1 Multiple Regression

1.1 Introduction

We assume that we have the \( p \)-dimensional input vectors \( x_i = (x_{i1}, x_{i2}, \ldots, x_{ip}) \), and we want to predict the real-valued output \( Y_i \)'s for \( i = 1, \ldots, n \) where \( n \) is the number of datapoints. The linear regression model has the form

\[
Y_i = \beta_0 + \sum_{j=1}^{p} x_{ij} \beta_j + \varepsilon_i,
\]

where \( \varepsilon_i \) is a residual term. Here the \( \beta_j \)'s are unknown coefficients, and the input variables can come from different sources: quantitative inputs; transformations of quantitative inputs, such as log, square-root or square; interactions between variables, for example, \( x_{i3} = x_{i1} x_{i2} \).

1.2 Least Squares

1.2.1 Solution of least squares

We assume that we have a set of training data \((x_1, y_1), \ldots, (x_n, y_n)\) from which to estimate the \( \beta \) parameters. Each \( x_i = (x_{i1}, x_{i2}, \ldots, x_{ip}) \) is a vector of measurements for the \( i^{th} \) individual and \( y_i \) is the \( i^{th} \) output. The most popular estimation method is least squares, in which we choose the coefficients \( \beta = (\beta_0, \ldots, \beta_p) \) that minimize the residual sum of squares

\[
RSS(\beta) = \sum_{i=1}^{n} \left( y_i - \beta_0 - \sum_{j=1}^{p} x_{ij} \beta_j \right)^2.
\]

We can show that the unique solution is

\[
\hat{\beta} = (X^TX)^{-1}X^Ty,
\]
where $X$ is the design matrix

$$X = \begin{pmatrix}
1 & x_{11} & \cdots & x_{1p} \\
\vdots & \vdots & \ddots & \vdots \\
1 & x_{n1} & \cdots & x_{np}
\end{pmatrix}, \quad \text{and } y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}. $$

The regression model can also be expressed in matrix form as

$$y = X\beta + \varepsilon, \quad (4) $$

where $\varepsilon = (\varepsilon_0, \ldots, \varepsilon_n)$.

### 1.2.2 Coefficient of determination

We denote by $y_i^*$ the fitted values from regression analysis

$$y_i^* = \hat{\beta}_0 + \sum_{j=1}^{p} x_{ij} \hat{\beta}_j. $$

The coefficient of determination is defined as

$$R^2 = Cor^2(y_i^*, y_i), $$

where Cor denotes the empirical correlation coefficient. We can show that

$$R^2 = \frac{\text{Var}(y_i) - \text{Var}(\hat{\varepsilon}_i)}{\text{Var}(y_i)}, \quad (5) $$

where the $\hat{\varepsilon}_i = y_i - y_i^*$ are the residuals from the regression. The numerator of equation [5] is equal to the difference between the total variance $\text{Var}(y_i)$ and the residual variance $\text{Var}(\hat{\varepsilon}_i)$ and is called the explained variance. The coefficient of determination $R^2 \in [0, 1]$ is therefore interpreted as the proportion of variance that is explained by the $p$ predictors of the regression.

**Remark: regression as conditional expectation** For this paragraph only, we assume that $x_{ip}$ is not fixed but rather a realization of a random variable $X_{ip}$ for each $i$ and $p$. Assuming that $Y_i$ has finite variance, it can be shown that the best approximation

$$\arg \min_f E[(Y_i - f(X_{i1}, \ldots, X_{ip}))^2]$$

of $Y_i$ by a function $f(X_{i1}, \ldots, X_{ip})$ with finite variance is the function $E[Y_i|X_{i1}, \ldots, X_{ip}]$. Since the functional form of the conditional expectation of $Y_i$ given $X_{i1}, \ldots, X_{ip}$ is unknown, we have to assume a particular
parametric form for $f$. When $f$ belongs to a parametric family $\{f_\beta\}_{\beta \in B}$, the regression model has the general form

$$Y_i = f_\beta(x_{i1}, \ldots, x_{i11}) + \varepsilon_i,$$

where $\varepsilon_i$ is the residual term with $E[\varepsilon_i] = 0$ and $\text{Var}(\varepsilon_i) = \sigma^2$. If $f$ is assumed to depend linearly on its parameters $\beta$, we obtain the linear regression model of equation (1).

1.3 Gaussian model

The Gaussian model further assumes that $\varepsilon \sim N(0, \sigma^2)$.

1.3.1 $F$-test

$F$-test for a group of variables. With the Gaussian hypothesis, we can use the $F$-test to test if a group of variables, let us say the $p'$th to the $p$th variable, significantly improves the fit of the regression. The null hypothesis is $H_0 : (\forall j \in [p', p], \beta_j = 0)$ against $H_1 : (\exists j \in [p', p], \beta_j \neq 0)$. The test statistic is the $F$ statistic,

$$F = \frac{(RSS_{p'} - RSS_p)/(p - p')}{RSS_p/(n - p - 1)} \quad (6)$$

where $RSS_p$ is the residual sum-of-squares of the bigger model with $p + 1$ parameters, and $RSS_{p'}$ is the same for the nested model with $p' + 1$ parameters. The $F$ statistic measures the decrease in residual sum-of-squares per additional parameter (the numerator of equation (6)), and it is normalized by an estimate of $\sigma^2$ (the denominator of equation (6)). Under $H_0$, the $F$ statistic follows a Fisher distribution $F_{p-p', n-p-1}$ with $p - p'$ and $n - p - 1$ degrees of freedom.

The result of a statistical test is nowadays reported using $P$-values. The $P$-value is the probability under the null hypothesis to have a test statistic more atypical—in the direction of the alternative hypothesis—than the observed one. Here, we will reject $H_0$ for large enough values of $F$

$$P\text{-value} = P(F \geq f),$$

where $f$ is the $F$-statistic computed for the data and $F \sim F_{p-p', n-p-1}$. The null hypothesis is rejected when the $P$-value is small enough ($P < 5\%, P < 1\%, P < 0.1\%, \ldots$, depending on the context).
This test can be generalized to the null hypothesis that \( \beta \) belongs to an affine subspace \( E \) of \( \mathbb{R}^{p+1} \) of dimension \( p' \), i.e., satisfies given linear constraints. As an example, the null hypothesis could be \( H_0 : \beta_1 = 2\beta_3 \) against \( H_1 : \beta_1 \neq 2\beta_3 \). The test statistic is still given by equation (6), where \( RSS_p \) now refers to the residual sum-of-squares for the model estimated under the linear constraints. Also, the ratio \( (RSS_{p'} - RSS_p)/(p - p') \) should be replaced by \( (RSS_{p'} - RSS_p)/(p + 1 - p') \). Note that if \( A \) is a \((p + 1) \times p'\) matrix whose columns represent a basis of \( E \), a model satisfying these constraints is obtained using the design matrix \( XA \).

