Laporan Praktikum 3 AMP

Antonius Aditya Rizky Wijaya

G5402221003

2025-02-06

# Classification Methods

## The Stock Market Data

library(ISLR2)

## Warning: package 'ISLR2' was built under R version 4.3.3

names(Smarket)

## [1] "Year" "Lag1" "Lag2" "Lag3" "Lag4" "Lag5"   
## [7] "Volume" "Today" "Direction"

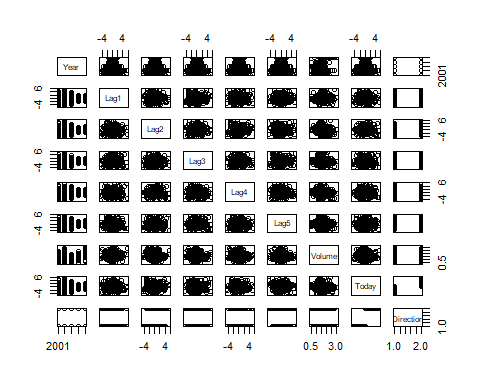
dim(Smarket)

## [1] 1250 9

summary(Smarket)

## Year Lag1 Lag2 Lag3   
## Min. :2001 Min. :-4.922000 Min. :-4.922000 Min. :-4.922000   
## 1st Qu.:2002 1st Qu.:-0.639500 1st Qu.:-0.639500 1st Qu.:-0.640000   
## Median :2003 Median : 0.039000 Median : 0.039000 Median : 0.038500   
## Mean :2003 Mean : 0.003834 Mean : 0.003919 Mean : 0.001716   
## 3rd Qu.:2004 3rd Qu.: 0.596750 3rd Qu.: 0.596750 3rd Qu.: 0.596750   
## Max. :2005 Max. : 5.733000 Max. : 5.733000 Max. : 5.733000   
## Lag4 Lag5 Volume Today   
## Min. :-4.922000 Min. :-4.92200 Min. :0.3561 Min. :-4.922000   
## 1st Qu.:-0.640000 1st Qu.:-0.64000 1st Qu.:1.2574 1st Qu.:-0.639500   
## Median : 0.038500 Median : 0.03850 Median :1.4229 Median : 0.038500   
## Mean : 0.001636 Mean : 0.00561 Mean :1.4783 Mean : 0.003138   
## 3rd Qu.: 0.596750 3rd Qu.: 0.59700 3rd Qu.:1.6417 3rd Qu.: 0.596750   
## Max. : 5.733000 Max. : 5.73300 Max. :3.1525 Max. : 5.733000   
## Direction   
## Down:602   
## Up :648   
##   
##   
##   
##

pairs(Smarket)



Keterangan :

Melakukan eksplorasi awal terhadap data untuk memahami struktur dan distribusinya.

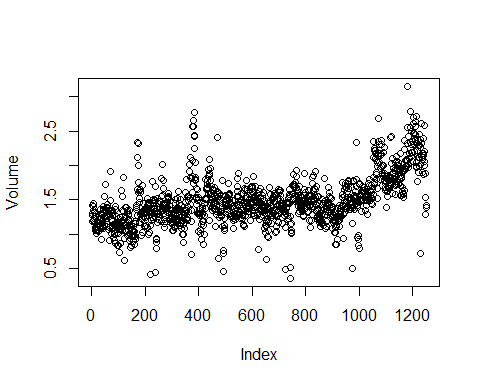
#cor(Smarket)  
cor(Smarket[, -9])

## Year Lag1 Lag2 Lag3 Lag4  
## Year 1.00000000 0.029699649 0.030596422 0.033194581 0.035688718  
## Lag1 0.02969965 1.000000000 -0.026294328 -0.010803402 -0.002985911  
## Lag2 0.03059642 -0.026294328 1.000000000 -0.025896670 -0.010853533  
## Lag3 0.03319458 -0.010803402 -0.025896670 1.000000000 -0.024051036  
## Lag4 0.03568872 -0.002985911 -0.010853533 -0.024051036 1.000000000  
## Lag5 0.02978799 -0.005674606 -0.003557949 -0.018808338 -0.027083641  
## Volume 0.53900647 0.040909908 -0.043383215 -0.041823686 -0.048414246  
## Today 0.03009523 -0.026155045 -0.010250033 -0.002447647 -0.006899527  
## Lag5 Volume Today  
## Year 0.029787995 0.53900647 0.030095229  
## Lag1 -0.005674606 0.04090991 -0.026155045  
## Lag2 -0.003557949 -0.04338321 -0.010250033  
## Lag3 -0.018808338 -0.04182369 -0.002447647  
## Lag4 -0.027083641 -0.04841425 -0.006899527  
## Lag5 1.000000000 -0.02200231 -0.034860083  
## Volume -0.022002315 1.00000000 0.014591823  
## Today -0.034860083 0.01459182 1.000000000

Keterangan :

Melihat hubungan linear antar variabel numerik untuk memahami mana yang mungkin berhubungan dengan Direction.

attach(Smarket)  
plot(Volume)



Keterangan :

Melihat bagaimana tren Volume perdagangan berubah selama periode waktu yang ada dalam dataset.

### Code Halaman 175

train <- (Year < 2005)  
Smarket.2005 <- Smarket[!train, ]  
dim(Smarket.2005)

## [1] 252 9

Direction.2005 <- Direction[!train]

Keterangan :

Pada baris awal kita mencoba memisahkah training set dengan test set. Data sebelum 2005 (training set), dan data dari 2005 ke atas (test set). Lalu dengan Smarket[!train, ], kita memilih hanya data dengan Year >= 2005, sehingga Smarket.2005 berisi hanya test set. Fungsi dim() menampilkan jumlah baris dan kolom dari Smarket.2005, yaitu . Selanjutnya, Direction adalah variabel respons yang menunjukkan apakah pasar saham naik (“Up”) atau turun (“Down”). Direction[!train] mengambil hanya nilai Direction dari test set (2005 ke atas) dan menyimpannya di Direction.2005.

## -Nearest Neighbors

library(class)  
train.X <- cbind(Lag1, Lag2)[train, ]  
test.X <- cbind(Lag1, Lag2)[!train, ]  
train.Direction <- Direction[train]

Keterangan :

Dataset dipecah menjadi training set dan test set, hanya menggunakan prediktor Lag1 dan Lag2 karena diduga memiliki hubungan dengan Direction. Sementara train.Direction berisi label kategori untuk data latih.

set.seed(1)  
knn.pred <- knn(train.X, test.X, train.Direction, k = 1)  
table(knn.pred, Direction.2005)

## Direction.2005  
## knn.pred Down Up  
## Down 43 58  
## Up 68 83

(83 + 43) / 252

## [1] 0.5

Keterangan :

Model KNN dengan dibuat dan hasilnya dibandingkan dengan data aktual. Akurasi dihitung dengan , yang menunjukkan model ini tidak jauh lebih baik dari tebakan acak.

knn.pred <- knn(train.X, test.X, train.Direction, k = 3)  
table(knn.pred, Direction.2005)

## Direction.2005  
## knn.pred Down Up  
## Down 48 54  
## Up 63 87

mean(knn.pred == Direction.2005)

## [1] 0.5357143

Keterangan :

Model KNN dengan diuji untuk melihat apakah lebih baik dari . Menggunakan mean(knn.pred == Direction.2005), akurasi dihitung dan dibandingkan dengan model sebelumnya.

dim(Caravan)

## [1] 5822 86

attach(Caravan)  
summary(Purchase)

## No Yes   
## 5474 348

348 / 5822

## [1] 0.05977327

Keterangan :

Dataset Caravan memiliki observasi dan target variabel Purchase. Hanya orang yang membeli asuransi (sekitar ), menunjukkan dataset sangat tidak seimbang.

