M1-applied solutions

Q8

8(a)

In [71]:
```
#M1-Applied
#8(a)
College = read.csv("/Users/priyanka/desktop/College.csv")
```

In [72]:
```
dim(College)
```

1. 777
2. 19

8(b)

In [42]:
```
#8(b)
head(College)    #pops up the data instead of fix()
```

	X	Private	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Un
Abilene Christian University	Abilene Christian University	Yes	1660	1232	721	23	52	2885	
Adelphi University	Adelphi University	Yes	2186	1924	512	16	29	2683	
Adrian College	Adrian College	Yes	1428	1097	336	22	50	1036	
Agnes Scott College	Agnes Scott College	Yes	417	349	137	60	89	510	
Alaska Pacific University	Alaska Pacific University	Yes	193	146	55	16	44	249	
Albertson College	Albertson College	Yes	587	479	158	38	62	678	

In [21]:
```
#setting row names
rownames(College) = College[,1]
# droping the first coloumn
college = College[,-1]
head(College)
```

	X	Private	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Un
Abilene Christian University	Abilene Christian University	Yes	1660	1232	721	23	52	2885	
Adelphi University	Adelphi University	Yes	2186	1924	512	16	29	2683	
Adrian College	Adrian College	Yes	1428	1097	336	22	50	1036	
Agnes Scott College	Agnes Scott College	Yes	417	349	137	60	89	510	
Alaska Pacific University	Alaska Pacific University	Yes	193	146	55	16	44	249	
Albertson College	Albertson College	Yes	587	479	158	38	62	678	

8(c) i

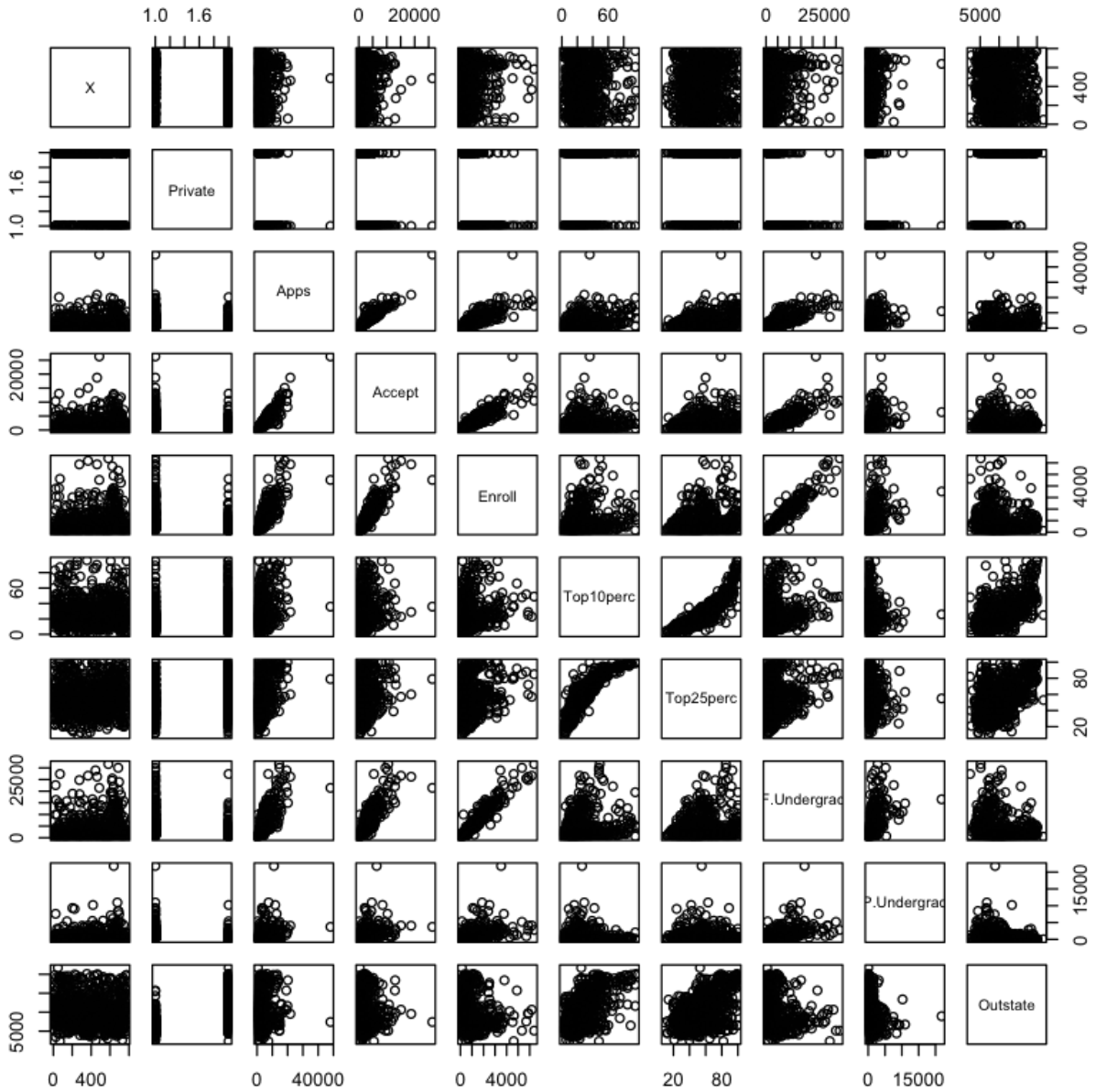
In [22]:
```
#8(c)i
summary(College)
```

```
                             X       Private        Apps           Accept
Abilene Christian University:  1   No :212   Min.   :   81   Min.   :   72
Adelphi University           :  1   Yes:565   1st Qu.:  776   1st Qu.:  604
Adrian College               :  1             Median : 1558   Median : 1110
Agnes Scott College          :  1             Mean   : 3002   Mean   : 2019
Alaska Pacific University    :  1             3rd Qu.: 3624   3rd Qu.: 2424
Albertson College            :  1             Max.   :48094   Max.   :26330
(Other)                      :771
    Enroll       Top10perc       Top25perc      F.Undergrad
Min.   :  35   Min.   : 1.00   Min.   :  9.0   Min.   :  139
1st Qu.: 242   1st Qu.:15.00   1st Qu.: 41.0   1st Qu.:  992
Median : 434   Median :23.00   Median : 54.0   Median : 1707
Mean   : 780   Mean   :27.56   Mean   : 55.8   Mean   : 3700
3rd Qu.: 902   3rd Qu.:35.00   3rd Qu.: 69.0   3rd Qu.: 4005
Max.   :6392   Max.   :96.00   Max.   :100.0   Max.   :31643

 P.Undergrad         Outstate       Room.Board       Books
Min.   :    1.0   Min.   : 2340   Min.   :1780   Min.   :  96.0
1st Qu.:   95.0   1st Qu.: 7320   1st Qu.:3597   1st Qu.: 470.0
Median :  353.0   Median : 9990   Median :4200   Median : 500.0
Mean   :  855.3   Mean   :10441   Mean   :4358   Mean   : 549.4
3rd Qu.:  967.0   3rd Qu.:12925   3rd Qu.:5050   3rd Qu.: 600.0
Max.   :21836.0   Max.   :21700   Max.   :8124   Max.   :2340.0

   Personal         PhD            Terminal       S.F.Ratio
Min.   : 250   Min.   :  8.00   Min.   : 24.0   Min.   : 2.50
1st Qu.: 850   1st Qu.: 62.00   1st Qu.: 71.0   1st Qu.:11.50
Median :1200   Median : 75.00   Median : 82.0   Median :13.60
Mean   :1341   Mean   : 72.66   Mean   : 79.7   Mean   :14.09
3rd Qu.:1700   3rd Qu.: 85.00   3rd Qu.: 92.0   3rd Qu.:16.50
Max.   :6800   Max.   :103.00   Max.   :100.0   Max.   :39.80

 perc.alumni        Expend        Grad.Rate
Min.   : 0.00   Min.   : 3186   Min.   : 10.00
1st Qu.:13.00   1st Qu.: 6751   1st Qu.: 53.00
Median :21.00   Median : 8377   Median : 65.00
Mean   :22.74   Mean   : 9660   Mean   : 65.46
3rd Qu.:31.00   3rd Qu.:10830   3rd Qu.: 78.00
Max.   :64.00   Max.   :56233   Max.   :118.00
```