**F-test of the regression.** There is an instance of the F-test that is of particular interest. It is the situation where the bigger model is the complete model that includes all explanatory variables whereas the nested model is the model with parameter \( \beta_0 \) only. This test is called the F-test of the regression and it assesses whether there is an effect of the predictors on the variable to predict \( Y \). Formally, the null hypothesis is \( H_0 : (\forall j \geq 1, \beta_j = 0) \) against \( H_1 : (\exists j \geq 1, \beta_j \neq 0) \). We can show that the \( F \) statistic of the regression is

\[
F = \frac{p}{n - p - 1} \frac{R^2}{1 - R^2},
\]

and it follows a Fisher distribution \( F_{p,n-p-1} \) with \( p \) and \( n - p - 1 \) degrees of freedom under \( H_0 \). The formula of the \( F \) statistic given in equation (7) can be derived using equations (6) and (5) (an exercise for you!).

Here, we will reject \( H_0 \) for large enough values of \( R^2 \), i.e. large enough values of \( F \) so that the \( P \)-value is equal to \( P(F \geq f) \), where \( f \) is the \( F \)-statistic computed for the data and \( F \sim F_{p,n-p-1} \).

1.3.2 **t-tests of the regression**

We test the hypothesis that a particular coefficient is null. The null hypothesis is \( H_0 : \beta_j = 0 \), and the alternative one is \( H_1 : \beta_j \neq 0 \). The \( T \) statistic is

\[
T = \frac{\hat{\beta}_j}{\hat{\sigma} \sqrt{v_j}},
\]

where \( v_j \) is the \( j \)th diagonal element of \((X^TX)^{-1}\) and

\[
\hat{\sigma}^2 = \frac{1}{n - p - 1} \sum_{i=1}^{n} (y_i - y_i^*)^2.
\]

Under \( H_0 \), the \( T \) statistic follows a Student’s \( t \) distribution with \( n - p - 1 \) degrees of freedom.
1.3.3 R Example

In this example, we regress the murder rate, obtained for each of the 50 US states in 1973, by two potential explicative variables: the assault rate and the percentage of urban populations.

Violent Crime Rates by US State

This data set contains statistics, in arrests per 100,000 residents for assault, murder, and rape in each of the 50 US states in 1973. Also given is the percent of the population living in urban areas.

Format:

A data frame with 50 observations on 4 variables.

| [,1] Murder   numeric Murder arrests (per 100,000) |
| [,2] Assault numeric Assault arrests (per 100,000) |
| [,3] UrbanPop numeric Percent urban population |
| [,4] Rape numeric Rape arrests (per 100,000) |

```
> mylm<-lm(Murder~UrbanPop+Assault,data=USArrests)
> summary(mylm)

lm(formula = Murder ~ UrbanPop + Assault, data = USArrests)

Coefficients:

Estimate Std. Error t value  Pr(>|t|)
(Intercept) 3.207153  1.740790  1.842   0.0717 .
UrbanPop   -0.044510  0.026363  -1.688   0.0980 .
Assault     0.043910  0.004579   9.590  1.22e-12 ***

---

Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.58 on 47 degrees of freedom
Multiple R-squared: 0.6634, Adjusted R-squared: 0.6491
F-statistic: 46.32 on 2 and 47 DF,  p-value: 7.704e-12
```
There is a signficative effect of the explicative variables on the murder rate because the $P$-value of the $F$-test is $7.704 \times 10^{-12}$. Once the percentage of urban population has been accounted for, the assault rate has a signficative effect on the murder rate because the $P$-value of the $t$-test is $1.22 \times 10^{-12}$. However, once the murder rate is accounted for, there is no additional effect of the percentage of urban population on the murder rate ($P = 0.0980$) unless we accept a large enough type I error (e.g. $\alpha = 10\%$).

1.4 Variable selection

Variable selection or feature selection consists of selecting a subset of relevant variables (or features) among the $p$ initial variables. There are two main reasons to perform variable selection.

- The first is prediction accuracy: the least squares estimates often have low bias but large variance especially when the number of variables $p$ is large, of the same order as the sample size $n$. Prediction accuracy can be improved by variable selection.

- The second reason is interpretation. With a large number of predictors, we want to sacrifice some of the small details in order to get the ‘big picture’ carried by a smaller subset of variables that exhibit the strongest effects.

1.4.1 The bias-variance decomposition

Assume that we want to make a prediction for an input $x_0 = (x_{01}, \ldots, x_{0p})$. Denoting by $\hat{f}$ the regression function $\hat{f}(x_0) = \hat{\beta}_0 + \sum_{j=1}^{p} x_{0j} \hat{\beta}_j$, the prediction error is given by

$$E_{pred}(x_0) = E[(Y - \hat{f}(x_0))^2]$$

(9)

$$= E[(f(x_0) + \epsilon - \hat{f}(x_0))^2]$$

(10)

$$= \sigma^2 + \text{bias}^2(\hat{f}(x_0)) + \text{Var}(\hat{f}(x_0))$$

(11)

$$= \text{irreducible error} + \frac{\text{bias}^2 + \text{variance}}{\text{mean square error of } \hat{f}(x_0)}$$

When $p$ is large (e.g. large number of predictors, large-degree polynomial regression...), the model can accommodate complex regression functions (small bias) but because there are many $\beta$ parameters to estimate, the variance of the estimates may be large. For small values of $p$, the variance of the estimates gets smaller but the bias increases. There is therefore an optimal value to find for the number of predictors that reaches
the so-called bias-variance tradeoff.

To give a concrete example, assume that we make a polynomial regression \( f(x) = \beta_0 + \beta_1 x + \cdots + \beta_p x^p \) with \( p = n - 1 \). Since \( p = n - 1 \), the regression function will interpolate all of the \( n \) points. This is not a desirable feature and it suffers from overfitting. Rather than fitting all the points in the training sample, the regression function should rather provide a good generalization, i.e. be able to make good predictions for new points that have not been used during the fit. This is the reason why choosing the model that minimizes the residual squared error computed from the whole dataset is not a reasonable strategy. In subsections 1.4.2 and 1.4.3 we present different methods to compute the prediction error so that we can choose a model that achieves optimal generalization.