standardized.X <- scale(Caravan[, -86])  
var(Caravan[, 1])

## [1] 165.0378

var(Caravan[, 2])

## [1] 0.1647078

var(standardized.X[, 1])

## [1] 1

var(standardized.X[, 2])

## [1] 1

Keterangan :

Variabel prediktor distandarisasi menggunakan scale(), karena KNN sensitif terhadap skala variabel. Setelah standarisasi, semua variabel memiliki varians yang seragam.

test <- 1:1000  
train.X <- standardized.X[-test, ]  
test.X <- standardized.X[test, ]  
train.Y <- Purchase[-test]  
test.Y <- Purchase[test]  
set.seed(1)  
knn.pred <- knn(train.X, test.X, train.Y, k = 1)  
mean(test.Y != knn.pred)

## [1] 0.118

mean(test.Y != "No")

## [1] 0.059

Keterangan :

observasi pertama digunakan sebagai test set, sisanya sebagai training set. Model KNN dengan diuji dan error rate dihitung. Akurasi model tidak terlalu baik karena dataset tidak seimbang (mean(test.Y != "No") menunjukkan mayoritas prediksi adalah “No”).

table(knn.pred, test.Y)

## test.Y  
## knn.pred No Yes  
## No 873 50  
## Yes 68 9

9 / (68 + 9)

## [1] 0.1168831

Keterangan :

Dari prediksi “Yes”, hanya yang benar. Rasio keberhasilan deteksi pembelian asuransi sangat kecil, sekitar . Model cenderung bias ke “No” karena mayoritas sampel memang “No”.

knn.pred <- knn(train.X, test.X, train.Y, k = 3)  
table(knn.pred, test.Y)

## test.Y  
## knn.pred No Yes  
## No 920 54  
## Yes 21 5

5 / 26

## [1] 0.1923077

knn.pred <- knn(train.X, test.X, train.Y, k = 5)  
table(knn.pred, test.Y)

## test.Y  
## knn.pred No Yes  
## No 930 55  
## Yes 11 4

4 / 15

## [1] 0.2666667

Keterangan :

Dengan , hanya 5 dari 26 prediksi “Yes” yang benar, sekitar akurasi pada kelas “Yes”. Dengan , hanya 4 dari 15 prediksi “Yes” yang benar, sekitar akurasi pada kelas “Yes”. Semakin besar , semakin sedikit false positive, tetapi semakin banyak false negative.

glm.fits <- glm(Purchase ~ ., data = Caravan,  
 family = binomial, subset = -test)

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

glm.probs <- predict(glm.fits, Caravan[test, ],  
 type = "response")  
glm.pred <- rep("No", 1000)  
glm.pred[glm.probs > .5] <- "Yes"  
table(glm.pred, test.Y)

## test.Y  
## glm.pred No Yes  
## No 934 59  
## Yes 7 0

glm.pred <- rep("No", 1000)  
glm.pred[glm.probs > .25] <- "Yes"  
table(glm.pred, test.Y)

## test.Y  
## glm.pred No Yes  
## No 919 48  
## Yes 22 11

11 / (22 + 11)

## [1] 0.3333333

Keterangan :

Model regresi logistik dibuat dan dilatih hanya pada training test. Jika menggunakan ambang batas probabilitas , hampir semua prediksi adalah “No”, karena dataset tidak seimbang. Jika ambang batas diturunkan ke , lebih banyak “Yes” yang terdeteksi. Akurasi untuk mendeteksi pembelian asuransi tetap rendah pada kategori “Yes”).

# Exercise

# Nomor 13

This question should be answered using the Weekly data set, which is part of the ISLR2 package. This data is similar in nature to the Smarket data from this chapter’s lab, except that it contains 1,089 weekly returns for 21 years, from the beginning of 1990 to the end of 2010.

library(ISLR2)  
library(class)  
data(Weekly)  
names(Weekly)

## [1] "Year" "Lag1" "Lag2" "Lag3" "Lag4" "Lag5"   
## [7] "Volume" "Today" "Direction"

dim(Weekly)

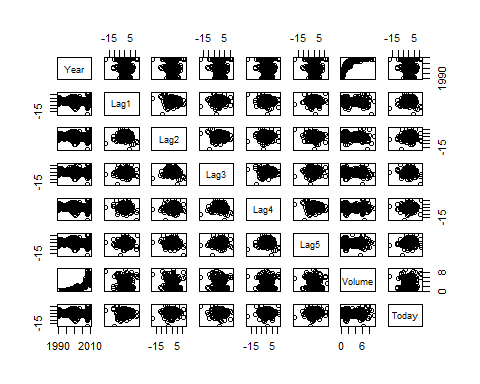
## [1] 1089 9

1. Produce some numerical and graphical summaries of the Weekly data. Do there appear to be any patterns?

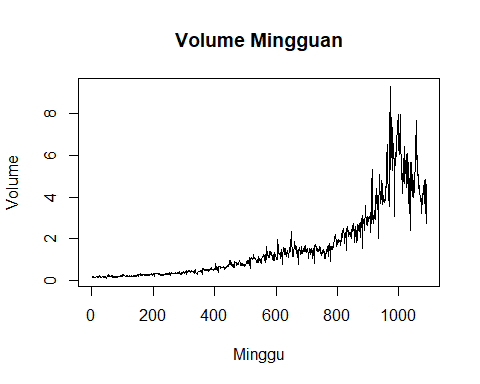
summary(Weekly)

## Year Lag1 Lag2 Lag3   
## Min. :1990 Min. :-18.1950 Min. :-18.1950 Min. :-18.1950   
## 1st Qu.:1995 1st Qu.: -1.1540 1st Qu.: -1.1540 1st Qu.: -1.1580   
## Median :2000 Median : 0.2410 Median : 0.2410 Median : 0.2410   
## Mean :2000 Mean : 0.1506 Mean : 0.1511 Mean : 0.1472   
## 3rd Qu.:2005 3rd Qu.: 1.4050 3rd Qu.: 1.4090 3rd Qu.: 1.4090   
## Max. :2010 Max. : 12.0260 Max. : 12.0260 Max. : 12.0260   
## Lag4 Lag5 Volume Today   
## Min. :-18.1950 Min. :-18.1950 Min. :0.08747 Min. :-18.1950   
## 1st Qu.: -1.1580 1st Qu.: -1.1660 1st Qu.:0.33202 1st Qu.: -1.1540   
## Median : 0.2380 Median : 0.2340 Median :1.00268 Median : 0.2410   
## Mean : 0.1458 Mean : 0.1399 Mean :1.57462 Mean : 0.1499   
## 3rd Qu.: 1.4090 3rd Qu.: 1.4050 3rd Qu.:2.05373 3rd Qu.: 1.4050   
## Max. : 12.0260 Max. : 12.0260 Max. :9.32821 Max. : 12.0260   
## Direction   
## Down:484   
## Up :605   
##   
##   
##   
##

pairs(Weekly[, -9])



plot(Weekly$Volume, type = "l", main = "Volume Mingguan", ylab = "Volume", xlab = "Minggu")



Year dan Volume tampaknya memiliki hubungan, trennya relatif positif.

