

ECON 573 Problem Set 1 Part 2 and 3

Part 2

8a) Use the `read.csv()` function to read the data into R. Call the loaded data college. Make sure that you have the directory set to the correct location for the data.

```
college = read.csv("College.csv")
```

8b) Look at the data using the `fix()` function. You should notice that the first column is just the name of each university. We don't really want R to treat this as data. However, it may be handy to have these names for later. Try the following commands:

```
rownames(college) = college[,1]
college = fix(college)
college = college [,-1]
college = fix(college)
```

8ci) Use the `summary()` function to produce a numerical summary of the variables in the data set.

```
summary(college)
```

```
##    Private          Apps        Accept      Enroll
##  Length:777      Min.   : 81   Min.   : 72   Min.   : 35
##  Class :character 1st Qu.: 776   1st Qu.: 604   1st Qu.: 242
##  Mode  :character   Median :1558   Median :1110   Median : 434
##                  Mean   :3002   Mean   :2019   Mean   : 780
##                  3rd Qu.:3624   3rd Qu.:2424   3rd Qu.: 902
##                  Max.   :48094   Max.   :26330   Max.   :6392
##    Top10perc     Top25perc    F.Undergrad    P.Undergrad
##  Min.   : 1.00   Min.   : 9.0   Min.   : 139   Min.   : 1.0
##  1st Qu.:15.00  1st Qu.: 41.0  1st Qu.: 992   1st Qu.: 95.0
##  Median :23.00  Median : 54.0  Median :1707   Median : 353.0
##  Mean   :27.56  Mean   : 55.8  Mean   :3700   Mean   : 855.3
##  3rd Qu.:35.00  3rd Qu.: 69.0  3rd Qu.:4005   3rd Qu.: 967.0
##  Max.   :96.00  Max.   :100.0  Max.   :31643   Max.   :21836.0
##    Outstate       Room.Board      Books        Personal
##  Min.   : 2340   Min.   :1780   Min.   : 96.0   Min.   : 250
##  1st Qu.: 7320   1st Qu.:3597   1st Qu.: 470.0  1st Qu.: 850
##  Median : 9990   Median :4200   Median :500.0   Median :1200
##  Mean   :10441   Mean   :4358   Mean   :549.4   Mean   :1341
##  3rd Qu.:12925   3rd Qu.:5050   3rd Qu.:600.0   3rd Qu.:1700
##  Max.   :21700   Max.   :8124   Max.   :2340.0  Max.   :6800
##    PhD           Terminal      S.F.Ratio    perc.alumni
##  Min.   : 8.00   Min.   : 24.0  Min.   : 2.50   Min.   : 0.00
##  1st Qu.: 62.00  1st Qu.: 71.0  1st Qu.:11.50  1st Qu.:13.00
```

```

## Median : 75.00   Median : 82.0   Median :13.60   Median :21.00
## Mean   : 72.66   Mean   : 79.7   Mean   :14.09   Mean   :22.74
## 3rd Qu.: 85.00   3rd Qu.: 92.0   3rd Qu.:16.50   3rd Qu.:31.00
## Max.   :103.00   Max.   :100.0   Max.   :39.80   Max.   :64.00
##      Expend          Grad.Rate
## Min.   : 3186   Min.   : 10.00
## 1st Qu.: 6751   1st Qu.: 53.00
## Median : 8377   Median : 65.00
## Mean   : 9660   Mean   : 65.46
## 3rd Qu.:10830   3rd Qu.: 78.00
## Max.   :56233   Max.   :118.00

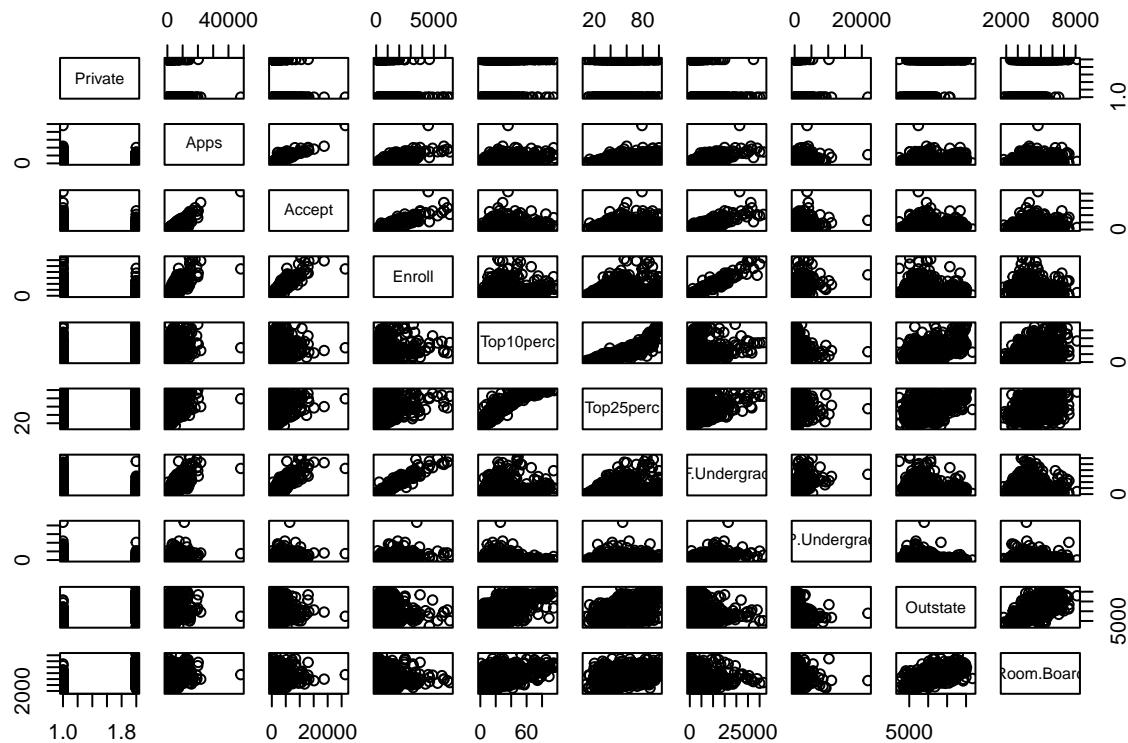
```

8cii). Use the pairs() function to produce a scatterplot matrix of the first ten columns or variables of the data. Recall that you can reference the first ten columns of a matrix A using A[,1:10].

```

college[,1] = as.numeric(factor(college[,1]))
pairs(college[, 1:10])

```

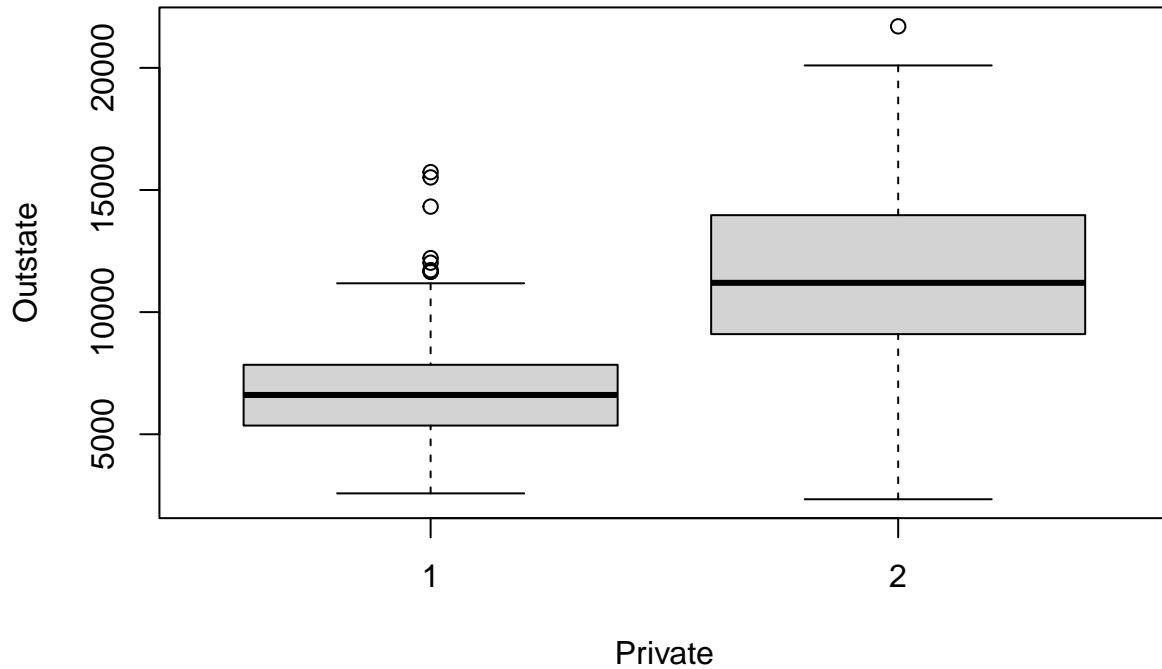


8ciii) Use the plot() function to produce side-by-side boxplots of Outstate versus Private.

```

attach(college)
boxplot(Outstate ~ Private, xlab = "Private", ylab = "Outstate")

```



8civ) Create a new qualitative variable, called Elite, by binning the Top10perc variable. We are going to divide universities into two groups based on whether or not the proportion of students coming from the top 10% of their high school classes exceeds 50%.