1.4.2 Split-validation

If we are in a data-rich situation, a standard approach to estimate the prediction error is to randomly divide the dataset into two parts: a training set with \( n_1 \) datapoints, and a validation set with \( n_2 \) datapoints \((n_1 + n_2 = n)\). The training set is used to fit the models, i.e. to provide estimates of the regression coefficients \((\hat{\beta}_0,\ldots,\hat{\beta}_p)\) and the validation set is used to estimate the prediction error. The prediction error is estimated by averaging the squared errors over the points of the validation set

\[
E_{\text{pred}} = \frac{1}{n_2} \sum_{(x_i,y_i) \in \text{Validation Set}} \left( y_i - (\hat{\beta}_0 + \sum_{j=1}^{p} x_{ij}\hat{\beta}_j) \right)^2 .
\]  

We typically choose the model that generates the smallest prediction error.

1.4.3 Cross-validation

The caveat of the split-validation approach is that the prediction error may severely depend on a specific choice of the training and validation sets. To overcome this problem, the following cross-validation approaches consider a rotation of the validation and training sets.

- **K-fold cross-validation.** The idea is to split the data into \( K \) parts. To start, \( K - 1 \) subsets are used as the training sets and the remaining subset is used as a validation subset to compute a prediction error \( E_{\text{pred}}^1 \) as given by equation (12). This procedure is repeated \( K \) times by considering each subset as the validation subset and the remaining subsets as the training sets. The average prediction error is
defined as

\[ E_{\text{pred}} = \frac{1}{K} \sum_{k=1}^{K} E_{k,\text{pred}} \]

- **Leave-one-out.** It is the same idea as \( K \)-fold cross validation with \( K = n \). The prediction error is computed as

\[ E_{\text{pred}} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{f}_{-i}(x_i))^2, \]

where \( \hat{f}_{-i} \) denotes the estimated regression function when the \( i \)th point \( x_i \) has been removed from the training dataset.

To show the potential of cross-validation, we consider the following polynomial example

\[ Y = X - X^2 + X^3 + \varepsilon, \]

\[ \varepsilon \sim N(0, \sigma^2 = 10^2). \]

We simulate a dataset with \( n = 100 \) using the polynomial model of equation (14) (see the left panel of Figure 1). We then consider a polynomial regression to smooth the data

\[ Y = \beta_0 + \beta_1 x + \cdots + \beta_p x^p + \varepsilon \]

We consider a range of polynomial degrees \( p = 1, \ldots, 20 \) for polynomial regression. In the right panel of Figure 1, we show the leave-one-out prediction error (equation (13)) and the residual squared error for the entire data set \( \text{RSS}(\hat{\beta}) \) in equation (2) as a function of the number of degrees. The residual squared error is a decreasing function of the number of degrees so that minimizing naively the residual squared error would lead to the selection of the most complex model. The prediction error has a nonmonotonous behavior and points to a polynomial regression of degree 3, which is consistent with the generating mechanism given in equation (14).

### 1.4.4 Akaike information criterion (AIC)

The computation of a prediction error based on cross-validation is time consuming because the regression function must be fitted several times. Alternative criteria, such as the Akaike Information Criterion (AIC), do not require several trainings of the regression function. Model selection by AIC picks the model that minimizes

\[ \text{AIC} = n \log \frac{\text{RSS}(\hat{\beta})}{n} + 2p, \]
Figure 1 – Polynomial regression for artificial data points. The data points are generated with the model given in equation (14). The prediction error is estimated with a leave-one-out cross-validation method.

where $RSS(\cdot)$ is the residual sum of squares given in equations (2). The AIC is obtained as an approximation of an information-theoretic criterion. It achieves a tradeoff between model fit (1st term in equation (15)) and model complexity (2nd term in equation (15)).

1.4.5 Optimization algorithms

If there are $p$ predictors, the number of potential subsets (i.e. regression models) to explore is $2^p$ and estimating a prediction error for all models might be impossible. Instead, greedy algorithms are usually considered. Forward algorithms start with one predictor, keep the best predictor according to one of the criteria proposed above, and add a second predictor if it decreases the prediction error. The procedure is continued until 1) it is not possible to decrease the prediction error or until 2) all predictors have been included. Backward algorithms work similarly but start with the complete regression model that includes all predictors ($X_1, \ldots, X_p$) and remove the predictors one after another. Using the USArrests example, we show how to perform in R the backward algorithm with AIC.

```r
> mylm<-lm(Murder~.,data=USArrests)

####By default, the step function proceeds backward
```
> stepAIC(mylm)

Start:  AIC=98.39

Murder ~ Assault + UrbanPop + Rape

Df Sum of Sq RSS AIC
- Rape 1 8.041 312.87 97.689
<none> 304.83 98.387
- UrbanPop 1 25.503 330.33 100.404
- Assault 1 300.020 604.85 130.648

Step:  AIC=97.69

Murder ~ Assault + UrbanPop

Df Sum of Sq RSS AIC
<none> 312.87 97.689
- UrbanPop 1 18.983 331.85 98.633
- Assault 1 612.18 925.05 149.891

Call:

lm(formula = Murder ~ Assault + UrbanPop, data = USArrests)

Coefficients:

(Intercept) Assault UrbanPop
    3.20715   0.04391  -0.04451
2 Analysis of variance (ANOVA)


Analysis of variance (ANOVA) is a collection of statistical models, and their associated procedures, in which the observed variance in a particular variable is partitioned into components attributable to different sources of variation. In its simplest form ANOVA provides a statistical test of whether or not the means of several groups are all equal, and therefore generalizes $t$-test to more than two groups.

2.1 Introduction to one-way ANOVA

In one-way ANOVA, we test if a qualitative variable called a factor has a significant effect on a quantitative variable. Generalizing this approach to two qualitative explicative variables is possible with two-way ANOVA as shown in subsection 2.5.

To give a concrete example, one-way ANOVA will provide a statistical framework to test if the Ensimag specialization (mathematical finance, telecommunications,...) influences the marks of the students in the course of introductory statistics (PMS, Principes et Méthodes Statistiques). The factor corresponding to the students' specialization can take $K = 7$ different levels (in 2009/2010, the levels were MMIS, TEL, ISI, MIF, SIF, SLE, m1rmosig). In ANOVA, we assume that the data—the marks in the example—are Gaussian

\[ X_k \sim \mathcal{N}(m_k, \sigma^2), \]

where $m_k$, $k = 1, \ldots, K$, denotes the mean of the data for the $k^{th}$ level. We test the null hypothesis $H_0 : m_1 = \ldots = m_K$ against the alternative one $H_1 : \exists k, l \ m_k \neq m_l$.

