POLITECNICO DI TORINO

Mathematical Engineering

Master's Thesis

Statistics and Volleyball: detection of the most significant skills and their importance in the results prediction

Supervisors:

Prof. Franco Pellerey

Prof. José María F.dez Ponce

Candidate:

Francesca Leo

A.Y. 2018/2019

A Te che non sarai con me in nessuna foto,
ma sei dentro ad ogni singolo gesto
compiuto per arrivare fino a qui.
Ti voglio bene papà.

Contents

Abstract

In this work we tried to identify the most relevant technical skills in the volleyball world and thanks to them to predict the result of a match. In particular, our dataset contains 120 male volleyball matches played during the 2018 Nations League by the best national teams. First of all, we used logistic regression, but it requires that both the observations and the predictors are independent. For this reason, the algorithm was initially implemented by extracting independent subsets of variables, but the maximum model accuracy was 78.13%. In order to improve this percentage, a phase of pre-processing on the original dataset has been started: Principal Component Analysis (PCA). Logistic regression is again implemented basing on new generated variables and the accuracy of the prediction increases to 82.59%. A second technique called Random Forest allowed us to predict the result of each set with an accuracy very similar to the previous one without any phase of preparation of data. It provides us a direct ranking of the considered variables. The offensive variables are underlined as very significant compared to the defensive ones. Finally, two non-parametric tests were selected to compare the empirical distribution of data on winners and losers, the Kolmogorov-Smirnov test between two samples and that of Mann-Whitney. They confirm the above: the reception phase has both very similar Empirical Cumulative Distribution Functions (ECDF) and almost overlapped density curves. At the contrary, for relevant positive variables the winners' ECDF is under the losers' one and the density is visibly shifted on the right values. A further study of this type of data could be the implementation of the PP-plot to make a stochastic order of samples.

Abstract

In questo elaborato si è cercato di individuare i gesti tecnici più rilevanti nel mondo della pallavolo e, grazie ad essi, predire il risultato di un match. In particolare, il dataset utilizzato contiene 120 partite di volley maschile giocate durante la Nations League 2018. Il primo strumento usato è la regressione logistica, che richiede sia osservazioni che predittori tra loro indipendenti. Inizialmente si è implementato l'algoritmo estraendo sottoinsiemi indipendenti di variabili, ma la massima accuratezza ottenuta è stata del 78.13%. Per migliorare tale percentuale, è stata avviata una fase di pre-processing sulle variabili: l'Analisi delle Componenti Principali (PCA). La regressione è stata nuovamente implementata basandosi sulle nuove variabili generate e l'accuratezza della predizione è salita all'82.59%. Una seconda tecnica denominata Random Forest ha concesso di predire il risultato di ogni set con un'accuratezza molto simile alla precedente e senza alcuna fase di preparazione dei dati. Tale algoritmo è in grado di fornirci un ranking diretto delle variabili considerate. Le variabili offensive sono risultate decisamente significative rispetto a quelle di difesa. Infine, sono stati selezionati due test non parametrici di confronto tra le distribuzioni empiriche dei dati relativi a vincitori e perdenti, il test di Kolmogorov-Smirnov tra due campioni e quello di Mann-Whitney. Essi confermano quanto riscontrato in precedenza: la fase di ricezione, presenta sia Funzioni di Distribuzioni Cumulative Empiriche (ECDF) che curve di densità molto simili. Al contrario, per variabili psitive rilevanti l'ECDF dei vincitori è situata sotto quella dei perdenti e la densità è visibilmente spostata verso destra. Un approfondimento relativo a questa tipologia di dati potrebbe essere l'implementazione del PP-plot per effettuare un ordinamento stocastico dei campioni.

Chapter 1

Introduction

"*The beginning is the most important part of the work*"

Plato, 398 a.C.

A simple equation is the origin of this work, $P = R$ that is Potential=Result. This is what every sportsman desires from his performance, but how to reach this goal? In the last years, to combine sport and statistical analysis is increasingly common. Machine learning techniques allow to improve game strategies and analyze large quantities of data from every sport. According to WhaTech channel reports, in 2016 the usage of analytics in sports like baseball has increased to more than 90%, football more than 50% and basketball more than 75%.

The focus of the work is on volleyball. Also in this environment statistical analysis is more and more present and the existence of a dedicated software called DataVolley demonstrates it. In this team sport six players interact with each other in a 81m^2 field to win the opposite team over a net whose top is 2.43m for masculine competitions and 2.24 for female ones. Volleyball matches are composed by sets of 25 points. The first team that wins three sets is the winner of the match considering that if they arrive to the fifth set, they play only to 15 points.

It is appropriate to apply mathematical models to volleyball because six technical gestures are repeated hundreds of times in every match:

1. **Serve**, the skill that begins an action from behind the back-line to the opponent court.

2. **Reception**, the skill that is contrary to 1. It tries to prevent the ball from hitting the court.

3. **Set**, the skill that pushes the ball such that a teammate can hit it into the opponent's court.

4. **Spike**, the attack gesture that tries to score a point.

5. **Block**, the first skill that wants to stop or alter an offensive spike of the opposite team.

6. **Dig**, similarly to the reception prevents the ball from hitting the court, but after an attack gesture.

The aim of this work is to understand if it is possible to predict the result of a match basing on some of these variables and to detect their importance in this prediction. The difficulty is to manage percentages about skills that do not have a known distribution and that, often, are not independent. For these reasons, after a descriptive analysis of data, we have selected logistic regression and random forest models.
In particular, Chapter 2 describes the theory behind every implemented technique while in Chapter 3 is provided a description of the volleyball dataset. Before the application of logistic regression, we need to select the predictors to be used. For this reason in Chapter 4 we apply an unsupervised algorithm of machine learning, the Principal Component Analysis (PCA). The most interesting part of the work is probably in Chapter 5, where we really use the previous models to predict the result of a match and check their operation and accuracy. Finally, in Chapter 6, two tests are implemented to confirm the obtained outcome. The principal difference is that the first models consider a single team at a time and use their skills to predict the result. At the contrary, these last tests take into account the two opposite teams in every match and

how much their performances differ. Results and deals to think about are illustrated in Chapter 7 and the only way to arrive there is to start.

Chapter 2

Statistical analysis and machine learning theory

"Luck is not involved.
Our strategy in Australia
was based on statistical data
and the calculation of the probabilities.
And it turned out to be right."

Sebastian Vettel, 2018

The purpose of this chapter is to provide necessary and solid theoretical basis to our applications and results. The reader can observe an almost perfect correspondence with the next chapters in which we are going to apply the explained methods to a chosen dataset.

2.1 Pearson's correlation and Bartlett's test

The Pearson's coefficient, also called Pearson Product-Moment Correlation, is the ratio between the covariance of two numerical variables and the square root of their variances. We consider N observations of two variables: $\{x_1, x_2, ..., x_N\}$ and $\{y_1, y_2, ..., y_N\}$ and their sample mean:

$$\bar{x} = \frac{1}{N} \sum_i x_i \tag{2.1}$$

$$\bar{y} = \frac{1}{N} \sum_i y_i \tag{2.2}$$

Now we can calculate their sample variance:

$$\sigma_x^2 = \frac{1}{N-1}\sum_i (x_i - \bar{x})^2 \tag{2.3}$$

$$\sigma_y^2 = \frac{1}{N-1}\sum_i (y_i - \bar{y})^2 \tag{2.4}$$

and their covariance:

$$C_{xy} = \frac{1}{N-1}\sum_i (x_i - \bar{x})(y_i - \bar{y}) \tag{2.5}$$

to have every necessary element for the Pearson's coefficient formula:

$$r = \frac{C_{xy}}{\sigma_x \sigma_y} \tag{2.6}$$

The coefficient *r* is measured on a scale with no units and can take a value from -1 to 1. More the coefficient absolute value is near to 1, more the correlation between x and y is strong. The sign adds some information, in fact if the coefficient is positive then a positive correlation would exist indicating that a large number of x is associated to a large number of y. The opposite situation occurs when the sign of the correlation is negative.

An assumption of Pearson's statistic (2.6) is that the tested relationship is a linear one that could be easy to detect also thanks to the so-called scatter plot of the two variables values [20]. Observing the left part of the Figure 2.1 we see two very strongly related variables, at the contrary in the right part two very uncorrelated variables without any recognizable path. If all the points on the scatter plot lay on a straight line, then a perfect correlation exists and the correlation coefficient is 1 or -1 depending on the slope of the line. If $y_i = Ax_i + B$ and $\bar{y} = A\bar{x} + B$, the variables x and y are perfectly linearly related, so we can prove that r is equal to ± 1.

$$C_{xy} = \frac{1}{N-1}\sum_i (x_i - \bar{x})(y_i - \bar{y}) = \frac{1}{N-1}\sum_i (x_i - \bar{x})(Ax_i + B - A\bar{x} - B) = \tag{2.7}$$

$$= \frac{A}{N-1}\sum_i (x_i - \bar{x})^2$$

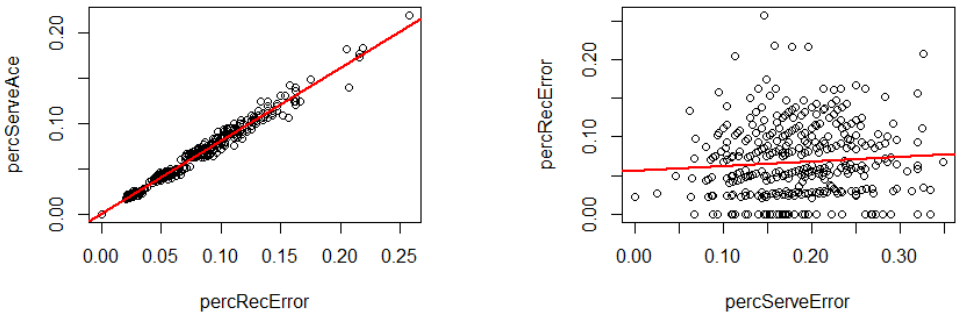


Figure 2.1: Strong and weak correlation examples

$$\sigma_x^2 = \frac{1}{N-1}\sum_i (x_i - \bar{x})^2$$

$$\sigma_y^2 = \frac{A^2}{N-1}\sum_i (x_i - \bar{x})^2$$

And finally:

$$r = \frac{C_{xy}}{\sigma_x \sigma_y} = \frac{A}{|A|} = \pm 1 \tag{2.8}$$

It is important to remember that r does not evaluate hypothesized relationship between data, does not test a hypothesis for the origin of the data and does not give more weight to some data points respect to others [7], moreover the temporal nature of the data is ignored. A relevant feature related to the statistical significance of r is the sample size, large samples accept the correlation coefficient to have a smaller value for the association to be significant, while two variables concerning a few data collection need $r > 0.5$ to be considered linearly correlated.

It is evident that

$$r = \frac{C_{xy}}{\sigma_x \sigma_y} = \frac{C_{yx}}{\sigma_y \sigma_x} \tag{2.9}$$

so when we have a set of more than two variables and we want to calculate the paired correlation and to build the so-called correlation matrix, the result is a symmetric matrix with ones on the principal diagonal because each variable is perfectly correlated to itself. Clearly, the correlation matrix could be the first tool to analyze the correlation

in a set of variables, but to read properly it could be difficult when we do not have a lot of knowledge about data and when the number of variables is very large.

The Bartlett's test is one of the several method that can help us in these situations, with the characteristic of being a parametric test: it works with normal distributions of variables. There is also a formal Pearson's test but it only acts on pairs of variables, the null hypothesis is that r_{xy} can be considered a null coefficient while the alternative hypothesis states that x and y are correlated to each other. We define α as the level of significance of the test, normally $\alpha = 0.05$, and we compare it with the p-value of the t-test. The value of the t-statistic is

$$t = \frac{r}{\sqrt{1 - r^2}} \sqrt{N - 2} \tag{2.10}$$

and the corresponding p-value is determined using the t distribution table for freedom degrees $df = N - 2$. If the p-value of the test is less than the significance level α the correlation between x and y is significant and the null hypothesis is rejected.

Bartlett's test aim is to measure the similarity between a correlation matrix and an identity one, any number of variables in play, it tests whether or not the off-diagonal coefficients are significantly different from zero. *"Little is known of the power of such a test, but an intuitive judgment would suggest that its power is reasonably high against normally correlated alternatives"* (Kendall, M. G., 1957).

The statistic of the test is

$$-ln(det(R))[(N - 1) - \frac{(2p + 5)}{6}]$$

and it is distribuited as chi-square if R is an identity matrix. N is the sample size, p is the number of involved variables, R is the sample correlation matrix and the degrees of freedom of χ^2 are $\frac{p(p-1)}{2}$. The null hypothesis is $H_0 : R_{pop} = I$, the problem is to detect the probability of rejecting it and the level of significance of the test (i.e. 0.05). To solve directly the problem is considered too difficult, so in [11] we find a practical application to a sample set of size N with induced higher and higher correlation among variables. The results infer that with $N = 20$ we reject almost

certainly the null hypothesis if ten considered variables are inter-correlated through a population correlation coefficient of 0.36 or more. Thanks to ten replicates, for this value of induced correlation, the mean of χ^2 values is 92.91.

For $N = 200$ the correlation value that leads us to reject the null hypothesis is 0.09 and the mean of χ^2 values is 102.18. The value of r is smaller because the sample size is ten times greater than the previous one and in this case also a small esteem of correlation can result significant. This evidence suggest that Bartlett's test of the significance of a correlation matrix is quite sensitive for both small and large N but it is also very sensitive to non-normal variables: the p-value of the test will be very small also in case of no strong correlation when the variables are not good fitted by a Gaussian distribution. Its power appears to be quite high to be considered a first fundamental step before factorial analysis by the most part of the modern statisticians.

Unfortunately, not all collinearity problems are visible by the analysis of the correlation matrix: it is possible that collinearity exists among three or more variables even if there is not a specific pair of variables with a particularly high correlation. In [10] this situation is called multicollinearity and it is inspected thanks to the computation of the *variance inflation factor* (VIF). It owes its name to the fact that reports how much the variance of a linear regression model estimated coefficients increases because of the collinearity between almost independent variables. Its value represents *"how much of a regressor's variability is explained by the rest of the regressors in the model due to correlation among those regressors."* [5]. For n independent variables:

$$VIF_i = \frac{1}{1 - r_i^2} \quad \text{for } i = 1, ..., n \tag{2.11}$$

where r_i^2 is the coefficient of determination obtained by fitting a regression model for the ith variable on the other $n - 1$ variables. It is evident that an environment with a perfectly orthogonal set of variables will present $VIF_i = 1$ for i=1,...,n. Although there is not any formal criterion to define when the variance inflation factor is too large, 5 or 10 are often considered cutoff values to determine if the collinearity is strong enough to require remedial measures.

2.2 Theory of Principal Component Analysis

In a certain sense, this section also concerns the dependence of the variables, or better to say it is in charge to explain how we can deal with it. The technique that we are going to use is the Principal Component Analysis (PCA), an *unsupervised learning algorithm*: it is not interested on prediction but it considers a set of features without any related response.

Usually, this tool is used like a kind of pre-processing before of the use of supervised techniques, but how does it work? First of all, the PCA is able to summarize a large set of correlated variables in a smaller number of new variables that are almost representative as the total and overall that are independent of each other, moreover it serves as a tool for data visualization. *"In particular, we would like to find a low-dimensional representation of the data that captures as much of the information as possible."* [10]. For example, the simplest result would be to have a representation of data in a two-dimensional space to plot and observe them, but the real difficulty is to keep in this representation the majority of the information that is provided to us by the data explanatory features, that are more than two, we say *n*.

We suppose to have m observations described by n features $X_1, X_2, ..., X_n$, PCA looks for a small number of dimensions that are as interesting as possible. Here the concept of interesting is measured by the amount that the observations vary along each dimension, called principal component, so it is important to understand the manner in which these dimensions are found.

The *first principal component* of the previous set of features is their normalized linear combination with the largest variance:

$$Z_1 = \phi_{11}X_1 + \phi_{21}X_2 + ... + \phi_{n1}X_n \tag{2.12}$$

$\phi_{11}, ..., \phi_{n1}$ are the *loadings* of the first principal component, so the first principal component loading vector is $\phi_1 = (\phi_{11}\phi_{21}...\phi_{n1})^T$ and it solves a sort of optimization

problem:

$$\max_{\phi_{11},...,\phi_{n1}} \left\{ \frac{1}{m} \sum_{i=1}^{m} \left(\sum_{j=1}^{n} \phi_{j1} x_{ij} \right)^2 \right\} \quad subject \ \ to \sum_{j=1}^{n} \phi_{j1}^2 = 1 \tag{2.13}$$

The solution comes from a simple eigen-decomposition but the details are out from the scope of this work. Now we define the linear combination:

$$\sum_{j=1}^{n} \phi_{j1} x_{ij} = z_{i1} \tag{2.14}$$

and since we are only interested in the variance of our variables we assume that each of the variables have mean zero and consequently the average of $z_{11}, ..., z_{m1}$ will be zero. Hence the function that we are maximizing in (2.13) is $\frac{1}{m} \sum_{i=1}^{m} z_{i1}^2$ and it is just the sample variance of the m values of z_{i1}.

The loading vector has a geometric interpretation because with its elements defines a direction in variables space where the data vary the most. If we project the m data in this direction, the projected values are the principal component *scores* $z_{11}, ..., z_{m1}$. Now, how to compute the second principal component is easy, we want the linear combination of $X_1, X_2, ..., X_n$ with the maximal variance and that is $uncorrelated$ with Z_1.

$$z_{i2} = \phi_{12} x_{i1} + \phi_{22} x_{i2} + ... + \phi_{n2} x_{in} \tag{2.15}$$

The geometric interpretation of the no-correlation is to require for ϕ_2 an orthogonal direction to ϕ_1.

Due to the use of linear combinations PCA provides low-dimensional linear surfaces that are closest to the observations: with the first component it seeks a single dimension of the data that lies as close as possible to all of the data points, with two of them PCA spans the closest plan to the m observations and so on. Each loading vector of a PCA specifies a direction in a n-directional space to rotate the initial surface, because of the fact that the sign has no effect on the direction, each loading vector is unique, up to a sign flip. The same occurs for the score vectors, in fact the variance of Z is the same of $-Z$. But the natural question is how much of the variance is explained by

each principal component? The total variance of a dataset with mean equal to zero is:

$$\sum_{j=1}^{n} Var(X_j) = \sum_{j=1}^{n} \frac{1}{m} \sum_{i=1}^{m} x_{ij}^2 \tag{2.16}$$

while the variance explained only thanks to the p-th component is:

$$\frac{1}{m} \sum_{i=1}^{m} z_{ip}^2 = \frac{1}{m} \sum_{i=1}^{m} \left(\sum_{j=1}^{n} \phi_{jp} x_{ij} \right)^2 \tag{2.17}$$

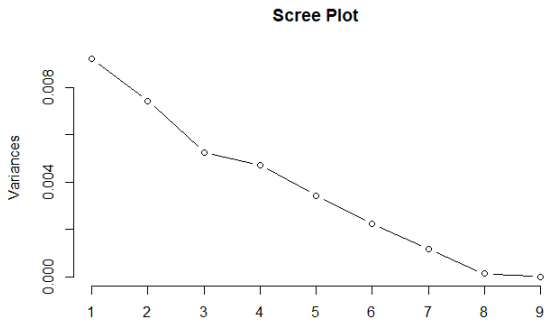
If we want the proportion of variance we just compute:

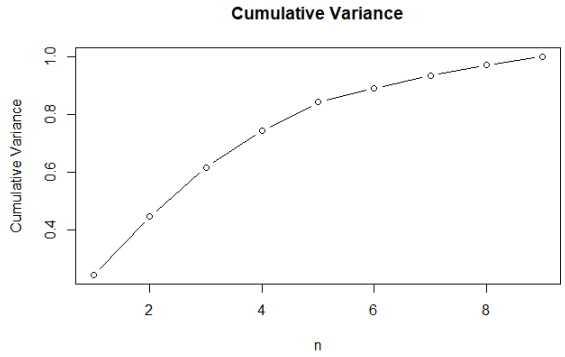
$$\frac{\sum_{i=1}^{m} \left(\sum_{j=1}^{n} \phi_{jp} x_{ij} \right)^2}{\sum_{j=1}^{n} \sum_{i=1}^{m} x_{ij}^2} \tag{2.18}$$

In total there are $\min(m-1, n)$ principal components and if we iteratively sum the proportion of variance that they explain we obtain one.