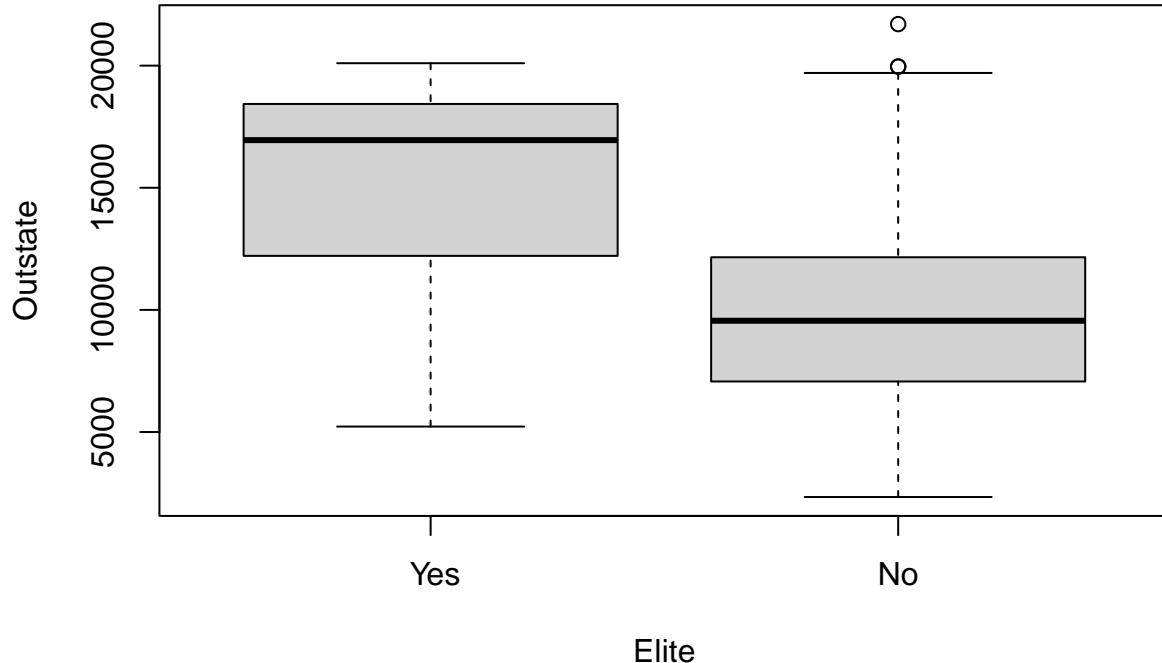
```
Elite = rep("No",nrow(college ))
Elite [college$Top10perc >50]=" Yes"
Elite = as.factor (Elite)
college = data.frame(college ,Elite)
```

Use the summary() function to see how many elite universities there are. Now use the plot() function to produce side-by-side boxplots of Outstate versus Elite.

```
summary(college$Elite)
```

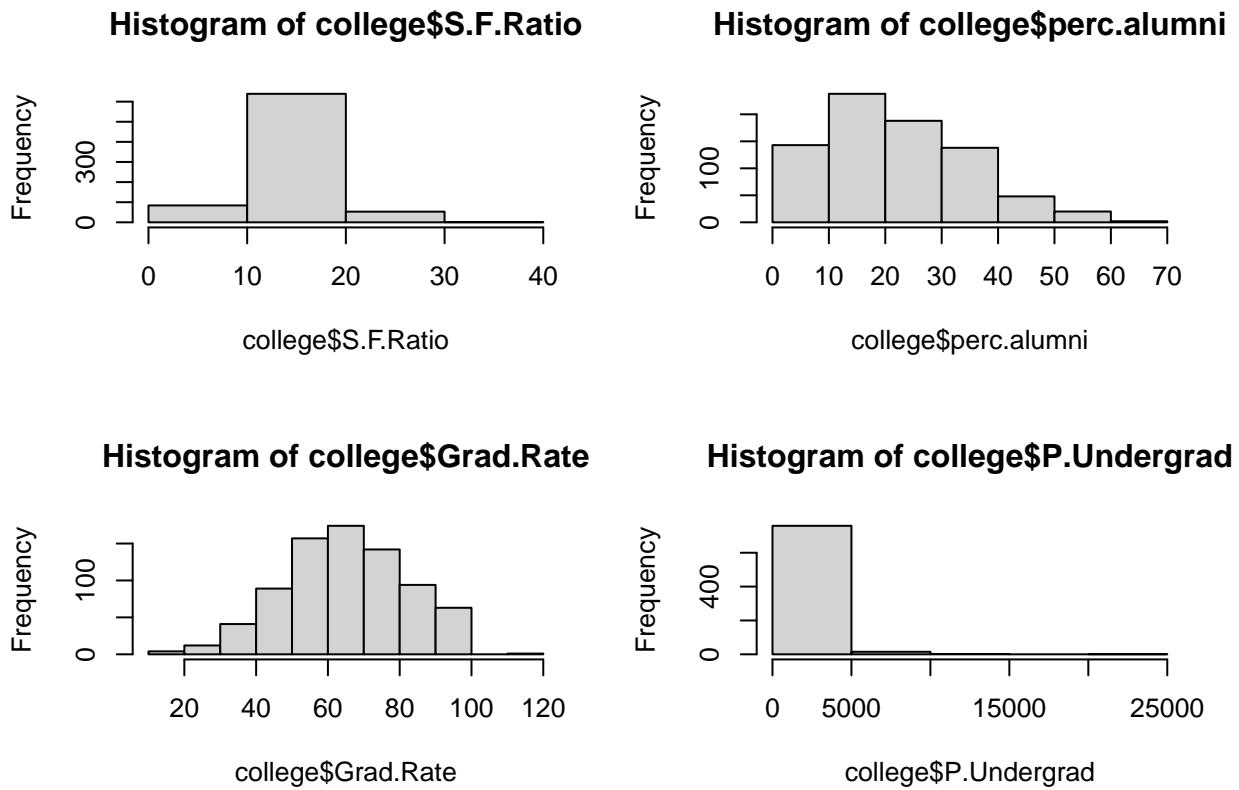
```
##   Yes    No
##    78   699
```

```
boxplot(college$Outstate ~ college$Elite, xlab = "Elite", ylab = "Outstate")
```



8cv) Use the hist() function to produce some histograms with differing numbers of bins for a few of the quantitative variables. You may find the command par(mfrow=c(2,2)) useful: it will divide the print window into four regions so that four plots can be made simultaneously. Modifying the arguments to this function will divide the screen in other ways.

```
par(mfrow=c(2,2))
hist(college$S.F.Ratio, breaks = 5)
hist(college$perc.alumni, breaks = 6)
hist(college$Grad.Rate, breaks = 8)
hist(college$P.Undergrad, breaks = 4)
```



8cvi) Continue exploring the data, and provide a brief summary of what you discover.

