E1 assignment

2023-11-12

Problem 1 *[6 points]*

In this problem, you will preprocess and explore a data set about atomic structures of molecules. The data set p1.csv is a subset of the GeckoQ data set from the term project.

Lab section 2.3 of ISLR_v2 (or ISLP) contains useful information for solving this problem.

P1-Task a

Task a: Question

Read p1.csv into a data frame and familiarize yourself with the data. You may want to read the data description from the term project.

Drop the columns [“id”, “SMILES”, “InChlKey”].

Task a: My answer

These are the columns remained before dropping three as required

```
##  [1] "id"                            "SMILES"
##  [3] "InChIKey"                      "pSat_Pa"
##  [5] "ChemPot_kJmol"                 "FreeEnergy_kJmol"
##  [7] "HeatOfVap_kJmol"               "MW"
##  [9] "NumOfAtoms"                    "NumOfC"
## [11] "NumOfO"                        "NumOfN"
## [13] "NumHBondDonors"                "NumOfConf"
## [15] "NumOfConfUsed"                 "parentspecies"
## [17] "C=C (non-aromatic)"            "C=C-C=O in non-aromatic ring"
## [19] "hydroxyl (alkyl)"              "aldehyde"
## [21] "ketone"                        "carboxylic acid"
## [23] "ester"                         "ether (alicyclic)"
## [25] "nitrate"                       "nitro"
## [27] "aromatic hydroxyl"             "carbonylperoxynitrate"
## [29] "peroxide"                      "hydroperoxide"
## [31] "carbonylperoxyacid"            "nitroester"
```

These are the columns remained after dropping three as required

```
##  [1] "pSat_Pa"                       "ChemPot_kJmol"
##  [3] "FreeEnergy_kJmol"              "HeatOfVap_kJmol"
##  [5] "MW"                            "NumOfAtoms"
##  [7] "NumOfC"                         "NumOfO"
##  [9] "NumOfN"                         "NumHBondDonors"
## [11] "NumOfConf"                      "NumOfConfUsed"
## [13] "parentspecies"                  "C=C (non-aromatic)"
## [15] "C=C-C=O in non-aromatic ring" "hydroxyl (alkyl)"
## [17] "aldehyde"                       "ketone"
## [19] "carboxylic acid"                "ester"
## [21] "ether (alicyclic)"              "nitrate"
## [23] "nitro"                          "aromatic hydroxyl"
## [25] "carbonylperoxynitrate"          "peroxide"
## [27] "hydroperoxide"                  "carbonylperoxyacid"
## [29] "nitroester"
```

P1-Task b

PQuestion

Select the columns [“pSat_Pa”,“NumOfConf”,“ChemPot_kJmol”] from the data frame and print their summary statistics.

My answer

The summary, after removing the three columns, is Min. : 0.0000 , 1st Qu.: 0.0000 , Median : 0.0001 , Mean : 2.9620 , 3rd Qu.: 0.0023 , Max. :562.8970 , Min. : 2.00 , 1st Qu.: 73.25 , Median : 172.50 , Mean : 223.50 , 3rd Qu.: 324.25 , Max. :1058.00 , Min. :-3.160 , 1st Qu.: 9.723 , Median :12.781 , Mean :12.434 , 3rd Qu.:15.659 , Max. :28.096 .

P1-Task c

Task c: Question

Extract the data in the column ChemPot_kJmol of the data frame to an array. Calculate the mean and standard deviation of this array.

Task c: My answer

Mean is 12.4344271, standard deviation is 4.7788722.

P1-Task d

P1-Task d: Question

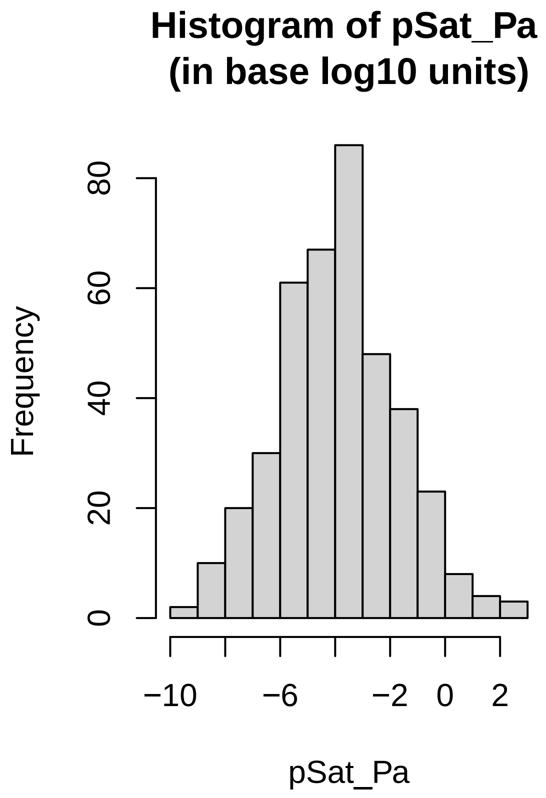
Produce side-by-side plots of:

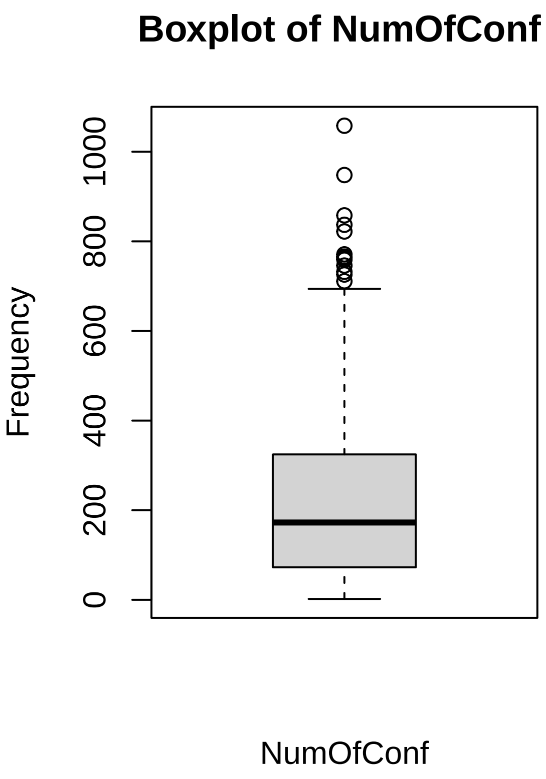
- a histogram of pSat_Sa in base 10 logarithmic units.
- a boxplot of NumOfConf.