Usually we are not interested in all the principal components or rather we are willing to lose some information if this allows us to work with a less large number of variables, overall if we are facing with a huge dataset. Unfortunately there is not any formal computing or method to evince how many components we have to deal with, but a graphical method can help us. We plot a sort of *elbow graph* that is called *scree plot* and wee look for *"a point at which the proportion of variance explained by each subsequent principal component drops off"* [10]. In general the choice of the number of components remains a fairly subjective aspect of the analysis.

Figure 2.2: Example of Proportion of Variance Explained and Cumulative Variance





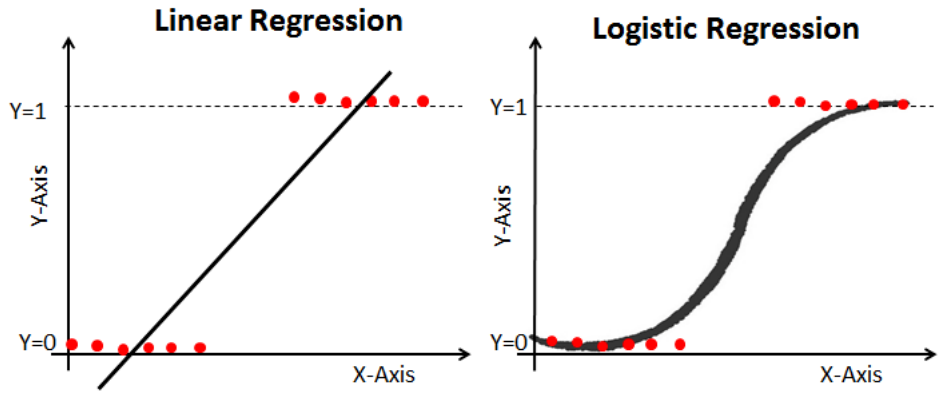
2.3 Theory about Logistic Regression Model

Machine learning algorithms are split into two categories, in the previous section we have seen an example of unsupervised technique while the logistic regression is a supervised algorithm that provides us a method to predict a binomial result Y starting from some independent variables called *predictors* $\boldsymbol{X}$. The dependent variable, said outcome, can assume only two values, they are labeled 0 and 1, but they can indicate every possible qualitative binary response like dead/alive, off/on and so on.
The equation for a linear regression model would be:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k \tag{2.19}$$

and the obtained $\boldsymbol{X}\hat{\beta}$ could estimate the $Pr(Y = 1|\mathbf{X})$ instead of the real value of Y, but some values lay outside the [0,1] range and it would be difficult to give them an interpretation in terms of probability. We can immediately notice one of the most important difference between the two models: *"rather than modeling this response Y directly, logistic regression models the probability that Y belongs to a particular category"* [10].

Figure 2.3: Linear vs Logistic Regression



We want to find the relationship between $\mathbf{X}$ and $p(\mathbf{X}) = Pr(Y = 1|\mathbf{X})$ that gives us as more information as possible about Y. We have already explained that using the straight line represented in (2.19) to fit a binary response we could wrongly predict $p(\mathbf{X}) < 0$ or $p(\mathbf{X}) > 1$ depending on $\mathbf{X}$ values. In the logistic regression model we avoid the problem using a function that has its image in [0,1]:

$$p(\mathbf{X}) = \frac{e^{\beta_0+\beta_1 X_1+\cdot+\beta_k X_k}}{1 + e^{\beta_0+\beta_1 X_1+\cdot+\beta_k X_k}} \tag{2.20}$$

More than one method exists to estimate the vector of coefficients β, but we are going to explain only the maximum likelihood method that is the one used in our applications in Section 5.1. The idea behind this technique is very simple and is to choose $\beta_0, \beta_1, \cdots, \beta_k$ that pushed into the model yield a $p(\mathbf{X})$ close to 1 if the initial response is $Y = 1$ or, in the opposite case, close to 0. The function that formalizes this intuition is the so called, *likelihood function* that gives the name to the model and that we want to maximize:

$$l(\beta_0, \beta_1, \cdots, \beta_k) = \prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}=0} (1 - p(x_{i'})) \tag{2.21}$$

Every statistical software can solve the problem of optimization of (2.21) without a high computational effort so we do not want to focus on all passages.

From (2.20) we evince that:

$$\frac{p(\mathbf{X})}{1-p(\mathbf{X})} = e^{\beta_0+\beta_1 X_1+\cdot+\beta_k X_k} \tag{2.22}$$

The left hand-side of the equation is called $odds$ and can take on any value in $[0, +\infty)$, values very close to 0 indicate that the probability of $Y = 1$ is very low, on the contrary, if the $odds$ is near to ∞, $Y = 1$ is an almost certain event. Simply computing the $log - odds$ we can see that it is linear in $\mathbf{X}$:

$$\log\left(\frac{p(\mathbf{X})}{1-p(\mathbf{X})}\right) = \beta_0 + \beta_1 X_1 + \cdot + \beta_k X_k \tag{2.23}$$

Since the relationship between $\mathbf{X}$ and $p(\mathbf{X})$ seen in (2.20) is not linear, β_1 is not the change in $p(\mathbf{X})$ associated to a one-unit increase in X_1, but thanks to (2.23) we are sure that if β_1 is positive then increasing X_1 will be associated with increasing $p(\mathbf{X})$ and that the opposite situation occurs if $\beta_1 < 0$. The logistic function always produces an $S - shaped$ curve, so the amount of the change of $p(\mathbf{X})$ due to a one-unit change of $\mathbf{X}$ depends on the current value of $\mathbf{X}$.

Once the coefficients have been estimated, the computation of 2.20 is easy and takes us to a value of the $\hat{p}(\mathbf{X}) = \hat{Pr}(Y = 1|\mathbf{X})$ that has to be analyzed according to the requirements of the study. For example, one might predict $Y = 1$ if $\hat{p}(\mathbf{X}) > 0.5$ or can be more conservative and predict the success only for values of $\hat{p}(\mathbf{X})$ higher than 0.7 and so on.

To better understand some aspects of logistic regression we can observe the Table 2.1 in which an example with five predictors has been implemented. First of all, the values of coefficients underline that the increasing of all variables contributes to raise the probability that the result is equal to one except for the second variable that has the opposite effect. It is possible to compute the accuracy of these coefficients thanks to the second column of the table, the Standard Errors. They are useful to perform hypothesis test on each coefficient in which the null hypothesis is:

$$H_0 : \beta_1 = 0$$

Table 2.1: Example of logistic regression model

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-9.0638	0.8414	-10.77	<0.0001
Variable1	14.4852	2.2963	6.31	<0.0001
Variable2	-7.3471	1.3426	-5.47	<0.0001
Variable3	0.6442	0.9779	0.66	0.5101
Variable4	15.8707	1.4151	11.22	<0.0001
Variable5	8.9546	1.1264	7.95	<0.0001

versus the alternative hypothesis

$$H_1 : \beta_1 \neq 0$$

If the null hypothesis is accepted we state that there is no relationship between the output and the variable X_1. In practice, we construct the z-statistic given by the third column of the Table 2.1:

$$z = \frac{\hat{\beta}_1}{SE(\hat{\beta}_1)} \tag{2.24}$$

A large absolute value of (2.24) indicates evidence against the null hypothesis, to formalize the test it is computed the probability, called p-value, of observing values equal to $|z|$ or larger assuming $\beta_1 = 0$. We reject the null hypothesis inferring that there is an association between the considered predictor and the response if the p-value is smaller than 1% or 5%, depending on the level of the significance that we want for the test. It is obvious that small p-values correspond to high absolute values of z and the interpretation is: *"a small p-value indicates that it is unlikely to observe such a substantial association between the predictor and the response due to chance, in the absence of any real association between the predictor and the response"* [10]. For what we have explained above we can state that in the example of the Table 2.1 only the fifth variable has not relationship with the outcome, while the others are significant for the prediction model.

2.3.1 Goodness-of-fit

Thanks to the z and the p-values we can obtain a sort of rank of the significance that the predicting variables have in the model, but another of the most important aim of the regression model is to use the estimated coefficients to predict the desired result, in the logistic case the dichotomous dependant variable Y. How can we evaluate the efficiency of the model and the goodness of fit? Sure we can use more than one criterion, some of them will be more mathematically correct than others. We know that the perfect model does not exist, for the definition of model itself, but we can try to do the best with what we have, that is for example comparing the results of the evaluation criteria to identify among many models the one that works better. First of all, in front of a large dataset it result useful to split it into the training and the test part. Normally the first is the bigger one because we really use these data to $train$ the model, it means that thanks to the observations in this subset the model estimates the coefficient to be used in logistic regression. To improve the efficacy of the technique these observations must be independent from each other. After this step we use the $trained$ model on the other part of data to $test$ if these same coefficients are good to make the model able to detect the the correct response. Every statistic software, as R, has the command `predict` to implement a model, previously performed, on a selected set of data.

Probably, the most common method to evaluate the exactness of predictive methods is the confusion matrix, a sort of summary about the results of the analysis. Its layout is represented here:

$$\begin{array}{c} \\ \text{Actual Y=1} \\ \text{Actual Y=0} \end{array} \begin{array}{c} \begin{array}{cc} \text{Predicted Y=1} & \text{Predicted Y=0} \end{array} \\ \begin{pmatrix} TP & FN \\ FP & TN \end{pmatrix} \end{array}$$

By summing all values in the matrix we obtain the number of total observation: in the first row we have the actual $Y = 1$, in the second row we have the real number of $Y = 0$, in the first column there are the predicted $Y = 1$ and in the second one the

predicted $Y = 0$. The principal diagonal, with True Positive (TP) and True Negative (TN) values, tells us how many results are correctly classified, respectively $Y = 1$ and $Y = 0$. Instead the False Negative values (FN) detect how many $Y = 1$ the model erroneously classify like $Y = 0$ and alternatively for the False Positive values (FP).
The accuracy of the model indicates how often is the classifier correct:

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN}.$$

Thanks to the confusion matrix it is also possible to compute other rates related to the model, but we leave this knowledge for the most practical section.
Another estimator of the quality of statistical models is the Akaike information criterion, its value is defined like

$$AIC = -2\log[L(\hat{\theta})] + 2K \tag{2.25}$$

where K is the number of estimable parameters and $L(\hat{\theta})$ is the maximum value of the likelihood function of the parameter vector θ for the model. Log-likelihood is a measure of model fit, the higher is the value, the better the fit, so the AIC takes in account the goodness of fit but also a sort of penalty that discourages the overfitting because it increases with the number of parameters.
We have to underline that: *"AIC is not a criterion for the estimation of the true order but the one for the best model"* [19], it means that AIC won't say anything about the absolute quality of a model, but compares the quality of a set of models to each other. Thanks to this estimator we can rank the selected models from best to worst, but we have to detect with other techniques if all the candidates fit poorly our dataset.
From (2.25) we immediately understand that models with a small AIC value are preferable. If there is a set of models, for the ith we define the AIC difference as:

$$\Delta AIC_i = AIC_i - \min AIC$$

The first idea is to use this quantity to direct rank, but in [17] it is suggested that

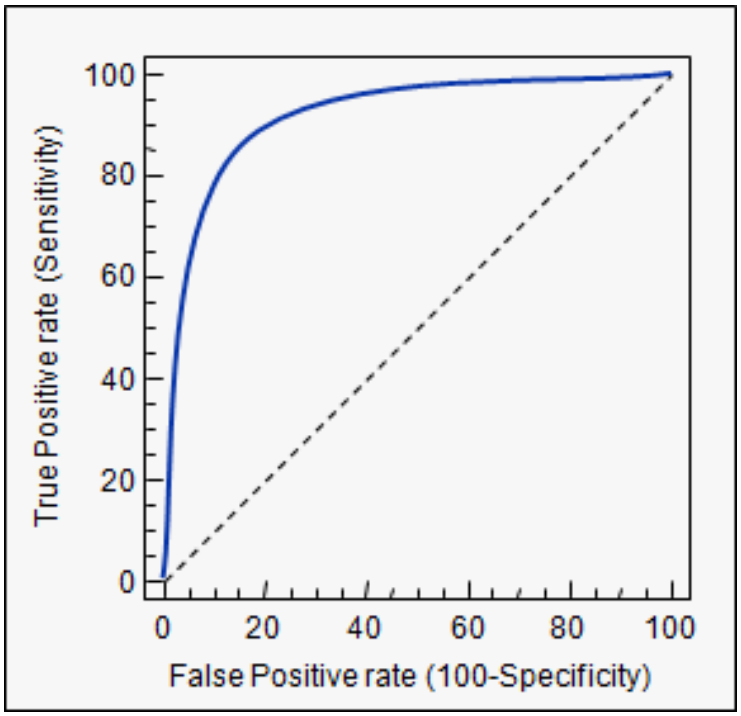
$$\exp\left(\frac{-\Delta AIC_i}{2}\right)$$

can be interpreted as being proportional to the probability that the ith model minimizes the estimated information loss and this probability approaches to zero when AIC_i is large.

The last presented criterion to study the exactness of a predictive method in presence of binary output is the Receiver Operating Characteristic curve, i.e., ROC curve. We use the AUC (Area Under the Curve) as a measure of a classifier's performance, in practice this is a graphical method strictly related with the confusion matrix.

Suppose that t is the value of a threshold so that an individual is allocated to population 1 if its classification score s exceeds t and otherwise to population 0. The probability that an individual from 1 is correctly classified is the *true positive rate* and we use $tp = p(s > t|1)$ to represent it. At the contrary, the probability that an individual from 1 is misclassified is $fn = p(s \leq t|1)$. Equally for the population 0 we use $fp = p(s > t|0)$ and $tn = p(s \leq t|0)$ to define the *false positive* and the *true negative* rate. The ROC curve is obtained on varying t and plotting (fp, tp) where the *false positive rate* is the value on the horizontal axis and the *true positive rate* is the value on the vertical one. Figure 2.4 shows an example of ROC curve. Clearly, for good performance the requirement is high *"true"* and low *"false"*, knowing that $tp + fn = 1$ and $fp + tn = 1$. The worth of a classifier can be judged by how much the two distribution of its scores $p(s|0)$ and $p(s|1)$ differ: the classifier will be least successful when the two population are exactly the same. In such a case the probability of allocating an individual to population 1 is the same whether that individual has come from 1 or 0. In this case, even if t varies, fp and tp will be always equal and the ROC curve is not a curve anymore, but a straight line from $(0, 0)$ to $(1, 1)$. The most desirable result would be the perfect allocation of each individual, so we would have at least one t in which $tp = 1$ and $fp = 0$. Since the ROC curve focuses only on the probabilities that $s > t$, for all smaller values of t, tp will be equal to 1 and fp varies from 0 to 1, while for all larger values of t, fp will be equal to 0 and tp varies from 1 to 0. The perfect, but usually unattainable, situation would be a straight line from $(0, 0)$ to $(0, 1)$ joined to a straight line from $(0, 1)$ to $(1, 1)$.

Figure 2.4: Example of ROC curve



"In practice, the ROC curve will be a continuous curve lying between these two extremes, so it will lie in the upper triangle of the graph. The closer it comes to the top left-hand corner of the graph, the closer do we approach a situation of complete separation between populations, and hence the better is the performance of the classifier" [12].

2.4 Theory about Random Forest technique

The purpose of this chapter is to understand how the random forest technique works on data to classify or do regression on them. Random forests are very simple to train, so the authors in [8] make grand claims about their success: "most accurate", "most interpretable", and so on. They are classified as ensemble learning methods because they build a large collection of trees to average their results. In this case, we need a brief introduction to the construction of trees. They can be applied to both regression and classification problems, in this last case random forests use their majority vote.

2.4.1 Decision Trees

Decision trees are divided in regression and classification trees. A *classification tree* is very similar to a *regression tree* but is used for the prediction of a qualitative response rather than a quantitative one. In particular every observation can be classified in one of K different classes, with $K \geq 2$.

We now discuss how to build a tree dividing the process in two steps:

1. The predictor space, that is the set of possible values for $X_1, X_2, \cdots, X_p$ is divided into J distinct and non overlapping regions, $R_1, R_2, \cdots, R_j$

2. Each observation present in the region R_j is predicted as the mean of the response values for the training observations in R_j. For classification, the observation belongs to the most commonly occurring class.

How to implement the Step 1? It would be computationally infeasible to consider every possible partition of the feature space, so a *recursive binary splitting* approach is used. It is *top-down* because it begins with all the observation belonging to a single region and each successive split into two branches goes further down on the tree. This approach is also said $greedy$ because at each step the algorithm does not look ahead in the future, but just makes the best split at that particular step.

To perform recursive binary splitting we select the predictor X_j and the threshold s such that splitting the predictor space into the regions $R_1(j, s) = \{X|X_j < s\}$ and $R_2(j, s) = \{X|X_j \geq s\}$ leads to the greatest reduction in the Residual Sum of Squares (RSS). We seek the values of j and s that minimize:

$$\sum_{i:x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i:x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2 \tag{2.26}$$

Once j and s are found the process is repeated splitting one of the two previously identified regions. The algorithm continues until a stopping criterion, as the minimum number of observations in a region, is reached. At the end $R_1, R_2, \cdots, R_j$ have been created to minimize RSS and we can predict the response averaging the training

observations in the region to which the considered observation belongs.
The measure to select the best split is different between regression and classification trees, in particular we have seen that regression trees minimize RSS, while for the classification ones we expound other three indexes.

- Classification Error Rate:

$$E = 1 - \max_k(\hat{p}_{mk}) \tag{2.27}$$

  is the fraction of the training observations in a region that do not belong to the most common class. Here $\hat{p}_{mk}$ represents the proportion of training observations in the mth region that are from the kth class.

- Gini Index:

$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}) \tag{2.28}$$

  a measure of total variance in the K classes.