- 9) This exercise involves the Auto data set studied in the lab. Make sure that the missing values have been removed from the data.

```
auto = read.csv("Auto.csv")
auto = na.omit(auto)
detach(college)
attach(auto)
```

- 9a) Which of the predictors are quantitative, and which are qualitative?

```
sapply(auto, class)
```

```
##      mpg      cylinders displacement horsepower      weight acceleration
## "numeric" "integer"    "numeric"   "character" "integer"    "numeric"
##      year      origin       name
## "integer" "integer"   "character"
```

From the output of the command it seems as if name is the only qualitative variable

- 9b) What is the range of each quantitative predictor? You can answer this using the range() function.

```
apply(auto[,1:7], 2, range)

##      mpg      cylinders displacement horsepower weight acceleration year
## [1,] "9.0"    "3"          "68.0"       "?"        "1613"    "8.0"      "70"
## [2,] "46.6"   "8"          "455.0"      "98"       "5140"    "24.8"     "82"
```

9c) What is the mean and standard deviation of each quantitative predictor?

```
sapply(auto[,1:7], mean, na.rm = TRUE)
```

```
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
```

```
##      mpg      cylinders displacement horsepower weight acceleration
##      23.515869    5.458438    193.532746      NA 2970.261965    15.555668
##      year
##      75.994962
```

```
sapply(auto[,1:7], sd, na.rm = TRUE)
```

```
## Warning in var(if (is.vector(x) || is.factor(x)) x else as.double(x), na.rm =
## na.rm): NAs introduced by coercion
```

```
##      mpg      cylinders displacement horsepower weight acceleration
##      7.825804    1.701577    104.379583    38.491160    847.904119    2.749995
##      year
##      3.690005
```

9d) Now remove the 10th through 85th observations. What is the range, mean, and standard deviation of each predictor in the subset of the data that remains?

```
sapply(auto[-c(10:85),1:7], range, na.rm = TRUE)
```

```
##      mpg      cylinders displacement horsepower weight acceleration year
## [1,] "11"    "3"          "68"       "?"        "1649"    "8.5"      "70"
## [2,] "46.6"   "8"          "455"      "98"       "4997"    "24.8"     "82"
```

```
sapply(auto[-c(10:85),1:7], mean, na.rm = TRUE)
```

```
## Warning in mean.default(X[[i]], ...): argument is not numeric or logical:
## returning NA
```

```
##      mpg      cylinders displacement horsepower weight acceleration
##      24.438629    5.370717    187.049844      NA 2933.962617    15.723053
##      year
##      77.152648
```

```

sapply(auto[-c(10:85), 1:7], sd, na.rm = TRUE)

## Warning in var(if (is.vector(x) || is.factor(x)) x else as.double(x), na.rm =
## na.rm): NAs introduced by coercion

##          mpg      cylinders displacement horsepower      weight acceleration
##    7.908184     1.653486     99.635385    35.895567   810.642938    2.680514
##      year
##    3.111230

```

9e) Using the full data set, investigate the predictors graphically, using scatterplots or other tools of your choice. Create some plots highlighting the relationships among the predictors. Comment on your findings.

```
auto$horsepower <- as.numeric(type.convert(auto$horsepower))
```

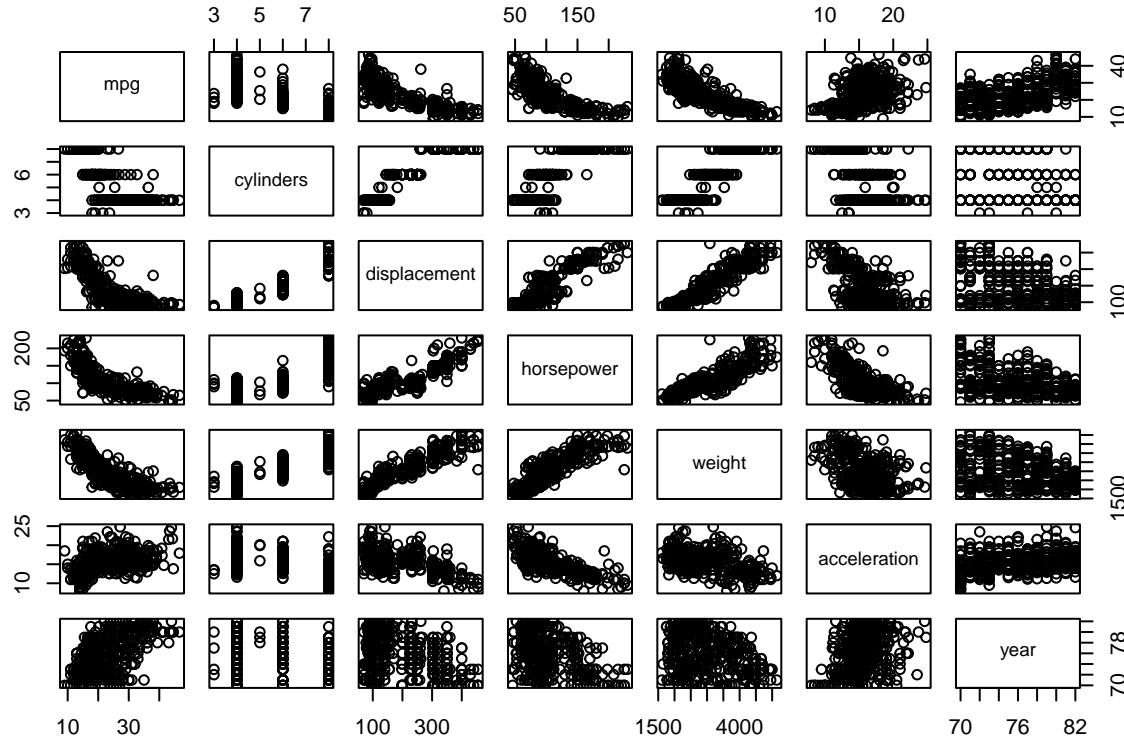
```

## Warning in type.convert.default(auto$horsepower): 'as.is' should be specified by
## the caller; using TRUE

## Warning: NAs introduced by coercion

pairs(auto[, 1:7])

```



9f) Suppose that we wish to predict gas mileage (mpg) on the basis of the other variables. Do your plots suggest that any of the other variables might be useful in predicting mpg? Justify your answer.

```
regression = lm(mpg ~ cylinders+year+weight+horsepower+displacement+acceleration+origin, data= auto)
regression
```

```
## 
## Call:
## lm(formula = mpg ~ cylinders + year + weight + horsepower + displacement +
##     acceleration + origin, data = auto)
## 
## Coefficients:
## (Intercept)      cylinders          year        weight      horsepower
## -17.218435    -0.493376     0.750773    -0.006474    -0.016951
## displacement   acceleration       origin
## 0.019896       0.080576     1.426140
```

10) This exercise involves the Boston housing data set.

```
library(MASS)
data(Boston)
```

10a) How many rows are in this data set? How many columns? What do the rows and columns represent?
506 rows, 14 columns

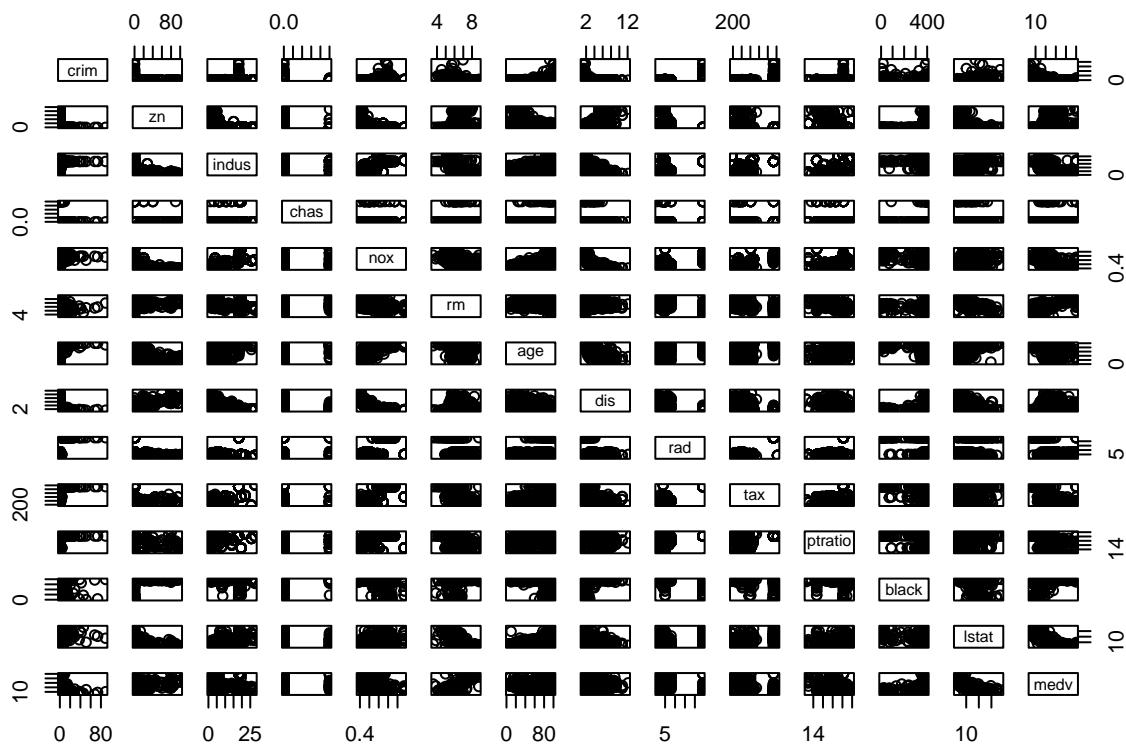
```
dim(Boston)
```



```
## [1] 506 14
```