2.2 The $F$-test for one-way ANOVA

We denote by $S^2_W$ the residual variance or the variance within groups. It is defined by

\[ S^2_W = \frac{1}{n} \sum_{k=1}^{K} \sum_{i=1}^{n_k} (X^i_k - \bar{X}_k)^2, \]
where $\bar{X}_k$ and $n_k$ denotes the empirical mean and the sample size for the data in the $k^{th}$ level. We denote by $S_B^2$ the variance that is attributable to the factor or the variance between groups. It is defined by

$$S_B^2 = \frac{1}{n} \sum_{k=1}^{K} n_k (\bar{X} - \bar{X}_k)^2.$$ 

We finally denote by $S^2$ the total variance

$$S^2 = \frac{1}{n} \sum_{k=1}^{K} \sum_{i=1}^{n_k} (X_{ik} - \bar{X})^2.$$ 

The total variance can be partitioned as

$$S^2 = S_B^2 + S_W^2. \quad (16)$$

Using equation (16), it can be shown that under the null hypothesis, the $F$-statistic

$$\frac{S_B^2/(K-1)}{S_W^2/(n-K)} \sim F_{K-1, n-K},$$

where $F_{df_1, df_2}$ is the Fisher distribution with $df_1$ and $df_2$ degrees of freedom. The $P$-value of the test equals

$$P\text{-value} = \Pr(F \geq f), \text{ where } F \sim F_{K-1, n-K},$$

and $f$ is the $F$-statistic computed for the data.

### 2.3 ANOVA in R

In R, the one way ANOVA to test if the students’ specialization influences their mark in introductory statistics (PMS) is performed as follows

```r
#Read the data
> alldata<-read.table("pms_anova.data",header=T)

#What is in the data
> names(alldata)
[1] "notes" "prof" "specialization"

#E.g. first mark in the list
alldata[1,]
  notes  prof specialization
```

\[ \text{12} \]
#convert the column corresponding to the specialization into a factor
> alldata[,3]<-factor(alldata[,3])

#Perform anova
> anova(lm(marks~specialization,data=alldata))

Analysis of Variance Table

Response: marks

<table>
<thead>
<tr>
<th>Df</th>
<th>Sum Sq</th>
<th>Mean Sq</th>
<th>F value</th>
<th>Pr(&gt;F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>specialization</td>
<td>6</td>
<td>245.84</td>
<td>40.973</td>
<td>3.601</td>
</tr>
<tr>
<td>Residuals</td>
<td>190</td>
<td>2161.83</td>
<td>11.378</td>
<td>---</td>
</tr>
</tbody>
</table>

---

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Since $P = 0.2\%$, there is a significant effect of the student’s specialization on the marks in introductory statistics.

### 2.4 Estimation of the effects

The model of one-way ANOVA can be written with a linear model as follows

\[ X_k^i = \mu + \alpha_k + \varepsilon_k^i \]  \hspace{1cm} \text{(17)}

\[ \varepsilon_k^i \sim \mathcal{N}(0, \sigma^2). \]

The model is not identifiable because we clearly have $\mu + \alpha_k = \bar{x}_k$, $k = 1, \ldots, K$, so that there is an infinite number of solutions for the $(K+1)$ parameters $(\mu, \alpha_1, \ldots, \alpha_K)$. To solve this problem, we usually add the constraint (also called contrast) that the mean of the effects is null $\sum_{k=1}^{K} n_k \alpha_k = 0$, so that the least squares solution is

\[ \hat{\mu} = \bar{x}, \]

\[ \hat{\alpha}_k = \bar{x}_k - \bar{x}. \]

With this constraint, the $\alpha_k$’s measure the difference between the mean in the $k^{th}$ factor and the average mean.
Using matrix notations, the model of one-way ANOVA can be described with a regression model of the same form as equation \[4\]

\[
\begin{pmatrix}
y_1 \\
\vdots \\
y_n
\end{pmatrix} =
\begin{pmatrix}
1 & 1 & 0 & \cdots & 0 \\
1 & 1 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & 0 & 0 & \cdots & 1
\end{pmatrix}
\begin{pmatrix}
\mu \\
\alpha_1 \\
\vdots \\
\alpha_K
\end{pmatrix}
+ \begin{pmatrix}
\varepsilon_1 \\
\vdots \\
\varepsilon_n
\end{pmatrix}
\]

The matrix that contains the 0’s and 1’s is the design matrix \(X\) of the regression. In the \(i\)th row and \((j+1)\)th column of the matrix, there is a 1 if the factor of the \(i\)th individual corresponds to the \(j\)th level and a 0 otherwise. The 2nd to the \((p+1)\)th columns of \(X\) correspond to the so-called dummy variables. To account for the constraint \(\sum_{k=1}^{K} n_k \alpha_k = 0\), we can suppress one of the column of \(X\) and replace all the remaining columns by their difference with the removed column. The results established for multiple regression now hold with the design matrix \(X\), which contains the 0’s and 1’s. For instance, the least-squares solution of ANOVA is given by equation \[3\] and testing if the \(\alpha_k\)’s are significantly different from 0 can be done with the \(t\)-test.

Analyzing in R the effects of the different specializations produces the following output

```r
# We set the constraints
> options(contrasts=c("contr.sum","contr.sum"))

# Estimation of the effects
> summary(lm(marks ~ specialization))

Call:
  lm(formula = marks ~ specialization)

Residuals:
    Min      1Q  Median      3Q     Max
-9.790e+00 -2.231e+00 -1.518e-14 2.250e+00 8.328e+00

Coefficients:
                        Estimate Std. Error t value  Pr(>|t|)
(Intercept)            10.5000    0.5474   19.181  < 2e-16 ***
specialization1        0.2900    0.6799    0.427      0.670 MMIS
specialization2       -1.5000    2.9029   -0.517      0.606 TEL
specialization3        2.3489    0.6991    3.360   0.000942 ***

```

14
The two values of the linear model that are significantly different from 0 ($P < 5\%$) are

- The intercept: the mean $\mu$ of the marks is unsurprisingly significantly different from 0 ($P < 2\times10^{-16}$).
- The parameter $\alpha_3$: the mean of the marks for the students with the MIF specialization (mathematical finance) is significantly larger than the average marks ($P = 0.1\%$).

Note that the variance $\sigma^2$ is assumed to be constant across groups in ANOVA (equation (17)). Although this assumption should be in principle tested before performing ANOVA (using for instance the Bartlett test), it is actually rarely tested.

