UNIVERSITÀ DEGLI STUDI DI PADOVA

Dipartimento di Scienze Statistiche
Facoltà di Scienze Statistiche

CORSO DI LAUREA SPECIALISTICA IN STATISTICA E INFORMATICA

PARAMETRIC AND NONPARAMETRIC MANCOVA METHODS FOR COMPARING AND RANKING OF MULTIVARIATE POPULATIONS, WITH APPLICATION TO INDUSTRIAL EXPERIMENTS

Relatore: Ch.^{mo} Prof. Fortunato Pesarin

Correlatore: Ch.^{mo} Dott. Livio Corain

Laureando: Daniele Gomiero

Contents

Riassunto

Lo scopo di questa tesi è quello di proporre e validare alcune nuove soluzioni parametriche e non parametriche nel contesto del modello multivativariato dell'analisi della varianza/covarianza – MANCOVA per il confronto e l'ordinamento di alcune popolazioni multivariate di interesse. La necessità di modellizzare l'effetto multivariato del trattamento con un modello MANCOVA, invece di utilizzare un più tradizionale schema MANOVA, a volte è una necessità pratica, in particolare per gli esperimenti industriali dove il tempo e il costo per ottenere repliche sono troppo elevati da sostenere, per cui si ricorre ad esperimenti con differenti covariate di cui è doveroso tener conto. Un esempio di questa situazione si verifica nel settore dell'industria della detergenza per valutare le cosiddette prestazioni secondarie durante il processo di sviluppo di un nuovo detersivo. In questo contesto, è interessante valutare i benefici che sono misurabili solo dopo un certo numero minimo di cicli di lavaggio, che in questo contesto svolge il ruolo della covariata, senza possibilità di replicare gli esperimenti.

Nel primo capitolo "Introduction and motivation", introduciamo e motiviamo il problema di interesse, mostrando alcuni casi di studio reali con riferimento al processo di sviluppo industriale di un nuovo detersivo.

L'obiettivo del secondo capitolo "Formalization of the problem" è quello di formalizzare il problema e di presentare due diverse soluzioni: la prima parametrica e la seconda non parametrica. La prima deriva dalla assunzione di normalità degli errori casuali e si basa sui test tradizionali, test *t* e test F, mentre la seconda è una soluzione non parametrica condizionale di permutazione. Anche se la soluzione non parametrica sembra essere più robusta e flessibile, si deve rilevare che potrebbe non essere molto potente a causa della bassa numerosità campionaria e della relativa bassa cardinalità dello spazio di permutazione.

Il terzo capitolo "Simulation study" è dedicato a studiare e a confrontare le soluzioni proposte per mezzo di un adeguato studio di simulazione, al fine di convalidare ed esaminare il comportamento delle soluzioni proposte con diverse forme funzionali delle variabili di risposta.

Nel quarto capitolo "Application to real case studies" applichiamo le soluzioni proposte in alcuni casi di studio reali, riguardanti la valutazione delle cosiddette prestazioni

secondarie durante il processo di sviluppo di un nuovo detersivo. In questa sezione le soluzioni proposte sono applicate anche al caso di esperimenti con repliche, anche se questi sono un po' insoliti nel quadro di riferimento del settore industriale di interesse.

Nel quinto ed ultimo capitolo "Conclusion and future research", le principali conclusioni di questo lavoro sono presentate e discusse, assieme ad alcune indicazioni per le ricerche future. Uno dei risultati più rilevanti è che a livello univariato i trattamenti sono correttamente discriminati utilizzando l'approccio parametrico proposto, mentre, al fine di discriminare e classificare adeguatamente le popolazioni da un punto di vista multivariato, la soluzione non parametrica basata sulla combinazione e sui test di permutazione sembra essere il metodo più efficace.

Infine, in appendice vengono riportiamo tutti i codici R sviluppati, che sono stati utilizzati per la realizzazione della presente tesi.

Summary

The purpose of this thesis is to propose and validate some novel parametric and nonparametric solutions within the multivariate analysis of variance/covariance – MANCOVA layout for comparing and ranking several multivariate populations of interest. The requirement to model the multivariate treatment effect with a MANCOVA model instead of using a more traditional ANOVA layout is sometime a practical need, especially for industrial experiments where time and cost to get replications are to high to support, for this reason is used experiments with different covariates to which we must take into account. One example of this situations occurs in the laundry industry when evaluating the so called secondary performance during the process of new detergent development. In this context, it is interesting to assess the benefits which are measurable only after a certain minimum number of washing cycles, which plays in this framework the role of covariate, without the possibility to replicate the experiment at all.

In chapter one "Introduction and motivation" we introduce and motivate the problem at hand, showing several real case studies with reference to the industrial process of new detergent development.

The aim of the second chapter "Formalization of the problem" is to formalize the problem and to present two different solutions: the first parametric and the second nonparametric. The first one derives from the assumption of normality of random errors and it is based on the traditional *t* and F test, the second one is a nonparametric combination-based permutation solution. Even if the nonparametric solution appears to be more robust and flexible, it should be noted that it could be not very powerful due the relative small sample size and to the consequent small cardinality of the permutation space.

The third chapter "Simulation study" is devoted to study and compare the proposed solutions by means of a suitable simulation study in order to validate them and to investigate their behaviours with different distribution for the response variables.

In chapter four "Applications to real case studies" we apply the proposed solutions to several real case studies concerned with the evaluation of the so called secondary performance during the process of new detergent development. In this section the

proposed solutions are applied also to the case of experiments with replications, even if these are somewhat unusual in the framework of our reference industrial field.

In the fifth and final chapter “Conclusion and future research”, the main conclusions of this work are presented and discussed, along with some directions for future research. One of the most relevant obtained results is that at univariate level the treatments are properly discriminated using the proposed parametric approach whereas, in order to suitably discriminate and rank the populations from a multivariate point of view, the nonparametric combination-based permutation solution appears to be the most effective method.

Finally, appendix A reports all developed R codes we used for the realization of the present thesis.

Chapter 1. Introduction and motivation

The need to define an appropriate ranking of items (products, services, teaching courses, degree programs, and so on) is very common in both experimental and observational studies within the areas of business and industrial research. In the field of New Product Development the research aim is often focused on evaluating the product/service performances from a multivariate point of view, i.e. in connection with more than one aspect (dimension) and/or under several conditions (strata).

For example, when developing new detergents, the laundry industry is used to refer to the so-called secondary detergency, that is the assessment of benefits which are measurable only after a certain number of washing cycles (usually 5, 10 or 15). When performing a secondary detergency experiment, given that the benefits are simultaneously evaluated on several different piece of fabrics (which differ one each other by the type of textile), the response variable can be actually considered as multivariate in nature (Bonnini et al., 2009). There are several useful secondary detergency performances: Whiteness Degree, Greying or Y-Value, Tint Value, Dye Fading and Dye Transfer Inhibition (AISE, 2009).

In this context, the need to take into account for the presence of one covariate (washing cycles) suggests us to consider as reference statistical multivariate model a regression-based layout where the effect of the items under investigation (products) can be modelled by some possible change in the slope of the response variable (secondary performance). As an alternative nonparametric combination-based permutation solution, a more general model could be considered as well.

Revision of the literature highlights some gaps on the problem at hand. In fact, even if MCPs - Multiple Comparison Procedure methods address the problem of ranking the treatment groups (Westfall et al., 1999), there is, in general, no clear indications on how dealing with the information from pair-wise comparisons, especially in case of a multivariate response variable. Moreover, the ranking and selection approach in multiple decision theory (Gupta and Panchapakesan, 2002) provides some hints, but essentially for univariate problems and under assumption of normality.

With reference to the inferential analysis, our goals are: (i) determining whether the items are equivalent against the alternative that they are different; (ii) if the items have been found as different:

- for each item, we define and estimate a suitable (either point or interval) multivariate indicator to quantify the relative preference that can be assigned to each item in comparison with everyone else;
- we determine a global (multivariate) preference ranking (from the "best" to the "worse").

1.1 Some statistical issues on the problem

The topic of the thesis is concerned with some novel parametric and nonparametric solutions within the MANCOVA layout for comparing and ranking several multivariate populations of interest.. The problem of ranking populations arises from the need for a realistic formulation of the practical problem of comparing C given populations with the goal of rank them. This problem is not only of theoretical interest but also it has a recognized relevance, as it will be pointed out below. Note that the traditional approach of homogeneity tests, i.e. testing for the equality of the means or distributions, does not address such goals.

Historically, then, the problems of statistical inference were basically formulated as those of estimation or testing of hypotheses. This formulation, however, does not exactly fulfil the objectives of an experimenter in many situations when he is faced with the problem of comparing several populations. These are generally the populations of the responses to certain "treatments". In all these problems, we have $C \geq 2$ populations and each population is characterized by the value of a parameter θ, which may denote, for example, the average of some meaningful variables for a variety of treatments that may represents different type of new products or prototypes under investigation by the research and development division of a firm. In fact, very often, when developing new products the research aim is often focused on evaluating the product/service performances from a multivariate point of view, that is, in connection with more than one aspect (dimension) and/or under several conditions (strata).

Indeed, in many practical situations, when comparing systems or groups, we are interested simultaneously in two or more characteristics of each component or

individual, so our observations are vector-valued and the components of the vector observations may be referred to correlated random variables.

Moreover, when the performance evaluation takes into account more than one aspect, the problem can be complicated and some methodological and practical issues arise: standardization, multivariate structure of data, accuracy of partial indicators, distance with respect to a target (highest satisfaction level), stratification in presence of confounding factors (see Fayers and Hand, 2002).

The classical approach in all the preceding situations is to test the hypothesis of homogeneity of the parameters: $H_0 : \theta_1 = ... = \theta_C$ where $\theta_1, \ldots, \theta_C$ are the values of the parameter θ for these populations. In the general setting θs are functionals, i.e. functions of all parameters defining the involved distributions. With clear meaning of the symbols, $\theta_i = \int_{\Omega_i} \theta(X)\, dF_i(X)$, where $F_i(X)$ is the c.d.f. and Ω_i is the sample space of the i-th group. If the populations are assumed to be normal with means $\theta_1, \ldots, \theta_C$, and a common variance σ^2, then the test can be carried out by using the traditional one-way analysis of variance technique.

In cases of other distributions for which θ may denote a different measure, one can develop a test of the null hypothesis H_0 using the Neyman-Pearson theory. Such tests have been developed for various situations and many of these are available in the statistical literature. It should, however, be recognized that a satisfactory solution to any statistical inference problem depends on the goal of the experiment. In this sense, the classical tests of homogeneity cannot provide a satisfactory solution for these problems. In situations like these, the goal of the experimenter is to identify the variety with the largest average (most effective detergent in removing soils, most effective educational system, most effective drug, and so on) rather than just to accept or reject the homogeneity hypothesis.

When the homogeneity test is carried out and its result is significant, the experimenter faces some real problems: he could use the method of least significant differences, based on t-tests, to detect differences between θs and thus to choose the "best" population. Nevertheless this method is at best indirect and quite inefficient, because it lacks protection in terms of a guaranteed probability against selecting a wrong population as the "best".

Furthermore, although in general one may define a partial order relation on the set of multivariate normal distributions, defining a real-valued function θ_i of the parameters μ_i (usual mean), and Σ_i (variance-covariance matrix), and use the θ_i to rank the populations, when distribution are different and particularly when the assumption of multivariate normal distribution is not realistic (quite often in practical situations) these function θ_i has to be constructed carefully and possibly using different parameters.

As mentioned earlier it is clear that the problem of finding out a global ranking in terms of performance of all investigated products/prototypes, which is a very natural goal for business and industrial research, can be viewed as a problem of ranking of multivariate populations. In general, from the statistical point of view, when the response variable of interest is multivariate in nature, the problem may become in general quite difficult to cope with due to the high dimensionality of the parameter's space.

In order to reach our goal, i.e. ranking treatmets from the "best" to the "worst", we will use intensive computer calculation methods, namely nonparametric methods referred to various resampling strategies (Arboretti et al., 2010). To be more specific, we will use *permutation tests* approach to the *NonParametric Combination* – NPC methodology (Pesarin and Salmaso, 2010). It is noteworthy that this methodology allow the experimenter for not modeling the dependence structure among partial tests (and so among variables) involved in it, because it is nonparametric with respect to the underlying distribution.

1.2 Industrial experiments for laundry manufacturing: secondary performances

Some very important aspects for laundry industry when developing new products is to consider the so-called secondary performances, that is the assessment of benefits which are measurable by a suitable response variable only after a certain minimum number of washing cycles. Secondary performances can be divided into:

- *Whiteness degree*, *Tint value* and *Redeposition*: whiteness degree and tint value measure the effect of optical brightener in the detergent on the white fabrics, while the redeposition measures greying/redeposition of soil onto white fabrics;
- *Colour transfer*: it measures the dye transfer from a certain number dye donors onto white acceptors;

- *Colour fading*: it measures the fading of a certain number of dyes after repeated wash cycles;
- *Inorganic incrustation*: it is the assessment of limescale build up on the heating element and the fabrics after several repeated wash cycles.

For each secondary performance experiment it is important to set up all test conditions before the beginning of the test. It must be highlighted that, in this case of secondary performances, it is common practice to not carry out any replications of the experiments because of time and cost constraints.

1.3 Examples of real data

In this section we present some real case studies concerned with industrial experiments on the assessment of secondary performances. Data are displayed in graphical form, so as to capture the trend of observations at different covariate levels (number of washes) and for different products under investigation. These data are based on historical database and/or ad-hoc real experiments of secondary performances achieved by the Research and Development – R & D division of Reckitt Benckiser Spa, Mira (VE), Italy.

Although the graphics are presented at univariate level, we recall that the secondary performance is simultaneously evaluated on several different piece of fabrics (which differ one each other by the type of textile), so that the response variable can be actually considered as multivariate in nature.

1.3.1 Whiteness

The following experiment shows the measurements which have been observed in performing the secondary performance called *Whiteness*, where eight products (P1, ..., P8) are compared at different numbers of washes (5,10,15,20,25) and assessing the goodness of these products in seven different types of textile (labelled WFK10, WFK20, WFK30, WFK40, WFK50, Empa221, Terry-cloth cotton). For the sake of simplicity we present below (Fig. 1 and 2) only the experimental data obtained for some of these textiles, in particular the responses observed on both types of tissue “WFK20” and “WFK50”.

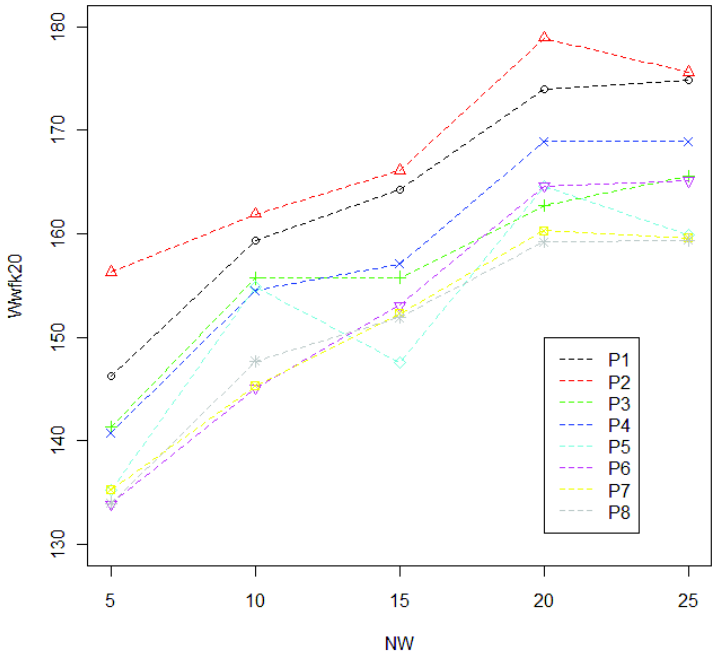

Figure 1. Whiteness by product and number of washes (WFK20 textile)

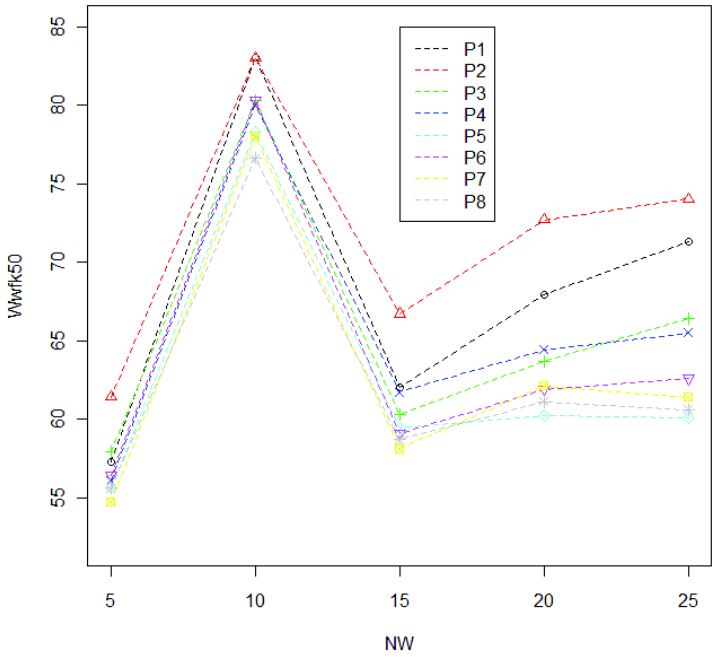

Figure 2. Whiteness by product and number of washes (WFK50 textile)

1.3.2 Redeposition

As regards the secondary performance called *Redeposition*, the experiment is performed with the same structure of Whiteness. Again (Fig. 3 and 4) we consider only two different tissues, that is “WFK20” and “Empa221”.