8(c)ii

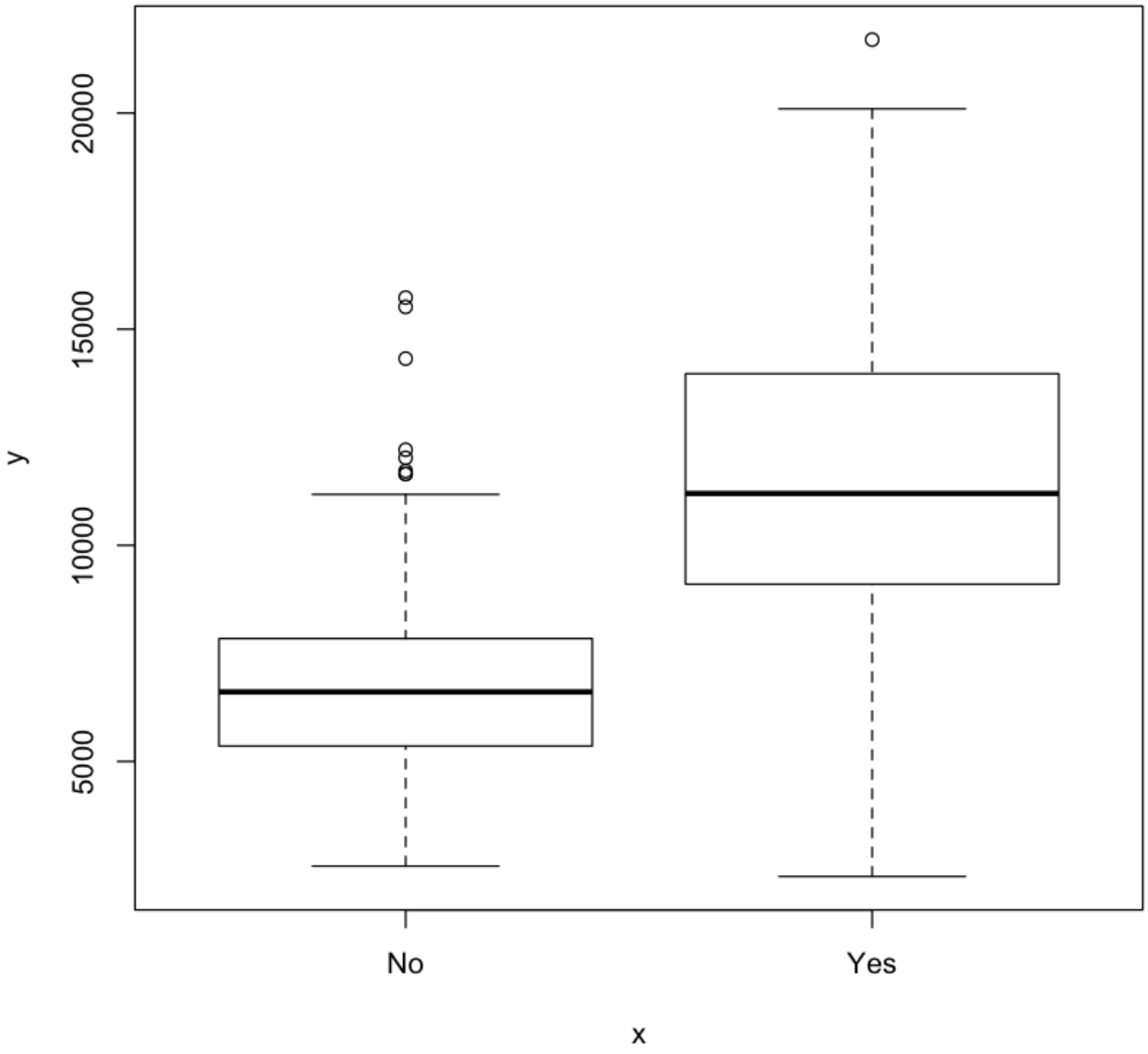
In [26]:

```
#8(c)ii
pairs(College[, 1:10])
```



8(c)iii

In [28]:
```
#8(c)iii
plot(College$Private,College$Outstate)
#attach(College) (alternative way)
#plot(Private,Outstate)
```

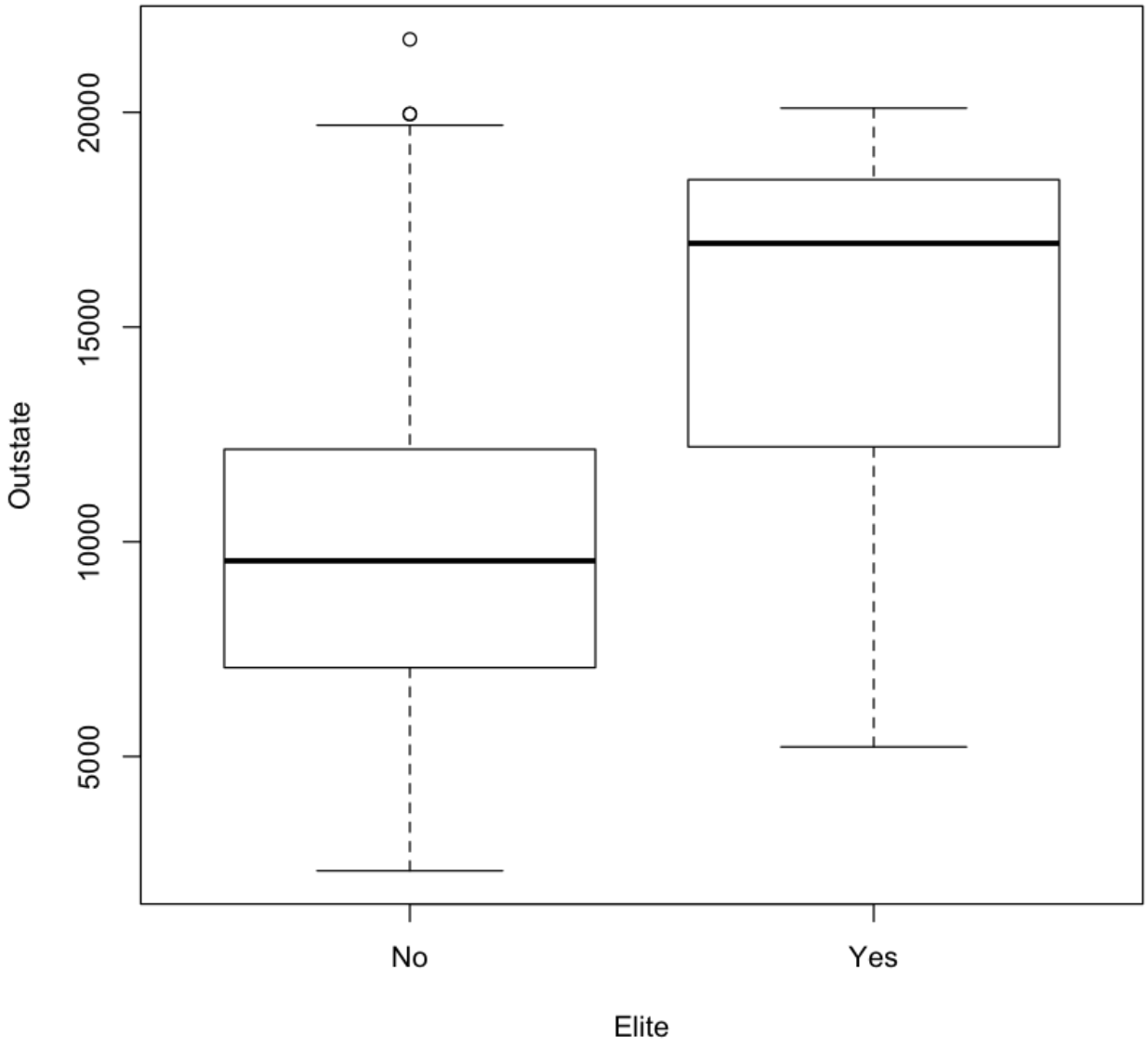


8(c)iv

In [31]:
```
#8(c)iv

Elite = rep("No", nrow(college))
#Created a vector with equal length to the columns of of college with
#the inital value of "No"

Elite[college$Top10perc >50] = "Yes"
#indexing rows of the college data where the Top10perc column
#is greater than 50 and changing that row value to "Yes"

Elite = as.factor(Elite)
#converting quanti to quali

college=data.frame(college, Elite)
 #joining the college and the Elite

summary(college$Elite)
#78 Elite
```