- Cross-entropy:

$$D = -\sum_{k=1}^{K} \hat{p}_{mk} \log \hat{p}_{mk}. \tag{2.29}$$

The Formula (2.27) is not sensitive to the tree-growing, while (2.28) and (2.29) are referred to the node $purity$, in fact they are as smaller as greater is the number of observations of a single class in the node. Usually, these last two measures are used to evaluate the quality of a split because of their sensitivity to node purity, *"but the classification error rate is preferable if prediction accuracy of the final pruned tree is the goal"* [10].
In the last quote there is a reference to a $pruned$ tree, the reason is that the above generated tree could be too complex. A smaller tree with fewer splits might lead to lower variance and to a better interpretation of the results, without overfitting the data. The idea is to create a very large T_0, stopping when the chosen minimum node size is reached. Then, using the *cost-complexity pruning* we minimize a cost function and find

a subtree T_α. We want to detect the best trade-off between tree size and goodness of fit to the data because a too small tree might not capture the totality of information and structure. For the details about the algorithm, that are out of the interest of this work, see [4] ad [18]. Random forest builds unpruned trees.

2.4.2 Random Forest Implementation

As we have already seen, *Random Forest* is a technique that builds a committee of decision trees. In particular, successive trees do not depend on earlier ones, each tree is independently constructed using a bootstrap sample of the dataset. Hence, each tree is fitted to a different dataset of the same size as the original one. Moreover, to avoid correlation, a subset of predictors is randomly chosen at each node. In fact, if there was a very strong predictor, this would be used in the top split of every tree and the predictions would be highly correlated.
More in detail, we will show the algorithm found in [13] for the construction of random forests:

1. Draw n bootstrap samples from the original data.

2. For each of the bootstrap samples, grow an unpruned classification or regression trees. At each node, rather than choosing the best split among all predictors, randomly choose $m \leq p$ predictors and do the best split among those variables.

3. Predict new data by aggregating the predictions of the n trees (i.e., majority votes for classification, average for regression).

Clearly, the algorithm is user-friendly having only two parameters: n, the number of trees in the forest and m, the number of variables in the subset at each node. Usually the choice is $m = \sqrt{p}$ or even 1 can give very good performance for some data, anyway m can always be adapted to the problem. To select the best number of trees, a cross-validation function can be implemented, computing the mean squared error for different values of n. In particular, we stop to increase n when adding further trees

does not bring improvement to the model.

To evaluate the goodness of random forest fit, we can use the area under ROC curve and the confusion matrix already explained in Section 2.3.1. Another important aspect of the random forest is the importance that every variable has in the model. The simplest variable importance measure to use in tree-based ensemble methods is to merely count the number of times each predictor is selected, but more elaborate and reliable measures are available. An example for classification trees is the *Gini importance*, also called *Mean Decrease in Impurity* because it defines the total decrease in node impurity. The Gini impurity at node τ in a binary tree is $i(\tau)$, an efficient approximation of the entropy measuring how well a potential split separates the samples of the two classes in this particular node.

Knowing that $p_k = \frac{n_k}{n}$ is the fraction of the n_k samples from $k = \{0, 1\}$ on the total n samples, the impurity of the node τ is:

$$i(\tau) = 1 - p_1^2 - p_0^2$$

Its decrease Δi that results from splitting and sending the samples to two sub-nodes τ_l and τ_r is

$$\Delta i(\tau) = i(\tau) - p_l i(\tau_l) - p_r i(\tau_r)$$

After an exhaustive search over all available variables at the considered node, the one that leads to the maximal Δi is determined. This decrease in Gini impurity is recorded and accumulated for all nodes τ and all trees in the forest individually for all variables θ. A new quantity, called Gini importance is determined:

$$I_G(\theta) = \sum_T \sum_\tau \Delta i_\theta(\tau, T)$$

It indicates how often a particular feature θ has been selected for a split and how large its value was discriminating for the classification problem under study [14].

The correspondent measure for regression random forest is RSS that records the total decrease in node impurity from splitting on the considered variable, averaged over all trees.

The last presented variable importance measure for random forest algorithm in case of regression is based on the idea that the prediction accuracy before and after permuting a single variable represents the association between this predictor and the response. When the model builds a tree, some observations from the original dataset are not chosen in the bootstrapping process, such observations are called out-of-bag sample (OOB). For each tree, the *Mean Squared Error* (MSE) of the prediction is computed on the OOB portion of the data. Then, the same measure is recorded after permuting each predictor variable. More the accuracy of the model decreases using a permuted variable more the original predictor X_j is associated with the result. For this reason, Breiman [3] suggests as measure of X_j importance the difference in MSE before and after permuting it. This difference is then averaged over all trees of the forest and sometimes is scaled dividing by the standard deviation of the variation. One of the advantage of the permutation accuracy importance is to cover the impact of each predictor variable individually even predictors are correlated each other.

Chapter 3

Analysis of a volleyball dataset

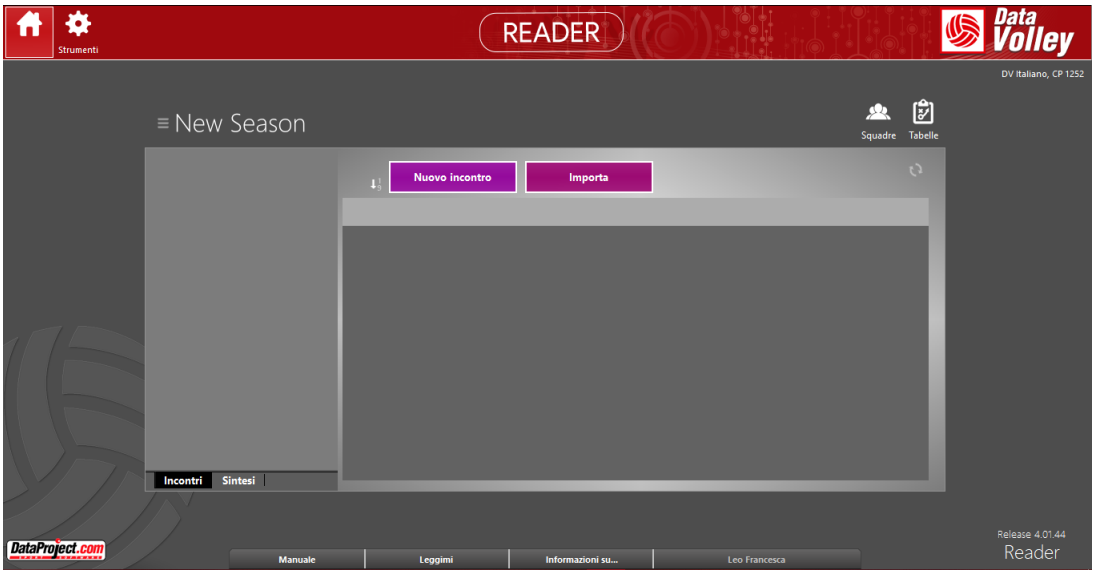
"Volleyball is a sport where you are always looking up"

Anime Haikyu, 2014

In this chapter we will be able to observe the application of the previously described methods in a whole data set from the world of volleyball. By an empirical examination, we hope to make more understandable the aim of the techniques seen above. To prevent the data from being scattered or incomplete, we chose not to analyze an entire season of a championship. We collected data from a different league competed in about two months in which sixteen National male teams played one against each other: the Volleyball Nations League 2018.

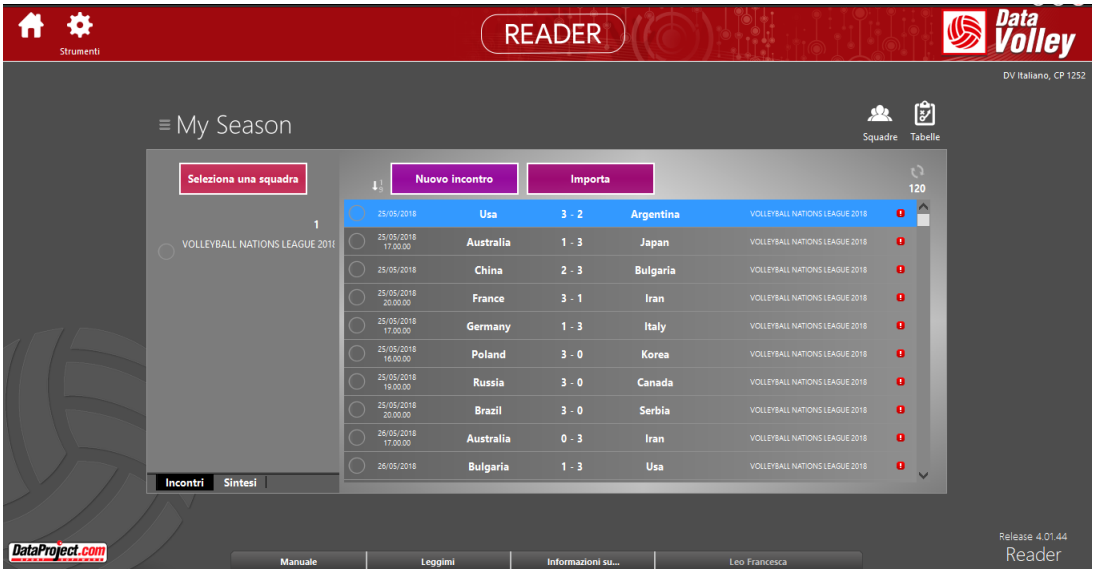
Each team has a reference figure called scout-man who is in charge of recording the performances of the athletes, that is the evaluation of every technical gesture. To realize the size of their work is enough to think that in every action there are on average six technical gestures and that each match is composed by about 150 actions. During the Nations League 2018 the scout-men used a very comfortable software called DataVolley. Its homepage is represented below.

Figure 3.1: DataVolley Homepage



The space in the center is filled by the matches that we want to analyze.

Figure 3.2: DataVolley Homepage



In DataVolley it is possible to record every single gesture and also its upshot thanks to different symbols, i.e. a scored point corresponds to #, while an error is identified by =. An example of its operation is in Figure 3.3. Since everything is codified, it would be

necessary a legend and a detailed explanation overall for statisticians that do not know this sport. So, to better manipulate the data we have moved all them on the software R. To demonstrate the interest around this topic, in R exists a package that reads the files output by DataVolley and in the next lines we write the instructions to use it:

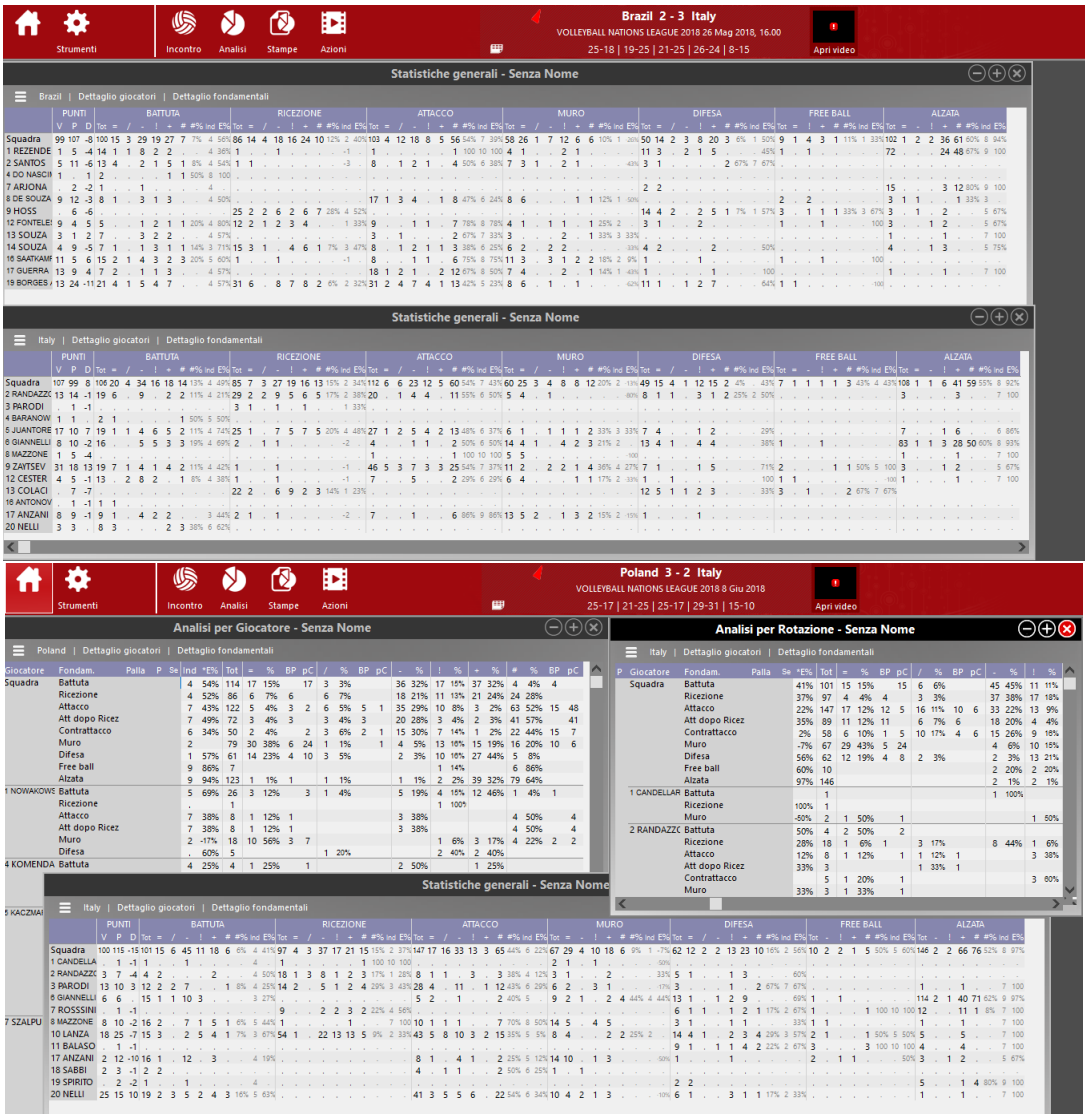
```
library(devtools)
install_github("raymondben/datavolley")
library(datavolley)
```

Moreover, R has a lot of statistical available packages and thanks to it we can always apply the most appropriate and necessary algorithm.
The total number of matches can be obtained from a simple combination:

$$\binom{16}{2} = 120$$

Now, every single technical gesture is assessed into one row of the data set and for each one several characteristics are annotated into the columns, i.e. the starting and the arrival zone of the ball. At the end, the full size of the data set is 156.175 rows and 79 columns. In our analysis we are committed to remain faithful to the assertion that the team that wins corresponds to the one that scores more points, for this reason we decided to split the data set into the 446 sets that compose it. We know that if we are considering an entire match it could happen that the total number of points done by the loser team is greater than the winner team number of points, i.e. a match that ends 1-3 with this kind of partial results 25-15, 23-25, 23-25, 23-25.

Figure 3.3: DataVolley Record



Once all the data has been recorded we have to define which variables are relevant to the development of our work. We relied on the experts' opinion and previously published articles about volleyball as [9], [21]. Therefore, for each set, we have calculated:

1. **Percentage of Serve Errors:** # of serve Errors / # of Total serves

2. **Percentage of Serve Aces:** # of Serve Direct Points / # of Total Serves

3. **Percentage of Reception Errors:** # of Reception Errors / # of Total Receptions
4. **Percentage of Positive Receptions:** # of Positive Receptions / # of Total Receptions
5. **Percentage of Perfect Receptions:** # of Perfect Receptions / # of Total Receptions
6. **Percentage of Attack Errors:** # of Attack Errors / # of Total Attacks
7. **Percentage of Blocked Attacks:** # of Blocked Attacks / # of Total Attacks
8. **Percentage of Winning Attacks:** # of Winning Attacks / # of Total Attacks
9. **Percentage of Winning Blocks:** # of Winning Blocks / # of Total Blocks

3.1 Descriptive Analysis

Before we start to perform any analysis, to make a plot, one for each of the nine variables, it can help us to detect the presence of real outliers or any possible error recording the data. For example, the percentage of serve errors for each of the 446 sets is represented in 3.4, and this shows that the values are evenly distributed in an area in which happens very rarely that they are close to zero or bigger than 30%. The same occurs with the other eight variables even if within a different range of values, Figure 3.5. Because of no presence of values out of range, we do not need to remove any outlier.

To have a full and a more general vision about the distribution and the range of definition of each of the nine variables we can use the function `boxplot` in R and observe its result in 3.6. A boxplot is very useful to graphically display a batch of data distribution when we need some information about its variability and dispersion. It represents a rectangular box in which the vertical axis has the scale of collected data. The rectangle is divided by an horizontal line: the median. Its top and bottom

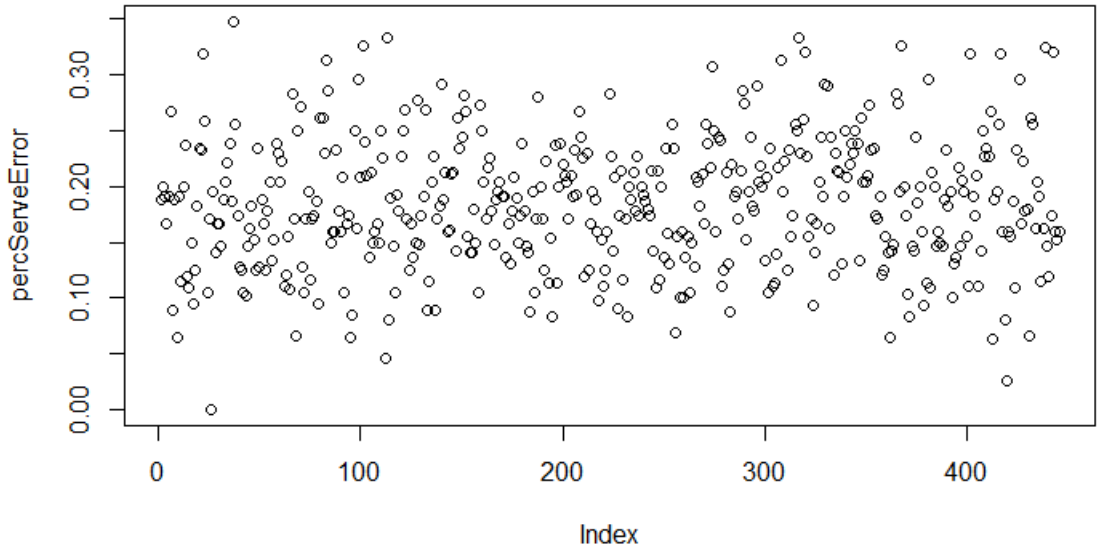


Figure 3.4: Values of the percentage for the 1^{st} variable

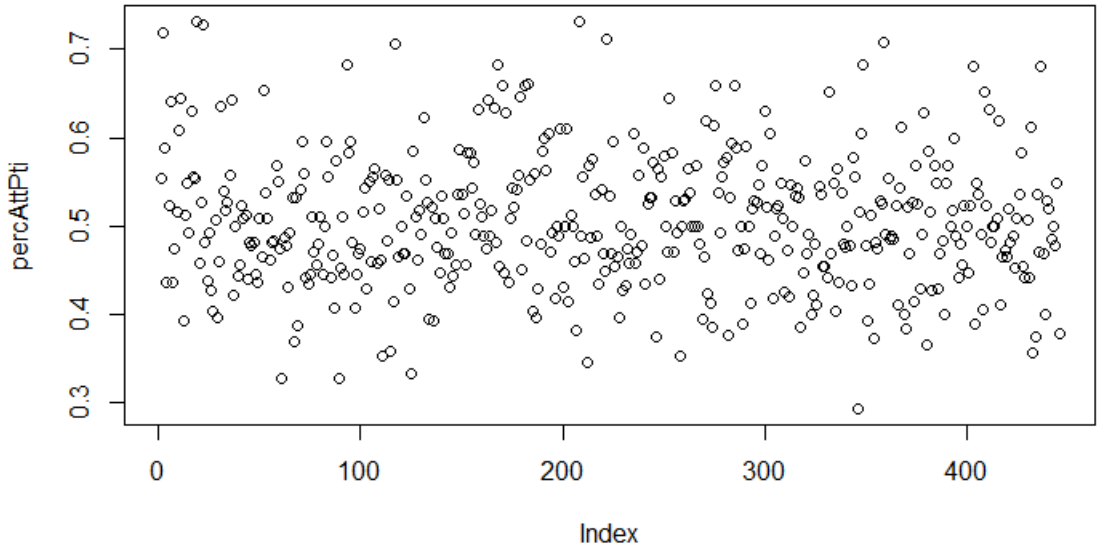


Figure 3.5: Values of the percentage for the 8^{th} variable

correspond to the upper and lower quartiles of the batch, for this reason the extreme lines of the box define the Interquartile Range (IQR). Usually, softwares define a step equal to 1.5 times the interquartile range and a vertical line is extended from the top of the box to the largest observation within a step from the top. A similarly defined line extends from the bottom of the box to the smallest observation within a step from the bottom [2]. The circles in the figure represent the observations more distant from the boxes than the described lines, and they take the name of outliers. Using the boxplot we have a summary about location, spread and symmetry of the batch data. Concerning the location of data, it is evident that the percentage of attack points (variable (8)) is characterized by higher values respect to all other variables. Moreover, the shape of the rectangle gives us instructions about the spread, and consequently about the variance of the data as the distance between the end of the whiskers and the range does. For example, the variables (2), (6) and (7), should have a very low variance, meaning that the distribution of the 446 data in these variables is less sparse. Finally, we know that under and over the median there is the same number of data, in fact in a Gaussian distribution it is perfectly centered in the box: so the median and its centering are an important information related to the symmetry of the data. We can notice it also comparing the length of the upper whisker with the length of the lower one, and the number of individual observations displayed on each side. This could be a first informal and visual step to observe if our variables can be considered normally distributed, since we use their normality in a next test we will prove it in a more formal and mathematical way using the Kolmogorov-Smirnov's test in the next section.