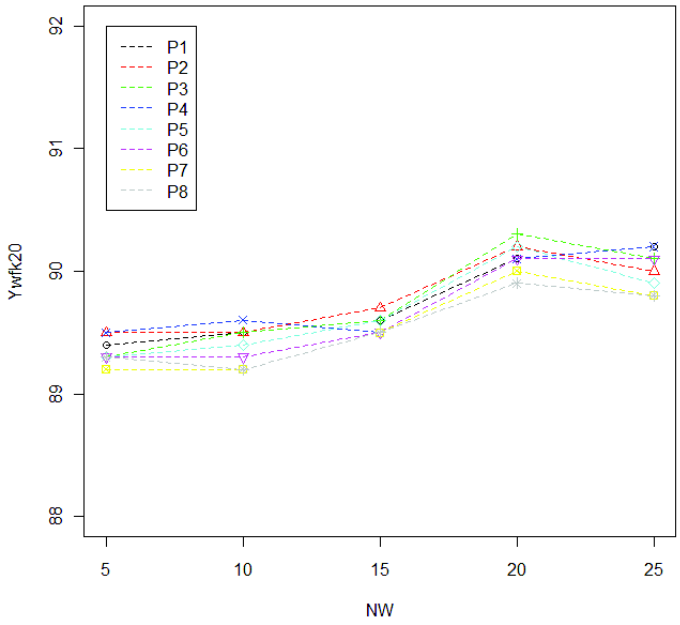

Figure 3. Redeposition by product and number of washes (WFK20 textile)

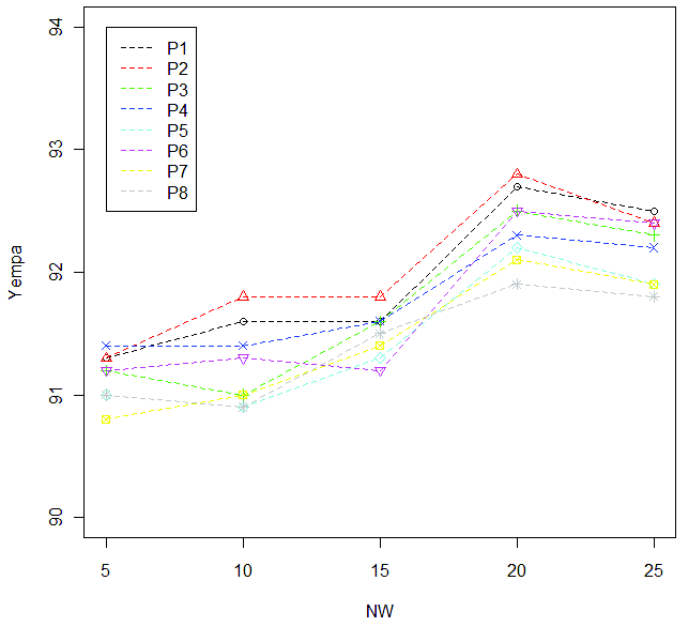

Figure 4. Redeposition by product and number of washes (Empa221 textile)

1.3.3 Colour Transfer

In assessing the secondary performance called *Colour Transfer*, instead of carrying out the experiment to test the quality of the product on different types of textile, it goes to test the quality of several products in the transfer of colour for different type of colours (15, in specific). In this experiment we are going to test three different products (P1,….,P3) . For simplicity again we present data for a single colour, that is "C5" (Fig. 5).

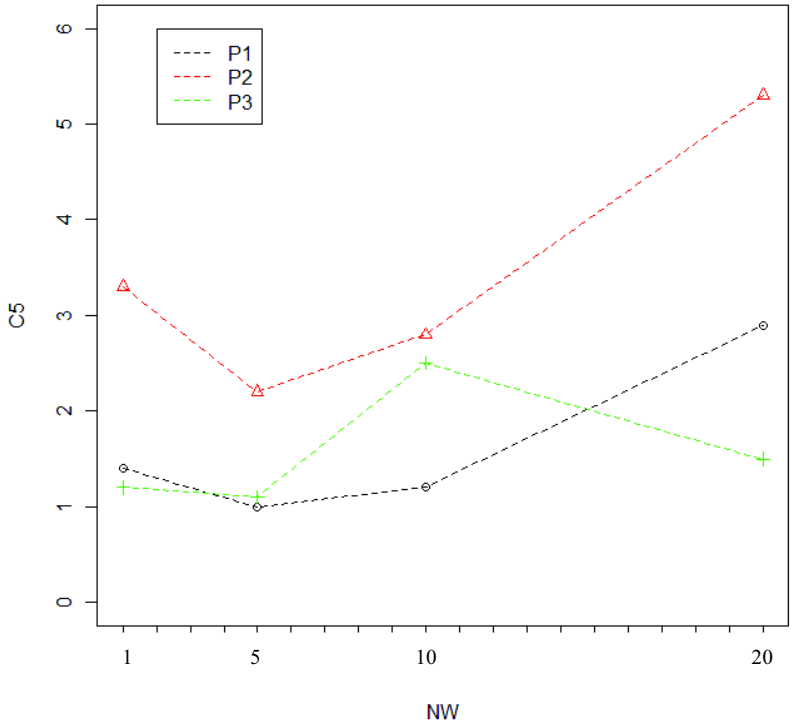

Figure 5. Colour Transfer by product and number of washes (C5 colour)

1.3.4 Colour Fading

The experiment related to the secondary service called *Colour Fading* is very similar to the experiment on Colour Transfer (colour fading instead of transfer is here considered). In the specific case study (4 products: P1,…,P4), three replications of the experiment were obtained for each product. It must be emphasized that in the other presented cases there were not replications but just an observation for each product. In fact, experiments with replication are generally not carried out because of the considerable costs involved. Here we present data for colour named as AISE 1 (Fig. 6).

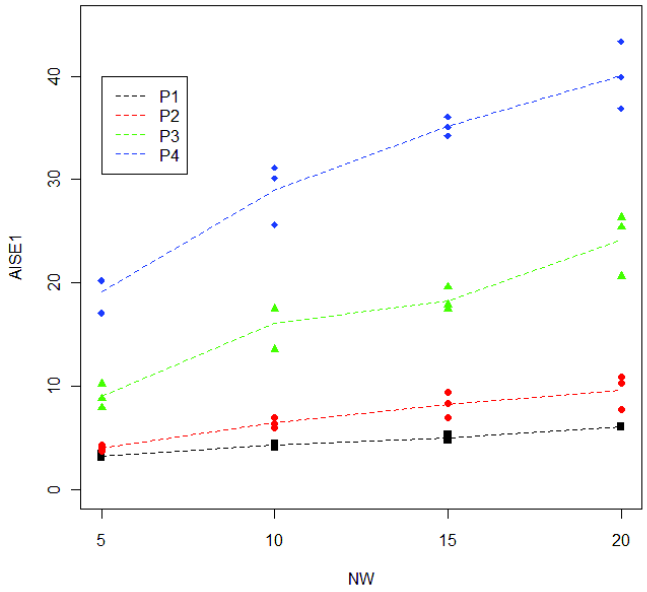

Figure 6. Colour Fading by product and number of washes (AISE1 colour)

1.3.5 Heating Element Incrustation

Unlike all other secondary performances, in the experiment to evaluate the performance called *Heating Element Incrustation* the response variable obtained from the experiment is unique (univariate problem). In this case we consider three products (P1,…,P3).

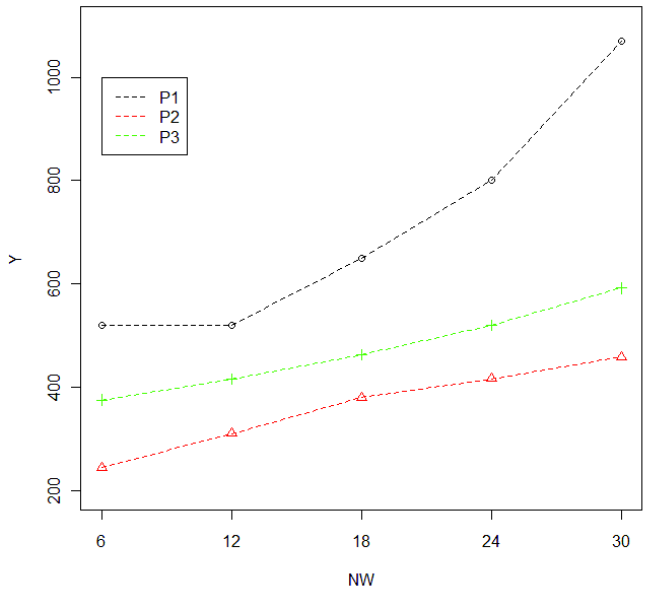

Figure 7. Heating element incrustation by product and number of washes

Chapter 2. Formalization of the problem

This chapter is devoted to formalize the problem at hand and to present two different solutions: the first parametric and the second nonparametric. The first one derives from the assumption of normality of random errors and it is based on the traditional *t* and F test whereas the second one is a nonparametric combination-based permutation solution. In particular, in Section 2.1 we are going to state the model used to represent the experiment of interest and the inferential analysis of the data; in Section 2.2 the goal is to propose solutions for pairwise comparisons between populations through parametric and nonparametric hypothesis testing procedures. In Section 2.3 we tackle the problem of improving the fitting of the regression model, with reference to the application context of our interest. Finally, in section 2.4 we will present the methods of the so-called global ranking, in order to obtain an order of populations from a multivariate point of view.

2.1 Statistical model

Let **Y** be the multivariate response variable representing a *p*-vector of the observed data. The actual design is defined by the comparison of C treatments with respect to *p* different variables, observed in correspondence to *n* values of a covariate X. For sake of simplicity we refer to the unreplicated design but the extension to the replicated case is straightforward.

The C-group multivariate statistical model can be represented as follows:

$$\mathbf{Y}_{ij} = \boldsymbol{\mu}_{ij} + \boldsymbol{\varepsilon}_{ij}, \quad i = 1,...,n\ , \quad j = 1,...,C, \tag{1}$$

where $\boldsymbol{\mu}_{ij} = f(X_i;\boldsymbol{\beta}_j)$, is the *p*-dimensional mean effect conditioned on X_i, $\boldsymbol{\varepsilon}_{ij} \sim IID(0,\boldsymbol{\Sigma})$ is a *p*-variate random term of experimental errors with zero mean, variance/covariance matrix $\boldsymbol{\Sigma}$.

Within the parametric framework, for the *k*-th response variable, $k=1,\ldots,p$, a reasonable choice is to assume a *q*-degree polynomial, that is

$$\mu_{ijk} = f(X_i;\boldsymbol{\beta}_{jk}) = \beta_{0k} + \beta_{1jk} X_i + \sum_{s=2}^{q} \beta_{sk} X_i^s\ . \tag{2}$$

Note that, in order the make as small as possible the number of parameters to be estimated, the effect of the *j*-th treatment is set up only on the first degree part of the

polynomial. Given that the treatment effect could be properly defined also on a higher polynomial order, an empirical investigation on real datasets should be performed in order to find out the more effective parameterisation.

As an alternative specification within a nonparametric framework and without the need of specifying any functional form of the link function, two-way layout may be taken into consideration, that is

$$\mu_{ijk} = \mu_k + \tau_{jk}(\mathrm{X}_i). \tag{3}$$

Note that expression (3) is quite general, because it simply states that the *j*-th treatment effect on the *k*-th response variable is a function of the *i*-th level of the covariate X.

2.2 Statistical inference

In this section we present two different inferential solutions for the problem of interest, that is comparing C treatments with respect to each one of the *p* different response variables, observed in correspondence to *n* values of a covariate X. The first solution is a parametric one and is referred to expression (2), whereas the second is a nonparametric combination-based permutation solution which should be referred to expression (3).

2.2.1 A parametric solution

The first parametric solution may be referred to expression (2) and derives from the assumption of normality for random errors. It is based on the traditional *t* and F tests where, as well known, random errors have to be assumed as independent and identical distributed normal, with zero mean and constant variance (Draper and Smith, 1998).

Let us consider the univariate regression model derived by expressions (1) and (2),

$$\mathrm{Y}_{ijk} = \beta_{0k} + \beta_{1jk}\,\mathrm{X}_i + \sum_{s=2}^{q} \boldsymbol{\beta}_{sk}\mathrm{X}_i^s + \varepsilon_{ijk}, \tag{4}$$

where $i=1,\ldots,n$, $j=2,\ldots\mathrm{C}$, $k=1,\ldots,p$ and where, without loss of generality, the first treatment is considered as baseline and so removed from the model. In order to meet our first inferential goal, that is determining whether the treatments are equivalent against the alternative that they are different, the first step is related to the usual global test on all β_1s. The appropriate hypotheses are:

$$\begin{cases} H_{0k} : \beta_{1jk} = \ldots = \beta_{1Ck} \\ H_{1k} : \exists \beta_{1jk} \neq \beta_{1hk} \end{cases}, j,h = 1,\ldots,\mathrm{C}, j \neq h, k = 1,..,p \ . \tag{5}$$

Rejection of H_{0k} implies that at least one of the C treatments contributes significantly to the regression model for the k-th response variable. A proper p-value can be computed from F distribution, which is a valid reference distribution for the test statistic under normality assumption. The suitable test statistic has the following form:

$$F = \frac{\sum_{i=1}^{n}\sum_{j=2}^{C}(\hat{Y}_{ijk} - \overline{Y}_k)^2 / (C+q-1)}{\sum_{i=1}^{n}\sum_{j=2}^{C}(Y_{ijk} - \hat{Y}_{ijk})^2 / [(n \cdot C) - C + q]} \sim F_{(\alpha,[C+q-1],[(n \cdot C)-C+q])}. \quad (6)$$

When the test described above would result in rejection of the null hypothesis, i.e. there is at least one significantly different treatment, we are interested in analyzing the individual levels of treatment through a procedure of pairwise comparisons. For this goal, we consider the following pairwise hypothesis testing layout:

$$\begin{cases} H_{0k(jh)} : \beta_{1jk} = \beta_{1hk} \\ H_{1k(jh)} : \beta_{1jk} \neq \beta_{1hk} \end{cases}, j,h = 1,...,C, j \neq h, k = 1,..,p \quad . \quad (7)$$

When assuming normality of random errors, solution to this problem may be find out within the parametric approach via t and F tests. For the set of the first C–1 pairwise comparisons with the baseline treatment level, we may refer to the Student's t statistic,

$$t_{jk} = \frac{\hat{\beta}_{jk}}{se(\hat{\beta}_{jk})} \sim t_{n \cdot C - C + q}, j=2,\ldots \text{C}, k=1,\ldots,p, \quad (8)$$

whereas for the remaining pairwise comparisons we may refer to the well-know approach for testing on constrained linear models. For this goal, let us formalize the problem to estimate the parameters under a linear constraint that is

$$R'\boldsymbol{\beta}_{1k} = 0$$

in which R is a vector of known elements and $\boldsymbol{\beta}_{1k} = [\beta_{12k},...,\beta_{1Ck}]$, note that the vector R and the vector $\boldsymbol{\beta}_{1k}$ have the same length. The element of R are all zeros except for the two βs related to the pair of the treatment levels we want to compare.

To estimate the parameters under this constraint we can use the estimator of Lagrange. (for sake of simplicity, we remove indexes from β_{1jk}). In order to achieve this result, we want to minimize the sum of squares of residuals, under the considered constraint, so we have to minimize with respect to β and λ the following formula:

$$L(\beta,\lambda) = (y - Z\beta)'(y - Z\beta) - 2\lambda'(R\beta - r)$$

where λ is a scalar representing the Lagrange multiplier. Indicating with $\hat{\beta}_v$ and $\hat{\lambda}$, the values of β and λ that minimizing $L(\beta,\lambda)$, we obtain:

$$\hat{\beta}_v = (Z'Z)^{-1}Z'y + (Z'Z)^{-1}R'\lambda = \hat{\beta} + (Z'Z)^{-1}R'\lambda =$$
$$= \hat{\beta} - (Z'Z)^{-1}R'[R(Z'Z)^{-1}R']^{-1}(R\hat{\beta} - r)$$

So the sum of squares of residuals with constraint is major than the sum of squares of residues without constraint:

$$\hat{\varepsilon}_v = y - Z\hat{\beta}_v = y - Z\hat{\beta} - Z(\hat{\beta}_v - \hat{\beta}) = \hat{\varepsilon} - Z(\hat{\beta}_v - \hat{\beta})$$

and we obtain:

$$\hat{\varepsilon}_v'\hat{\varepsilon}_v = \hat{\varepsilon}'\hat{\varepsilon} + (\hat{\beta}_v - \hat{\beta})'Z'Z(\hat{\beta}_v - \hat{\beta})$$

Therefore, the appropriate test statistic for the hypothesis $H_{0k(jh)} : R'\boldsymbol{\beta}_{1k} = 0$, that is the two treatment levels are equal, is the following:

$$\frac{\hat{\varepsilon}_v'\hat{\varepsilon}_v - \tilde{\varepsilon}'\tilde{\varepsilon}}{\tilde{\varepsilon}'\tilde{\varepsilon}}[(n\cdot C) - C + q] \sim F_{(1,[(n\cdot C)-C+q])}, \qquad (9)$$

where $\hat{\varepsilon}_v$ is the residuals of the model with the constraints and $\tilde{\varepsilon}$ is the residuals of the model without the linear constraint $R'\boldsymbol{\beta}_{1k} = 0$.

2.2.2 Improving the fitting of the regression model

In the paragraph 2.1 “Statistical Model” we pointed out that the treatment effect may be specified on the first degree part of the polynomial (see formula (2)) or alternatively on a higher polynomial order. This issue leads on the need of performing an empirical investigation on real datasets in order to find out case by case the more effective parameterisation.

Moreover, when considering the fit of a linear model, if linearity fails to hold, even approximately, it is sometimes possible to transform either the independent or dependent variables in the regression model to improve the linearity. Another assumption of linear regression is that the dependent variable should vary approximately normally around its expected value, with the same variance for each possible expected value (this is known as homoskedasticity).

Although normality is not required for least squares estimates of the regression parameters to be meaningful (see Gauss-Markov theorem) confidence intervals and hypothesis tests will have better statistical properties if the dependent variable is approximately normal with respect to its mean, with constant variance. The logarithm and square root transformations are commonly used for positive data, and the multiplicative inverse (reciprocal) transformation can be used for non-zero data (Draper and Smith, 1998).

To drive the choice of the model with the best fit (Draper and Smith, 1998) we may refer on the values taken by two fitting indexes, that is the *adjusted* R^2 and the *AIC* (Akaike Information Criterion),

$$adjusted\ R^2 = 1 - \frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)/(n-k)}{\sum_{i=1}^{n}(Y_i - \overline{Y})^2/(n-1)} = \frac{n-1}{n-k}(1-R^2),$$

$$AIC = -2\log L(\hat{\vartheta}) + 2k,$$

where n is the number of observations, k the number of parameters to be estimated and $\log L(\hat{\vartheta})$ is logarithm of the likelihood function.

We recall that, when applied to non nested models, the use of R^2 index could lead to misleading conclusions.

In Chapter 4, where we will present some analysis performed on real data, we will note that both indexes lead to the choice of the same model which also leads to the construction of tests of hypothesis testing able to discriminate between treatments under investigation.

2.2.3 A nonparametric combination-based permutation solution

As alternative solution of parametric test statistics (8) and (9), let us consider the more general two-way layout reported in expression (1) and (3):

$$\mathbf{Y}_{ij} = \boldsymbol{\mu} + \boldsymbol{\tau}_j(\mathrm{X}_i) + \boldsymbol{\varepsilon}_{ij},\ \ i = 1,...,n,\ j = 1,...,\mathrm{C}. \qquad (10)$$

A first naïf nonparametric solution may be derived simply by considering the residuals of parametric model (4) and then processing those residual using the permutation approach (Pesarin and Salmaso, 2010). In fact, residuals allows us to approximately remove the effect of covariate X, in this way obtaining a sort of replicates which are

not available in our context. So that, the reference framework becomes the one-way MANOVA layout.