Tip: In Python, you can use hist() and boxplot() in the matplotlib package. In R, you can use hist and boxplot. The command par(mfrow=c(1,2)) divides the plot window into two regions so that you can visualize the 2 plots simultaneously.

P1-Task d: My answer

The boxplot and histogram are shown below.





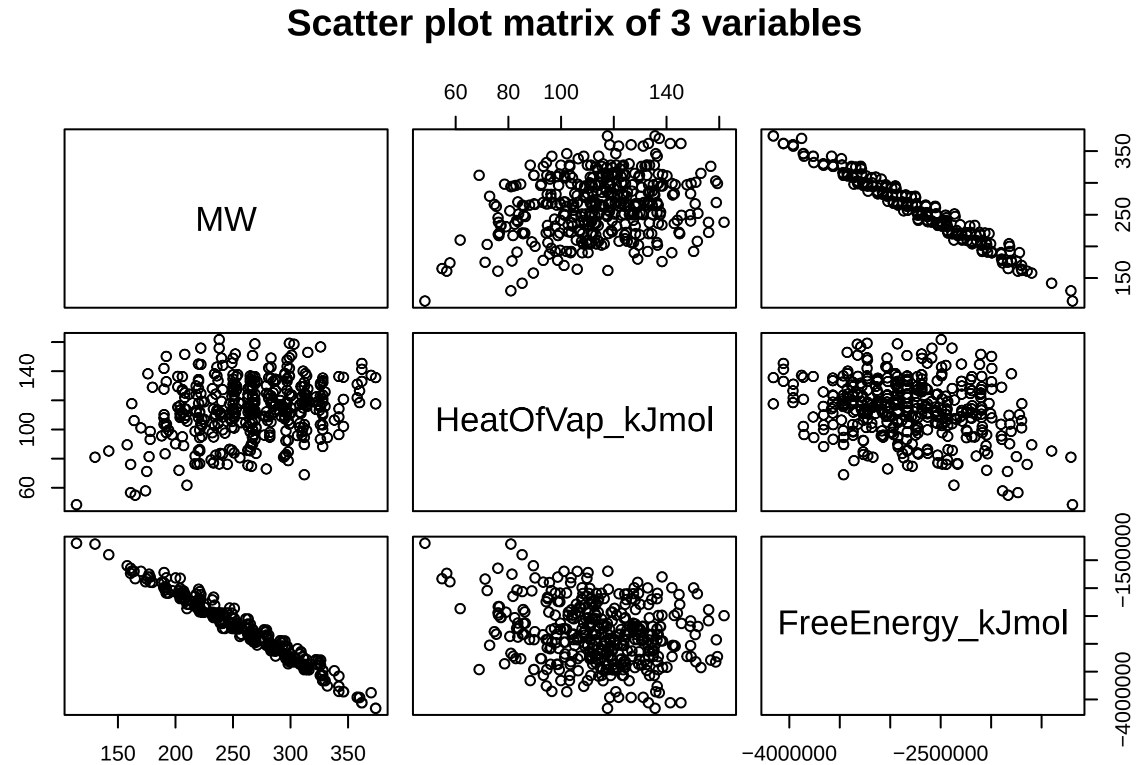
P1-Task e

Question

Produce a scatterplot matrix of the variables [“MW”, “HeatOfVap_kJmol”, “FreeEnergy_kJmol”]. Tip: In Python you can use seaborn.pairplot(). In R, you can use pairs().

My answer

The scatterplot matrix is shown below.



Problem 2 [8 points]

In this problem, you will fit regression models and study their losses. One of the purposes of this problem - in addition to theory - is to make you more comfortable with various machine learning workflows. Sections 5.1 and 5.3.2 (lab section) of ISLR_v2 contain helpful information for solving this problem. Tasks a-b use a synthetic data set, and Tasks c-d use a real data set:

- The synthetic data are given in the CSV files train_syn, valid_syn, and test_syn (the training set, validation set, and test set respectively).

- The real data are meteorological forecasts and geographic data from Cho et al. (2020)1. They are given in the CSV files train_real and test_real (the training set and test set respectively).

P2-Task a

Task a: Question

In this task, you will fit polynomials $y^{=}\Sigma pk = 0wkx$ to the synthetic data for several polynomial degrees p by using ordinary least squares (OLS) regression. Produce the following table: Degree Train Validation Test TestTRVA CV

Task a: my answer

Degree	Train	Validation	Test	TestTRVA	CV
0	4.51226	6.65934	11.71646	11.26322	6.17572
1	4.08854	7.12784	8.87631	9.93494	7.04803
2	0.21859	0.29373	0.24585	0.21402	0.30789
3	0.21682	0.28345	0.29008	0.27511	0.36536
4	0.11880	0.62473	0.96908	0.22399	0.53730
5	0.09653	0.57348	4.89484	1.03909	0.46264
6	0.00757	3.41679	213.29714	0.88147	0.59347
7	0.00500	6.86299	1261.98807	0.27172	0.65830
8	0.00208	401.65178	154266.87136	11.22355	2.03738

I would choose polynomial order = 2. The 5 fold cross-validation set on training and validation set produced a mean MSE 0.30789, which is the tiniest MSE across results from all cross validations.