1. Repeat (d) using KNN with .

library(class)  
train <- Weekly$Year < 2009  
test <- Weekly$Year > 2008  
  
knn\_model <- knn(  
 Weekly[train, "Lag2", drop = FALSE],  
 Weekly[test, "Lag2", drop = FALSE],  
 Weekly$Direction[train], k = 1  
)  
(confusion\_matrix\_knn <- table(Prediction = knn\_model, Actual = Weekly[test, ]$Direction))

## Actual  
## Prediction Down Up  
## Down 21 29  
## Up 22 32

(akurasi\_knn <- mean(knn\_model == Weekly[test, ]$Direction))

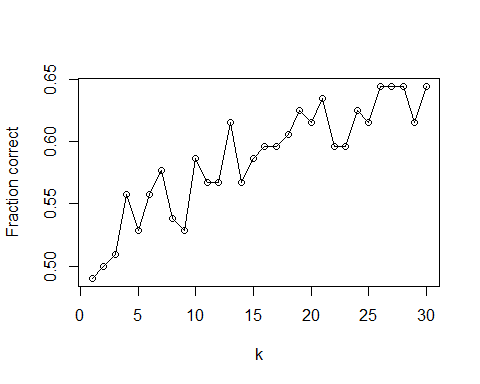
## [1] 0.5096154

#sum(diag(confusion\_matrix\_knn)) / sum(confusion\_matrix\_knn)

Dengan model KNN pada , didapat akurasi

1. Experiment with different combinations of predictors, including possible transformations and interactions, for each of the methods. Report the variables, method, and associated confusion matrix that appears to provide the best results on the held out data. Note that you should also experiment with values for in the KNN classifier.

set.seed(1)  
res <- sapply(1:30, function(k) {  
 fit <- knn(  
 Weekly[train, 2:4, drop = FALSE],  
 Weekly[test, 2:4, drop = FALSE],  
 Weekly$Direction[train],  
 k = k  
 )  
 mean(fit == Weekly[test, ]$Direction)  
})  
plot(1:30, res, type = "o", xlab = "k", ylab = "Fraction correct")



(k <- which.max(res))

## [1] 26

fit <- knn(  
 Weekly[train, 2:4, drop = FALSE],  
 Weekly[test, 2:4, drop = FALSE],  
 Weekly$Direction[train],  
 k = k  
)  
table(Prediction = fit, Aktual = Weekly[test, ]$Direction)

## Aktual  
## Prediction Down Up  
## Down 23 18  
## Up 20 43

mean(fit == Weekly[test, ]$Direction)

## [1] 0.6346154

KNN menggunakan variabel 3 Lag pertama performa marginalnya lebih baik dari regresi logistik dengan Lag2 jika kita setel menjadi dengan tingkat akurasi

# Nomor 14

In this problem, you will develop a model to predict whether a given car gets high or low gas mileage based on the Auto data set.

1. Create a binary variable, mpg01, that contains a 1 if mpg contains a value above its median, and a 0 if mpg contains a value below its median. You can compute the median using the median() function. Note you may find it helpful to use the data.frame() function to create a single data set containing both mpg01 and the other Auto variables.

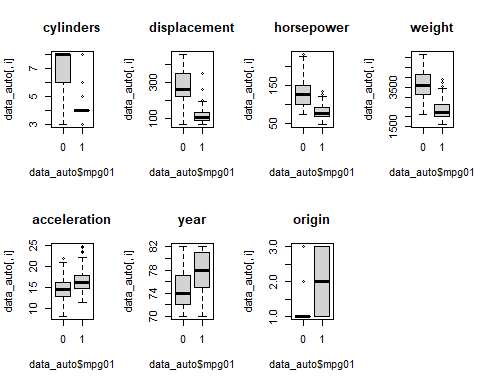
library(ISLR2)  
data(Auto)  
mpg01 <- ifelse(Auto$mpg > median(Auto$mpg), 1, 0)  
(data\_auto <- data.frame(Auto[,-1], mpg01))

