

We now consider the `cylinders` variable. Typing in `Auto.cylinders.dtype` reveals that it is being treated as a quantitative variable. However, since there is only a small number of possible values for this variable, we may wish to treat it as qualitative. Below, we replace the `cylinders` column with a categorical version of `Auto.cylinders`. The function `pd.Series()` owes its name to the fact that `pandas` is often used in time series applications.

`pd.Series()`

```
In [106]: Auto.cylinders = pd.Series(Auto.cylinders, dtype='category')
Auto.cylinders.dtype
```

Now that `cylinders` is qualitative, we can display it using the `boxplot()` method.

`.boxplot()`

```
In [107]: fig, ax = subplots(figsize=(8, 8))
Auto.boxplot('mpg', by='cylinders', ax=ax);
```

The `hist()` method can be used to plot a *histogram*.

`.hist()`

```
In [108]: fig, ax = subplots(figsize=(8, 8))
Auto.hist('mpg', ax=ax);
```

The color of the bars and the number of bins can be changed:

```
In [109]: fig, ax = subplots(figsize=(8, 8))
Auto.hist('mpg', color='red', bins=12, ax=ax);
```

See `Auto.hist?` for more plotting options.

We can use the `pd.plotting.scatter_matrix()` function to create a *scatterplot matrix* to visualize all of the pairwise relationships between the columns in a data frame.

`pd.plotting.  
scatter_  
matrix()`

```
In [110]: pd.plotting.scatter_matrix(Auto);
```

We can also produce scatterplots for a subset of the variables.

```
In [111]: pd.plotting.scatter_matrix(Auto[['mpg',
                                           'displacement',
                                           'weight']]);
```

The `describe()` method produces a numerical summary of each column in a data frame.

`.describe()`

```
In [112]: Auto[['mpg', 'weight']].describe()
```

We can also produce a summary of just a single column.

```
In [113]: Auto['cylinders'].describe()
Auto['mpg'].describe()
```

To exit Jupyter, select `File / Close and Halt`.

## 2.4 Exercises

### Conceptual

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.
  - (b) The number of predictors  $p$  is extremely large, and the number of observations  $n$  is small.
  - (c) The relationship between the predictors and response is highly non-linear.
  - (d) The variance of the error terms, i.e.  $\sigma^2 = \text{Var}(\epsilon)$ , is extremely high.
2. Explain whether each scenario is a classification or regression problem, and indicate whether we are most interested in inference or prediction. Finally, provide  $n$  and  $p$ .
  - (a) We collect a set of data on the top 500 firms in the US. For each firm we record profit, number of employees, industry and the CEO salary. We are interested in understanding which factors affect CEO salary.
  - (b) We are considering launching a new product and wish to know whether it will be a *success* or a *failure*. We collect data on 20 similar products that were previously launched. For each product we have recorded whether it was a success or failure, price charged for the product, marketing budget, competition price, and ten other variables.
  - (c) We are interested in predicting the % change in the USD/Euro exchange rate in relation to the weekly changes in the world stock markets. Hence we collect weekly data for all of 2012. For each week we record the % change in the USD/Euro, the % change in the US market, the % change in the British market, and the % change in the German market.
3. 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).

4. You will now think of some real-life applications for statistical learning.
  - (a) Describe three real-life applications in which *classification* might be useful. Describe the response, as well as the predictors. Is the goal of each application inference or prediction? Explain your answer.
  - (b) Describe three real-life applications in which *regression* might be useful. Describe the response, as well as the predictors. Is the goal of each application inference or prediction? Explain your answer.
  - (c) Describe three real-life applications in which *cluster analysis* might be useful.
5. What are the advantages and disadvantages of a very flexible (versus a less flexible) approach for regression or classification? Under what circumstances might a more flexible approach be preferred to a less flexible approach? When might a less flexible approach be preferred?
6. 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 non-parametric approach)? What are its disadvantages?
7. The table below provides a training data set containing six observations, three predictors, and one qualitative response variable.

Obs.	$X_1$	$X_2$	$X_3$	$Y$
1	0	3	0	Red
2	2	0	0	Red
3	0	1	3	Red
4	0	1	2	Green
5	-1	0	1	Green
6	1	1	1	Red

Suppose we wish to use this data set to make a prediction for  $Y$  when  $X_1 = X_2 = X_3 = 0$  using  $K$ -nearest neighbors.

- (a) Compute the Euclidean distance between each observation and the test point,  $X_1 = X_2 = X_3 = 0$ .
- (b) What is our prediction with  $K = 1$ ? Why?
- (c) What is our prediction with  $K = 3$ ? Why?
- (d) If the Bayes decision boundary in this problem is highly non-linear, then would we expect the *best* value for  $K$  to be large or small? Why?

*Applied*

8. This exercise relates to the `College` data set, which can be found in the file `College.csv` on the book website. 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 `Python`, it can be viewed in Excel or a text editor.

- (a) Use the `pd.read_csv()` function to read the data into `Python`. Call the loaded data `college`. Make sure that you have the directory set to the correct location for the data.
- (b) Look at the data used in the notebook by creating and running a new cell with just the code `college` in it. You should notice that the first column is just the name of each university in a column named something like `Unnamed: 0`. We don't really want `pandas` to treat this as data. However, it may be handy to have these names for later. Try the following commands and similarly look at the resulting data frames:

```
college2 = pd.read_csv('College.csv', index_col=0)
college3 = college.rename({'Unnamed: 0': 'College'},
                          axis=1)
college3 = college3.set_index('College')
```