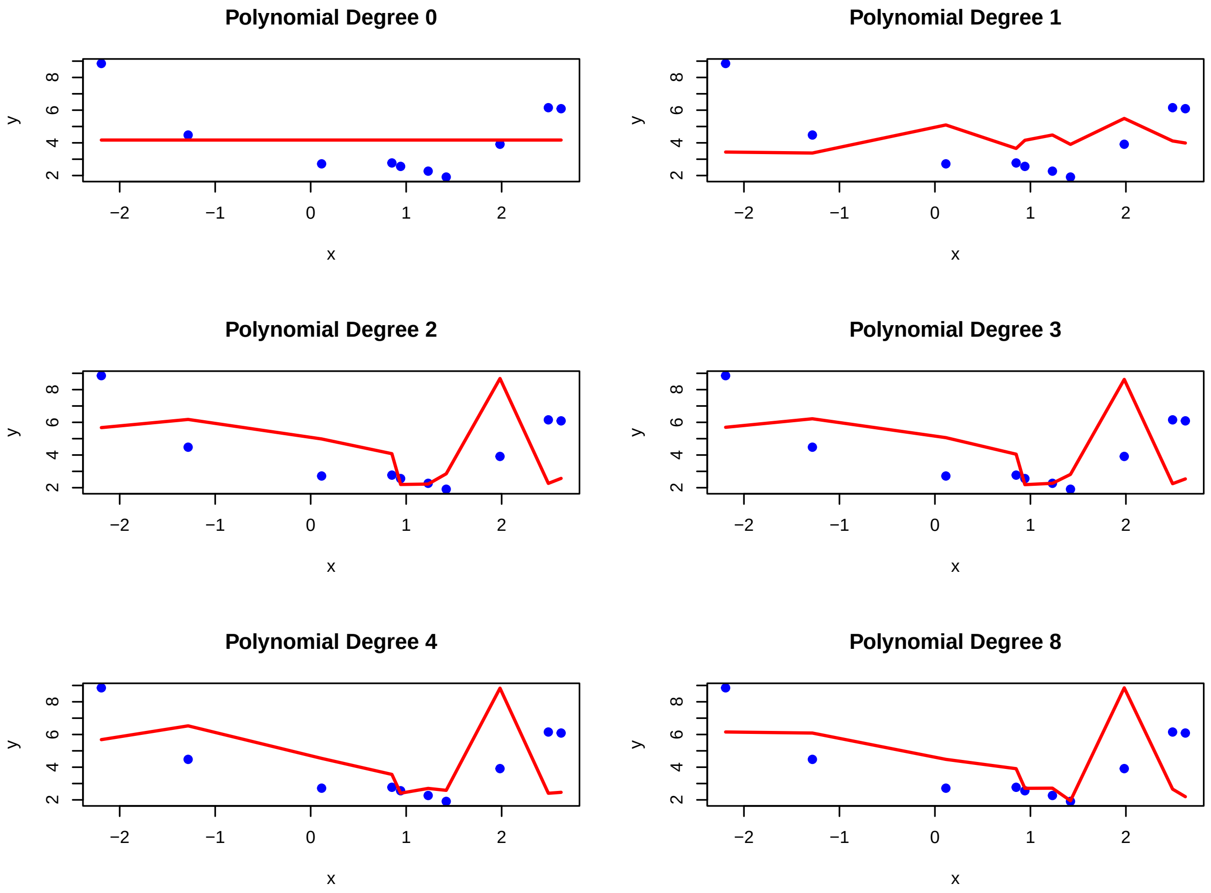
P2-Task b

Task b: Question

For each value of $p \in \{0, 1, 2, 3, 4, 8\}$, produce a plot showing the points (x_i, y_i) in the training set and the fitted polynomial in the interval $[-3, 3]$.

Task b: My answer

The plots required are shown below:



P2-Task c

Task c: Question

In this task, you will fit the following regressors to the real data to predict the next day's maximum temperature (variable Next_Tmax):

- dummy model (see the discussion below) • OLS linear regression (simple baseline)
- random forest (RF)
- support vector regression (SVR)
- one more regression model implemented in your machine learning library not mentioned above. Produce the following table:

where Train is the training loss, Test is the testing loss, and CV is the loss for 10-fold cross-validation. Using the table, answer the following:

1. Which regressor is the best? Why?
2. How does Train compare to Test? How does CV compare to Test?
3. How can you improve the performance of these regressors (on this training set)?

Task c: my answer

I picked up Neural Network (NN) as the last model.

This is the final table required for the question, note that NN is for neural network:

Regressor	Train	Test	CV
Dummy	10.632	9.411	10.421
OLS	2.058	2.337	2.400
RF	0.349	0.479	0.373
SVR	0.923	1.753	1.089
NN	10.632	9.411	10.388

Which regressor is best?

RF is best, since it has lowest values in both testing loss and CV loss, measuring by MSE.

How does Train compare to Test? How does CV compare to Test?

Comparing Training loss to Testing loss, I found OLS, RF and SVR were over-fitted, due to Train < TEST. OLS model underwent the least serious over-fitting, comparing to other two over-fitted models (accoding to the ratio of (Test-Train)/Test).

Comparing Test to CV, SVR model showed relatively large discrepancy between the pair. Other models have fairly small Test and CV difference, indicating good model effectiveness. Again OLS has the smallest Test-CV difference

How can you improve the performance of these regressors (on this training set)?

Several ways that I can come up with:

a. subset selection: select meaningful (theory-driven) and significant (data-driven) variables as predictors. For exmaple, stepwise selection can be used.

b. feature engineering: in addition to subset selection, creating new variables based on the existing ones.

c. Check and remove low quality observations.

How can you improve the performance of these regressors (on this training set)?

Problem 3 [6 points]

In this problem, you will study the bias-variance decomposition in the context of model selection. Section 2.2 of ISLR_v2 will be helpful in solving this problem.

P3-Task a

Task a: Question:

Describe the typical behaviour of the following terms, as we go from less flexible to more flexible statistical learning methods:

- training error and testing error • (squared) bias
- variance
- irreducible (or Bayes) error.

Explain why each term has the described behaviour.

You can describe the behaviours in words or you can sketch them as curves. In the sketches the x-axis should represent the flexibility of the method, and the y-axis should represent the values for each term. There should be five curves in total so make sure to label each one.

Task a: my answer:

a. As model flexibility increases, the training error typically decreases;

b. Testing error generally decreases initially as flexibility increases, but then starts to increase after reaching a certain point. This is because beyond a certain point, increased flexibility leads to overfitting the training data, and the model fails to generalize well to new, unseen data, causing the testing error to increase.

c.As the model becomes more flexible, it can better represent complex relationships, thereby bias tends to decrease as model flexibility increases.

d. More flexible models, which adapt closely to the training data, are likely to capture not only the underlying pattern but also the noise, leading to high variance.

e. Inreducible error is due to the noise in the data itself. No matter how flexible or sophisticated the model is, this component of the error cannot be reduced as it's inherent to the problem and data.