## cylinders displacement horsepower weight acceleration year origin  
## 1 8 307.0 130 3504 12.0 70 1  
## 2 8 350.0 165 3693 11.5 70 1  
## 3 8 318.0 150 3436 11.0 70 1  
## 4 8 304.0 150 3433 12.0 70 1  
## 5 8 302.0 140 3449 10.5 70 1  
## 6 8 429.0 198 4341 10.0 70 1  
## 7 8 454.0 220 4354 9.0 70 1  
## 8 8 440.0 215 4312 8.5 70 1  
## 9 8 455.0 225 4425 10.0 70 1  
## 10 8 390.0 190 3850 8.5 70 1  
## 11 8 383.0 170 3563 10.0 70 1  
## 12 8 340.0 160 3609 8.0 70 1  
## 13 8 400.0 150 3761 9.5 70 1  
## 14 8 455.0 225 3086 10.0 70 1  
## 15 4 113.0 95 2372 15.0 70 3  
## 16 6 198.0 95 2833 15.5 70 1  
## 17 6 199.0 97 2774 15.5 70 1  
## 18 6 200.0 85 2587 16.0 70 1  
## 19 4 97.0 88 2130 14.5 70 3  
## 20 4 97.0 46 1835 20.5 70 2  
## 21 4 110.0 87 2672 17.5 70 2  
## 22 4 107.0 90 2430 14.5 70 2  
## 23 4 104.0 95 2375 17.5 70 2  
## 24 4 121.0 113 2234 12.5 70 2  
## 25 6 199.0 90 2648 15.0 70 1  
## 26 8 360.0 215 4615 14.0 70 1  
## 27 8 307.0 200 4376 15.0 70 1  
## 28 8 318.0 210 4382 13.5 70 1  
## 29 8 304.0 193 4732 18.5 70 1  
## 30 4 97.0 88 2130 14.5 71 3  
## 31 4 140.0 90 2264 15.5 71 1  
## 32 4 113.0 95 2228 14.0 71 3  
## 34 6 232.0 100 2634 13.0 71 1  
## 35 6 225.0 105 3439 15.5 71 1  
## 36 6 250.0 100 3329 15.5 71 1  
## 37 6 250.0 88 3302 15.5 71 1  
## 38 6 232.0 100 3288 15.5 71 1  
## 39 8 350.0 165 4209 12.0 71 1  
## 40 8 400.0 175 4464 11.5 71 1  
## 41 8 351.0 153 4154 13.5 71 1  
## 42 8 318.0 150 4096 13.0 71 1  
## 43 8 383.0 180 4955 11.5 71 1  
## 44 8 400.0 170 4746 12.0 71 1  
## 45 8 400.0 175 5140 12.0 71 1  
## 46 6 258.0 110 2962 13.5 71 1  
## 47 4 140.0 72 2408 19.0 71 1  
## 48 6 250.0 100 3282 15.0 71 1  
## 49 6 250.0 88 3139 14.5 71 1  
## 50 4 122.0 86 2220 14.0 71 1  
## 51 4 116.0 90 2123 14.0 71 2  
## 52 4 79.0 70 2074 19.5 71 2  
## 53 4 88.0 76 2065 14.5 71 2  
## 54 4 71.0 65 1773 19.0 71 3  
## 55 4 72.0 69 1613 18.0 71 3  
## 56 4 97.0 60 1834 19.0 71 2  
## 57 4 91.0 70 1955 20.5 71 1  
## 58 4 113.0 95 2278 15.5 72 3  
## 59 4 97.5 80 2126 17.0 72 1  
## 60 4 97.0 54 2254 23.5 72 2  
## 61 4 140.0 90 2408 19.5 72 1  
## 62 4 122.0 86 2226 16.5 72 1  
## 63 8 350.0 165 4274 12.0 72 1  
## 64 8 400.0 175 4385 12.0 72 1  
## 65 8 318.0 150 4135 13.5 72 1  
## 66 8 351.0 153 4129 13.0 72 1  
## 67 8 304.0 150 3672 11.5 72 1  
## 68 8 429.0 208 4633 11.0 72 1  
## 69 8 350.0 155 4502 13.5 72 1  
## 70 8 350.0 160 4456 13.5 72 1  
## 71 8 400.0 190 4422 12.5 72 1  
## 72 3 70.0 97 2330 13.5 72 3  
## 73 8 304.0 150 3892 12.5 72 1  
## 74 8 307.0 130 4098 14.0 72 1  
## 75 8 302.0 140 4294 16.0 72 1  
## 76 8 318.0 150 4077 14.0 72 1  
## 77 4 121.0 112 2933 14.5 72 2  
## 78 4 121.0 76 2511 18.0 72 2  
## 79 4 120.0 87 2979 19.5 72 2  
## 80 4 96.0 69 2189 18.0 72 2  
## 81 4 122.0 86 2395 16.0 72 1  
## 82 4 97.0 92 2288 17.0 72 3  
## 83 4 120.0 97 2506 14.5 72 3  
## 84 4 98.0 80 2164 15.0 72 1  
## 85 4 97.0 88 2100 16.5 72 3  
## 86 8 350.0 175 4100 13.0 73 1  
## 87 8 304.0 150 3672 11.5 73 1  
## 88 8 350.0 145 3988 13.0 73 1  
## 89 8 302.0 137 4042 14.5 73 1  
## 90 8 318.0 150 3777 12.5 73 1  
## 91 8 429.0 198 4952 11.5 73 1  
## 92 8 400.0 150 4464 12.0 73 1  
## 93 8 351.0 158 4363 13.0 73 1  
## 94 8 318.0 150 4237 14.5 73 1  
## 95 8 440.0 215 4735 11.0 73 1  
## 96 8 455.0 225 4951 11.0 73 1  
## 97 8 360.0 175 3821 11.0 73 1  
## 98 6 225.0 105 3121 16.5 73 1  
## 99 6 250.0 100 3278 18.0 73 1  
## 100 6 232.0 100 2945 16.0 73 1  
## 101 6 250.0 88 3021 16.5 73 1  
## 102 6 198.0 95 2904 16.0 73 1  
## 103 4 97.0 46 1950 21.0 73 2  
## 104 8 400.0 150 4997 14.0 73 1  
## 105 8 400.0 167 4906 12.5 73 1  
## 106 8 360.0 170 4654 13.0 73 1  
## 107 8 350.0 180 4499 12.5 73 1  
## 108 6 232.0 100 2789 15.0 73 1  
## 109 4 97.0 88 2279 19.0 73 3  
## 110 4 140.0 72 2401 19.5 73 1  
## 111 4 108.0 94 2379 16.5 73 3  
## 112 3 70.0 90 2124 13.5 73 3  
## 113 4 122.0 85 2310 18.5 73 1  
## 114 6 155.0 107 2472 14.0 73 1  
## 115 4 98.0 90 2265 15.5 73 2  
## 116 8 350.0 145 4082 13.0 73 1  
## 117 8 400.0 230 4278 9.5 73 1  
## 118 4 68.0 49 1867 19.5 73 2  
## 119 4 116.0 75 2158 15.5 73 2  
## 120 4 114.0 91 2582 14.0 73 2  
## 121 4 121.0 112 2868 15.5 73 2  
## 122 8 318.0 150 3399 11.0 73 1  
## 123 4 121.0 110 2660 14.0 73 2  
## 124 6 156.0 122 2807 13.5 73 3  
## 125 8 350.0 180 3664 11.0 73 1  
## 126 6 198.0 95 3102 16.5 74 1  
## 128 6 232.0 100 2901 16.0 74 1  
## 129 6 250.0 100 3336 17.0 74 1  
## 130 4 79.0 67 1950 19.0 74 3  
## 131 4 122.0 80 2451 16.5 74 1  
## 132 4 71.0 65 1836 21.0 74 3  
## 133 4 140.0 75 2542 17.0 74 1  
## 134 6 250.0 100 3781 17.0 74 1  
## 135 6 258.0 110 3632 18.0 74 1  
## 136 6 225.0 105 3613 16.5 74 1  
## 137 8 302.0 140 4141 14.0 74 1  
## 138 8 350.0 150 4699 14.5 74 1  
## 139 8 318.0 150 4457 13.5 74 1  
## 140 8 302.0 140 4638 16.0 74 1  
## 141 8 304.0 150 4257 15.5 74 1  
## 142 4 98.0 83 2219 16.5 74 2  
## 143 4 79.0 67 1963 15.5 74 2  
## 144 4 97.0 78 2300 14.5 74 2  
## 145 4 76.0 52 1649 16.5 74 3  
## 146 4 83.0 61 2003 19.0 74 3  
## 147 4 90.0 75 2125 14.5 74 1  
## 148 4 90.0 75 2108 15.5 74 2  
## 149 4 116.0 75 2246 14.0 74 2  
## 150 4 120.0 97 2489 15.0 74 3  
## 151 4 108.0 93 2391 15.5 74 3  
## 152 4 79.0 67 2000 16.0 74 2  
## 153 6 225.0 95 3264 16.0 75 1  
## 154 6 250.0 105 3459 16.0 75 1  
## 155 6 250.0 72 3432 21.0 75 1  
## 156 6 250.0 72 3158 19.5 75 1  
## 157 8 400.0 170 4668 11.5 75 1  
## 158 8 350.0 145 4440 14.0 75 1  
## 159 8 318.0 150 4498 14.5 75 1  
## 160 8 351.0 148 4657 13.5 75 1  
## 161 6 231.0 110 3907 21.0 75 1  
## 162 6 250.0 105 3897 18.5 75 1  
## 163 6 258.0 110 3730 19.0 75 1  
## 164 6 225.0 95 3785 19.0 75 1  
## 165 6 231.0 110 3039 15.0 75 1  
## 166 8 262.0 110 3221 13.5 75 1  
## 167 8 302.0 129 3169 12.0 75 1  
## 168 4 97.0 75 2171 16.0 75 3  
## 169 4 140.0 83 2639 17.0 75 1  
## 170 6 232.0 100 2914 16.0 75 1  
## 171 4 140.0 78 2592 18.5 75 1  
## 172 4 134.0 96 2702 13.5 75 3  
## 173 4 90.0 71 2223 16.5 75 2  
## 174 4 119.0 97 2545 17.0 75 3  
## 175 6 171.0 97 2984 14.5 75 1  
## 176 4 90.0 70 1937 14.0 75 2  
## 177 6 232.0 90 3211 17.0 75 1  
## 178 4 115.0 95 2694 