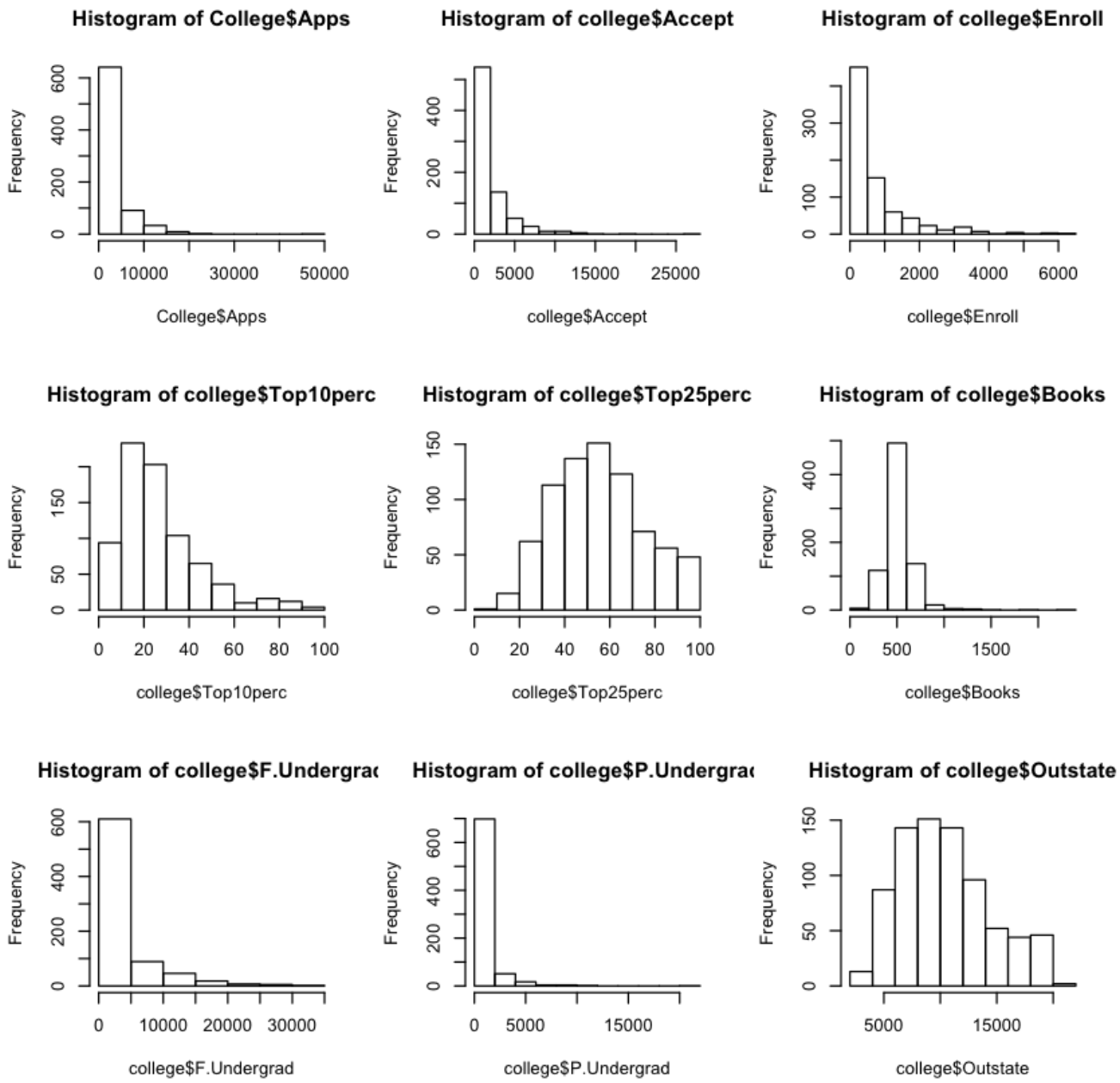
No	699
Yes	78

In [34]:
```
plot(college$Elite, college$Outstate, xlab="Elite", ylab="Outstate")
```



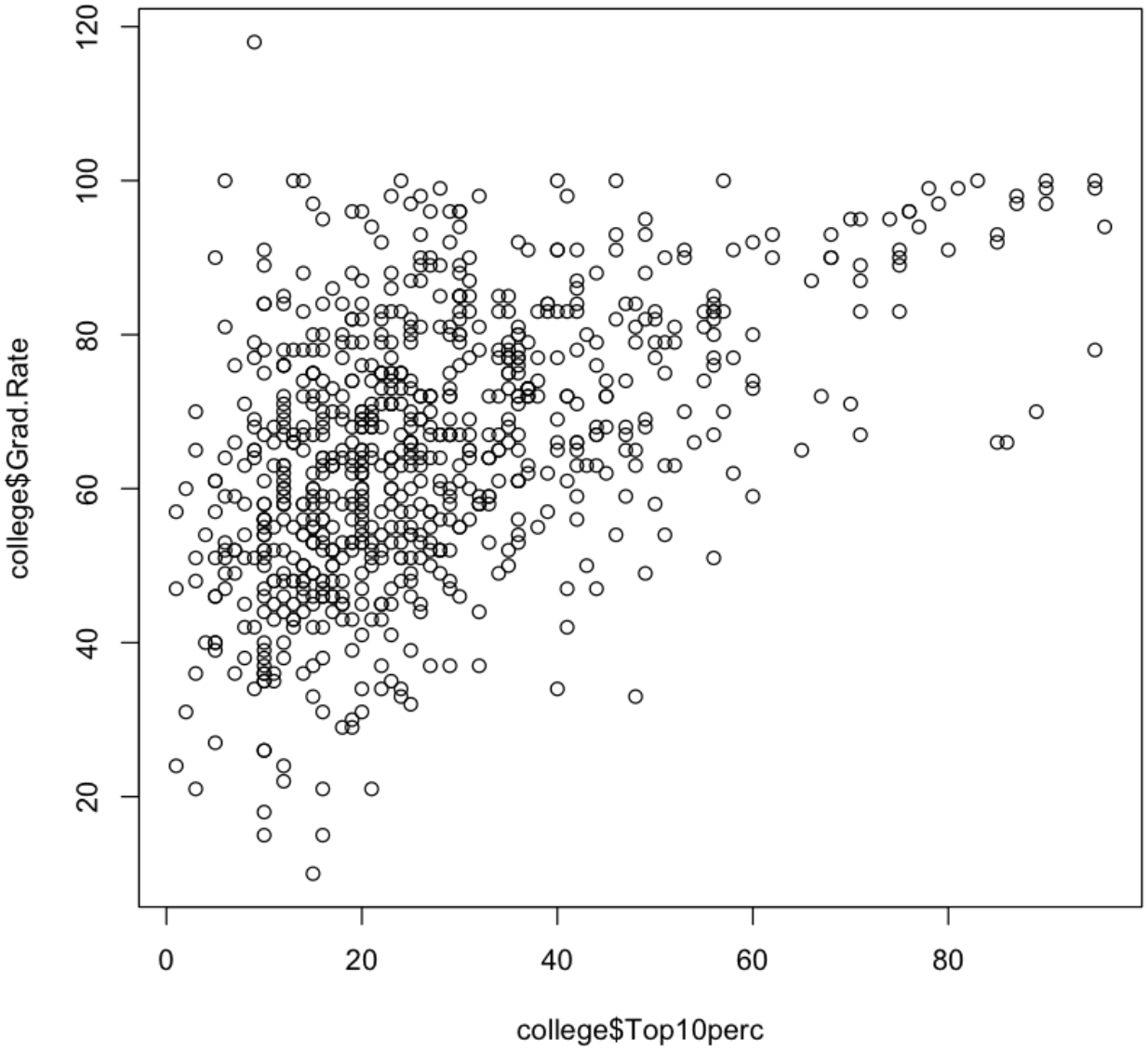
8(c)v

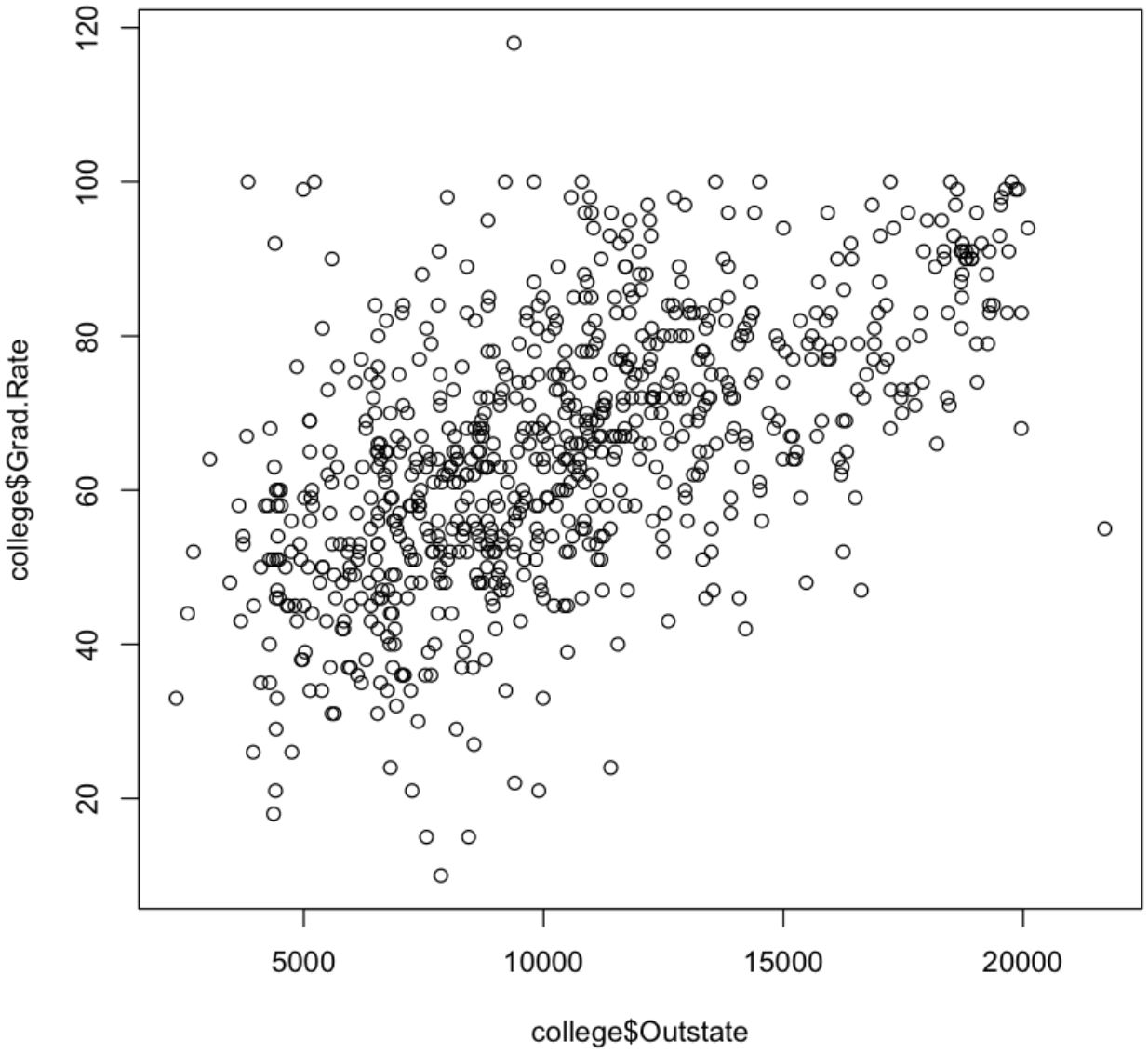
In [40]:
```
#8(c)v
par(mfrow=c(3,3))
hist(College$Apps)
hist(college$Accept)
hist(college$Enroll)
hist(college$Top10perc)
hist(college$Top25perc)
hist(college$Books)
hist(college$F.Undergrad)
hist(college$P.Undergrad)
hist(college$Outstate)
```