10b) Make some pairwise scatterplots of the predictors (columns) in this data set. Describe your findings.
looking att the pairs theres a large jumble of data, some of the variable are definitely coorelated while others seem to show nothing at all

```
pairs(Boston)
```



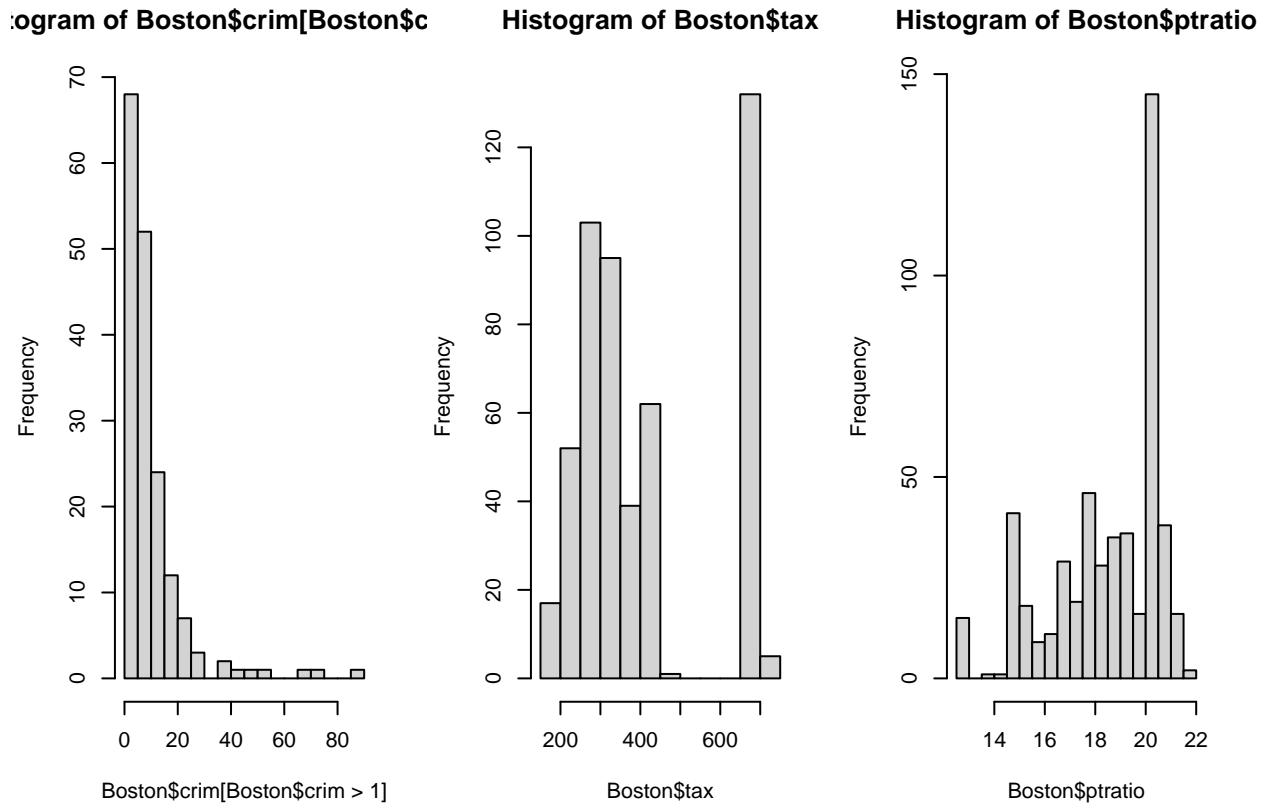
10c) Are any of the predictors associated with per capita crime rate? If so, explain the relationship. It seems that variables with the strongest correlation coefficient are positive

```
Boston.corr.crim = cor(Boston)[-1,1]
print(Boston.corr.crim[order(abs(Boston.corr.crim))])
```

```
##      chas      zn      rm      ptratio      age      dis
## -0.05589158 -0.20046922 -0.21924670  0.28994558  0.35273425 -0.37967009
##      black      medv      indus      nox      lstat      tax
## -0.38506394 -0.38830461  0.40658341  0.42097171  0.45562148  0.58276431
##      rad
##  0.62550515
```

10d) Do any of the suburbs of Boston appear to have particularly high crime rates? Tax rates? Pupil-teacher ratios? Comment on the range of each predictor.
-About 15 suburbs have high crime rates
-There seems to be one outlier in the data, but otherwise tax is uniform
-There seems to be higher skew in teacher ratios

```
par(mfrow=c(1,3))
hist(Boston$crim[Boston$crim > 1], breaks=15)
hist(Boston$tax, breaks=15)
hist(Boston$ptratio, breaks=15)
```



10e) How many of the suburbs in this data set bound the Charles river?

```
sum(Boston$chas == 1)
```

```
## [1] 35
```

10f) What is the median pupil-teacher ratio among the towns in this data set?

```
median(Boston$ptratio)
```

```
## [1] 19.05
```

10g) Which suburb of Boston has lowest median value of owneroccupied homes? What are the values of the other predictors for that suburb, and how do those values compare to the overall ranges for those predictors? Comment on your findings.

```
t(subset(Boston, medv == min(medv)))
```

```
##          399      406
##  crim    38.3518  67.9208
##  zn      0.0000  0.0000
##  indus   18.1000  18.1000
##  chas    0.0000  0.0000
##  nox     0.6930  0.6930
```

```

## rm      5.4530  5.6830
## age    100.0000 100.0000
## dis     1.4896  1.4254
## rad     24.0000 24.0000
## tax    666.0000 666.0000
## ptratio 20.2000 20.2000
## black   396.9000 384.9700
## lstat   30.5900 22.9800
## medv    5.0000  5.0000

```

10h) In this data set, how many of the suburbs average more than seven rooms per dwelling? More than eight rooms per dwelling? Comment on the suburbs that average more than eight rooms per dwelling.

```
sum(Boston$rm > 7)
```

```
## [1] 64
```

```
sum(Boston$rm > 8)
```

```
## [1] 13
```

```
summary(subset(Boston, rm > 8))
```

	crim	zn	indus	chas
## Min.	:0.02009	Min. : 0.00	Min. : 2.680	Min. :0.0000
## 1st Qu.	:0.33147	1st Qu.: 0.00	1st Qu.: 3.970	1st Qu.:0.0000
## Median	:0.52014	Median : 0.00	Median : 6.200	Median :0.0000
## Mean	:0.71879	Mean :13.62	Mean : 7.078	Mean :0.1538
## 3rd Qu.	:0.57834	3rd Qu.:20.00	3rd Qu.: 6.200	3rd Qu.:0.0000
## Max.	:3.47428	Max. :95.00	Max. :19.580	Max. :1.0000
## nox		rm	age	dis
## Min.	:0.4161	Min. :8.034	Min. : 8.40	Min. :1.801
## 1st Qu.	:0.5040	1st Qu.:8.247	1st Qu.:70.40	1st Qu.:2.288
## Median	:0.5070	Median :8.297	Median :78.30	Median :2.894
## Mean	:0.5392	Mean :8.349	Mean :71.54	Mean :3.430
## 3rd Qu.	:0.6050	3rd Qu.:8.398	3rd Qu.:86.50	3rd Qu.:3.652
## Max.	:0.7180	Max. :8.780	Max. :93.90	Max. :8.907
## rad		tax	ptratio	black
## Min.	: 2.000	Min. :224.0	Min. :13.00	Min. :354.6
## 1st Qu.	: 5.000	1st Qu.:264.0	1st Qu.:14.70	1st Qu.:384.5
## Median	: 7.000	Median :307.0	Median :17.40	Median :386.9
## Mean	: 7.462	Mean :325.1	Mean :16.36	Mean :385.2
## 3rd Qu.	: 8.000	3rd Qu.:307.0	3rd Qu.:17.40	3rd Qu.:389.7
## Max.	:24.000	Max. :666.0	Max. :20.20	Max. :396.9
## lstat		medv		
## Min.	:2.47	Min. :21.9		
## 1st Qu.	:3.32	1st Qu.:41.7		
## Median	:4.14	Median :48.3		
## Mean	:4.31	Mean :44.2		
## 3rd Qu.	:5.12	3rd Qu.:50.0		
## Max.	:7.44	Max. :50.0		

Part 3

These lines of code are from the starter file!