15.0 75 2  
## 179 4 120.0 88 2957 17.0 75 2  
## 180 4 121.0 98 2945 14.5 75 2  
## 181 4 121.0 115 2671 13.5 75 2  
## 182 4 91.0 53 1795 17.5 75 3  
## 183 4 107.0 86 2464 15.5 76 2  
## 184 4 116.0 81 2220 16.9 76 2  
## 185 4 140.0 92 2572 14.9 76 1  
## 186 4 98.0 79 2255 17.7 76 1  
## 187 4 101.0 83 2202 15.3 76 2  
## 188 8 305.0 140 4215 13.0 76 1  
## 189 8 318.0 150 4190 13.0 76 1  
## 190 8 304.0 120 3962 13.9 76 1  
## 191 8 351.0 152 4215 12.8 76 1  
## 192 6 225.0 100 3233 15.4 76 1  
## 193 6 250.0 105 3353 14.5 76 1  
## 194 6 200.0 81 3012 17.6 76 1  
## 195 6 232.0 90 3085 17.6 76 1  
## 196 4 85.0 52 2035 22.2 76 1  
## 197 4 98.0 60 2164 22.1 76 1  
## 198 4 90.0 70 1937 14.2 76 2  
## 199 4 91.0 53 1795 17.4 76 3  
## 200 6 225.0 100 3651 17.7 76 1  
## 201 6 250.0 78 3574 21.0 76 1  
## 202 6 250.0 110 3645 16.2 76 1  
## 203 6 258.0 95 3193 17.8 76 1  
## 204 4 97.0 71 1825 12.2 76 2  
## 205 4 85.0 70 1990 17.0 76 3  
## 206 4 97.0 75 2155 16.4 76 3  
## 207 4 140.0 72 2565 13.6 76 1  
## 208 4 130.0 102 3150 15.7 76 2  
## 209 8 318.0 150 3940 13.2 76 1  
## 210 4 120.0 88 3270 21.9 76 2  
## 211 6 156.0 108 2930 15.5 76 3  
## 212 6 168.0 120 3820 16.7 76 2  
## 213 8 350.0 180 4380 12.1 76 1  
## 214 8 350.0 145 4055 12.0 76 1  
## 215 8 302.0 130 3870 15.0 76 1  
## 216 8 318.0 150 3755 14.0 76 1  
## 217 4 98.0 68 2045 18.5 77 3  
## 218 4 111.0 80 2155 14.8 77 1  
## 219 4 79.0 58 1825 18.6 77 2  
## 220 4 122.0 96 2300 15.5 77 1  
## 221 4 85.0 70 1945 16.8 77 3  
## 222 8 305.0 145 3880 12.5 77 1  
## 223 8 260.0 110 4060 19.0 77 1  
## 224 8 318.0 145 4140 13.7 77 1  
## 225 8 302.0 130 4295 14.9 77 1  
## 226 6 250.0 110 3520 16.4 77 1  
## 227 6 231.0 105 3425 16.9 77 1  
## 228 6 225.0 100 3630 17.7 77 1  
## 229 6 250.0 98 3525 19.0 77 1  
## 230 8 400.0 180 4220 11.1 77 1  
## 231 8 350.0 170 4165 11.4 77 1  
## 232 8 400.0 190 4325 12.2 77 1  
## 233 8 351.0 149 4335 14.5 77 1  
## 234 4 97.0 78 1940 14.5 77 2  
## 235 4 151.0 88 2740 16.0 77 1  
## 236 4 97.0 75 2265 18.2 77 3  
## 237 4 140.0 89 2755 15.8 77 1  
## 238 4 98.0 63 2051 17.0 77 1  
## 239 4 98.0 83 2075 15.9 77 1  
## 240 4 97.0 67 1985 16.4 77 3  
## 241 4 97.0 78 2190 14.1 77 2  
## 242 6 146.0 97 2815 14.5 77 3  
## 243 4 121.0 110 2600 12.8 77 2  
## 244 3 80.0 110 2720 13.5 77 3  
## 245 4 90.0 48 1985 21.5 78 2  
## 246 4 98.0 66 1800 14.4 78 1  
## 247 4 78.0 52 1985 19.4 78 3  
## 248 4 85.0 70 2070 18.6 78 3  
## 249 4 91.0 60 1800 16.4 78 3  
## 250 8 260.0 110 3365 15.5 78 1  
## 251 8 318.0 140 3735 13.2 78 1  
## 252 8 302.0 139 3570 12.8 78 1  
## 253 6 231.0 105 3535 19.2 78 1  
## 254 6 200.0 95 3155 18.2 78 1  
## 255 6 200.0 85 2965 15.8 78 1  
## 256 4 140.0 88 2720 15.4 78 1  
## 257 6 225.0 100 3430 17.2 78 1  
## 258 6 232.0 90 3210 17.2 78 1  
## 259 6 231.0 105 3380 15.8 78 1  
## 260 6 200.0 85 3070 16.7 78 1  
## 261 6 225.0 110 3620 18.7 78 1  
## 262 6 258.0 120 3410 15.1 78 1  
## 263 8 305.0 145 3425 13.2 78 1  
## 264 6 231.0 165 3445 13.4 78 1  
## 265 8 302.0 139 3205 11.2 78 1  
## 266 8 318.0 140 4080 13.7 78 1  
## 267 4 98.0 68 2155 16.5 78 1  
## 268 4 134.0 95 2560 14.2 78 3  
## 269 4 119.0 97 2300 14.7 78 3  
## 270 4 105.0 75 2230 14.5 78 1  
## 271 4 134.0 95 2515 14.8 78 3  
## 272 4 156.0 105 2745 16.7 78 1  
## 273 4 151.0 85 2855 17.6 78 1  
## 274 4 119.0 97 2405 14.9 78 3  
## 275 5 131.0 103 2830 15.9 78 2  
## 276 6 163.0 125 3140 13.6 78 2  
## 277 4 121.0 115 2795 15.7 78 2  
## 278 6 163.0 133 3410 15.8 78 2  
## 279 4 89.0 71 1990 14.9 78 2  
## 280 4 98.0 68 2135 16.6 78 3  
## 281 6 231.0 115 3245 15.4 79 1  
## 282 6 200.0 85 2990 18.2 79 1  
## 283 4 140.0 88 2890 17.3 79 1  
## 284 6 232.0 90 3265 18.2 79 1  
## 285 6 225.0 110 3360 16.6 79 1  
## 286 8 305.0 130 3840 15.4 79 1  
## 287 8 302.0 129 3725 13.4 79 1  
## 288 8 351.0 138 3955 13.2 79 1  
## 289 8 318.0 135 3830 15.2 79 1  
## 290 8 350.0 155 4360 14.9 79 1  
## 291 8 351.0 142 4054 14.3 79 1  
## 292 8 267.0 125 3605 15.0 79 1  
## 293 8 360.0 150 3940 13.0 79 1  
## 294 4 89.0 71 1925 14.0 79 2  
## 295 4 86.0 65 1975 15.2 79 3  
## 296 4 98.0 80 1915 14.4 79 1  
## 297 4 121.0 80 2670 15.0 79 1  
## 298 5 183.0 77 3530 20.1 79 2  
## 299 8 350.0 125 3900 17.4 79 1  
## 300 4 141.0 71 3190 24.8 79 2  
## 301 8 260.0 90 3420 22.2 79 1  
## 302 4 105.0 70 2200 13.2 79 1  
## 303 4 105.0 70 2150 14.9 79 1  
## 304 4 85.0 65 2020 19.2 79 3  
## 305 4 91.0 69 2130 14.7 79 2  
## 306 4 151.0 90 2670 16.0 79 1  
## 307 6 173.0 115 2595 11.3 79 1  
## 308 6 173.0 115 2700 12.9 79 1  
## 309 4 151.0 90 2556 13.2 79 1  
## 310 4 98.0 76 2144 14.7 80 2  
## 311 4 89.0 60 1968 18.8 80 3  
## 312 4 98.0 70 2120 15.5 80 1  
## 313 4 86.0 65 2019 16.4 80 3  
## 314 4 151.0 90 2678 16.5 80 1  
## 315 4 140.0 88 2870 18.1 80 1  
## 316 4 151.0 90 3003 20.1 80 1  
## 317 6 225.0 90 3381 18.7 80 1  
## 318 4 97.0 78 2188 15.8 80 2  
## 319 4 134.0 90 2711 15.5 80 3  
## 320 4 120.0 75 2542 17.5 80 3  
## 321 4 119.0 92 2434 15.0 80 3  
## 322 4 108.0 75 2265 15.2 80 3  
## 323 4 86.0 65 2110 17.9 80 3  
## 324 4 156.0 105 2800 14.4 80 1  
## 325 4 85.0 65 2110 19.2 80 3  
## 326 4 90.0 48 2085 21.7 80 2  
## 327 4 90.0 48 2335 23.7 80 2  
## 328 5 121.0 67 2950 19.9 80 2  
## 329 4 146.0 67 3250 21.8 80 2  
## 330 4 91.0 67 1850 13.8 80 3  
## 332 4 97.0 67 2145 18.0 80 3  
## 333 4 89.0 62 1845 15.3 80 2  
## 334 6 168.0 132 2910 11.4 80 3  
## 335 3 70.0 100 2420 12.5 80 3  
## 336 4 122.0 88 2500 15.1 80 2  
## 338 4 107.0 72 2290 17.0 80 3  
## 339 4 135.0 84 2490 15.7 81 1  
## 340 4 151.0 84 2635 16.4 81 1  
## 341 4 156.0 92 2620 14.4 81 1  
## 342 6 173.0 110 2725 12.6 81 1  
## 343 4 135.0 84 2385 12.9 81 1  
## 344 4 79.0 58 1755 16.9 81 3  
## 345 4 86.0 64 1875 16.4 81 1  
## 346 4 81.0 60 1760 16.1 81 3  
## 347 4 97.0 67 2065 17.8 81 3  
## 348 4 85.0 65 1975 19.4 81 3  
## 349 4 89.0 62 2050 17.3 81 3  
## 350 4 91.0 68 1985 16.0 81 3  
## 351 4 105.0 63 2215 14.9 81 1  
## 352 4 98.0 65 2045 16.2 81 1  
## 353 4 98.0 65 2380 20.7 81 1  
## 354 4 105.0 74 2190 14.2 81 2  
## 356 4 107.0 75 2210 14.4 81 3  
## 357 4 108.0 75 2350 16.8 