### 2.5 Two-way ANOVA

In two-way ANOVA, we test simultaneously the effect of two qualitative variables on a quantitative variable. We will not give the mathematical details of two-way ANOVA. The model is simply defined as

$$X_{i,k,l} \sim \mathcal{N}(m_{k,l}; \sigma^2), \quad k = 1, \ldots, K, \ l = 1, \ldots, L, \ i = 1, \ldots, N_{k,l},$$

where $m_{k,l} = \alpha_k + \beta_l + \gamma_{k,l}$, $K$ and $L$ are the number of levels of the two factors, and $N_{k,l}$ is the number of measurements for the couple of factors ($k,l$). We show that it can be easily performed using the `anova` function in R. We consider the same example of the marks in introductory statistics. In addition to the specialization of the students, we now consider the professor that corrected the copies as a second factor. To test simultaneously the effect of the two factors we use the following command line.

```r
> anova(lm(marks~specialization+prof,data=alldata))
```

Analysis of Variance Table
Response: marks

<table>
<thead>
<tr>
<th></th>
<th>Df</th>
<th>Sum Sq</th>
<th>Mean Sq</th>
<th>F value</th>
<th>Pr(&gt;F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>specialization</td>
<td>6</td>
<td>245.84</td>
<td>40.973</td>
<td>3.6913</td>
<td>0.001730 **</td>
</tr>
<tr>
<td>prof</td>
<td>5</td>
<td>108.36</td>
<td>21.672</td>
<td>1.9525</td>
<td>0.087717 .</td>
</tr>
<tr>
<td>Residuals</td>
<td>185</td>
<td>2053.47</td>
<td>11.100</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

The p-values result from a F-test for testing linear hypotheses (see Section 1.3.1), e.g. $H_0$ : for all $k = 1, \ldots, K$ and for all $l = 1, \ldots, L$, $\alpha_k = \gamma_{k,l} = 0$. This two-way ANOVA analysis confirms that there is a significative effect of the specialization on the marks ($P = 0.2\%$). By contrast, we do not find a clear evidence that the identity of the corrector impacts the marks ($P = 9\%$).
3 Principal Component Analysis (PCA)

The data consist of \(n\) individuals and for each of these individuals, we have \(p\) measurements (or variables).

The data are given in the following design matrix

\[
X = \begin{pmatrix}
x_{11} & \cdots & x_{1p} \\
\vdots & \ddots & \vdots \\
x_{n1} & \cdots & x_{np}
\end{pmatrix}.
\]

Quoting once again Wikipedia (http://en.wikipedia.org/wiki/Principal_component_analysis)

Principal component analysis (PCA) is a mathematical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated \(p\) variables into a set of values of \(d\) uncorrelated variables called principal components (\(d \leq p\)). This transformation is defined in such a way that the first principal component has as high a variance as possible (that is, accounts for as much of the variability in the data as possible), and each succeeding component in turn has the highest variance possible under the constraint that it be orthogonal to (uncorrelated with) the preceding components.

3.1 Preliminary notations

3.1.1 Covariance and correlation matrices

We denote by \(c_{kl}\) the empirical covariance between the \(k^{\text{th}}\) and \(l^{\text{th}}\) variable

\[
c_{kl} = 1_{n-1} \sum_{i=1}^{n} (x_{ik} - \bar{x}_k)(x_{il} - \bar{x}_l).
\]

The covariance matrix of the design matrix \(X\) is

\[
\Sigma = \begin{pmatrix}
c_{11} & \cdots & c_{1p} \\
\vdots & \ddots & \vdots \\
c_{p1} & \cdots & c_{pp}
\end{pmatrix}.
\]

Note that the diagonal of the covariance matrix contains the vector of the variances. Next, we denote by \(X'\) the matrix obtained from the design matrix \(X\) after we have centered all of the columns (i.e. all of the columns of \(X'\) have a mean of 0), then we have

\[
\Sigma = \frac{1}{n-1} X'X'.
\]
Next, we define the empirical correlation between the \( k \)th and \( l \)th variable
\[
r_{kl} = \frac{c_{kl}}{s_k s_l}, \text{ where } s_k = \sqrt{c_{kk}}.
\]
Accordingly the correlation matrix can be written as
\[
R = D^{-1} \Sigma D^{-1},
\]
where
\[
D = \begin{pmatrix}
s_1 & 0 & \cdots & 0 \\
0 & s_2 & \ddots & \vdots \\
\vdots & \ddots & \ddots & 0 \\
0 & \cdots & 0 & s_p
\end{pmatrix}.
\]

### 3.1.2 Eigenvectors and eigenvalues

We rank the \( p \) eigenvectors of the covariance matrix \( \Sigma \) by decreasing order of their corresponding eigenvalues. The first eigenvector, the column vector \( a_1 \), has the largest eigenvalue \( \lambda_1 \), the second eigenvector, the column vector \( a_2 \), has the second largest eigenvalue \( \lambda_2 \), etc.

The \((p \times p)\) matrix \( A \) of the eigenvectors is
\[
A = [a_1 \ a_2 \ \ldots \ a_p],
\]
and the matrix of the eigenvalues is the diagonal matrix
\[
\Sigma_Y = \begin{pmatrix}
\lambda_1 & 0 & \cdots & 0 \\
0 & \lambda_2 & \ddots & \vdots \\
\vdots & \ddots & \ddots & 0 \\
0 & \cdots & 0 & \lambda_p
\end{pmatrix}.
\]
By definition of the matrix of the eigenvectors, we have
\[
\Sigma = A \Sigma_Y A^T.
\]

### 3.2 Solution of PCA

We can show that PCA can be done by eigenvalue decomposition of the data covariance matrix \( \Sigma \). The first principal component corresponds to the eigenvector \( a_1 \). It means that \( a_1 \) defines the one-dimensional
projection that maximizes the variance of the projected values. The second eigenvector $a_2$ defines the one-dimensional projection that maximizes the variance of the projected values among the vectors orthogonal to $a_1$, and so on.

The $(n \times p)$ matrix $Y$ of the projected values on the principal component axes is

$$Y = XA.$$ 

In the first column of $Y$, we read the projections of the data on the first principal component, in the second column of $Y$, we read the projections of the data on the second principal component, and so on.

### 3.3 Variance captured by the principal components

We can easily show that 1) the variance-covariance matrix of the projected values $Y$ is given by $\Sigma_Y$, and that 2) the sum of the variances of the original data $\sum_{j=1}^{p} \sigma_j^2$ is equal to the sum of the eigenvalues $\sum_{j=1}^{p} \lambda_j$.

This means that the $k^{th}$ principal component captures a fraction $\lambda_k/\sum_{j=1}^{p} \lambda_j$ of the total variance, and that the $k$ first principal components capture a fraction $\sum_{j=1}^{k} \lambda_j/\sum_{j=1}^{p} \lambda_j$ of the total variance.