8(c)vi - observations

In [44]:
```
#8(c)vi - observations

par(mfrow=c(1,1))

plot(college$Top10perc, college$Grad.Rate)
#Colleges with the most students from top 10% perc don't necessarily have the
#highest graduation rate.Also, rate > 100 is wrong

plot(college$Outstate, college$Grad.Rate)
# high tution fee coorelates to high grad rate
```


college$Grad.Rate
120
100
80
60
40
20
0
20
40
60
80
college$Top10perc



In [46]:
```
#highest accept rate?
acceptance_rate = college$Accept/college$Apps
college[which.min(acceptance_rate), ]
# Princeton univ has the higest acceptance rate
```

	Private	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Undergrad	O
Princeton University	Yes	13218	2042	1153	90	98	4540	146	

In [47]:
```
college[which.max(college$Top10perc), ]
#univ has most 10% of students from high scools - massachusetts
```

	Private	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Undergra
Massachusetts Institute of Technology	Yes	6411	2140	1078	96	99	4481	2

9

In [48]:
```
#9
Auto = read.csv("/Users/priyanka/desktop/Auto.csv", header = T, na.strings =
Auto = na.omit(Auto)
dim(Auto)
summary(Auto)
```

1. 392
2. 9

```
      mpg          cylinders      displacement     horsepower        weight
 Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0   Min.   :1613
 1st Qu.:17.00   1st Qu.:4.000   1st Qu.:105.0   1st Qu.: 75.0   1st Qu.:2225
 Median :22.75   Median :4.000   Median :151.0   Median : 93.5   Median :2804
 Mean   :23.45   Mean   :5.472   Mean   :194.4   Mean   :104.5   Mean   :2978
 3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:275.8   3rd Qu.:126.0   3rd Qu.:3615
 Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0   Max.   :5140

  acceleration        year           origin                      name
 Min.   : 8.00   Min.   :70.00   Min.   :1.000   amc matador       :  5
 1st Qu.:13.78   1st Qu.:73.00   1st Qu.:1.000   ford pinto        :  5
 Median :15.50   Median :76.00   Median :1.000   toyota corolla    :  5
 Mean   :15.54   Mean   :75.98   Mean   :1.577   amc gremlin       :  4
 3rd Qu.:17.02   3rd Qu.:79.00   3rd Qu.:2.000   amc hornet        :  4
 Max.   :24.80   Max.   :82.00   Max.   :3.000   chevrolet chevette:  4
                                                 (Other)           :365
```

9(a) and (b)

In [65]:
```
#9(a)

#Quantitative:mpg, cylinders(also qualita), displ, horspower, weight, accel,
#Qualitative:name,origin

#9(b) range of each quantitative predictor
range(Auto$mpg)
range(Auto$cylinders)
range(Auto$displacement)
range(Auto$horsepower)
range(Auto$weight)
range(Auto$acceleration)
range(Auto$year)
```

1. 9
2. 46.6

1. 3
2. 8

1. 68
2. 455

1. 46
2. 230

1. 1613
2. 5140

1. 8
2. 24.8

1. 70
2. 82

In [52]:
```
sapply(Auto[, 1:7], range)
```

mpg	cylinders	displacement	horsepower	weight	acceleration	year
9.0	3	68	46	1613	8.0	70
46.6	8	455	230	5140	24.8	82

9(c)

In [53]:
```
#9(c)

sapply(Auto[, 1:7], mean)
sapply(Auto[, 1:7], sd)
```

mpg	23.4459183673469
cylinders	5.4719387755102
displacement	194.411989795918
horsepower	104.469387755102
weight	2977.58418367347
acceleration	15.5413265306122
year	75.9795918367347

mpg	7.8050074865718
cylinders	1.70578324745278
displacement	104.644003908905
horsepower	38.4911599328285
weight	849.402560042949
acceleration	2.75886411918808
year	3.68373654357783

9(d)

In [54]:
```
#9(d)
new_Auto = Auto [-(10:85), ]
dim(new_Auto)
new_Auto[9, ] == Auto[9, ]
new_Auto[10, ] == Auto[86, ]
```

1. 316
2. 9

	mpg	cylinders	displacement	horsepower	weight	acceleration	year	origin	name
9	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE

	mpg	cylinders	displacement	horsepower	weight	acceleration	year	origin	name
87	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE

In [55]:
```
sapply(new_Auto[,1:7],range)
sapply(new_Auto[,1:7],mean)
sapply(new_Auto[,1:7],sd)
```

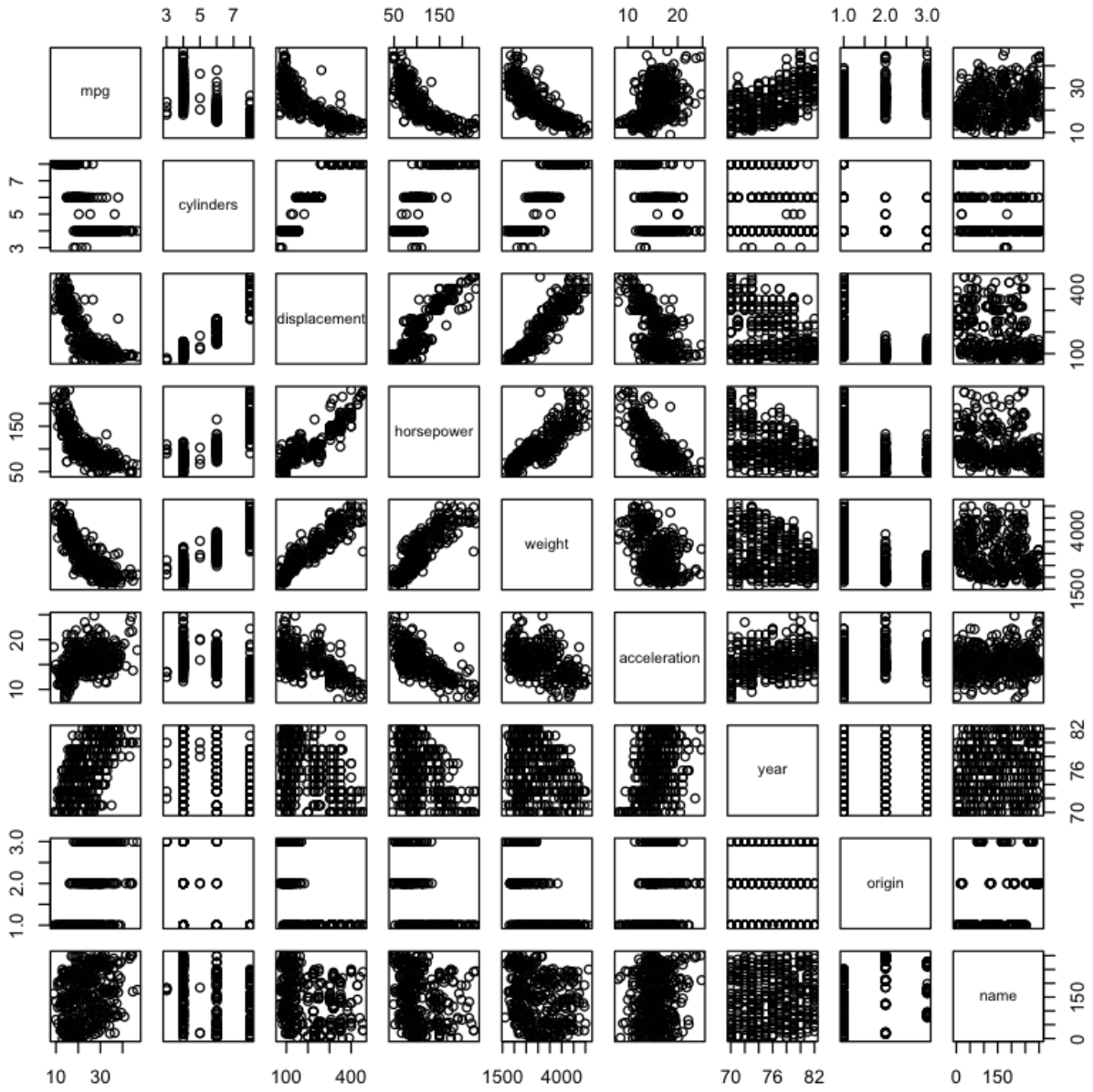
mpg	cylinders	displacement	horsepower	weight	acceleration	year
11.0	3	68	46	1649	8.5	70
46.6	8	455	230	4997	24.8	82

mpg	24.4044303797468
cylinders	5.37341772151899
displacement	187.240506329114
horsepower	100.721518987342
weight	2935.97151898734
acceleration	15.7268987341772
year	77.1455696202532

