-02  1.782e-03  14.591
workclassLocal-gov                  -6.450e-01  1.188e-01  -5.429
workclassPrivate                    -4.707e-01  9.832e-02  -4.788
workclassSelf-emp-inc               -3.557e-01  1.305e-01  -2.726
workclassSelf-emp-not-inc           -1.044e+00  1.156e-01  -9.034
workclassState-gov                  -7.809e-01  1.316e-01  -5.932
workclassWithout-pay                -8.698e-01  8.391e-01  -1.037
education11th                       -5.727e-02  2.310e-01  -0.248
education12th                        4.781e-01  2.779e-01   1.720
education1st-4th                    -6.945e-01  4.871e-01  -1.426
education5th-6th                    -8.873e-01  3.900e-01  -2.275
education7th-8th                    -5.300e-01  2.521e-01  -2.102
education9th                        -3.869e-01  2.835e-01  -1.365
educationAssoc-acdm                  1.278e+00  1.917e-01   6.666
educationAssoc-voc                   1.178e+00  1.846e-01   6.383
educationBachelors                   1.891e+00  1.720e-01  10.998
educationDoctorate                   2.775e+00  2.336e-01  11.881
educationHS-grad                     7.408e-01  1.672e-01   4.431
educationMasters                     2.201e+00  1.827e-01  12.044
```

```
educationPreschool                     -1.909e+01  9.942e+01  -0.192
educationProf-school                    2.736e+00  2.215e-01  12.353
educationSome-college                   1.140e+00  1.698e-01   6.715
marital_statusMarried-AF-spouse         1.979e+00  6.262e-01   3.161
marital_statusMarried-civ-spouse        2.063e+00  2.862e-01   7.208
marital_statusMarried-spouse-absent     1.117e-01  2.421e-01   0.461
marital_statusNever-married            -5.057e-01  9.605e-02  -5.265
marital_statusSeparated                -1.160e-01  1.864e-01  -0.622
marital_statusWidowed                   1.109e-02  1.722e-01   0.064
occupationArmed-Forces                  2.684e-01  1.187e+00   0.226
occupationCraft-repair                  2.944e-02  8.478e-02   0.347
occupationExec-managerial               7.466e-01  8.191e-02   9.115
occupationFarming-fishing              -9.791e-01  1.478e-01  -6.626
occupationHandlers-cleaners            -8.351e-01  1.551e-01  -5.383
occupationMachine-op-inspct            -3.351e-01  1.077e-01  -3.111
occupationOther-service                -8.966e-01  1.268e-01  -7.071
occupationPriv-house-serv              -2.003e+00  1.029e+00  -1.946
occupationProf-specialty                4.916e-01  8.670e-02   5.670
occupationProtective-serv               3.795e-01  1.338e-01   2.836
occupationSales                         2.302e-01  8.780e-02   2.622
occupationTech-support                  6.579e-01  1.178e-01   5.586
occupationTransport-moving             -4.383e-02  1.046e-01  -0.419
relationshipNot-in-family               2.637e-01  2.828e-01   0.932
relationshipOther-relative             -5.982e-01  2.650e-01  -2.257
relationshipOwn-child                  -7.991e-01  2.819e-01  -2.835
relationshipUnmarried                   2.374e-02  3.021e-01   0.079
relationshipWife                        1.157e+00  1.113e-01  10.395
raceAsian-Pac-Islander                  2.593e-01  2.531e-01   1.025
raceBlack                               1.961e-01  2.396e-01   0.818
raceOther                               7.774e-02  3.659e-01   0.212
raceWhite                               4.103e-01  2.275e-01   1.803
sexMale                                 6.759e-01  8.581e-02   7.877
capital_gain                            3.195e-04  1.141e-05  27.999
capital_loss                            6.281e-04  4.017e-05  15.635
hours_per_week                          3.019e-02  1.781e-03  16.950
                                       Pr(>|z|)
(Intercept)                             < 2e-16 ***
```

```
age                                   < 2e-16 ***
workclassLocal-gov                   5.67e-08 ***
workclassPrivate                     1.69e-06 ***
workclassSelf-emp-inc                 0.00640 **
workclassSelf-emp-not-inc             < 2e-16 ***
workclassState-gov                   2.99e-09 ***
workclassWithout-pay                  0.29992
education11th                         0.80416
education12th                         0.08538 .
education1st-4th                      0.15396
education5th-6th                      0.02290 *
education7th-8th                      0.03555 *
education9th                          0.17232
educationAssoc-acdm                  2.64e-11 ***
educationAssoc-voc                   1.74e-10 ***
educationBachelors                    < 2e-16 ***
educationDoctorate                    < 2e-16 ***
educationHS-grad                     9.38e-06 ***
educationMasters                      < 2e-16 ***
educationPreschool                    0.84775
educationProf-school                  < 2e-16 ***
educationSome-college                1.88e-11 ***
marital_statusMarried-AF-spouse       0.00157 **
marital_statusMarried-civ-spouse     5.68e-13 ***
marital_statusMarried-spouse-absent  0.64459
marital_statusNever-married          1.40e-07 ***
marital_statusSeparated               0.53380
marital_statusWidowed                 0.94865
occupationArmed-Forces                0.82117
occupationCraft-repair                0.72845
occupationExec-managerial             < 2e-16 ***
occupationFarming-fishing            3.45e-11 ***
occupationHandlers-cleaners          7.32e-08 ***
occupationMachine-op-inspct           0.00186 **
occupationOther-service              1.54e-12 ***
occupationPriv-house-serv             0.05163 .
occupationProf-specialty             1.43e-08 ***
```