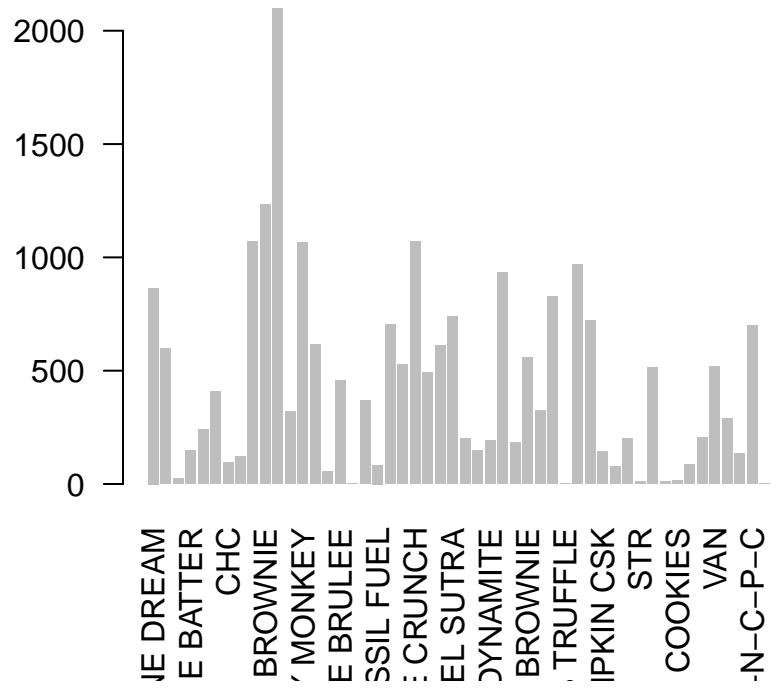
```
benjer = read.csv("BenAndJerry.csv")
names(benjer)

## [1] "quantity"                      "price_paid_deal"
## [3] "price_paid_non_deal"            "coupon_value"
## [5] "promotion_type"                 "total_spent"
## [7] "size1_descr"                   "flavor_descr"
## [9] "formula_descr"                 "household_id"
## [11] "household_size"                "household_income"
## [13] "age_of_female_head"             "age_of_male_head"
## [15] "age_and_presence_of_children"   "male_head_employment"
## [17] "female_head_employment"         "male_head_education"
## [19] "female_head_education"          "marital_status"
## [21] "male_head_occupation"           "female_head_occupation"
## [23] "household_composition"         "race"
## [25] "hispanic_origin"               "region"
## [27] "scantrack_market_identifier"    "projection_factor"
## [29] "fips_state_code"                "fips_county_code"
## [31] "census_tract_county_code"       "type_of_residence"
## [33] "kitchen_appliances"             "tv_items"
## [35] "female_head_birth"              "male_head_birth"
## [37] "household_internet_connection"

priceper1 = (benjer$price_paid_deal + benjer$price_paid_non_deal)/benjer$quantity
y <- log(1+priceper1)
x <- benjer[,c("flavor_descr", "size1_descr",
              "household_income", "household_size")]
x$flavor_descr <- relevel(factor(x$flavor_descr), "VAN")
x$usecoup = factor(benjer$coupon_value>0)
x$couponper1 <- benjer$coupon_value/benjer$quantity
x$region <- factor(benjer$region,
                     levels=1:4, labels=c("East", "Central", "South", "West"))
x$married <- factor(benjer$marital_status==1)
x$race <- factor(benjer$race,
                  levels=1:4, labels=c("white", "black", "asian", "other"))
x$hispanic_origin <- benjer$hispanic_origin==1
x$microwave <- benjer$kitchen_appliances %in% c(1,4,5,7)
x$dishwasher <- benjer$kitchen_appliances %in% c(2,4,6,7)
x$sfh <- benjer$type_of_residence==1
x$internet <- benjer$household_internet_connection==1
x$tv cable <- benjer$tv_items>1
fit <- glm(y~., data=x)
pvals <- summary(fit)$coef[-1,4]
par(mfrow=c(1,1))
par(mar=c(5,10,5,5))
barplot(table(benjer$flavor), border=NA, las=2)
class(benjer$promotion_type)

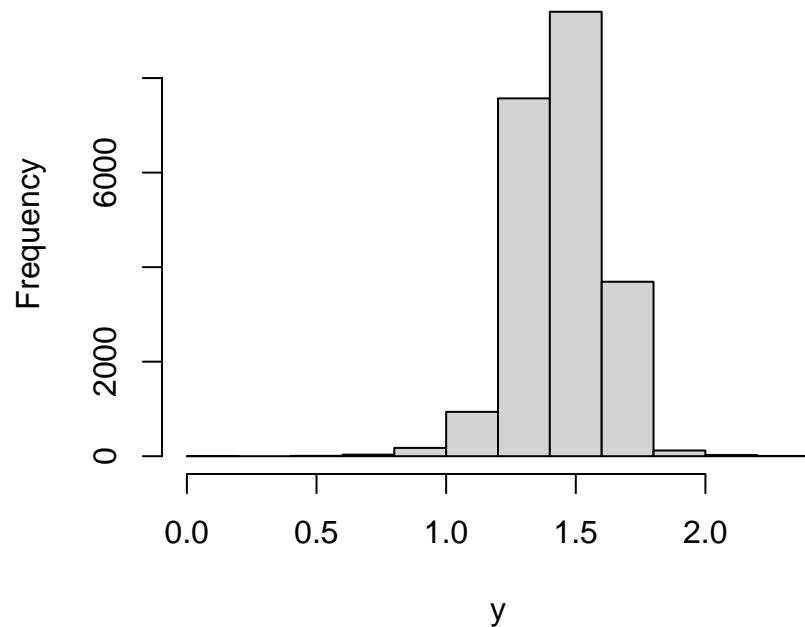
## [1] "integer"
```

```
barplot(table(benjer$flavor), border=NA, las=2)
```