Until now we have just observed the variables, but we can also give an exact value to mean and variance we have talked about, using default R functions. They confirm everything that we have already noticed into the boxplot, and we can see the values in the figure 3.7. Moreover we can plot the frequency histogram for the variables and observe its shape.

If we compare two histograms related to consecutive gestures of an action, we probably expect the values to be divided into two similar ranges. This occurs with the percentage

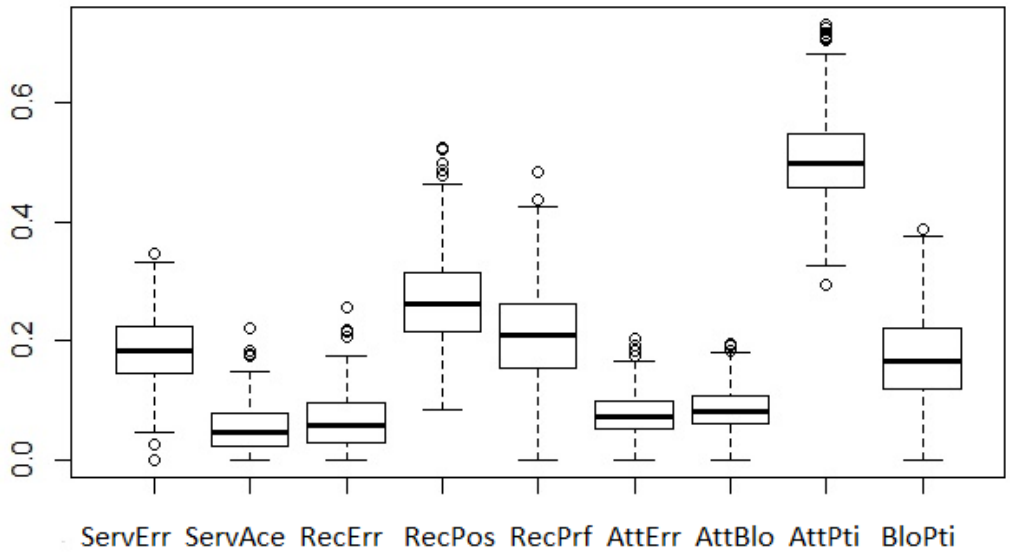


Figure 3.6: Nine variables boxplot

	Mean	Standard Deviation
percServeError	0.18406483	0.05864604
percServeAce	0.05464210	0.03666603
percRecError	0.06720590	0.04505738
percRecPos	0.26784095	0.08050431
percRecPrf	0.21064550	0.08240160
percAttError	0.07574511	0.03600434
percAttBlo	0.08388616	0.03737144
percAttPti	0.50518112	0.07526476
percBloPti	0.17141321	0.07192415

Figure 3.7: Mean and Standard Deviation for the nine variables

of serve aces and reception errors one, Figure 3.8, giving us some clues about their correlation. At the contrary, the attack points percentage lives in a very different range compared to the positive receptions one, indicating that it is very likely, in this competition, to score attack points even if the previous passing is not precise, Figure 3.9. From these two comparisons we have a second important clue about the normality of the variables, observing that the variables (4) and (8) are much more like a Gaussian distribution than the variables (2) and (3).

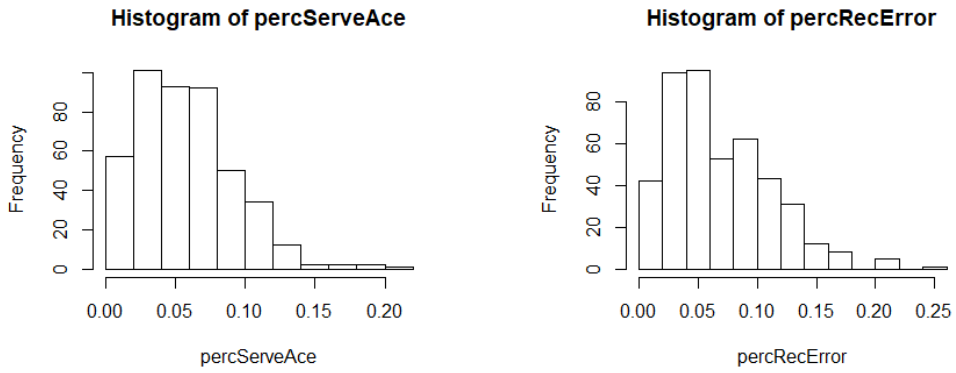


Figure 3.8: Histograms with similar range of values

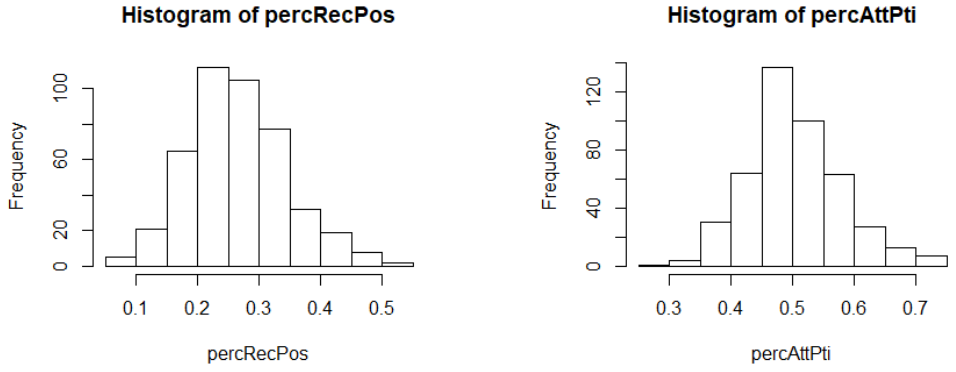


Figure 3.9: Histograms with no similar range of values

3.2 Correlation

In this section we will apply some methods to deepen the correlation subject that we have previously mentioned. First of all, as described in Section 2.1, we have applied Pearson's correlation formula that is useful to identify the dependence level between all the variable pairs. Obviously, the resulting matrix is symmetric as we can see in Table 3.1.

Table 3.1: Correlation matrix

	ServeE	ServeA	RecE	RecPs	RecPf	AttE	AttB	AttP	BloP
ServeE	1.00	-0.03	0.07	-0.03	-0.03	-0.02	0.02	-0.02	0.01
ServeA	-0.03	1.00	0.99	-0.16	-0.28	-0.05	0.11	-0.03	0.13
RecE	0.07	0.99	1.00	-0.17	-0.28	-0.05	0.11	-0.04	0.13
RecPs	-0.03	-0.16	-0.17	1.00	-0.34	0.02	-0.00	0.02	0.04
RecPf	-0.03	-0.28	-0.28	-0.34	1.00	0.03	-0.08	0.07	-0.10
AttE	-0.02	-0.05	-0.05	0.02	0.03	1.00	-0.12	-0.29	0.03
AttB	0.02	0.11	0.11	-0.00	-0.08	-0.12	1.00	-0.24	0.92
AttP	-0.02	-0.03	-0.04	0.02	0.07	-0.29	-0.24	1.00	-0.22
BloP	0.01	0.13	0.13	0.04	-0.10	0.03	0.92	-0.22	1.00

The values confirm our intuition whereby the serve direct points and the errors during the reception phase have a high correlation. In fact, their Pearson's correlation index is $\rho = 0.99$, that is the highest one excluding the principal diagonal of 1. Two other almost perfectly correlates variables are *percAttBlo* and *percBloPti*, in this case ρ is slightly lower because of the fact that for every blocked attack a block point is registered, but is not certain that every block point corresponds to a blocked attack:

some “conflict” balls near to the net are categorized like block points if a team succeeds in scoring thanks to them, even if no attack is made.

As predictable, the variable (1) is not strongly correlated to any other one in fact, there is nothing that precedes it because the serve is the gesture that starts every action of a match and there is nothing that follows it because it corresponds to a direct error. It is also interesting to notice the negative relationship between the variable (2) and the variables (4) and (5): when serve direct points increase, sure the number of positive/perfect receptions decreases.

It might also seem that the percentage of aces increasing is correlated to a percentage of attack errors decreasing, but this is a sort of “imaginary” dependence. It does not exist any direct link between a serve point and an attack error, because after a scored point the action ends. So, what does it happen? It is clear that a higher number of direct aces means less possibility for the opposite team to build a counterattack game action and then also to do an error. The same principle regulates also the value $\rho = -0.05$ referring to (3) and (6).

Finally, it is not surprising that *percBloPti* is positively correlated to *percServeAce* and consequently to *percRecError*. In the first case we observe $\rho = 0.13$, that could appear not very relevant but, overall in a large size data set, is sufficient to understand that when the serve level and difficulty are high, to stop the opposite team’s actions thanks to the block is easier.

We lead this part of the analysis not only to have a more global and full vision of our data set, but also because predictive methods like regression need the use of almost completely independent variables, so it is important to have a clear idea about how to detect and manage some kind of dependence among data. To confirm the intensity and the sign that the matrix 3.1 shows, and to add information concerning the correlation typology we can use another important tool: the scatter-plot.

Like the correlation matrix, also the scatter-plot is related to pairs of variables and observing the image 3.10 we can immediately guess its function, already explained in Section 2.1. Even if it was clear the presence of a strong dependence between serve

Figure 3.10: Scatter Plot

points and reception errors, as between blocked attack and block points, it could be a logarithmic or quadratic correlation and so on. Thanks to the scatter-plot and to red line that remarks the shape that the dots form, we are sure to fall in the linear case. This relation is often called collinearity and its presence can rise problems when the two concerned variables are used together in some predictive model since it can be difficult to separate their individual effects. In other word, since the serve aces and the reception errors tend to move together, like blocked attacks and block points do, it can be more complex to determine how each one individually is linked with the response [10]. As predictable from correlation matrix, into the scanner-plot we can observe also some negative trends: the most evident appears between the variable (5) and the variables (2) and (3). Surely, to increase the number of perfect reception indicates a good capability to contain the opposite serve and, consequently, a small percentage of aces.

Table 3.2: Bartlett's test for sphericity

	Results
chisq	3461.75
p.value	0.00
df	36.00

The last presented technique to detect correlation is the Bartlett's test for sphericity. It is very different from the previous methods because it is not used on a pair of variables but when we just want to discover if the correlation among more than two variables is significant. Using this instrument we do not detect the sign and the value of the correlation but it is very useful to understand if a large set of variables is exploitable in a prediction context.
We have already seen in Section 2.1 that it measures how much the correlation matrix

differs from the identity one, in fact if the nine variables were completely independent we would observe a correlation matrix with nine 1s on the principal diagonal while each other space would be filled by 0s. Clearly, it does not occur with our data, as it is evident in the table 3.2. The Chi-square test is performed on 36 degrees of freedom and the Chi-Square value 3461.75 depends on the determinant of the correlation matrix and on the number of observations and variables. We cannot accept the null hypothesis of independence among our variables, or better, at least two variable are so strongly correlated that the p-value is considered null. A great deal must be considered about the sensitivity of the Bartlett's test concerning the non normality of data, so it is possible that the rejecting of the null hypothesis is more probable when our variables are not normally distributed. As promised in Section 3.1, now we provide the results of the Kolmogorov-Smirnov's test for each variable in 3.3.

Table 3.3: Kolmogorov-Smirnov's test for normality

Variable	p-value
percServeError	0.722
percServeAce	1.823e-05
percRecError	0.0011
percRecPos	0.1765
percRecPrf	0.525
percAttError	0.2619
percAttBlo	0.2799
percAttPti	0.1149
percBloPti	0.4547

The percentage of aces and the one of reception errors do not satisfy the null hypothesis of the test and cannot be considered like gaussian distribution variables. They could affect the Bartlett's test but it suffices to generate a subset that contains only the seven normally distributed variables. Moreover, the two removed variables have a Pearson's coefficient $\rho = 0.99221786$ so we are already sure about the presence of correlation between them. Now we apply again the test and we can keep on rejecting the independence, because the p-value is $3.92e-208$. The new result in Tab 3.4 assures us that, even if the first test could prove a little pessimistic vision, it was working good. Choosing significant and possibly unrelated subsets of variables will be relevant in the next part of the work.

Table 3.4: Bartlett's test for a subset of variables

	Results
chisq	1046.235
p.value	3.91978e-208
df	21

3.3 The dataset used for predictive analysis

To approach the next part, we need a different data set obtained from the previous one. In particular, since the aim of this work is to predict the probability of winning or losing every singular set basing on the performance in technical gestures, we split into two part the total percentage of each variable. We had the total values of variables during each set that do not take in account the result, but now we need for each

variable a value that is the percentage of the winning team and a second one with the percentage of the loser team. Moreover, we add another binary variable, the response of our future analysis and regressions, that is 1 if the line is referred to the winners' team percentages, 0 if we are considering the loser team. Clearly, the number of rows in this new data set is $446 \times 2 = 892$, because in two different lines we are referring to the same set. A better view of the considered dataset is in the Table 3.5.

Table 3.5: Dataset

	Set	Ris	ServE	ServA	RecE	RecPs	RecPf	AttE	AttB	AttP	BloP	code_result	score
1	1	winner	0.20	0.12	0.05	0.05	0.32	0.00	0.00	0.56	0.17	1.00	25
2	1	loser	0.17	0.04	0.15	0.35	0.15	0.03	0.10	0.55	0.00	0.00	23
3	2	winner	0.21	0.08	0.00	0.35	0.06	0.05	0.00	0.70	0.43	1.00	25
4	2	loser	0.19	0.00	0.11	0.26	0.42	0.05	0.16	0.74	0.00	0.00	20
5	3	winner	0.16	0.00	0.00	0.29	0.24	0.05	0.00	0.64	0.07	1.00	25
6	3	loser	0.23	0.00	0.00	0.24	0.19	0.17	0.03	0.55	0.00	0.00	22
7	4	winner	0.24	0.16	0.05	0.33	0.10	0.10	0.03	0.42	0.27	1.00	25
8	4	loser	0.09	0.04	0.21	0.11	0.11	0.00	0.17	0.46	0.08	0.00	23
9	5	winner	0.25	0.04	0.06	0.12	0.44	0.05	0.00	0.70	0.42	1.00	25
10	5	loser	0.11	0.06	0.06	0.11	0.06	0.12	0.21	0.38	0.00	0.00	17
11	6	winner	0.31	0.00	0.00	0.24	0.19	0.11	0.07	0.68	0.23	1.00	29
12	6	loser	0.22	0.00	0.00	0.30	0.15	0.05	0.14	0.59	0.17	0.00	27
13	...	...	...	...	...	...	...	...	...	...	...	...	...
14	...	...	...	...	...	...	...	...	...	...	...	...	...
891	446	winner	0.20	0.00	0.00	0.22	0.22	0.05	0.00	0.40	0.11	1.00	15
892	446	loser	0.10	0.00	0.00	0.25	0.08	0.18	0.06	0.35	0.00	0.00	10

There is a significant difference between the two datasets: in the first one we are analyzing a whole portion of a match, joining the values of the losing team and the winning one. Now, it is as if in every line there is a different team, that is not completely true because for example we find the Italian team minimum three times for fifteen matches, but we consider it like different teams: the "First Italy" is Italy in the first set against Argentina, the second is in the second set and then we have another Italy against Brazil and so on. This is the reason why if we repeat all the steps previously performed we will get different results.

First of all we consider the correlation and we notice immediately how the percentage

of serve aces and the one of passing errors are no longer related. This is obvious because we are splitting the performances of the two teams involved in a set, and the same team registers better or worse values to the serve than the reception ones independently from each other. What actually happens is that a percentage represented in a row influences another one of the row below, this also is an aspect that we will consider in the future chapters, but the Pearson's correlation that is calculated in Table 3.6 works only on values of the same line.

