Statistics for Data Science / Exercise 04

Pierpaolo Brutti

Multiple linear regression / Recap

Assume y_i represents the value of the *response variable* on the i^{th} individual, and that $\boldsymbol{x}_i = [x_{i,1}, x_{i,2}, \ldots, x_{i,d}]^{\mathsf{T}}$ represents the individual's values on d *explanatory variables*, with $i \in \{1, \ldots, n\}$. The multiple linear regression model is given by

$$y_i = \beta_0 + \boldsymbol{x}_i^{\mathsf{T}}\boldsymbol{\beta} + \varepsilon_i = \beta_0 + \beta_1 \cdot x_{i,1} + \cdots + \beta_1 \cdot x_{i,d} + \varepsilon_i.$$

The error terms ε_i are assumed to be independent random variables typically having a Normal distribution with mean zero and constant variance σ_ε^2, but this is not needed for *prediction*. Consequently, the distribution of the random response variable Y has expected value given by the linear combination of the explanatory variables

$$\mathbb{E}(Y \mid \boldsymbol{X} = \boldsymbol{x}) = \beta_0 + \boldsymbol{x}^{\mathsf{T}}\boldsymbol{\beta} = \beta_0 + \beta_1 \cdot x_1 + \cdots + \beta_1 \cdot x_d,$$

and with variance σ_ε^2.

The parameters of the model $\{\beta_1, \ldots, \beta_d\}$ are known as *regression coefficients* with β_0 corresponding to the overall mean.

The regression coefficients can be interpreted as the expected change in the response variable associated with a unit change in the corresponding explanatory variable, when the remaining explanatory variables are held constant.

The *linear* in multiple linear regression applies to the regression parameters, **not** to the response or explanatory variables. Consequently, models in which, for example, the logarithm of a response variable is modelled in terms of quadratic functions of some of the explanatory variables would be included in this class of models: actually, some of the most effective machine learning techniques available are simply linear models in a suitably *engineered* feature space (e.g. support vector machines and kernel methods).

The multiple linear regression model can be conveniently written for all n individuals by using matrices and vectors as

$$\boldsymbol{y} = \mathbb{X} \cdot \boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

where $\boldsymbol{y} = [y_1, \ldots, y_n]^{\mathsf{T}}$ is the vector of response variables, $\boldsymbol{\beta} = [\beta_0, \beta_1, \ldots, \beta_d]^{\mathsf{T}}$ is the vector of regression coefficients, and $\boldsymbol{\epsilon} = [\epsilon_1, \ldots, \epsilon_n]^{\mathsf{T}}$ are the error terms. The *design* or model matrix $\mathbb{X}$ consists of the d measured explanatory variables and a column of ones corresponding to the intercept term

$$\begin{bmatrix} 1 & x_{1,1} & x_{1,2} & \cdots & x_{1,d} \\ 1 & x_{2,1} & x_{2,2} & \cdots & x_{2,d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & x_{n,2} & \cdots & x_{n,d} \end{bmatrix} = \begin{bmatrix} 1 & \boldsymbol{x}_1^{\mathsf{T}} \\ 1 & \boldsymbol{x}_2^{\mathsf{T}} \\ \vdots & \vdots \\ 1 & \boldsymbol{x}_n^{\mathsf{T}} \end{bmatrix}.$$

In case one or more of the explanatory variables are nominal or ordinal variables, they are represented by a 0-1 dummy coding. For exammple, assume that X_1 is a factor with p levels, the submatrix of $\mathbb{X}$ corresponding to X_1 is a $(n \times p)$ matrix of 0 and 1, where the j^{th} element in the i^{th} row is 1 when $x_{i,1}$ is at the j^{th} level.

Assuming that the cross-product matrix $(\mathbb{X}^{\mathsf{T}}\mathbb{X})$ is *non-singular*, i.e., can be inverted, then the least squares estimator of the parameter vector $\boldsymbol{\beta}$ is unique and in matrix form equal to

$$\widehat{\beta} = (\mathbb{X}^{\mathsf{T}}\mathbb{X})^{-1}\mathbb{X}^{\mathsf{T}}\boldsymbol{y}.$$

If the cross-product $(\mathbb{X}^{\mathsf{T}}\mathbb{X})$ is singular we need to reformulate the model to $\boldsymbol{y} = (\mathbb{X}\mathbb{C}) \cdot \boldsymbol{\beta}^\star + \boldsymbol{\varepsilon}$, such that $(\mathbb{X}\mathbb{C})$ has full rank. The matrix $\mathbb{C})$ is called the *contrast matrix*, and the result of the model fit is an estimate $\widehat{\beta}^\star$.