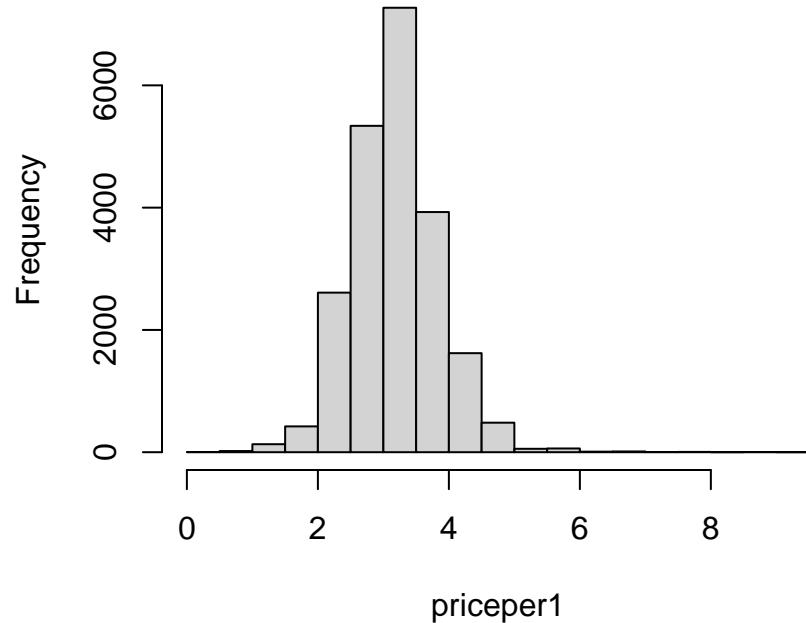


```
hist(y); hist(priceper1);
```

Histogram of y

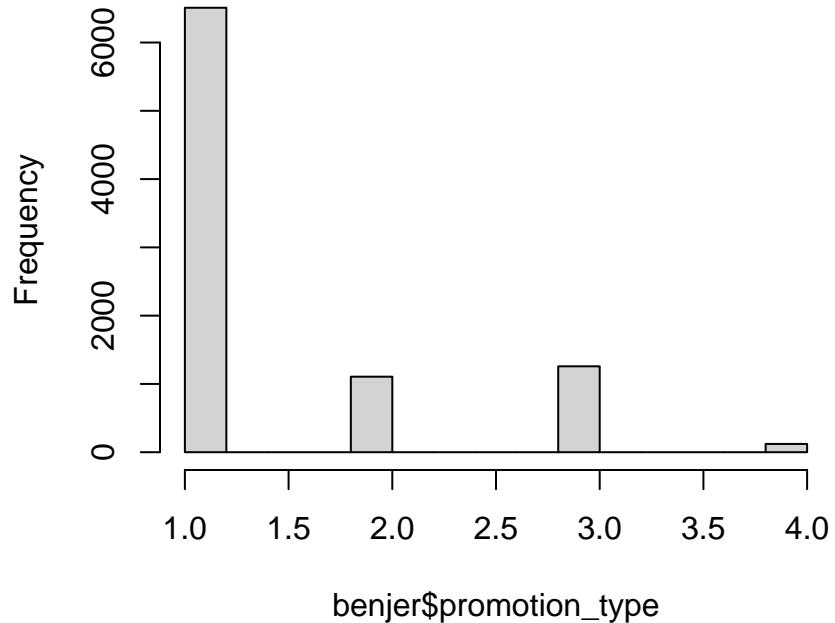


Histogram of priceper1



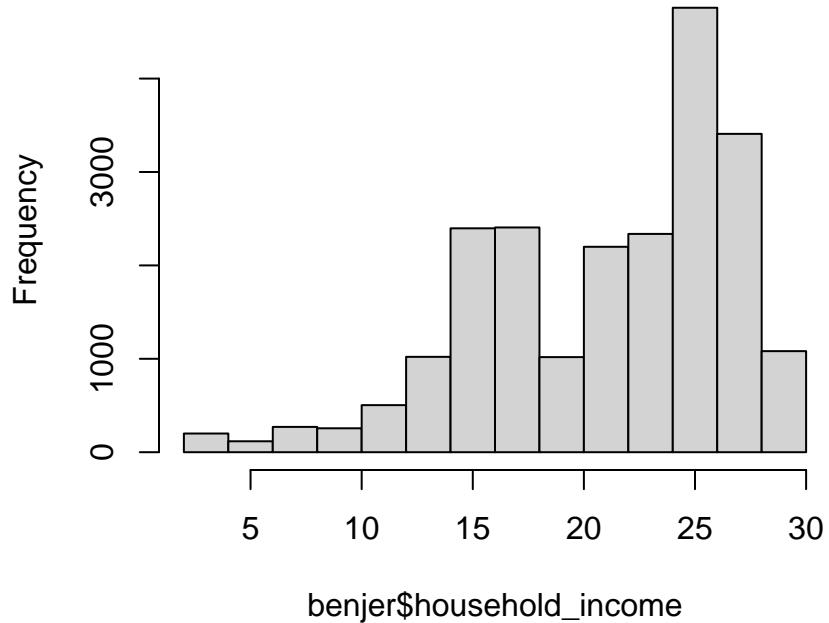
```
hist(benjer$promotion_type)
```

Histogram of benjer\$promotion_type



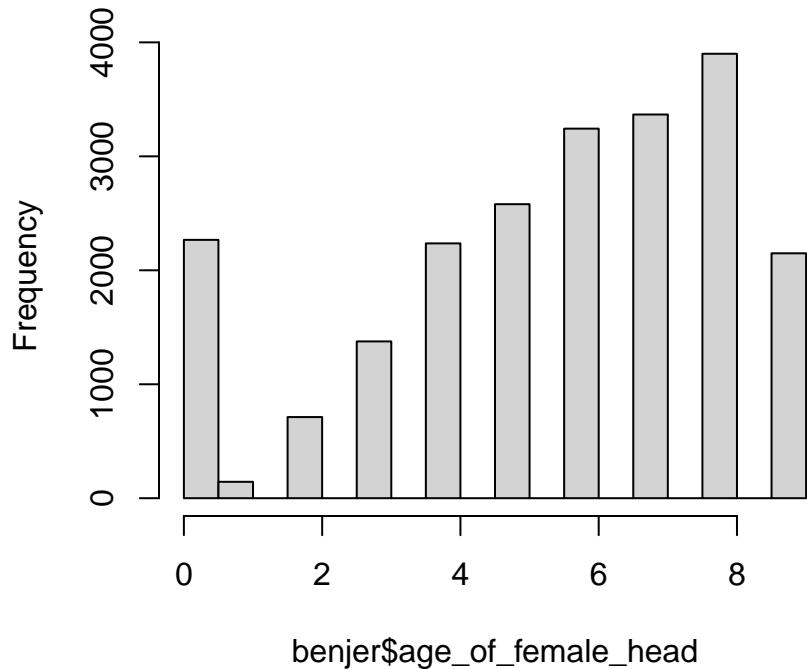
```
hist(benjer$household_income)
```

Histogram of benjer\$household_income



```
hist(benjer$age_of_female_head)
```

Histogram of benjer\$age_of_female_head



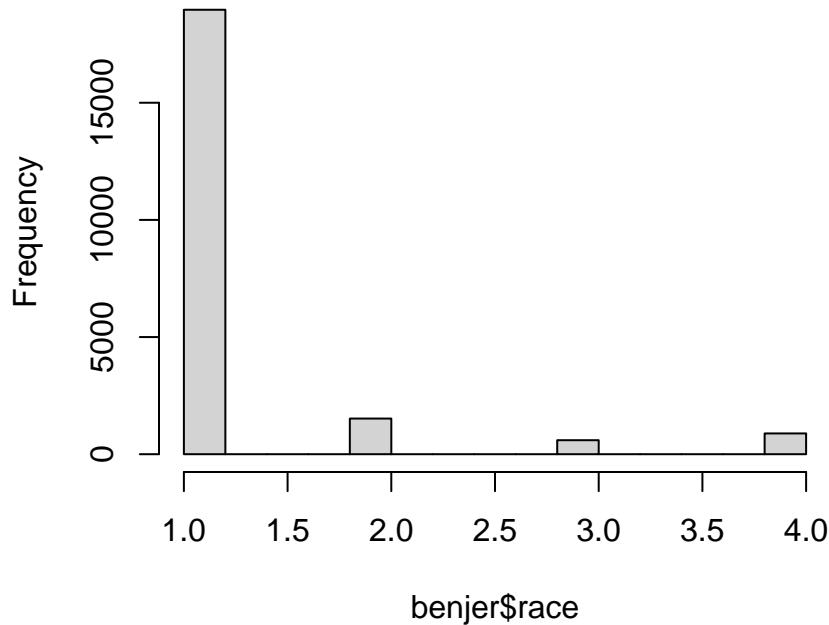
```
hist(benjer$age_of_male_head)
```

Histogram of benjer\$age_of_male_head

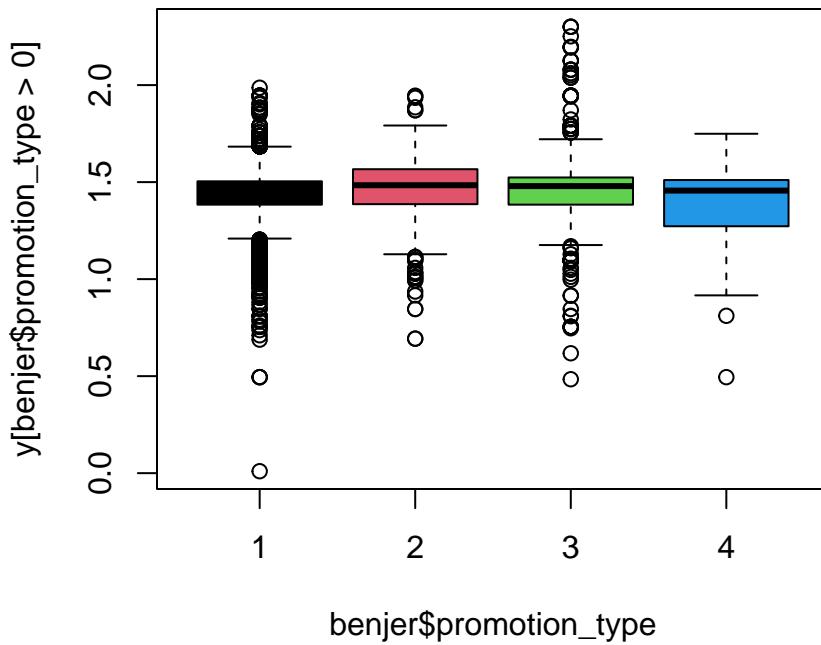


```
hist(benjer$race)
```

Histogram of benjer\$race



```
num_cols <- unlist(lapply(benjer, is.numeric))
boxplot(y[benjer$promotion_type>0] ~ benjer$promotion_type, col= levels(factor(benjer$promotion)))
```



```
summary(fit)
```

```
##
## Call:
## glm(formula = y ~ ., data = x)
##
## Deviance Residuals:
##      Min        1Q     Median        3Q       Max
## -1.45414   -0.06617    0.01078    0.08318    0.64849
##
## Coefficients:
## (Intercept)              Estimate Std. Error t value
## flavor_descrAMERICONE DREAM 1.4940011 0.0104454 143.029
## flavor_descrBANANA SPLIT   -0.0270131 0.0082384 -3.279
## flavor_descrBLACK & TAN   -0.0408377 0.0088887 -4.594
## flavor_descrBROWNIE BATTER -0.0769758 0.0302280 -2.547
## flavor_descrBUTTER PECAN   -0.0081789 0.0138350 -0.591
## flavor_descrCAKE BATTER   -0.0130912 0.0115124 -1.137
## flavor_descrCHC             -0.0385251 0.0097894 -3.935
## flavor_descrCHC ALMOND NOUGAT -0.0090327 0.0164365 -0.550
## flavor_descrCHC CHIP C-DH   -0.0206583 0.0150504 -1.373
## flavor_descrCHC FUDGE BROWNIE -0.0235566 0.0079638 -2.958
## flavor_descrCHERRY GRCA    -0.0243969 0.0077820 -3.135
## flavor_descrCHUBBY HUBBY   -0.0310569 0.0072923 -4.259
## flavor_descrCHUBBY HUBBY   -0.0098033 0.0105370 -0.930
```

