

Classification

Aloksai Choudari, Alekhya Pinnamaneni

2022-09-25

How do linear models for classification work?

Linear models for classification create boundaries for decisions, which separate the observations into different categories that are of the same class. The decision boundaries in classification are linear combinations of different parameters for each side of the boundary. Strengths for these linear models include: the ability to add new data and update the graph with an easier approach, better probabilistic interpretations, and the avoidance of overfitting with algorithms. A few weaknesses for linear models in classification are: tendencies to fail with an increase in non-linear decision boundaries and less flexibility to adopt relationships as they get more complex.

Select a dataset

Data set: Adult Data Set

Source: <https://archive.ics.uci.edu/ml/datasets/Adult>

Target column: 'predicted_salary_range'

No. of rows: 48,842 rows

Load the data

```
adult <- read.csv("adult.data", header=FALSE)

# Adds columns names to the data table
colnames(adult) <- c('age', 'workclass', 'fnlwgt', 'education', 'education_num', 'marital_status', 'occu
```

Data Cleaning

```
# Changes the character values in the predicted_salary_range column to integer values
adult$predicted_salary_range[adult$predicted_salary_range == "<=50K"] <- "0"
adult$predicted_salary_range[adult$predicted_salary_range == ">50K"] <- "1"
adult$predicted_salary_range <- as.integer(adult$predicted_salary_range)
```

Split data into train and test data

```
set.seed(1234)
sample <- sample(1:nrow(adult), nrow(adult)*0.8, replace=FALSE)
train <- adult[sample,]
test <- adult[-sample,]
```

Data exploration on the train data

```
attach(train)

# Prints the first 10 rows of the train data for adults
head(train, n=10)
```

##	age	workclass	fnlwgt	education	education_num	marital_status
## 7452	17	Private	110798	11th	7	Never-married
## 8016	34	Private	202450	HS-grad	9	Married-civ-spouse
## 7162	24	Private	259351	Some-college	10	Never-married
## 8086	67	?	81761	HS-grad	9	Divorced
## 23653	25	Private	109532	12th	8	Never-married
## 9196	24	Private	237928	Bachelors	13	Never-married
## 623	65	Private	109351	9th	5	Widowed
## 15241	44	Private	368757	Some-college	10	Married-civ-spouse
## 10885	45	Private	189225	HS-grad	9	Never-married
## 934	23	Private	375871	HS-grad	9	Married-civ-spouse
##	occupation	relationship	race	sex		
## 7452	Sales	Own-child	White	Female		
## 8016	Craft-repair	Husband	White	Male		
## 7162	Craft-repair	Unmarried	Amer-Indian-Eskimo	Male		
## 8086	?	Own-child	White	Male		
## 23653	Craft-repair	Own-child	White	Male		
## 9196	Prof-specialty	Not-in-family	White	Male		
## 623	Priv-house-serv	Unmarried	Black	Female		
## 15241	Machine-op-inspct	Husband	White	Male		
## 10885	Other-service	Unmarried	Black	Female		
## 934	Adm-clerical	Wife	White	Female		
##	capital_gain	capital_loss	hours_per_week	native_country		
## 7452	0	0	20	United-States		
## 8016	0	0	55	United-States		
## 7162	0	0	40	Mexico		
## 8086	0	0	20	United-States		
## 23653	0	0	40	United-States		
## 9196	0	0	39	United-States		
## 623	0	0	24	United-States		
## 15241	0	0	40	United-States		
## 10885	0	0	40	United-States		
## 934	0	0	40	Mexico		
##	predicted_salary_range					
## 7452	0					
## 8016	1					
## 7162	0					
## 8086	0					
## 23653	0					

```
## 9196          0
## 623           0
## 15241         0
## 10885         0
## 934           0
```

```
# Prints the mean of education_num
mean(education_num)
```

```
## [1] 10.08561
```

```
# Prints the median of hours worked per week for adults
median(hours_per_week)
```

```
## [1] 40
```

```
# Prints the smallest and largest capital_gain across the adults
range(capital_gain)
```