P3-Task b

Task b: Question

(not copied)

Task b: my answer

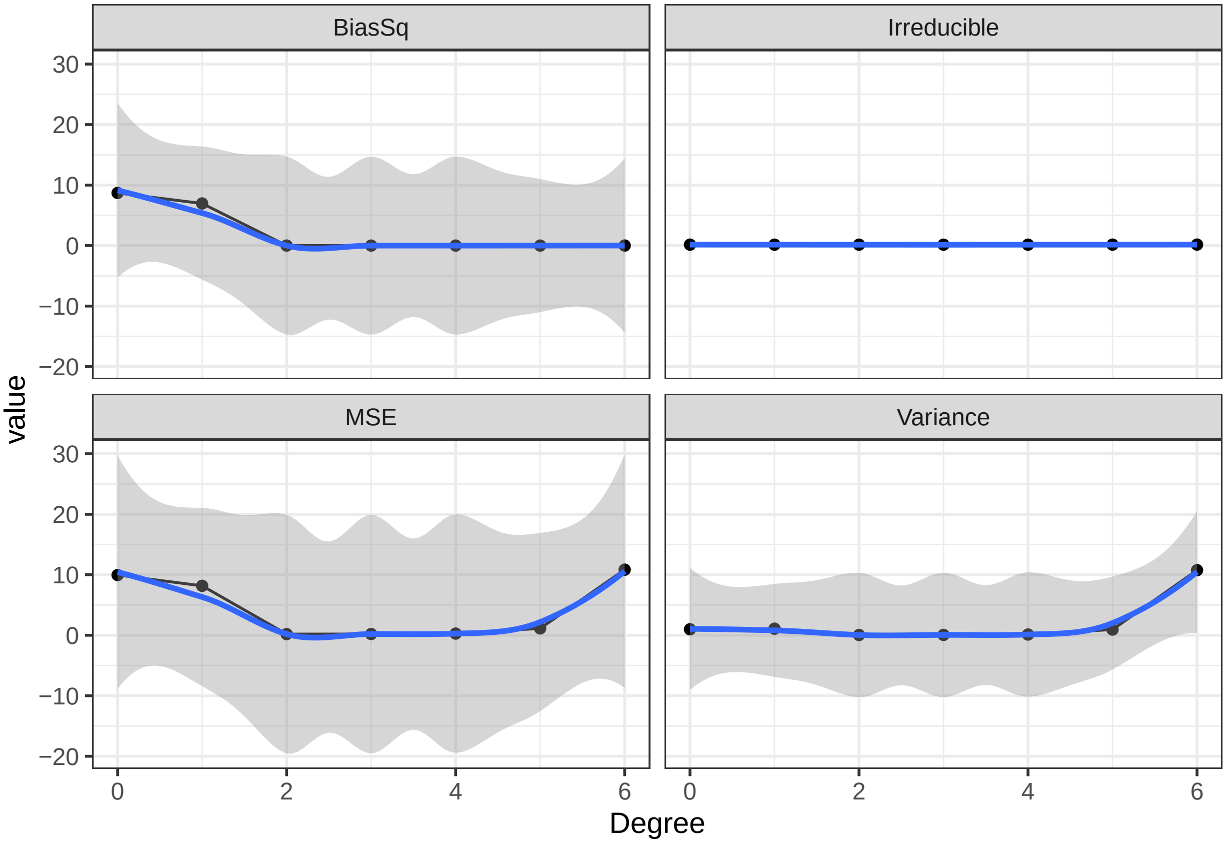
(i)produce the table as required:

Table required:

Degree	Irreducible	BiasSq	Variance	Total	MSE
0	0.15774	8.69803	0.98663	9.84240	9.95568
1	0.15256	6.96064	1.09051	8.20370	8.17293
2	0.15008	0.00027	0.04708	0.19744	0.19385
3	0.14613	0.00004	0.06301	0.20917	0.20458
4	0.15171	0.00006	0.12291	0.27467	0.27620
5	0.15512	0.00360	0.94517	1.10389	1.13238
6	0.16299	0.01135	10.75559	10.92994	10.84646

(ii) plot four terms as a function of poly-nomial degree

The plots are generated as below.



(iii) are the terms as expected?

Yes. They behave as expected. More specifically,

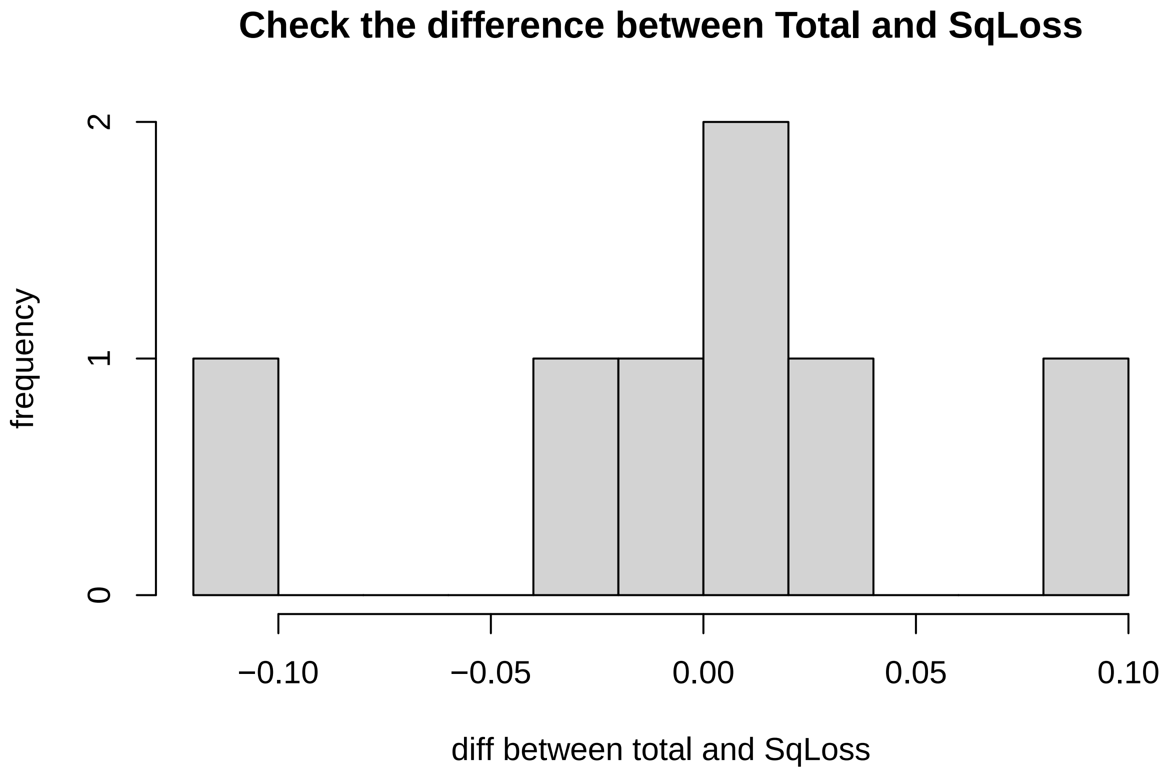
a. Irreducible errors went as expected because their expectation should be constant across different nominal degree models. In addition to that, I saw what I expect to see– an irreducible error roughly close to 0.42ˆ2 = 0.1764, and

b. Squared bias went as expected because it goes down with polynomial degree at the beginning until a point, after which it keeps consistent. This corresponds to the fact that newly-introduced polynomial degrees will not help capture more variance as soon as it has reached the proper polynomial degree.

c. I expected to see a variance that is close to the variance of predicted y value when x = 0, and I did see this.

d. MSE went as expected because it reduces with the reduce of bias and grows again with increasing variance.

