Homework Assignment 1

1. For each of parts (a) through (d), indicate whether we would generally expect the performance of a flexible statistical learning method to be better or worse than an inflexible method. Justify your answer.

(a) The sample size n is extremely large, and the number of predictors p is small.

Given the scenario where the sample size (n) is extremely large and the features or predictors (p) are small in size it seems that a more flexible statistical learning method is generally expected to perform better than an inflexible method. As we have a large amount of data available, a flexible method can better capture complex relationships and patterns within the data, approximating the true distribution of the data, leading to potentially more accurate predictions. But there can be another scenario in which if the relationship between features or predictors and the response variable is relatively simple then an overly flexible method could lead to overfitting the dataset and not generalizing to unseen data.

(b) The number of predictors p is extremely large, and the number of observations n is small.

Given the scenario where the sample size (n) is small and the features or predictors (p) is extremely large in size it seems that a more flexible statistical learning method is generally expected to perform worse than an inflexible method. As we have a large amount of predictors available, a flexible method can capture noise instead of capturing the underlying relationships and patterns within the data, leading to a risk of overfitting when using a flexible method. Inflexible methods might be better suited for such situations as they are less prone to overfitting and may provide more stable estimates. Inflexibility might be preferred in situations with high-dimensional data and limited observations to avoid overfitting as with the given scenario.

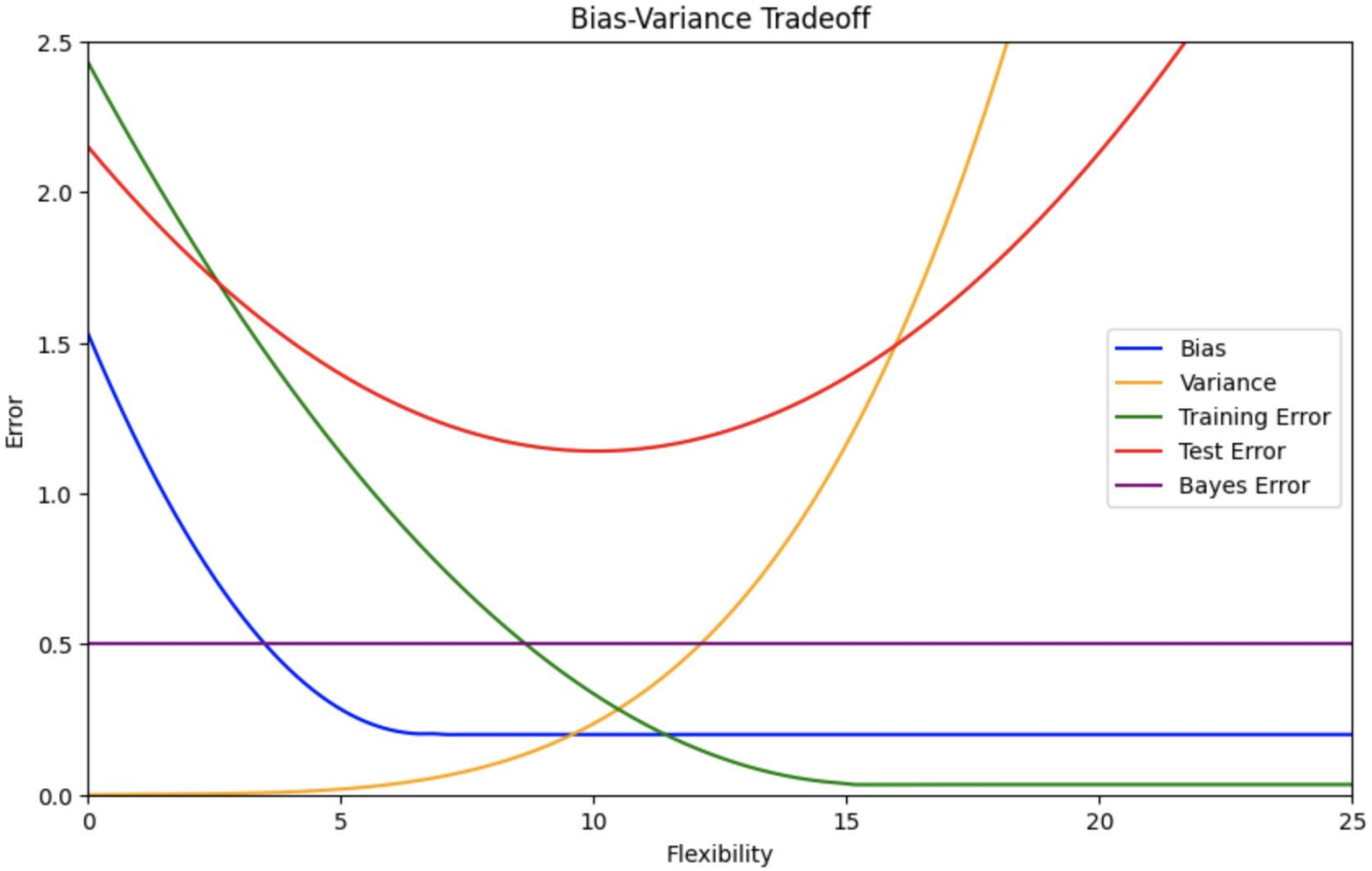
(c) The relationship between the predictors and response is highly non-linear.

Given the scenario that the relationship between the predictors and response is highly non-linear, a flexible statistical learning method is generally expected to perform better than an inflexible method. As we have a high non-linearity in the data, a flexible method can better capture complex non-linear relationships and underlying patterns within the data, leading to improved predictive performance compared to methods that assume linear relationships.

(d) The variance of the error terms, i.e. $\sigma 2 = \text{Var}(\varepsilon)$, is extremely high.

Given the scenario that the variance of the error terms, i.e. $\sigma 2 = \text{Var}(\varepsilon)$, is extremely high, the performance of both flexible and inflexible methods may suffer. High variance in the error terms indicates that there is a large amount of noise in the data, which can make it challenging for any method to accurately model the underlying patterns. Flexible models might likely overfit on the dataset. It is better to treat the issue of high variance first before proceeding to selection of any statistical learning method to tackle the problem. We can try techniques like data preprocessing, feature selection, feature engineering using feature interaction, or regularization, rather than solely relying on the flexibility of the modeling approach.

2. We now revisit the bias-variance decomposition.
(a) Provide a sketch of typical (squared) bias, variance, training error, test error, and Bayes (or irreducible) error curves, on a single plot, as we go from less flexible statistical learning methods towards more flexible approaches. The x-axis should represent the amount of flexibility in the method, and the y-axis should represent the values for each curve. There should be five curves. Make sure to label each one.



(b) Explain why each of the five curves has the shape displayed in part (a).

1) Squared Bias: Initially high as less flexible methods may not capture the true underlying relationship well. As flexibility increases, bias decreases since more complex models can better fit the data, until a point where increasing flexibility may lead to overfitting, causing bias to stabilize.
2) Variance: Initially low as less flexible models have simpler structures and are less sensitive to fluctuations in the training data. However, as flexibility increases, variance increases because more complex models are sensitive to fluctuations in the training data, resulting in greater variability in predictions. It stabilizes after a certain point because excessively flexible models start fitting the noise in the data, leading to a plateau in variance.
3) Training Error: Decreases monotonically with increasing flexibility as more flexible models can better fit the training data.
4) Test Error: Initially decreases as more flexible models better capture the underlying patterns, but after a point, it starts to increase due to overfitting. Overly flexible models start capturing noise in the training data, leading to poor generalization performance on unseen/test data.
5) Bayes (Irreducible) Error: Represents the inherent noise and complexity in the underlying true relationship between predictors and the response variable. It remains constant as it is independent of the modeling approach and represents the best achievable error rate regardless of the method used. It can also be looked as the noise present in the data that cannot be reduced by any model, no matter how complex. This error rate serves as a theoretical lower bound for the error rate of any classifier. Given a dataset, even if it were possible to form a perfect estimate for f, so that our estimated response took the form $\hat{Y} = f(X)$, our prediction would still have some error in it as this is because Y is also a function of ε, which, by definition, cannot be predicted using X. Therefore, variability associated with ε also affects the accuracy of our predictions. This is known as the irreducible error, because no matter how well we estimate f, we cannot reduce the error introduced by ε.

3. Describe the differences between a parametric and a non-parametric statistical learning approach. What are the advantages of a parametric approach to regression or classification (as opposed to a nonparametric approach)? What are its disadvantages?

- ✓ In the case of parametric statistical learning approach, we make an assumption about the functional form, or shape, of f. Since we are making an assumption about the functional form it reduces the problem of estimating f down to one of estimating a set of parameters. Assuming a parametric form for f simplifies the problem of estimating f because it is generally much easier to estimate a set of parameters, such as $\beta 0, \beta 1, \ldots, \beta p$ in the linear model, than it is to fit an entirely arbitrary function f. Parametric models often have easily interpretable coefficients or parameters, making it easier to understand the relationship between predictors and the response variable. The potential disadvantage of a parametric statistical learning approach is that the model we choose will usually not match the true unknown form of f. If the chosen model is too far from the true f, then our estimate will be poor. We can try to address this problem by choosing flexible models that can fit many different possible functional forms for f. But in general, fitting a more flexible model requires estimating a greater number of parameters. These more complex models can lead to a overfitting the data, which essentially means they follow the errors, or noise, too closely and it is an undesirable situation because the fit obtained will not yield accurate estimates of the response on new observations that were not part of the original training data set. Additionally, since they require fewer parameters to estimate, it leads to faster training times and less computational resources. Some examples include linear regression, logistic regression, naive bayes, and linear discriminant analysis.

- ✓ In the case of non-parametric statistical learning approach, we do not make explicit assumptions about the the functional form, or shape, of f. Instead they seek an estimate of f that gets as close to the data points as possible without being too rough or wiggly. Such approaches can have a major advantage over parametric approaches by avoiding the assumption of a particular functional form for f, they have the potential to accurately fit a wider range of possible shapes for f. Any parametric approach brings with it the possibility that the functional form used to estimate f is very different from the true f, in which case the resulting model will not fit the data well. In contrast, non-parametric approaches completely avoid this danger, since essentially no assumption about the form of f is made. But non-parametric approaches do suffer from a major disadvantage since they do not reduce the problem of estimating f to a small number of parameters, a very large number of observations (typically more data is needed compared to parametric method) is required in order to obtain an accurate estimate for f. Non-parametric models can be more challenging to interpret, particularly when they involve complex interactions or decision boundaries and are more susceptible to overfitting, especially with small datasets or when the model complexity is not properly controlled. Some examples include k-nearest neighbors (KNN), support vector machines (SVM) with radial basis function (RBF) kernels, decision trees and neural networks.

4. This exercise relates to the College data set, which can be found in the file College.csv. It contains a number of variables for 777 different universities and colleges in the US. The variables are

• Private : Public/private indicator
• Apps : Number of applications received
• Accept : Number of applicants accepted
• Enroll : Number of new students enrolled
• Top10perc : New students from top 10 % of high school class
• Top25perc : New students from top 25 % of high school class
• F.Undergrad : Number of full-time undergraduates
• P.Undergrad : Number of part-time undergraduates
• Outstate : Out-of-state tuition
• Room.Board : Room and board costs
• Books : Estimated book costs
• Personal : Estimated personal spending
• PhD : Percent of faculty with Ph.D.'s
• Terminal : Percent of faculty with terminal degree
• S.F.Ratio : Student/faculty ratio
• perc.alumni : Percent of alumni who donate
• Expend : Instructional expenditure per student
• Grad.Rate : Graduation rate

Before reading the data into R, it can be viewed in Excel or a text editor.

(a) 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")

(b) 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]
> fix(college)

R Data Editor

Copy Paste Quit

	row.names	X	Private	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Undergrad	Outstate	Room.Board	Books	Personal	PhD	Terminal
1	Abilene Christian University	Abilene Christian University	Yes	1660	1232	721	23	52	2885	537	7440	3300	450	2200	70	78
2	Adelphi University	Adelphi University	Yes	2186	1924	512	16	29	2683	1227	12280	6450	750	1500	29	30
3	Adrian College	Adrian College	Yes	1428	1097	336	22	50	1036	99	11250	3750	400	1165	53	66
4	Agnes Scott College	Agnes Scott College	Yes	417	349	137	60	89	510	63	12960	5450	450	875	92	97
5	Alaska Pacific University	Alaska Pacific University	Yes	193	146	55	16	44	249	869	7560	4120	800	1500	76	72
6	Albertson College	Albertson College	Yes	587	479	158	38	62	678	41	13500	3335	500	675	67	73
7	Albertus Magnus College	Albertus Magnus College	Yes	353	340	103	17	45	416	230	13290	5720	500	1500	90	93
8	Albion College	Albion College	Yes	1899	1720	489	37	68	1594	32	13868	4826	450	850	89	100
9	Albright College	Albright College	Yes	1038	839	227	30	63	973	306	15595	4400	300	500	79	84
10	Alderson-Broaddus College	Alderson-Broaddus College	Yes	582	498	172	21	44	799	78	10468	3380	660	1800	40	41
11	Alfred University	Alfred University	Yes	1732	1425	472	37	75	1830	110	16548	5406	500	600	82	88
12	Allegheny College	Allegheny College	Yes	2652	1900	484	44	77	1707	44	17080	4440	400	600	73	91
13	Allentown Coll. of St. Francis de Sales	Allentown Coll. of St. Francis de Sales	Yes	1179	780	290	38	64	1130	638	9690	4785	600	1000	60	84
14	Alma College	Alma College	Yes	1267	1080	385	44	73	1306	28	12572	4552	400	400	79	87
15	Alverno College	Alverno College	Yes	494	313	157	23	46	1317	1235	8352	3640	650	2449	36	69
16	American International College	American International College	Yes	1420	1093	220	9	22	1018	287	8700	4780	450	1400	78	84
17	Amherst College	Amherst College	Yes	4302	992	418	83	96	1593	5	19760	5300	660	1598	93	98
18	Anderson University	Anderson University	Yes	1216	908	423	19	40	1819	281	10100	3520	550	1100	48	61
19	Andrews University	Andrews University	Yes	1130	704	322	14	23	1586	326	9996	3090	900	1320	62	66
20	Angelo State University	Angelo State University	No	3540	2001	1016	24	54	4190	1512	5130	3592	500	2000	60	62
21	Antioch University	Antioch University	Yes	713	661	252	25	44	712	23	15476	3336	400	1100	69	82
22	Appalachian State University	Appalachian State University	No	7313	4664	1910	20	63	9940	1035	6806	2540	96	2000	83	96
23	Aquinas College	Aquinas College	Yes	619	516	219	20	51	1251	767	11208	4124	350	1615	55	65
24	Arizona State University Main campus	Arizona State University Main campus	No	12809	10308	3761	24	49	22593	7585	7434	4850	700	2100	88	93
25	Arkansas College (Lyon College)	Arkansas College (Lyon College)	Yes	708	334	166	46	74	530	182	8644	3922	500	800	79	88
26	Arkansas Tech University	Arkansas Tech University	No	1734	1729	951	12	52	3602	939	3460	2650	450	1000	57	60
27	Assumption College	Assumption College	Yes	2135	1700	491	23	59	1708	689	12000	5920	500	500	93	93
28	Auburn University-Main Campus	Auburn University-Main Campus	No	7548	6791	3070	25	57	16262	1716	6300	3933	600	1908	85	91
29	Augsburg College	Augsburg College	Yes	662	513	257	12	30	2074	726	11902	4372	540	950	65	65
30	Augustana College IL	Augustana College IL	Yes	1879	1658	497	36	69	1950	38	13353	4173	540	821	78	83
31	Augustana College	Augustana College	Yes	761	725	306	21	58	1337	300	10990	3244	600	1021	66	70
32	Austin College	Austin College	Yes	948	798	295	42	74	1120	15	11280	4342	400	1150	81	95
33	Averett College	Averett College	Yes	627	556	172	16	40	777	538	9925	4135	750	1350	59	67
34	Baker University	Baker University	Yes	602	483	206	21	47	958	466	8620	4100	400	2250	58	68
35	Baldwin-Wallace College	Baldwin-Wallace College	Yes	1690	1366	662	30	61	2718	1460	10995	4410	1000	1000	68	74
36	Barat College	Barat College	Yes	261	192	111	15	36	453	266	9690	4300	500	500	57	77
37	Bard College	Bard College	Yes	1910	838	285	50	85	1004	15	19264	6206	750	750	98	98
38	Barnard College	Barnard College	Yes	2496	1402	531	53	96	2121	69	17926	8124	600	850	83	93

You should see that there is now a row.names column with the name of each university recorded. This means that R has given each row a name corresponding to the appropriate university. R will not try to perform calculations on the row names. However, we still need to eliminate the first column in the data where the names are stored. Try

> college=college[,-1]
> fix(college)

Now you should see that the first data column is Private. Note that another column labeled row.names now appears before the Private column. However, this is not a data column but rather the name that R is giving to each row.

