Data learning 5 : Linear and logistic regression

In the previous lectures, we have studied three methods of multivariate analysis, Principal Component Analysis, Determinant Analysis and Cluster Analysis, using mainly a geometrical approach. Indeed the individuals were regarded as points, members of a cloud of points, and not as a value taken by a random vector $(x^1, x^2, \ldots, x^p)$ and the data matrix also as been regarded mainly as a table of coordinates and not as a sample of n values of a random vector. As long as we are just interested in finding a formula for the various parameters (finding the best subspace, finding the best partition, ...), it is not necessary to introduce a stochastic model ; the problem is just an optimisation problem. But as soon as we want to draw inferences about parameters, namely to test the adequacy of the output to the data (does the subspace fit well ? are the clusters well separated ?), additional assumptions on the data are needed, and especially a convenient statistical model has to be chosen. Never forget that with such multivariate methods, the user will always obtain a result, even if it fit very poorly with the data.

As an example of these two points of vue (geometric (without model) / considering a statistic model), let us recall the *simple regression* method. Typically we have a set of training data $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ and we would like to predict the value of y (*to be explain variable*) from the one of x (*explicative variable* or *regressor*). The most easy way to do this is to find the *best* straight line that fit the data, regarded as a set of points in the plan. There are many methods to find this line, $y = \beta_0 + \beta x$, but the most popular is the *least square* method that consists in picking the coefficients β_0 and β such that they minimize the *sum* of *square of errors* :

$$SSE(\beta) = \sum_{i=1}^{n} (y_i - (\beta_0 + bx_i))^2.$$

As $SSE(\beta)$ is a quadratic and convexe function of the β's, it has a unique minimum that can be computed easily as an optimisation problem that can be solved without using any statistic model.

On the other hand, we can also assume that the data satisfy the following statistical model, called the *linear model :* assume that X and Y are two r.v. and assume that the data are a random sample of the random vector (X, Y). Assume that the regression line is a model for the conditional expectation $\mathbb{E}(Y/X) = \beta_0 + \beta x$ and thus consider the following model $Y = \beta_0 + \beta x + \varepsilon$, where ε is an independant of X random variable such that $E(\varepsilon) = 0$ and $\mathrm{Var}\,(\varepsilon) = \sigma^2$.

The linear model : More generally, given a $(p+1)$-dim random vector $(Y, X^1, X^2, \ldots, X^p)$, the linear model is defined by :

$$Y = \beta_0 + \sum_{j=1}^{p} X^j \beta_j + \varepsilon$$

where ε is as above. β_0 is called the *intercept.* It is often convenient to include the constant variable 1 in $(X^1, X^2, \ldots, X^p)$ and to write the model $Y = X'\beta + \varepsilon$ where $\beta = (\beta_0, \beta_1, \ldots, \beta_p)$ and X' is the transpose of the matrix having 1 and the X^j as column. Given a data set $\{(y_i, 1, x_i^1, \ldots, x_i^p), i = 1, \ldots, n\}$, the β^j's are obtained by minimization of

$$SSE(\beta) = \sum_{i=1}^{n} (y_i - (\beta_0 + x_i'\beta))^2 = \|Y - X'\beta\|^2.$$

The solution is easy to compute in matrix notation. Indeed, as $SSE(\beta) = (y - X\beta)'(Y - X\beta)$, differentiating w.r.t. β, we get $-2X'(Y - X\beta) = 0$. If $X'X$ is non singular (if not, just reduce the number of variables), the unique solution is given by $\hat{\beta} = (X'X)^{-1}X'Y$ and the predicted value of Y given X, denoted by $\hat{Y}$, is $\hat{Y} = X\hat{\beta} = (X(X'X)^{-1}X')Y$. The matrix $H = X(X'X)^{-1}X'$ is called the *hat matrix* (because it put a hat on Y).

Geometrically the linear regression of y can be viewed as the orthogonal projection of y on the subspace $Vect(X)$ generated by the variables. The hat matrix is the matrix of this projection.

One can show that each $\hat{\beta}_j$ is an unbiaised estimator of β_j and even that it is the best linear unbiaised estimators (BLUE) in the sens of minimal variance. Moreover the mean square error

$$\hat{\sigma}^2 = MSE = \frac{SSE}{n - p - 1}$$

is an unbiaised estimator of the variance σ^2 of the residuals ε and the j^{th} diagonal coefficient of the matrix $\hat{\sigma}^2(X'X)^{-1}$ gives an estimation of the variance $\hat{s}_j$ of the estimators $\hat{\beta}_j$.

