

## CHAPTER OVERVIEW

### 7: Summary

LINEAR regression modeling is one of the most basic of a broad collection of data mining techniques. It can demonstrate the relationships between the inputs to a system and the corresponding output. It also can be used to predict the output given a new set of input values. While the specifics for developing a regression model will depend on the details of your data, there are several key steps to keep in mind when developing a new model using the R programming environment:

#### 1. Read your data into the R environment.

As simple as it sounds, one of the trickiest tasks oftentimes is simply reading your data into R. Because you may not have controlled how data was collected, or in what format, be prepared to spend some time writing new functions to parse your data and load it into an R data frame. Chapter 6 provides an example of reading a moderately complicated csv file into R.

#### 2. Sanity check your data.

Once you have your data in the R environment, perform some sanity checks to make sure that there is nothing obviously wrong with the data. The types of checks you should perform depend on the specifics of your data. Some possibilities include:

- Finding the values' minimum, maximum, average, and standard deviation in each data frame column.
- Looking for any parameter values that seem suspiciously outside the expected limits.
- Determining the fraction of missing ( `NA` ) values in each column to ensure that there is sufficient data available.
- Determining the frequency of categorical parameters, to see if any unexpected values pop up.
- Any other data-specific tests.

Ultimately, you need to feel confident that your data set's values are reasonable and consistent.

#### 3. Visualize your data.

It is always good to plot your data, to get a basic sense of its shape and ensure that nothing looks out of place. For instance, you may expect to see a somewhat linear relationship between two parameters. If you see something else, such as a horizontal line, you should investigate further. Your assumption about a linear relationship could be wrong, or the data may be corrupted (see item no. 2 above). Or perhaps something completely unexpected is going on. Regardless, you must understand what might be happening before you begin developing the model. The `pairs()` function is quite useful for performing this quick visual check, as described in Section 4.1.

#### 4. Identify the potential predictors.

Before you can begin the backward elimination process, you must identify the set of all possible predictors that could go into your model. In the simplest case, this set consists of all of the available columns in your data frame. However, you may know that some of the columns will not be useful, even before you begin constructing the model. For example, a column containing only a few valid entries probably is not useful in a model. Your knowledge of the system may also give you good reason to eliminate a parameter as a possible predictor, much as we eliminated TDP as a possible predictor in Section 4.2, or to include some of the parameters' non-linear functions as possible predictors, as we did when we added the square root of the cache size terms to our set of possible predictors.

#### 5. Select the predictors.

Once you have identified the potential predictors, use the backward elimination process described in Section 4.3 to select the predictors you'll include in the final model, based on the significance threshold you decide to use.

#### 6. Validate the model.

Examine your model's  $R^2$  value and the adjusted- $R^2$  value. Use residual analysis to further examine the model's quality. You also should split your data into training and testing sets, and then see how well your model predicts values from the test set.

#### 7. Predict.

Now that you have a model that you feel appropriately explains your data, you can use it to predict previously unknown output values.

A deep body of literature is devoted to both statistical modeling and the R language. If you want to learn more about R as a programming language, many good books are available, including [11, 12, 15, 16]. These books focus on specific statistical ideas and use R as the computational language [1, 3, 4, 14]. Finally, this book [9] gives an introduction to computer performance measurement.

As you continue to develop your data-mining skills, remember that what you have developed is only a model. Ideally, it is a useful tool for explaining the variations in your measured data and understanding the relationships between inputs and output. But like all models, it is only an approximation of the real underlying system, and is limited in what it can tell us about that system. Proceed with caution.

---

This page titled [7: Summary](#) is shared under a [CC BY-NC 4.0](#) license and was authored, remixed, and/or curated by [David Lilja \(University of Minnesota Libraries Publishing\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.