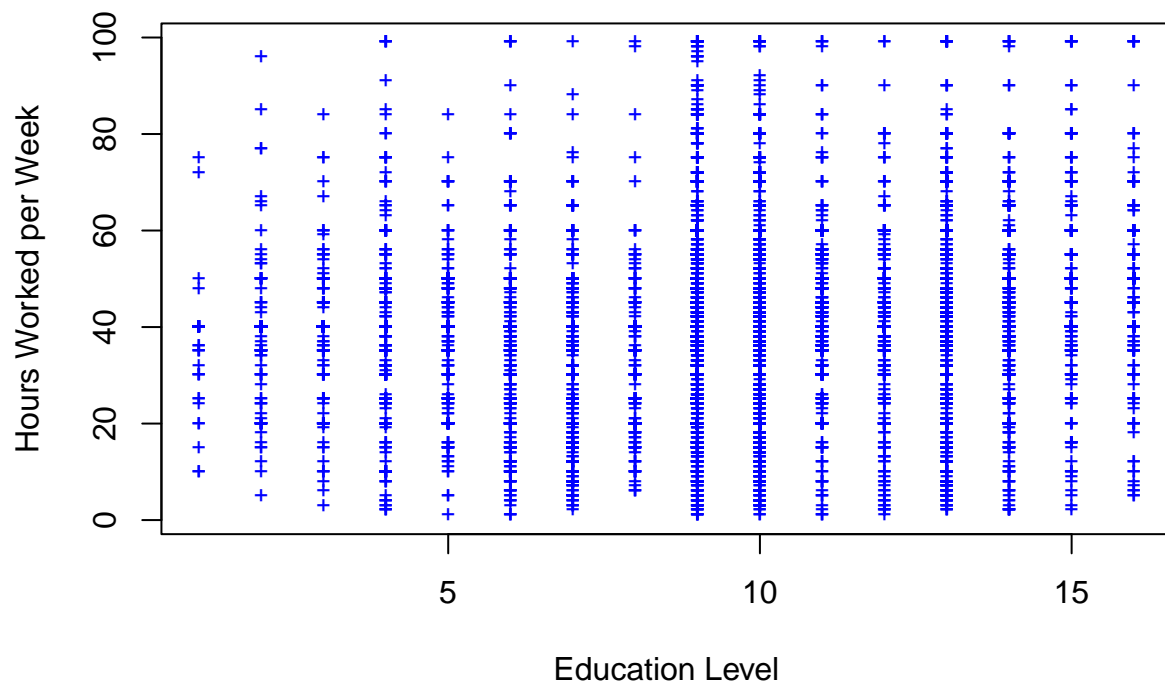
```
## [1]      0 99999
```

```
# Prints statistics for the age across the adults data
summary(age)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      17.00   28.00   37.00   38.59   48.00   90.00
```

Informative graphs of train data

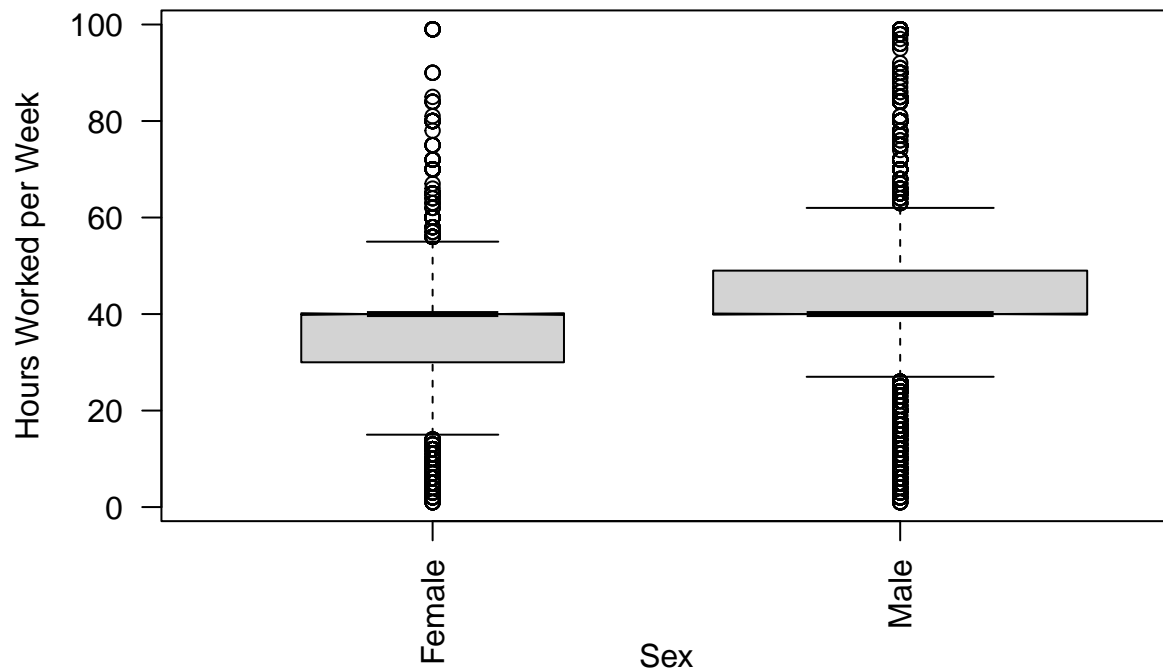
```
# Scatterplot of education level vs. hours worked per week
plot(train$education_num, train$hours_per_week, pch='+', cex=0.75, col="blue", xlab="Education Level", ylab="Hours Worked per Week")
```



```
# Boxplot of hours worked per week based on sex
```

```
boxplot(train$hours_per_week~train$sex, varwidth=TRUE, notch=TRUE, xlab="Sex", ylab="Hours Worked per W
```

```
## Warning in (function (z, notch = FALSE, width = NULL, varwidth = FALSE, : some
## notches went outside hinges ('box'): maybe set notch=FALSE
```