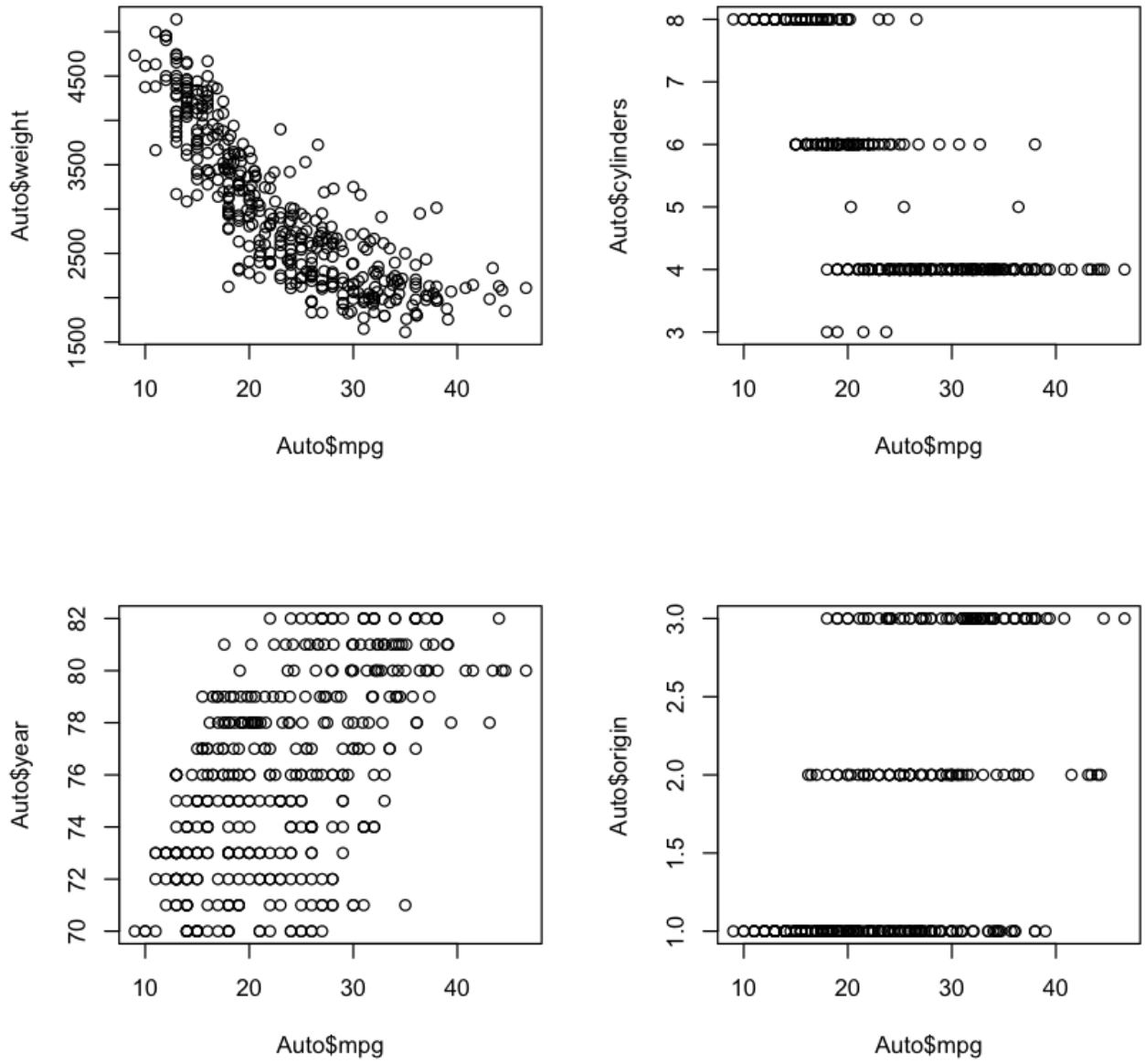
mpg	7.86728282443069
cylinders	1.65417865185607
displacement	99.6783672303628
horsepower	35.7088532738003
weight	811.30020815829
acceleration	2.69372071752036
year	3.10621690872137

9(e)

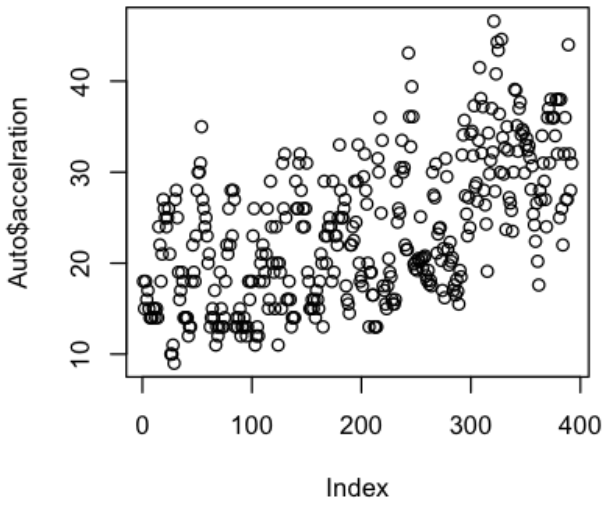
In [56]:
```
#9(e)
pairs(Auto)
```

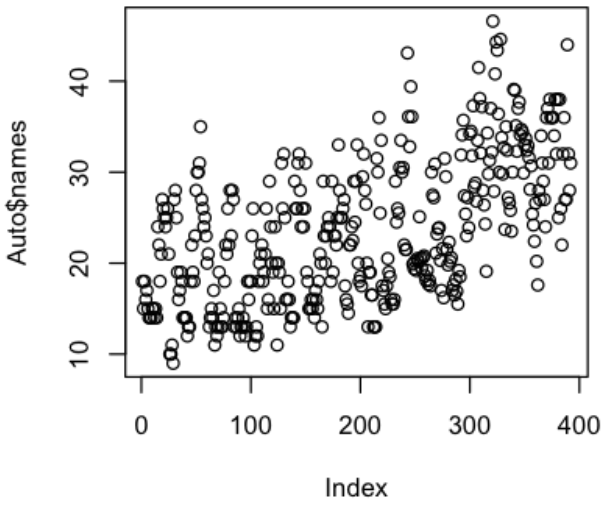


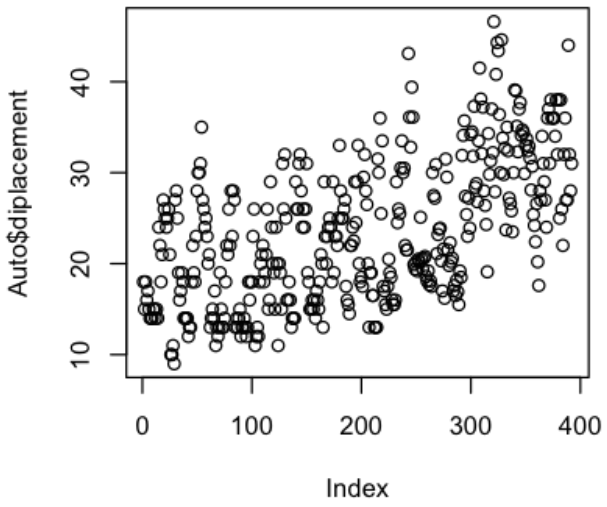
In [57]:
```
#(e)
par(mfrow=c(2,2))
plot(Auto$mpg, Auto$weight)   #heavier weights correlates with lower mpg
plot(Auto$mpg, Auto$cylinders) #more cylinder and less mpg
plot(Auto$mpg, Auto$year) #cars become more efficient with increasing years
plot(Auto$mpg, Auto$origin) # less correlates
```

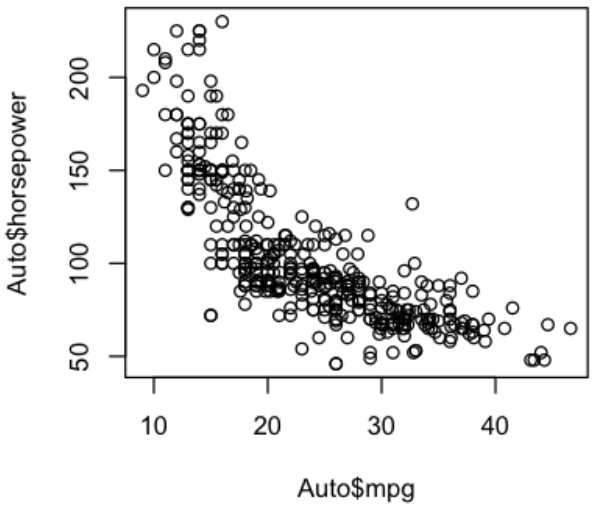


In [59]:
```
par(mfrow=c(2,2))
plot(Auto$mpg, Auto$accelration)
plot(Auto$mpg, Auto$names)
plot(Auto$mpg, Auto$diplacement)
plot(Auto$mpg, Auto$horsepower)
```