And the difference between Total and MSE are small. Most of them range from -0.1 to 0.1. See the plot below:



Problem 5 [6 points]

Objective: properties of estimators

In this problem, we study 1-variable linear regression $y^{=}w0 + w1x$ using the four data sets named d1.csv, d2.csv, d3.csv, and d4.csv.

P5-Task a

Task a: Question

For each data set, fit an OLS linear regression and report:

- the intercept term estimate, standard error, and p-value,
- the slope term estimate, standard error, and p-value,
- the R-squared value of the model.

The slope term may be positive (or negative) with high confidence. Can you safely conclude that when x increases (or decreases), y tends to increase (and vice versa)?

Task a: My answer

Below, I generate a table showing the intercept and slope estimates for the models, and relevant statistics. Note that numbers 1 to 4 in column `model_NO` correspond to models fitted for 4 data sets.

term	estimate	std.error	statistic	p.value	conf.low	conf.high	Rsquared	model_No
(Intercept)	3.00009	1.12475	2.66735	0.02573	0.45574	5.54444	NA	1
slope	0.50009	0.11791	4.24146	0.00217	0.23337	0.76681	0.66654	1
(Intercept)	3.00091	1.12530	2.66676	0.02576	0.45530	5.54652	NA	2
slope	0.50000	0.11796	4.23859	0.00218	0.23315	0.76685	0.66624	2
(Intercept)	3.00245	1.12448	2.67008	0.02562	0.45870	5.54621	NA	3
slope	0.49973	0.11788	4.23937	0.00218	0.23307	0.76639	0.66632	3
(Intercept)	3.00173	1.12392	2.67076	0.02559	0.45924	5.54421	NA	4
slope	0.49991	0.11782	4.24303	0.00216	0.23338	0.76643	0.66671	4

Can you safely conclude that when x increases (or decreases), y tends to increase (and vice versa)

Since the coefficient estimates for the slope of all models are positive and the lower end of 95% CI are above 0, I am confident that when the predictors x increases, the responses y tend to increase.

term	estimate	conf.low	conf.high	model_No
slope	0.50009	0.23337	0.76681	1
slope	0.50000	0.23315	0.76685	2
slope	0.49973	0.23307	0.76639	3
slope	0.49991	0.23338	0.76643	4

Problem 4 [6 points]

Topic: theoretical properties of generalisation loss and OLS linear regression [Ch. 2-3] Consider a linear regression model $f(x) = \beta x$, where $\beta \in R$ is fit by ordinary least squares (OLS) to a set of training data $(x_1, y_1), \dots, (x_n, y_n)$, where the pairs (x_i, y_i) have been drawn at random with replacement from a finite population, where $x_i \in R_p$, $y_i \in R$, and $i \in 1, ..., n$. Suppose we have a testing data $(x_1, y_1), \dots, (x_m, y_m)$ drawn in the same way from the same population. Denote

$L_{train} = \frac{1}{n}\Sigma_{i=1}^{n}(y_i - \hat{\beta^T} x_i)^2$

and

$L_{test} = \frac{1}{n}\Sigma_{i=1}^{m}(\bar{y}_i - \hat{\beta^T} \bar{x}_i)^2$

The expectations below are defined over the resampling of both the training and testing data.

P4-Task a

Task a

Task a: Question

Prove that

$\mathbb{E}[L_{test}] = \mathbb{E}[(\bar{y}_1 - \hat{\beta^T} \bar{x}_1)^2]$

Task a:: My Answer

Expectation of $L_{\mathbf{test}}$:

$$\mathbb{E}[L_{\text{test}}] = \mathbb{E}\left[\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{\beta}^T x_i)^2\right]$$

The expectation is linear, so we have:

$$\mathbb{E}[L_{\text{test}}] = \frac{1}{m} \sum_{i=1}^{m} \mathbb{E}[(y_i - \hat{\beta}^T x_i)^2]$$

Each pair (x_i, y_i) in the test set is drawn independently and identically from the population (i.i.d). Thus, each $(y_i - \hat{\beta}^T x_i)^2$ has the same distribution, and their expectations are equal. Therefore:

$$\mathbb{E}[(y_i - \hat{\beta}^T x_i)^2] = \mathbb{E}[(y_1 - \hat{\beta}^T x_1)^2]$$

for all i. Hence:

$$\mathbb{E}[L_{\text{test}}] = \mathbb{E}[(y_1 - \hat{\beta}^T x_1)^2]$$

## P4-Task b

Task b: Question

Prove that L_{test} is an unbiased estimate of the generalization error for the OLS regression.

Task b: My answer

An unbiased estimate in statistics is an estimator that, on average, correctly targets the parameter it is estimating. In more technical terms, an estimator is unbiased if its expected value is equal to the true value of the parameter it is estimating.

To formalize this, let $\hat{\theta}$ be an estimator of a parameter$theta$. The estimator $\hat{\theta}$ is unbiased if:

$$\mathbb{E}[\hat{\theta}] = \theta$$

As such, in our case, we need to prove:

$$\mathbb{E}[L_{\text{test}}] = \frac{1}{m} \sum_{i=1}^{m} \mathbb{E}[(y_i - \hat{\beta}^T x_i)^2] = \text{expected prediction error of an independent dataset}$$

From the previous question, we already proved:

$$\mathbb{E}[L_{\text{test}}] = \mathbb{E}[(y_i - \hat{\beta}^T x_i)^2]$$

The test samples are drawn from the same finite population we target. Further, the pairs (x_i, y_i) have been drawn at random with replacement from this finite population, where $x_i \in R_p$, $y_i \in R$, and $i \in 1, ..., n$. We can say each $(y_i - \hat{\beta}^T x_i)^2$ is i.i.d, and their expected value represents the expected prediction error. Therefore:

$$\mathbb{E}[L_{\text{test}}] = \text{expected prediction error of an independent dataset}$$

P4-Task c

Task c: Question

Prove that

$$\mathbb{E}[L_{\text{train}}] \leq \mathbb{E}[L_{\text{test}}]$$

.

Task c: My answer

Quit…

P4-Task d

Task d: Question

Explain how the task result above is related to the generalisation problem in machine learning.