R Data Editor

Copy Paste Quit

	row.names	Private	Apps	Accept	Enroll	Top10perc	Top25perc	F.Undergrad	P.Undergrad	Outstate	Room.Board	Books	Personal	PhD	Terminal	S.F.Ratio	perc.alumni	Expend	Grad.Rate
1	Abilene Christian Universitys	Yes	1660	1232	721	23	52	2885	537	7440	3300	450	2200	70	78	18.1	12	7041	60
2	Adelphi University	Yes	2186	1924	512	16	29	2683	1227	12280	6450	750	1500	29	30	12.2	16	10527	56
3	Adrian College	Yes	1428	1097	336	22	50	1036	99	11250	3750	400	1165	53	66	12.9	30	8735	54
4	Agnes Scott College	Yes	417	349	137	60	89	510	63	12960	5450	450	875	92	97	7.7	37	19016	59
5	Alaska Pacific University	Yes	193	146	55	16	44	249	869	7560	4120	800	1500	76	72	11.9	2	10922	15
6	Albertson College	Yes	587	479	158	38	62	678	41	13500	3335	500	675	67	73	9.4	11	9727	55
7	Albertus Magnus College	Yes	353	340	103	17	45	416	230	13290	5720	500	1500	90	93	11.5	26	8861	63
8	Albion College	Yes	1899	1720	489	37	68	1594	32	13868	4826	450	850	89	100	13.7	37	11487	73
9	Albright College	Yes	1038	839	227	30	63	973	306	15595	4400	300	500	79	84	11.3	23	11644	80
10	Alderson-Broaddus College	Yes	582	498	172	21	44	799	78	10468	3380	660	1800	40	41	11.5	15	8991	52
11	Alfred University	Yes	1732	1425	472	37	75	1830	110	16548	5406	500	600	82	88	11.3	31	10932	73
12	Allegheny College	Yes	2652	1900	484	44	77	1707	44	17080	4440	400	600	73	91	9.9	41	11711	76
13	Allentown Coll. of St. Francis de Sales	Yes	1179	780	290	38	64	1130	638	9690	4785	600	1000	60	84	13.3	21	7940	74
14	Alma College	Yes	1267	1080	385	44	73	1306	28	12572	4552	400	400	79	87	15.3	32	9305	68
15	Alverno College	Yes	494	313	157	23	46	1317	1235	8352	3640	650	2449	36	69	11.1	26	8127	55
16	American International College	Yes	1420	1093	220	9	22	1018	287	8700	4780	450	1400	78	84	14.7	19	7355	69
17	Amherst College	Yes	4302	992	418	83	96	1593	5	19760	5300	660	1598	93	98	8.4	63	21424	100
18	Anderson University	Yes	1216	908	423	19	40	1819	281	10100	3520	550	1100	48	61	12.1	14	7994	59
19	Andrews University	Yes	1130	704	322	14	23	1586	326	9996	3090	900	1320	62	66	11.5	18	10908	46
20	Angelo State University	No	3540	2001	1016	24	54	4190	1512	5130	3592	500	2000	60	62	23.1	5	4010	34
21	Antioch University	Yes	713	661	252	25	44	712	23	15476	3336	400	1100	69	82	11.3	35	42926	48
22	Appalachian State University	No	7313	4664	1910	20	63	9940	1035	6806	2540	96	2000	83	96	18.3	14	5854	70
23	Aquinas College	Yes	619	516	219	20	51	1251	767	11208	4124	350	1615	55	65	12.7	25	6584	65
24	Arizona State University Main campus	No	12809	10308	3761	24	49	22593	7585	7434	4850	700	2100	88	93	18.9	5	4602	48
25	Arkansas College (Lyon College)	Yes	708	334	166	46	74	530	182	8644	3922	500	800	79	88	12.6	24	14579	54
26	Arkansas Tech University	No	1734	1729	951	12	52	3602	939	3460	2650	450	1000	57	60	19.6	5	4739	48
27	Assumption College	Yes	2135	1700	491	23	59	1708	689	12000	5920	500	500	93	93	13.8	30	7100	88
28	Auburn University-Main Campus	No	7548	6791	3070	25	57	16262	1716	6300	3933	600	1908	85	91	16.7	18	6642	69
29	Augsburg College	Yes	662	513	257	12	30	2074	726	11902	4372	540	950	65	65	12.8	31	7836	58
30	Augustana College IL	Yes	1879	1658	497	36	69	1950	38	13353	4173	540	821	78	83	12.7	40	9220	71
31	Augustana College	Yes	761	725	306	21	58	1337	300	10990	3244	600	1021	66	70	10.4	30	6871	69
32	Austin College	Yes	948	798	295	42	74	1120	15	11280	4342	400	1150	81	95	13	33	11361	71
33	Averett College	Yes	627	556	172	16	40	777	538	9925	4135	750	1350	59	67	22.4	11	6523	48
34	Baker University	Yes	602	483	206	21	47	958	466	8620	4100	400	2250	58	68	11	21	6136	65
35	Baldwin-Wallace College	Yes	1690	1366	662	30	61	2718	1460	10995	4410	1000	1000	68	74	17.6	20	8086	85
36	Barat College	Yes	261	192	111	15	36	453	266	9690	4300	500	500	57	77	9.7	35	9337	71
37	Bard College	Yes	1910	838	285	50	85	1004	15	19264	6206	750	750	98	98	10.4	30	13894	79
38	Barnard College	Yes	2496	1402	531	53	95	2121	69	17926	8124	600	850	83	93	10.3	33	12580	91

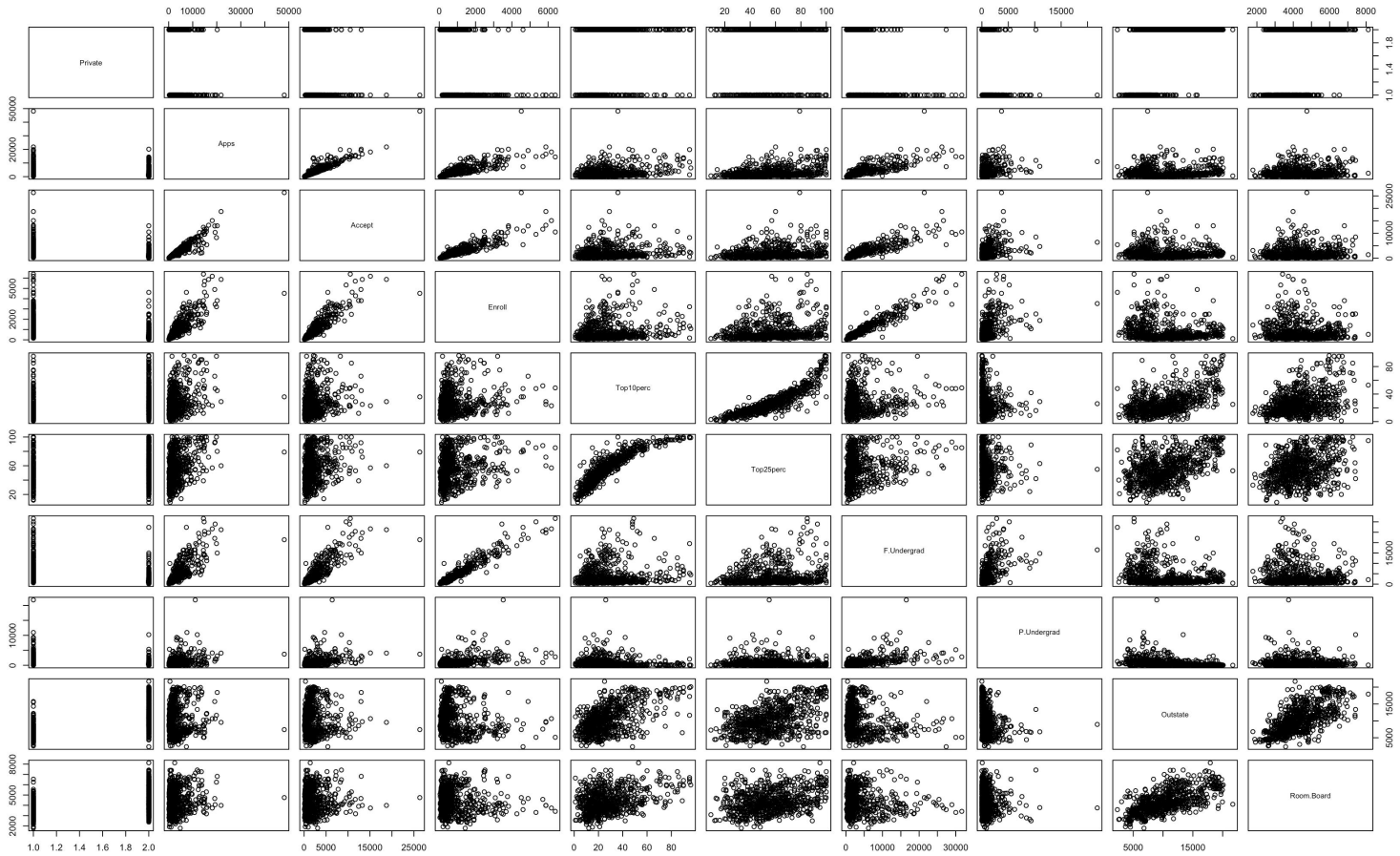
(c)
i. Use the summary() function to produce a numerical summary of the variables in the data set.

> summary(college)

```
> summary(college)
   Private             Apps           Accept          Enroll       Top10perc       Top25perc      F.Undergrad     P.Undergrad
 Length:777         Min.   :   81   Min.   :   72   Min.   :  35   Min.   : 1.00   Min.   :  9.0   Min.   :  139   Min.   :    1.0
 Class :character   1st Qu.:  776   1st Qu.:  604   1st Qu.: 242   1st Qu.:15.00   1st Qu.: 41.0   1st Qu.:  992   1st Qu.:   95.0
 Mode  :character   Median : 1558   Median : 1110   Median : 434   Median :23.00   Median : 54.0   Median : 1707   Median :  353.0
                    Mean   : 3002   Mean   : 2019   Mean   : 780   Mean   :27.56   Mean   : 55.8   Mean   : 3700   Mean   :  855.3
                    3rd Qu.: 3624   3rd Qu.: 2424   3rd Qu.: 902   3rd Qu.:35.00   3rd Qu.: 69.0   3rd Qu.: 4005   3rd Qu.:  967.0
                    Max.   :48094   Max.   :26330   Max.   :6392   Max.   :96.00   Max.   :100.0   Max.   :31643   Max.   :21836.0
    Outstate       Room.Board       Books           Personal         PhD            Terminal       S.F.Ratio      perc.alumni
 Min.   : 2340   Min.   :1780   Min.   :  96.0   Min.   : 250   Min.   :  8.00   Min.   : 24.0   Min.   : 2.50   Min.   : 0.00
 1st Qu.: 7320   1st Qu.:3597   1st Qu.: 470.0   1st Qu.: 850   1st Qu.: 62.00   1st Qu.: 71.0   1st Qu.:11.50   1st Qu.:13.00
 Median : 9990   Median :4200   Median : 500.0   Median :1200   Median : 75.00   Median : 82.0   Median :13.60   Median :21.00
 Mean   :10441   Mean   :4358   Mean   : 549.4   Mean   :1341   Mean   : 72.66   Mean   : 79.7   Mean   :14.09   Mean   :22.74
 3rd Qu.:12925   3rd Qu.:5050   3rd Qu.: 600.0   3rd Qu.:1700   3rd Qu.: 85.00   3rd Qu.: 92.0   3rd Qu.:16.50   3rd Qu.:31.00
 Max.   :21700   Max.   :8124   Max.   :2340.0   Max.   :6800   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
```

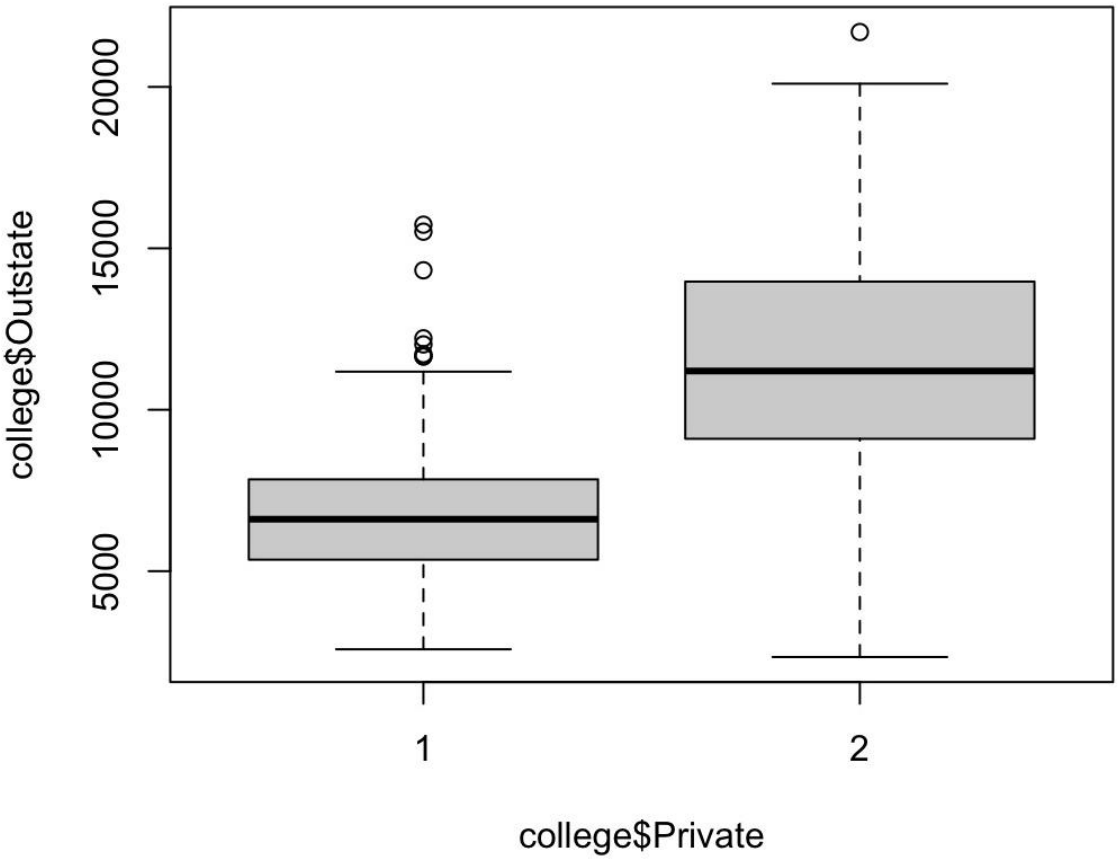
ii. Use the pairs() function to produce a scatterplot matrix of the first ten columns or variables of the data. Recall that you can reference the first ten columns of a matrix A using A[,1:10].

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



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

> boxplot(college$Outstate~college$Private)



iv. 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 %.