Logistic regression model of train data

```
glm1 <- glm(predicted_salary_range~education_num, data=train)
glm1
```

```
##
## Call:  glm(formula = predicted_salary_range ~ education_num, data = train)
##
## Coefficients:
##   (Intercept)  education_num
##      -0.32142      0.05566
##
## Degrees of Freedom: 26047 Total (i.e. Null);  26046 Residual
## Null Deviance:      4750
## Residual Deviance: 4213  AIC: 26470
```

```
# Outputs the summary of the model
summary(glm1)
```

```
##
## Call:
## glm(formula = predicted_salary_range ~ education_num, data = train)
##
```

```
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.5691  -0.2352  -0.1795   0.1544   1.2101
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.3214220  0.0100529  -31.97  <2e-16 ***
## education_num  0.0556599  0.0009657   57.64  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.1617512)
##
##      Null deviance: 4750.4  on 26047  degrees of freedom
## Residual deviance: 4213.0  on 26046  degrees of freedom
## AIC: 26473
##
## Number of Fisher Scoring iterations: 2
```

The residual deviance in the model summary shows how well the predicted_salary_range can be predicted by the model with education_num as the predictor variable. AIC is a measure of how well-fit the model is to the data set. A lower AIC value indicates a better-fitting model. The AIC value for this model is fairly large, meaning this model is underfit for the data.