The expectation and covariance of this estimator $\widehat{\beta}$ are given by

$$\begin{aligned}\mathbb{E}(\widehat{\boldsymbol{\beta}}) &= \boldsymbol{\beta} \quad \text{(unbiased)}\\ \mathbb{V}\mathrm{ar}(\widehat{\boldsymbol{\beta}}) &= \sigma_{\varepsilon}^2(\mathbb{X}^{\mathtt{T}}\mathbb{X})^{-1}.\end{aligned}$$

The diagonal elements of the covariance matrix $\mathbb{V}\mathrm{ar}(\widehat{\beta})$ give the variances of $\widehat{\beta}_j$ $j \in \{0, \ldots, d\}$, whereas the off diagonal elements give the covariances between pairs of $\widehat{\beta}_j$ and $\widehat{\beta}_k$.

The square roots of the diagonal elements of the covariance matrix are thus the *standard errors* of the estimates $\widehat{\beta}_j$.

The predicted value of the response variable for the $i^{\mathtt{th}}$ individual is equal to

$$\widehat{y}_i = \widehat{\beta}_0 + \boldsymbol{x}_i^{\mathtt{T}} \cdot \widehat{\beta} = \widehat{\beta}_0 + \big(\widehat{\beta}_1 \cdot x_{i,1} + \cdots + \widehat{\beta}_d \cdot x_{i,d}\big),$$

that is, in matrix form for the whole set of n observations,

$$\widehat{\boldsymbol{y}} = \mathbb{X}\,\widehat{\beta} = \big[\mathbb{X}\,(\mathbb{X}^{\mathtt{T}}\mathbb{X})^{-1}\mathbb{X}^{\mathtt{T}}\big]\boldsymbol{y} = \mathbb{H}\,\boldsymbol{y}.$$

The matrix $\mathbb{H}$ is usually called the *hat matrix* and encodes all the geometry behind the least squares solution[1].

An estimate of the noise variance σ_{ε}^2 is

$$\widehat{\sigma}_{\varepsilon}^2 = \frac{1}{n-d-1}\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2.$$

The correlation between the observed values $\{y_i\}_i$ and the fitted values $\{\widehat{y}_i\}_i$ is known as the *multiple correlation coefficient.*

Regression Diagnostics

The possible influence of outliers and the checking of assumptions made in fitting the multiple regression model, i.e., constant variance and normality of error terms, can both be undertaken using a variety of diagnostic tools, of which the simplest and most well known are the estimated residuals, i.e., the differences between the observed values of the response and the fitted values of the response. In essence these residuals estimate the error terms in the simple and multiple linear regression model. So, after estimation, the next stage in the analysis should be an examination of such residuals from fitting the chosen model to check on the normality and constant variance assumptions and to identify outliers. The most useful plots of these residuals are:

- A plot of residuals against each explanatory variable in the model. The presence of a non-linear relationship, for example, may suggest that a higher- order term, in the explanatory variable should be considered.
- A plot of residuals against fitted values. If the variance of the residuals appears to increase with predicted value, a transformation of the response variable may be in order.
- A normal probability plot of the residuals. After all the systematic variation has been removed from the data, the residuals should look like a sample from a standard normal distribution. A plot of the ordered residuals against the expected order statistics from a normal distribution provides a graphical check of this assumption.

[1] $\mathbb{H}$ is the projection matrix onto the column space of the $\mathbb{X}$ matrix. Hence, the matrix reformulation shows that the prediction are obtained by projecting the observed response vector $\boldsymbol{y}$ onto the column space of $\mathbb{X}$. This also shows that, with no big surprise, the linear model belongs to the large family of linear smoothers including all those regression techniques that build their prediction as linear combination of the observed response vector $\boldsymbol{y}$.

Exercise: Cloud seed

Description

Weather modification, or cloud seeding, is the treatment of individual clouds or storm systems with various inorganic and organic materials in the hope of achieving an increase in rainfall.

Introduction of such material into a cloud that contains supercooled water, that is, liquid water colder than zero degrees of Celsius, has the aim of inducing freezing, with the consequent ice particles growing at the expense of liquid droplets and becoming heavy enough to fall as rain from clouds that otherwise would produce none.