Table 3.6: New correlation matrix

	x2.ServeE	x2.ServeA	x2.RecE	x2.RecPs	x2.RecPf	x2.AttE	x2.AttB	x2.AttP	x2.BloP
x2.ServeE	1.00	-0.08	0.05	-0.01	-0.00	0.03	0.07	-0.05	-0.07
x2.ServeA	-0.08	1.00	-0.03	0.02	-0.05	-0.04	-0.02	0.04	0.10
x2.RecE	0.05	-0.03	1.00	-0.19	-0.22	-0.01	0.11	-0.10	0.00
x2.RecPs	-0.01	0.02	-0.19	1.00	-0.30	0.02	-0.02	0.06	0.02
x2.RecPf	-0.00	-0.05	-0.22	-0.30	1.00	-0.00	-0.09	0.12	-0.02
x2.AttE	0.03	-0.04	-0.01	0.02	-0.00	1.00	-0.06	-0.28	-0.02
x2.AttB	0.07	-0.02	0.11	-0.02	-0.09	-0.06	1.00	-0.34	-0.19
x2.AttP	-0.05	0.04	-0.10	0.06	0.12	-0.28	-0.34	1.00	0.08
x2.BloP	-0.07	0.10	0.00	0.02	-0.02	-0.02	-0.19	0.08	1.00

Other values of the matrix however have grown by highlighting what are the factors that feed each other when we consider a single team. Now, a perfect pass is more incisor to score a point in the attack phase, $\rho = 0.12$, and to perform blocked attacks negatively affects the possibility to be effective, in fact the correlation coefficient between the variables (7) and (8) is $\rho = -0.34$. There is also a strong negative correlation between perfect and positive passing and among them and the pass errors. A team that on the total of receptions has a higher number of perfect passes will have for sure fewer positive or wrong receptions. The scatter plot in Figure 3.11 shows that we lost the previous almost perfect collinearity as explained above, but also the normal distribution

Figure 3.11: New scatter Plot

of the variables. In the main diagonal in fact we see the frequency histograms for each variable and none of these has the Gaussian shape, as confirmed by the KS-test. For this reason the Bartlett's test cannot be implemented and we will rely on the Pearson's correlation test, by matching the variables, to choose a subset of variables that we can use as predictors.

Chapter 4

Features selection for logistic regression

"*It is our choices, Harry,*
that show what we truly are,
far more than our abilities."

J.K.Rowling , 1998 d.C.

The title of this chapter may be ambiguous as it is often the regression the instrument used to implement a feature selection. In this case the purpose is to find a subset or a linear combination of variables that makes the regression model as accurate and explanatory as possible. *"Unlike discriminant function analysis logistic regression does not assume that predictor variables are distributed as a multivariate normal distribution with equal covariance matrix"* [16] and this is appropriate for us since our data do not present any normal distribution. Instead, this model assumes that the binomial distribution describes the distribution of the errors that equal the actual result minus the predicted one. The binomial distribution is also the assumed distribution for the conditional mean of the outcome. The binomial assumption may be taken for granted as long as the sample is random, that means independent observations from each other. To assure this aspect we created a random mechanism that chooses just one line of the sets: a random number generator emits 446 uniforms between zero and one, if the generated number is higher than 0.5 is automatically selected the line related to the winning team, at the contrary, for every number smaller than 0.5 the losers' line is

chosen. Thanks to this method we are sure that each dichotomous result 1 or 0 is not influenced by another one. It is as if we were stating: "Thanks to these performance percentages you win or lose the set regardless of the percentage of the opposing team".

4.1 Independent predictors

Once we have solved the problem of the related observations we will deal the same aspect for the predictors of the model. As we have already said, the predictors of a regression model should not be highly correlated to each other. In this case in fact, it would be difficult to separate the individual effects of variables on the outcome. Looking at the data we have already ruled out the problem of collinearity which would significantly reduce the accuracy of the estimates of the regression coefficients. For this reason in this section we are projected to select the allowed subset of variables avoiding the correlated ones, that are the variables that do not pass the Pearson's correlation test. We have collected in a table the p-values obtained from the application of the test, remembering that even a value that in Table 3.6 may not seem very high can be relevant given the size of the dataset. As already explained in Section 2.1 a p-value > 0.05 allows us to accept the null hypothesis that is to consider equal to zero the coefficient of correlation between the two variables in question.
We can observe the Tab 4.1 and immediately note that the variable that indicates the passing errors percentage can not be used like predictor joined to the other variables related to the reception phase. Moreover it is better to avoid the combined use of this variable with the one related to the points in attack and to the blocked attacks.

Table 4.1: P-values of Pearson's correlation test

	percServE	percServA	percRecE	percRecPs	percRecPf
percServeE	<2.2e-16	0.01641	0.1361	0.7296	0.984
percServA	0.01641	<2.2e-16	0.3748	0.4948	0.1489
percRecE	0.1361	0.3748	<2.2e-16	1.581e-08	5.227e-11
percRecPs	0.7296	0.4948	1.581e-08	<2.2e-16	<2.2e-16
PercRecPf	0.984	0.1489	5.227e-11	<2.2e-16	<2.2e-16
PercAttE	0.3681	0.1818	0.8244	0.6291	0.9464
percAttB	0.0439	0.5641	0.001189	0.5843	0.005727
percAttP	0.1137	0.1999	0.004162	0.06719	0.00033
percBloP	0.05092	0.003785	0.8924	0.5684	0.5185
	percAttE	percAttB	percAttP	percBloP	
percServeE	0.3681	0.0439	0.1137	0.05092	
percServA	0.1818	0.5641	0.1999	0.003785	
percRecE	0.8244	0.001189	0.004162	0.8924	
percRecPs	0.6291	0.5843	0.06719	0.5684	
PercRecPf	0.9464	0.005727	0.00033	0.5185	
PercAttE	<2.2e-16	0.05498	<2.2e-16	0.5753	
percAttB	0.05498	<2.2e-16	<2.2e-16	1.268e-08	
percAttP	<2.2e-16	<2.2e-16	<2.2e-16	0.01242	
percBloP	0.5753	1.268e-08	0.01242	<2.2e-16	

A first subset that we can consider within our linear regression model is *percServE, percRecE, percAttE, percBloP*, in fact thanks to the table we confirm that all the pairs that can be formed have a p-value higher than 0.05.

To fully analyze the offensive phase of a team now we want to insert in the model the variable *percAttP*, but it is strongly related with many other variables, so the only "correct" subset that we can form is *percServA, percRecPs, percAttP*, possibly alternating the two variables concerning the serve phase that between them are correlated but, as

predictable, they are not with any other. To consider at least once each of our nine variables we will now exhibit two other possible subsets and we will then analyze their actual usefulness in the next chapter: *percServA, percRecPs, percAttE, percAttB* and *percServE, percRecPf, percAttE, percBloP*.

There is a last check, as we have already explained, to be carried out on the variables to ensure that the multicollinearity is excluded, and it is the VIF value. Since the values in the Table 4.2 are broadly lower than 5 it is not necessary to worry further about this aspect. Each of the above subsets of variables will be used and analysed in Section 5.1.

Table 4.2: VIF

	Variables	VIF
1	x2.percServE	1.02
2	x2.percServA	1.02
3	x2.percRecE	1.15
4	x2.percRecPs	1.20
5	x2.percRecPf	1.23
6	x2.percAttE	1.13
7	x2.percAttB	1.21
8	x2.percAttP	1.29
9	x2.percBloP	1.05

4.2 Application of Principal Component Analysis

What we have seen in the previous section could be avoided or supported by a technique called Principal Component Analysis (PCA). As explained in Section 2.2 it is an unsupervised learning algorithm because it is not focused on predict some results but on linearly combining our variables to obtain less and completely uncorrelated ones that explain the most part of their variance.

Table 4.3: Summary of PCA

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9
Prop. of Variance	0.2445	0.2031	0.1691	0.1266	0.0987	0.0482	0.0442	0.0359	0.0296
Cumulative Prop.	0.2445	0.4477	0.6168	0.7434	0.8421	0.8903	0.9345	0.9704	1.0000

We have 892 observation of our 9 variables, so we will obtain nine principal components, the first information that they give us is the proportion of variance that each of them explains and consequently the cumulative variance that we get using more than one component. In the Table 4.3 we observe that the first component, as normal, has the most explanatory power and that from the sixth they add only a 4% of the total information. This drop is directly related with the cumulative variance, in fact we get almost the 90% of the total knowledge thanks to the first six components.
The same concept we are explaining joined with the table it is shown in the Figure 4.1 and 4.2. The first plot is the useful one to help us in choosing the best number of variables to use in the next supervised learning techniques. We are carrying out our analysis without considering a huge number of variables, for this reason we could also keep on using all the nine variables, so we do not lose any amount of information, but knowing and appreciating that the new ones are completely independent thanks to PCA application. If instead, by computational necessity or desire, we want to reduce the number of variables to be used in our work, it is quite evident that the elbow of the function in 4.1 is at the sixth component.

Actually, with the application of this pre-processing analysis we obtain the coefficients of every component, each of tese vectors is called *loading* in Section 2.2. They give us the important information about the rotation that is applied on every component but they must be linearly combined to really obtain the new variables. First of all we can start observing and interpreting the loadings because they are the correlation between a component and a variable. Thanks to their sign and value we can estimate the knowledge that every component shares. To have a visual idea of it, we can construct the circle of correlations, that is a unit circle (because of the fact that the sum of the squared

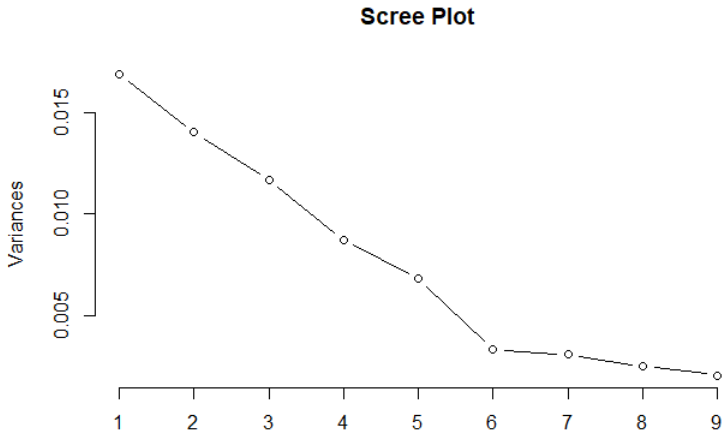


Figure 4.1: Scree Plot, Explained Proportion Value

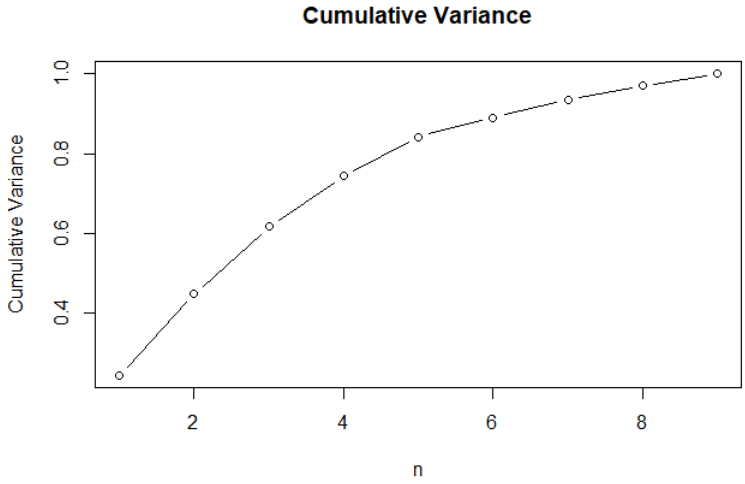


Figure 4.2: Explained Cumulative Variance

loadings for a variable is equal to one) with the variables inside. The coordinates of each variables are the loadings on the principal components, if a variable is perfectly explained by only two components it lays on the circle. In our case all the variables are inside the circle because we need nine components to represent them. *"The closer a variable is to the circle of correlations, the better we can reconstruct this variable from the first two components."* [1]. As we can see in Figure 4.3, the positive and the perfect receptions have the highest coefficients in the first component in comparison to the other variables, the attack and the block points in the second one. The closer to the center a variable is, the less important it is for the first two components. Even if we cannot plot a nine-dimensional plan with the same function that the circle has, we can take note of every coefficients in a Table (4.4). In this way we notice how each variable impact in each component: for instance, the third component explains a relevant quantity of variance for the last variable, in fact its coefficient is -0.71, while the fifth component is almost completely represented by the first variable, the errors in the serve phase.

Table 4.4: Loadings of PCA

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9
ServeErrors	-0.00	-0.12	0.02	-0.02	0.99	-0.07	0.04	-0.06	0.02
ServeAce	0.02	0.05	-0.03	0.02	-0.06	-0.04	0.24	-0.96	-0.05
RecError	0.03	-0.10	-0.16	0.27	0.06	0.72	-0.58	-0.17	0.06
RecPos	0.67	0.23	0.39	-0.55	0.03	0.19	-0.11	-0.03	0.03
RecPrf	-0.73	0.13	0.22	-0.58	0.01	0.22	-0.09	-0.06	0.02
AttErr	0.02	-0.09	-0.07	-0.13	-0.00	-0.33	-0.50	-0.08	-0.78
AttBlo	0.04	-0.22	-0.05	-0.05	0.01	0.52	0.58	0.15	-0.57
AttPti	-0.12	0.64	0.51	0.49	0.09	0.06	0.02	0.04	-0.25
BloPti	0.05	0.66	-0.71	-0.17	0.10	0.06	0.06	0.06	-0.05

Now that we have all the loadings we can create the new variables, it is a simple

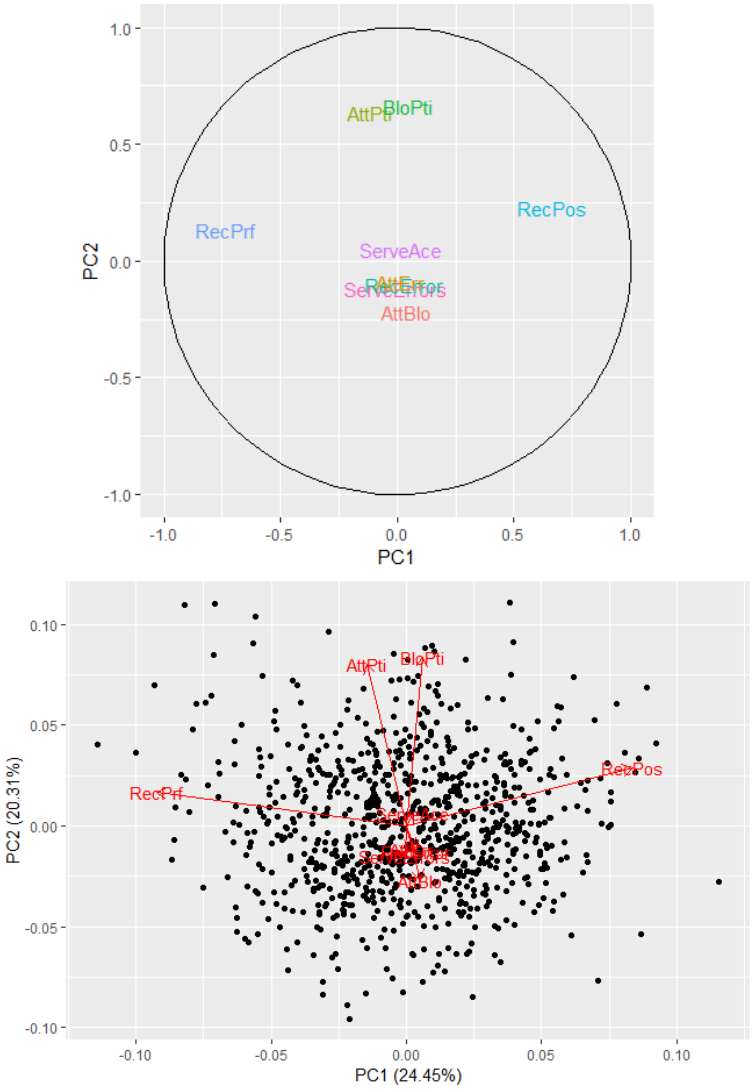


Figure 4.3: Visual representation of PCA

matrix product between the observations matrix (892x9) and the loadings matrix (9x9). We obtain 892 observations of nine variables, before using them we can perform some test to study their distribution and to confirm their total uncorrelation. In particular, only the $8th$ variable does not satisfy the Kolmogorov-Smirnov test of normality, so we can use the Bartlett's Correlation Test and get the result below:

chisq	-0.00
p.value	1.00
df	36.00

This is just a confirmation of the good operation of the Principal Component Analysis that provides us a set of independent variables that obviously excludes also the presence of multi-collinearity. The VIF computing is shown in Table 4.5 and it presents the smallest possible value 1 for each variable.

Table 4.5: Variance Inflation Function for the new variables

	Variables	VIF
1	PC1	1.00
2	PC2	1.00
3	PC3	1.00
4	PC4	1.00
5	PC5	1.00
6	PC6	1.00
7	PC7	1.00
8	PC8	1.00
9	PC9	1.00

Chapter 5

Predictive methods

"*In God we trust, all others bring data.*"

W. Edwards Deming, 1964

Predictive analysis is the use of historical data to provide a best assessment of what will happen in the future or identify future outcomes. There are a lot of statistical algorithm and machine learning techniques to approach this problem, more or less suitable depending on initial data. As we have explained above, the purpose of this work is to predict a dichotomous result, winner or loser, starting from a set of variables. Because of the fact that they can be considered continuous variables but not normally distributed the two most appropriate methods that we have implemented are the Logistic Regression and the Random Forest technique. We have already introduced their theoretical aspects in Sections 2.3 and 2.4, in the next part we will show the results of their application and we will comment the most interesting ones.

5.1 Logistic regression

To focus on our dataset we can see again the Table 3.5, there is a perfect correspondence between the second column, in which the result is expressed like a qualitative variable, and the twelfth column that represents the same result but with a codified binary

variable. This numeric column is the predicted one in the logistic regression in which we assume that if the probability $p(code\_result = 1) > 0.5$ the model identifies this line like a winner one.

We split the dataset containing all the 446 sets in two parts: the 75% of the volley sets forms the training dataset, the remaining ones are the so called test dataset. To better generalize this process and to not obtain outcomes only related to this specific partition of data, we have run the model on 100 different partitions. Essentially, the training and test dataset will always contain different lines from the previous experiment. Then, to show the general outcome, we will average the coefficient of the model and its accuracy.

After this step, every time we have 334 volleyball sets useful to train the model and 112 to validate it. We have to consider that for each set there are two correspondent lines, one referred to winners and the other one referred to losers. In the test dataset this is not a problem because we only want to predict the final result using the found coefficients. Instead, to train the model it is not appropriate to consider both lines related to a single set of a match because they affect each other. As we have already explained in Section 4, a mechanism is created to randomise the sample, basing on a random variable $U(0, 1)$. If it generates a number $u \geq 0.5$ we take in account the winner line, at the contrary, we add the loser line in the training dataset if $u < 0.5$. At the end of this second step we exactly have 334 lines to train the model, one for each selected set, and 224 lines to validate it, two for each remained set.

The first subset of variables that we use to implement logistic regression is described in Section 4.1 and includes:

- Percentage of Serve Errors
- Percentage of Reception Errors
- Percentage of Attack Errors
- Percentage of Block Points

After the training of this first model and averaging the 100 obtained coefficients, we obtain the Table 5.1. It tells us that all the four variables are significant for the model, the p-values are always smaller than 0.05 and it is important to notice the sign of every coefficient. The minus related to variables concerning errors implies that they negatively influence the probability of winning while it grows up with the increasing of the block points. More in detail, each estimated coefficient is the expected change in the $log\text{-}odds$ of winning for a unit increase in the corresponding predictor variable holding the other predictors constant at certain value.