### 3.4 Scale matters in PCA

Assume that we record the following information for $n$ sampled individuals: their weight in grams, their height in centimeters, their percentage of body fat and their age (in years). The corresponding design matrix $X$ has $n$ lines and $p = 4$ columns. If we perform PCA naively using the variance-covariance matrix $\Sigma$, it is quite clear that the first principal component will be almost equal to the first column (weight in grams). Indeed the choice of the unity of measure, gram here, is such that the variance of the first column will be extremely large compared to the variances of the remaining columns. The fact that the result of PCA depends on the choice of scale, or unity of measure, might be unsatisfactory in several settings. To overcome this problem, PCA may be performed using the correlation matrix $R$ rather than using the variance-covariance matrix $\Sigma$. It is the same as performing PCA after having standardized all of the columns of $X$ (i.e. the columns of the standardized matrix have an empirical variance equal to 1).
3.5 PCA in R

We consider the same USArrests dataset than in the section about multiple regression. This dataset contains four variables measuring three criminality measures and the percentage of urban population in each of the 50 US states in 1973. Using PCA, we can display the dataset in two dimensions (Figure 2).

```r
## The variances of the variables in the
## USArrests data vary by orders of magnitude, so scaling is appropriate
res<-prcomp(USArrests) # inappropriate
res<-prcomp(USArrests, scale = TRUE) # appropriate
## The matrix A of the eigenvectors
> res$rotation
       PC1     PC2     PC3     PC4
Murder -0.5358995 0.4181809 -0.3412327 0.64922780
Assault -0.5831836 0.1879856 -0.2681484 -0.74340748
UrbanPop -0.2781909 -0.8728062 -0.3780158 0.13387773
Rape -0.5434321 -0.1673186 0.8177779 0.08902432
> summary(res)

Importance of components:

PC1   PC2   PC3   PC4
Standard deviation 1.57 0.995 0.5971 0.4164
Proportion of Variance 0.62 0.247 0.0891 0.0434
Cumulative Proportion 0.62 0.868 0.9566 1.0000

##The following function displays the observations on the PC1, PC2 axes
## but also the original variables on the same axis (see Figure 2)
>biplot(res,xlim=c(-.3,.3),ylim=c(-.3,.3),cex=.7)
```

We see that the proportion of the total variance captured by the first 2 principal components is 0.86%. In Figure 2, it is tempting to interpret the first axis as a ‘violence’ axis and the second one as the ‘Urban versus Rural’ axis although this kind of interpretation should be done with caution.
Figure 2 – The 50 US states projected into the first two principal components. Here we use the biplot function in order to display additionally the original 4 variables to the PC1-PC2 graph.
4 Classification

4.1 Principles

4.1.1 The Classification problem

We introduce the classification problem with the classical example of handwritten digit recognition. The data from this example come from the handwritten ZIP codes on envelopes from U.S. postal mail. Each image is a segment from a five digit ZIP code, isolating a single digit. The images are $16 \times 16$ eight-bit grayscale maps, with each pixel ranging in intensity from 0 to 255. Some sample images are shown in Figure 3. The task is to predict, from the $16 \times 16$ matrix of pixel intensities, the identity of each image in $(0, 1, \ldots, 9)$ (for sake of simplicity we consider only the 3 digits $(1, 2, 3)$ in this lecture). If it is accurate enough, the resulting algorithm would be used as part of an automatic sorting procedure for envelopes.

Mathematically, the classifier is a function $\hat{G}$ that takes as input an image $X \in \mathbb{R}^p$ ($p = 16 \times 16$) and returns as an output $Y$ a digit in $G = \{0, 1, \ldots, 9\}$. In this example of image classification, each pixel corresponds to a variable so that an image is coded as a vector with $16 \times 16$ variables. To build the classifier, we have a training set $((x_1, y_1), \ldots, (x_n, y_n))$, where the data $x_i \in \mathbb{R}^p$ and the label $y_i \in G$. We refer to binary classification when $G = \{0, 1\}$ and more generally we have $G = \{0, K\}$.

We do not address the related problem of clustering. Clustering occurs when the data—the $x_i$’s—are not labelled in the training set. Classification is also referred as supervised learning and clustering as unsupervised learning.

4.1.2 Optimal/Bayes classifier

This subsection is quite technical and can be skipped. We assume to have a loss function $L$ that measures the cost of replacing the true and unknown label $Y$ with the label given by the classifier $\hat{G}(x)$. We want to minimize the expected loss function

$$C(x) = E[L(Y, \hat{G}(x))],$$

where $E$ refers to the expectation with respect to the conditional distribution of $Y$ given $x$. Using the law of total probability, we have

$$C(x) = \sum_{k=1}^{K} L(k, \hat{G}(x)) \Pr(Y = k | X = x).$$
Figure 3 – Examples of handwritten digits from U.S. postal envelopes. We consider only the 1, 2 and 3.

If we take the $0 - 1$ loss function that is equal to 0 when the classifier is right and 1 otherwise, we have

$$C(x) = 1 - \Pr(Y = \hat{G}(x)|X = x).$$

Minimizing the expected loss $C(x)$ amounts at choosing the class

$$\hat{G}(x) = \arg \max_{k \in \mathcal{G}} \Pr(Y = k|X = x)$$

This solution is known as the Bayes classifier, and says that we classify to the most probable class, using the conditional (discrete) distribution $\Pr(Y|X)$. Using the Bayes theorem

$$\Pr(Y = k|X = x) = \frac{\Pr(X = x|Y = k)\Pr(Y = k)}{\Pr(X = x)},$$

we find that the Bayes classifier maximizes the product of the conditional probability of the data given the class and the prior for the class

$$\hat{G}(x) = \arg \max_{k \in \mathcal{G}} \Pr(X = x|Y = k)\Pr(Y = k).$$

(18)
If we assume an uniform prior, the Bayes classifier proceeds as maximum likelihood that chooses the parameter—here the class—that makes the data $x$ the most likely.

### 4.1.3 Evaluating the classification error

Evaluating the classification error is an important task to compare different classifiers and to asses the overall performance of the classifier. We can use the same methods as for regression: split-validation, K-fold cross-validation, leave-one-out cross-validation. For split-validation, we split the data into two parts, use one part of the data to build the classifier and use the other part to estimate the error. A standard criterion for evaluating the classification error is the misclassification rate.