Observations

In [ ]:
```
#obvervations

# All predictors show some correlation with mpg. 'name' predictor is
#likely to result in overfitting the data and will not generalize well.
```

10(a)

In [60]:
```
#10
#10(a)
library(MASS)  #load in the Boston data set. load library(MASS)
Boston         #calling the object Boston
?Boston        #read about the dataset
dim(Boston)
```

crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	black	lstat	me
0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4.98	2
0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9.14	2
0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	3
0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	3
0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.90	5.33	3
0.02985	0.0	2.18	0	0.458	6.430	58.7	6.0622	3	222	18.7	394.12	5.21	2
0.08829	12.5	7.87	0	0.524	6.012	66.6	5.5605	5	311	15.2	395.60	12.43	2
0.14455	12.5	7.87	0	0.524	6.172	96.1	5.9505	5	311	15.2	396.90	19.15	2
0.21124	12.5	7.87	0	0.524	5.631	100.0	6.0821	5	311	15.2	386.63	29.93	1
0.17004	12.5	7.87	0	0.524	6.004	85.9	6.5921	5	311	15.2	386.71	17.10	1
0.22489	12.5	7.87	0	0.524	6.377	94.3	6.3467	5	311	15.2	392.52	20.45	1
0.11747	12.5	7.87	0	0.524	6.009	82.9	6.2267	5	311	15.2	396.90	13.27	1
0.09378	12.5	7.87	0	0.524	5.889	39.0	5.4509	5	311	15.2	390.50	15.71	2
0.62976	0.0	8.14	0	0.538	5.949	61.8	4.7075	4	307	21.0	396.90	8.26	2
0.63796	0.0	8.14	0	0.538	6.096	84.5	4.4619	4	307	21.0	380.02	10.26	1
0.62739	0.0	8.14	0	0.538	5.834	56.5	4.4986	4	307	21.0	395.62	8.47	1
1.05393	0.0	8.14	0	0.538	5.935	29.3	4.4986	4	307	21.0	386.85	6.58	2
0.78420	0.0	8.14	0	0.538	5.990	81.7	4.2579	4	307	21.0	386.75	14.67	1
0.80271	0.0	8.14	0	0.538	5.456	36.6	3.7965	4	307	21.0	288.99	11.69	2
0.72580	0.0	8.14	0	0.538	5.727	69.5	3.7965	4	307	21.0	390.95	11.28	1
1.25179	0.0	8.14	0	0.538	5.570	98.1	3.7979	4	307	21.0	376.57	21.02	1
0.85204	0.0	8.14	0	0.538	5.965	89.2	4.0123	4	307	21.0	392.53	13.83	1
1.23247	0.0	8.14	0	0.538	6.142	91.7	3.9769	4	307	21.0	396.90	18.72	1
0.98843	0.0	8.14	0	0.538	5.813	100.0	4.0952	4	307	21.0	394.54	19.88	1
0.75026	0.0	8.14	0	0.538	5.924	94.1	4.3996	4	307	21.0	394.33	16.30	1
0.84054	0.0	8.14	0	0.538	5.599	85.7	4.4546	4	307	21.0	303.42	16.51	1

0.67191	0.0	8.14	0	0.538	5.813	90.3	4.6820	4	307	21.0	376.88	14.81	1
0.95577	0.0	8.14	0	0.538	6.047	88.8	4.4534	4	307	21.0	306.38	17.28	1
0.77299	0.0	8.14	0	0.538	6.495	94.4	4.4547	4	307	21.0	387.94	12.80	1
1.00245	0.0	8.14	0	0.538	6.674	87.3	4.2390	4	307	21.0	380.23	11.98	2
...	...	...	...	...	...	...	...	...	...	...	...	...	
4.87141	0	18.10	0	0.614	6.484	93.6	2.3053	24	666	20.2	396.21	18.68	1
15.02340	0	18.10	0	0.614	5.304	97.3	2.1007	24	666	20.2	349.48	24.91	1
10.23300	0	18.10	0	0.614	6.185	96.7	2.1705	24	666	20.2	379.70	18.03	1
14.33370	0	18.10	0	0.614	6.229	88.0	1.9512	24	666	20.2	383.32	13.11	2
5.82401	0	18.10	0	0.532	6.242	64.7	3.4242	24	666	20.2	396.90	10.74	2
5.70818	0	18.10	0	0.532	6.750	74.9	3.3317	24	666	20.2	393.07	7.74	2
5.73116	0	18.10	0	0.532	7.061	77.0	3.4106	24	666	20.2	395.28	7.01	2
2.81838	0	18.10	0	0.532	5.762	40.3	4.0983	24	666	20.2	392.92	10.42	2
2.37857	0	18.10	0	0.583	5.871	41.9	3.7240	24	666	20.2	370.73	13.34	2
3.67367	0	18.10	0	0.583	6.312	51.9	3.9917	24	666	20.2	388.62	10.58	2
5.69175	0	18.10	0	0.583	6.114	79.8	3.5459	24	666	20.2	392.68	14.98	1
4.83567	0	18.10	0	0.583	5.905	53.2	3.1523	24	666	20.2	388.22	11.45	2
0.15086	0	27.74	0	0.609	5.454	92.7	1.8209	4	711	20.1	395.09	18.06	1
0.18337	0	27.74	0	0.609	5.414	98.3	1.7554	4	711	20.1	344.05	23.97	
0.20746	0	27.74	0	0.609	5.093	98.0	1.8226	4	711	20.1	318.43	29.68	
0.10574	0	27.74	0	0.609	5.983	98.8	1.8681	4	711	20.1	390.11	18.07	1
0.11132	0	27.74	0	0.609	5.983	83.5	2.1099	4	711	20.1	396.90	13.35	2
0.17331	0	9.69	0	0.585	5.707	54.0	2.3817	6	391	19.2	396.90	12.01	2
0.27957	0	9.69	0	0.585	5.926	42.6	2.3817	6	391	19.2	396.90	13.59	2
0.17899	0	9.69	0	0.585	5.670	28.8	2.7986	6	391	19.2	393.29	17.60	2
0.28960	0	9.69	0	0.585	5.390	72.9	2.7986	6	391	19.2	396.90	21.14	1
0.26838	0	9.69	0	0.585	5.794	70.6	2.8927	6	391	19.2	396.90	14.10	1
0.23912	0	9.69	0	0.585	6.019	65.3	2.4091	6	391	19.2	396.90	12.92	2
0.17783	0	9.69	0	0.585	5.569	73.5	2.3999	6	391	19.2	395.77	15.10	1
0.22438	0	9.69	0	0.585	6.027	79.7	2.4982	6	391	19.2	396.90	14.33	1
0.06263	0	11.93	0	0.573	6.593	69.1	2.4786	1	273	21.0	391.99	9.67	2