A more formal and appropriate combination-based permutation solution for the hypothesis testing problem on the equality of the treatment levels may be derived using a stratified by covariate level permutation strategy (for details on multivariate stratified analysis see Pesarin and Salmaso, 2010). Let us denote by $\mathbf{Y}$ an ($n\times p$) dataset:

$$\mathbf{Y}=[\mathbf{Y}_1,..., \mathbf{Y}_j, ..., \mathbf{Y}_C]'=[\mathbf{Y}_1,\ldots, \mathbf{Y}_i,\ldots, \mathbf{Y}_p],$$

where $\mathbf{Y}_j$, j=1,...,C, (C$\geq$2) represents the j-th $1\times p$ group, and $\mathbf{Y}_k$ is the k-th univariate aspect of $\mathbf{Y}$, k=1,...,p ($p\geq$1); moreover let $\mathbf{Y}_{jk}$ represents the k-th univariate aspect of $\mathbf{Y}_j$.

In the context of NonParametric Combination (NPC) of Dependent Permutation Tests a set of conditions should be jointly satisfied:

i) we suppose that for $\mathbf{Y}=[\mathbf{Y}_1,...,\mathbf{Y}_C]'$ an appropriate probabilistic p-dimensional distribution structure P exists, $P_j\in$F, j=1,...,C, belonging to a (possibly non-specified) family F of non-degenerate probability distributions.

ii) the null hypothesis H_0 of no treatment effect stating $\boldsymbol{\tau}_j(X_i)=\mathbf{0}$, $\forall i,j,$ implies the equality in distribution of the multivariate distribution of the p variables in all C groups and strata:

$$H_0:\left[P_1=\ldots=P_C\right]=\left[\mathbf{Y}_1\overset{d}{=}\ldots\overset{d}{=}\mathbf{Y}_C\right].$$

The null hypothesis H_0 implies the exchangeability of the individual data vector with respect to the groups and within the i-th level of covariate X. Moreover H_0 is supposed to be properly decomposed into $n\times p$ sub-hypotheses H_{0ik}, i=1,...,n, k=1,...,p, each appropriate for partial (univariate) p-th aspect within the i-th stratum (level of the covariate X), thus H_0 (multivariate) is true if all the H_{0ik} (univariate) are jointly true:

$$H_0:\bigcap_{k=1}^{p}\left[\bigcap_{i=1}^{n}Y_{1ik}\overset{d}{=}\ldots\overset{d}{=}Y_{Cik}\right]=\bigcap_{k=1}^{p}\left[\bigcap_{i=1}^{n}H_{0ik}\right].$$

H_0 is called the *global* or *overall null hypothesis*, and H_{0ik} are called *partial null hypotheses*.

iii) The alternative hypothesis H_1 is represented by the union of partial H_{1ik} sub-alternatives:

$$H_1 : \bigcup_{k=1}^{p} \bigcup_{i=1}^{n} H_{1ik} ,$$

so that H_1 is true if at least one of sub-alternatives is true.

In this context, H_1 is called the *global* or *overall alternative*, and H_{1ik}, are called *partial alternatives.*

iv) let $\mathbf{T}=\mathbf{T}(\mathbf{Y})$ represent a $n \times p$-dimensional vector of test statistics, whose components T_{ik}, represent the univariate and non-degenerate *partial test* appropriate for testing the sub-hypothesis H_{0ik} against H_{1ik}. Without loss of generality, all partial tests are assumed to be marginally unbiased, consistent and significant for large values (for more details see Pesarin and Salmaso, 2010).

At this point, in order to test the global null hypothesis H_0, the key idea comes from the partial (univariate) tests which are focused on $n \times p$ partial aspects, and then, combining them with an appropriate combining function, from a global (multivariate) test which is referred to as the global null hypothesis.

However, before introducing the combination methodology, we should observe that in most real problems, when the sample size is large enough, there is a clash over the problem of computational difficulties in calculating the conditional permutation space. This means that it is not possible to calculate the exact p-value of observed statistic T_{ik0}. This is brilliantly overcome by using the CMCP (Conditional Monte Carlo Procedure).

The CMCP on the pooled data set X is a random simulation of all possible permutations of the same data under H_0 (for more details refer to Pesarin and Salmaso, 2010). Hence, in order to obtain an estimate of the permutation distribution under H_0 of all test statistics, a CMCP can be used. Every resampling without replacement $\mathbf{Y}^*$ from the pooled data set $\mathbf{Y}$ actually consists of a random attribution of individual data vectors to the C groups. In every $\mathbf{Y}_r^*$ resampling, $r=1,...,B$, the k partial tests are calculated to obtain the set of values $[T_{ikr}^*=T(Y_{ikr}^*),\ i=1,...,n,\ k=1,...,p,\ r=1,\ldots,B]$, from the B independent random resamplings.

It should be emphasized that CMCP only considers permutations of individual data vectors, so that all underlying dependence relations which are present in the component variables are preserved. From this point of view, the CMCP is essentially a multivariate procedure.

Once we have defined the hypotheses system and an appropriate set of $n \times p$ statistics $T_{ik}=T_{ik}(Y_{ik})$, the natural way to test the global null hypothesis consists of two sequential phases:

1. performing $n \times p$ partial tests;
2. combing them in a second-order global test.

Assuming that the partial tests have real values and are marginally unbiased, consistent and significant for large values, then the first phase consists in:

1.a calculating the $n \times p$-vector of observed values of test statistics $\mathbf{T}_0$:

$$\mathbf{T}_0=\mathbf{T}(\mathbf{Y})=[T_{ik0}(Y_{ik}),\ i=1,...,n,\ k=1,...,p];$$

1.b considering a data permutation of $\mathbf{Y}$ by a random resampling $\mathbf{Y}_r^*$, in order to randomly assign every individual data vector to a proper group and then calculate the vector statistics $\mathbf{T}_r^*$:

$$\mathbf{T}_r^* = \mathbf{T}_r^*(\mathbf{Y}_r^*) =[T_{ikr}^*(Y_{ikr}^*),\ i=1,...,n,\ k=1,...,p];$$

1.c carrying out B independent repetitions of step 1.b; the result is a set $\mathbf{T}^*$ of $B \times (n \times p)$ CMC

$$\mathbf{T}^*=[\mathbf{T}_r^*,\ r=1,\ldots,B]=[\mathbf{T}_1^*,\ldots,\mathbf{T}_r^*,\ldots,\mathbf{T}_B^*]'$$

is thus a random sampling from the permutation $n \times p$-variate distribution of vector test statistics $\mathbf{T}$;

1.d the $n \times p$-variate EDF (Empirical Distribution Function) $\hat{F}_B(\mathbf{z} \mid \mathbf{Y})$

$$\hat{F}_B(\mathbf{z} \mid \mathbf{Y}) = \left[1/2 + \sum\nolimits_r \mathbf{I}(\mathbf{T}_r^* \le \mathbf{z})\right] \Big/ (B+1),\ \forall \mathbf{z} \in \Re^{n \times p},$$

where $\mathbf{I}(\cdot)$ is the indicator function, and gives an estimate of the corresponding $n \times p$-dimensional permutation distribution $F_B(\mathbf{z} \mid \mathbf{Y})$ of $\mathbf{T}$. Moreover

$$\hat{L}_{ik}(z \mid \mathbf{Y}) = \left[1/2 + \sum\nolimits_r \mathbf{I}(T_{ikr}^* \ge z)\right] \Big/ (B+1),$$

gives an estimate $\forall z \in \mathbb{R}^1$ of the marginal permutation *significance level function* $L_{ik}(z \mid \mathbf{Y}) = \Pr\left\{T_{ik}^* \ge z \mid \mathbf{Y}\right\}$; thus

$$\hat{L}_{ik}(T_{ik0} \mid \mathbf{Y}) = \hat{\lambda}_{ik},$$

gives an estimate of the marginal p-value $\lambda_{ik} = \Pr\left\{ T_{ik}^{*} \geq T_{ik0} \mid \mathbf{Y} \right\}$ relative to test T_{ik}, i=1,...,n, k=1,...,p. All these are unbiased and consistent estimates of corresponding true values;

1.e if $\hat{\lambda}_{ik} < \alpha$, the null hypothesis H_{0ik} relating to the p-th variable on the i-th stratum (level of covariate X) is rejected at the significance level α.

The second phase, based on a nonparametric combination of the dependent tests previously obtained, consists in the following steps:

2.a the combined observed value of the second-order test is evaluated through the same CMC results as the first phase, and is given by:

$$T_0'' = \psi(\hat{\lambda}_1, ..., \hat{\lambda}_{n \times p});$$

2.b the r-th combined value of vector statistics (step 1.d) is then calculated by:

$$T_r''^{*} = \psi(\hat{\lambda}_{1r}^{*}, ..., \hat{\lambda}_{(n \times p)r}^{*}),$$

where $\lambda_{ikr}^{*} = \hat{L}_{ik}(T_{ikr}^{*} \mid \mathbf{Y})$, i = 1,…,k, r =1,…,B;

2.c hence, the p-value of combined test T'' is estimated as:

$$\lambda_{\psi}'' = \sum_{r} \mathbf{I}\left(T_r''^{*} \geq T_0'' \right) \Big/ B;$$

2.d if $\lambda_{\psi}'' \leq \alpha$, the global null hypothesis H_0 is rejected at significant level α;

where ψ is an appropriate combining function.

Remember that, in order to preserve the underlying dependence relations among variables, permutations must always be carried out on individual data vectors, so that all component variables and partial tests must be jointly analyzed.

It can be seen that under the general null hypothesis the CMC procedure allows a consistent estimation of the permutation distributions, both marginal and combined, of the k partial tests. In the nonparametric combination procedure, Fisher's combination function is usually considered, principally for its good properties which are both finite and asymptotic (Pesarin and Salmaso, 2010). Of course, if it were considered appropriate, it would be possible to take into consideration any other combining function (Folks, 1984; Pesarin and Salmaso, 2010). The combined test is unbiased and consistent; also it has interesting asymptotic properties.

A general characterization of the class of combining functions is given by the following three main features for the combining function ψ:

a) it must be non-increasing in each argument:

$$\psi(...,\lambda_i,...) \geq \psi(...,\lambda_i',...) \; if \; \lambda_i < \lambda_i' \, , \; i \in \{1,\ldots,k\};$$

b) it must attain its supreme value $\overline{\psi}$, possibly non finite, even when only one argument reaches zero:

$$\psi(...,\lambda_i,...) \to \overline{\psi} \; if \; \lambda_i \to 0 \, , \; i \in \{1,\ldots,k\};$$

c) $\forall \alpha > 0$, the critical value of every ψ is assumed to be finite and strictly smaller than the supreme value:

$$T''_\alpha < \overline{\psi} \, .$$

The above properties define the class *C* of combining functions. Some of the functions most often used to combine independent tests (Fisher, Lancaster, Liptak, Tippett, Mahalanobis, etc.) are included in this class. If in the overall analysis, distinguishing the importance of partial tests by using appropriate weights opportunely fixed: $w_i \geq 0$, $i = 1,..,k$, with at least one strong inequality is considered more suitable, then the combined test using the Fisher combination is:

$$T'' = -\sum_i w_i \cdot \log(\lambda_i) \, .$$

2.3 Global ranking methods

Since the focus of this work is not only on hypothesis testing on the treatment effect for each univariate response variable, but also on defining and estimating a suitable multivariate indicator to quantify the relative preference of each treatment in comparison with each other (in order to rank them), let us consider a so-called "ranking parameter" $\vartheta_j = f_j(\boldsymbol{\mu}_1,....,\boldsymbol{\mu}_C), \; j = 1,...,C$ (details on the concept of "ranking parameter" can be found in Gupta and Panchapakesan, 2002), such that the rank transformation of ϑ_j may be able to provide a meaningful ranking of the treatment *j*-th from a multivariate point of view.

It is worth noting that the choice of the functions $f_j(.)$ is particularly sensitive, in fact it represents the way in which the data dimensionality is reduced. So in general:

- we can not think of an optimal solution because it depends on the unknown underlying data structure;

- goodness of $f_j(.)$ depends on the underlying metric, geometrical functions can be used only with continuous r.v.s whereas in case of categorical r.v.s it is more appropriate to use a more robust approach (such as goodness-of-fit functions).

Examples of ranking parameters are:

- Euclidean distance: $\vartheta_j^{dist} = \left\| \boldsymbol{\mu}_j - \boldsymbol{\mu}_0 \right\|$;
- squared Mahalanobis distance: $\vartheta_j^{Mah} = (\boldsymbol{\mu}_j - \boldsymbol{\mu}_0)' \Sigma^{-1} (\boldsymbol{\mu}_j - \boldsymbol{\mu}_0)$;

where $\boldsymbol{\mu}_0$ is a known reference p-dimensional point, for example the minimum or maximum value that can be reached by the response variable. Mahalanobis distance differs from Euclidean distance in that it takes into account the correlations of the dataset and is scale-invariant, i.e. it is not dependent on the scale of measurements. For this reasons, the Mahalanobis distance is often preferred with respect to the Euclidean distance. Note that we are implicitly assuming that all response variables are defined in the same metric and this is true in many real cases of interest.

By combination of the p-values directly related to the set of original univariate testing procedures (see expression (7)), a robust and even more informative ranking parameter can be defined:

- NPC score: $\vartheta_j^{NPC} = -2 \sum_{k=1}^{p} \sum_{\substack{h=1 \\ h \neq j}}^{C} \log p_{jh|k}$,

where $p_{jh|k}$ is a p-value suitable for testing the hypothesis $H_{0k(jh)} : \beta_{1jk} = \beta_{1hk}$. Note that the NPC score is actually the so-called nonparametric Fisher combining function (see Pesarin and Salmaso, 2010), often used to derive multivariate testing procedures. Note that the NPC score depends on the test statistics involved in it and it is a function not only of all the true means but also of the unknown dependence structure of the multivariate random errors. In order to make the NPC score more informative with respect to our goal of ranking the multivariate treatments, we can take account of directional type p-values, namely those that are suitable for testing the hypotheses:

$$\begin{cases} H_{0k(jh)} : \beta_{1jk} \leq \beta_{1hk} \\ H_{1k(jh)} : \beta_{1jk} > \beta_{1hk} \end{cases} \qquad (11)$$

$$i \neq h, \quad i,h = 1,....,C, \quad j = 1,....,p .$$

Note that, similarly to the distance-based ranking parameters, the NPC score has the following characteristics: (i) it takes values greater (or equal) to zero and tends to take lower values when the hypothesis of equality of treatments is true; (ii) conversely, it tends to take large values under the alternative hypothesis of difference between treatments. The fact of using directional *p*-values helps us to make the NPC score more suitable to our problems and to better discriminate treatments in order to obtain a meaningful ranking of them.

In order to achieve the objective of finding out a ranking of multivariate populations, let us now rewrite the inferential problem in terms of ranking parameters:

$$\begin{cases} H_0 : \boldsymbol{\beta}_{11} = \ldots = \boldsymbol{\beta}_{1C} \\ H_1 : \exists\, j,h \mid \boldsymbol{\beta}_{1j} \neq \boldsymbol{\beta}_{1h} \end{cases} \rightarrow \qquad \begin{cases} {}^{\vartheta}H_0 : \vartheta_1 = \ldots = \vartheta_C \\ {}^{\vartheta}H_1 : \exists\, j,h \mid \vartheta_j = \vartheta_h \end{cases} \tag{12}$$

Then, if ${}^{\vartheta}H_0$ is rejected we may consider the testing procedure on pairwise ranking parameters, that is

$$\begin{cases} {}^{\vartheta}H_0^{jh} : \vartheta_j = \vartheta_h \\ {}^{\vartheta}H_1^{jh} : \vartheta_j \neq \vartheta_h \end{cases} \qquad 1 \leq j < h \leq C\,, \tag{13}$$

where for the univariate test statistics in the NPC methods we use directional *p*-values as it is described on the previous page. Note that the two approaches of testing of hypotheses can not be considered equivalent, but in this context we are more interested in estimation and ranking than in hypothesis testing, provided that in this conversion the "lack of information" is as small as possible. In fact, if the global null hypothesis on the original parameters $\boldsymbol{\beta}_{1}$s is true, then the global null hypothesis on the ranking parameters is also true, but in general the *vice versa* does not always hold; this is due to the reduction of dimensionality operated by the synthesis functions $f(.)s$ in which two treatments could differ in opposite direction in two different variables, hiding these differences on the global testing side. Nevertheless we will base on the hypothesis testing (11) to justify one of the two *p*-value calculation way.

So our multivariate inferential problem of interest can be viewed as a simultaneously interval estimation procedure on the differences $\vartheta_{jh}^{T} = \vartheta_j^{T} - \vartheta_h^{T} \quad 1 \leq j < h \leq C$; where T is the type of the ranking parameter we decide to adopt. It is noteworthy that the problem becomes only apparently univariate because the ranking parameter estimator (details below) depends on the multivariate distribution of the error components ε s,

even more with reference to the NPC score ranking parameter, where a single ϑ_j depends on all the comparisons involving the *j*-th treatment in all variables at the same time.

Now we present two nonparametric permutation-based methods. Both of them consist in two main steps that are described below.

1. Choose a suitable score-statistic (i.e. ranking parameter) that summarise the "position" of each treatment (on the metric of the statistic). This results in a C-dimensional vector of scores.
2. Construct the confidence intervals for the pairwise differences of ranking parameters $\hat{\vartheta}_{jh}$, as described in the subsequent section.
3. Use these scores to test the C*(C-1)/2 pairwise comparisons hypotheses (13) on page 26. This results in a C*C zero-one matrix for the rank assignment rule.

Because of their complexity, from a parametric point of view, the distribution of these ranking parameters will be obtained *via* permutation.

In order to calculate the score-statistics, called *NPC* (NonParametric Combination), let us consider the *k*-th response variable. We used two different approaches to obtain *p*-values that have to be combined with the Fisher's combining function.

In the first approach we have considered parametric *p*-values, performing the following steps:

1. the *t*-test statistics is calculated for each of the C*(C-1)/2 pairwise comparisons between two treatments, formally:

$$t_{jk} = \frac{\hat{\beta}_{jk}}{se(\hat{\beta}_{jk})} \sim t_{n \cdot C - C + q}\,, j=1,\ldots C,\ k=1,\ldots,p;$$

   where $se(\hat{\beta}_{jk})$ is the estimated standard error of the estimator β_{jk}

2. *p*-values are then calculated.

In the second approach we have considered permutation *p*-values (so the reference null distribution were the permutation one), performing the following steps:

- the statistic to be calculated for each of the *C**(*C*-1)/2 pairwise comparisons between two treatments is:

$$T_{jh|k} = \bar{y}_{j|k} - \bar{y}_{h|k},\ \ 1 \le j < h \le C\,;$$

where $\bar{y}_{j|k}$ is the sample mean of the *j*-th treatment in the *k*-th variable;

- *p*-values are then calculated as

$$\frac{1}{B}\#({}^{b}T_{jh|k} \geq T_{jh|k}) = \frac{1}{B}\#[({}^{b}\bar{y}_{j|k} - {}^{b}\bar{y}_{h|k}) \geq (\bar{y}_{j|k} - \bar{y}_{h|k})];$$

where the superscript *b* indicate that the statistics is calculated using the *b*-th permuted sample, #(.) is the function that count the number of elements of the set that satisfies the condition (here the set is the *B* values of the statistic) and the statistic without superscript is the observed one.