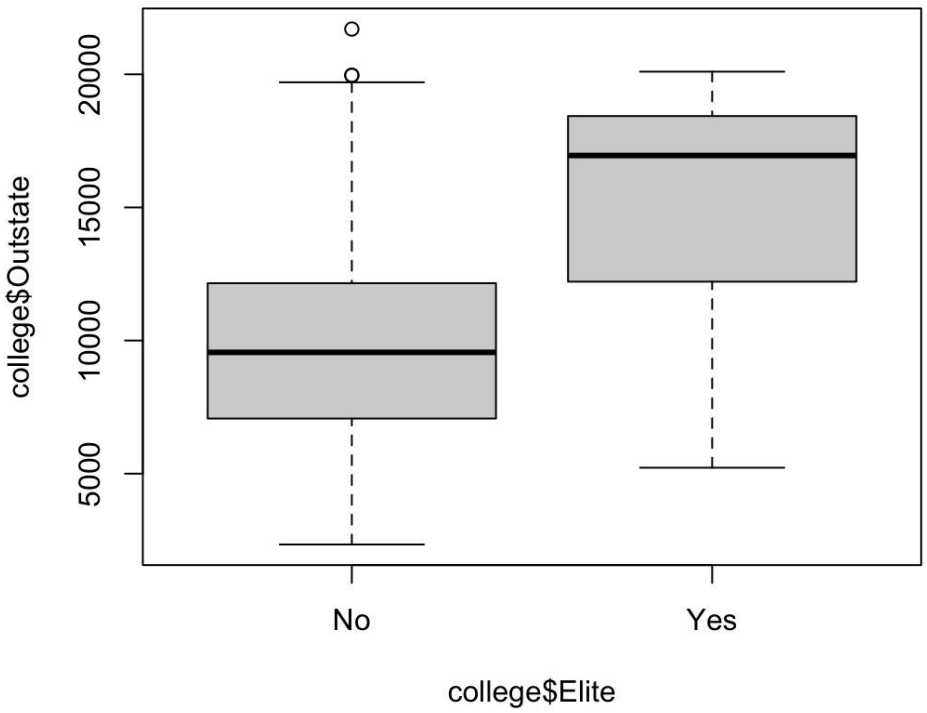
> Elite=rep("No",nrow(college))
> Elite[college$Top10perc >50]="Yes"
> Elite=as.factor(Elite)
> college=data.frame(college ,Elite)

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)

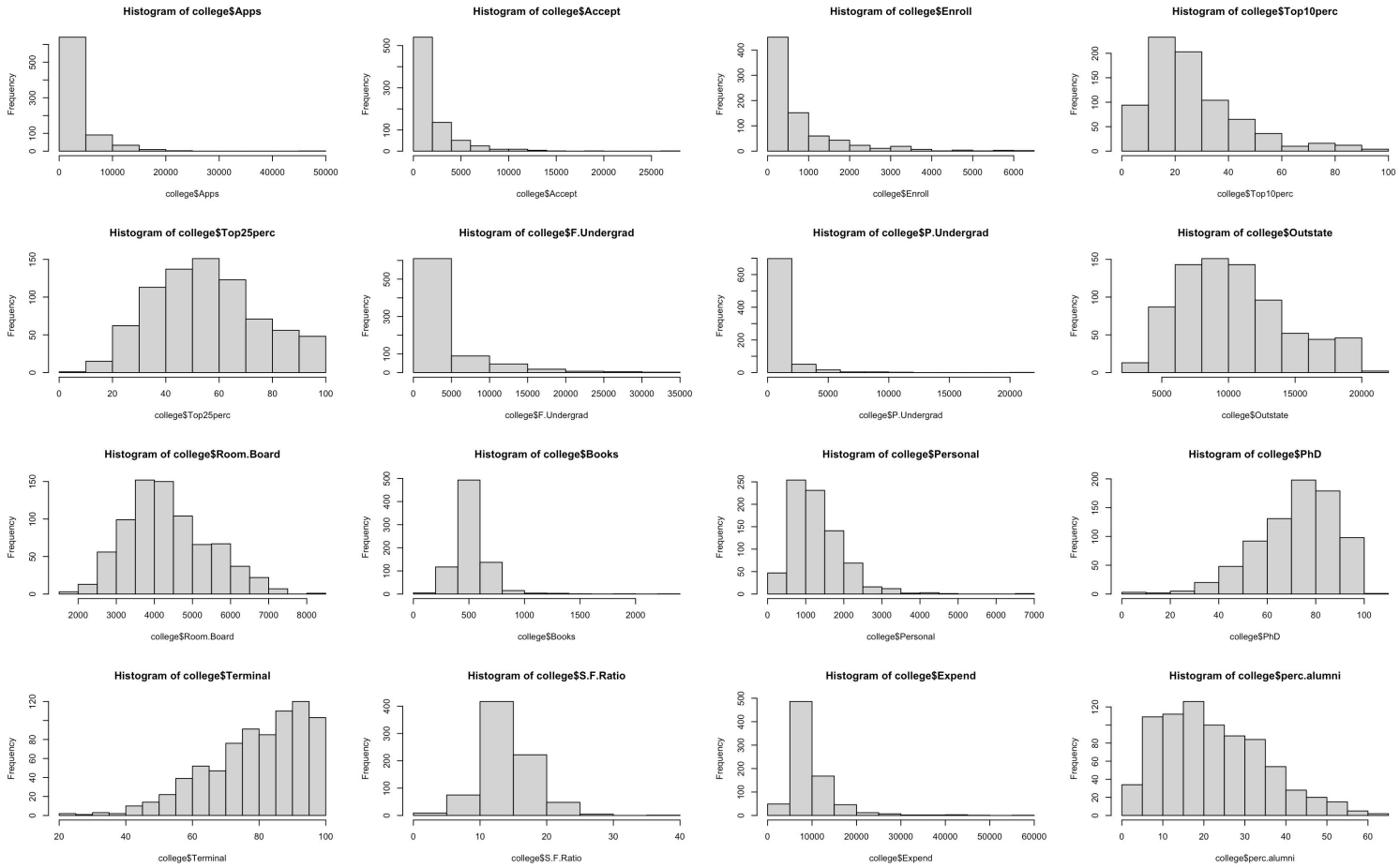
```
> summary(college$Elite)
 No Yes
699  78
```

> boxplot(college$Outstate~college$Elite)



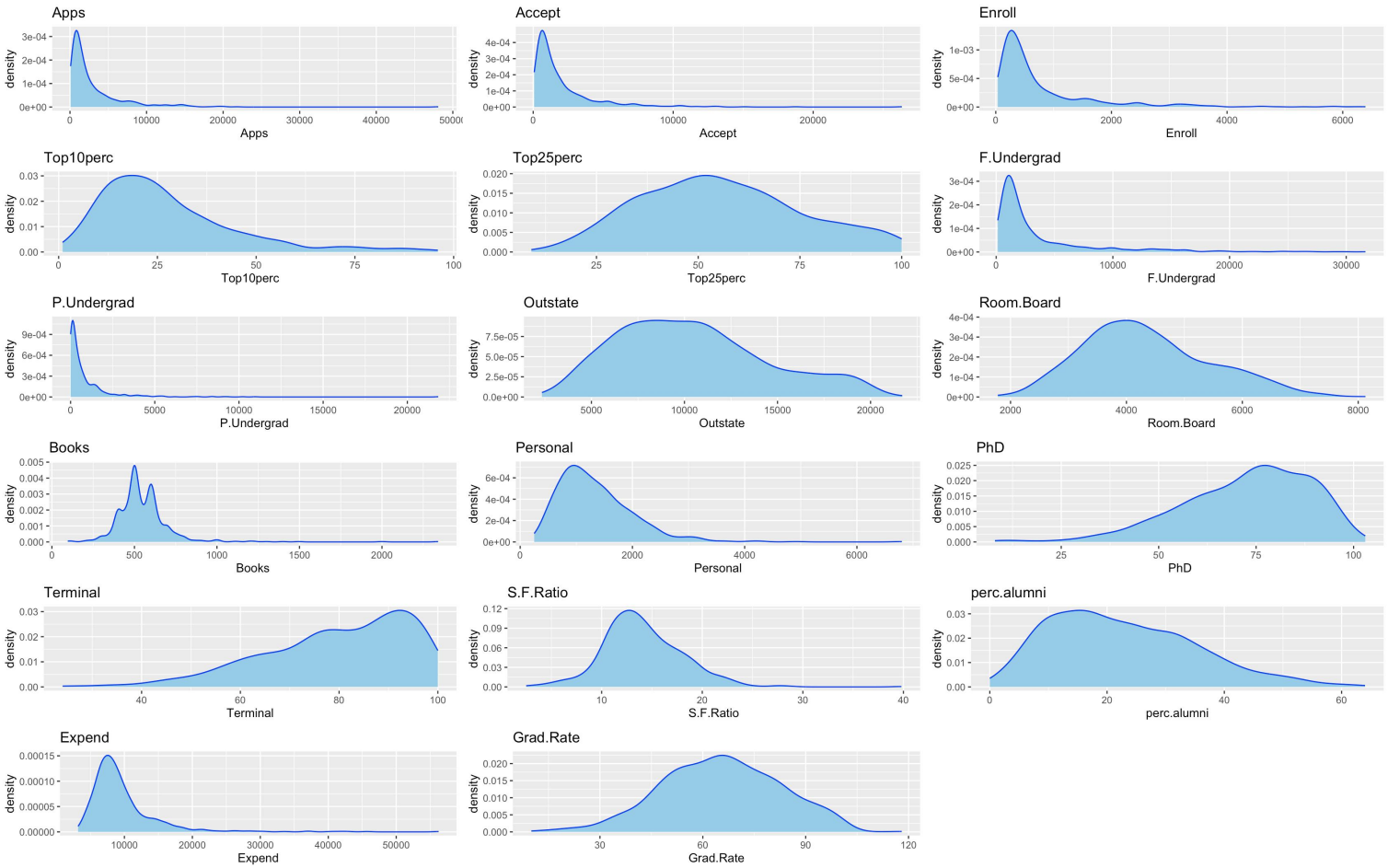
v. Use the hist() function to produce some histograms with differing numbers of bins for a few of the quantitative variables. You may find the command par(mfrow=c(2,2)) useful: it will divide the print window into four regions so that four plots can be made simultaneously. Modifying the arguments to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

```
> par(mfrow=c(4,4))
> hist(college$Apps)
> hist(college$Accept)
> hist(college$Enroll)
> hist(college$Top10perc)
> hist(college$Top25perc)
> hist(college$F.Undergrad)
> hist(college$P.Undergrad)
> hist(college$Outstate)
> hist(college$Room.Board)
> hist(college$Books)
> hist(college$Personal)
> hist(college$PhD)
> hist(college$Terminal)
> hist(college$S.F.Ratio)
> hist(college$Expend)
> hist(college$perc.alumni)
```



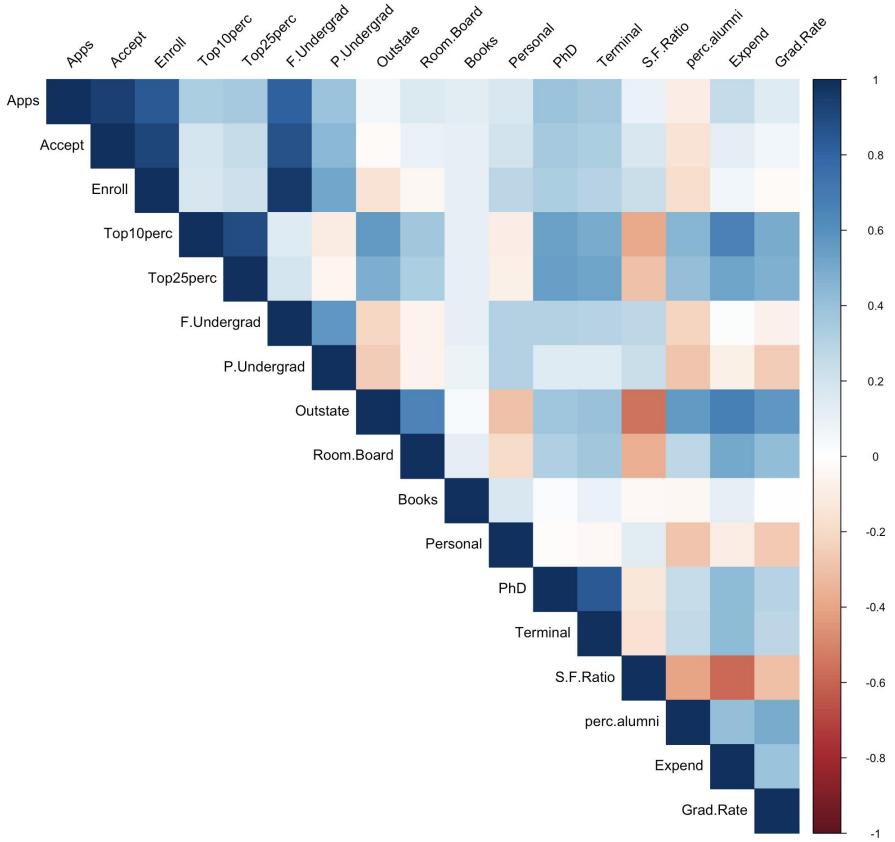
vi. Continue exploring the data, and provide a brief summary of what you discover.

```
> library(ggplot2)
> library(cowplot)

> num_vars<-
c("Apps","Accept","Enroll","Top10perc","Top25perc","F.Undergrad","P.Undergrad","Outstate","Room.
Board","Books","Personal","PhD","Terminal","S.F.Ratio","perc.alumni","Expend","Grad.Rate")

> density_plots<-list()
> for (var in num_vars) {
  density_plot<-ggplot(college,aes(x=.data[[var]])) + geom_density(fill="skyblue",color="blue") +
labs(title=var)
  density_plots[[var]]<-density_plot
}
> plot_grid(plotlist=density_plots,nrow=6,ncol=3)
```



Based on the density plot, it seems that for few of the variables the plot follows gaussian distribution and for few the data is highly skewed either positively or negatively skewed.

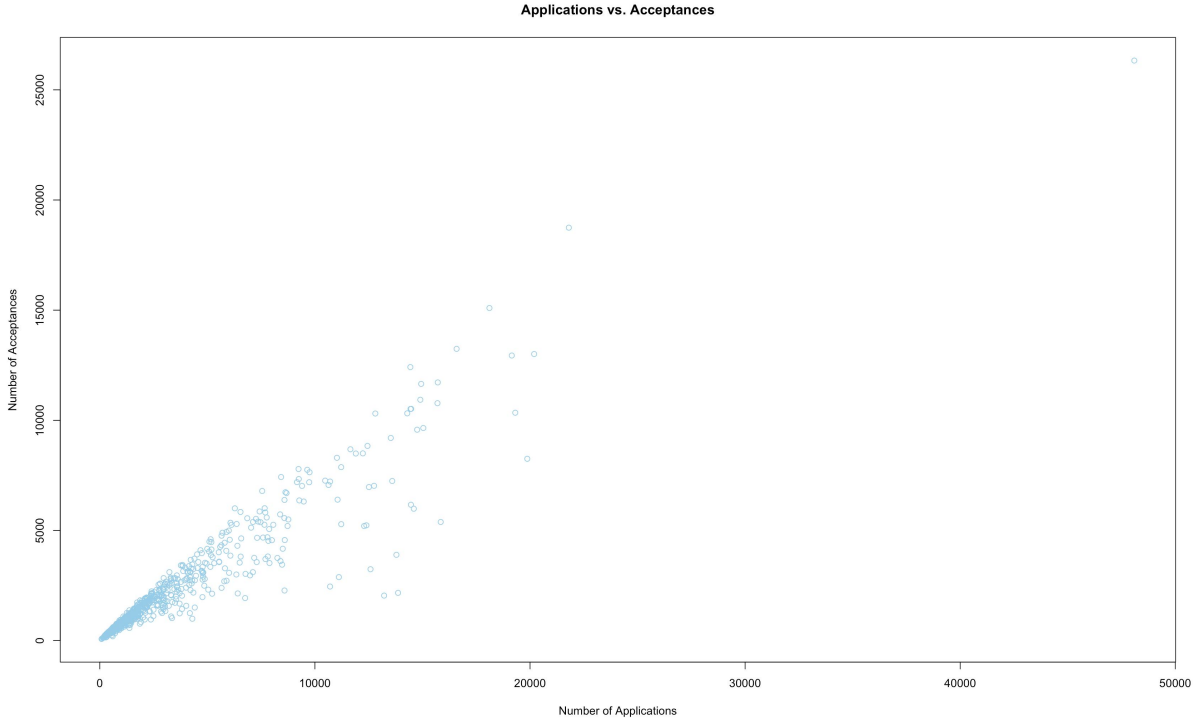
> cor_matrix<-cor(college[num_vars])
> corrplot(cor_matrix,method="color",type="upper", tl.col="black",tl.srt=45)



Based on the above heatmap, there seems to be a high correlation both positive and negative between variables which is not a good sign. To name a few we can see that Outstate is highly and positively correlated to P.Undergrad whereas

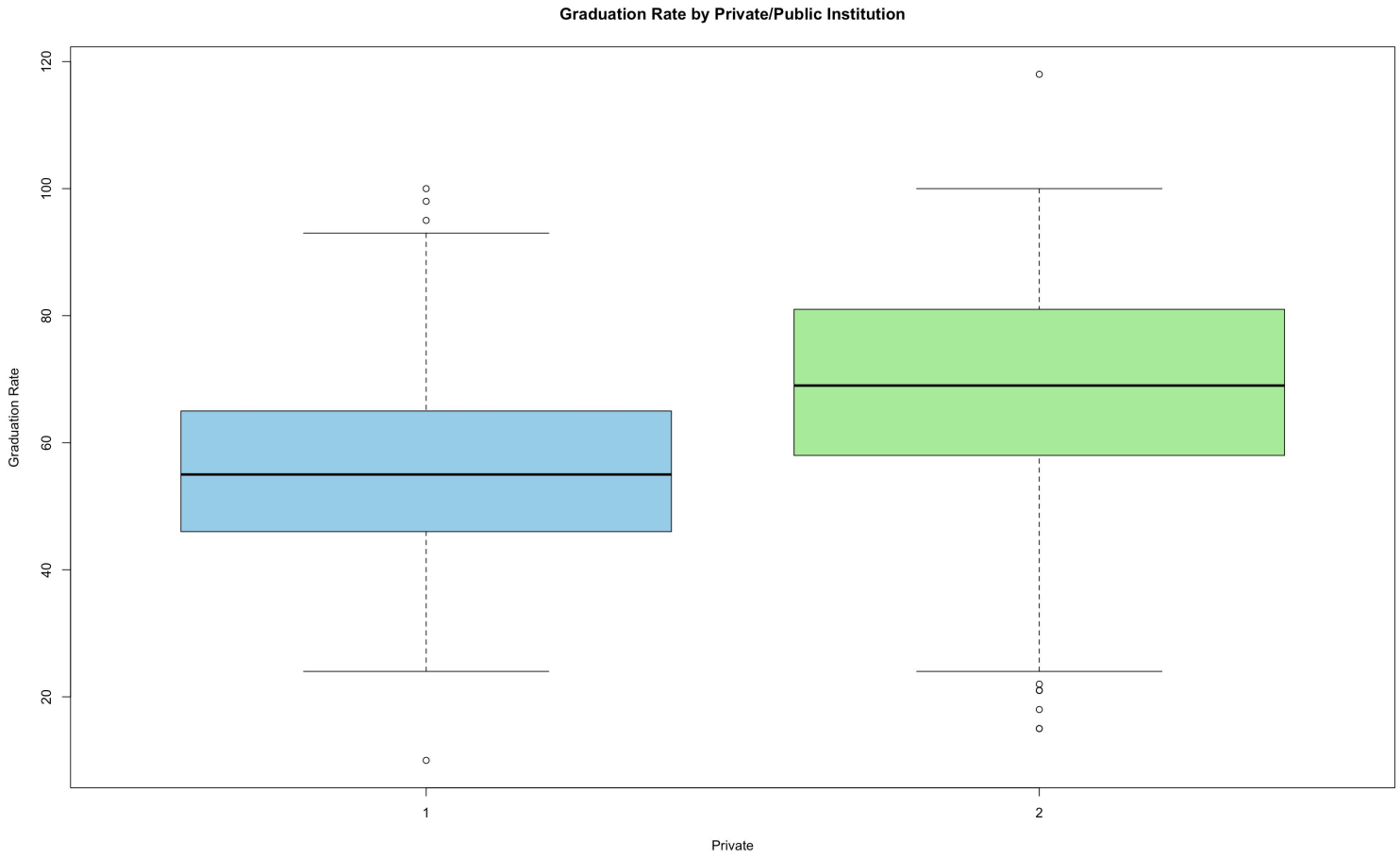
highly and negatively correlated with S.F.Ratio. The darker blue represents high positive correlation whereas darker red/orange represents high negative correlation.