### 4.2 Classic classifiers

#### 4.2.1 $k$-nearest neighbor classifier

The $k$-nearest neighbor classifier is based on a simple counting procedure. A new individual $x$ is classified by a majority vote of its neighbors, with the individual being assigned to the most common class amongst its $k$ nearest neighbors ($k \geq 1$). A distance has to be defined in order to find the $k$-nearest neighbors. For image classification, a simple Euclidean distance between two images can be used although more elaborate versions of distances that are robust to deformations may be appropriate. If $k = 1$, then the individual is simply assigned to the class of its nearest neighbor. Using the methodology of subsection 4.1.3, we can choose the value of $k$ that minimizes a misclassification rate estimated with a validation technique.

#### 4.2.2 Linear discriminant analysis (LDA)

In LDA, we assume that the probability distribution of $X$ given the class $k$ is a Gaussian distribution $N(\mu_k, \Sigma)$. We denote by $f_k$ the p.d.f. of the multivariate Gaussian $N(\mu_k, \Sigma)$

$$f_k(x) = \frac{1}{(2\pi)^{p/2}\lvert \Sigma \rvert^{1/2}} e^{-\frac{1}{2}(x-\mu_k)\Sigma^{-1}(x-\mu_k)},$$

where $\lvert \Sigma \rvert$ denotes the determinant of $\Sigma$. What is particularly important in LDA is that we assume that the variance-covariance matrices are the same for all classes. Using the Bayes classifier given by equation (18), we assign an individual to the class $k$ so that

$$k = \arg \max_{k \in \mathcal{G}} f_k(x) \pi_k,$$

24
where $\pi_k$ is the prior for the class $k$. With standard algebra, we can show that it is equivalent to assigning $x$ to the class $k$ that maximizes the linear discriminant functions

$$\delta_k(x) = T_x \Sigma^{-1} \mu_k - \frac{1}{2} T \mu_k \Sigma^{-1} \mu_k + \log \pi_k.$$ 

To compute the linear discriminant functions, the $\mu_k$'s are estimated using the empirical means in each class and $\Sigma$ is estimated as the (weighted) average of the covariance matrices estimated in each class.

In the following, we show how to perform LDA in R using the example of handwritten digit recognition.

# Classification for the ZIP code data

```r
> require(MASS)
> require(ElemStatLearn)

##### Read the data
> zippict<-zip.train

##### We keep only the 1, 2 and 3
> bool<-(zippict[,1]==1 | zippict[,1]==2) | zippict[,1]==3
> zipaux<-zippict[bool,]

##### Dimension of the data
> n<-dim(zipaux)[1]
> p<-dim(zipaux)[2]

##### Display 9 images
> par(mfrow=c(3,3),mar=c(2, 2, 4, 2)+0.1)
> for (i in 10+(1:9)) {mm<-(matrix(as.numeric(zipaux[i,2:257]),nrow=16,byrow=F))
  image(z=mm[,16:1],col=thegrey,xaxt="n",yaxt="n")
}

##### Put the data in a data frame

##### First column is the digit
> data.zip<-data.frame(zipaux)
> data.zip[,1]<-as.factor(data.zip[,1])
> names(data.zip)[1]<="number"

##### We remove the pixels with a constant variance in each group
```
Otherwise LDA returns an error message because the covariance matrix is not invertible

```r
#tor<NULL

# for (var in 1:3)
{thevar<-(zipaux[,1]==var)

for (i in 2:p){if(length(unique(zipaux[thevar,i]))==1 )
    tor<-(tor,i)}

>data.zip<-data.zip[,tor]

###Training set

>train <- sample(1:n,floor(n/2))
>table(data.zip$number[train])

###Linear discriminant analysis

>z <- lda(number ~ ., data.zip, prior = rep(1,3)/3, subset = train)

###Prediction

>myp<-predict(z, data.zip[-train, ])$class

###Confusion matrix

>myconf<-table(myp, data.zip[-train,1])

>myconf

myp 1 2 3
   1 487 1 0
   2 0 363 13
   3 0 8 325

###Display the confusion matrix (see Figure 4)

>barplot(table(myp, data.pca[-train,1]),col=c('black','grey40','lightgrey'))
>legend("topright",paste(1:3),col=c('black','grey40','lightgrey'),lwd=5,cex=2)

###Misclassification rate

>1-sum(diag(myconf))/sum(myconf)

[1] 0.01670844
```
We consider a split-sample validation approach and find a misclassification rate of 1.6%. In addition to the misclassification rate, we compute in the variable `myconf` a confusion matrix, where each row of the matrix represents the instances in a predicted class, and each column represents the instances in an actual class. One benefit of a confusion matrix is that it is easy to see if the classifier is confusing two classes, e.g. the digits 2 and 3 here. Figure 4 displays the confusion matrix using the `barplot` function.

Figure 4 – Confusion matrix for the LDA classifier. In each bar there are the instances in an actual class.

4.2.3 Quadratic discriminant analysis (LDA)

In QDA, we no longer assume that the variance-covariance matrices are the same for all classes. The densities $f_k$ are now the p.d.f. of multivariate Gaussian $\mathcal{N}(\mu_k, \Sigma_k)$, $k = 1, \ldots, K$. Using the Bayes classifier, we can show that we assign an individual $x$ to the class $k$ that maximizes the quadratic discriminant functions

$$
\delta_k(x) = -\frac{1}{2} \log \Sigma_k - \frac{1}{2} (x - \mu_k)\Sigma_k^{-1}(x - \mu_k) + \log \pi_k.
$$
4.3 Logistic regression

Logistic regression is a useful way of describing the relationship between quantitative variables (e.g., age, weight, etc.) and a binary response variable that has only two possible values. Logistic regression models are used mostly as a data analysis and inference tool, where the goal is to understand the role of the input variables in explaining the outcome. It is widely used in biostatistical applications where binary responses (two classes) occur quite frequently. For example, patients survive or die, have heart disease or not, pass a test or not... Compared to discriminant analysis, logistic regression does not assign deterministically an individual $x$ to the classes 0 or 1 but rather provides a probabilistic classifier because it computes $P(Y = 0|X = x)$ and $P(Y = 1|X = x)$.

Unsurprisingly, logistic regression relies on the logistic function (see Figure 5)

$$f(z) = \frac{1}{1 + e^{-z}}, \ z \in \mathbb{R},$$

![Logistic function](image)

Figure 5 – Graph of the logistic function.