Thus each treatment is matched against all others as if we were to test the hypotheses (11) on page 25. These steps has to be repeated for every variable obtaining a set of $p^* C^* (C - 1)/2$ *p*-values.

Finally the global score for the *j*-th treatment is calculated with:

$$\vartheta_j^{NPC} = -2\sum_{k=1}^{p}\sum_{\substack{h=1 \\ h>j}}^{C} \log p_{jh|k} ,$$

where we use as *p*-values synthesis criterion the Fisher's combining function. It is noteworthy that this combining function is nonparametric with respect to the underlying dependence structure among *p*-values, in that all kinds of monotonic dependencies are implicitly captured (Pesarin and Salmaso, 2010). The distribution of this statistics would be $X^2_{2p^*(C^*(C-1))/2}$ if all partial hypotheses involved were true and the terms of the summation were independent, but this is not the case. The *p*-values can not be considered independent because of the common denominator in the parametric *p*-value calculation, and because of the linear relation between $T_{j|k}$ s, in the permutation *p*-value calculation.

After obtaining the permutation distributions of the considered ranking parameters, in order to test the pairwise hypotheses, we decided to construct the $K=C^*(C-1)/2$ simultaneous confidence intervals for pairwise differences of scores. So the α (significance level) was corrected for multiplicity with Bonferroni's method, i.e. $\alpha' = \alpha / K$. Then, after this, a "one" was associated to all comparisons for which the confidence interval contains the origin and a "zero" elsewhere, producing the C^*C zero-one matrix for the ranking rule.

It is worth to describe here how the issue of constructing confidence intervals just mentioned was solved. The simplest way to build these confidence intervals is to

directly use the bootstrap or permutation distribution, taking the sample $\alpha'/2$ and $1-\alpha'/2$ quantiles as estimates of true quantiles, in this way a proportion of α' observations are excluded from the interval and the confidence level is theoretically satisfied. Problems arises when α' is too small with respect to the sample size, as it is in our case (even though there are B = 1000 bootstrap or permutation replications), because estimates become inaccurate due to the small probability associated with the binomial distribution of the estimators (it is based on the empirical c.d.f. statistics). To get rid of this limitations and following the ideas from Hinkley (1975) we decided to use the *unconditional* version of the Box-Cox's transformation (that is, there is no regression model), appeared in Box and Cox (1964) and further explored by Draper and Cox (1969) which expression is:

$$z_\lambda := \begin{cases} \dfrac{y^\lambda - 1}{\lambda}, & if\ \lambda \neq 0 \\ \log y, & if\ \lambda = 0 \end{cases},$$

and it is monotonic for every fixed "λ", which is found with a numerical maximisation. From interval estimates of ϑ_{jh} (or equivalently observed p-value related to H_0^{ij}) it is desirable to define a suitable algorithm able to estimate the multivariate ranking of the C treatments. In fact, only in the few cases when all differences are declared significant, it would be easy to find out a meaningful ranking but, since a sort of transitive property of significant differences obviously does not exist, we need a general rule able to assign a ranking to the multivariate treatments. All the procedures that will be presented here have a common outcome: a zero-one *C*C* matrix where the (*j,h*)-th cell over the main diagonal take the value "one" if the (*j,h*)-th pairwise null hypothesis is not rejected and "zero" elsewhere (hence if the two treatments can not be considered as equal), "ones" on the main diagonal (every treatment is always equal to itself), while cells under the main diagonal can be considered as *N.A.*s. Note that treatments have been ordered from the highest ("best") to the lowest ("worst") according to the point estimates (ϑ_j s), before calculating pairwise comparisons (and hence before constructing the matrix), thus the first row contains the comparisons between the best treatment against each other, and so on.

Starting from this matrix the ranking rule can be described as follows:

a. row 1 is multiplied by 1, so the rank 1 is assigned to all treatments that are non-significantly different from (1) ("best"), including (1) itself;
b. row 2 is multiplied by 2, so the rank 2 is assigned to all treatments that are non-significantly different from (2), including (2) itself;
c. the iterated procedures stops when a rank is assigned to all treatments;
d. mean by columns (without considering zeros) provides a synthesis of the rank of each of the *C* treatments, it is a sort mid-rank;
e. finally to obtain the global ranking it is enough to apply the rank transformation where, in the case of ties, the minimum value is repeated (this because we used the convention that "the lower the rank, the better the treatment").

For a more detailed presentation of the global ranking methodology see Arboretti et al. (2010).

Chapter 3. Simulation study

This chapter is devoted to study and compare the solutions proposed in Chapter 2 by means of a suitable simulation study in order to validate them and to investigate their behaviours with different functional forms for the univariate response variables. In fact, as displayed in Chapter one, the real case studies show quite different effect patterns which can be sometime simply monotone, or sometime with a maximum/minimum and eventually a flex point as well.

The simulation study is designed in order to meet both the univariate and multivariate inferential goals we set up in this thesis. Specifically, the simulation setting is the following:

- univariate data are generated following one out of three selected different patterns (see Fig. 7, 10, 12) and using a three degree polynomial, where the values of parameters have been suggested by real dataset related to secondary performance experiments for laundry industry (see Fig. 1−6 on Chapter 1);
- C=5 number of treatments;
- n=5 different levels of the covariate X (in order to mimic the typical secondary performance experiment with number of washings equal to 5,10,15,20,25);
- the specific values of the treatment effects are set up on the first polynomial degree (see expression (4), Chapter 2; see Fig. 7, 10, 12) and these values has been calibrated in order to keep approximately fixed the distance among the curves (conventionally, we called this distance as "delta" - δ); moreover, a small normal random effect has been also considered in order to make a bit different the effect parameters with respect to each independent simulation;
- p=3,6,9 number of response variables; the distance δ between treatments has been changed with reference to different response variables in order to represent increasing situation from very small effects till to larger effects;
- random errors are generated by standard normal distribution (note that they are homoscedastic and independent with respect to the p response variables);
- 1000 datasets have been generated independently for each combination of the above settings.

Rejections rates (α=0.05) have been calculated under both the null and the alternative hypothesis by each one of the three univariate solutions proposed in Chapter 2, while for the multivariate ranking problem we calculated the counting cross table between the estimated and the true ranking.

For the sake of simplicity, since the 10 pairs of treatments (recall that we have five treatments) may be classified on the basis of a distance of 1,2,3 or 4 δ, the univariate rejection rates will be presented showing the corresponding rejection rates for each one of these four categories. Note that, as the distance increases we expect an increase in power as well.

In the next paragraph we present the simulation results for univariate analysis while paragraph 3.2 is devoted for results on multivariate analysis.

3.1 Simulation results for univariate analysis

As mentioned earlier, the simulation study considers three different effect patterns:

1. a simple increasing nonlinear monotone curve (see Fig. 7),
2. a concave curve with a minimum point (see Fig. 10),
3. a more complex nonlinear non monotone curve with one maximum, one minimum and a flex point as well (see Fig. 12).

3.1.1 First effect pattern (monotone curve)

The first simulation involves 3 response variables (Y1, Y2, Y3) following a simple increasing nonlinear monotone effect curve with increasing distance among the five treatments (from Y1, see Fig. 8 to Y3, see Fig. 7). For the sake of simplicity we present results only for Y3 and Y1.

Table 1 reports the rejection rates (α=0.05) for testing the univariate hypothesis of equality of pairwise treatments on variable Y3 we obtained using the two proposed approaches. We recall that for the second nonparametric permutation approach, we proposed also a naïf solution based on the residuals from the parametric model.

Rejection rates in Table 1 definitely suggest us that, for the goal of discriminate at univariate level the treatment effect, the *t* and F-based parametric approach is much more powerful than the proposed nonparametric permutation-based approach. For this reason, hereafter we will present results only for the parametric approach.

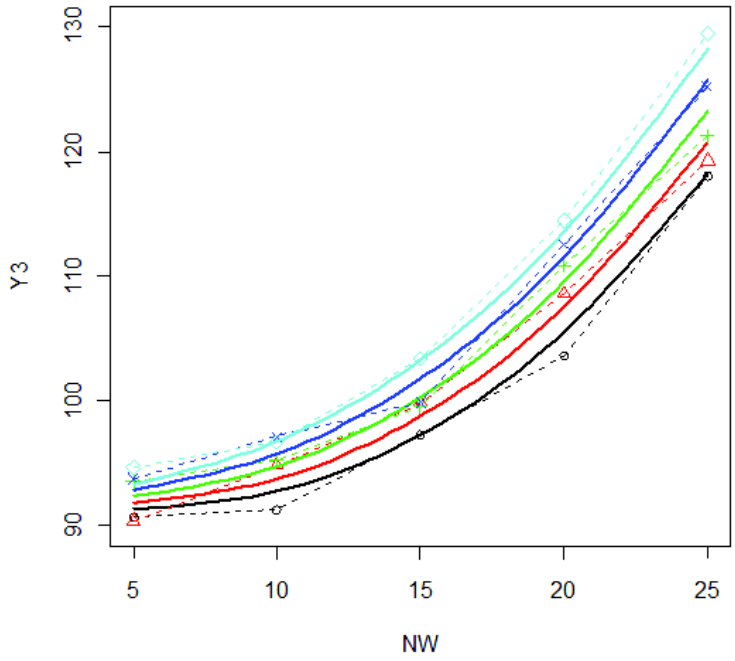

Figure 7. Example of one generated dataset for response Y3 (monotone curve)

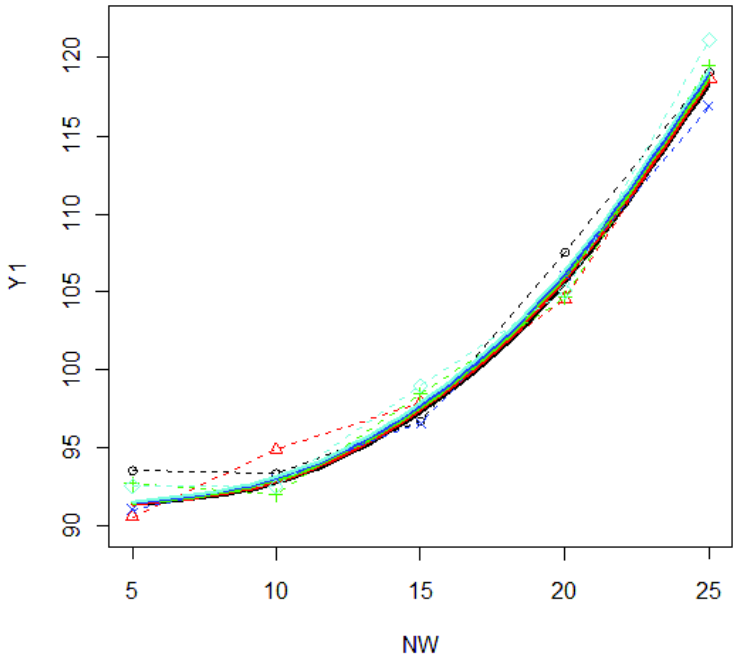

Figure 8. Example of one generated dataset for response Y1 (monotone curve)

Pairwise difference between treatments – δ	parametric	nonparametric	
		permutation	perm. residuals
1 δ	0.6892	0.0512	0.0480
2 δ	0.9983	0.1223	0.1133
3 δ	1.0000	0.8951	0.9552
4 δ	1.0000	1.0000	1.0000

Table 1. Rejection rates for response Y3 by δ and methods (monotone curve)

Table 2 reports the rejection rates (α=0.05) for testing the univariate hypothesis of equality of pairwise treatments on variable Y1 we obtained using the proposed parametric approach. Obviously, since for the response Y1 the five treatments are much closer one each other, the estimated power results on lower values than for the response Y3.

Pairwise difference between treatments – δ	Rejection rate
1 δ	0.0555
2 δ	0.0761
3 δ	0.1115
4 δ	0.1615

Table 2. Rejection rates for response Y1 by δ (monotone curve)

In order to validate the proposed approach, it is necessary that this is also valid under the null hypothesis of equality of all treatments (see Fig. 9). Table 3 reports the rejection rates under the null hypothesis for the two proposed methods with reference to the simple increasing nonlinear monotone curve effect we considered in this section.

Nominal level	parametric	nonparametric	
		permutation	perm. residuals
0.01	0.0129	0.0089	0.0247
0.05	0.0509	0.0611	0.0765
0.10	0.1046	0.1162	0.1312
0.15	0.1577	0.1771	0.1731
0.20	0.2044	0.2413	0.2314

Table 3. Rejection rates by methods, under the null hypothesis (monotone curve)

Note that while the parametric approach shows rejection rates very close to the nominal levels, the nonparametric approach seems to be somewhat anticonservative.

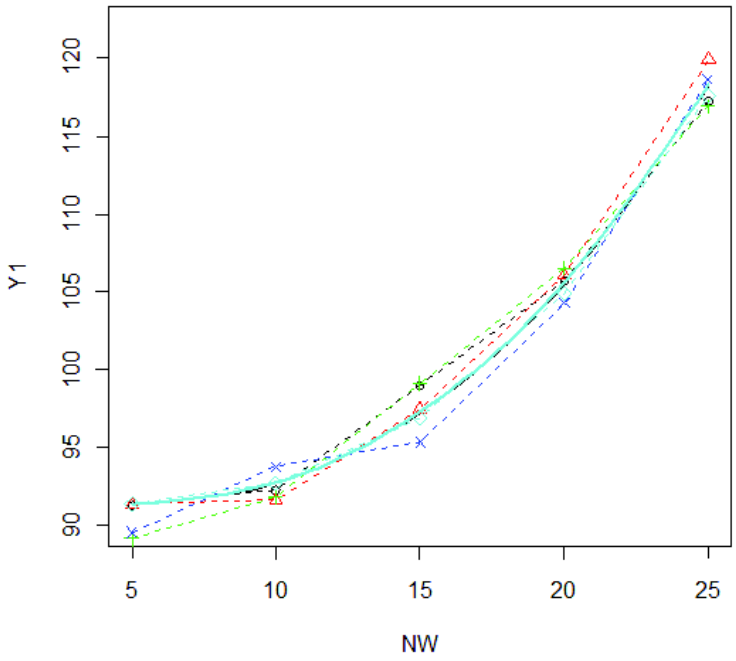

Figure 9. Example of one generated dataset under the null hypothesis (monotone curve)

3.1.2 Second effect pattern (concave curve with min)

The second simulation involves 6 response variables (Y1,…,Y6) following a concave curve with a minimum point with increasing distance among the five treatments (from Y1, see Fig. 10 to Y6). For the sake of simplicity we present results only for Y1, Y4 and Y6.

Table 4, 5 and 6 reports the rejection rates (α=0.05) for testing the univariate hypothesis of equality of pairwise treatments on variable Y1, Y4 and Y6 we obtained using the proposed parametric approach.

Pairwise difference between treatments – δ	Rejection rate
1 δ	0.0543
2 δ	0.0773
3 δ	0.1005
4 δ	0.1582

Table 4. Rejection rates for response Y1 by δ (concave curve)

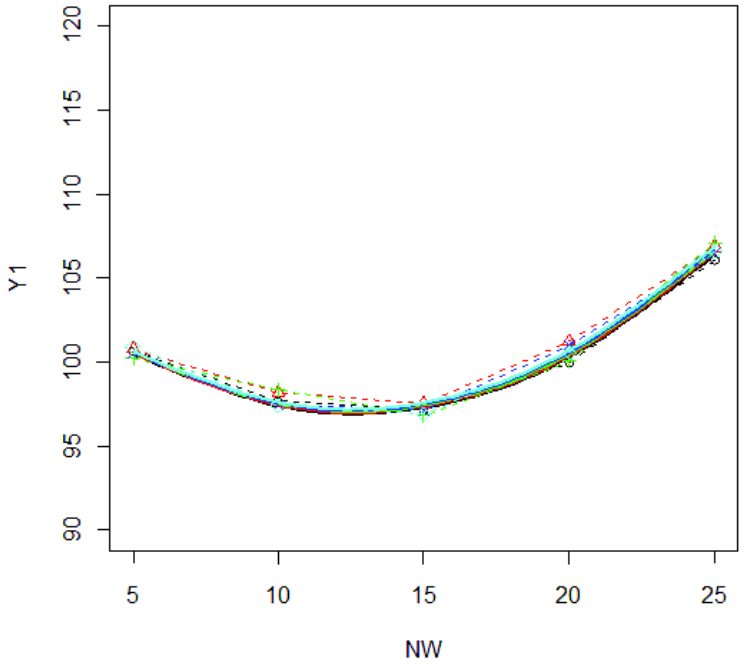

Figure 10. Example of one generated dataset for response Y1 (concave curve)

Pairwise difference between treatments – δ	Rejection rate
1 δ	0.2697
2 δ	0.7996
3 δ	0.9885
4 δ	1.0000

Table 5. Rejection rates for response Y4 by δ (concave curve)

Pairwise difference between treatments – δ	Rejection rate
1 δ	0.8952
2 δ	1.0000
3 δ	1.0000
4 δ	1.0000

Table 6. Rejection rates for response Y6 by δ (concave curve)

In order to validate the proposed approach under the null hypothesis of equality of all treatments (see Fig. 11) also in case of a non monotone pattern, we report on Table 7 the rejection rates we obtained under the null hypothesis.

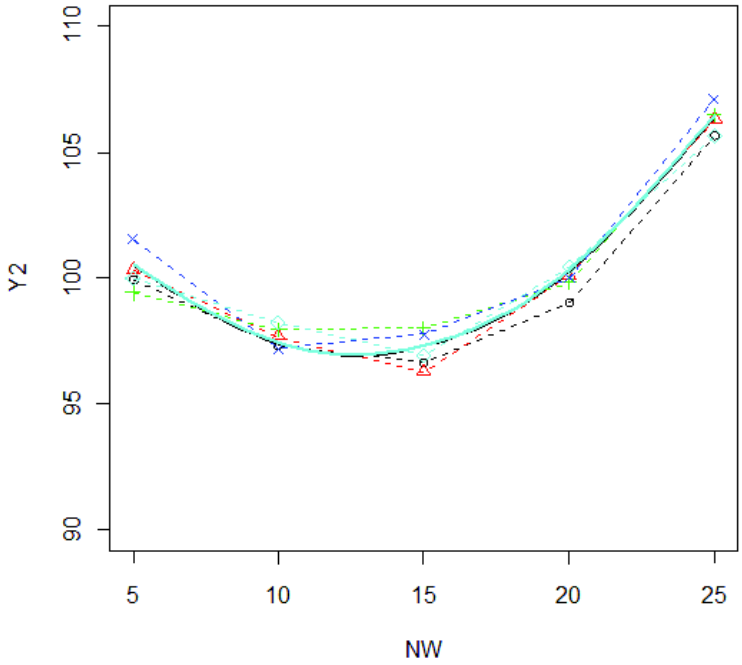

Figure 11. Example of one generated dataset under the null hypothesis (concave curve)

Nominal level	Rejection rates
0.01	0.0122
0.05	0.0481
0.10	0.0954
0.15	0.1447
0.20	0.1929

Table 7. Rejection rates under the null hypothesis (concave curve)

3.1.3 Third effect pattern (non monotone curve with max/min)

The third simulation involves 9 response variables (Y1,…,Y9) following a more complex nonlinear and non monotone curve with one maximum, one minimum and a flex point (see Fig. 12) with increasing distance among the five treatments (from Y1 to Y9). For the sake of simplicity we present results only for Y3, Y5 and Y9.