Task d: My answer

`Task a`: it proved that the expected error of the model on the entire test dataset is equivalent to the expected error on any single data point from the test set. This information highlights the importance of gernalization problem: a good model should work for any new unseen data. But if we only have one unseen data point we are not able to approximate its expectation. Hence, we used a number of such data point to form a test set.

`Task b`: L_{test} is an unbiased estimate of the generalization error, namely, the average error the model makes on the test data (data not seen during training) reflects its true error on any new data from the same population. In machine learning, one of the key challenges is to assess how well a model will perform on new data. This proven judgment can be seen as theoretical foundation for this assessment.

`Task c`: In machine learning we hope for generalizing a model trained on training data to unseen population data. That is, we want a model that performs well not only on the data it has seen during training but also on new data, the statistics of which can be the expectation of the statistics of a test data (i.i.d). The result $\mathbb{E}[L_{\text{train}}] \leq \mathbb{E}[L_{\text{test}}]$ captures this notion, indicating that models tend to perform better on the data they were trained on.

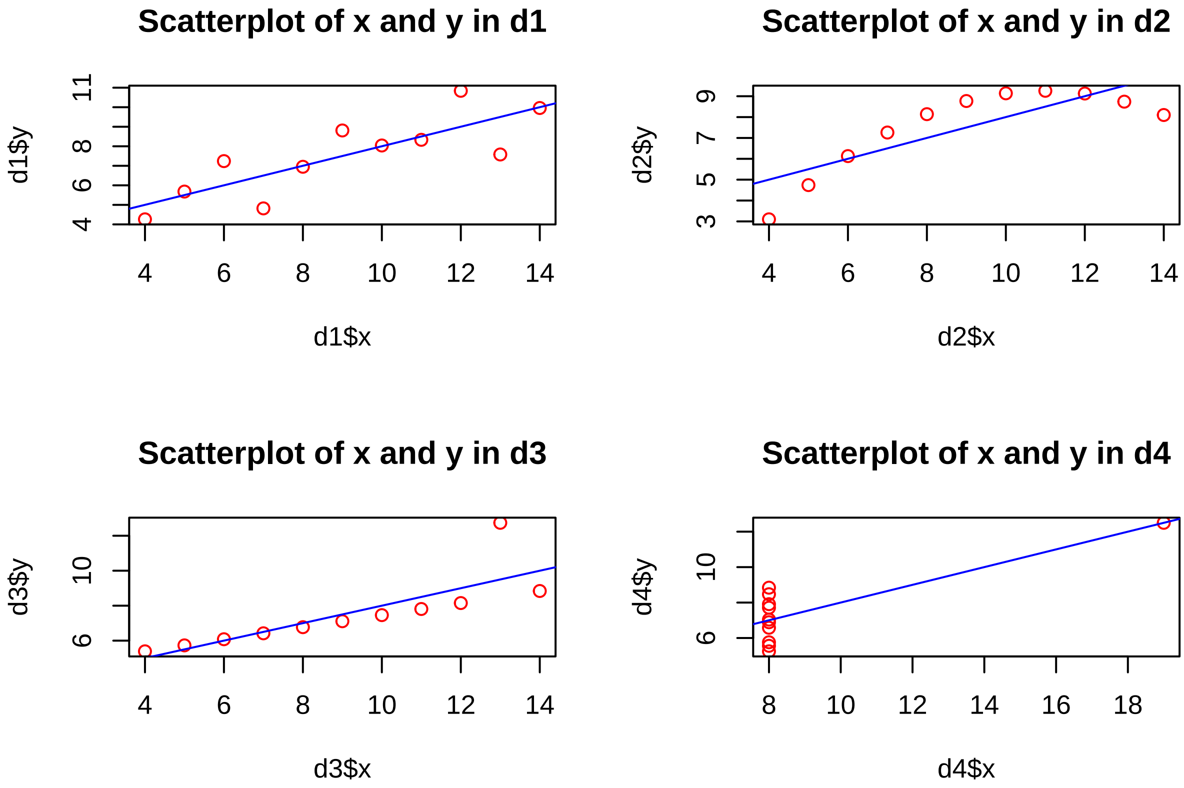
P5-Task b

Task b: Question

Make a plot of each data set showing a scatterplot of x vs y along with the fitted regression line. What commonalities do you notice between the data sets and their fitted models?

Task b: my answer

The plot as required:



commonalities I noticed is:

Even though the points are distributed distinctly across the plots, the fitted lines are roughly the same.

P5-Task c

Task c: Question

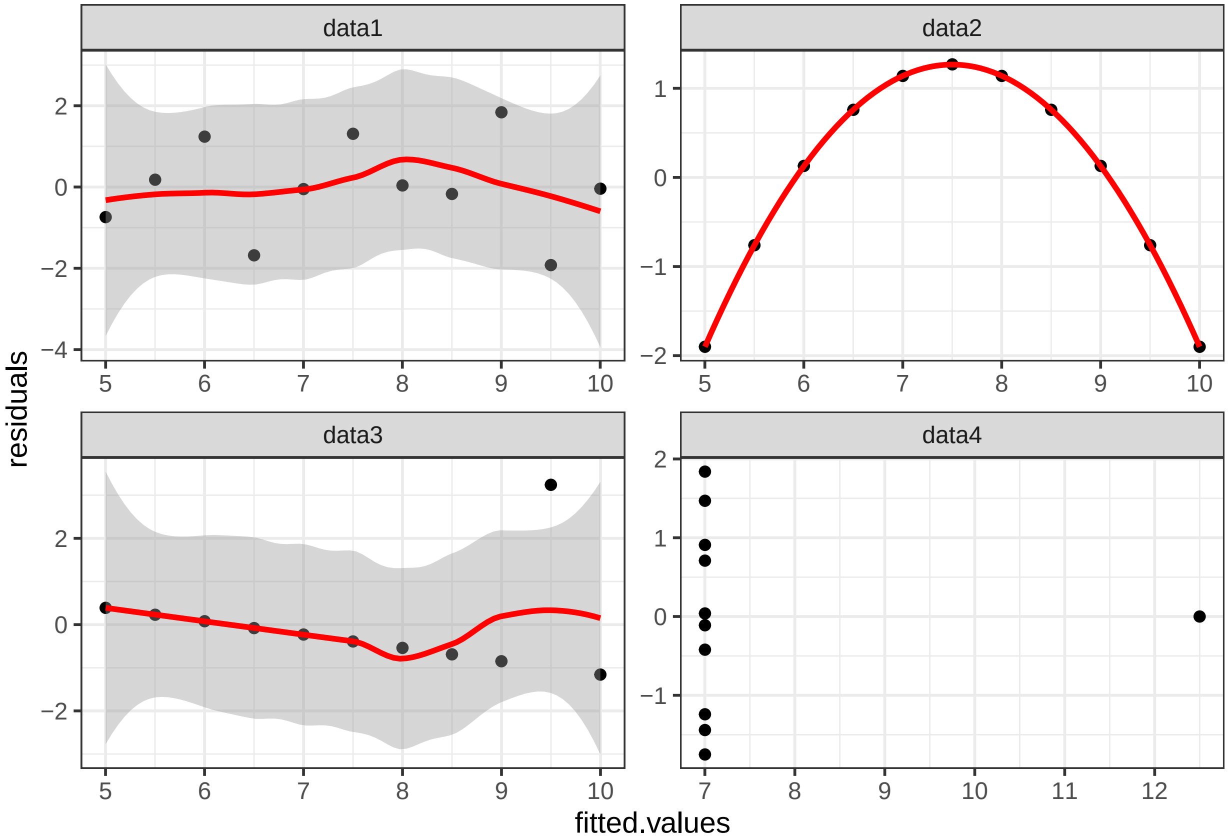
Sect. 3.3.3 of ISLR_v2 lists six potential problems with linear regression models. Which of the six problems would (potentially) apply to each dataset? What tricks and plots did you use to detect and diagnose the problems? Produce at least one diagnostic plot that shows these problems (other than just plotting x vs y, which you can do here because the data is 1-dimensional and which would not work for higher-dimensional data sets).