> plot(college$Apps,college$Accept,xlab="Number of Applications",ylab="Number of Acceptances",main="Applications vs. Acceptances",col="skyblue")



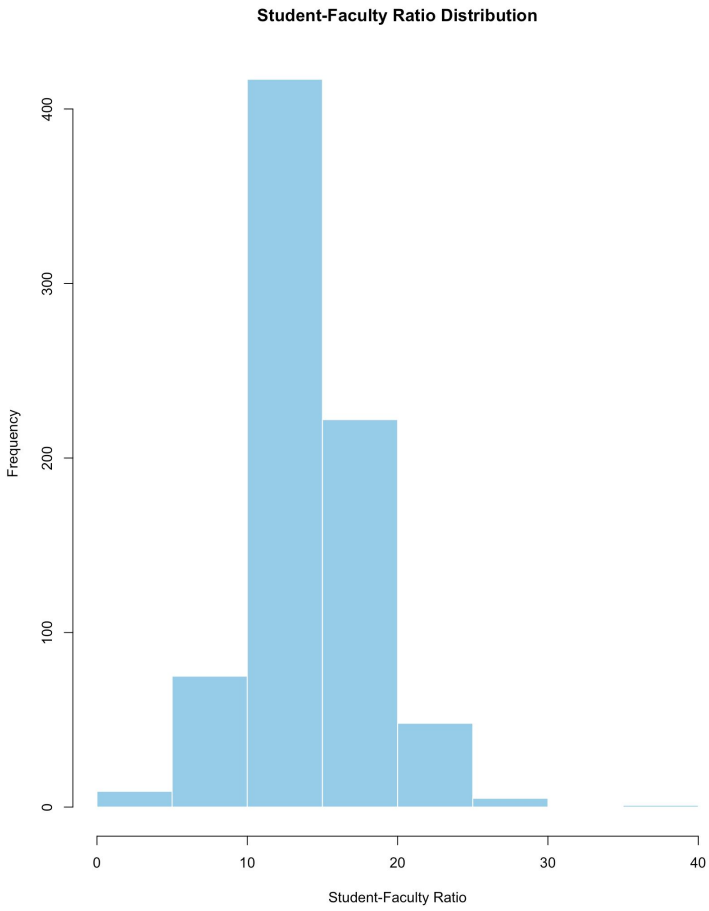
Based on the above scatter plot, there seems to be a positive trend between No. of Applications vs No. of Acceptances.

> boxplot(college$Grad.Rate ~ college$Private,main="Graduation Rate by Private/Public Institution",xlab="Private",ylab="Graduation Rate",col=c("skyblue","lightgreen"))



The boxplot clearly indicates that there is a shift of distribution when Graduation Rate is being compared for Public and Private Institutions.

> hist(college$S.F.Ratio,main="Student-Faculty Ratio Distribution",xlab="Student-Faculty Ratio",col="skyblue",border="white")



The above histogram shows that the average lies between 10 to 15 Faculty handling a bulk of student population.

5. 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",header=TRUE,na.strings="?")
> summary(auto)

```
> auto <- read.csv("./Auto.csv",header=TRUE,na.strings="?")
> summary(auto)
      mpg          cylinders      displacement     horsepower        weight      acceleration        year           origin
 Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0   Min.   :1613   Min.   : 8.00   Min.   :70.00   Min.   :1.000
 1st Qu.:17.50   1st Qu.:4.000   1st Qu.:104.0   1st Qu.: 75.0   1st Qu.:2223   1st Qu.:13.80   1st Qu.:73.00   1st Qu.:1.000
 Median :23.00   Median :4.000   Median :146.0   Median : 93.5   Median :2800   Median :15.50   Median :76.00   Median :1.000
 Mean   :23.52   Mean   :5.458   Mean   :193.5   Mean   :104.5   Mean   :2970   Mean   :15.56   Mean   :75.99   Mean   :1.574
 3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:262.0   3rd Qu.:126.0   3rd Qu.:3609   3rd Qu.:17.10   3rd Qu.:79.00   3rd Qu.:2.000
 Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0   Max.   :5140   Max.   :24.80   Max.   :82.00   Max.   :3.000
                                                 NA's   :5
     name
 Length:397
 Class :character
 Mode  :character
```

> auto<-na.omit(auto)
> head(auto)

```
> auto<-na.omit(auto)
> head(auto)
  mpg cylinders displacement horsepower weight acceleration year origin                       name
1  18         8          307        130   3504         12.0   70      1 chevrolet chevelle malibu
2  15         8          350        165   3693         11.5   70      1         buick skylark 320
3  18         8          318        150   3436         11.0   70      1        plymouth satellite
4  16         8          304        150   3433         12.0   70      1             amc rebel sst
5  17         8          302        140   3449         10.5   70      1               ford torino
6  15         8          429        198   4341         10.0   70      1          ford galaxie 500
```

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

```
> str(auto)
'data.frame':   392 obs. of  9 variables:
 $ mpg         : num  18 15 18 16 17 15 14 14 14 15 ...
 $ cylinders   : int  8 8 8 8 8 8 8 8 8 8 ...
 $ displacement: num  307 350 318 304 302 429 454 440 455 390 ...
 $ horsepower  : int  130 165 150 150 140 198 220 215 225 190 ...
 $ weight      : int  3504 3693 3436 3433 3449 4341 4354 4312 4425 3850 ...
 $ acceleration: num  12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...
 $ year        : int  70 70 70 70 70 70 70 70 70 70 ...
 $ origin      : int  1 1 1 1 1 1 1 1 1 1 ...
 $ name        : chr  "chevrolet chevelle malibu" "buick skylark 320" "plymouth satellite" "amc rebel sst" ...
 - attr(*, "na.action")= 'omit' Named int [1:5] 33 127 331 337 355
  ..- attr(*, "names")= chr [1:5] "33" "127" "331" "337" ...
```

Based on the given dataset
Quantitative predictors are

- ✧ mpg
- ✧ cylinders
- ✧ displacement
- ✧ horsepower
- ✧ weight
- ✧ acceleration
- ✧ year

Qualitative predictors are

- ✧ name
- ✧ origin

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

> sapply(auto[,c("mpg","cylinders","displacement","horsepower","weight","acceleration","year")],range)

```
> sapply(auto[, c("mpg", "cylinders", "displacement", "horsepower", "weight", "acceleration", "year")], range)
      mpg cylinders displacement horsepower weight acceleration year
[1,]  9.0         3           68         46   1613          8.0   70
[2,] 46.6         8          455        230   5140         24.8   82
```

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

> sapply(auto[,c("mpg","cylinders","displacement","horsepower","weight","acceleration","year")],mean)

```
> sapply(auto[, c("mpg", "cylinders", "displacement", "horsepower", "weight", "acceleration",
         mpg    cylinders displacement   horsepower       weight acceleration         year
   23.445918     5.471939   194.411990   104.469388  2977.584184    15.541327    75.979592
```

> sapply(auto[,c("mpg","cylinders","displacement","horsepower","weight","acceleration","year")],sd)

```
> sapply(auto[, c("mpg", "cylinders", "displacement", "horsepower", "weight", "acceleration", "year")], sd)
         mpg    cylinders displacement   horsepower       weight acceleration         year
    7.805007     1.705783   104.644004    38.491160   849.402560     2.758864     3.683737
```

(d) 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?

> auto_subset<-auto[-c(10:85),]
>
sapply(auto_subset[,c("mpg","cylinders","displacement","horsepower","weight","acceleration","year")],range)

```
> sapply(auto_subset[, c("mpg", "cylinders", "displacement", "horsepower", "weight", "acceleration", "year")], range)
      mpg cylinders displacement horsepower weight acceleration year
[1,] 11.0         3           68         46   1649          8.5   70
[2,] 46.6         8          455        230   4997         24.8   82
```

>
sapply(auto_subset[,c("mpg","cylinders","displacement","horsepower","weight","acceleration","year")],mean)

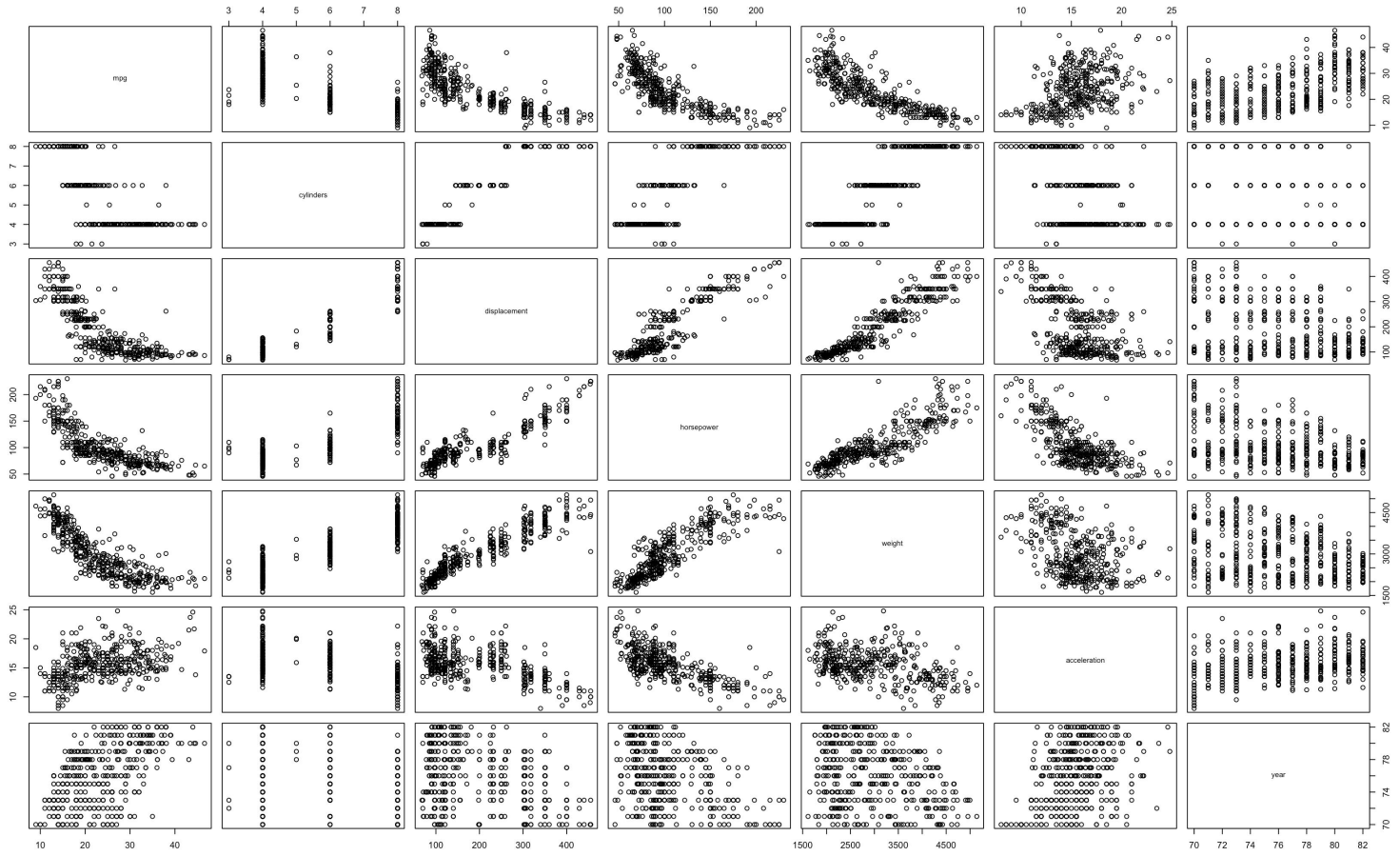
```
> sapply(auto_subset[, c("mpg", "cylinders", "displacement", "horsepower", "weight", "acceleration", "year")], mean)
        mpg    cylinders displacement   horsepower       weight acceleration         year
  24.404430     5.373418   187.240506   100.721519  2935.971519    15.726899    77.145570
```

>
sapply(auto_subset[,c("mpg","cylinders","displacement","horsepower","weight","acceleration","year")],sd)

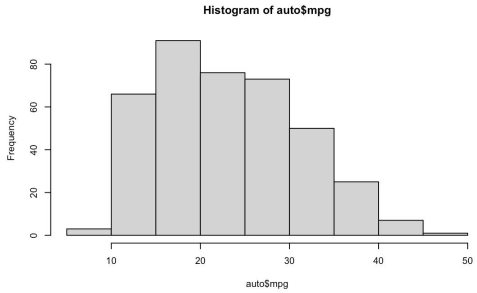
```
> sapply(auto_subset[, c("mpg", "cylinders", "displacement", "horsepower", "weight", "acceleration", "year")], sd)
        mpg    cylinders displacement   horsepower       weight acceleration         year
   7.867283     1.654179    99.678367    35.708853   811.300208     2.693721     3.106217
```

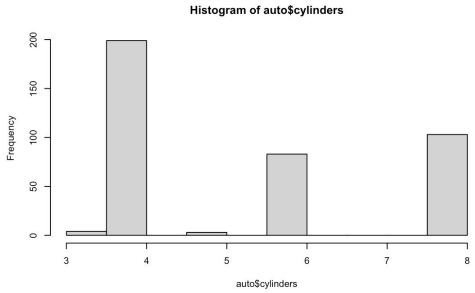
(e) Using the full data set, investigate the predictors graphically, using scatterplots or other tools of your choice. Create some plots highlighting the relationships among the predictors. Comment on your findings.