Table 8, 9 and 10 reports the rejection rates (α=0.05) for testing the univariate hypothesis of equality of pairwise treatments on variable Y3, Y5 and Y9 we obtained using the proposed parametric approach.

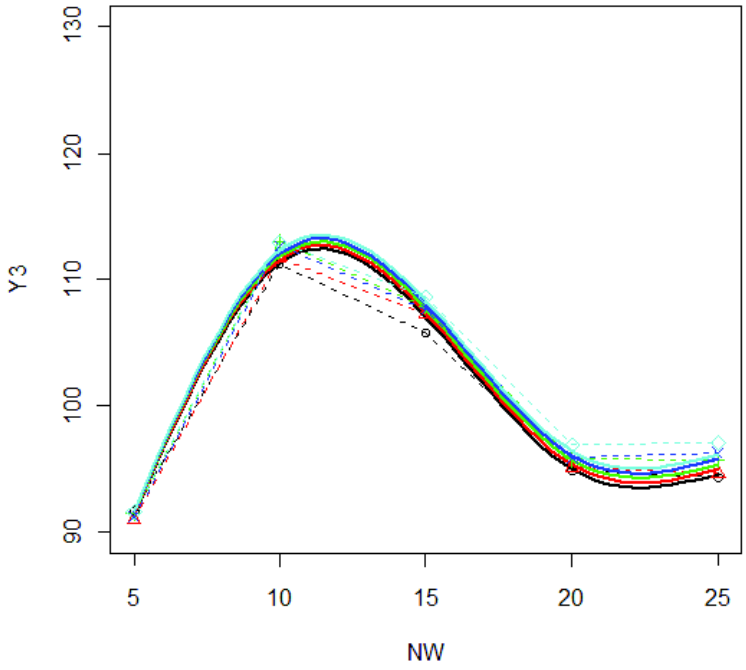

Figure 12. Example of one generated dataset for response Y3 (non monotone curve with max/min)

Pairwise difference between treatments – δ	Rejection rate
1 δ	0.1221
2 δ	0.3786
3 δ	0.7134
4 δ	0.9166

Table 8. Rejection rates for response Y3 by δ (non monotone curve with max/min)

Pairwise difference between treatments – δ	Rejection rate
1 δ	0.3555
2 δ	0.9256
3 δ	0.9995
4 δ	1.0000

Table 9. Rejection rates for response Y5 by δ (non monotone curve with max/min)

Pairwise difference between treatments – δ	Rejection rate
1 δ	0.9672
2 δ	1.0000
3 δ	1.0000
4 δ	1.0000

Table 10. Rejection rates for response Y9 by δ (non monotone curve with max/min)

In order to validate the proposed approach under the null hypothesis of equality of all treatments (see Fig. 13) also in case of a more complex nonlinear and non monotone curve with one maximum, one minimum and a flex point, we report on Table 11 the rejection rates we obtained under the null hypothesis.

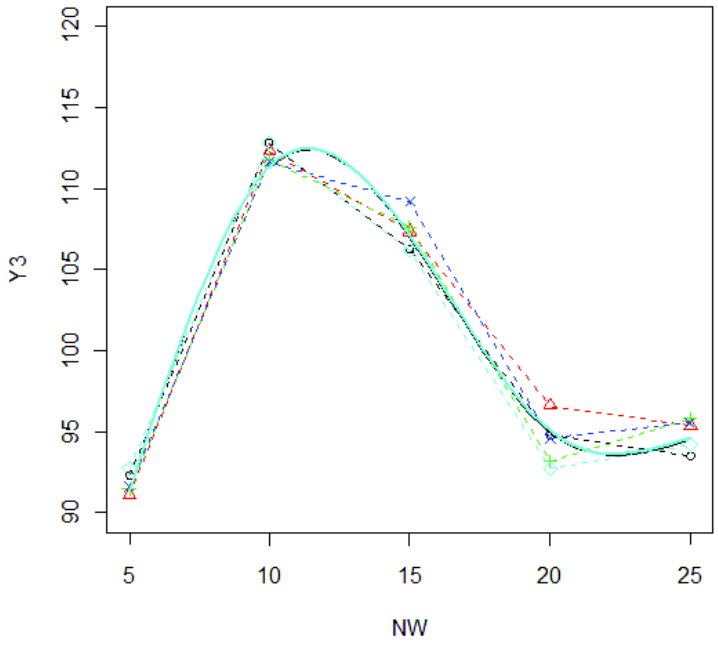

Figure 13. Example of one generated dataset under the null hypothesis (non monotone curve with max/min)

Nominal level	Rejection rates
0.01	0.0110
0.05	0.0492
0.10	0.0974
0.15	0.1484
0.20	0.2146

Table 11. Rejection rates under the null hypothesis (non monotone curve with max/min)

3.2 Simulation results for multivariate analysis (global ranking)

After carrying out simulations for the univariate analysis we proceed with the simulations for the Global Ranking methods. We recall that the simulation study is designed in order to meet both the univariate and multivariate inferential goals we set up in this thesis.

The data used are those considered in Section 3.1, where, however, in this case we are no longer interested in studying a variable independently of each other, but we use the information brought by each response variable in order to obtain a ranking from a multivariate point of view for the five treatments under analysis.

Note that for the first, second and third effect pattern there will be respectively available, three, six and nine response variables from which to summarize the information available for estimating the global ranking.

Despite the global ranking methods we propose in Chapter 3 are three (residual, parametric *p*-values, nonparametric permutation *p*-values), simulations were made using only nonparametric permutation *p*-values. This is because with the hardware at our disposal the simulation for the other two types would take a burden and a very high computational time, about ten days every setting. Conversely, for the simulations with nonparametric permutation *p*-values the time required greatly decreases because it takes a total of approximately 40 hours.

3.2.1 First effect pattern (monotone curve)

First of al we report results of global ranking analysis for the first effect pattern, i.e. where the tree response variables (Y1, Y2, Y3) follow a simple increasing nonlinear monotone effect curve.

In this framework, simulation results are presented via a counting cross table between the estimated and the true ranking (Table 12). Note that along the diagonal there are the right classifications.

Moreover, it is interesting to consider the exact rank rate, that is the number of times when all treatments are jointly classified in their own true position.

		Estimated ranking				
		1°	2°	3°	4°	5°
True ranking	1°	990	10	0	0	0
	2°	52	948	0	0	0
	3°	4	840	119	37	0
	4°	21	310	111	529	29
	5°	24	25	72	530	349

Table 12. Counting cross table between estimated and true ranking (monotone curve)

From the above cross classification table we note that the first two positions are almost always correctly identified, respectively 990 and 948 cases, whereas the last three positions are sometimes confused with the nearby positions. Finally, the number of times that the estimated ranking agrees with the true ranking is equal to 15.

3.2.2 Second effect pattern (concave curve with min)

The second effect pattern is concerned with six response variables a concave curve with a minimum point. Table 13 displays the counting cross table between the estimated and the true ranking.

		Estimated ranking				
		1°	2°	3°	4°	5°
True ranking	1°	1000	0	0	0	0
	2°	0	1000	0	0	0
	3°	0	799	201	0	0
	4°	0	21	180	799	0
	5°	0	0	112	760	128

Table 13. Counting cross table between estimated and true ranking (concave curve with min)

From Table 13 we note that in this case the first two positions are correctly identified in 100% of cases, while for the last three positions we observe the same phenomenon of confounding mentioned above. As for the number of times that the estimated ranking agrees with the true one, this value increases, the number of correct classifications in fact is equal to 32.

3.2.3 Third effect pattern (non monotone curve with max/min)

Finally, the third effect pattern is related to nine response variables following a more complex nonlinear and non monotone curve with one maximum, one minimum and a flex point. Table 14 displays the counting cross table between the estimated and the true ranking.

		Estimated ranking				
		1°	2°	3°	4°	5°
True ranking	1°	1000	0	0	0	0
	2°	0	1000	0	0	0
	3°	0	941	59	0	0
	4°	0	0	22	978	0
	5°	0	0	21	940	39

Table 14. Counting cross table between estimated and true ranking (non monotone curve with max/min)

From Table 14 we note that in this case the first two positions are correctly identified in 100% of cases and the position number four is properly classified in most cases, i.e. 978 times in 1000, while the positions number 3 and 5 are often confused with the position respectively two and four. As for the number of correct classifications it is equal to 35.

Chapter 4. Applications to real case studies

The present chapter is focused on the application of the solutions proposed and validated in the previous chapters to several real case studies concerned with the evaluation of the so called secondary performance during the process of new detergent development (see section 1.2, page 8). In addition to the more traditional not replicated case studies, the proposed solutions are applied also to the case of experiments with replications, even if these ones are somewhat unusual in the framework of our reference industrial field.

Statistical study starts from the univariate analysis, through the choice of regression model with the best fit, then the calculation of parametric tests for pairwise comparisons between treatments until to multivariate analysis, with which calculate the global ranking that will allow us to establish which is the best treatment among those investigated.

Since the models used are of the type

$$Y_{ijk} = \beta_{0k} + \beta_{1jk} X_i + \sum_{s=2}^{q} \boldsymbol{\beta}_{sk} X_i^s + \varepsilon_{ijk},$$

(for details, see section 2.1, p. 15), they can be differentiated according to two aspects:

1. the degree q of the polynomial, that here we considered on the three values 1,2 and 3;
2. the degree of the polynomial where is set the treatment effect of (β_{1jk}), which in the model described above is just on the first degree; but actually, given that as an alternative parameterisation the treatment effect could be properly defined also on a higher polynomial order, it is interesting to empirically investigate this point; for simplicity, we will refer to “linear”, “quadratic” and “cubic” model respectively each one of the three alternative parameterisation where the treatment effect is alternatively set on the first, second or third polynomial order.

Note that, when crossing the tree values of the polynomial (item 1) by the possibility to set the treatment effect on the first, second or third polynomial order (item 2), we get a total number of 6 alternative models.

In section 2.2.2 “Improving the fitting of the regression model” page 18 we pointed out that in the fitting of linear models to data, the transformations of both independent variables and the variable response may play an important role. In this regard, for each of the six models considered previously we apply all the following transformations:

- no transformation;
- logarithmic transformation for Y (response variable);
- logarithmic transformation for the covariate *number of washing* – NW (independent variable);
- logarithmic transformation for both Y and NW;
- square root transformation for Y;
- square root transformation for NW;
- square root transformation for both Y and NW;
- logarithmic transformation for and square root transformation for NW;
- square root transformation for Y and logarithmic transformation for NW.

Note that we obtain a total number of 6×9=54 possible alternative models, to be compared one each other by means of two indexes, namely adjusted R-square and AIC (see section 2.2, pages 18-19).

4.1 Unreplicated case studies

In this section we analyze some real cases in which, as traditionally happens in the framework of the evaluation of the so called secondary performance during the process of new detergent development, there are no independent replications of the experiment.

4.1.1 Whiteness

In this case study we investigate the so-called whiteness degree on eight different products (P1, ..., P8), where these products are evaluated in five different washing cycles (5,10,15,20,25), and where whiteness measurements are obtained for seven different types of textile (wfk10,wfk20,wfk30,wfk40,wfk50,empa,tecc).

To illustrate the univariate statistical analysis, for the sake of simplicity we consider only the type of textile wfk20. The observed values for this experiment are presented in Fig. 1 of Chapter 1 (p. 10).

The first step is to choose the model that best fits the data in our possession, so we will choose among the 54 considered models the one which shows the best fitting. The choice of this model is based on indexes *adjusted* R^2 (Fig. 14) and *AIC* (Fig. 15).

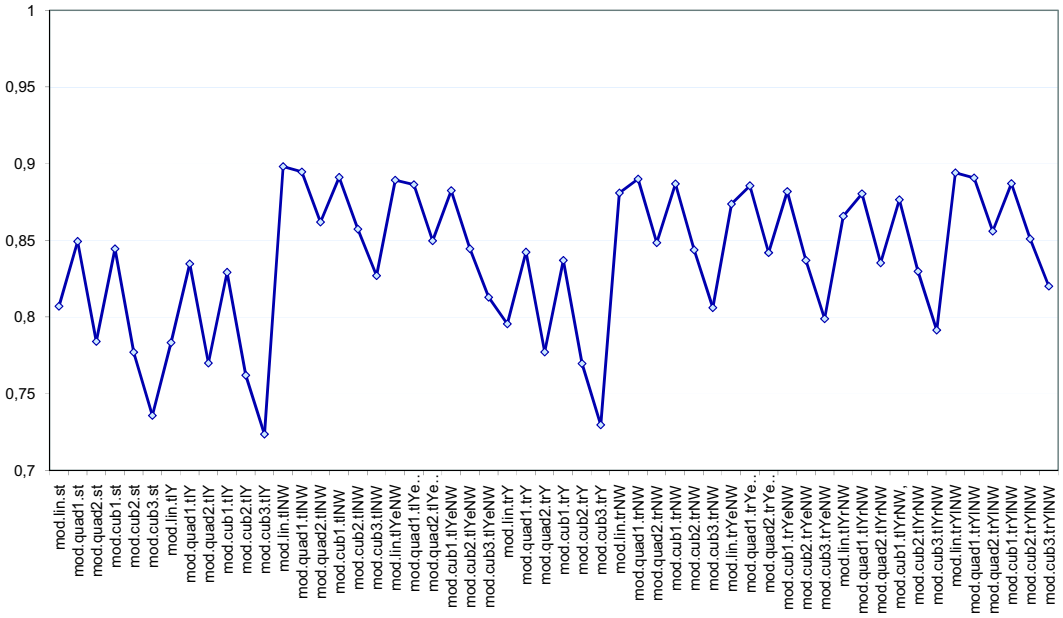

Figure 14. Unreplicated whiteness case study: adjusted R^2 for wfk20 textile

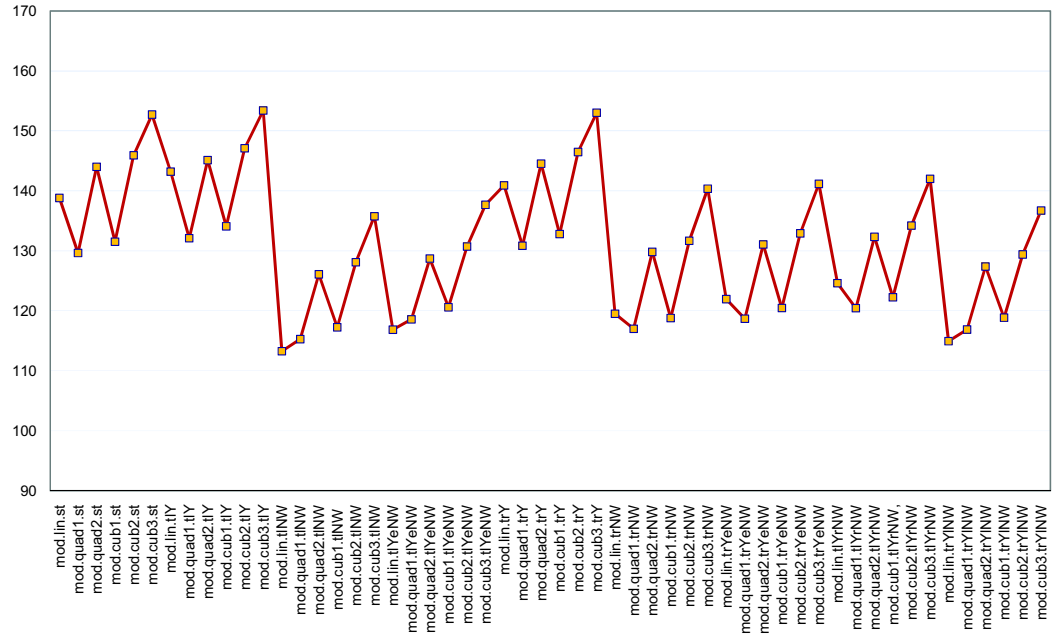

Figure 15. Unreplicated whiteness case study: AIC for wfk20 textile

From the above graphs the model suitable for our purposes is the linear model with log transformation of the covariate NW (number of washing). It is also important to state that the model chosen is also the one that best discriminate among different products (this is found in all the cases we analyzed, hereafter). For a better understanding of this concept, we insert the number of tests for effect pairwise comparisons that lead to rejection (α=0.05), indicating a significant difference between pairs of products.

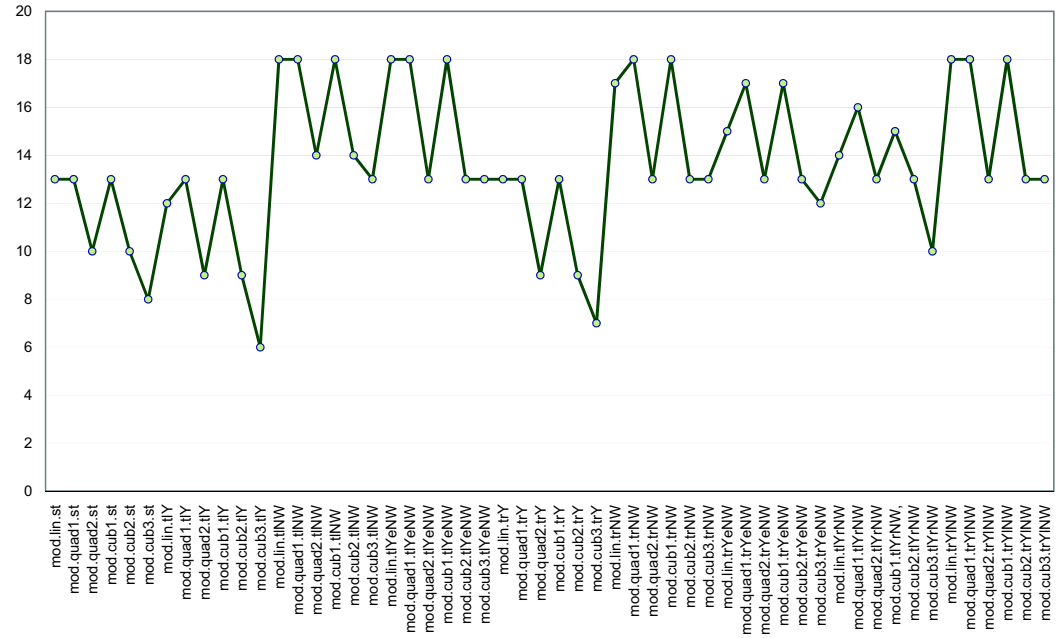

Figure 16. Unreplicated whiteness case study: no. of significant p-values (wfk20 textile)

Table 15 shows the detail of the *p*-values of pairwise comparisons (calculated using the parametric approach), for the same tissue wfk20.