Task c: My answer

We considered 4 of the 6 assumptions below:

1. Non-linearity of the response-predictor relationships.
2. Correlation of error terms (`not considered since it usually plagues time series data`).
3. Non-constant variance of error terms.
4. Outliers.
5. High-leverage points.
6. Collinearity. (`not considered since it involves multiple x`).

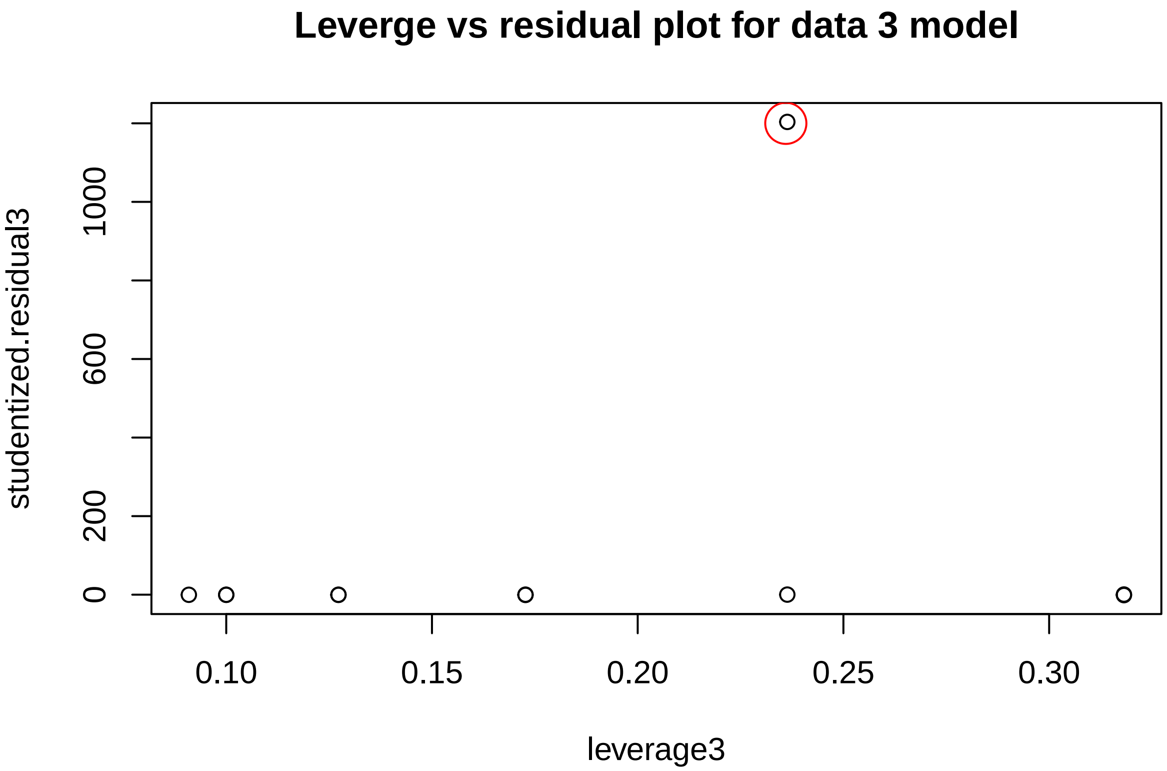
X and y in data 2 and data 4 do not show sufficient linearity in their relationship. And their variance do not appear to be constant. See plots below.



In the table below, for each row except for the first, an x is removed from a dataset each time, refitted, and the MSE change from initial MSE (all x reserved model) is calcuated. Outliers that have large influence on the loss will be detected. I found data 3 suffers from an outlier. Removal of x in row 3 resulted in 99.99% MSE change.

Rmoved.row.number	x.removed	y.removed	MSE	MSE.change
no x removed (initial)	NA	NA	1.25056	0.00000
row 1	10	7.46	1.34325	-0.07412
row 2	8	6.77	1.36973	-0.09529
row 3	13	12.74	0.00001	0.99999
row 4	9	7.11	1.35889	-0.08662
row 5	11	7.81	1.32115	-0.05645
row 6	14	8.84	1.17873	0.05744
row 7	6	6.08	1.37486	-0.09939
row 8	4	5.39	1.35347	-0.08229
row 9	12	8.15	1.28845	-0.03030
row 10	7	6.42	1.37488	-0.09941
row 11	5	5.73	1.36876	-0.09451

Besides, the leverage vs studentized residual plot below also shows there is an out-lier with medium level of leverage data point in data 3 model. See below:

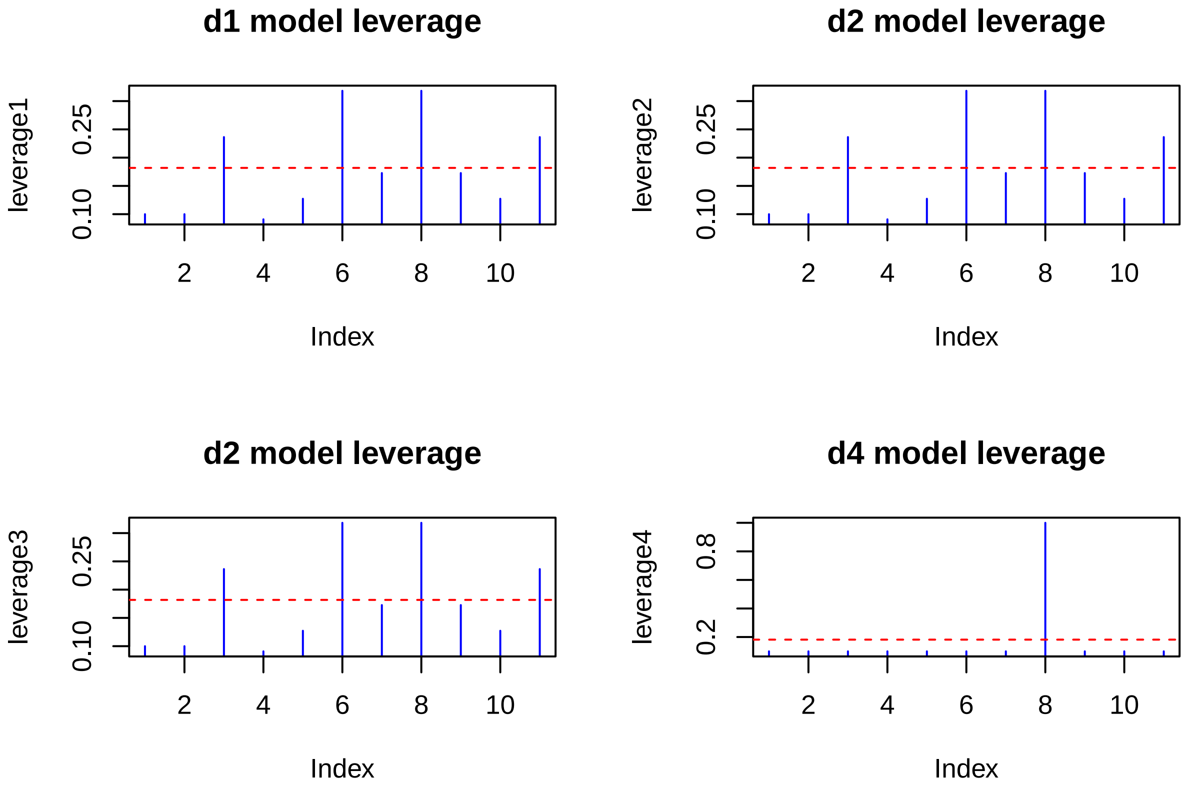


Below I plotted leverage statistics versus index, with a red dotted line showing the reference average value given in p99, ISLR_v2.(

$$\frac{(p+1)}{n}$$

)

The finding is for d4 model, observation 8 has massive leverage that could distort the regression coefficient.



Problem 6 [6 points]

Objective: properties of estimators and Bootstrap [Ch 3 & 5.2]

P6-Task a

Task a: Question

Compute the standard errors for the regression coefficient estimates for the data set d2.csv of the previous problem using bootstrap. Compare the bootstrap standard errors to the ones you got in Task A of the previous problem; which of the estimates is more trustworthy and why?

Task a: My answer

The standard errors for regression coefficient estimates for the data set d2 is 0.1638545.

The previous exercise derived std.error for slope as 0.117963745967641.

Comparing with each other, I found the SE from bootstrap is much larger than SE from conventional computation. Conventional SE computation are based on the assumptions of normality and homoscedasticity of residuals. However, I already discussed in task c of previous question that data 2 violates these assumptions. In such case, bootstrapping method, which does not assume normality and homoscedasticity, should work better and more trustworthy.

P6-Task b

Task b: Question

Describe briefly in your own words how the bootstrap algorithm computes the standard errors for the intercept and slope parameters in the task above.

Task b: my answer

The standard error is the standard deviation of a sample population. It measures the accuracy with which a sample represents a population. Conventional method uses sample statistics to approximate population parameter and computes SE. To extrapolate population from a sample, some data distribution is often assumed, or else it will not work nicely.

In task a, instead, I calculated SE for regression coefficient in a bootstrapping way: I re-sampled n observations with replacement from d2 data for 1000 times. The resulting 1000 sample were fitted using linear regression, respectively. This gave me 1000 'population' of regression coefficients. Computing the Standard Deviation for them and I got the SE, according to definition of SE described above.

P6-Task c:

Task c: Question

In bootstrap, you sample n data points from a population of n points with replacement. Argue that the probability that the j_{th} observation is not in the bootstrap sample is about 0.368 when n is very large.

Task c: my answer

Since replacement is allowed,

$$P(\text{not picked in any draw}) = \left(1 - \frac{1}{n}\right)^n$$

Then,

$$lim_{n\to\infty}\left(1 - \frac{1}{n}\right)^n = \frac{1}{\lim_{n\to\infty}\left(1 + \frac{1}{n}\right)^n} = \frac{1}{e}$$

So we have:

$$\frac{1}{e} \approx \frac{1}{2.71828} \approx 0.3678$$

Problem 7 [2 points]

P7-Task a

Task a: Questions

Write a learning diary of the topics of lectures 1-4 and this exercise set.

What did I learn? What did I not understand? Was there something relevant for other studies or (future) work? The length of your reply should be 1-3 paragraphs of text. You can also give feedback on the course.

Task a: My answer

I am from the disciplines of medicine and educational science. I used to use most of the techniques in chapter 1-4 in my studies, by following how the analyses were done in publications, without realizing they are rooted in the power of machine learning. Thanks to these sessions, I finally systematically learnt them and got my mind much better organzied.

I used to be confused about training, validation and testing data sets, but now I see how they relate to each other in a more structured way.

I used to select predictors in a model using step-wise method. Now I get a bigger picture that it belongs to subset selection and there are some competing techniques. And I see the reason why it was developed: for the sake of efficiency.

I used to determine the polynomial degree of my model by observing the lowess curve with my bare eyes, but now I get a more powerful data-driven approach: polynomial degree comparison and selection with loss function.

Gladly that this course is not that mathematically heavy, at least until now, or else I might not be able to follow. But still, in the sessions I found some math blind spots (such as optimization) and I will make them up by joining more basic level courses, after this course.

P7-Task b

Task b: Question

Give an estimate of the hours used in solving the problems in this exercise set.

Task b: My answer

I used a day from 10:00 to 17:00 getting 4 questions done. I used another 3~4 hours of another day to get the remaining done. (I haven't done question 4 yet; I might quit it). In all, I spent ~10 hours solving the problems.