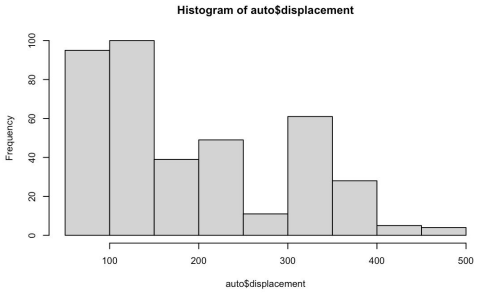
> pairs(auto[,c("mpg","cylinders","displacement","horsepower","weight","acceleration","year")])

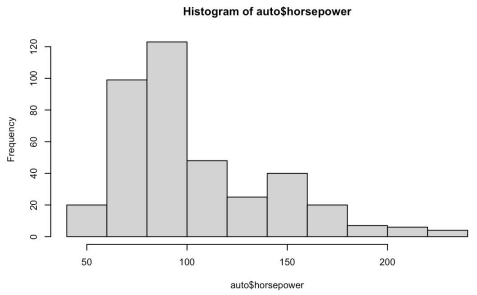


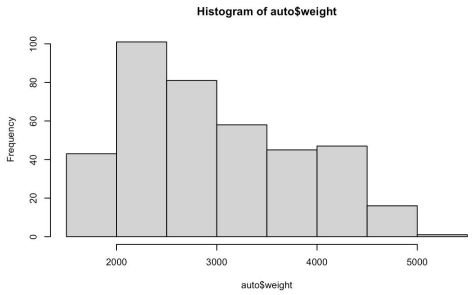
> **par(mfrow=c(3,3))**
> hist(auto$mpg)
> hist(auto$cylinders)
> hist(auto$displacement)
> hist(auto$horsepower)
> hist(auto$weight)
> hist(auto$acceleration)
> hist(auto$year)

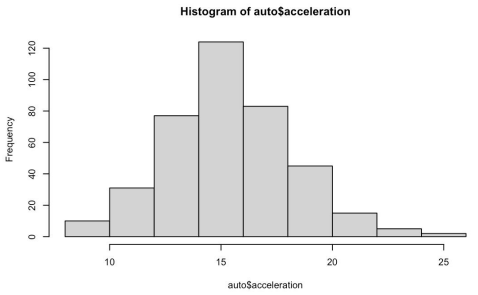


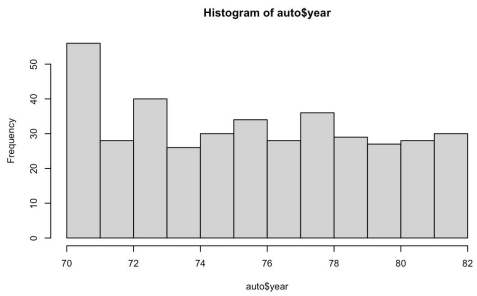








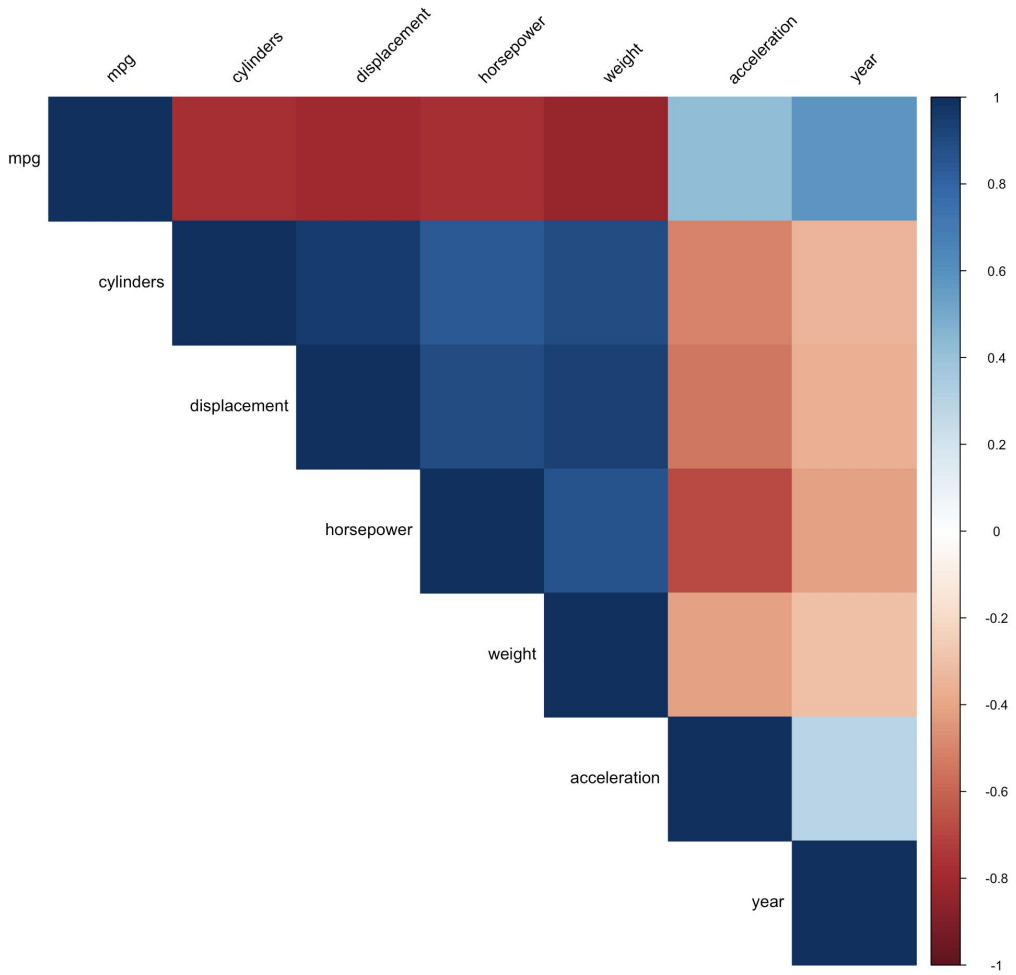




The above histogram plots shows the skewed data distribution for given features. Some Features such as acceleration have gaussian normal distribution whereas for horsepower the distribution is skewed or tailed towards one end (positive skewness). For cylinders we can see that there are no records for 7 cylinder car in the dataset.

(f) 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.

> library(corrplot)
> cor_matrix<-
cor(auto[,c("mpg","cylinders","displacement","horsepower","weight","acceleration","year")])
> corrplot(cor_matrix,method="color",type="upper",tl.col="black",tl.srt=45)



Based on the above heatmap, there seems to be a high correlation both positive and negative between variables which is not a good sign. To name a few we can see that horsepower is highly and negatively correlated to displacement whereas likely and positively correlated with cylinders. Likewise the correlation between horsepower and mpg fuel efficiency is negative. The darker blue represents high positive correlation whereas darker red/orange represents high negative correlation.

6. Describe the null hypotheses to which the p-values given in Table 3.4 correspond. Explain what conclusions you can draw based on these p-values. Your explanation should be phrased in terms of sales, TV, radio, and newspaper, rather than in terms of the coefficients of the linear model.

The small p-values in Table 3.4 for TV and radio indicate that, holding the other predictors constant, each of these predictors will, with high likelihood, change the amount of sales. For example, fixing the amount of TV and newspaper advertising, while increasing the amount of radio advertising, will very likely lead to an increase in sales because the p-value in Table 3.4 is very small for radio and radio's coefficient is large and positive. On the other hand, since the p-value for newspaper is quite large, the data indicates that newspaper advertising is unlikely to have any effect on sales when TV and radio are held fixed. Note, Table 3.3 does show a small p-value for a single linear regression across newspaper and sales, but this is likely because newspaper advertising is predictive of TV and radio advertising, not necessarily because newspaper advertising directly influences sales.

7. Consider the fitted values that result from performing linear regression without an intercept. In this setting, the ith fitted value takes the form

Date

$\hat{y}_i = x_i \hat{\beta}$ where $\hat{\beta} = \dfrac{\sum_{i=1}^{n} x_i y_i}{\sum_{j=1}^{n} x_j^2}$

Instead of x_i we assume x_j for convinience (i.e replace 'i' with 'j')

To prove $\hat{y}_i = \sum_{j=1}^{n} a_j y_i$ what is a_j?

$$y_i = x_i \frac{\sum_{i=1}^{n} x_i y_i}{\sum_{j=1}^{n} x_j^2}$$

$$= x_i \cdot \sum_{i=1}^{n} \frac{1}{x_j^2} \cdot x_i y_i$$

$$= \sum_{i=1}^{n} \frac{1}{n} \cdot x_i \cdot \frac{1}{x_j^2} \cdot x_i \cdot y_i$$

$$= \sum_{j=1}^{n} a_j \cdot y_i \text{ where } a_j = \frac{1}{n} \cdot x_j \cdot \frac{1}{x_j^2} \cdot x_j$$

$$= \sum_{j=1}^{n} a_j \cdot y_i \text{ where } a_j = x_j \cdot \frac{1}{n \cdot x_j^2} \cdot x_j$$

$$= \sum_{j=1}^{n} a_j y_i \text{ where } a_j = \left(x_j^2 / n \cdot x_j^2\right)$$

$$\therefore \sum_{j=1}^{n} a_j y_i \text{ where } a_j = \frac{1}{n}$$

8. Using (3.4), argue that in the case of simple linear regression, the least squares line always passes through the point $(\bar{x}, \bar{y})$.

for $x = \text{mean}(x)$ i.e μ_x & $y = \text{mean}(y)$ i.e μ_y

$$\hat{\beta}_0 = \mu_y - \hat{\beta}_1 \cdot \mu_x$$

$$\hat{\beta}_1 = \left[\sum_{i=1}^{n} ((x_i - \mu_x)(y_i - \mu_y)) \Big/ \sum_{i=1}^{n} (x_i - \mu_x)^2 \right]$$

least square line is the line

$$\bar{y} = \beta_0 + \beta_1 \cdot \mu_x$$

$$\therefore \mu_y = \beta_0 + \beta_1 \cdot \mu_x$$

$$\mu_y = \mu_y - (\beta_1 \cdot \mu_x + \beta_1 \cdot \mu_x)$$

$$\therefore \mu_y = \mu_y$$

$\therefore \mu_x$ & μ_y must be within the least square line for any given such μ_x & μ_y. Thus it passes through the point $(\bar{x}, \bar{y})$

9. This question involves the use of simple linear regression on the Auto data set.
(a) Use the lm() function to perform a simple linear regression with mpg as the response and horsepower as the predictor. Use the summary() function to print the results. Comment on the output. For example:

> lm_auto_fit<-lm(mpg~horsepower,data=auto)
> summary(lm_auto_fit)

```
> lm_auto_fit <- lm(mpg~horsepower,data=auto)
> summary(lm_auto_fit)

Call:
lm(formula = mpg ~ horsepower, data = auto)

Residuals:
     Min       1Q   Median       3Q      Max
-13.5710  -3.2592  -0.3435   2.7630  16.9240

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 39.935861   0.717499   55.66   <2e-16 ***
horsepower  -0.157845   0.006446  -24.49   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.906 on 390 degrees of freedom
Multiple R-squared:  0.6059,    Adjusted R-squared:  0.6049
F-statistic: 599.7 on 1 and 390 DF,  p-value: < 2.2e-16
```

i. Is there a relationship between the predictor and the response?

Following the analysis of the null hypothesis, which assumes that all regression coefficients are zero, we have discovered compelling evidence supporting a correlation between horsepower and mpg in the dataset. This conclusion is substantiated by the observation of a markedly large F-statistic and an exceedingly small p-value associated with the F-statistic. These statistical indicators suggest that the relationship between horsepower and mpg is highly significant from a statistical standpoint. Consequently, we reject the null hypothesis, indicating that there is indeed a meaningful and statistically significant relationship between the horsepower of a vehicle and its miles per gallon efficiency.

ii. How strong is the relationship between the predictor and the response?

Utilizing the mean of the response variable and the Residual Standard Error (RSE), we can gauge the extent of residual error relative to the response variable. The mean value of mpg in the dataset is calculated to be 23.4459. The RSE, a measure of the typical deviation of the observed values from the regression line, is estimated to be 4.906. This RSE value implies a percentage error of approximately 20.9248%. Additionally, the coefficient of determination (R-squared) of the linear model fit is approximately 0.6059. This statistic indicates that approximately 60.5948% of the variability observed in mpg can be accounted for by the predictor variable, horsepower. In other words, the linear regression model explains a substantial portion of the variance in mpg based on the relationship with horsepower.

iii. Is the relationship between the predictor and the response positive or negative?

The analysis reveals a negative correlation between horsepower and mpg fuel efficiency. Specifically, the results of the linear regression analysis indicate that as the horsepower of a vehicle increases, its miles per gallon (mpg) fuel efficiency tends to decrease. This suggests an inverse relationship between horsepower and fuel efficiency, where higher horsepower is associated with lower mpg values. Such findings are valuable in understanding the trade-off between engine power and fuel economy in automobiles.

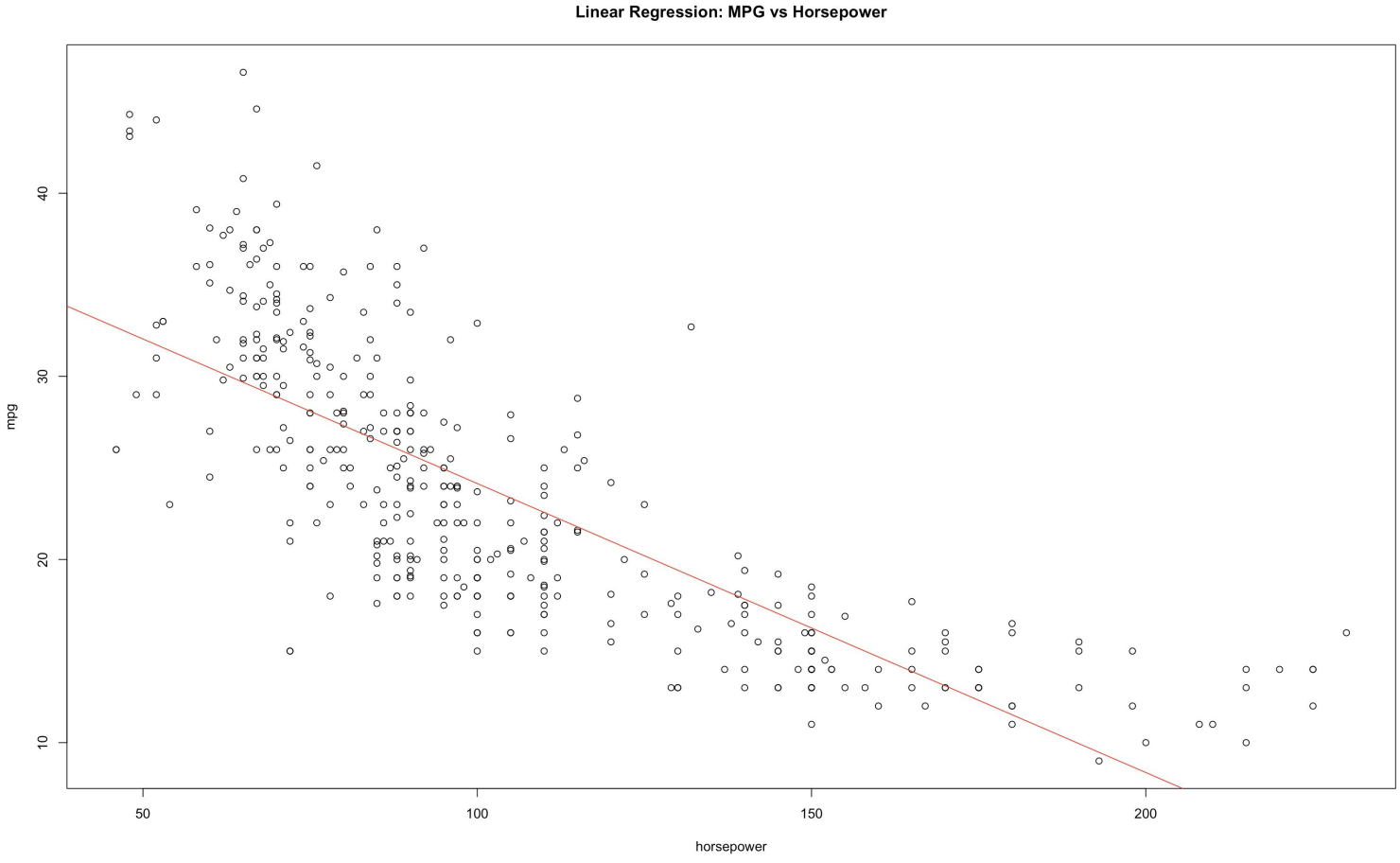
iv. What is the predicted mpg associated with a horsepower of 98? What are the associated 95 % confidence and prediction intervals?

> predict(lm_auto_fit,newdata=data.frame(horsepower=98),interval="confidence",level=0.95)
> predict(lm_auto_fit,newdata=data.frame(horsepower=98),interval="prediction",level=0.95)

```
> predict(lm_auto_fit, newdata = data.frame(horsepower = 98), interval = "confidence", level = 0.95)
       fit      lwr      upr
1 24.46708 23.97308 24.96108
> predict(lm_auto_fit, newdata = data.frame(horsepower = 98), interval = "prediction", level = 0.95)
       fit     lwr      upr
1 24.46708 14.8094 34.12476
```

(b) Plot the response and the predictor. Use the abline() function to display the least squares regression line.