...which maps an input value in $\mathbb{R}$ to an output confined between 0 and 1. The inverse of the logistic function is the logit function

$$g(p) = \log \frac{p}{1 - p}, \ p \in [0, 1],$$

...and can therefore map a probability to a value between $-\infty$ and $+\infty$. In logistic regression, we use the logit
function to link the probability \( P(Y = 1|X = x) \) to a linear combination of the explanatory variables

\[
\log \frac{P(Y = 1|X = x)}{1 - P(Y = 1|X = x)} = \beta_0 + \sum_{j=1}^{p} x_j \beta_j,
\]

(19)

and we assume a Bernoulli distribution for the response variables

\[ Y_i \sim \text{Ber}(P(Y = 1|X = x_i)). \]

Estimation of the \( \beta \) parameters can be obtained with maximum likelihood. Although there is no closed-form formula for finding the optimal \( \beta \) values, an optimization algorithm—derived from a Newton’s method—called the *Fisher scoring algorithm* is usually used. Logistic regression can be extended to handle responses that are polytomous (more than two alternative categories).

In the following, we show how to use R to perform logistic regression. We consider a dataset where the response variable is the presence or absence of myocardial infarction (MI) at the time of the survey. There are 160 cases in the data set, and a sample of 302 controls. The aim of the study was to establish the intensity of some heart disease risk factors.

#Load the library where there are the data

#The library contains the examples from the book of Hastie et al. (2009)

> library(ElemStatLearn)

> ?SAheart

SAheart  package:ElemStatLearn R Documentation

South African Hearth Disease Data

Description:

A retrospective sample of males in a heart-disease high-risk region of the Western Cape, South Africa.

Usage:

data(SAheart)

Format:

A data frame with 462 observations on the following 10 variables.

sbp systolic blood pressure
tobacco cumulative tobacco (kg)
ldl low density lipoprotein cholesterol
adiposity a numeric vector
famhist family history of heart disease, a factor with levels 'Absent' 'Present'
typea type-A behavior
obesity a numeric vector
alcohol current alcohol consumption
age age at onset
chd response, coronary heart disease

Details:
A retrospective sample of males in a heart-disease high-risk region of the Western Cape, South Africa. There are roughly two controls per case of CHD. Many of the CHD positive men have undergone blood pressure reduction treatment and other programs to reduce their risk factors after their CHD event. In some cases the measurements were made after these treatments. These data are taken from a larger dataset, described in Rousseauw et al, 1983, South African Medical Journal.

#Perform logistic regression with the glm function
# For more details about glm, look in Venables and Ripley (2002)
> myfit<-glm(chd~.,data=SAheart,family=binomial())
> summary(myfit)

Call:
glm(formula = chd ~ ., family = binomial(), data = SAheart)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
  -1.7781  -0.8213  -0.4387   0.8889   2.5435
Coefficients:

|            | Estimate | Std. Error | z value | Pr(>|z|) |
|------------|----------|------------|---------|----------|
| (Intercept)| -6.1507209 | 1.3082600  | -4.701  | 2.58e-06 *** |
| sbp        | 0.0065040 | 0.0057304  | 1.135   | 0.256374 |
| tobacco    | 0.0793764 | 0.0266028  | 2.984   | 0.002847 ** |
| ldl        | 0.1739239 | 0.0596617  | 2.915   | 0.003555 ** |
| adiposity  | 0.0185866 | 0.0292894  | 0.635   | 0.525700 |
| famhistPresent | 0.9253704 | 0.2278940  | 4.061   | 4.90e-05 *** |
| typea      | 0.0395950 | 0.0123202  | 3.214   | 0.001310 ** |
| obesity    | -0.0629099 | 0.0442477  | -1.422  | 0.155095 |
| alcohol    | 0.0001217 | 0.0044832  | 0.027   | 0.978350 |
| age        | 0.0452253 | 0.0121298  | 3.728   | 0.000193 *** |

---

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 596.11 on 461 degrees of freedom
Residual deviance: 472.14 on 452 degrees of freedom
AIC: 492.14

Number of Fisher Scoring iterations: 5

The summary provided by R shows the $Z$-scores associated with the different variables. A $Z$-score is equal to the value of the $\beta$ coefficient divided by its standard error and follows a standardized centered Gaussian variable under the null hypothesis ($\beta = 0$). Quoting Hastie et al. (2009),

there are some surprises in this table of coefficients, which must be interpreted with caution. Systolic blood pressure (sbp) is not significant! Nor is obesity, and its sign is negative. This confusion is a result of the correlation between the set of predictors. On their own (i.e. simple linear regression), both sbp and obesity are significant, and with positive sign. However, in the presence of many other correlated variables, they are no longer needed (and can even get a
negative sign). At this stage the analyst might do some model (variable) selection.

Let us do that in R

```r
> stepAIC(myfit) -> newfit
> summary(newfit)
```

Call:
```r
glm(formula = chd ~ tobacco + ldl + famhist + typea + age, family = binomial(),
    data = SAheart)
```

Deviance Residuals:

<table>
<thead>
<tr>
<th>Min</th>
<th>1Q</th>
<th>Median</th>
<th>3Q</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1.9165</td>
<td>-0.8054</td>
<td>-0.4429</td>
<td>0.9329</td>
<td>2.6139</td>
</tr>
</tbody>
</table>

Coefficients:

| Estimate | Std. Error | z value | Pr(>|z|) |
|----------|------------|---------|---------|
| (Intercept) | -6.44644 | 0.92087 | -7.000 | 2.55e-12 *** |
| tobacco | 0.08038 | 0.02588 | 3.106 | 0.00190 ** |
| ldl | 0.16199 | 0.05497 | 2.947 | 0.00321 ** |
| famhistPresent | 0.90818 | 0.22576 | 4.023 | 5.75e-05 *** |
| typea | 0.03712 | 0.01217 | 3.051 | 0.00228 ** |
| age | 0.05046 | 0.01021 | 4.944 | 7.65e-07 *** |

---

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 596.11 on 461 degrees of freedom

Residual deviance: 475.69 on 456 degrees of freedom

AIC: 487.69

Number of Fisher Scoring iterations: 5

Here we see that the retained variables are: tobacco consumption, cholesterol, family history, type A behavior.
To understand what the coefficient means in logistic regression, we give the last words to Hastie and colleagues (2009).

How does one interpret a coefficient of 0.080 (Std. Error = 0.026) for tobacco, for example? Tobacco is measured in total lifetime usage in kilograms. Thus an increase of 1kg in lifetime tobacco usage accounts for an increase in the odds of coronary heart disease (the odds are $p/(1 - p)$) of $e^{0.080} = 1.083$ or 8.4% (see equation 19). Incorporating the standard error we get an approximate 95% confidence interval of $e^{0.080 \pm 2 \times 0.026} = (1.03, 1.14)$.

References


Section 3 about PCA


Section 2 about ANOVA


The holy book, sections 1, 3, and 4 mostly come from this book

Saporta G (1990) Probabilités, analyse des données et statistique, 1ère édition, Technip

Section 2 about ANOVA


The reference book for modern statistics in S (or R that is almost the same)