Table 5.1: Logistic Regression - Model 1

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	1.5410	0.4649	3.31	0.00278
ServeErrors	-5.2623	1.5965	-3.30	0.00076
RecError	-11.1503	2.3584	-4.73	2.27e-06
AttErr	-14.3047	2.7936	-5.12	1.63e-06
BloPti	7.4018	1.3823	5.35	1.13e-09

Thanks to the model we can compute the probability of winning, then the model itself will use this value to predict if one team will be winner or loser with a certain accuracy. If we suppose that exists a team with 20% of Serve Errors, 10% of Reception and Attack Errors and finally the 17% of Block Points we obtain:

$$p(win) = \frac{\exp\left(1.54 - 5.26 \times 0.2 - 11.15 \times 0.1 - 14.30 \times 0.1 + 7.40 \times 0.17\right)}{1 + \exp\left(1.54 - 5.26 \times 0.2 - 11.15 \times 0.1 - 14.30 \times 0.1 + 7.40 \times 0.17\right)}$$

and so

$$p(win) = 0.31$$

Since we do not have any reason to choose a threshold different from 0.5, the model should allocate this team as a loser one. Clearly, this probability does not take into account the skill of the opposite team or the delta between the two teams in the match. This aspect will be discussed in the next chapter. Anyway we are able to reach our

goal: detect which variable has the major relevance for the victory.
It could happen that quite small percentages of errors or large percentages of points lead to an erroneous classification, how many times it occurs? We can average the results and summarize the situation in the table called confusion matrix below:

Table 5.2: Confusion matrix - Model 1

	FALSE	TRUE
0	81	31
1	31	81

The accuracy of the model is

$$Accuracy = \frac{81 + 81}{81 + 31 + 31 + 81} = 0.7232$$

To have more information about our first fit, we also compute the area under the 100 ROC curves (Figure 5.1) as explained in Section 2.3.1, then we average them and obtain $AUC = 0.7897$. While these two indexes are objective and provide us an immediate measure of the model exactness, the $AIC = 368.5283$ is just useful to compare models with the same size and the same number of predictors, so now it does not add anything to our knowledge.
We can summarize other models in which the predictors are independent. The second one is represented by:

- Percentage of Serve Points
- Percentage of Positive Receptions
- Percentage of Attack Errors
- Percentage of Blocked Attacks

In this case the AIC is higher than the previous one: 382.4704, so the first model is preferred. Observing the Table 5.3 we can give it an explanation. One of the predictors

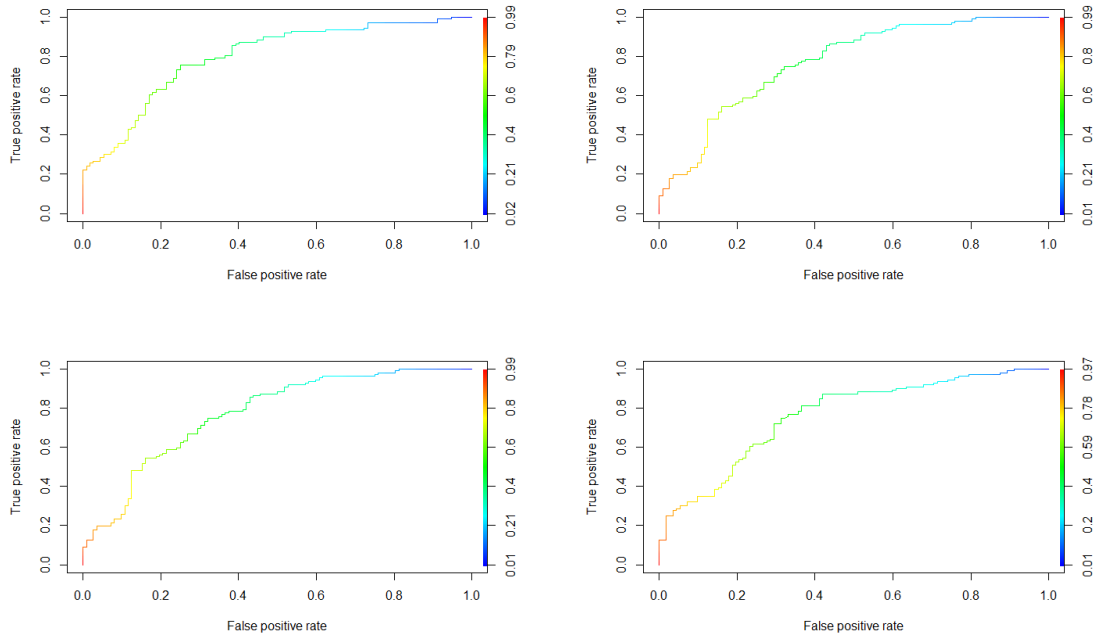


Figure 5.1: ROC curve - Model 1

is irrelevant in the study of the model, in fact the p-value of the positive receptions is 17.18%. In the Figure 5.2 and in the Table 5.4 it is confirmed that the second model fits

Table 5.3: Logistic Regression - Model 2

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	1.4257	0.4449	3.20	0.00301
ServeAce	14.2199	2.8745	4.95	5.43e-06
PositiveRec	1.6026	1.1955	1.34	0.17182
AttErr	-15.0853	2.6892	-5.61	1.98e-07
BloAtt	-18.0434	2.7692	-6.52	7.73e-10

our data in a worse way respect to the first one. The ROC curve is visually lower, in fact the underlying area is 0.7638 and the accuracy of the model is $0.6830 < 0.7232$.

Now the process is known, so we consecutively present other two subsets of variables that we use in a logistic regression model to predict the probability of winning.

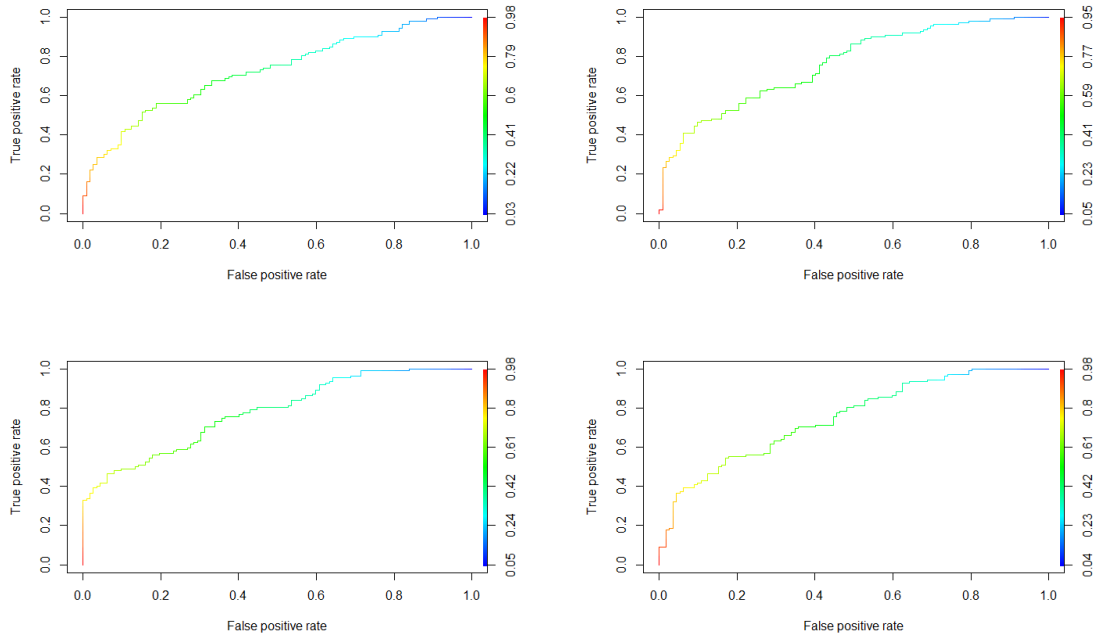


Figure 5.2: ROC curve - Model 2

Table 5.4: Confusion Matrix - Model 2

	FALSE	TRUE
0	78	34
1	37	75

As exposed in Section 4 one subset is:

- Percentage of Serve Points
- Percentage of Positive Receptions
- Percentage of Attack Points

Because of the fact that there are only three predictors the AIC index is not comparable with the previous values. Anyway, we have selected this subset to underline the importance of the points scored in the attack phase. In fact, in the Table 5.5 the third variable assumes the highest value of the z-statistic. Thanks to the p-value we observe

that, as in the second model, the percentage of positive receptions is not correlated with the result.

Table 5.5: Logistic Regression - Model 3

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-6.7055	0.9018	-7.44	1.04e-13
ServeAces	15.1095	2.8453	5.31	1.65e-06
PositiveRec	1.0759	1.1949	0.90	0.897
AttPti	12.9084	1.6087	8.02	5.24e-14

This third model is mainly described by the percentage of attack points and gets the highest accuracy, confirming the relevance of this variable in the analysis. The mean of the area under the curves in the Figure 5.3 is 0.8153, clearly this model represents curves that most approach the top-left corner of the graph. We can observe also the next confusion matrix and the goodness of fit of the model equal to:

$$Accuracy = \frac{87 + 88}{87 + 25 + 24 + 88} = 0.7813$$

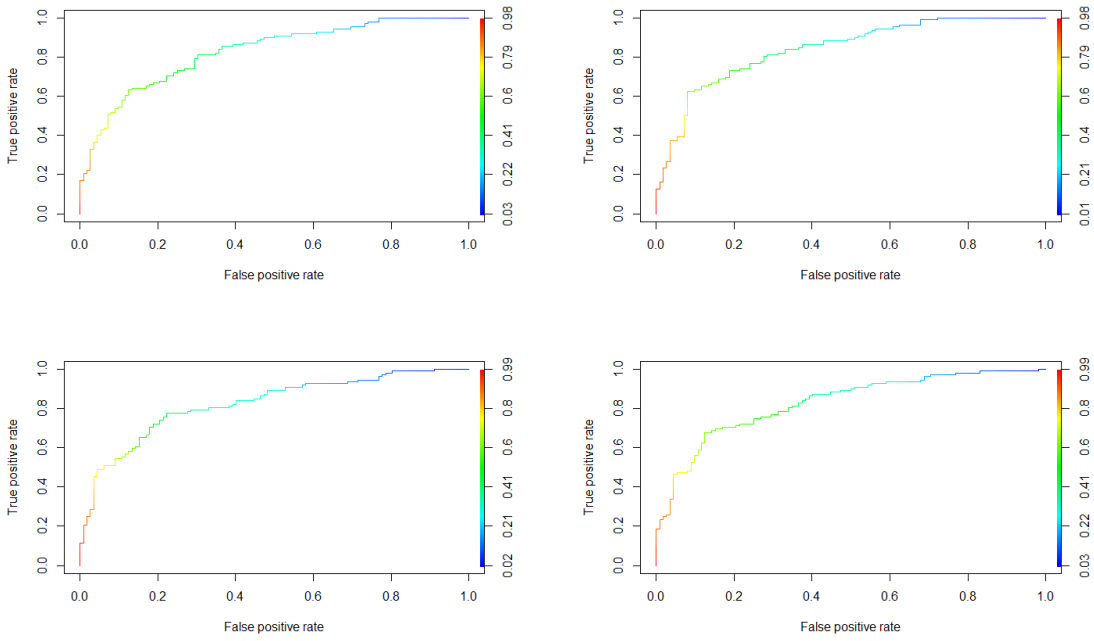
Table 5.6: Confusion Matrix - Model 3

	FALSE	TRUE
0	87	24
1	25	88

This matrix is almost symmetric, the results equal to 1 classified as losers are defined False Negatives (FN), while the real losers that the model allocates as winners are the False Positive (FP).
The last model of this part is composed by:

- Percentage of Serve Error

Figure 5.3: ROC curve - Model 3



- Percentage of Perfect Receptions
- Percentage of Attack Errors
- Percentage of Block Points

and the results are below.

Table 5.7: Logistic Regression - Model 4

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	0.1194	0.4523	0.26	0.9253
ServErr	-5.5380	1.5389	-3.60	0.0001
RecPrf	3.3671	1.1478	2.93	0.0066
AttErr	-13.3379	2.7091	-4.92	7.45e-06
BloPti	7.1772	1.2216	5.88	7.97e-08

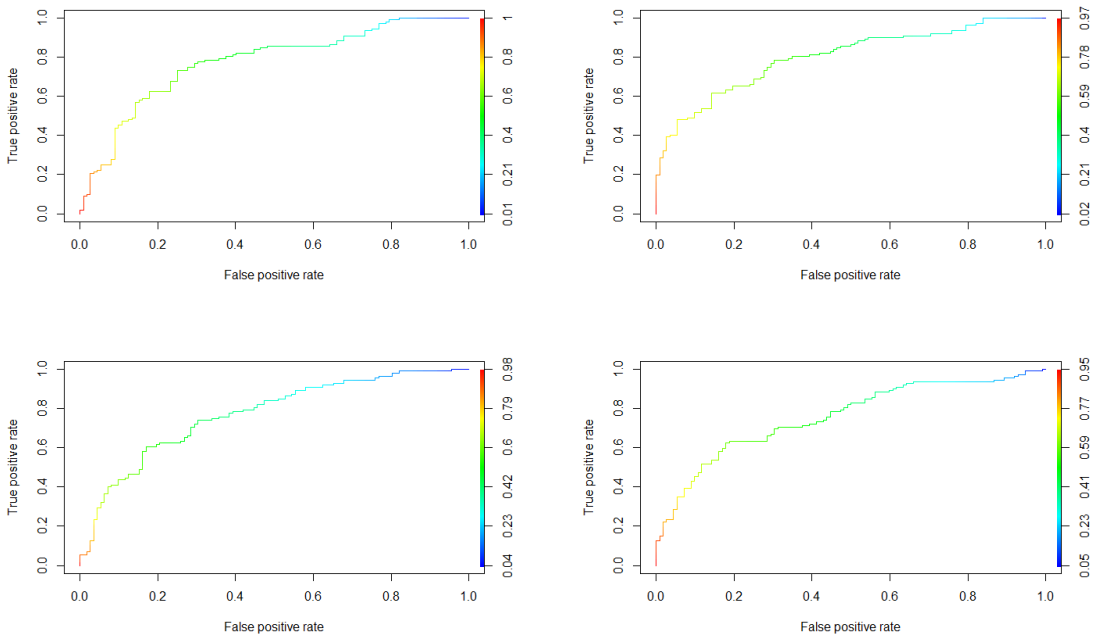
Table 5.8: Confusion Matrix - Model 4

	FALSE	TRUE
0	78	34
1	34	78

The average of the accuracy is 0.6964, while under the ROC curve in Figure 5.4 the area is 0.7679.

We never have obtained an accuracy higher than 80% so we would like to merge the information obtained by the four implemented models, but joining the variables could affect the model because of the loss of independence. The Principal Component Analysis (PCA) implemented in Section 4.2 is the technique that can help us in this situation. PCA is a pre-processing analysis so we will use it before of the logistic regression, we will find new independent variables and we will use them as predictors in our model. At the end of this section we will try to give an explanation to our new

Figure 5.4: ROC curve - Model 4



variables born as linear combination of every original features.

The computation of the new variables is not difficult. They are the matrix product between PCA coefficients, already presented in Table 4.4, and the values of the original nine variables. The result is a sort of new dataset in the Table 5.10.

We have already observed the elbow graph related to PCA analysis. Really, using six or nine variables as predictors does not change the computational effort. We can choose to not lose any percentage of variance using all the new created variables and observe which of them is more useful in the model.

Implementing this last model of logistic regression we greatly improve the accuracy of the prediction that becomes higher than 80%. Moreover we can do a sort of rank for the significance of the old variables even if they are not so explicit and understandable.

Table 5.9: Logistic Regression - Model 5

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-5.5073	1.4216	-3.87	0.0001
PC1	-0.7237	1.3069	-0.55	0.5797
PC2	21.5987	2.4996	8.64	<2e-16
PC3	4.1323	1.5606	2.65	0.0081
PC4	5.5712	1.7791	3.13	0.0017
PC5	-5.7028	1.9559	-2.92	0.0035
PC6	-10.5802	2.9863	-3.54	0.0004
PC7	12.1855	3.0857	3.95	7.9e-05
PC8	-17.4333	3.5993	-4.84	1.3e-06
PC9	12.9400	3.7821	3.42	0.0006

Once that the logistic regression has been applied 100 times to the training dataset (75% of the total), the values in Table 5.7 have been detected. We notice that to include the last components has been a correct idea. Observing the p-values, the seventh, the eighth and the ninth variable are significant. How can it happens that the first

component is the less relevant if it explains the 24% of the model variance while the ninth one only the 3% ?

The answer is to be found in the Table 4.4. First of all, PC2 is the only component with a perfect correspondence: positive gesture or points - plus, negative gesture or error - minus. Moreover in this column we observe the highest coefficients referred to attack points and block points that suggests us the relevance of these variables in the masculine volleyball. This thesis is confirmed by the fact that the first component is an-useful in our predictive model. Here, the two above mentioned variables have a low coefficient in absolute value and they reduce the effect each other because of the opposite sign. Moreover, the first component should include the most part of the model variance, but it gives high weights to the reception phase and this is not a winning aspect. Finally, the serve phase is almost completely explained by the PC5 and PC8, we can observe that in logistic model both these components present a negative coefficient. This is coherent with the fact that serve errors have a positive influence in the fifth component while aces are represented as negative in the eighth one.

What is the accuracy of this model that, in some way, includes all the nine variables? The Aikake Information Criterion 255.3084, not comparable with the previous values because of the nine predictors, but it is strictly lower. The ROC curve (Figure 5.5) is visibly closer to the top-left corner of the graph and underlines an area of 0.9088, almost 10% more than the third model and the biggest observed until now.