Naive Bayes Model

```
#install.packages('e1071', dependencies=TRUE)
library(e1071)
nb <- naiveBayes(predicted_salary_range~., data=train)
nb

##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##      0      1
## 0.7600584 0.2399416
##
## Conditional probabilities:
##      age
## Y      [,1]      [,2]
## 0 36.81897 14.08123
## 1 44.21488 10.41266
##
##      workclass
## Y      ? Federal-gov Local-gov Never-worked Private
## 0 0.0673805435 0.0232851803 0.0596524902 0.0003030609 0.7177997778
## 1 0.0244800000 0.0483200000 0.0795200000 0.0000000000 0.6326400000
```

```

##      workclass
## Y      Self-emp-inc  Self-emp-not-inc      State-gov  Without-pay
## 0  0.0196484493      0.0739468633 0.0374785332 0.0005051015
## 1  0.0803200000      0.0902400000 0.0444800000 0.0000000000
##
##      fnlwgt
## Y      [,1]      [,2]
## 0 189674.5 105991.1
## 1 187899.4 101477.8
##
##      education
## Y      10th      11th      12th      1st-4th      5th-6th      7th-8th
## 0 0.036013739 0.044751995 0.016314779 0.006566320 0.012779069 0.025659157
## 1 0.007360000 0.007680000 0.004320000 0.000800000 0.001760000 0.004160000
##      education
## Y      9th  Assoc-acdm  Assoc-voc  Bachelors  Doctorate  HS-grad
## 0 0.019345388 0.032983130 0.041266795 0.127891706 0.004293363 0.354278210
## 1 0.003360000 0.035200000 0.044640000 0.287680000 0.038240000 0.210240000
##      education
## Y      Masters  Preschool  Prof-school  Some-college
## 0 0.031467825 0.002171937 0.006566320 0.237650268
## 1 0.121280000 0.000000000 0.053920000 0.179360000
##
##      education_num
## Y      [,1]      [,2]
## 0  9.597939 2.452206
## 1 11.630400 2.359654
##
##      marital_status
## Y      Divorced  Married-AF-spouse  Married-civ-spouse  Married-spouse-absent
## 0 0.1597131023      0.0005556117      0.3353369027      0.0159106981
## 1 0.0614400000      0.0016000000      0.8544000000      0.0043200000
##      marital_status
## Y      Never-married  Separated  Widowed
## 0  0.4139812102 0.0374785332 0.0370239418
## 1  0.0596800000 0.0086400000 0.0099200000
##
##      occupation
## Y      ?  Adm-clerical  Armed-Forces  Craft-repair  Exec-managerial
## 0 0.0676836044 0.1323365997 0.0003535711 0.1277906859 0.0854631781
## 1 0.0244800000 0.0652800000 0.0001600000 0.1190400000 0.2483200000
##      occupation
## Y      Farming-fishing  Handlers-cleaners  Machine-op-inspct  Other-service
## 0  0.0354581271      0.0518234165      0.0708657440 0.1273866047
## 1  0.0129600000      0.0105600000      0.0331200000 0.0179200000
##      occupation
## Y      Priv-house-serv  Prof-specialty  Protective-serv  Sales
## 0  0.0057581574      0.0923325588      0.0175270229 0.1080412163
## 1  0.0001600000      0.2382400000      0.0272000000 0.1265600000
##      occupation
## Y      Tech-support  Transport-moving
## 0 0.0257096676      0.0514698454
## 1 0.0352000000      0.0408000000
##

```

```

## relationship
## Y Husband Not-in-family Other-relative Own-child Unmarried Wife
## 0 0.29487827 0.30361653 0.03732700 0.20097990 0.12996262 0.03323568
## 1 0.75488000 0.10864000 0.00368000 0.00832000 0.02784000 0.09664000
##
## race
## Y Amer-Indian-Eskimo Asian-Pac-Islander Black Other White
## 0 0.01136478 0.03101323 0.10940499 0.00989999 0.83831700
## 1 0.00480000 0.03440000 0.05184000 0.00352000 0.90544000
##
## sex
## Y Female Male
## 0 0.3863522 0.6136478
## 1 0.1529600 0.8470400
##
## capital_gain
## Y [,1] [,2]
## 0 149.6236 970.3204
## 1 4018.6848 14581.7440
##
## capital_loss
## Y [,1] [,2]
## 0 51.62001 307.1221
## 1 194.93104 594.0050
##
## hours_per_week
## Y [,1] [,2]
## 0 38.84317 12.29426
## 1 45.44384 10.94613
##
## native_country
## Y ? Cambodia Canada China Columbia
## 0 1.818365e-02 5.051015e-04 2.980099e-03 2.121426e-03 2.070916e-03
## 1 1.872000e-02 8.000000e-04 5.120000e-03 2.720000e-03 1.600000e-04
## native_country
## Y Cuba Dominican-Republic Ecuador El-Salvador England
## 0 2.929589e-03 2.727548e-03 1.111223e-03 3.687241e-03 2.474997e-03
## 1 2.560000e-03 3.200000e-04 4.800000e-04 1.280000e-03 3.840000e-03
## native_country
## Y France Germany Greece Guatemala Haiti
## 0 6.061218e-04 4.040812e-03 6.566320e-04 2.778058e-03 1.515305e-03
## 1 1.120000e-03 5.760000e-03 1.120000e-03 1.600000e-04 6.400000e-04
## native_country
## Y Holand-Netherlands Honduras Hong Hungary India
## 0 5.051015e-05 4.040812e-04 6.061218e-04 4.040812e-04 2.323467e-03
## 1 0.000000e+00 1.600000e-04 4.800000e-04 3.200000e-04 4.480000e-03
## native_country
## Y Iran Ireland Italy Jamaica Japan
## 0 1.010203e-03 9.091827e-04 2.121426e-03 2.778058e-03 1.363774e-03
## 1 2.080000e-03 4.800000e-04 3.520000e-03 1.120000e-03 2.400000e-03
## native_country
## Y Laos Mexico Nicaragua Outlying-US(Guam-USVI-etc)
## 0 6.566320e-04 2.429538e-02 1.111223e-03 6.566320e-04
## 1 1.600000e-04 4.000000e-03 3.200000e-04 0.000000e+00