	P2	P3	P4	P5	P6	P7	P8
P1	0.1670	0.0024	0.0238	0.0000	0.0001	0.0000	0.0000
P2	-	0.0001	0.0007	0.0000	0.0000	0.0000	0.0000
P3	-	-	0.3581	0.1445	0.2108	0.0321	0.0286
P4	-	-	-	0.0211	0.0346	0.0034	0.0029
P5	-	-	-	-	0.8279	0.4605	0.4307
P6	-	-	-	-	-	0.3412	0.3167
P7	-	-	-	-	-	-	0.9596

Table 15. Pairwise p-values between pairs of product (wfk20 textile)

The values predicted by the selected model, represented by a continuous line, are shown in Figure 17. They can provide a graphical sketch of the goodness of fit.

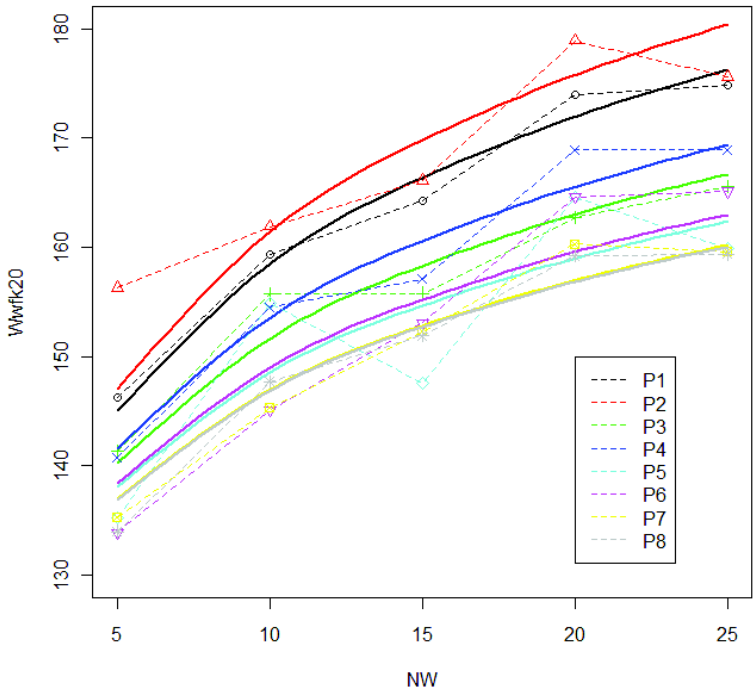

Figure 17. Predicted and observed values for whiteness case study (wfk20 textile)

This procedure was performed for all the remaining six types of textile and the results are not reported for exposition simplicity.

Once we have treated the univariate analysis we carry on with the multivariate analysis, i.e. the calculation of the global rankings to find out which of the eight products is the best with respect to the whiteness degree, considering jointly the performance achieved by the eight products in relation to all types of textile. For this purpose we calculate the global rankings on all three methods described in Section 2.4. From results we may note that the best product for the present case study is that one labelled by P2.

Fig. 18 and 19 show the global ranking and confidence intervals for differences in pairs of ranking parameters obtained through directional *p*-values calculated by permutation tests.

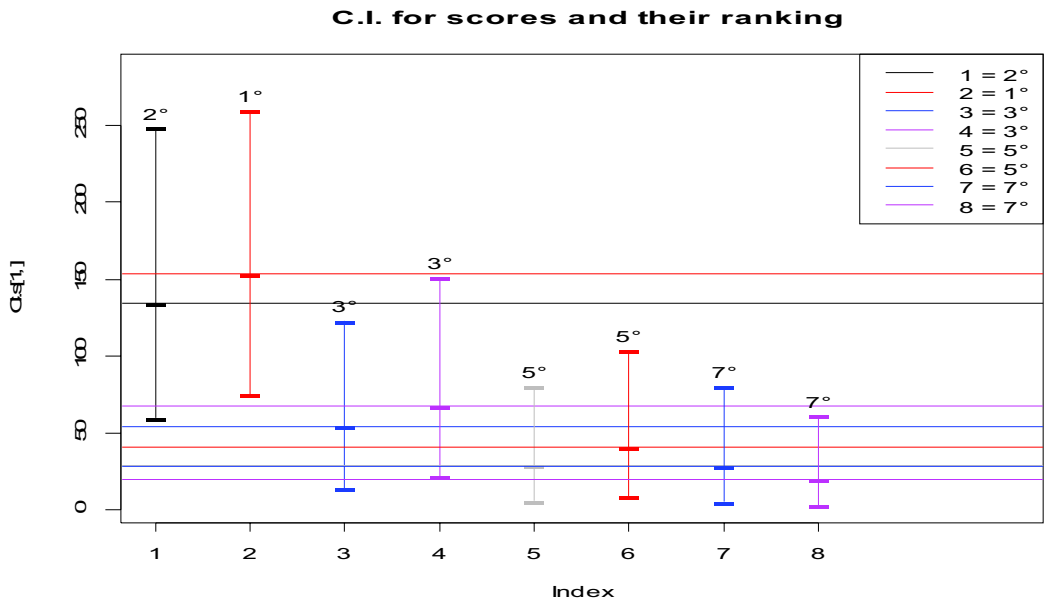

Figure 18. Global ranking for the unreplicated whiteness case study

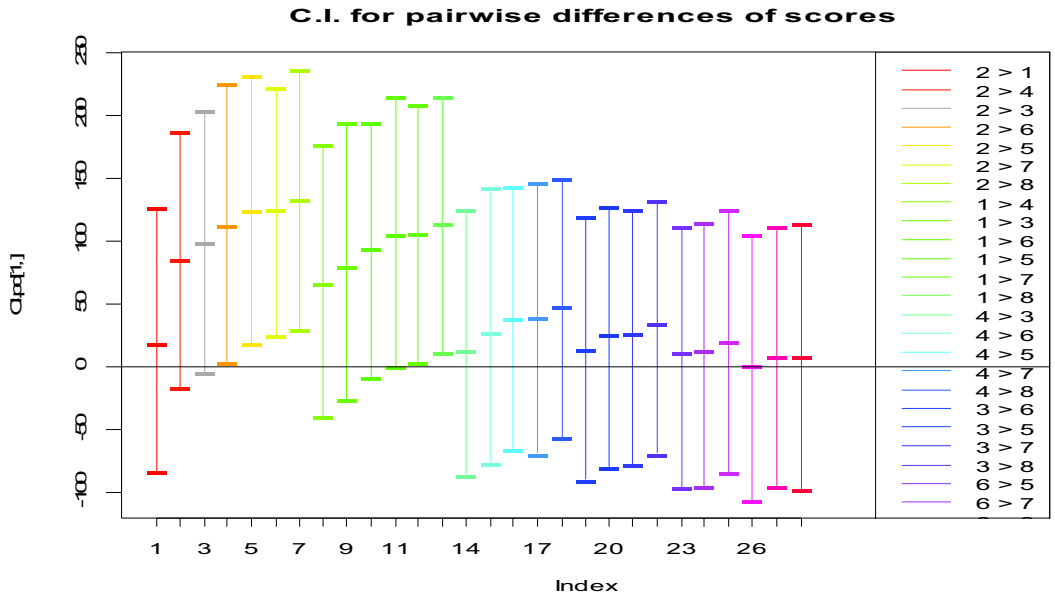

Figure 19. Pairwise differences of ranking parameters for the unreplicated whiteness case study

4.1.2 Redeposition

In the case study on the secondary performance called redeposition, the products under investigation are those considered for the whiteness case study presented in the previous paragraph. In this case we consider the type of textile empa, whose values are shown in

Figure 4 of Chapter 1. As before, first of all we proceed in the choice of the model with the best fitting (Fig. 20 and 21).

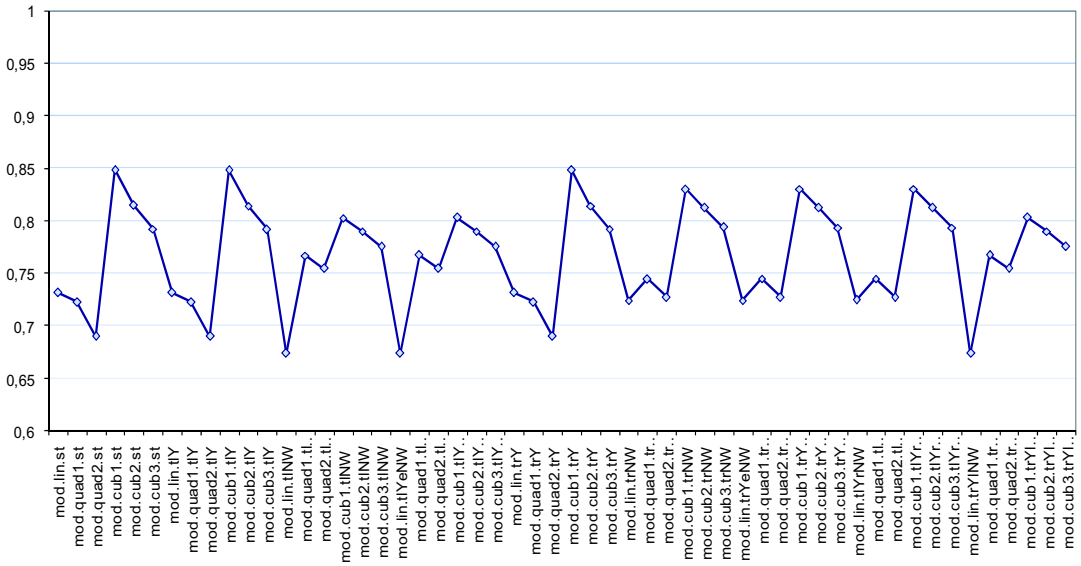

Figure 20. Unreplicated redeposition case study: adjusted R^2 for empa textile

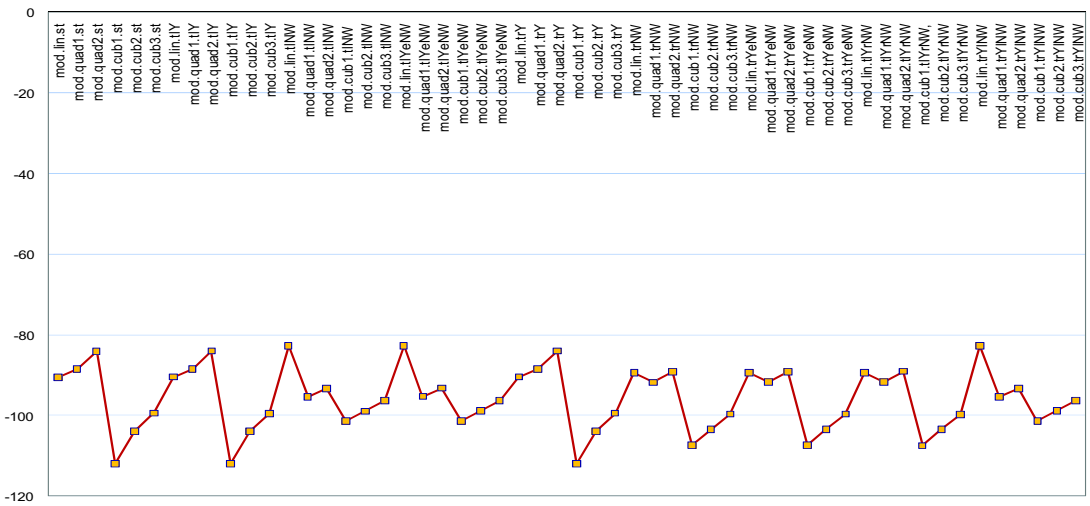

Figure 21. Unreplicated redeposition case study: AIC for empa textile

For the type of textile empa, the selected model is the cubic one without any transformation of variables. Again we see how the chosen model is the one that best discriminate between the different products (Fig. 22). Table 16 shows the detail of the *p*-values of pairwise comparisons and Fig. 23 shows the predicted values.

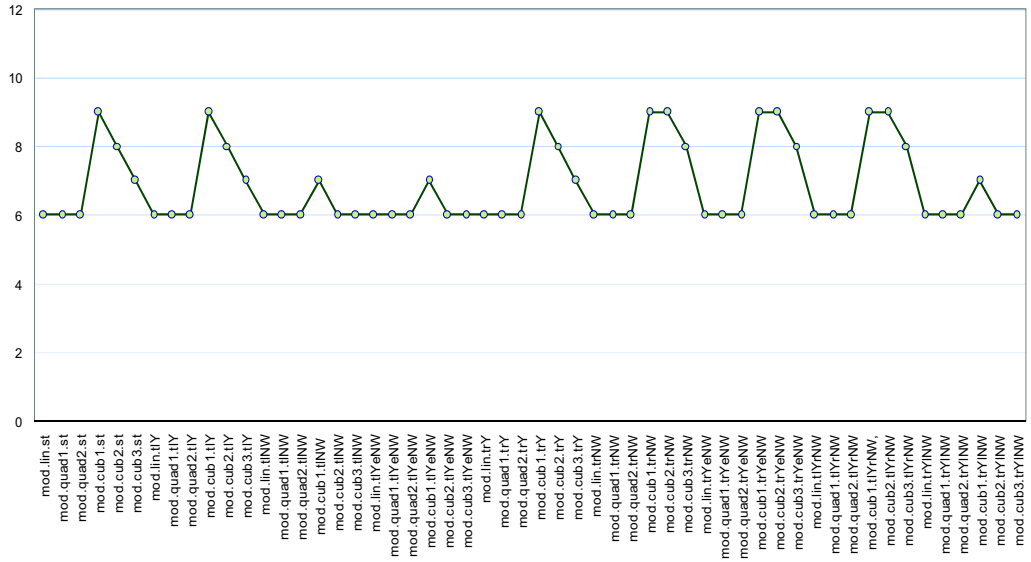

Figure 22. Unreplicated redeposition case study: no. of significant p-values (empa textile)

	P2	P3	P4	P5	P6	P7	P8
P1	0.6990	0.1889	0.1508	0.0026	0.1755	0.0023	0.0007
P2	-	0.0932	0.0722	0.0009	0.0857	0.0008	0.0003
P3	-	-	0.8973	0.0605	0.9657	0.0554	0.0215
P4	-	-	-	0.0787	0.9314	0.0722	0.0288
P5	-	-	-	-	0.0661	0.9657	0.6367
P6	-	-	-	-	-	0.0605	0.0237
P7	-	-	-	-	-	-	0.6675

Table 16. Pairwise p-values between pairs of product (empa textile)

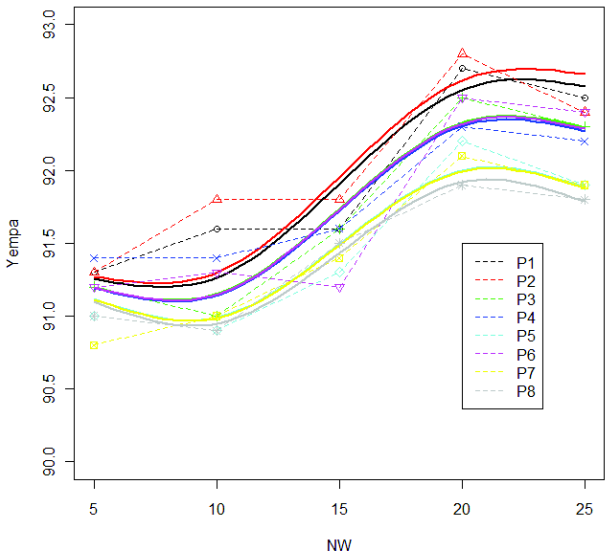

Figure 23. Predicted and observed values for redeposition case study (empa textile)

Again the univariate statistical analysis was performed for all other types of stains used in the experiment and then calculated the global ranking for redeposition secondary performance. Fig. 24 shows the global ranking produced by parametric directional *p*-values (F and *t* test) while Figure 25 shows the estimated pairwise differences for ranking parameters.

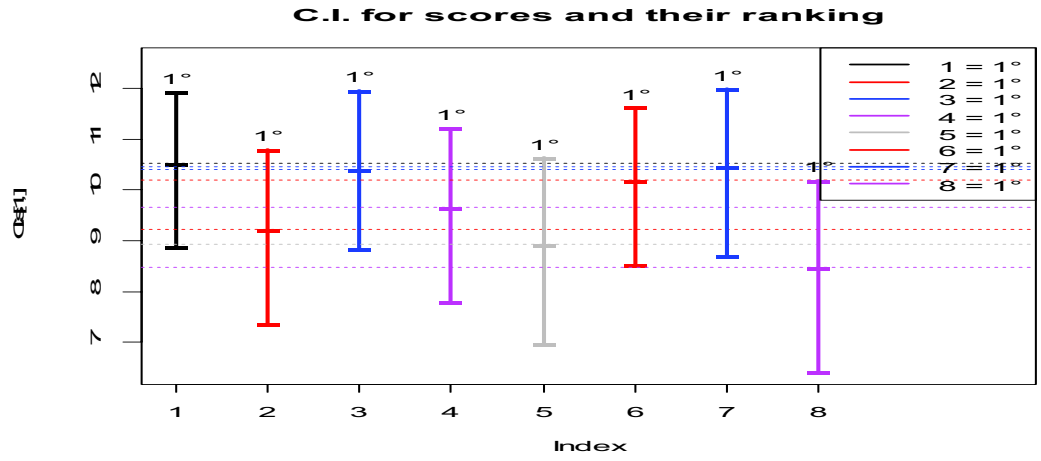

Figure 24. Global ranking for redeposition case study

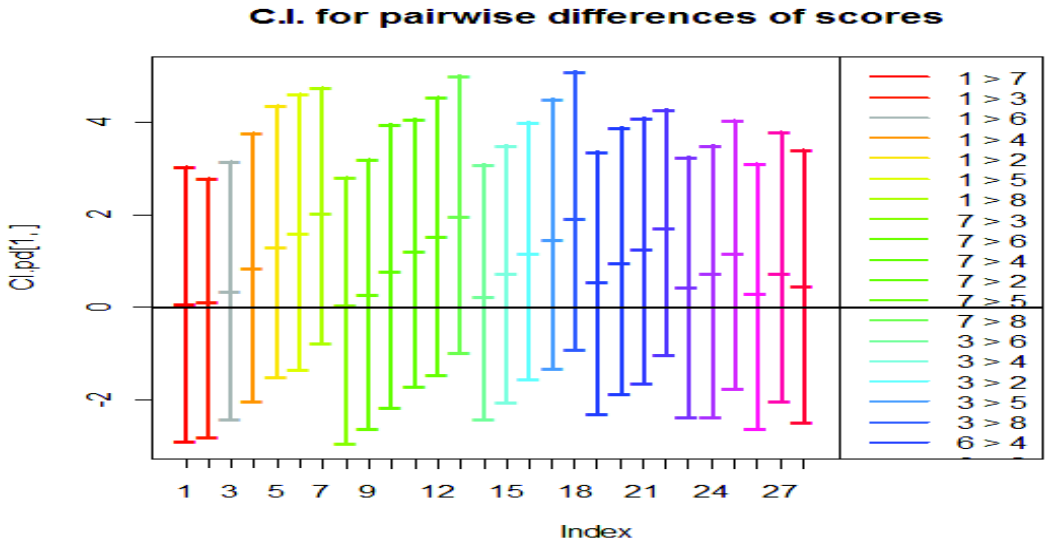

Figure 25. Pairwise differences of ranking parameters for redeposition case study

In this case, the procedure of global ranking indicates that the eight products tested are not significantly different from each other, there is therefore no one best product from the multivariate point of view.