81 3  
## 358 4 119.0 100 2615 14.8 81 3  
## 359 4 120.0 74 2635 18.3 81 3  
## 360 4 141.0 80 3230 20.4 81 2  
## 361 6 145.0 76 3160 19.6 81 2  
## 362 6 168.0 116 2900 12.6 81 3  
## 363 6 146.0 120 2930 13.8 81 3  
## 364 6 231.0 110 3415 15.8 81 1  
## 365 8 350.0 105 3725 19.0 81 1  
## 366 6 200.0 88 3060 17.1 81 1  
## 367 6 225.0 85 3465 16.6 81 1  
## 368 4 112.0 88 2605 19.6 82 1  
## 369 4 112.0 88 2640 18.6 82 1  
## 370 4 112.0 88 2395 18.0 82 1  
## 371 4 112.0 85 2575 16.2 82 1  
## 372 4 135.0 84 2525 16.0 82 1  
## 373 4 151.0 90 2735 18.0 82 1  
## 374 4 140.0 92 2865 16.4 82 1  
## 375 4 105.0 74 1980 15.3 82 2  
## 376 4 91.0 68 2025 18.2 82 3  
## 377 4 91.0 68 1970 17.6 82 3  
## 378 4 105.0 63 2125 14.7 82 1  
## 379 4 98.0 70 2125 17.3 82 1  
## 380 4 120.0 88 2160 14.5 82 3  
## 381 4 107.0 75 2205 14.5 82 3  
## 382 4 108.0 70 2245 16.9 82 3  
## 383 4 91.0 67 1965 15.0 82 3  
## 384 4 91.0 67 1965 15.7 82 3  
## 385 4 91.0 67 1995 16.2 82 3  
## 386 6 181.0 110 2945 16.4 82 1  
## 387 6 262.0 85 3015 17.0 82 1  
## 388 4 156.0 92 2585 14.5 82 1  
## 389 6 232.0 112 2835 14.7 82 1  
## 390 4 144.0 96 2665 13.9 82 3  
## 391 4 135.0 84 2370 13.0 82 1  
## 392 4 151.0 90 2950 17.3 82 1  
## 393 4 140.0 86 2790 15.6 82 1  
## 394 4 97.0 52 2130 24.6 82 2  
## 395 4 135.0 84 2295 11.6 82 1  
## 396 4 120.0 79 2625 18.6 82 1  
## 397 4 119.0 82 2720 19.4 82 1  
## name mpg01  
## 1 chevrolet chevelle malibu 0  
## 2 buick skylark 320 0  
## 3 plymouth satellite 0  
## 4 amc rebel sst 0  
## 5 ford torino 0  
## 6 ford galaxie 500 0  
## 7 chevrolet impala 0  
## 8 plymouth fury iii 0  
## 9 pontiac catalina 0  
## 10 amc ambassador dpl 0  
## 11 dodge challenger se 0  
## 12 plymouth 'cuda 340 0  
## 13 chevrolet monte carlo 0  
## 14 buick estate wagon (sw) 0  
## 15 toyota corona mark ii 1  
## 16 plymouth duster 0  
## 17 amc hornet 0  
## 18 ford maverick 0  
## 19 datsun pl510 1  
## 20 volkswagen 1131 deluxe sedan 1  
## 21 peugeot 504 1  
## 22 audi 100 ls 1  
## 23 saab 99e 1  
## 24 bmw 2002 1  
## 25 amc gremlin 0  
## 26 ford f250 0  
## 27 chevy c20 0  
## 28 dodge d200 0  
## 29 hi 1200d 0  
## 30 datsun pl510 1  
## 31 chevrolet vega 2300 1  
## 32 toyota corona 1  
## 34 amc gremlin 0  
## 35 plymouth satellite custom 0  
## 36 chevrolet chevelle malibu 0  
## 37 ford torino 500 0  
## 38 amc matador 0  
## 39 chevrolet impala 0  
## 40 pontiac catalina brougham 0  
## 41 ford galaxie 500 0  
## 42 plymouth fury iii 0  
## 43 dodge monaco (sw) 0  
## 44 ford country squire (sw) 0  
## 45 pontiac safari (sw) 0  
## 46 amc hornet sportabout (sw) 0  
## 47 chevrolet vega (sw) 0  
## 48 pontiac firebird 0  
## 49 ford mustang 0  
## 50 mercury capri 2000 1  
## 51 opel 1900 1  
## 52 peugeot 304 1  
## 53 fiat 124b 1  
## 54 toyota corolla 1200 1  
## 55 datsun 1200 1  
## 56 volkswagen model 111 1  
## 57 plymouth cricket 1  
## 58 toyota corona hardtop 1  
## 59 dodge colt hardtop 1  
## 60 volkswagen type 3 1  
## 61 chevrolet vega 0  
## 62 ford pinto runabout 0  
## 63 chevrolet impala 0  
## 64 pontiac catalina 0  
## 65 plymouth fury iii 0  
## 66 ford galaxie 500 0  
## 67 amc ambassador sst 0  
## 68 mercury marquis 0  
## 69 buick lesabre custom 0  
## 70 oldsmobile delta 88 royale 0  
## 71 chrysler newport royal 0  
## 72 mazda rx2 coupe 0  
## 73 amc matador (sw) 0  
## 74 chevrolet chevelle concours (sw) 0  
## 75 ford gran torino (sw) 0  
## 76 plymouth satellite custom (sw) 0  
## 77 volvo 145e (sw) 0  
## 78 volkswagen 411 (sw) 0  
## 79 peugeot 504 (sw) 0  
## 80 renault 12 (sw) 1  
## 81 ford pinto (sw) 0  
## 82 datsun 510 (sw) 1  
## 83 toyouta corona mark ii (sw) 1  
## 84 dodge colt (sw) 1  
## 85 toyota corolla 1600 (sw) 1  
## 86 buick century 350 0  
## 87 amc matador 0  
## 88 chevrolet malibu 0  
## 89 ford gran torino 0  
## 90 dodge coronet custom 0  
## 91 mercury marquis brougham 0  
## 92 chevrolet caprice classic 0  
## 93 ford ltd 0  
## 94 plymouth fury gran sedan 0  
## 95 chrysler new yorker brougham 0  
## 96 buick electra 225 custom 0  
## 97 amc ambassador brougham 0  
## 98 plymouth valiant 0  
## 99 chevrolet nova custom 0  
## 100 amc hornet 0  
## 101 ford maverick 0  
## 102 plymouth duster 1  
## 103 volkswagen super beetle 1  
## 104 chevrolet impala 0  
## 105 ford country 0  
## 106 plymouth custom suburb 0  
## 107 oldsmobile vista cruiser 0  
## 108 amc gremlin 0  
## 109 toyota carina 0  
## 110 chevrolet vega 0  
## 111 datsun 610 0  
## 112 maxda rx3 0  
## 113 ford pinto 0  
## 114 mercury capri v6 0  
## 115 fiat 124 sport coupe 1  
## 116 chevrolet monte carlo s 0  
## 117 pontiac grand prix 0  
## 118 fiat 128 1  
## 119 opel manta 1  
## 120 audi 100ls 0  
## 121 volvo 144ea 0  
## 122 dodge dart custom 0  
## 123 saab 99le 1  
## 124 toyota mark ii 0  
## 125 oldsmobile omega 0  
## 126 plymouth duster 0  
## 128 amc hornet 0  
## 129 chevrolet nova 0  
## 130 datsun b210 1  
## 131 ford pinto 1  
## 132 toyota corolla 1200 1  
## 133 chevrolet vega 1  
## 134 chevrolet chevelle malibu classic 0  
## 135 amc matador 0  
## 136 plymouth satellite sebring 0  
## 137 ford gran torino 0  
## 138 buick century luxus (sw) 0  
## 139 dodge coronet custom (sw) 0  
## 140 ford gran torino (sw) 0  
## 141 amc matador (sw) 0  
## 142 audi fox 1  
## 143 volkswagen dasher 1  
## 144 opel manta 1  
## 145 toyota corona 1  
## 146 datsun 710 1  
## 147 dodge colt 1  
## 148 fiat 128 1  
## 149 fiat 124 tc 1  
## 150 honda civic 1  
## 151 subaru 1  
## 152 fiat x1.9 1  
## 153 plymouth valiant custom 0  
## 154 chevrolet nova 0  
## 155 mercury monarch 0  
## 156 ford maverick 0  
## 157 pontiac catalina 0  
## 158 chevrolet bel air 0  
## 159 plymouth grand fury 0  
## 160 ford ltd 0  
## 161 buick century 0  
## 162 chevroelt chevelle malibu 0  
## 163 amc matador 0  
## 164 plymouth fury 0  
## 165 buick skyhawk 0  
## 166 chevrolet monza 2+2 0  
## 167 ford mustang ii 0  
## 168 toyota corolla 1  
## 169 ford pinto 1  
## 170 amc gremlin 0  
## 171 pontiac astro 1  
## 172 toyota corona 1  
## 173 volkswagen dasher 1  
## 174 datsun 710 1  
## 175 ford pinto 0  
## 176 