Figure 5.5: ROC curve - Model 5

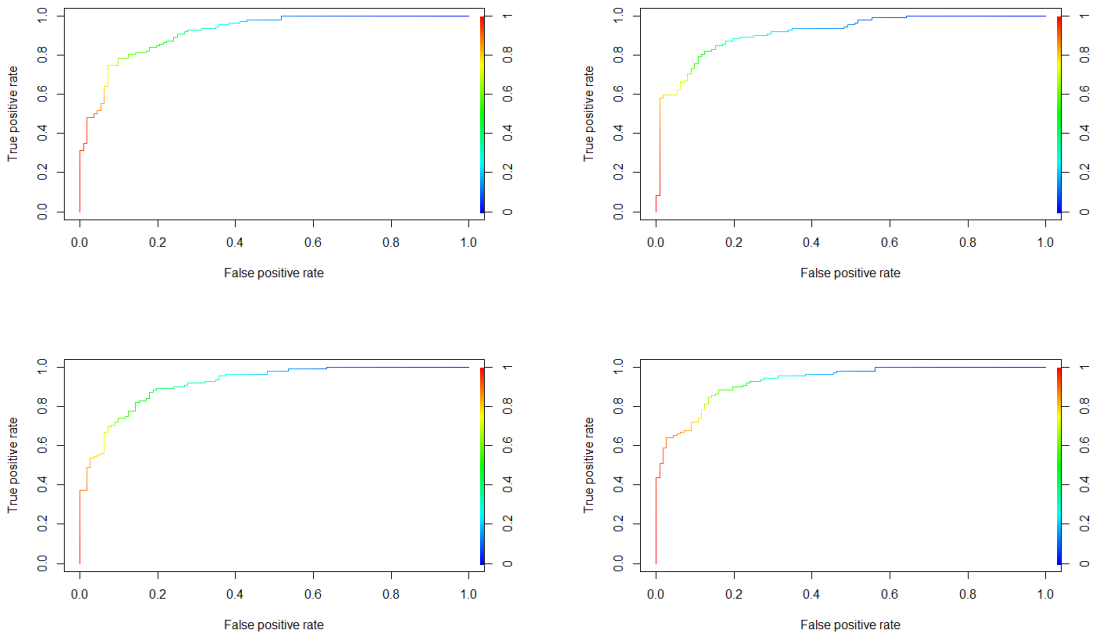


Table 5.10: New Variables

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9
1	-0.25	0.50	0.25	0.05	0.26	0.14	-0.01	-0.13	-0.14
2	0.07	0.39	0.42	0.02	0.24	0.27	-0.07	-0.07	-0.19
3	0.13	0.79	0.20	0.04	0.31	0.11	-0.00	-0.06	-0.22
4	-0.21	0.51	0.55	-0.01	0.27	0.31	-0.04	-0.02	-0.29
5	-0.05	0.53	0.44	-0.00	0.23	0.12	-0.05	-0.01	-0.18
6	-0.04	0.38	0.41	0.00	0.28	0.07	-0.09	-0.02	-0.28
7	0.12	0.49	0.16	-0.08	0.30	0.12	-0.03	-0.17	-0.20
8	-0.04	0.32	0.20	0.14	0.15	0.31	-0.02	-0.05	-0.20
9	-0.30	0.77	0.19	-0.04	0.36	0.19	-0.06	-0.05	-0.22
10	0.00	0.20	0.22	0.08	0.15	0.15	0.04	-0.04	-0.30
11	-0.05	0.60	0.31	0.03	0.40	0.12	-0.02	0.00	-0.29
12	0.03	0.52	0.33	-0.00	0.30	0.18	0.04	0.02	-0.25
13	-0.06	0.49	0.26	-0.19	0.15	0.16	-0.05	-0.00	-0.21
14	-0.09	0.39	0.27	-0.12	0.15	0.11	-0.07	-0.01	-0.31
15	0.04	0.52	0.21	0.06	0.33	0.15	-0.07	-0.01	-0.16
16	-0.02	0.32	0.36	-0.09	0.17	0.14	-0.01	-0.09	-0.26
17	-0.14	0.43	0.35	-0.02	0.10	0.18	-0.05	-0.01	-0.20
...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...
888	-0.04	0.43	0.39	-0.11	0.22	0.22	-0.03	-0.02	-0.14
889	-0.07	0.60	0.25	-0.10	0.23	0.15	0.01	-0.11	-0.19
890	0.04	0.40	0.32	0.10	0.21	0.25	-0.05	-0.01	-0.26
891	-0.06	0.38	0.26	-0.08	0.25	0.09	-0.05	-0.02	-0.13
892	0.07	0.25	0.28	-0.04	0.14	0.05	-0.08	-0.01	-0.25

The last results must be computed on the confusion matrix in the Table 5.11.

Table 5.11: Confusion Matrix - Model 5

	FALSE	TRUE
FALSE	93	19
TRUE	20	92

$$Accuracy = \frac{93 + 92}{93 + 92 + 19 + 20} = 0.8259$$

$$Sensitivity = \frac{92}{92 + 20} = 0.8214$$

$$Specificity = \frac{93}{93 + 19} = 0.8304$$

$$NegativePredictedValue = \frac{93}{93 + 20} = 0.8230$$

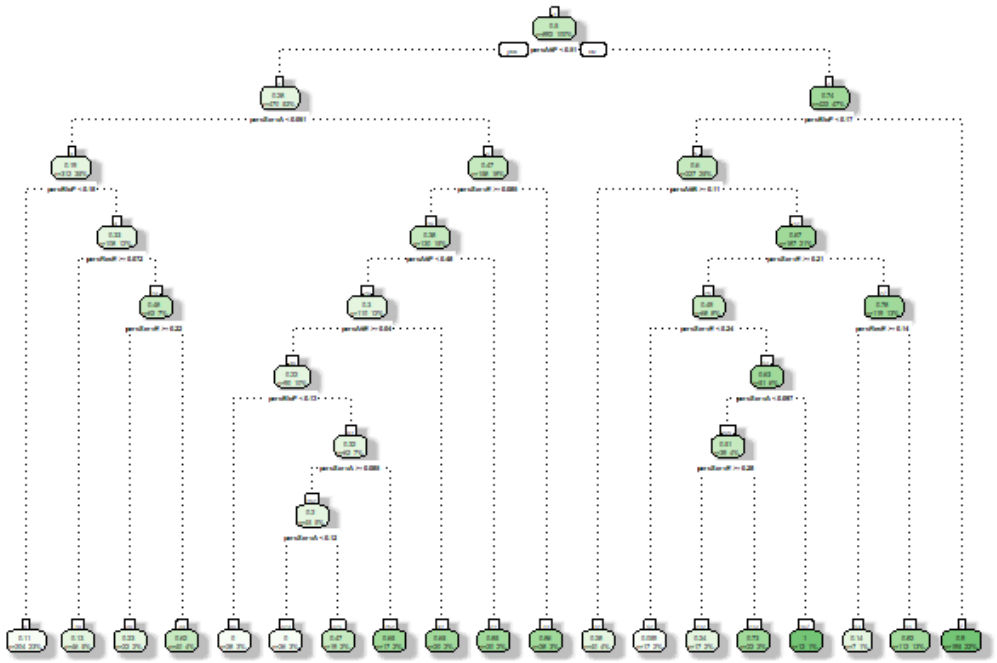
$$PositivePredictedValue = \frac{92}{92 + 19} = 0.8288$$

All the indexes are larger than 80%, so we can consider the logistic regression as a good model to predict a volleyball set result. The rank of principal components is *PC2, PC8, PC7, PC6, PC9, PC4, PC5, PC3, PC1.* From them and combining previous models, we can derive the indirect relevance of variables even if not in a detailed way. The attack phase is the most important one, followed by the serve and the blocking phase, while the reception inhabits the last place. Substantially, offensive gestures drown out the defensive ones.

5.2 Random Forest

The second algorithm is implemented by the package `randomforest` in R. First of all, it does not make any assumption about independence among variables, so the initial set will be used in the model. Random forest does not require any pre-processing analysis on the variables.

Figure 5.6: Decision tree



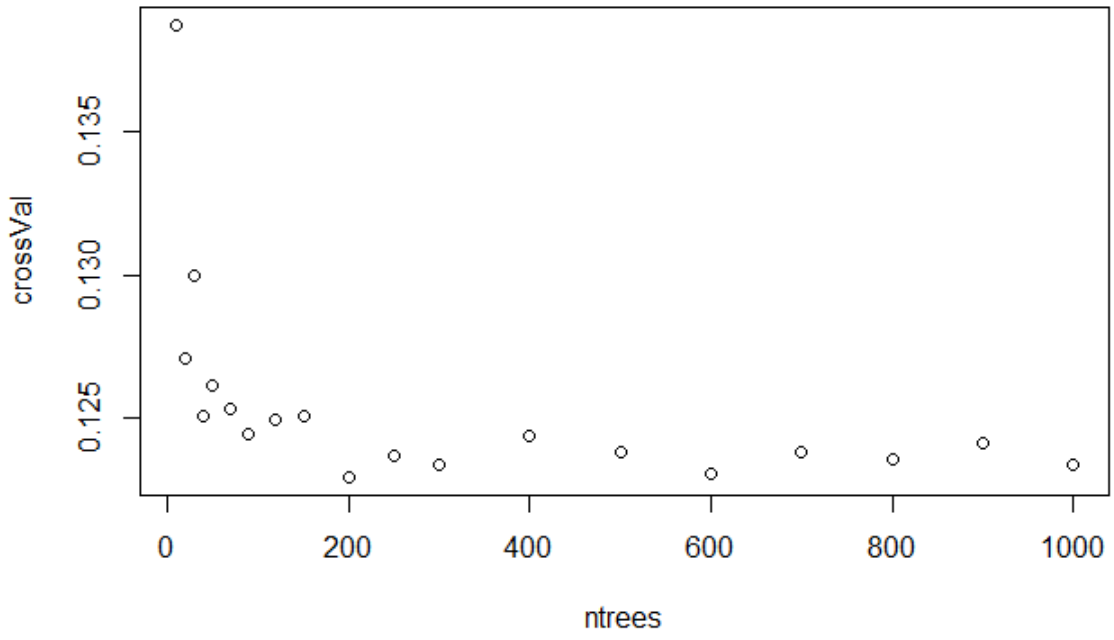
We know that the algorithm ensembles different decision trees, an example of a portion of forest is represented in Figure 5.6.

As explained in Section 2.4.2, we just need two parameters to set: the number of trees in the forest and how many variables to use in each split. For the second one we base on the suggestion $m = \sqrt{p}$ and we will use three of our nine variables. Instead, the number of trees can be tuned using a cross-validation mechanism.

Specifically, the cross-validation function found in [6] computes the mean squared

error between the codified result $\{0, 1\}$ and the predicted value by the regression random forest algorithm. It has been executed for random forests built with different number of trees. We obtain the plot in Figure 5.7 in which the error depends on the number of trees. In a qualitative way, we discover that 200 is a good choice for the number of trees. In fact, adding more trees does not improve the model, while using less trees would not be optimal.

Figure 5.7: Mean Squared Error - Number of trees



We have already observed that two versions of random forest exist and can be implemented to predict the result in our case. Substantially, in the classification random forest, the observations are classified in different qualitative sets, in this specific case winner or loser, thanks to the vote of each tree. Instead, using the regression algorithm, a numerical value is computed as result. We have selected this second version because:

- The numerical result, detected averaging the result of each tree, shows how much we are near to the winning result (1) or to the loser one (0). It is not limited in classifying the team in one class or in the other one.
- In the predictive phase, if the result is greater than 0.5 the team is considered as winner. This is exactly the same that we have done in the logistic models and so they are comparable.
- The regression random forest is more accurate than the classification version.

Anyway, after the prediction each result larger than 0.5 is classified as winner, on the contrary losers have lower values. Similarly to the application of logistic regression, we do not want to fix the training and the test dataset. For 100 times, we rotate the 75% of the dataset to train the model and the remaining part is used to test the model and its results.

Table 5.12: Confusion Matrix - Model 6

	FALSE	TRUE
FALSE	93	19
TRUE	21	91

In the Table 5.12, we observe the confusion matrix in which the number of predictions in each of the four classes has been averaged. Two different goodness of fit measures are computed, the mean of the accuracy that is 0.8214 and the mean of the area under the ROC curves: 0.9067.
The choice of the "split-variables" is of key importance in order to assure accurate outcomes. Based on what the Section 2.4.2 explains, the algorithm generates a real rank of variables according to their role in achieving the result.
The left part of the Figure 5.9 represents how much the accuracy of the model increases, permuting the considered variable and computing the difference of MSE. The right part analyzes the purity of each node through the RSS. We can observe that it gives

Figure 5.8: ROC curve - Model 6

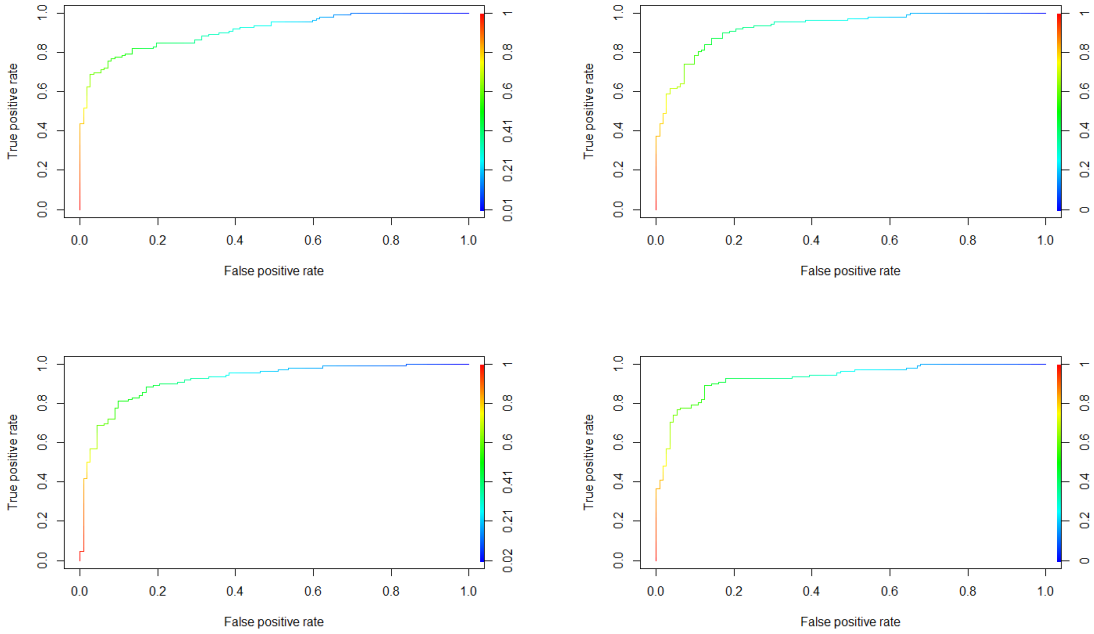
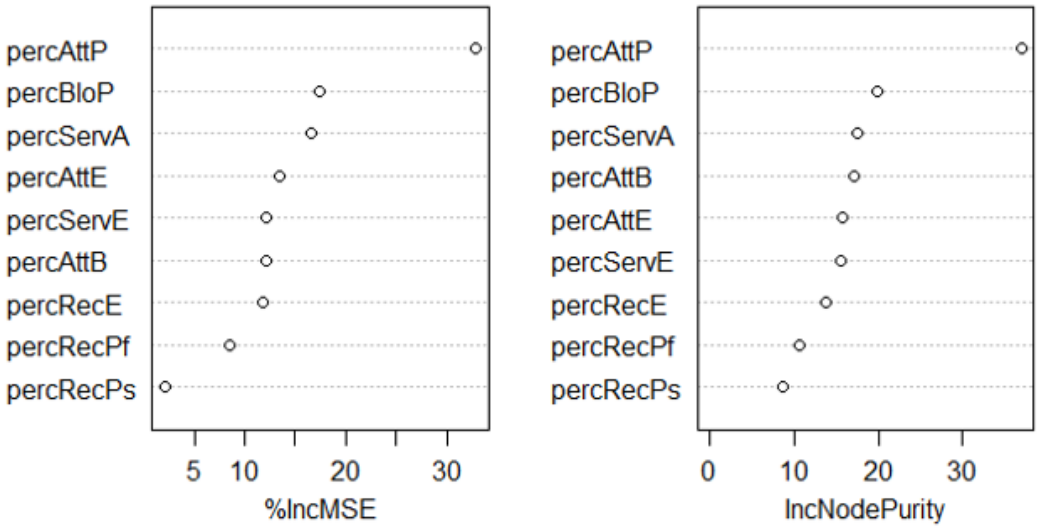


Figure 5.9: Variable importance measures



	%IncMSE	IncNodePurity
percAttP	32.85	37.03
percBloP	17.43	19.80
percServA	16.62	17.42
percAttE	13.39	15.77
percServE	12.07	15.53
percAttB	12.06	17.21
percRecE	11.76	13.82
percRecPf	8.46	10.59
percRecPs	2.10	8.68

more importance to blocked attacks respect to the erroneous ones.

The variables concerning the offensive phase are the most relevant, in particular attack and block points. This aspect underlines that is more important to score a point respect to try not to do an error, both in attack gestures and in serve ones.

In the conclusions we will compare the different implemented models to underline which is preferable and which leads to best results.

Chapter 6

Insights

"Essentially, all models are wrong,
but some are useful."

George Box,1987

In this short chapter the aim is to confirm or deny the thesis which previous models lead to. We used separately the losers and winners' skills to predict both results, now we want to compare them. Considering one skill at a time, more the data distribution for winners is similar to the losers one, less the variable is relevant.
This is a sort of new and less common criterion to reveal features importance. Substantially, if losers do some gesture better than winners but they still lose, this gesture is not so significant.
In particular, in front of non-normal distributions of data, we have used:

i. Mann-Whitney test (or Wilcoxon-Mann-Whitney) [15]: given two independent samples, it tests whether one tends to have values higher than the other. It is commonly regarded as a test of population medians, but this is not strictly true. In fact, Mann-Whitney is a test of both location and shape. Defining X the first population and Y the second one, the null hypothesis is $P(X > Y) = 0.5$. We can choose the alternative hypothesis: concerning "positive" skills we wonder if $P(X > Y) > 0.5$, meaning that winners do it better than losers, the opposite

situation for variables related to errors. For the implementation of the test a rank is given to all observations belonging to the two samples. The statistic is:

$$W = R_1 - \frac{n_1(n_1+1)}{2} \tag{6.1}$$

where, considering only the first sample, R_1 is the sum of rank of observations and n_1 the number of observations. In our case, high values of R_1 underline that the percentages of winners are in general greater respect to the losers ones.

ii. Kolmogorov-Smirnov [22] is a non-parametric test that compares a sample with a known probability distribution or, as in our case, two different samples are compared. K-S is sensitive to differences in both location and shape of the Empirical Cumulative Distribution Functions (ECDF) of the two samples. We can implement different versions of the test remembering that if $P(X > z) \leq P(Y > z)$ then $F_X(z) \geq F_Y(z)$, with X referred to winners and Y to losers. In general, the null hypothesis for positive variables is that $F_X \geq F_Y$ that means X stochastically smaller than Y. The opposite hypothesis is formulated for errors or negative variables. The statistic of the test is:

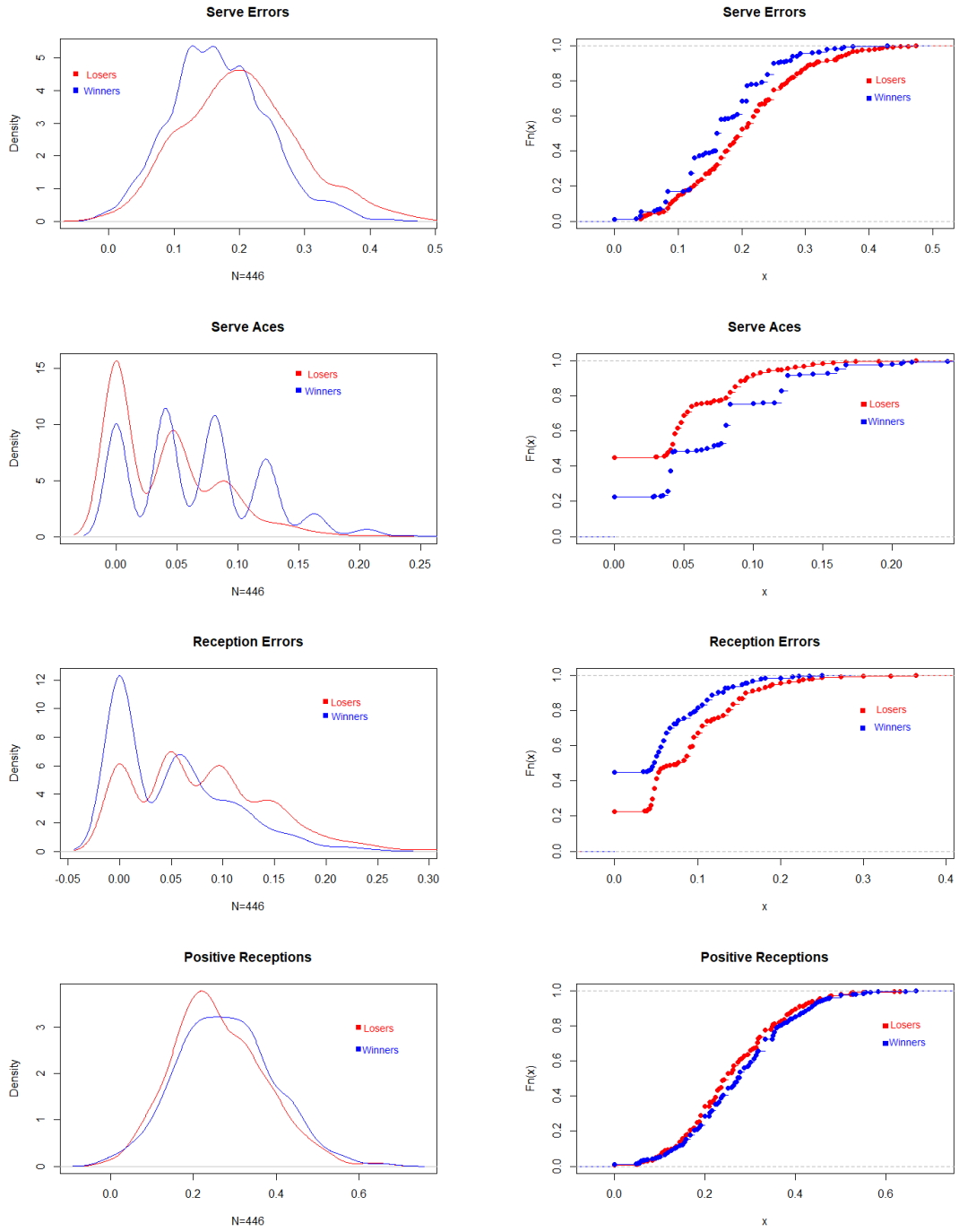
$$D_{n,m} = \underset{z}{max}|F_{X,n}(z) - F_{Y,m}(z)| \tag{6.2}$$

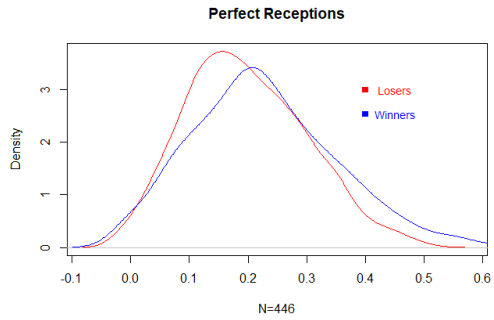
in our case $n = m = 446$. The maximum value of D is 1 and is clearly impossible to reach. Higher is D, larger is the difference between losers and winners teams ECDF.

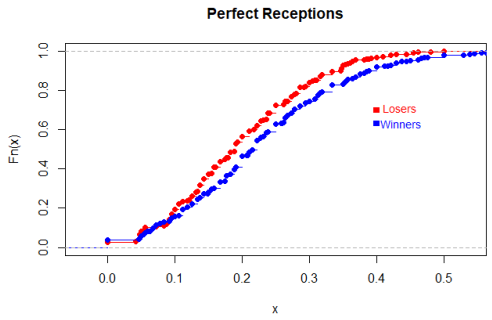
Substantially, accepting the null hypothesis would correspond to detect some "anomalies". This situation does not occur and in the Table 6.1 we can observe very small p-values that suggest us to reject the null hypothesis. Moreover, we use them to confirm the rank of variables importance already found in the previous chapter. Smaller is the p-value related to a variable, stronger is the power of the alternative hypothesis in which we state that winners are better than losers in this specific skill.
The Figure 6.1 is very representative. For less important variables like *Positive Receptions* the shape of densities is very similar and the ECDFs are almost overlapped.

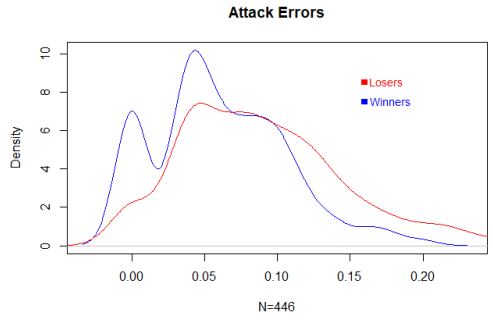
At the contrary, the graph related to *Attack Points*, *Blocked Attacks* and *Block Points* underlines the differences between winners and losers performance.

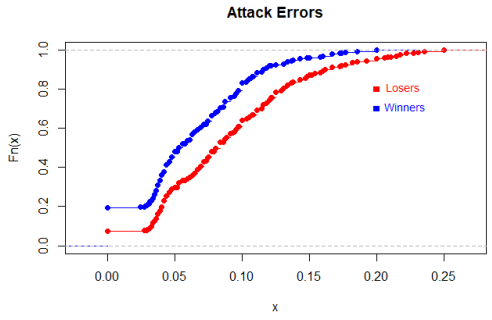
Figure 6.1: Comparing winners and losers performance

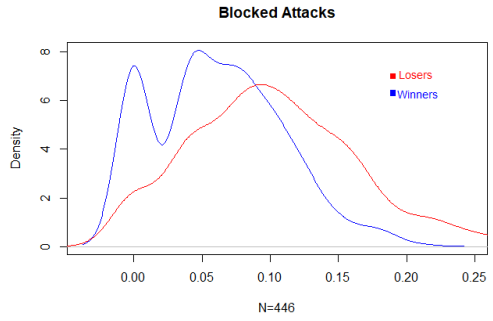


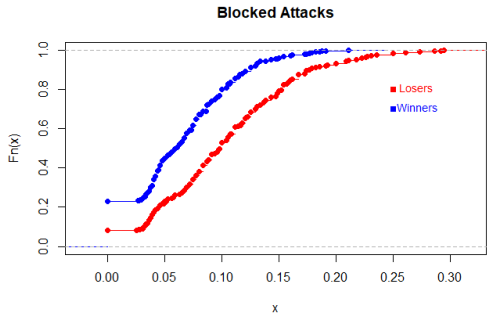

Perfect Receptions
Losers
Winners
Density
N=446

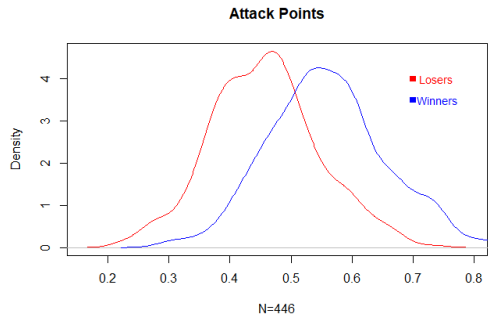

Perfect Receptions
Losers
Winners
Fn(x)
x

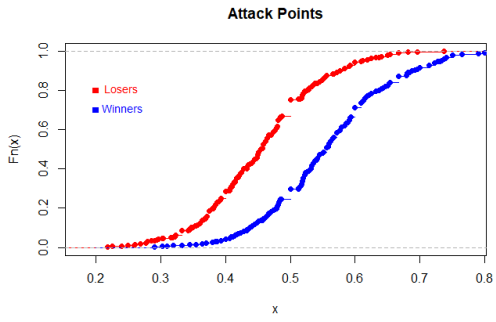

Attack Errors
Losers
Winners
Density
N=446

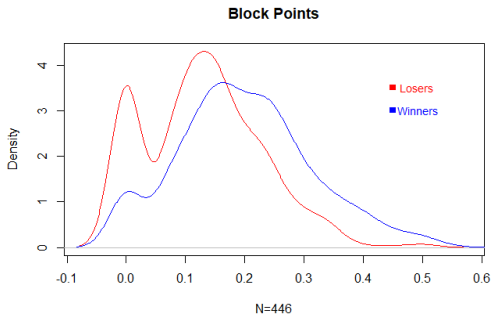

Attack Errors
Losers
Winners
Fn(x)
x


Blocked Attacks
Losers
Winners
Density
N=446


Blocked Attacks
Losers
Winners
Fn(x)
x


Attack Points
Losers
Winners
Density
N=446


Attack Points
Losers
Winners
Fn(x)
x



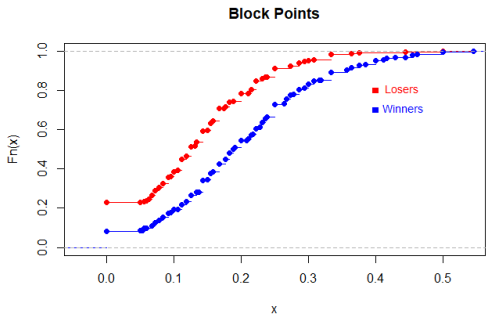


Table 6.1: P-values, MW and Two samples KS test

Variables	MW p-values	MW Statistic	KS p-values	KS Statistic
Attack Points	$<$ 2.2e-16	155910	$<$ 2.2e-16	0.4574
Block Points	$<$ 2.2e-16	135150	$<$ 2.2e-16	0.2848
Blocked Attacks	$<$ 2.2e-16	61228	$<$ 2.2e-16	0.2937
Attack Errors	3.3e-14	70658	1.6e-09	0.2130
Serve Aces	4.5e-12	125180	2.8e-14	0.2646
Reception Errors	1.3e-12	73080	4.4e-12	0.2422
Serve Errors	1.1e-09	76462	1.1e-11	0.2377
Perfect Receptions	0.0002	113020	0.0009	0.1256
Positive Receptions	0.0079	108750	0.0037	0.1121

Chapter 7

Conclusions

"Life is the art of drawing sufficient conclusions from insufficient premises."

Samuel Butler , 1912 d.C.

We have already explained that the aim of this thesis was to identify variables importance and, thanks to them, predict the result of a volleyball match. Moreover, we wish to deliver results also to sportsmen or, more in general, to non-statisticians.
First of all, it is evident that a pre-processing phase is fundamental if we need independent variables. In fact, splitting the set of features has not lead to significant result neither in models in which each predictor was relevant. The accuracy in models with three or four predictors is always between 68% and 78%. After the application of PCA the accuracy of the logistic regression model is higher (82.59%) and it is almost equal to the random forest exactness (82.14%). Since random forest models do not need any algorithm to generate independence among predictors, it is preferable and less computational heavy.
We used the performance of every single team as predictor both in logistic regression and in random forest. This aspect allows to have a larger dataset but it does not take into account the delta between the two opposite teams, better said they never relate to each other.
Anyway, it is possible to understand and underline which technical gesture is more

important in male volleyball and more relevant for our problem. Logistic regression helps us, but because of the sub-division of features and their linear combination in PCA is not so clear. For example, thanks to the first model, we classify in order of importance *Block Points*, *Attack Error*, *Pass Error* and *Serve Error*. From the second model we understand that *Blocked Attacks* have the major relevance followed by *Attack Errors* and *Serve Points*. Finally, thanks to the third model we understand that *Attack Points* dominates all other variables while the reception is the less significant one. Substantially, the rank is the same as the one obtained thanks to random forest indexes, both the difference in MSE and the node purity, but less detailed. This is another reason to prefer the second algorithm.

In Chapter 6, we have compared winners and losers performance using two non parametric tests. Both Mann-Whitney and Kolmogorov-Smirnov show that the winners do less errors and more points than losers, or in general they have higher percentages in positive skills and lower percentages in negative ones. We noticed that an evident gap in the graphical representation of variables corresponds to very relevant features. At the contrary, when winners and losers ECDF and density function are almost overlapped, the considered skill is less significant for the result.

We can certainly state that the offensive gestures are the most important and that, in this phase, the risk must be considered. In fact, to score a direct attack or serve point is much more influential on the result than an error. An interesting aspect is also that if a player could choose the typology of attack error, according to logistic regression, it would be preferable a direct error rather than to be blocked by the opposite team. Blocked attacks have a strong negative impact in the second model on the chance of winning a match, even more considering that the quantity of points in the blocking phase is positively evaluated and occupies the second place in the rank.

Clearly the study is not finished, it is possible to add also less technical variables, as for example the “home-factor”, that is not suitable in a World League. Moreover, it would be curious to implement exactly the same models for the same competition regarding the female volleyball and to highlight the differences. It is very likely that the reception

phase is not so disconnected from the final result and from the attack efficacy. If the computation in female analysis confirms this theory, we can think that males are able to replace some weakness in reception thanks to their physical power. In fact, the most significant variable in our study is the one in which the physical strength is more visible: the spike, and its relevance is not near or comparable with the effect of other predictors.

Another further, but more difficult development of the study is to create a marriage with less mathematical subject. In volleyball, as in many other sporting disciplines, the human aspect is very relevant. For this reason, it could be interesting to combine statistical and psychological studies and observe how the results are affected.

Bibliography

[1] H. Abdi and L. J. Williams. Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2, 2010.

[2] Y. Benjamini. Opening the box of a boxplot. *The American Statistician*, 42, 1988.

[3] L. Breiman. Random forests. *Machine Learning*, 45, 2001.

[4] L. Breiman and R. Ihaka. Nonlinear discriminant analysis via scaling and ace. *Technical report*, 1984.

[5] T. A. Craney and J. G. Surles. Model-dependent variance inflation factor cutoff values. *Quality Engineering*, 14, 2002.

[6] S. Fiorito. Variable importance in modern regression with application to supplier productivity analysis, Thesis, Politecnico di Torino, 2018/2019.

[7] G. Hall. Pearson's correlation coefficient. 2015.

[8] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer, 2nd edition, 2009.

[9] M. Häyrinen, T. Hoivala, and M. Blomqvist. Differences between winning and losing teams in men's european top-level volleyball. In *Proceedings of VI Conference Performance Analysis.* Citeseer, 2004.

[10] G. James, D. Witten, T. Hastie, and R. Tibshirani. *An Introduction to Statistical Learning with Applications in R.* Springer Science & Business Media New York, 2013.

[11] T. Knapp and V. Swoyer. Some empirical results concerning the power of bartlett's test of the significance of a correlation matrix. *American Educational Research Journal*, 4, 1967.

[12] W. J. Krzanowski and D. J. Hand. *ROC Curves for Continuous Data.* Chapman & Hall / CRC, 1st edition, 2009.

[13] A. Liaw, M. Wiener, et al. Classification and regression by random forest. *R news*, 2, 2002.

[14] B. H. Menze, B. M. Kelm, R. Masuch, et al. A comparison of random forest and its gini importance with standard chemometric methods for the feature selection and classification of spectral data. *BMC Bioinformatics*, 2009.

[15] N. Nachar et al. The mann-whitney u: A test for assessing whether two independent samples come from the same distribution. *Tutorials in Quantitative Methods for Psychology*, 4, 2008.

[16] C. Y. J. Peng, K. L. Lee, and G. M. Ingersoll. An introduction to logistic regression analysis and reporting. *The journal of educational research*, 2002.

[17] D. Posada and T. R. Buckley. Model selection and model averaging in phylogenetics: Advantages of akaike information criterion and bayesian approaches over likelihood ratio tests. *Systematic Biology*, 53, 2004.

[18] B. D. Ripley. *Pattern Recognition and Neural Networks.* Cambridge University Press, 1996.

[19] Y. Sakamoto, M. Ishiguro, and G. Kitagawa. Akaike information statistics. *KTK Scientific*, 1986.

[20] P. Sedgwick. Pearson's correlation coefficient. *British Medical Journal Publishing Group*, 2012.

[21] G. Vagenas and S. Drikos. Multivariate assessment of selected performance indicators in relation to the type and result of a typical set in men's elite volleyball. *International Journal of Performance Analysis in Sport*, 2011.

[22] Y. Xiao. A fast algorithm for two-dimensional kolmogorov–smirnov two sample tests. *Computational Statistics and Data Analysis*, 105, 2017.

List of Figures

List of Tables

Ringraziamenti

"*Great achievement is usually born*
of great sacrifice, and is never
the result of selfishness."

Napoleon Hill, 1928

Al mio relatore, Franco Pellerey, e al professor José María F.dez Ponce va il primo sentito ringraziamento per l'entusiasmo riposto nel mio progetto, per la fiducia e la disponibilità che non sono mai venute a mancare. Il soggiorno all'estero che mi è stato permesso di fare per lavorare a quest'elaborato rimarrà per sempre una delle più importanti esperienze della mia vita.

Nel percorso che mi ha portato fino a qui, sono passata attraverso periodi in cui mi sembrava di non poter essere grata di nulla, giornate in cui questa pagina sarebbe rimasta bianca, riempita solo di rabbia e dolore. Ma è stato proprio in quei momenti che, voltando lo sguardo, mi sono accorta di non essere sola…

Ringrazio quindi Giulia, Luisa e Melissa, che c'erano, ci sono e ci saranno sempre, compagne di studi e di vita senza le quali questo traguardo sarebbe stato ancora più faticoso.

Ringrazio la mia squadra di pallavolo e Daniele, per non aver mai smesso di credere in me, dentro e fuori dal campo, per aver compreso e appoggiato la mia partenza e per mantenere viva con me la fiamma della passione per questo sport.

A tutte le altre amicizie, che mi sopportano e mi supportano da anni, chi dalle scuole medie, chi dalle superiori e chi, come Filippo, è entrato nella mia vita quasi per caso e da quel momento non ha mai smesso di fare per me un tifo sincero.

A Gianluca, che ha vissuto con me le decisioni più importanti della mia vita, guidandomi senza mai soffocare la mia essenza, grazie per non esserti mai tirato indietro anche quando la vita ci ha reso distanti.

A Edoardo, un grazie di cuore per tutte quelle volte in cui con il tuo amore sei riuscito ad attenuare le ansie e le paure di questi ultimi due anni.

Grazie a nonna Rosa, per essere seplicemente così come è, sento il tuo pensiero qui

con me anche se sei lontana.

A nonna Lia, per non avermi mai negato un pasto caldo, anche quando ti chiamavo all'ultimo minuto e quasi ti sentivi triste per non avere cucinato per me niente di elaborato, senza sapere che io in realtà venissi solo per la tua compagnia.

C'è un ringraziamento però, a cui qualsiasi parola scritta non rende abbastanza giustizia, quello dedicato alle tre persone più importanti della mia vita. Al mio papà, per i sacrifici e per l'esempio di serietà e altruismo che ti sei sempre dimostrato. Alla mia mamma, un immenso grazie per come riesci a riempire casa nostra con tutto il tuo amore, senza il quale sembrerebbe fredda e vuota. A mia sorella Roberta a cui, se pur così diversa da me, sono grata per essere da sempre la miglior complice che si possa desiderare nella vita…

Frenchi, Robi, abbiamo vissuto un inferno, perdendo un padre esemplare ed un marito affettuoso, ma il modo in cui ci siamo unite, sostenendoci a vicenda, lo rende sicuramente fiero di noi in qualsiasi posto si trovi, nella speranza che oggi lo sia di me, ancora di più.

"Perchè la vita senza te non può essere perfetta…"