4.1.3 Colour fading

Regards to the case study on secondary performance called colour fading, the number of products under investigation is three, while the number of washing cycles is four (1,5,10,20) and the number of response variables is 14, one for each colour used. To illustrate the univariate analysis we examine the colour C5, whose data are presented in Figure 5 of Chapter 1. As usual, we begin with the choice of model with the best fitting.

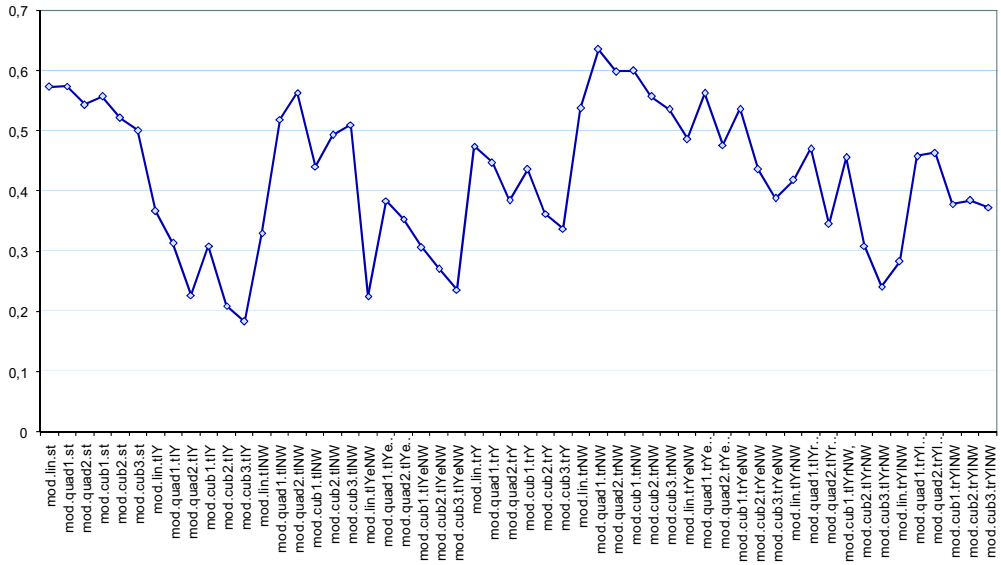

Figure 26. Unreplicated colour fading case study: adjusted R^2 for colour C5

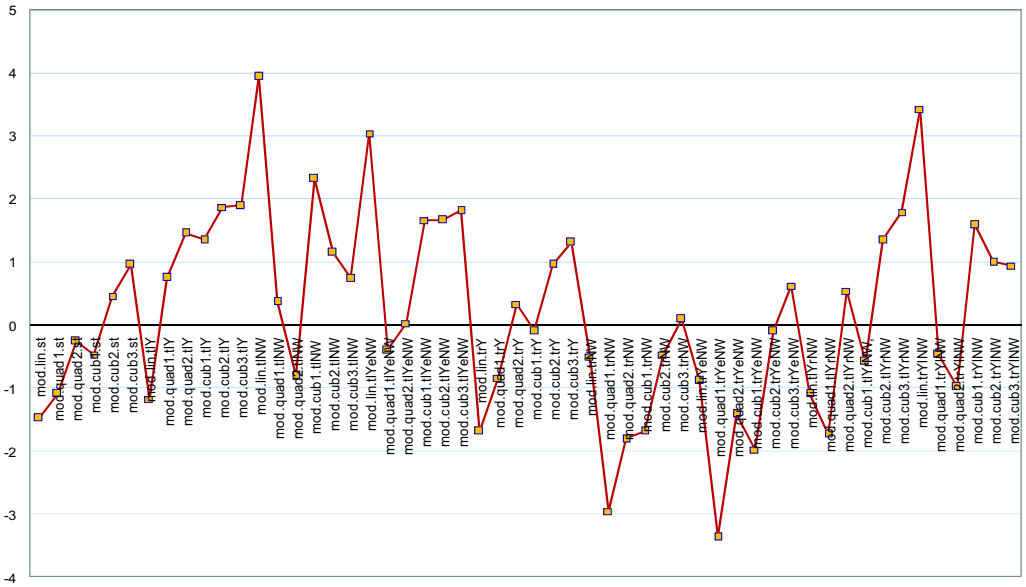

Figure 27. Unreplicated colour fading case study: AIC for colour C5

In this case the model which corresponds to higher adjusted R^2 and lower AIC is the quadratic model transformed with square root on the variable NW. The number of *p*-values indicating a significant difference between products compared is equal to two, and is the highest compared to other models, as in all other cases already treated. We report details on *p*-values for pairwise comparisons (Tab. 17) and a graph reporting the predicted values by the model (Fig. 28).

	P2	P3
P1	0.0162	0.7521
P2	-	0.0103

Table 17. Pairwise p-values between pairs of product (colour C5)

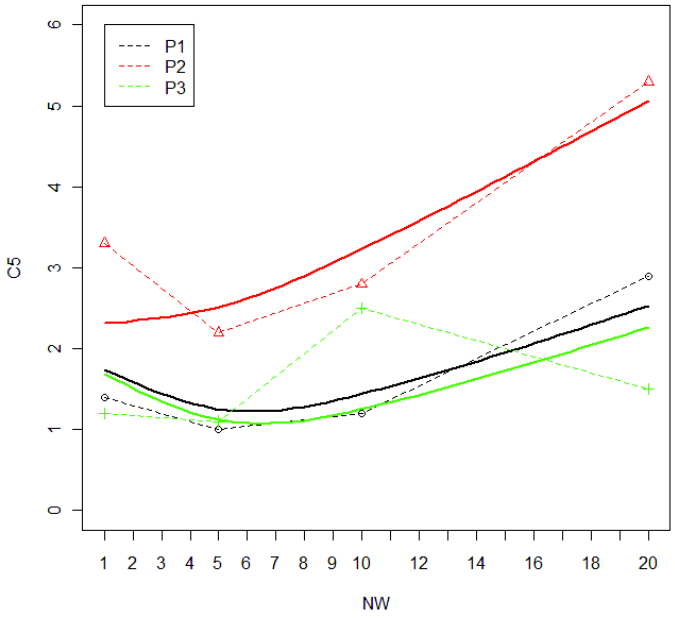

Figure 28. Predicted and observed values for colour fading case study (colour C5)

We now calculate the global ranking in order to find out the best product among those tested with reference to the colour fading secondary performance. We report the global ranking found through directional permutation *p*-value (Fig. 29) and the pairwise difference between ranking parameters (Fig. 30).

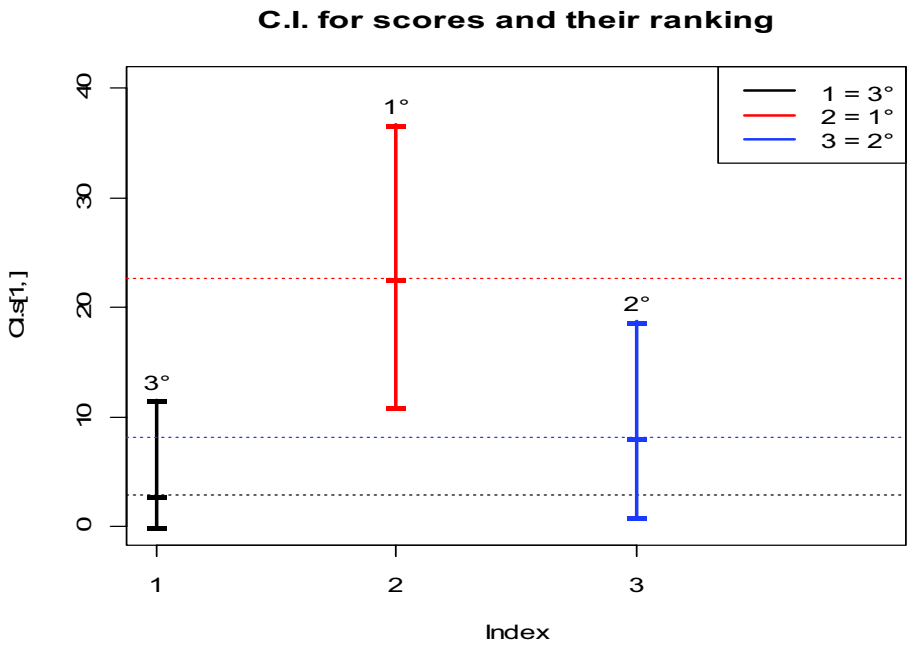

Figure 29. Global ranking for colour fading case study

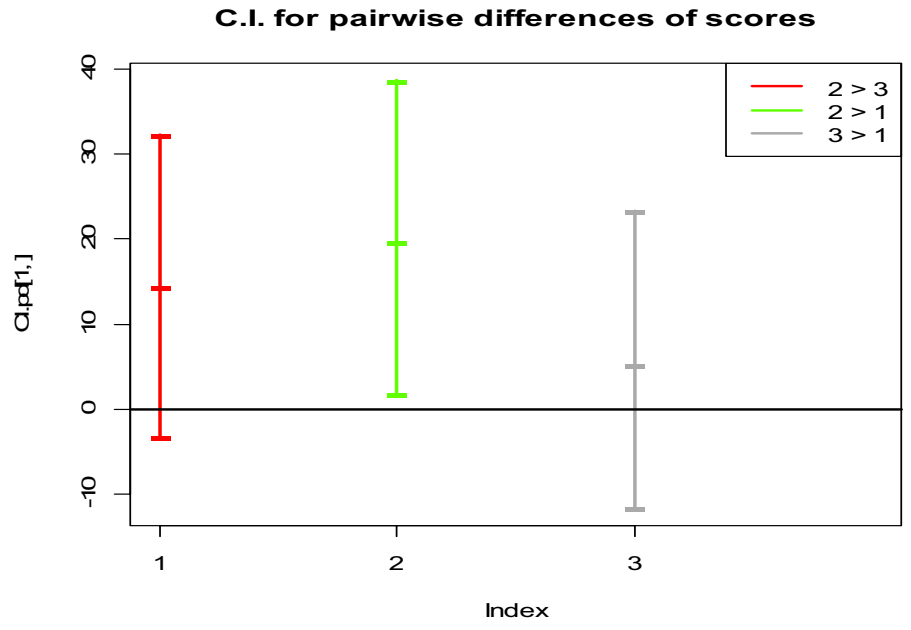

Figure 30. Pairwise differences of ranking parameters for colour fading case study

Results suggests that the best product is the second one.

4.1.4 Heating element incrustation

Regarding the case study for the secondary performance called heating element incrustation there are three products under investigation which are measured on five different number of washes (6,12,18,24,30). An analysis of 54 alternative models showed that the model with best fitting is the quadratic one without transformation. For this model the number of *p*-values that lead to rejection of the null hypothesis, detailed in Table 18, is just three.

	P2	P3
P1	0.0000	0.0183
P2	-	0.0221

Table 18. Pairwise p-values between pairs of product

In this type of experiment the response variable, unlike the cases previously analyzed, is just one so that it is not necessary to implement a global ranking procedure. The results reported in Table 18 along with the observed values (Fig. 31) allow us to state that the best product is the anti-lime P2 (the lower the deposit of limestone, the better the product).

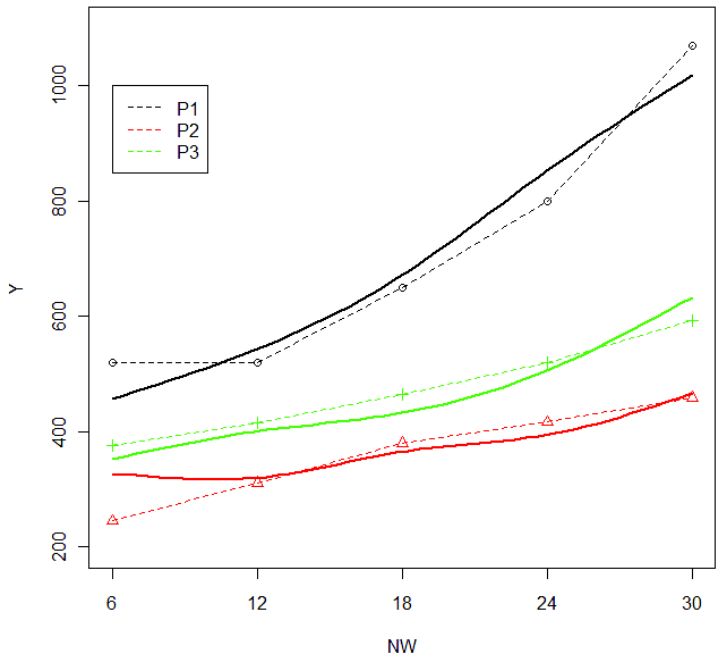

Figure 31. Predicted and observed values for heating element incrustation case study

4.2 Replicated case studies

In this section we apply the proposed solutions to the case of secondary performance analysis of new detergents with replicated experiments that is when are available several independent replicates of the same treatment (product).

4.2.1 Whiteness

In this case study we investigate the whiteness of four different products (P1, ..., P4), in case of two independent replicates per product and where these products are evaluated in five different washing cycles (5,10,15,20,25), and the whiteness measurements are obtained for seven different types of textile (wfk10,wfk20,wfk30,wfk40,wfk50, empa,tecc). The statistical methodology is obviously the same we applied in the previous paragraph 4.1.

To illustrate the univariate statistical analysis, for the sake of simplicity we consider only the type of textile wfk40. As usual, the first step consists in choosing the model that best fits the data in our possession, so we will choose among the 54 considered models the one which maximize the *adjusted* R^2 (Fig. 32) and minimize the *AIC* (Fig. 33). The more appropriate model is in this case the cubic model with logarithmic transformation for the covariate NW.

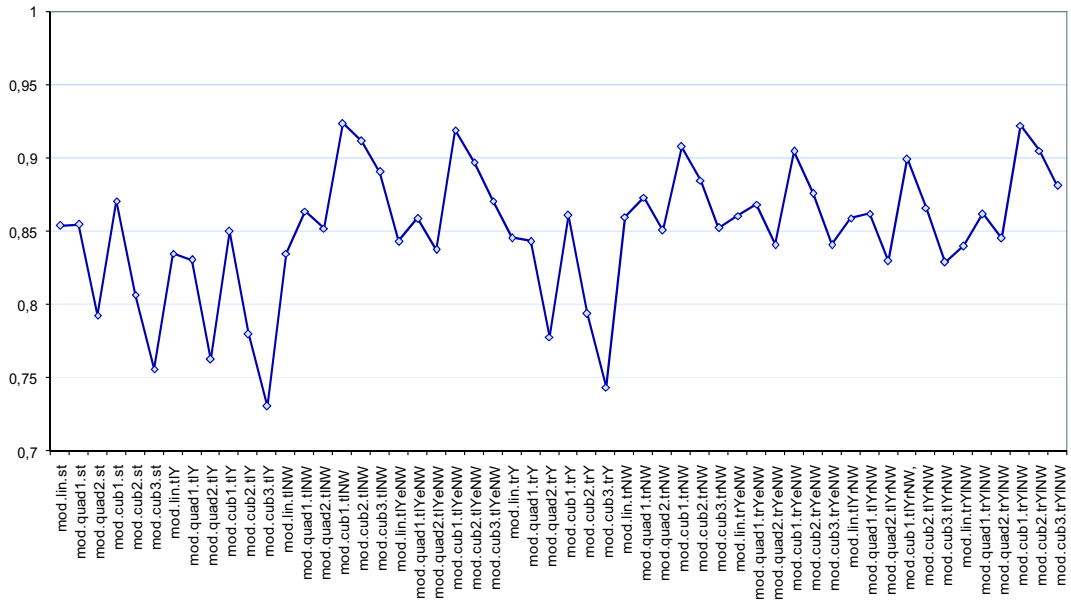

Figure 32. Replicated whiteness case study: adjusted R^2 for wfk40 textile

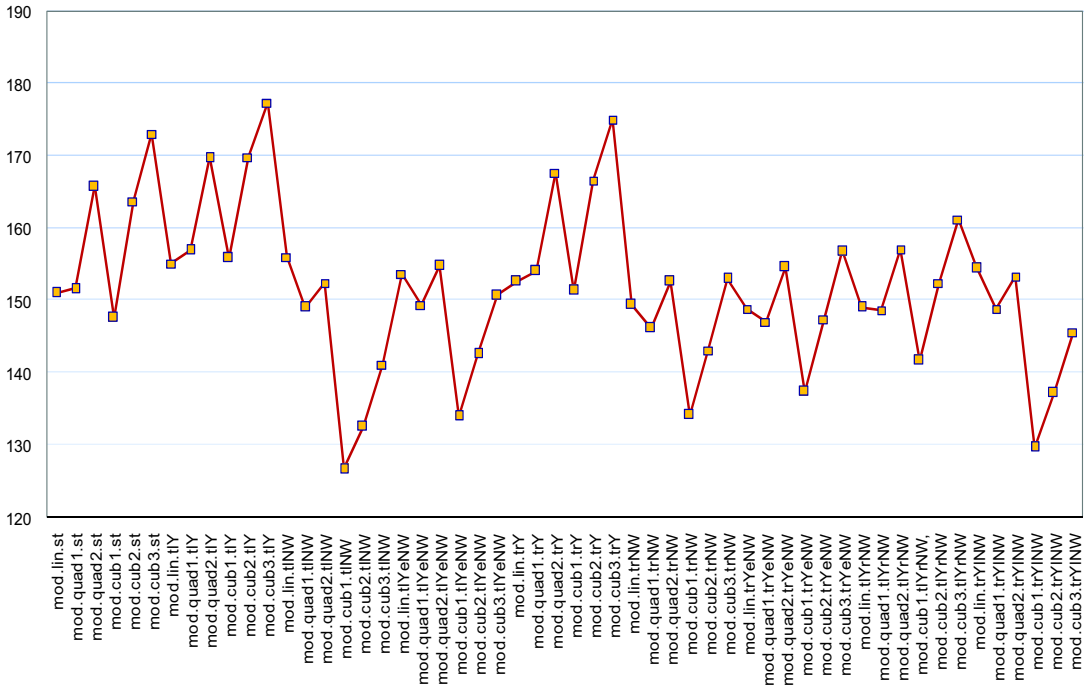

Figure 33. Replicated whiteness case study: AIC for wfk40 textile

We report in Figure 34 the number of significant *p*-values by model. Table 19 shows the details of the *p*-values of pairwise comparisons.