```



```
## native_country
## Y Peru Philippines Poland Portugal Puerto-Rico
## 0 1.212244e-03 5.909688e-03 1.969896e-03 1.262754e-03 3.737751e-03
## 1 3.200000e-04 8.480000e-03 1.280000e-03 6.400000e-04 1.600000e-03
## native_country
## Y Scotland South Taiwan Thailand Trinidad&Tobago
## 0 3.535711e-04 2.626528e-03 1.060713e-03 7.071421e-04 8.586726e-04
## 1 4.800000e-04 2.240000e-03 2.400000e-03 1.600000e-04 3.200000e-04
## native_country
## Y United-States Vietnam Yugoslavia
## 0 8.902919e-01 2.576018e-03 3.535711e-04
## 1 9.166400e-01 4.800000e-04 6.400000e-04
```

The data above displays the probability of each result (salary \leq 50K or salary $>$ 50K) based on the value of each attribute.

Predict and evaluate on the test data

```
probs <- predict(glm1, newdata=test, type="response")
pred <- ifelse(probs>0.5, 2, 1)

# Calculate accuracy
acc1 <- mean(pred==as.integer(test$predicted_salary_range))
print(paste("accuracy = ", acc1))
```

Logistic Regression Model

```
## [1] "accuracy = 0.220789190849071"
```

```
# Table of predictions and true values
tab <- table(pred, as.integer(test$predicted_salary_range))
tab
```

```
##
## pred 0 1
## 1 4877 1438
## 2 45 153
```

```
TP <- tab[1, 1]
FN <- tab[2, 1]
TN <- tab[2, 2]
FP <- tab[1, 2]
```

```
# Sensitivity
sens <- TP / (TP + FN)
print(paste("sensitivity = ", sens))
```

```
## [1] "sensitivity = 0.990857375050792"
```

```
# Specificity
spec <- TN / (TN + FP)
print(paste("specificity = ", spec))
```

```
## [1] "specificity = 0.0961659333752357"
```

```
p1 <- predict(nb, newdata=test, type="class")

# Calculate accuracy
acc2 <- mean(p1==(test$predicted_salary_range))
print(paste("accuracy = ", acc2))
```

Naive Bayes Model

```
## [1] "accuracy = 0.826193766313527"
```

```
# Table of predictions and true values
tab2 <- table(p1, test$predicted_salary_range)
tab2
```

```
##
## p1      0      1
##    0 4608  818
##    1  314  773
```

```
TP2 <- tab2[1, 1]
FN2 <- tab2[2, 1]
TN2 <- tab2[2, 2]
FP2 <- tab2[1, 2]
```

```
# Sensitivity
sens2 <- TP2 / (TP2 + FN2)
print(paste("sensitivity = ", sens2))
```

```
## [1] "sensitivity = 0.936204794798862"
```

```
# Specificity
spec2 <- TN2 / (TN2 + FP2)
print(paste("specificity = ", spec2))
```

```
## [1] "specificity = 0.48585795097423"
```

The naive bayes model has a much higher accuracy rate compared to the logistic regression model. This can be explained by the fact that the naive bayes model uses all attributes as predictors to make a more accurate prediction of the target variable (predicted_salary_range).

Strengths and weaknesses of Naive Bayes and Logistic Regression

Logistic regression is accurate for simpler, more linear data sets. However, it can only be useful for data sets that follow a linear trend. The naive bayes algorithm computes a model quickly in a short amount of time. However, the model assumes that all predictor attributes are independent of each other, which is rarely true.

Classification metrics

The most common and simplest classification metric is accuracy. This represents the proportion of predictions made by the model that are accurate. Sensitivity measures the rate of true positives. Specificity measures the rate of true negatives.