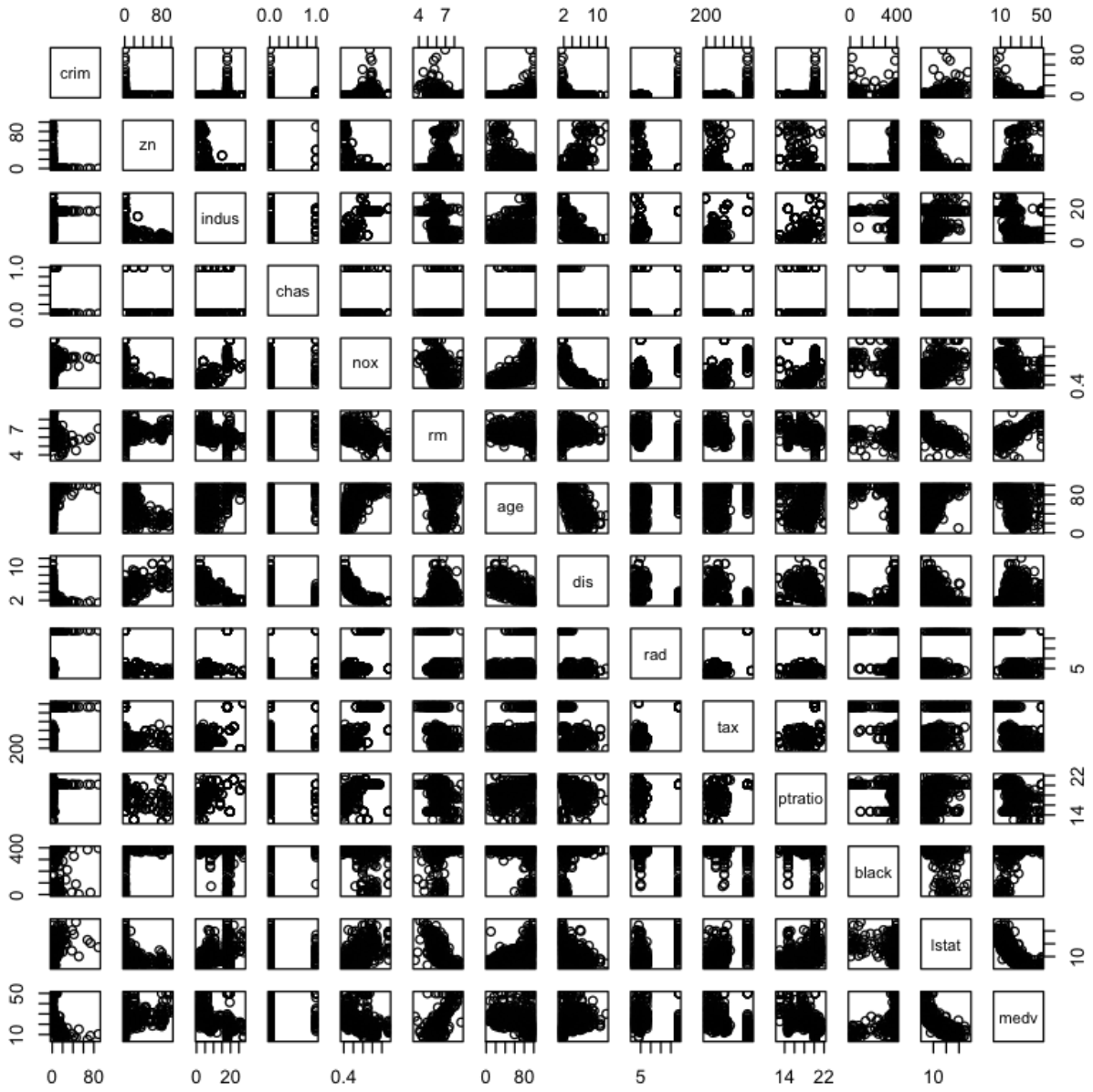
0.04527	0	11.93	0	0.573	6.120	76.7	2.2875	1	273	21.0	396.90	9.08	2(
0.06076	0	11.93	0	0.573	6.976	91.0	2.1675	1	273	21.0	396.90	5.64	2:
0.10959	0	11.93	0	0.573	6.794	89.3	2.3889	1	273	21.0	393.45	6.48	2:
0.04741	0	11.93	0	0.573	6.030	80.8	2.5050	1	273	21.0	396.90	7.88	1

1. 506
2. 14

In [ ]:
```
#data set includes 506 rows and 14 columns
#rows represent observations for each town
#columns represent features
```

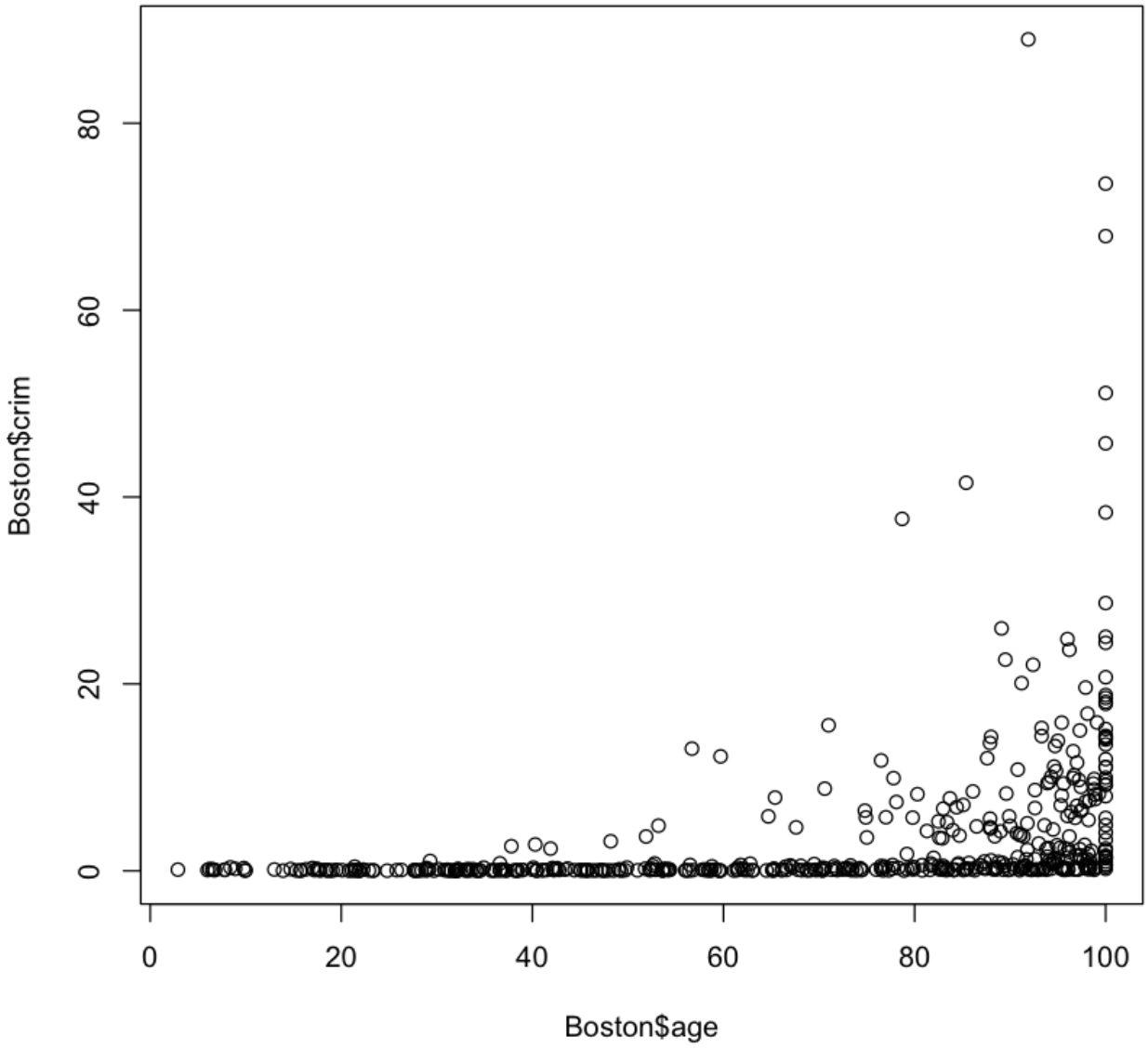
10(b)

In [61]:
```
#10(b)
pairs(Boston)
#from the pairs, observations are-
# crime rate correlates with: age, dis, rad, tax, ptratio
# zn correlates with: indus, nox, age, lstat
# indus with: age, dis
# nox with age, dis
# dis with lstat
# lstat with medv
```

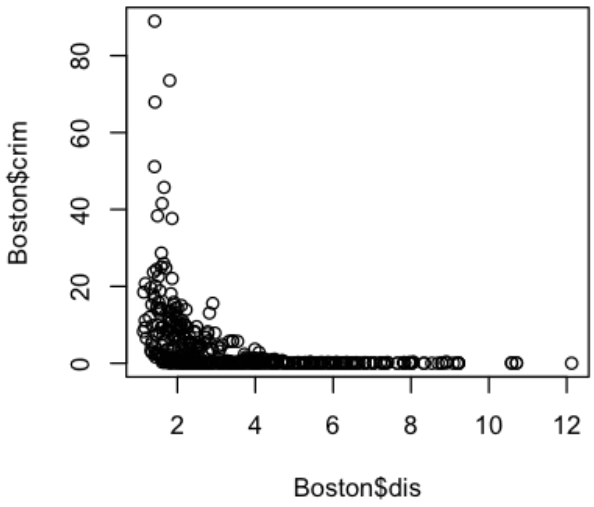


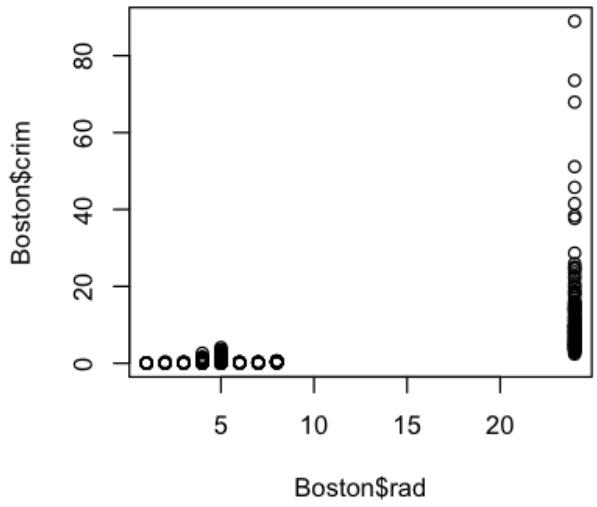
10(c)

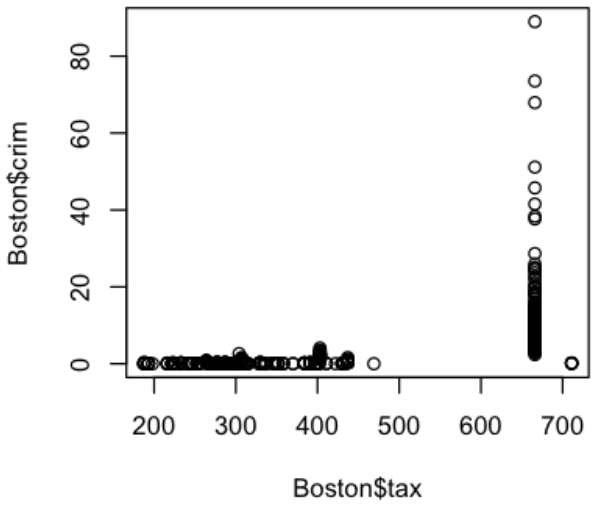
In [62]:
```
#10(c)
plot(Boston$age, Boston$crim)
#older homes and more the crime rate
```