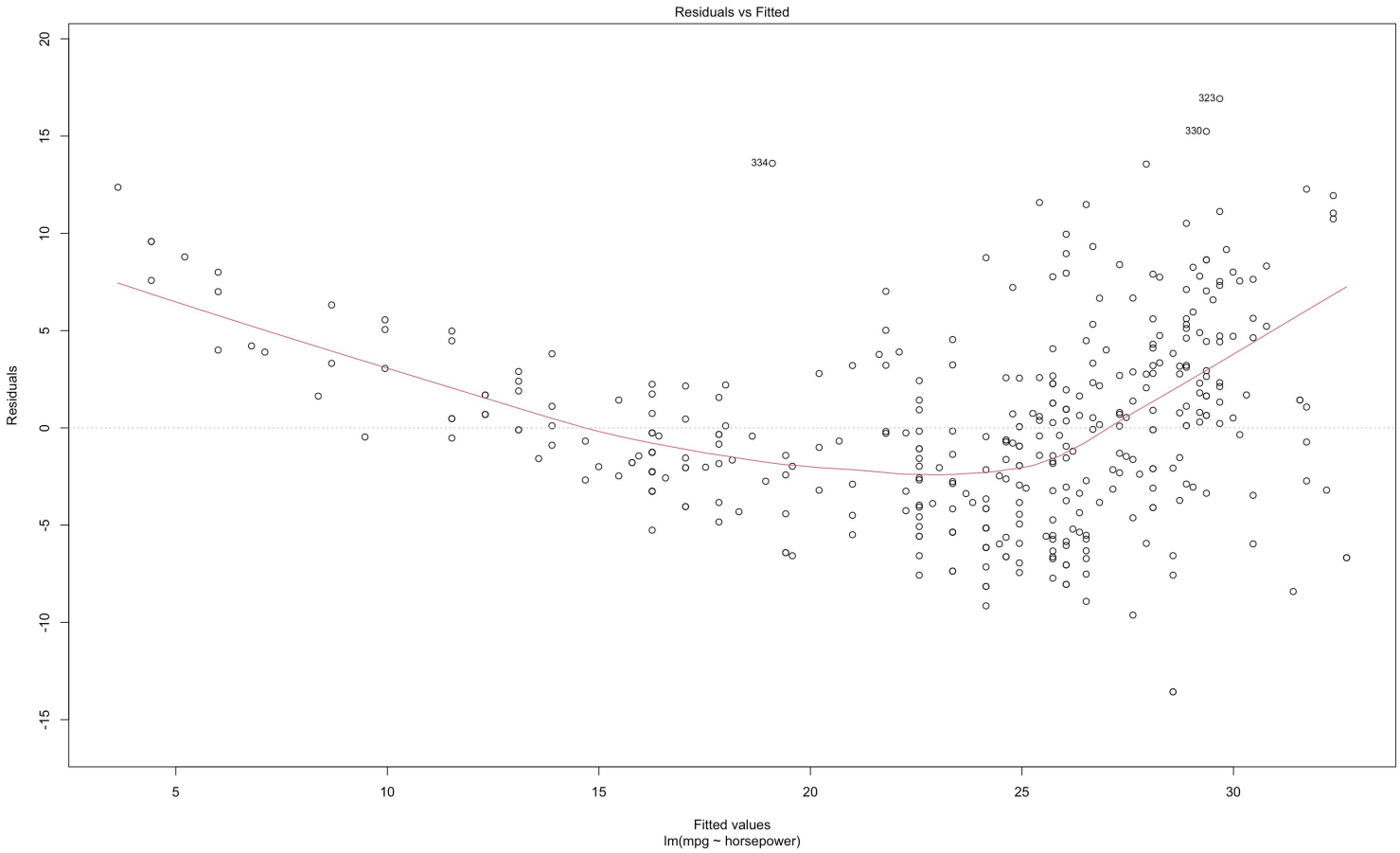
> plot(auto$horsepower,auto$mpg,main="Linear Regression: MPG vs Horsepower",xlab="horsepower",ylab="mpg",col="black")
> abline(lm_auto_fit,col="red")

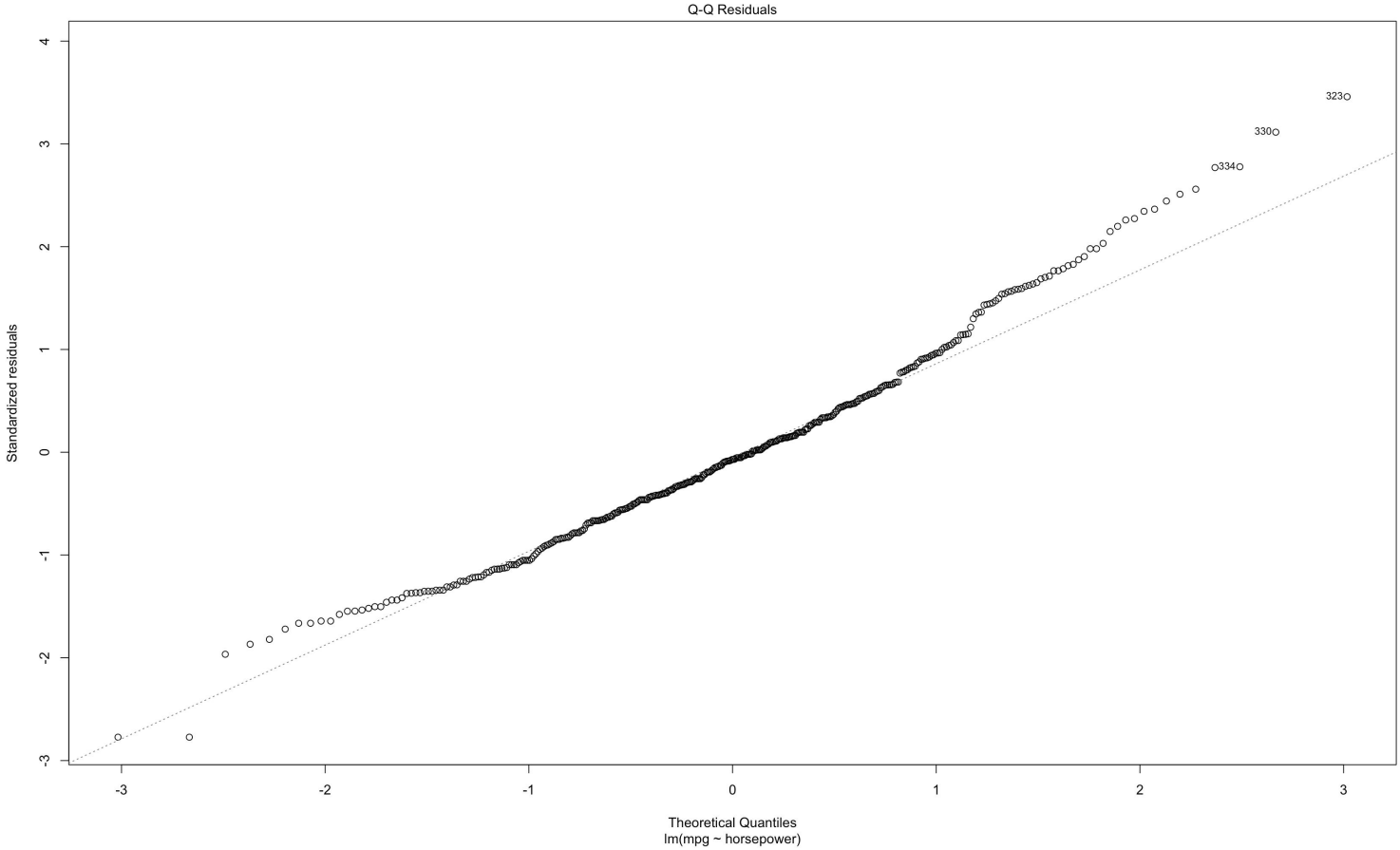


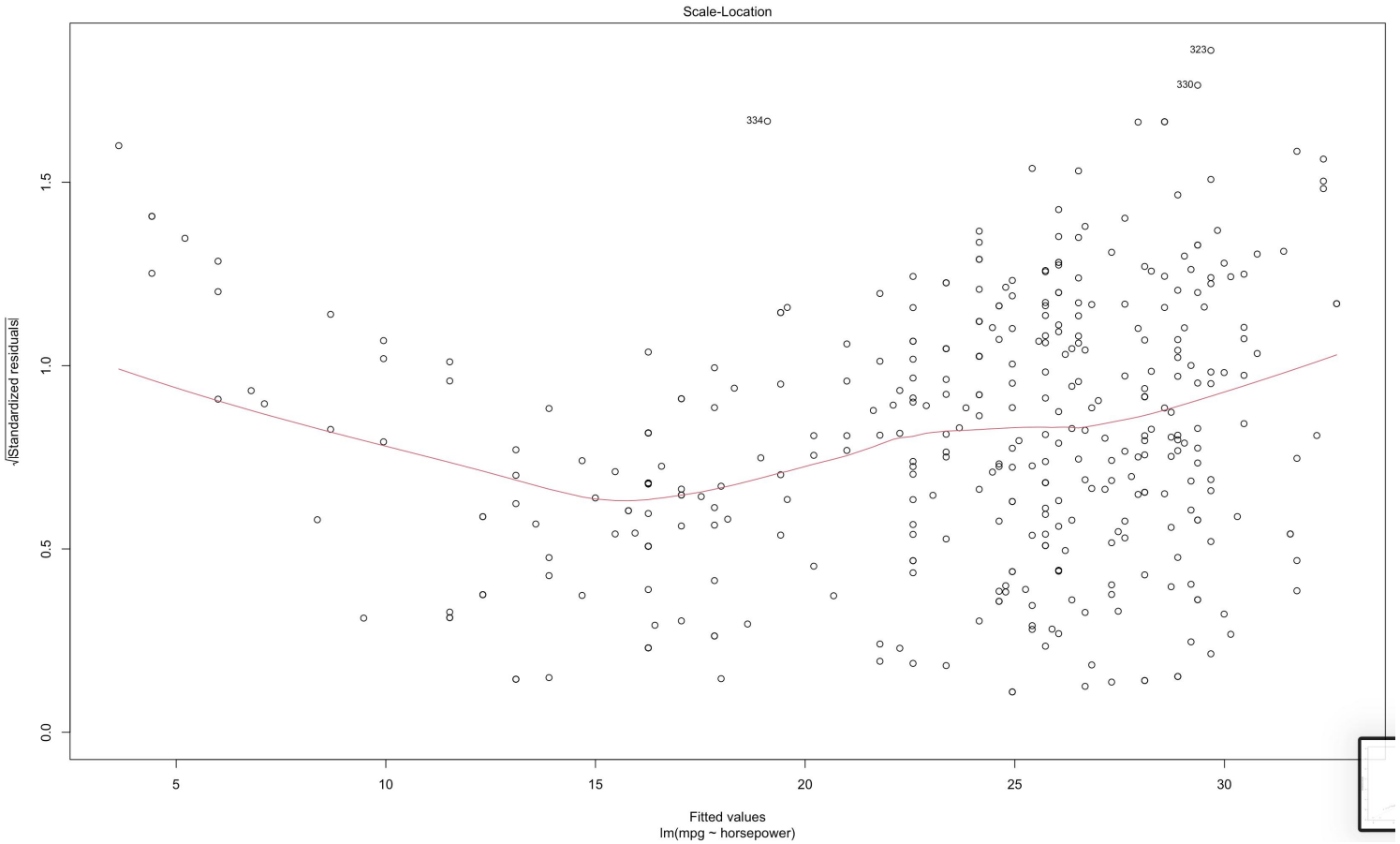
(c) Use the plot() function to produce diagnostic plots of the least squares regression fit. Comment on any problems you see with the fit.

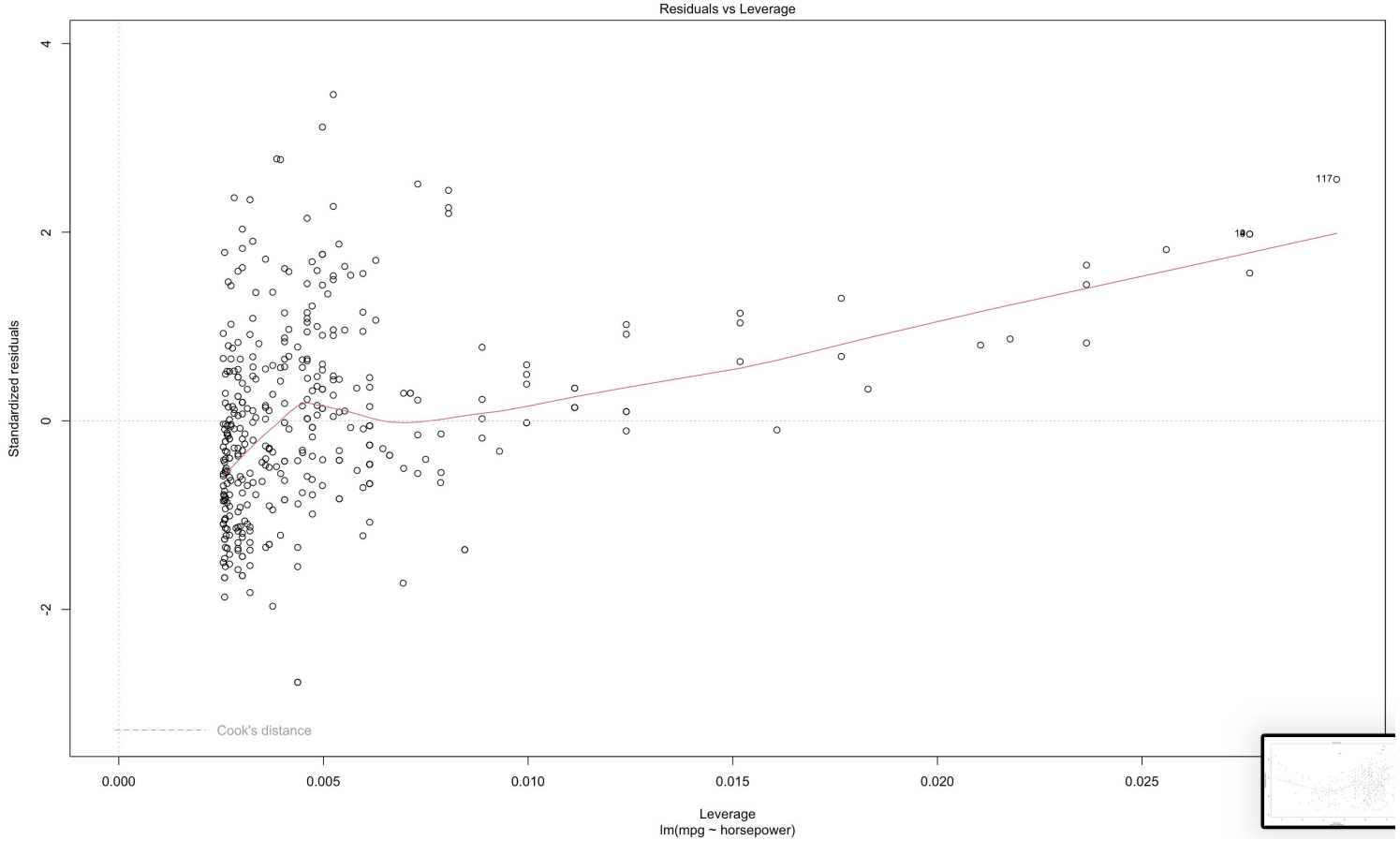
> plot(lm_auto_fit)

Upon examination of the plots depicting residuals versus fitted values, it becomes apparent that the data displays indications of non-linearity. This observation, coupled with the insights garnered from the plot obtained in section 9(b), suggests that the simple linear regression model may not adequately capture the underlying relationship between the predictor and response variables. Further analysis of the diagnostic plots reveals several noteworthy observations. Firstly, the plot of residuals versus fitted values indicates potential departures from linearity, implying that the relationship between the predictor and response variables may be more complex than initially assumed. However, it is worth noting that the plot depicting residuals versus fitted values shows that the residuals follow a normal distribution, suggesting that the assumption of normality is reasonable. Additionally, the plot of standardized residuals versus leverage identifies the presence of outliers, characterized by values exceeding 2 or falling below -2, as well as a few high leverage points. These outliers and high leverage points may exert a disproportionate influence on the regression model and warrant further investigation. While the simple linear regression model provides insights into the relationship between the predictor and response variables, the presence of non-linearity, outliers, and high leverage points suggests that a more sophisticated modeling approach may be necessary to accurately capture the underlying data structure.