If we assume in addition that the r.v. ε has a gaussian distribution $\mathcal{N}(0,\sigma)$ then the r.v. $\hat{\beta}_j - \beta_j$ has a $(n-p-1)$-Student ditribution and then one can test the hypothesis H_0 : "$\beta_j = b$" or defined a confident interval for each β_j. And it is also possible to test the hypothesis H_0 : "$\beta_0 = \beta_1 = \ldots = \beta_p = 0$" using a Fischer's test (the part of the total variance of Y explained by the regression $\hat{Y}$ (MRS/MSE) has a Fischer distribution).

Logistic regression : A *categorical* variable can assume only a finite number of discrete values. It may be *nominal* like gender (M/F) or *ordinal* when the observed values are ordered. But even in the ordinal case, one can not hope to *explain* such a categorical variable Y by a family of explicatives variables $(X^1, X^2, \ldots, X^p)$ using a linear regression. Indeed, as image of a linear function, the values of Y described by a linear regression will never be discrete values. In that case, one has to use a *logistic regression*.

Assume for simplicity that Y can be codded as 0 and 1. In the logistic model, we assume that Y is a Bernoulli r.v. and thus the conditional probabilities p=p(X)=$\mathbb{P}$(Y=0/X) are numbers belonging to $(0,1)$. Thus the image of p by any function Φ mapping $(0,1)$ to $\mathbb{R}$ will belong to $\mathbb{R}$. The idea of the logistic regression is to apply a linear model not to Y but to $\Phi(p)$:

$$\Phi(p) = \beta_0 + \beta_1 X^1 + \ldots + \beta_p X^P + \varepsilon$$

Different *link functions* $\Phi(p)$ may be chosen. Usually one take the *logit* function $\Phi(p) = \ln\frac{p}{1-p}$, but it can also be the *log-log* function $\Phi(p) = \ln(-\ln(1-p))$ or the *probit* function which is the inverse of the gaussian distribution. Notice that by the Bayes formula, the conditional probability $p = p(X)$ can be express

$$\mathbb{P}(Y=0/X) = \frac{\mathbb{P}(X/Y=0)\mathbb{P}(Y=0)}{\mathbb{P}(X/Y=0)\mathbb{P}(Y=0) + \mathbb{P}(X/Y=1)\mathbb{P}(Y=1)} = \frac{\pi}{1+\pi}$$

where $\pi = \frac{\mathbb{P}(X/Y=0)\mathbb{P}(Y=0)}{\mathbb{P}(X/Y=1)\mathbb{P}(Y=1)}$ is the odds on $Y=0$ for any given values of the X^j's. As π is also equal to $\pi = \frac{p}{1-p}$, this means that, in the logit model, the quantity $\Phi(p)$ is the logarithm of the odds on $Y=0$ and each coefficients β_j represents the influence of the variable X^j on the odds on $Y=0$. To estimate the coefficients β^j, one can use, as in the linear regression, a least square method but as Φ is not linear, the $SSE(\beta)$ is no longer a convexe function of β and thus one can not be sure that the problem has a unique solution (and usually it is not the case). That is why one often use a maximal likelihood method choosing β that maximize the logarithm of the likelihood that is given by

$$L(\beta, Y) = \sum_{i=1}^{n} y_i p(x_i) + (1-y_i)(1-p(x_i))$$

that can be computed by a Newton-Rafson algorithm or another optimisation algorithm.

Perceptron : The perceptron is the most widely used vanilla neural net among a large family of neural learning methods. It can be viewed as a two stages regression as shown in the diagram.

One starts with a variable Y, that we want to explain by a family of explanatory variables $(X^1, \ldots, X^p)$. Derived features $(Z^1, \ldots, Z^M)$ are created from linear combinaisons of the inputs $(X^1, \ldots, X^p)$ using a logistic model and then the output Y is modelled as a function of linear combinaison of the Z_m's. The $(Z^1, \ldots, Z^M)$ are called hidden units (and build the hidden layer) because their values are not directly observed. Multi layers perceptron are also used. In analitical terms, a single layer perceptron corresponds to the following model :

$$\left\{ \begin{array}{rcl} Z_m & = & \Phi^{-1}(\beta_{0m} + \beta_{1m} X^1 + \ldots + \beta_{pm} X^p + \varepsilon_m), m = 1, \ldots, M \\ Y & = & \Phi_0^{-1}(\alpha_0 + \alpha_1 Z_1 + \ldots + \alpha_M Z_M + \varepsilon \end{array} \right. \quad (1)$$

Usually Φ is the logit function but it can be any logistic link function and Φ_0 is either linear or logistic or a threshold function like $\Phi_0(z) = 0$ if $z \leq 0$ and $\Phi_0(z) = 1$ if $z > 0$. Notice that the case of a perceptron without hidden layer corresponds simply to a logistic regression.