```
occupationProtective-serv            0.00457 **
occupationSales                      0.00874 **
occupationTech-support              2.33e-08 ***
occupationTransport-moving           0.67521
relationshipNot-in-family            0.35112
relationshipOther-relative           0.02400 *
relationshipOwn-child                0.00459 **
relationshipUnmarried                0.93738
relationshipWife                     < 2e-16 ***
raceAsian-Pac-Islander               0.30558
raceBlack                            0.41314
raceOther                            0.83176
raceWhite                            0.07131 .
sexMale                             3.35e-15 ***
capital_gain                         < 2e-16 ***
capital_loss                         < 2e-16 ***
hours_per_week                       < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 30249  on 27132  degrees of freedom
Residual deviance: 17501  on 27078  degrees of freedom
AIC: 17611

Number of Fisher Scoring iterations: 13
```

Accuracy and Predictions for Logistic Regression

```
probs <- predict(glm1, newdata=test, type="response")
pred <- ifelse(probs>.5, 2, 1)

table(test$income_level, probs >= .5) # data exploration function # 5

        FALSE  TRUE
  <=50K 12581   963
```

```
  >50K   1825  2720
acc <- mean(pred==as.integer(test$income_level))
print("Accuracy for Logistic Regression:")
[1] "Accuracy for Logistic Regression:"
print(acc)
[1] 0.8458732
```

Commentary on Logistic Regression

The logistic regression worked significantly well resulting in a accuracy of 84%. Predictors such as occupation, workclass, sex, race, education were clearly very strong predictors in managing the income level, as shown through '***'. The residual deviance - being at 17501, shows a relatively good response of the algorithm with predictors included, supported by the AIC of 17611. #

Naive Bayes - Algorithm # 2

```
library(e1071)
nb1 <- naiveBayes(income_level~.-native_country, data=train)
nb1

Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
    <=50K      >50K
0.7544319 0.2455681

Conditional probabilities:
       age
Y           [,1]     [,2]
  <=50K 36.77875 13.61431
  >50K  44.11887 10.40446

       workclass
Y         Federal-gov    Local-gov      Private Self-emp-inc
  <=50K 0.0256961407 0.0642403517 0.7677088422 0.0205666830
```

```
  >50K  0.0490769923 0.0787933363 0.6522587423 0.0790935014
       workclass
Y       Self-emp-not-inc    State-gov  Without-pay
  <=50K     0.0800683928 0.0412310699 0.0004885198
  >50K      0.0956025814 0.0448746811 0.0003001651

       education
Y               10th         11th         12th      1st-4th      5th-6th
  <=50K 0.0322911578 0.0455300440 0.0156326331 0.0064484612 0.0120175867
  >50K  0.0076542098 0.0075041273 0.0045024764 0.0009004953 0.0015008255
       education
Y            7th-8th          9th   Assoc-acdm    Assoc-voc    Bachelors
  <=50K 0.0229604299 0.0188080117 0.0333659013 0.0433805569 0.1298974108
  >50K  0.0051028065 0.0034518985 0.0354194807 0.0442743509 0.2781029566
       education
Y          Doctorate      HS-grad      Masters    Preschool  Prof-school
  <=50K 0.0045432340 0.3620420127 0.0337078652 0.0018563752 0.0059110894
  >50K  0.0361698934 0.2153684526 0.1260693381 0.0000000000 0.0528290560
       education
Y       Some-college
  <=50K 0.2316072301
  >50K  0.1811496323

       marital_status
Y           Divorced Married-AF-spouse Married-civ-spouse
  <=50K 0.1671226185      0.0004885198       0.3386419150
  >50K  0.0570313673      0.0013507429       0.8617739757
       marital_status
Y       Married-spouse-absent Never-married    Separated      Widowed
  <=50K          0.0141182218  0.4083048363 0.0377137274 0.0336101612
  >50K           0.0045024764  0.0583821102 0.0072039622 0.0097553655

       occupation
Y       Adm-clerical Armed-Forces Craft-repair Exec-managerial
  <=50K 0.1372740596 0.0002931119 0.1402051783    0.0929653151
  >50K  0.0670868978 0.0003001651 0.1223172745    0.2554404923
       occupation
```