Residuals vs Fitted
Residuals
Fitted values
lm(mpg ~ horsepower)
323
330
334

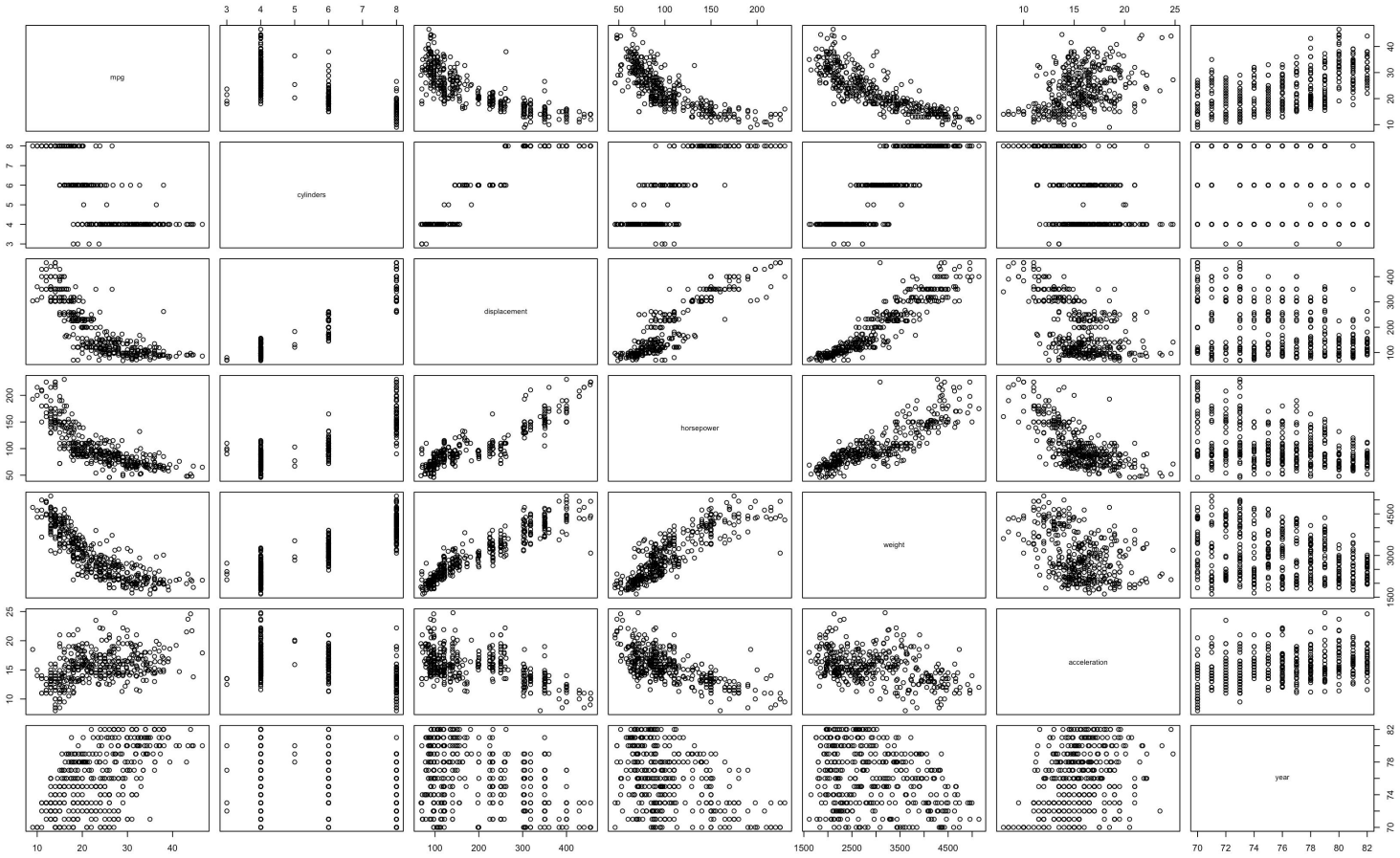

Q-Q Residuals
Standardized residuals
Theoretical Quantiles
lm(mpg ~ horsepower)
323
330
334


Scale-Location
323
330
334
√|Standardized residuals|
0.0
0.5
1.0
1.5
5
10
15
20
25
30
Fitted values
lm(mpg ~ horsepower)


Residuals vs Leverage
117
14
Standardized residuals
4
2
0
-2
Cook's distance
0.000
0.005
0.010
0.015
0.020
0.025
Leverage
lm(mpg ~ horsepower)

10. This question involves the use of multiple linear regression on the Auto data set.
(a) Produce a scatterplot matrix which includes all of the variables in the data set.

> pairs(auto[1:8])



(b) Compute the matrix of correlations between the variables using the function cor(). You will need to exclude the name variable, which is qualitative.

> cor(auto[1:8])

```
> cor(auto[1:8])
                    mpg  cylinders displacement horsepower     weight acceleration       year     origin
mpg           1.0000000 -0.7776175   -0.8051269 -0.7784268 -0.8322442    0.4233285  0.5805410  0.5652088
cylinders    -0.7776175  1.0000000    0.9508233  0.8429834  0.8975273   -0.5046834 -0.3456474 -0.5689316
displacement -0.8051269  0.9508233    1.0000000  0.8972570  0.9329944   -0.5438005 -0.3698552 -0.6145351
horsepower   -0.7784268  0.8429834    0.8972570  1.0000000  0.8645377   -0.6891955 -0.4163615 -0.4551715
weight       -0.8322442  0.8975273    0.9329944  0.8645377  1.0000000   -0.4168392 -0.3091199 -0.5850054
acceleration  0.4233285 -0.5046834   -0.5438005 -0.6891955 -0.4168392    1.0000000  0.2903161  0.2127458
year          0.5805410 -0.3456474   -0.3698552 -0.4163615 -0.3091199    0.2903161  1.0000000  0.1815277
origin        0.5652088 -0.5689316   -0.6145351 -0.4551715 -0.5850054    0.2127458  0.1815277  1.0000000
```

(c) Use the lm() function to perform a multiple linear regression with mpg as the response and all other variables except name as the predictors. Use the summary() function to print the results. Comment on the output. For instance:
i. Is there a relationship between the predictors and the response?

The analysis reveals that multiple predictors exhibit a significant relationship with the response variable. This is evidenced by their associated p-values, which quantify the probability of observing the coefficient value under the null hypothesis that the coefficient is zero. Typically, a p-value of 0.05 or less is considered significant, indicating a low probability of the coefficient being zero. To confirm the significance of this relationship, the hypothesis testing is reiterated. The p-value corresponding to the F-statistic is calculated to be approximately 2.037 x 10^-139, providing strong evidence of a relationship between "mpg" and the other predictors. This exceedingly small p-value suggests that the relationship between the response variable and the predictors is highly significant and unlikely to occur by chance. Hence, we can confidently conclude that there exists a substantial association between "mpg" and the predictors included in the multiple linear regression model.

> lm_auto_fit<-lm(mpg ~ . - name,data=auto)
> summary(lm_auto_fit)

```
> lm_auto_fit <- lm(mpg ~ . - name, data = auto)
> summary(lm_auto_fit)

Call:
lm(formula = mpg ~ . - name, data = auto)

Residuals:
    Min      1Q  Median      3Q     Max
-9.5903 -2.1565 -0.1169  1.8690 13.0604

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  -17.218435   4.644294  -3.707  0.00024 ***
cylinders     -0.493376   0.323282  -1.526  0.12780
displacement   0.019896   0.007515   2.647  0.00844 **
horsepower    -0.016951   0.013787  -1.230  0.21963
weight        -0.006474   0.000652  -9.929  < 2e-16 ***
acceleration   0.080576   0.098845   0.815  0.41548
year           0.750773   0.050973  14.729  < 2e-16 ***
origin         1.426141   0.278136   5.127 4.67e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.328 on 384 degrees of freedom
Multiple R-squared:  0.8215,    Adjusted R-squared:  0.8182
F-statistic: 252.4 on 7 and 384 DF,  p-value: < 2.2e-16
```

ii. Which predictors appear to have a statistically significant relationship to the response?

The statistical significance of each predictor can be assessed by examining their associated p-values. From the analysis, it is observed that with the exception of "cylinders", "horsepower", and "acceleration", all other predictors demonstrate statistical significance. This implies that for the majority of predictors, the probability of observing the coefficient value under the null hypothesis of zero is sufficiently low, indicating a meaningful relationship with the response variable. However, "cylinders", "horsepower", and "acceleration" do not exhibit statistically significant relationships with the response variable, suggesting that their coefficients may not significantly contribute to explaining the variability in the response.

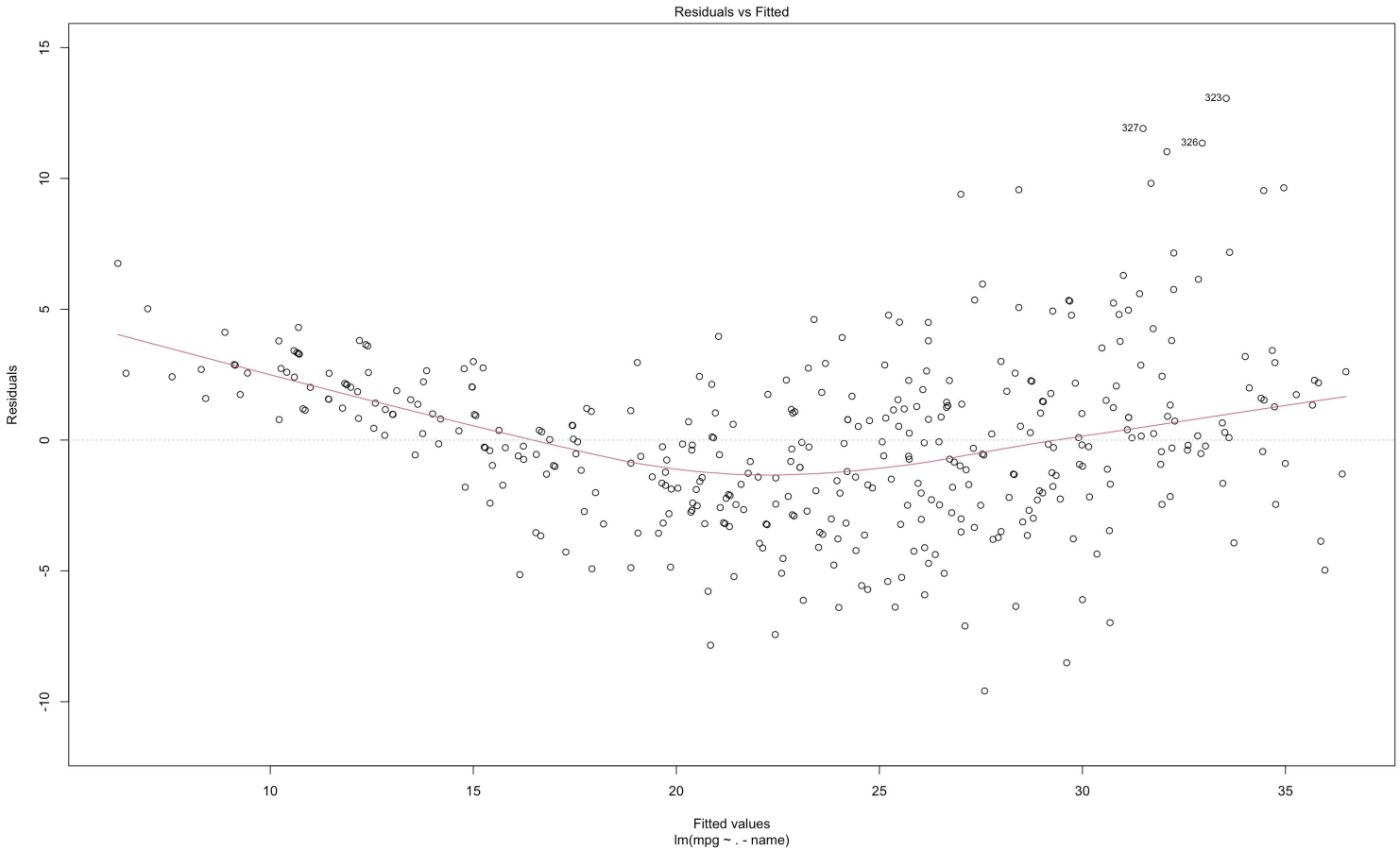
iv. What does the coefficient for the year variable suggest?

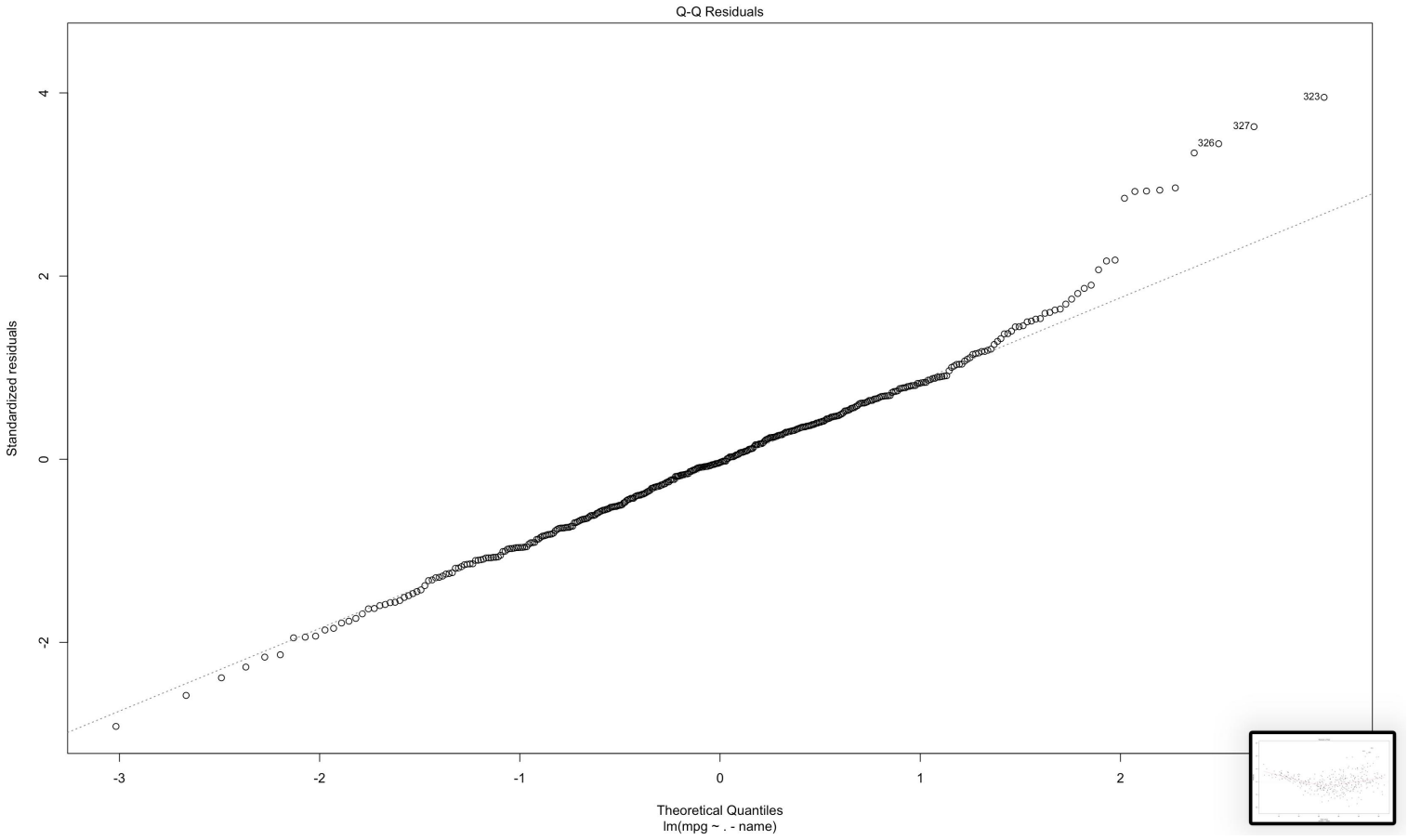
According to the coefficient associated with the "year" variable in the regression model, for every one-year increase in the vehicle's model year, there is a corresponding increase of approximately 0.7507727 in miles per gallon (mpg) fuel efficiency. This suggests that cars become almost 1 mpg more fuel-efficient per year, assuming all other predictors remain constant. In simpler terms, this means that newer cars tend to have slightly better fuel efficiency, with each passing year contributing to a gradual improvement in mpg. This insight highlights the trend of advancements in automotive technology and efficiency over time.