volkswagen rabbit 1  
## 177 amc pacer 0  
## 178 audi 100ls 1  
## 179 peugeot 504 1  
## 180 volvo 244dl 0  
## 181 saab 99le 1  
## 182 honda civic cvcc 1  
## 183 fiat 131 1  
## 184 opel 1900 1  
## 185 capri ii 1  
## 186 dodge colt 1  
## 187 renault 12tl 1  
## 188 chevrolet chevelle malibu classic 0  
## 189 dodge coronet brougham 0  
## 190 amc matador 0  
## 191 ford gran torino 0  
## 192 plymouth valiant 0  
## 193 chevrolet nova 0  
## 194 ford maverick 1  
## 195 amc hornet 0  
## 196 chevrolet chevette 1  
## 197 chevrolet woody 1  
## 198 vw rabbit 1  
## 199 honda civic 1  
## 200 dodge aspen se 0  
## 201 ford granada ghia 0  
## 202 pontiac ventura sj 0  
## 203 amc pacer d/l 0  
## 204 volkswagen rabbit 1  
## 205 datsun b-210 1  
## 206 toyota corolla 1  
## 207 ford pinto 1  
## 208 volvo 245 0  
## 209 plymouth volare premier v8 0  
## 210 peugeot 504 0  
## 211 toyota mark ii 0  
## 212 mercedes-benz 280s 0  
## 213 cadillac seville 0  
## 214 chevy c10 0  
## 215 ford f108 0  
## 216 dodge d100 0  
## 217 honda accord cvcc 1  
## 218 buick opel isuzu deluxe 1  
## 219 renault 5 gtl 1  
## 220 plymouth arrow gs 1  
## 221 datsun f-10 hatchback 1  
## 222 chevrolet caprice classic 0  
## 223 oldsmobile cutlass supreme 0  
## 224 dodge monaco brougham 0  
## 225 mercury cougar brougham 0  
## 226 chevrolet concours 0  
## 227 buick skylark 0  
## 228 plymouth volare custom 0  
## 229 ford granada 0  
## 230 pontiac grand prix lj 0  
## 231 chevrolet monte carlo landau 0  
## 232 chrysler cordoba 0  
## 233 ford thunderbird 0  
## 234 volkswagen rabbit custom 1  
## 235 pontiac sunbird coupe 1  
## 236 toyota corolla liftback 1  
## 237 ford mustang ii 2+2 1  
## 238 chevrolet chevette 1  
## 239 dodge colt m/m 1  
## 240 subaru dl 1  
## 241 volkswagen dasher 1  
## 242 datsun 810 0  
## 243 bmw 320i 0  
## 244 mazda rx-4 0  
## 245 volkswagen rabbit custom diesel 1  
## 246 ford fiesta 1  
## 247 mazda glc deluxe 1  
## 248 datsun b210 gx 1  
## 249 honda civic cvcc 1  
## 250 oldsmobile cutlass salon brougham 0  
## 251 dodge diplomat 0  
## 252 mercury monarch ghia 0  
## 253 pontiac phoenix lj 0  
## 254 chevrolet malibu 0  
## 255 ford fairmont (auto) 0  
## 256 ford fairmont (man) 1  
## 257 plymouth volare 0  
## 258 amc concord 0  
## 259 buick century special 0  
## 260 mercury zephyr 0  
## 261 dodge aspen 0  
## 262 amc concord d/l 0  
## 263 chevrolet monte carlo landau 0  
## 264 buick regal sport coupe (turbo) 0  
## 265 ford futura 0  
## 266 dodge magnum xe 0  
## 267 chevrolet chevette 1  
## 268 toyota corona 1  
## 269 datsun 510 1  
## 270 dodge omni 1  
## 271 toyota celica gt liftback 0  
## 272 plymouth sapporo 1  
## 273 oldsmobile starfire sx 1  
## 274 datsun 200-sx 1  
## 275 audi 5000 0  
## 276 volvo 264gl 0  
## 277 saab 99gle 0  
## 278 peugeot 604sl 0  
## 279 volkswagen scirocco 1  
## 280 honda accord lx 1  
## 281 pontiac lemans v6 0  
## 282 mercury zephyr 6 0  
## 283 ford fairmont 4 0  
## 284 amc concord dl 6 0  
## 285 dodge aspen 6 0  
## 286 chevrolet caprice classic 0  
## 287 ford ltd landau 0  
## 288 mercury grand marquis 0  
## 289 dodge st. regis 0  
## 290 buick estate wagon (sw) 0  
## 291 ford country squire (sw) 0  
## 292 chevrolet malibu classic (sw) 0  
## 293 chrysler lebaron town @ country (sw) 0  
## 294 vw rabbit custom 1  
## 295 maxda glc deluxe 1  
## 296 dodge colt hatchback custom 1  
## 297 amc spirit dl 1  
## 298 mercedes benz 300d 1  
## 299 cadillac eldorado 1  
## 300 peugeot 504 1  
## 301 oldsmobile cutlass salon brougham 1  
## 302 plymouth horizon 1  
## 303 plymouth horizon tc3 1  
## 304 datsun 210 1  
## 305 fiat strada custom 1  
## 306 buick skylark limited 1  
## 307 chevrolet citation 1  
## 308 oldsmobile omega brougham 1  
## 309 pontiac phoenix 1  
## 310 vw rabbit 1  
## 311 toyota corolla tercel 1  
## 312 chevrolet chevette 1  
## 313 datsun 310 1  
## 314 chevrolet citation 1  
## 315 ford fairmont 1  
## 316 amc concord 1  
## 317 dodge aspen 0  
## 318 audi 4000 1  
## 319 toyota corona liftback 1  
## 320 mazda 626 1  
## 321 datsun 510 hatchback 1  
## 322 toyota corolla 1  
## 323 mazda glc 1  
## 324 dodge colt 1  
## 325 datsun 210 1  
## 326 vw rabbit c (diesel) 1  
## 327 vw dasher (diesel) 1  
## 328 audi 5000s (diesel) 1  
## 329 mercedes-benz 240d 1  
## 330 honda civic 1500 gl 1  
## 332 subaru dl 1  
## 333 vokswagen rabbit 1  
## 334 datsun 280-zx 1  
## 335 mazda rx-7 gs 1  
## 336 triumph tr7 coupe 1  
## 338 honda accord 1  
## 339 plymouth reliant 1  
## 340 buick skylark 1  
## 341 dodge aries wagon (sw) 1  
## 342 chevrolet citation 1  
## 343 plymouth reliant 1  
## 344 toyota starlet 1  
## 345 plymouth champ 1  
## 346 honda civic 1300 1  
## 347 subaru 1  
## 348 datsun 210 mpg 1  
## 349 toyota tercel 1  
## 350 mazda glc 4 1  
## 351 plymouth horizon 4 1  
## 352 ford escort 4w 1  
## 353 ford escort 2h 1  
## 354 volkswagen jetta 1  
## 356 honda prelude 1  
## 357 toyota corolla 1  
## 358 datsun 200sx 1  
## 359 mazda 626 1  
## 360 peugeot 505s turbo diesel 1  
## 361 volvo diesel 1  
## 362 toyota cressida 1  
## 363 datsun 810 maxima 1  
## 364 buick century 0  
## 365 oldsmobile cutlass ls 1  
## 366 ford granada gl 0  
## 367 chrysler lebaron salon 0  
## 368 chevrolet cavalier 1  
## 369 chevrolet cavalier wagon 1  
## 370 chevrolet cavalier 2-door 1  
## 371 pontiac j2000 se hatchback 1  
## 372 dodge aries se 1  
## 373 pontiac phoenix 1  
## 374 ford fairmont futura 1  
## 375 volkswagen rabbit l 1  
## 376 mazda glc custom l 1  
## 377 mazda glc custom 1  
## 378 plymouth horizon miser 1  
## 379 mercury lynx l 1  
## 380 nissan stanza xe 1  
## 381 honda accord 1  
## 382 toyota corolla 1  
## 383 honda civic 1  
## 384 honda civic (auto) 1  
## 385 datsun 310 gx 1  
## 386 buick century limited 1  
## 387 oldsmobile cutlass ciera (diesel) 1  
## 388 chrysler lebaron medallion 1  
## 389 ford granada l 0  
## 390 toyota celica gt 1  
## 391 dodge charger 2.2 1  
## 392 chevrolet camaro 1  
## 393 ford mustang gl 1  
## 394 vw pickup 1  
## 395 dodge rampage 1  
## 396 ford ranger 1  
## 397 chevy s-10 1