## flavor_descrCHUNKY MONKEY	-0.0370629	0.0079417	-4.667
## flavor_descrCINNAMON BUNS	-0.0431549	0.0088415	-4.881
## flavor_descrCOFFEE	0.0267102	0.0207441	1.288
## flavor_descrCREME BRULEE	-0.0476383	0.0095128	-5.008
## flavor_descrDOUBLE CHOCOLATE FUDGE SWR	0.0146750	0.1474511	0.100
## flavor_descrDUBLIN MUDSLIDE	-0.0143134	0.0100695	-1.421
## flavor_descrFOSSIL FUEL	-0.0495130	0.0173355	-2.856
## flavor_descrHALF BAKED	-0.0337535	0.0085784	-3.935
## flavor_descrHEATH CANDY EVERYTHING BUT THE	-0.0356518	0.0091582	-3.893
## flavor_descrHEATH COFFEE CRUNCH	-0.0215018	0.0079153	-2.716
## flavor_descrHEATH CRUNCH	-0.0027050	0.0092997	-0.291
## flavor_descrIMAGINE WHIRLED PEACE	-0.0411508	0.0088443	-4.653
## flavor_descrKARAMEL SUTRA	-0.0316114	0.0084836	-3.726
## flavor_descrMAGIC BROWNIES	-0.0412340	0.0123096	-3.350
## flavor_descrMINT CHOCOLATE CHIP CHUNK	0.0127854	0.0138661	0.922
## flavor_descrNEAPOLITAN DYNAMITE	-0.0250128	0.0125600	-1.991
## flavor_descrNEW YORK SUPER FUDGE CHUNK	-0.0331176	0.0081133	-4.082
## flavor_descrOATMEAL COOKIE CHUNK	-0.0230802	0.0126699	-1.822
## flavor_descrONE CSK BROWNIE	-0.052211	0.0090597	-6.095
## flavor_descrOXFORD MINT CHOCOLATE COOKIE	-0.0254025	0.0104416	-2.433
## flavor_descrPB CUP	-0.0273918	0.0082965	-3.302
## flavor_descrPB TRUFFLE	-0.0585094	0.1474697	-0.397
## flavor_descrPHISH FOOD	-0.0191778	0.0080634	-2.378
## flavor_descrPISTACHIO PISTACHIO	-0.0273626	0.0085204	-3.211
## flavor_descrPUMPKIN CSK	-0.0965602	0.0139765	-6.909
## flavor_descrRSP CHOCOLATE CHIP CHUNK	0.0004508	0.0178187	0.025
## flavor_descrSMORES	0.0071988	0.0123221	0.584
## flavor_descrSTR	0.0482053	0.0449004	1.074
## flavor_descrSTR CSK	-0.0425389	0.0092260	-4.611
## flavor_descrSTRAWBERRIES & CREAM	-0.0193492	0.0413753	-0.468
## flavor_descrSWEET CREAM & COOKIES	0.0342785	0.0363482	0.943
## flavor_descrTRIPLE CARAMEL CHIP	-0.0404664	0.0170940	-2.367
## flavor_descrTURTLE SOUP	-0.0541433	0.0122557	-4.418
## flavor_descrVAN CARAMEL FUDGE	-0.0424817	0.0108471	-3.916
## flavor_descrVERMONTY PYTHON	-0.0229930	0.0143106	-1.607
## flavor_descriW-N-C-P-C	-0.0335837	0.0085858	-3.912
## flavor_descrWHITE RUSSIAN	0.1962617	0.1474626	1.331
## size1_descr32.0 MLOZ	0.3108154	0.0080789	38.472
## household_income	0.0020848	0.0002032	10.262
## household_size	-0.0040760	0.0008837	-4.612
## usecoupTRUE	-0.0709608	0.0045418	-15.624
## couponper1	0.0683420	0.0027052	25.263
## regionCentral	-0.0212732	0.0031093	-6.842
## regionSouth	-0.0247856	0.0029581	-8.379
## regionWest	-0.0184250	0.0030072	-6.127
## marriedTRUE	-0.0113292	0.0025633	-4.420
## raceblack	0.0123061	0.0040044	3.073
## raceasian	-0.0026165	0.0062476	-0.419
## raceother	0.0154499	0.0058016	2.663
## hispanic_originTRUE	-0.0010270	0.0053361	-0.192
## microwaveTRUE	-0.0350319	0.0075637	-4.632
## dishwasherTRUE	-0.0017803	0.0025721	-0.692
## sfhTRUE	-0.0091538	0.0024985	-3.664
## internetTRUE	0.0064933	0.0028001	2.319

```

## tvcableTRUE          0.0059701  0.0021529   2.773
##                               Pr(>|t|)
## (Intercept)                < 2e-16 ***
## flavor_descriAMERICONE DREAM      0.001044 **
## flavor_descriBANANA SPLIT        4.37e-06 ***
## flavor_descriBLACK & TAN         0.010887 *
## flavor_descriBROWNIE BATTER      0.554409
## flavor_descriBUTTER PECAN        0.255492
## flavor_descriCAKE BATTER        8.33e-05 ***
## flavor_descriCHC               0.582633
## flavor_descriCHC ALMOND NOUGAT  0.169888
## flavor_descriCHC CHIP C-DH       0.003100 **
## flavor_descriCHC FUDGE BROWNIE  0.001720 **
## flavor_descriCHERRY GRCA         2.06e-05 ***
## flavor_descriCHUBBY HUBBY        0.352189
## flavor_descriCHUNKY MONKEY      3.08e-06 ***
## flavor_descriCINNAMON BUNS       1.06e-06 ***
## flavor_descriCOFFEE             0.197896
## flavor_descriCREME BRULEE       5.55e-07 ***
## flavor_descriDOUBLE CHC FUDGE SWR 0.920723
## flavor_descriDUBLIN MUDSLIDE     0.155195
## flavor_descriFOSSIL FUEL        0.004292 **
## flavor_descriHALF BAKED          8.36e-05 ***
## flavor_descriHEATH CANDY EVERYTHING BUT THE 9.93e-05 ***
## flavor_descriHEATH COFFEE CRUNCH 0.006603 **
## flavor_descriHEATH CRUNCH        0.771151
## flavor_descriIMAGINE WHIRLED PEACE 3.29e-06 ***
## flavor_descriKARAMEL SUTRA       0.000195 ***
## flavor_descriMAGIC BROWNIES     0.000810 ***
## flavor_descriMINT CHC CHUNK      0.356509
## flavor_descriNEAPOLITAN DYNAMITE 0.046442 *
## flavor_descriNEW YORK SUPER FUDGE CHUNK 4.48e-05 ***
## flavor_descriOATMEAL COOKIE CHUNK 0.068522 .
## flavor_descriONE CSK BROWNIE     1.11e-09 ***
## flavor_descriOXFORD MINT CHC COOKIE 0.014989 *
## flavor_descriPB CUP              0.000963 ***
## flavor_descriPB TRUFFLE          0.691552
## flavor_descriPHISH FOOD         0.017398 *
## flavor_descriPISTACHIO PISTACHIO 0.001323 **
## flavor_descriPUMPKIN CSK          5.02e-12 ***
## flavor_descriRSP CHC CHUNK       0.979816
## flavor_descriSMORES              0.559080
## flavor_descriSTR                 0.283012
## flavor_descriSTR CSK              4.03e-06 ***
## flavor_descriSTRAWBERRIES & CREAM 0.640039
## flavor_descriSWEET CREAM & COOKIES 0.345660
## flavor_descriTRIPLE CARAMEL CHUNK 0.017927 *
## flavor_descriTURTLE SOUP          1.00e-05 ***
## flavor_descriVAN CARAMEL FUDGE    9.01e-05 ***
## flavor_descriVERMONTY PYTHON      0.108130
## flavor_descriW-N-C-P-C           9.20e-05 ***
## flavor_descriWHITE RUSSIAN        0.183228
## size1_descri32.0 MLOZ            < 2e-16 ***
## household_income                 < 2e-16 ***