```
Y       Farming-fishing Handlers-cleaners Machine-op-inspct Other-service
  <=50K    0.0376648754      0.0559355154      0.0770395701  0.1331704934
  >50K     0.0160588324      0.0105057782      0.0331682425  0.0178598229
       occupation
Y       Priv-house-serv Prof-specialty Protective-serv        Sales
  <=50K    0.0071812408   0.0984855887    0.0200781632 0.1178798241
  >50K     0.0001500825   0.2396818250    0.0262644454 0.1284706589
       occupation
Y       Tech-support Transport-moving
  <=50K 0.0290669272     0.0527601368
  >50K  0.0382710491     0.0444244334

       relationship
Y           Husband Not-in-family Other-relative   Own-child   Unmarried
  <=50K 0.299804592   0.310845139    0.039667807 0.189350269 0.129408891
  >50K  0.764820651   0.103707039    0.004202311 0.008704788 0.024913703
       relationship
Y              Wife
  <=50K 0.030923302
  >50K  0.093651508

       race
Y       Amer-Indian-Eskimo Asian-Pac-Islander       Black       Other
  <=50K        0.010649731        0.027796776 0.108353688 0.009135320
  >50K         0.005102807        0.032868077 0.047426084 0.003151733
       race
Y             White
  <=50K 0.844064485
  >50K  0.911451298

       sex
Y          Female      Male
  <=50K 0.3821690 0.6178310
  >50K  0.1491821 0.8508179

       capital_gain
Y            [,1]       [,2]
```

```
  <=50K  150.1334   968.1396
  >50K  3768.8972 13932.5981

       capital_loss
Y             [,1]     [,2]
  <=50K  54.55256 313.7422
  >50K  192.88189 589.8750

       hours_per_week
Y            [,1]     [,2]
  <=50K 39.36624 11.96859
  >50K  45.78013 10.80015
```

Accuracy and Predictions for Naive Bayes

```
p1 <- predict(nb1, newdata=test, type="class")
acc <- mean(p1==test$income_level) #calculating the accuracy
print("Accuracy for Naive Bayes:")
[1] "Accuracy for Naive Bayes:"
print(acc)
[1] 0.811156
```

Commentary on Naive Bayes

The Naive Bayes algorithm also works relatively well on this data set with an accuracy of 81%. Since the algorithm performs simple likelihood chances, it gave accurate results in the A-priori probabilities for the income level. #

Decision Trees - Algorithm 3

```
library(tree)
dtree1 <- tree(income_level~.-native_country, data=train)
summary(dtree1)

Classification tree:
tree(formula = income_level ~ . - native_country, data = train)
Variables actually used in tree construction:
[1] "relationship" "capital_gain" "education"    "occupation"
Number of terminal nodes:  8
```

```
Residual mean deviance:  0.7037 = 19090 / 27120
Misclassification error rate: 0.1597 = 4334 / 27133
```

Accuracy and Predictions for Decision Trees

```
p4 <- predict(dtree1, newdata=test, type="class")
accuracy4 <- mean(p4==test$income_level)
print("Accuracy for Decision Trees:")
[1] "Accuracy for Decision Trees:"
print(accuracy4)
[1] 0.8405108
```

Commentary for Decision Trees

The decision tree algorithm worked quite efficiently, almsot at the same level of accuracy as logistic regression in that it gave similar residual mean deviance, and had a significantly low misclassification error rate, assuring that the algorithm received and partitioned the data efficiently.

Random Forest - Ensemble Method

```
library(randomForest)
set.seed(1234)
rf <- randomForest(income_level~.-native_country, data=train, importance=TRUE)
pred <- predict(rf, newdata=test, type="response")
acc_rf <- mean(pred==test$income_level)
print("Accuracy for Random Forest")
[1] "Accuracy for Random Forest"
print(acc_rf)
[1] 0.8594726
```

Commentary for Random Forest

Random Forest outperformed all the other algorithms significantly with an accuracy of 85.9%

Results Analysis

Logistic Regression Accuracy: 84.58732 Decision Trees Accuracy: 84.05108 Naive Bayes Accuracy: 81.1156

As a natural classification algorithm, it is expected that Logistic Regression outperformed on this specific dataset as the predictors were clearly very connected to the target and were easier to predict compared to hypothetical predictors that weren't as correlated to the target (income). In comparison to Naive Bayes, both algorithms showed relatively the same p-values on strong predictors, assuring that they both

understood the data. But logistic regression has always outperformed naive bayes on simpler datasets, this dataset would be considered simple as its predictors were expectable and clear.
Naive Bayes did not perform as well for certain predictors, such as “relationship,” where it separated the predictors very conditionally, making “Husband” over 70% of the probability.

Decision Trees, on the other hand, showed a relatively good residual deviance at .7037 which was just about as close to logistic regression’s residual deviance. Therefore it justifies why the algorithms’ performance were so close to each other. Since Decision Trees do partitions over classification data sets using recursive, greedy methods, and do not rely on dummy variables, they were able to efficiently perform on this data set.

Using the R script, the algorithms were able to understand how significant predictors such as occupation, workclass, education levels, gender and race were all able to have extremely strong impact on income-levels, whereas predictors such as native_country were not as strong.