Data Analysis and Model Classification
Guidesheet VII: Supplementary material

Christoph Schneider Ping-Keng Jao Stéphanie Martin
Ruslan Aydarkhanov Ricardo Chavarriaga

November 20, 2016

Regression

Regression methods attempt to predict the value of an output variable y (dependent variable), given the values of one or more input (independent) variables $\mathbf{x} = \{x_1, x_2, ..., x_N\}$. Therefore, it attempts to define a mapping $f : y \leftarrow f(\mathbf{x})$ such that the error between $\mathbf{Y}$ and $f(\mathbf{X})$ is minimized, where $\mathbf{X}$ is a set of samples given and $\mathbf{Y}$ is their known respective output values. To sum up, in a regression problem y is a continuous variable whose values are predicted from measured input variables $\mathbf{x}$, whereas, in a classification problem, y is a discrete variable denoting the type of class we want to assign a sample $\mathbf{x}$.

Linear Regression

Linear Regression methods are the most simple and popular methods for regression. They define a mapping $f : \mathbf{y} \leftarrow f(\mathbf{x}, \mathbf{w})$, which is linear[1] with respect to the computed parameters $\mathbf{w}$ (**not necessarily linear w.r.t. the input variables x, though**). Intuitively, the regression function $f(\mathbf{x}, \mathbf{w})$ will be linear with respect to $\mathbf{w}$, if f is a linear combination of the elements of $\mathbf{w}$ (e.g. no multiplicative factors $w_i w_j$ appear in f). Summarizing the above, linear regression methods predict the value y_n given an input vector $\mathbf{x}_n$ through a linear function $f(\mathbf{x}_n, \mathbf{w})$:

$$y_n = f(\mathbf{x_n}, \mathbf{w}) = \sum_{i=0}^{P} w_i \phi_i(\mathbf{x_n}) \tag{1}$$

where $\phi(\mathbf{x})$ is any function (even a non-linear one) of the input vector $\mathbf{x}$, called the *basis* function. Under this perspective, a linear regressor attempts to estimate y_n through a linear combination of basis functions of the input variable(s). In the following, given the general linear regression model of Eq. 1, we will consider the following cases:

- Univariate linear regression : Unique input variable x (a scalar x, not a vector $\mathbf{x}$) and $\phi_0(x) = 1$, $\phi_1(x) = x$.
- Polynomial regression : Unique input variable x and $\phi_i = x^i$ (Note that previous case was special case of the polynomial with P=1)
- Multivariate polynomial regression : N input variables $\mathbf{x} = \{x_1, x_2, \ldots, x_N\}$, $\phi_0(\mathbf{x}) = 1$ and $\phi_i(\mathbf{x}) = \mathbf{x}^i \stackrel{\text{def}}{=\!=} \{x_1^i, x_2^i, \ldots, x_N^i\}, \forall i \in \mathbf{N}$

[1] a function $f(x)$ is linear with respect to x, if $f(ax_1 + bx_2) = af(x_1) + bf(x_2)$

Univariate Linear Regression

The univariate linear regression model as derived by the generic linear regression model will predict each coordinate y_n, as:

$$y_n = w_0 + w_1 x_n \tag{2}$$

We will respectively build a model for each coordinate. As can be seen in Eq. 2, the simple univariate linear regression model tries to predict the output variable assuming a linear relation between the input and the output (remember from previous exercises that Eq. 2 is the equation of a line in the plane formed by variables a and x)

More precisely, as said in the lecture slides, linear regression assumes that the output variable is a linear transformation of the input variable, where a random noise ϵ (with zero mean and variance σ^2) is added: $y_n = w_0 + w_1 x_n + \epsilon$.

Assuming a normal distribution for the noise ϵ the distribution of the output variable y, $f(y) = p(y|x, \mathbf{w})$ will also be normally distributed with mean $w_0 + w_1 x$ and variance σ^2. With this distribution, we can compute the likelihood of our dataset as $\prod_{n=1}^{N} p(y_n|x_n, w)$. It is found that maximizing this likelihood is equivalent to minimizing the Sum-of-Squares Error: $E = \sum_{i=1}^{N}(y_n - w_0 - w_1 x_n)^2$ (squares of the differences between the predictions and the actual values). With this reasoning, the weights w_0, w_1 that minimizes the error are given by:

$$w_0 = \frac{1}{N}\sum_{n=1}^{N} y_n - w_1 \frac{1}{N}\sum_{n=1}^{N} x_n \tag{3}$$

and

$$w_1 = \frac{\frac{1}{N}\sum_{n=1}^{N} x_n y_n - \frac{1}{N}\sum_{n=1}^{N} x_n \frac{1}{N}\sum_{n=1}^{N} y_n}{\sum_{n=1}^{N}(x_n)^2 - (\frac{1}{N}\sum_{n=1}^{N} x_n)^2} \tag{4}$$

Polynomial Linear Regression

Another choice of a linear regression model could be formed by using the powers of the input up to a certain degree P, instead of just using the input x_n and the constant term. In this case, instead of fitting a line to the input/output pairs, we are attempting to fit a polynomial of degree P by introducing $P+1$ basis functions $\phi_i(x_n) = x_n^i$.

The linear regression model in this case will have the form:

$$y_n = f(\mathbf{x_n}, \mathbf{w}) = w_0 + w_1 x_n^1 + w_2 x_n^2 + \cdots + w_P x_n^P \tag{5}$$

This function is P_{th} order polynomial (the degree of the polynomial is P), which gives this linear regression method its name. Note that this is still a linear regression model since the function f is linear with respect to $\mathbf{w}$, it is just (some of) the basis functions $\phi(\mathbf{x})$ that are non-linear. Another important issue is that by introducing the powers of the input variable into the model, in fact we are turning it to a multivariate regression problem. Therefore, the weights $\mathbf{w}$ of the model can be computed by the formula:

$$w = (X^T X)^{-1} X^T y \tag{6}$$

The matrix $(X^T X)^{-1} X^T$ is called the (Moore-Penrose) *pseudoinverse* of the data matrix X. In the polynomial case, the j^{th} column of the matrix X contains the $(j-1)^{th}$ power of the input variable, where the first column contains the constant 1 for all samples.