```

```

## household_size          4.01e-06 ***
## usecoupTRUE             < 2e-16 ***
## couponper1              < 2e-16 ***
## regionCentral            8.02e-12 ***
## regionSouth               < 2e-16 ***
## regionWest                9.11e-10 ***
## marriedTRUE                 9.93e-06 ***
## raceblack                  0.002121 **
## raceasian                   0.675362
## raceother                  0.007749 **
## hispanic_originTRUE        0.847390
## microwaveTRUE                3.65e-06 ***
## dishwasherTRUE                0.488849
## sfhTRUE                     0.000249 ***
## internetTRUE                  0.020406 *
## tvcableTRUE                  0.005559 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.0216865)
##
##      Null deviance: 537.46  on 21939  degrees of freedom
## Residual deviance: 474.33  on 21872  degrees of freedom
##   (34 observations deleted due to missingness)
## AIC: -21721
##
## Number of Fisher Scoring iterations: 2

fit2 <- glm(y~. - flavor_descr, data=x)
summary(fit2)

```

```

##
## Call:
## glm(formula = y ~ . - flavor_descr, data = x)
##
## Deviance Residuals:
##      Min        1Q        Median         3Q        Max
## -1.48246  -0.06547   0.01350   0.08183   0.65363
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)           1.4671681  0.0085509 171.581 < 2e-16 ***
## size1_descr32.0_MLOZ  0.3122884  0.0078001  40.037 < 2e-16 ***
## household_income       0.0022125  0.0002024  10.928 < 2e-16 ***
## household_size        -0.0044408  0.0008776 -5.060 4.23e-07 ***
## usecoupTRUE            -0.0700565  0.0045481 -15.404 < 2e-16 ***
## couponper1              0.0687980  0.0027103  25.384 < 2e-16 ***
## regionCentral           -0.0238388  0.0030879 -7.720 1.21e-14 ***
## regionSouth              -0.0272741  0.0029376 -9.285 < 2e-16 ***
## regionWest                -0.0190740  0.0029932 -6.372 1.90e-10 ***
## marriedTRUE                -0.0114444  0.0025625 -4.466 8.00e-06 ***
## raceblack                  0.0112838  0.0039853  2.831 0.004639 **
## raceasian                  -0.0034105  0.0062424 -0.546 0.584837
## raceother                  0.0170006  0.0057942  2.934 0.003349 **

```

```

## hispanic_originTRUE -0.0009265 0.0053390 -0.174 0.862230
## microwaveTRUE      -0.0378018 0.0075614 -4.999 5.80e-07 ***
## dishwasherTRUE     -0.0008110 0.0025715 -0.315 0.752469
## sfhTRUE            -0.0094224 0.0024914 -3.782 0.000156 ***
## internetTRUE       0.0054607 0.0027974  1.952 0.050949 .
## tvcableTRUE        0.0064237 0.0021490  2.989 0.002801 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.02183617)
##
## Null deviance: 537.46  on 21939  degrees of freedom
## Residual deviance: 478.67  on 21921  degrees of freedom
##   (34 observations deleted due to missingness)
## AIC: -21619
##
## Number of Fisher Scoring iterations: 2

fit3 <- glm(y~. -flavor_descr -race -dishwasher - hispanic_origin, data=x)
summary(fit3)

```

```

##
## Call:
## glm(formula = y ~ . - flavor_descr - race - dishwasher - hispanic_origin,
##      data = x)
##
## Deviance Residuals:
##      Min        1Q        Median         3Q        Max
## -1.48390  -0.06576   0.01470   0.08163   0.65247
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)           1.4682779  0.0085423 171.884 < 2e-16 ***
## size1_descr32.0_MLOZ  0.3125617  0.0077985  40.080 < 2e-16 ***
## household_income      0.0021845  0.0001961  11.140 < 2e-16 ***
## household_size        -0.0041371  0.0008737 -4.735 2.20e-06 ***
## usecoupTRUE          -0.0698324  0.0045478 -15.355 < 2e-16 ***
## couponper1            0.0687859  0.0027110  25.373 < 2e-16 ***
## regionCentral         -0.0238157  0.0030865 -7.716 1.25e-14 ***
## regionSouth           -0.0266641  0.0029023 -9.187 < 2e-16 ***
## regionWest            -0.0187114  0.0029605 -6.320 2.66e-10 ***
## marriedTRUE           -0.0119691  0.0025485 -4.697 2.66e-06 ***
## microwaveTRUE         -0.0380804  0.0075199 -5.064 4.14e-07 ***
## sfhTRUE               -0.0098788  0.0024807 -3.982 6.85e-05 ***
## internetTRUE          0.0054916  0.0027912  1.967  0.04915 *
## tvcableTRUE           0.0064773  0.0021471  3.017  0.00256 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.02184969)
##
## Null deviance: 537.46  on 21939  degrees of freedom
## Residual deviance: 479.08  on 21926  degrees of freedom
##   (34 observations deleted due to missingness)

```

```

## AIC: -21610
##
## Number of Fisher Scoring iterations: 2

fit4 <- glm(y~. -flavor_descr -race -dishwasher - hispanic_origin -internet -tvcable, data=x)
summary(fit4)

##
## Call:
## glm(formula = y ~ . - flavor_descr - race - dishwasher - hispanic_origin -
##      internet - tvcable, data = x)
##
## Deviance Residuals:
##    Min      1Q   Median      3Q     Max
## -1.48230 -0.06507  0.01479  0.08111  0.65518
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)           1.4741213  0.0083233 177.108 < 2e-16 ***
## size1_descr32.0 MLOZ  0.3121873  0.0077971  40.039 < 2e-16 ***
## household_income     0.0022875  0.0001941  11.783 < 2e-16 ***
## household_size       -0.0039566  0.0008705 -4.545 5.51e-06 ***
## usecoupTRUE          -0.0695301  0.0045478 -15.289 < 2e-16 ***
## couponper1            0.0686635  0.0027115  25.323 < 2e-16 ***
## regionCentral         -0.0246476  0.0030726 -8.022 1.10e-15 ***
## regionSouth           -0.0273725  0.0028818 -9.498 < 2e-16 ***
## regionWest             -0.0197059  0.0029398 -6.703 2.09e-11 ***
## marriedTRUE            -0.0123696  0.0025381 -4.873 1.10e-06 ***
## microwaveTRUE          -0.0359021  0.0074900 -4.793 1.65e-06 ***
## sfhTRUE                -0.0112333  0.0024533 -4.579 4.70e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.02186124)
##
## Null deviance: 537.46 on 21939 degrees of freedom
## Residual deviance: 479.37 on 21928 degrees of freedom
## (34 observations deleted due to missingness)
## AIC: -21600
##
## Number of Fisher Scoring iterations: 2

n=nrow(x)
BIC <- c(reg1=extractAIC(fit, k=log(n))[2],
        reg2=extractAIC(fit2, k=log(n))[2],
        reg3=extractAIC(fit3, k=log(n))[2],
        reg4=extractAIC(fit4, k=log(n))[2])
eBIC <- exp(-0.5*(BIC-min(BIC)))
probs <- eBIC/sum(eBIC)
round(probs, 5)

##    reg1    reg2    reg3    reg4
## 0.00000 0.00000 0.03917 0.96083

```

1. Explore the data and visualize: what variables are interesting? Choose a few, plot them together, and tell a story. As can be seen from the dataset the age of heads seems to have a large significant amount at the younger population as it steadily increases throughout the age group. Those who do eat at ben and jerry's also tend to have a higher household income. Furthermore the majority of people tend to of race type 1. There also seems to be a higher spread among promotions of type 3
2. Describe the regression model in the code. Improve it? The first model is a baseline in which the log of price per ice cream is regressed against many variables. When the flavor variable was taken out it was found that, asian, hispanic, and dishwashwer, were not significant so they were taken out of the model, which is the third one. Perhaps one way to improve the model is to only include regressors that are significant at the 0.01 alpha level. If this is done, we remove the varialbe internet and tvcable. Doing so will show a higher BIC value for this model. This can be seen in the code by fit4
3. Take the p-values from your regression and look for evidence of association. Relate what you learn to your story from 1. From the p-values it can be seen that all aggressors in fit4 are statistically significant at the 0.1 alpha level. Thus the variables of size1_descr32.0 MLOZ 0.3121873 0.0077971 40.039 < 2e-16 *** household_income, household_size, usecoupTRUE, couponper1, regionCentral, regionSouth,regionWest, marriedTRUE, microwaveTRUE, and sfhTRUE do seem to have an association on the log of price.