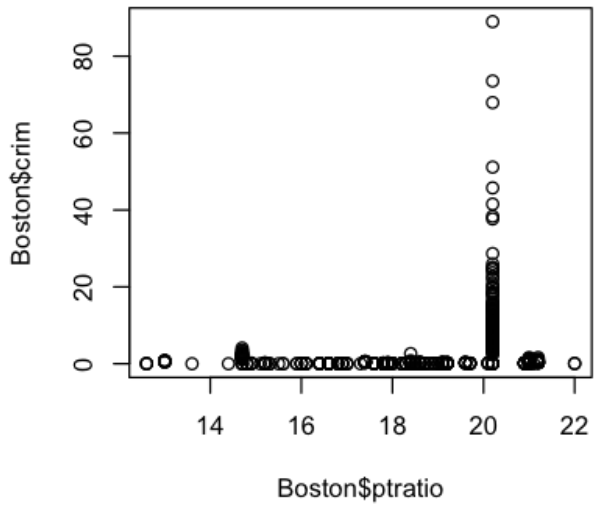

Boston$crim
0
20
40
60
80
0
20
40
60
80
100
Boston$age

In [63]:
```
#10(c)
par(mfrow=c(2,2))

plot(Boston$dis, Boston$crim)
#closer to work area, more crime rate

plot(Boston$rad, Boston$crim)
#as the index of accessibility to radial highways is higher and more crime ra

plot(Boston$tax, Boston$crim)
# when tax is high, crime rate is high

plot(Boston$ptratio, Boston$crim)
#higher pupil-teacher ratio by town and high crime rate
```



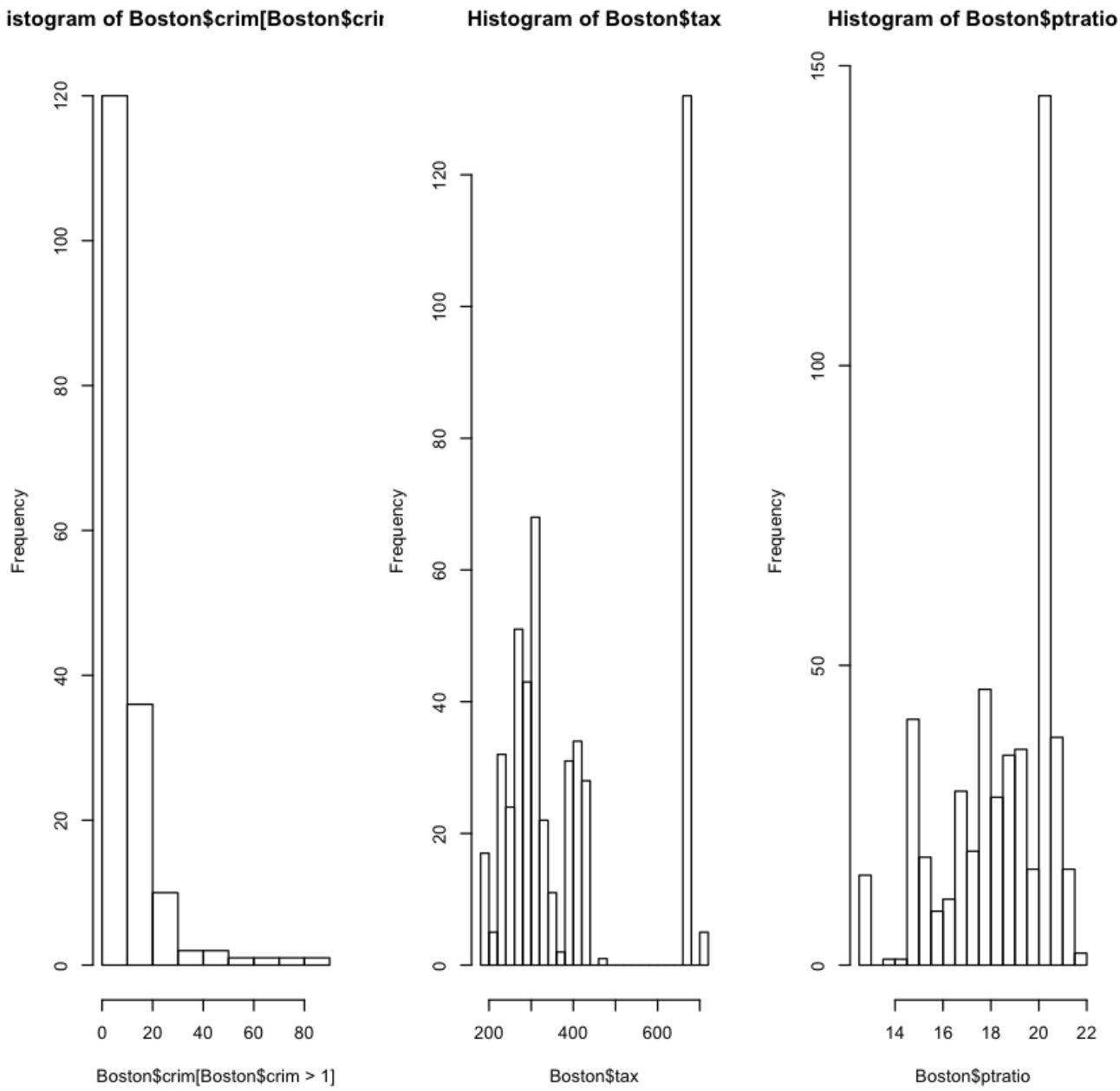






10(d)

In [64]:
```
#10(d)
par(mfrow=c(1,3))
hist(Boston$crim[Boston$crim>1], breaks=10)
#many cities have low crime rates, but there is more crime rate > 20 which is


hist(Boston$tax, breaks=20)
#there is a large divide between suburbs with low tax rates and a peak

hist(Boston$ptratio, breaks=20)
#a few have high ration , but not all high/
```



In [67]:
```
#10(e)
dim(subset(Boston, chas == 1))
# 35 suburbs bounds the Charles river
```

1. 35
2. 14

In [68]:
```
#10(f)
median(Boston$ptratio)
# 19.05
```

19.05

In [ ]:
```
#10(g)
t(subset(Boston, medv == min(Boston$medv)))
summary(Boston)
#crime rate is there, not very best place to live
```

In [69]:
```
#10(h)
dim(subset(Boston, rm>7))    #64

dim(subset(Boston, rm>8))    #13

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

1. 64
2. 14

1. 13
2. 14

```
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
 13.00   13.25   13.50   13.50   13.75   14.00 
     crim                zn             indus            chas        
 Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   Min.   :0.00000  
 1st Qu.: 0.08204   1st Qu.:  0.00   1st Qu.: 5.19   1st Qu.:0.00000  
 Median : 0.25651   Median :  0.00   Median : 9.69   Median :0.00000  
 Mean   : 3.61352   Mean   : 11.36   Mean   :11.14   Mean   :0.06917  
 3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10   3rd Qu.:0.00000  
 Max.   :88.97620   Max.   :100.00   Max.   :27.74   Max.   :1.00000  
      nox               rm              age              dis        
 Min.   :0.3850   Min.   :3.561   Min.   :  2.90   Min.   : 1.130  
 1st Qu.:0.4490   1st Qu.:5.886   1st Qu.: 45.02   1st Qu.: 2.100  
 Median :0.5380   Median :6.208   Median : 77.50   Median : 3.207  
 Mean   :0.5547   Mean   :6.285   Mean   : 68.57   Mean   : 3.795  
 3rd Qu.:0.6240   3rd Qu.:6.623   3rd Qu.: 94.08   3rd Qu.: 5.188  
 Max.   :0.8710   Max.   :8.780   Max.   :100.00   Max.   :12.127  
      rad              tax           ptratio          black       
 Min.   : 1.000   Min.   :187.0   Min.   :12.60   Min.   :  0.32  
 1st Qu.: 4.000   1st Qu.:279.0   1st Qu.:17.40   1st Qu.:375.38  
 Median : 5.000   Median :330.0   Median :19.05   Median :391.44  
 Mean   : 9.549   Mean   :408.2   Mean   :18.46   Mean   :356.67  
 3rd Qu.:24.000   3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:396.23  
 Max.   :24.000   Max.   :711.0   Max.   :22.00   Max.   :396.90  
     lstat            medv      
 Min.   : 1.73   Min.   : 5.00  
 1st Qu.: 6.95   1st Qu.:17.02  
 Median :11.36   Median :21.20  
 Mean   :12.65   Mean   :22.53  
 3rd Qu.:16.95   3rd Qu.:25.00  
 Max.   :37.97   Max.   :50.00  
```

In [ ]:
```
# comparitively lower crime (suburbs that average more than eight rooms per d
```