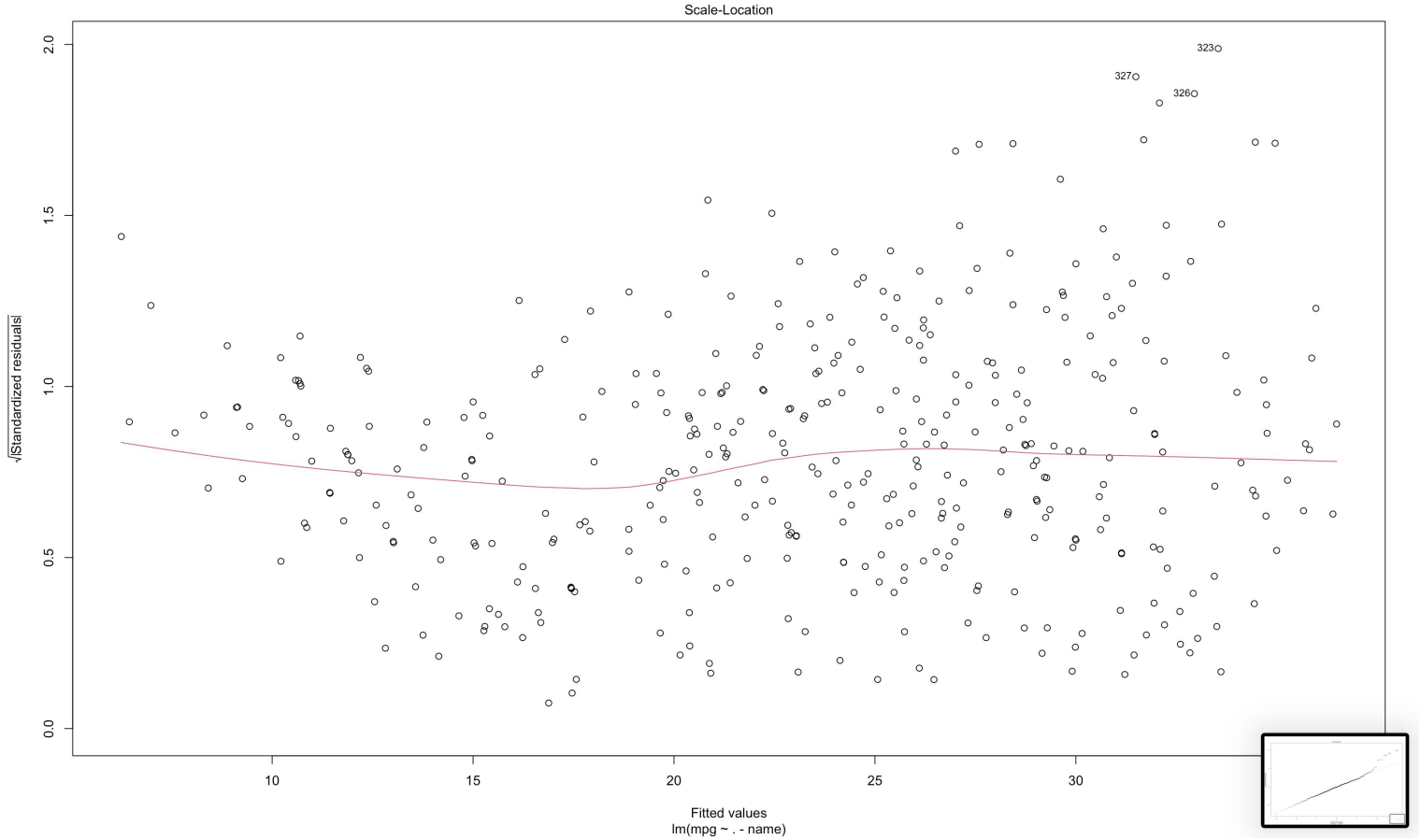
(d) Use the plot() function to produce diagnostic plots of the linear regression fit. Comment on any problems you see with the fit. Do the residual plots suggest any unusually large outliers? Does the leverage plot identify any observations with unusually high leverage?

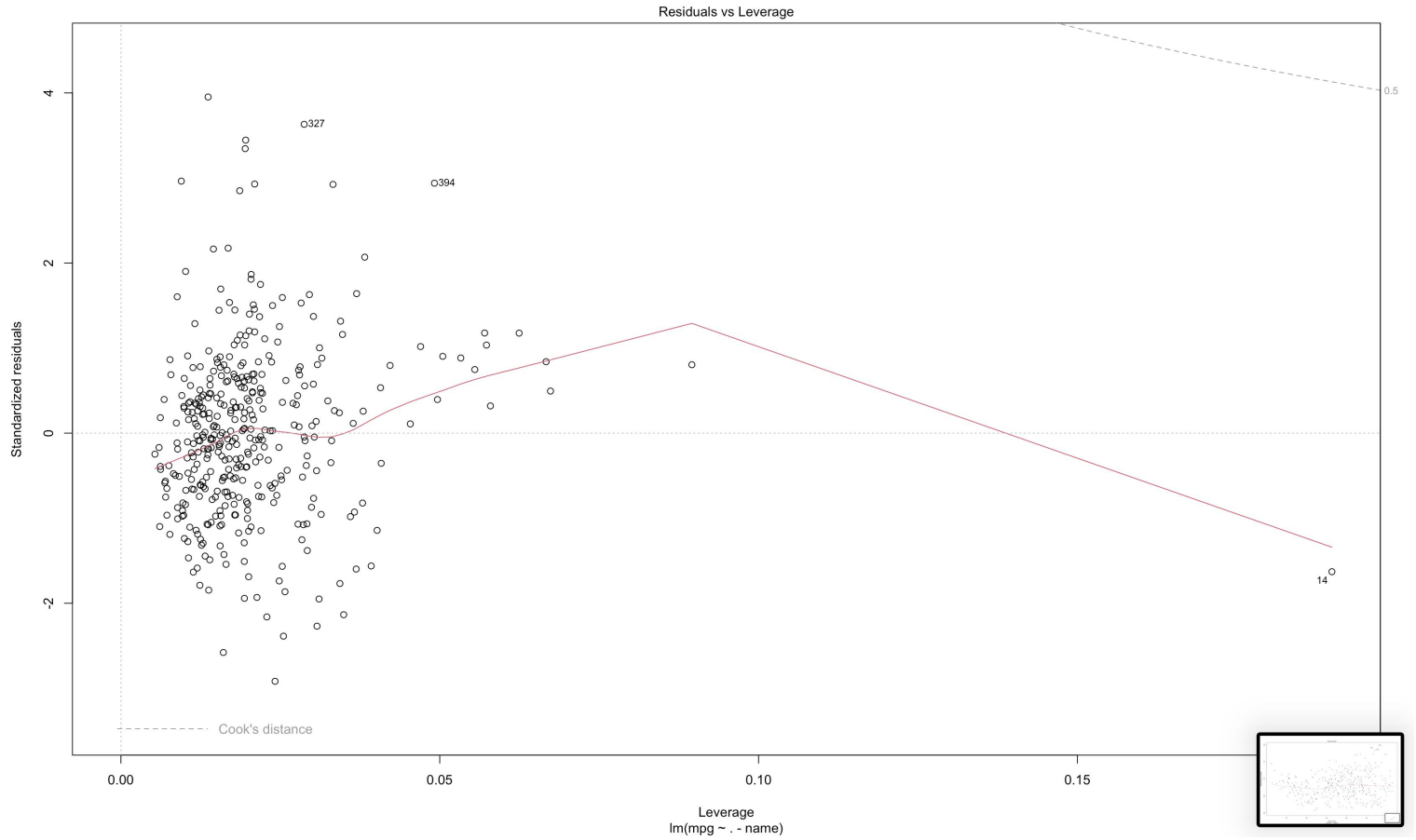
> plot(lm_auto_fit)

The examination of residual plots provides valuable insights into the adequacy of the linear regression model applied to the data. Notably, the presence of a U-shaped pattern in the residuals suggests potential non-linearity in the relationship between predictor variables and the response. Additionally, the funnel shape observed in the residual plot indicates heteroscedasticity, challenging the assumption of constant variance across the predictor range. Furthermore, deviations from the dashed line on the Normal Q-Q plot raise concerns about the normality of residuals, particularly for a few observations. Despite these challenges, the absence of outliers, as indicated by the Scale-Location plot, suggests robustness against extreme data points. Moreover, the Residuals vs Leverage plot indicates no high leverage points, alleviating concerns about individual observations disproportionately influencing the model. In summary, while the linear regression model offers valuable insights, the identified deviations from model assumptions highlight the importance of further exploration and potential model refinement to better capture the underlying data structure.


Residuals vs Fitted
Residuals
Fitted values
lm(mpg ~ . - name)
323
327
326


Q-Q Residuals
Standardized residuals
Theoretical Quantiles
lm(mpg ~ . - name)
323
327
326


Scale-Location
√|Standardized residuals|
Fitted values
lm(mpg ~ . - name)
323
327
326


Residuals vs Leverage
Standardized residuals
Leverage
lm(mpg ~ . - name)
Cook's distance
0.5
327
394
14

(e) Use the * and : symbols to fit linear regression models with interaction effects. Do any interactions appear to be statistically significant?

Linear regression typically assumes that the effects of predictors on the response variable are additive, meaning that changes in one predictor do not influence the relationship between another predictor and the response. However, in some cases, predictors may interact with each other, meaning that the effect of one predictor on the response variable depends on the value of another predictor. This interaction effect can be accounted for by creating a new term in the model, known as an interaction term, which is the product of the two predictors involved in the interaction. By including interaction terms in the model, we can capture and model the complex relationships between predictors and the response variable more accurately, allowing for a more nuanced understanding of the data.

> lm_auto_fit<-lm(mpg~horsepower*displacement,data=auto[,1:8])
> summary(lm_auto_fit)

```
Call:
lm(formula = mpg ~ horsepower * displacement, data = auto[, 1:8])

Residuals:
     Min       1Q   Median       3Q      Max
-10.9391  -2.3373  -0.5816   2.1698  17.5771

Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)              5.305e+01  1.526e+00   34.77   <2e-16 ***
horsepower              -2.343e-01  1.959e-02  -11.96   <2e-16 ***
displacement            -9.805e-02  6.682e-03  -14.67   <2e-16 ***
horsepower:displacement  5.828e-04  5.193e-05   11.22   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.944 on 388 degrees of freedom
Multiple R-squared:  0.7466,    Adjusted R-squared:  0.7446
F-statistic:   381 on 3 and 388 DF,  p-value: < 2.2e-16
```

```
> lm_auto_fit<-lm(mpg~horsepower*weight,data=auto[,1:8])
> summary(lm_auto_fit)

Call:
lm(formula = mpg ~ horsepower * weight, data = auto[, 1:8])

Residuals:
     Min       1Q   Median       3Q      Max
-10.7725  -2.2074  -0.2708   1.9973  14.7314

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       6.356e+01  2.343e+00  27.127  < 2e-16 ***
horsepower       -2.508e-01  2.728e-02  -9.195  < 2e-16 ***
weight           -1.077e-02  7.738e-04 -13.921  < 2e-16 ***
horsepower:weight  5.355e-05  6.649e-06   8.054 9.93e-15 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.93 on 388 degrees of freedom
Multiple R-squared:  0.7484,    Adjusted R-squared:  0.7465
F-statistic: 384.8 on 3 and 388 DF,  p-value: < 2.2e-16
```

```
> lm_auto_fit<-lm(mpg~horsepower*displacement,data=auto[,1:8])
> summary(lm_auto_fit)

Call:
lm(formula = mpg ~ horsepower * displacement, data = auto[, 1:8])

Residuals:
     Min       1Q   Median       3Q      Max
-10.9391  -2.3373  -0.5816   2.1698  17.5771

Coefficients:
                           Estimate Std. Error t value Pr(>|t|)
(Intercept)               5.305e+01  1.526e+00   34.77   <2e-16 ***
horsepower               -2.343e-01  1.959e-02  -11.96   <2e-16 ***
displacement             -9.805e-02  6.682e-03  -14.67   <2e-16 ***
horsepower:displacement   5.828e-04  5.193e-05   11.22   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.944 on 388 degrees of freedom
Multiple R-squared:  0.7466,    Adjusted R-squared:  0.7446
F-statistic:   381 on 3 and 388 DF,  p-value: < 2.2e-16
```

```
> lm_auto_fit<-lm(mpg~horsepower*weight+horsepower*displacement,data=auto[,1:8])
> summary(lm_auto_fit)

Call:
lm(formula = mpg ~ horsepower * weight + horsepower * displacement,
    data = auto[, 1:8])

Residuals:
     Min       1Q   Median       3Q      Max
-10.7636  -2.2071  -0.3269   1.9714  16.1650

Coefficients:
                           Estimate Std. Error t value Pr(>|t|)
(Intercept)               5.730e+01  2.889e+00  19.835  < 2e-16 ***
horsepower               -2.204e-01  2.828e-02  -7.794 6.09e-14 ***
weight                   -4.712e-03  1.836e-03  -2.566 0.010667 *
displacement             -5.688e-02  1.568e-02  -3.628 0.000324 ***
horsepower:weight         1.510e-05  1.302e-05   1.160 0.246734
horsepower:displacement   3.823e-04  1.112e-04   3.438 0.000650 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.874 on 386 degrees of freedom
Multiple R-squared:  0.7568,    Adjusted R-squared:  0.7536
F-statistic: 240.2 on 5 and 386 DF,  p-value: < 2.2e-16
```

(g) Try a few different transformations of the variables, such as log(X), √X, X2. Comment on your findings.

> lm_auto_fit<-lm(mpg ~ log(horsepower) + sqrt(displacement),data=auto[,1:8])
> summary(lm_auto_fit)

```
Call:
lm(formula = mpg ~ log(horsepower) + sqrt(displacement), data = auto[,
    1:8])

Residuals:
     Min       1Q   Median       3Q      Max
-11.2089  -2.7353  -0.3902   2.2022  15.8070

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)         80.1455     4.6289  17.314  < 2e-16 ***
log(horsepower)     -9.5222     1.3139  -7.247 2.31e-12 ***
sqrt(displacement)  -0.9668     0.1238  -7.810 5.36e-14 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.19 on 389 degrees of freedom
Multiple R-squared:  0.7133,    Adjusted R-squared:  0.7118
F-statistic: 483.9 on 2 and 389 DF,  p-value: < 2.2e-16
```

```
> lm_auto_fit<-lm(mpg ~ log(horsepower) + sqrt(horsepower),data=auto[,1:8])
> summary(lm_auto_fit)

Call:
lm(formula = mpg ~ log(horsepower) + sqrt(horsepower), data = auto[,
    1:8])

Residuals:
    Min      1Q  Median      3Q     Max
-14.608  -2.566  -0.271   2.496  15.211

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       172.991     17.208  10.053  < 2e-16 ***
log(horsepower)   -42.893      6.441  -6.660 9.37e-11 ***
sqrt(horsepower)    4.695      1.237   3.794 0.000172 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.425 on 389 degrees of freedom
Multiple R-squared:  0.6802,    Adjusted R-squared:  0.6785
F-statistic: 413.6 on 2 and 389 DF,  p-value: < 2.2e-16
```

```
> lm_auto_fit<-lm(mpg ~ log(horsepower) + sqrt(weight),data=auto[,1:8])
> summary(lm_auto_fit)

Call:
lm(formula = mpg ~ log(horsepower) + sqrt(weight), data = auto[,
    1:8])

Residuals:
     Min       1Q   Median       3Q      Max
-10.4597  -2.4807  -0.3574   2.2158  15.4697

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      89.2761     3.3768  26.438  < 2e-16 ***
log(horsepower)  -7.9526     1.2365  -6.431 3.71e-10 ***
sqrt(weight)     -0.5431     0.0554  -9.803  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.035 on 389 degrees of freedom
Multiple R-squared:  0.734,     Adjusted R-squared:  0.7327
F-statistic: 536.8 on 2 and 389 DF,  p-value: < 2.2e-16
```

```
> lm_auto_fit<-lm(mpg ~ log(horsepower) + acceleration^2,data=auto[,1:8])
> summary(lm_auto_fit)

Call:
lm(formula = mpg ~ log(horsepower) + acceleration^2, data = auto[,
    1:8])

Residuals:
     Min       1Q   Median       3Q      Max
-12.4273  -2.4701  -0.3334   2.4844  15.5008

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     142.3389     5.2839  26.938  < 2e-16 ***
log(horsepower) -23.1540     0.8658 -26.743  < 2e-16 ***
acceleration     -0.8148     0.1078  -7.562 2.89e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.208 on 389 degrees of freedom
Multiple R-squared:  0.7108,    Adjusted R-squared:  0.7094
F-statistic: 478.1 on 2 and 389 DF,  p-value: < 2.2e-16
```

> lm_auto_fit<-lm(mpg ~ log(horsepower) + sqrt(displacement) + sqrt(horsepower) + sqrt(weight) + acceleration^2,data=auto[,1:8])
> summary(lm_auto_fit)

```
Call:
lm(formula = mpg ~ log(horsepower) + sqrt(displacement) + sqrt(horsepower) +
    sqrt(weight) + acceleration^2, data = auto[, 1:8])

Residuals:
     Min       1Q   Median       3Q      Max
-11.9710  -2.2933  -0.2379   1.9148  15.9136

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)        177.74446   16.56755  10.728  < 2e-16 ***
log(horsepower)    -40.26533    6.05611  -6.649 1.01e-10 ***
sqrt(displacement)  -0.62723    0.18142  -3.457 0.000606 ***
sqrt(horsepower)     5.55327    1.12204   4.949 1.11e-06 ***
sqrt(weight)        -0.18777    0.09832  -1.910 0.056906 .
acceleration        -0.44172    0.12763  -3.461 0.000598 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.87 on 386 degrees of freedom
Multiple R-squared:  0.7573,    Adjusted R-squared:  0.7541
F-statistic: 240.8 on 5 and 386 DF,  p-value: < 2.2e-16
```

Based on the observed graph, it appears that log-transforming the horsepower values results in a more linear relationship between the predictor (horsepower) and the response variable. Log-transforming a variable can help to stabilize variance and make the relationship between variables more linear, particularly if the relationship is exponential in nature. In this case, transforming the horsepower values using a logarithmic function may have mitigated non-linearities or heteroscedasticity present in the original data, resulting in a clearer and more linear relationship between horsepower and the response variable. This transformation technique can be valuable in improving the fit of linear regression models and better capturing the underlying structure of the data.