#(data\_auto <- data.frame(Auto, mpg01))

1. Explore the data graphically in order to investigate the association between mpg01 and the other features. Which of the other features seem most likely to be useful in predicting mpg01? Scatterplots and boxplots may be useful tools to answer this question. Describe your findings.

par(mfrow = c(2, 4))  
for (i in 1:7) boxplot(data\_auto[, i] ~ data\_auto$mpg01, main = colnames(data\_auto)[i])

1. Cylinders : Mobil dengan sedikit silinder (misalnya 4) cenderung hemat BBM.

2. Displacement : Mobil dengan displacement (kapasitas mesin) lebih kecil cenderung hemat BBM.

3. Horsepower : Mobil dengan horsepower lebih rendah cenderung hemat BBM.

4. Weight : Mobil yang lebih ringan cenderung lebih hemat BBM.

5. Acceleration : Tidak ada perbedaan yang cukup signifikan.

6. Year : Mobil yang lebih baru (year tinggi) tidak selalu lebih hemat BBM.

7. Origin : Mobil yang berasal dari tempat 1,2,dan 3 banyak yang hemat BBM, tapi ada juga yang tidak, hubungan tidak cukup signifikan.

Kesimpulan:

Cylinders, Displacement, Horsepower, dan Weight bisa dijadikan prediktor yang kuat. Sementara Acceleration, Year, dan Originkurang memiliki hubungan yang signifikan dengan mpg.

1. Split the data into a training set and a test set.

set.seed(1)  
train\_index <- sample(1:nrow(data\_auto), nrow(data\_auto) \* 2/3)  
train\_data <- data\_auto[train\_index, ]  
test\_data <- data\_auto[-train\_index, ]

1. Perform KNN on the training data, with several values of , in order to predict mpg01. Use only the variables that seemed most associated with mpg01 in (b). What test errors do you obtain? Which value of seems to perform the best on this data set?

library(class)  
train\_X <- scale(train\_data[, c("cylinders", "horsepower", "weight", "displacement")])  
test\_X <- scale(test\_data[, c("cylinders", "horsepower", "weight", "displacement")])  
train\_Y <- train\_data$mpg01  
  
errors\_knn <- c()  
for (k in 1:10) {  
 knn\_predict <- knn(train\_X, test\_X, train\_Y, k = k)  
 error <- mean(knn\_predict != test\_data$mpg01)  
 errors\_knn <- c(errors\_knn, error)  
}  
errors\_knn

## [1] 0.13740458 0.09923664 0.12977099 0.12977099 0.12213740 0.12213740  
## [7] 0.13740458 0.12977099 0.12977099 0.11450382

which.min(errors\_knn)

## [1] 2

Untuk memprediksi mpg dengan prediktor Cylinders, Displacement, Horsepower, dan Weight menggunakan model K-Nearest Neighbors (KNN) nilai K yang paling bagus untuk data ini adalah dengan error (potensi salah) sebesar .

# Nomor 16

Using the Boston data set, fit classification models in order to predict whether a given census tract has a crime rate above or below the median. Explore logistic regression, LDA, naive Bayes and KNN models using various sub-sets of the predictors. Describe your findings.

library(ISLR2)  
data(Boston)  
crime01 <- ifelse(Boston$crim > median(Boston$crim), 1, 0)  
data\_boston <- data.frame(Boston, crime01)  
  
# Membagi data  
set.seed(123)  
train\_index2 <- sample(1:nrow(data\_boston), nrow(data\_boston) \* 2/3)  
train\_data2 <- data\_boston[train\_index2, ]  
test\_data2 <- data\_boston[-train\_index2, ]  
  
# KNN  
library(class)  
train2\_X <- scale(train\_data2[, c("lstat", "dis", "nox", "rm")])  
test2\_X <- scale(test\_data2[, c("lstat", "dis", "nox", "rm")])  
train2\_Y <- train\_data2$crime01  
  
errors2\_knn <- c()  
for (k in 1:16) {  
 knn\_predict2 <- knn(train2\_X, test2\_X, train2\_Y, k = k)  
 error2 <- mean(knn\_predict2 != test\_data2$crime01)  
 errors2\_knn <- c(errors2\_knn, error2)  
}  
errors2\_knn

## [1] 0.1834320 0.1952663 0.1538462 0.1834320 0.1656805 0.1715976 0.1538462  
## [8] 0.1656805 0.1656805 0.1715976 0.1715976 0.1715976 0.1715976 0.1715976  
## [15] 0.1834320 0.1893491

which.min(errors2\_knn)

## [1] 3

Untuk model K-Nearest Neighbors (KNN), nilai K yang paling bagus untuk data ini adalah dengan error (potensi salah) sebesar .