[DS-05] Linear regression

Miguel-Angel Canela
Associate Professor, IESE Business School

Introduction

In Data Science, one of the main jobs is **prediction**, that is, the description of one attribute (Y) in terms of the other attributes (X's). In the Machine Learning context, developing a predictive model is called **supervised learning**. The term **regression** applies to the prediction of a numeric variable, and **classification** to the prediction of a categorical variable. In an example of regression, we may try to predict the price of a house from a set of attributes of this house. In one of classification, to predict whether a customer is going to quit or not, from his/her demographics plus some measures of customer activity.

Regression models are not necessarily related to a mathematical equation, as in Statistics, although an equation is the first idea that comes to our mind when we think about "predicting". When the equation is linear, we have **linear regression**, which is the object of this note. The predictions of a linear regression model can typically be improved by more sophisticated techniques, but most data scientists start there, because it helps them to understand the data.

Two alternatives to linear regression are:

- **Regression trees**. Although in this course tree algorithms are discussed in a classification context, some of them, like the CART algorithm used in the `R` package `rpart` and the Python package `scikit-learn`, also apply to regression trees.
- **Neural networks**. A neural network is a programming device whose design is based on the models developed by neurologists for the brain neurons. Among the many types of neural networks, the most popular is the **multilayer perceptron** (MLP), which can be regarded as a set of (nonlinear) regression equations. Neural networks find their real power with big data sets. This is specially true for **deep learning** methods, based on networks with many layers of equations.

Evaluation of a linear regression model

In general, regression models are evaluated through their prediction errors. The basic schema is

$$\text{Prediction error} = \text{Actual value} - \text{Predicted value}.$$

Prediction errors are called **residuals** in linear regression. In that special case, the mean of the prediction errors is zero, which is no longer true in other models. The standard algorithm for calculating the regression coefficients is the **least squares method**, which minimizes the sum of the squared residuals.

Statisticians look at the **residual sum of squares** for evidence of good fit between the model and the data. The **R-squared statistic** is a standardized measure which operationalizes this. More specifically, we take advantage of the formula

$$\text{var}\big(\text{Actual values}\big) = \text{var}\big(\text{Predicted values}\big) + \text{var}\big(\text{Prediction error}\big)$$

to evaluate the model through the proportion of **variance explained**,

$$R^2 = \frac{\text{var}\big(\text{Predicted values}\big)}{\text{var}\big(\text{Actual values}\big)}\,.$$

It turns out that the square root of R-squared coincides with the **correlation** between actual values and predicted values, called the multiple correlation in Statistics textbooks. Although this stops being true for other regression methods, this correlation is still the simplest approach to the evaluation of a regression model.

References

1. MJ Crawley (2012), *The R Book*, Wiley. Free access at `ftp.tuebingen.mpg.de/pub/kyb/bresciani`.
2. F Provost & T Fawcett (2013), *Data Science for Business — What You Need to Know About Data Mining and Data-Analytic Thinking*, O'Reilly.
3. J VanderPlas (2017), *Python Data Science Handbook*, O'Reilly.