More in details, the data contained in `cseed.RData` comes from an experiment to investigate the use of massive amounts of silver iodide (100 to 1000 grams per cloud) in cloud seeding to increase rainfall. In the experiment, which was conducted in an area of Florida, 24 days were judged suitable for seeding on the basis that a measured suitability criterion, denoted `S-Ne`, was not less than 1.5. Here `S` is the *seedability*, the difference between the maximum height of a cloud if seeded and the same cloud if not seeded predicted by a suitable cloud model, and `Ne` is the number of hours between 1300 and 1600 G.M.T. with 10 centimetre echoes in the target; this quantity biases the decision for experimentation against naturally rainy days. Consequently, optimal days for seeding are those on which seedability is large and the natural rainfall early in the day is small. On suitable days, a decision was taken at random as to whether to seed or not. For each day a specific set of explanatory variables was measured.

The goal in analysing these data is to see how rainfall is related to the explanatory variables and, in particular, to determine the effectiveness of seeding.

Exercise

0. Place the file `cseed.RData` in a folder named `ex04`. In `RStudio`, create a new project inside this folder.
1. Load into `R` the data contained in the `cseed.RData` file.
2. Take a quick look using the appropriate `R` functions: what kind of object are you looking at? How many observations? How many variables? What are the names of the variables?
3. Evaluate basic *univariate* statistics for all the variables involved.
4. Produce suitable graphical summaries for each variable in the dataset. Briefly comment the result.
5. Produce a scatterplot matrix of the numerical variables involved (hint: take a look at `pairs()` and grab some code from the example section to use the `panels` better). Just by looking at this plot, what do you expect in terms of linear correlation between the numerical variables and the response variable `rainfall`? Do you see any evidence of *outliers*, that is, observations that are somewhat "different" from the bulk of the data?
6. Since the goal is to see if the factor `seeding` has some effect on the response `rainfall`, it makes sense to graphically explore better their relation by looking at a boxplot of `rainfall` grouped by the level of `seeding` (hint: see the help file of `boxplot()` and use a suitable `formula` to achieve this). Save the output of `boxplot()` in a variable, take a look inside, and recover the indeces of the outliers. Beside the grouped boxplot, can you come up with any other simple but useful visualization of their relation? Conclude briefly commenting the results.
7. To make a more quantitative (although still pretty rough) evaluation of the impact of `seeding` on `rainfall` we can calculate (basic) grouped statistics using `aggregate()`. Get grouped `mean()`, `median()` and `sd()`. Briefly comment the results.
8. Evaluate and visualize using the package `corrplot` the correlation matrix associated to the numerical variables in the dataset. Again briefly comment the results in light of the fact we are about to fit a linear model to predict `rainfall`.

9. Time to fit a linear model! In this example it is sensible to assume that the effect of the explanatory variables **is** modified by `seeding`. Hence we want to consider a model that includes `seeding` as covariate and, in addition, allows **interaction terms** for `seeding` with each of the covariates except `time`. This model can be succinctly described by the formula

```
clouds_frm <- rainfall ~ seeding + seeding:(sne + cloudcover + prewetness + echomotion) + time.
```

Notice the use of `:` to quickly specify interaction terms. To have a look at the implied design matrix $\mathbb{X}$ we can use

```r
Xstar <- model.matrix(clouds_frm, data = clouds)
Xstar
```

By default, *treatment contrasts* have been applied to the dummy codings of the factors `seeding` and `echomotion` as can be seen from the inspection of the contrasts attribute of the model matrix

```r
attr(Xstar, "contrasts")
```

All this said, and although here we have only few data points, use `caret::createDataPartition()` to create a 80%/20% train-test split, use `lm()` to fit a linear model based on `clouds_frm` to the train-set, explore the fit with `summary()` and `plot()`. Use `coef()` and `vcov()` on the resulting `lm` object to extract the least squares estimates and their estimated variance-covariance matrix respectively. Use the function `diag()` on the latter to evaluate the standard errors of the least squares estimates. Comment the results.

10. Looking at the results, it seems that the model indicates that higher values of the `S-Ne` criterion lead to less `rainfall`, but only on days when cloud seeding happened. A suitable scatterplot `rainfall` vs `sne` grouping by the level of `seeding` will help in the interpretation of this result. Try to build this plot and comment.

11. Predict on train and test and evaluate the corresponding RMSE. Briefly comment the results.

12. Use `caret::train()`, `caret::trainControl()` to fit the same model evaluating its predictive risk via a 10-fold cross-validation method. Comment the results.