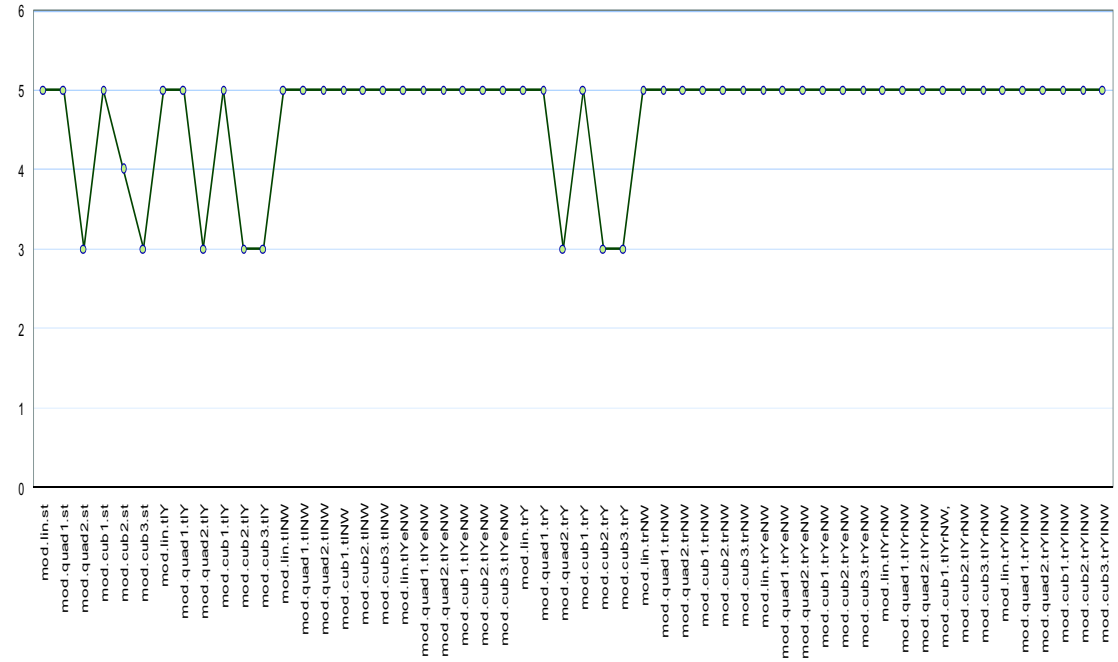

Figure 34. Replicated whiteness case study: no. of significant p-values (wfk40 textile)

	P2	P3	P4
P1	0.0000	0.0000	0.0000
P2	-	0.0011	0.0002
P3	-	-	0.5214

Table 19. Pairwise p-values between pairs of product (wfk40 textile)

The values predicted by the selected model, represented by a continuous line, are shown in Fig. 35.

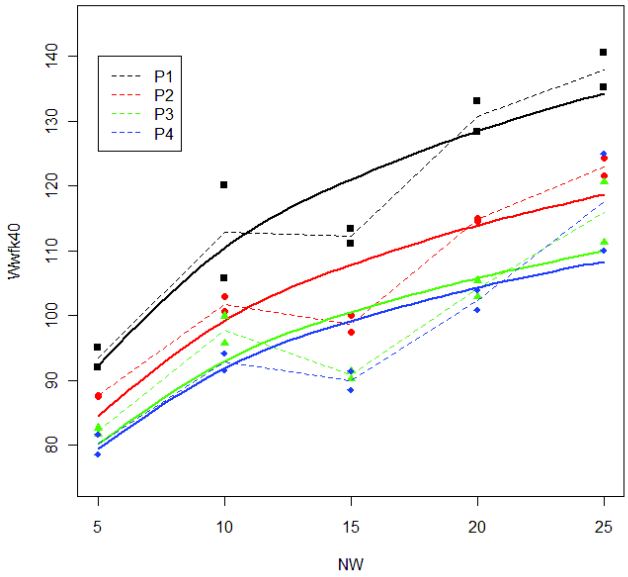

Figure 35. Predicted and observed values for whiteness case study (wfk40 textile)

As the unreplicated case studies we analyze all the experiment responses for all types of textile and then we proceed with the global ranking analysis. We report the results obtained with the global ranking methodology by means of residuals calculated from original data and by the estimates obtained through the regression model.

From the obtained results it is clear that the first product is the best, while the third and fourth product seem to be equivalent to each other and to be the worst among the four tested products.

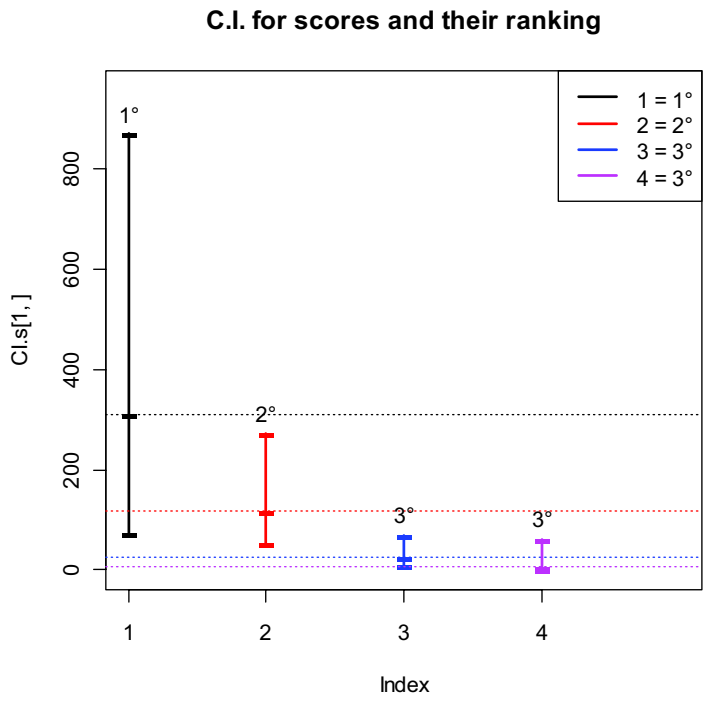

Figure 36. Global ranking for the replicated whiteness case study

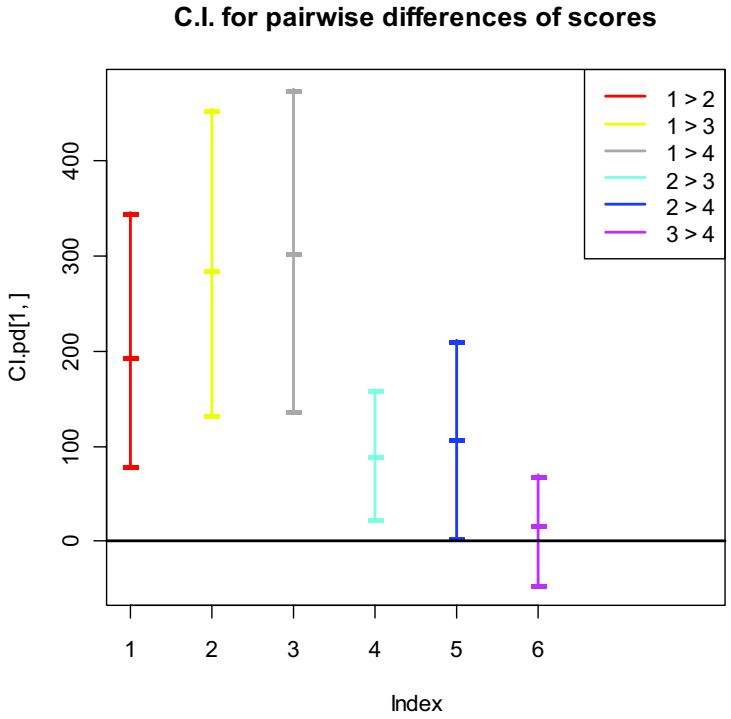

Figure 37. Pairwise differences of ranking parameters for the replicated whiteness case study

4.2.2 Colour transfer

In the experiment to evaluate the secondary performance called colour transfer we consider the responses of a study involving four different products, with reference to 14 colours, with 5 washing cycles (5,10,15,20) and with three independent replicates for each product the observed and estimated values for the colour AISE1.

After univariate analysis for all tested colours, adjusted R^2 and *AIC* indexes suggest to choose the linear regression model with logarithmic transformation for the independent variable NW. . In the following we report the observed and predicted values (Fig. 38), while Table 20 reports the details of the *p*-values of pairwise comparisons for colour AISE1.

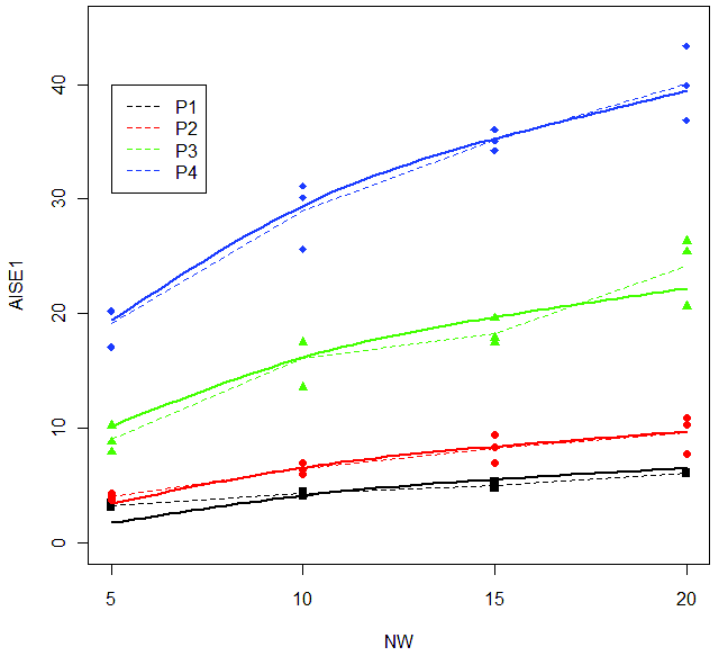

Figure 38. Predicted and observed values for colour transfer case study (colour AISE1)

	P2	P3	P4
P1	0.0001	0.0000	0.0000
P2	-	0.0000	0.0000
P3	-	-	0.0000

Table 20. Pairwise p-values between pairs of product (colour AISE1)

Again, for the global ranking analysis we use the residuals. Below are reported results we obtained (Fig. 39 and 40).

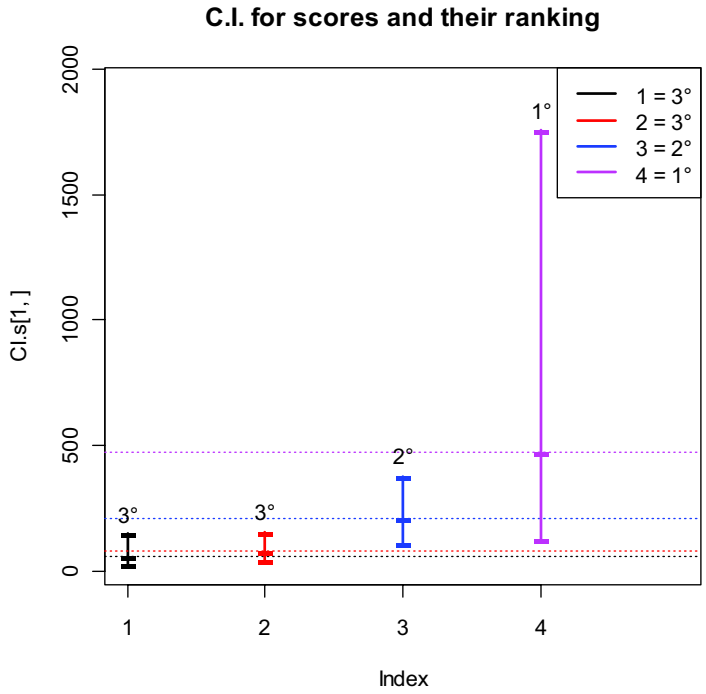

Figure 39. Global ranking for the replicated colour transfer case study

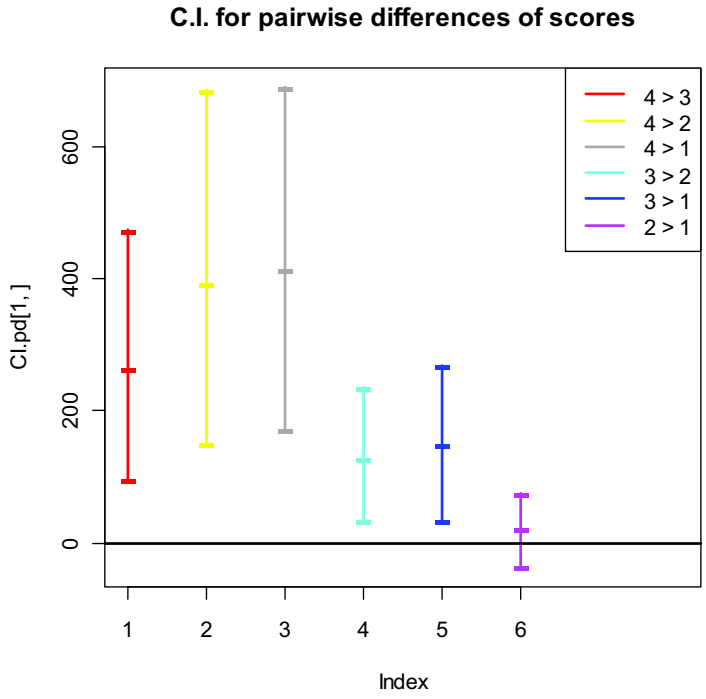

Figure 40. Pairwise differences of ranking parameters for the replicated colour transfer case study

In this case, the best product appears to be the one with the lowest values, as this product reduces the transfer of colour from one textile to another. For this reason, the products which should be considered best in this case are the first and second while the fourth product has to be considered as the worst.

Chapter 5. Conclusion and future research

In this thesis we presented some novel solutions within the multivariate analysis of variance/covariance – MANCOVA layout with the goal of finding out a global preference ranking of several treatments under investigation, when the treatment effect depends on one covariate. From a general point of view, both the simulation study and the application to a number of real case studies suggest us that the proposed solutions appear to be satisfactory, both in terms of the univariate and the multivariate analysis.

In particular, with regard to the univariate analysis on pairwise comparisons between treatments, from our simulations and analysis of real case studies we showed that the parametric tests (*t* test and F) lead to very satisfactory results, being able to discriminate different treatments also in cases where the difference seems to be very small.

The results obtained through the use of parametric tests were compared with those of nonparametric permutation tests where we used two different approaches, the first heuristic based on the residuals of the regression model and the second more formal based on a stratified by covariate level permutation strategy. Analysis of results proved that the parametric tests are more powerful than the permutation tests particularly to identify small differences, while when increasing the distance between treatments the power of the two approaches tends to be equivalent.

However, it is worth noting that the simulations were performed under the assumption of normality of errors, so that one could therefore expect that the parametric tests were more powerful than the permutation tests. It would be interesting for future research to relax from the assumption of normality in order to study the robustness of parametric and nonparametric tests when errors have skewed distribution and/or heavy tails. Moreover, it should be noted that the lack of power of permutation tests could be due to the relative small sample size of the reference experiments and to the consequent small cardinality of the permutation space.

Regarding the multivariate analysis, that is the aim of finding out a global ranking among the different treatments, it was performed using the same tests proposed for univariate analysis and then processing them through a permutation procedure to define appropriate interval estimators of the differences between so-called ranking parameters. The results are quite satisfactory, particularly the proposed procedures are almost always able to identify the best treatment from a multivariate point of view and then to hit our main goal.

Regarding the identification of treatment positions after the first, the proposed method showed some difficulties in discriminating treatments in which the responses are relatively close one each other. The result is that treatments tend to be drawn toward adjacent higher positions. This could however mainly due to the classification algorithm and certainly this point deserves more attention for future developments and improvements of the proposed methodology.

Another aspect worth to be noted is that while for the univariate procedure for pairwise comparisons was undoubtedly emerged as the parametric tests were the best, regarding the multivariate analysis there is not a preferred solution among the three different proposals. In particular, in cases where there are no replications of the experiment the solutions that lead to more satisfactory results are obtained by using univariate parametric tests and stratified permutation tests, while when there are replications of the experiment it seems that the best solution are univariate tests via permutation based on residuals resulting from the estimates of linear regression model. However, it is worth noting that the cases of greatest practical interest are those where we do not have replications of the experiment, since the costs for conducting these trials are often too high and it is preferred to perform experiments without replications.

A further remarkable aspect is that, when passing from univariate to multivariate analysis, the observed difference in performance between the three different types of tests becomes very small.

Finally, as regards future simulation studies on multivariate analysis related to the solution of global ranking, it would be interesting to have more powerful hardware equipment than that we had at our disposal. In fact, the computational

aspect has proved to be indeed a very important limitation in restricting the number of settings analyzed in this thesis.

Appendix: R code

– ***Creating a fictitious dataset for calculating regression models***

```
trasforma <- function(dati, vFact, Treat, baseL, NW, n_prodotti=4,n_var_risp){
    treat<- as.factor(dati$treat)
    n <- nrow(dati)
    nvar <- ncol(dati)
    for(i in vFact){
        dati[,i] <- as.factor(dati[,i])
        tab<- table(dati[,i]) ; C = length(tab)
        contrasts(dati[,i])<- contr.sum(C)
        ind <- c( baseL[i], (1:C)[-baseL] )
        contrasts(dati[,i])<- temp<- contrasts(dati[,i])[ind,]
    }#end-for
    nLev = length( levels(dati[,Treat]) )
    B <- n / nLev
    temp<- contrasts(dati[,Treat])
    for(j in 1:(B-1)) temp <- rbind( temp, contrasts(dati[,Treat]) )
    vNew.lin.st <- dati[,NW] * temp ; row.names(vNew.lin.st)<-NULL
    dimnames(vNew.lin.st)[[2]] <- paste("P", 2:(nLev), sep="")
    vNew.quad2.st <- (dati[,NW])^2 * temp ; row.names(vNew.quad2.st)<-NULL
    dimnames(vNew.quad2.st)[[2]] <- paste("P", 2:(nLev), sep="")
    vNew.cub3.st <- (dati[,NW])^3 * temp ; row.names(vNew.cub3.st)<-NULL
    dimnames(vNew.cub3.st)[[2]] <- paste("P", 2:(nLev), sep="")
    vNew.lin.tlNW <- log(dati[,NW]) * temp ; row.names(vNew.lin.tlNW)<-NULL
    dimnames(vNew.lin.tlNW)[[2]] <- paste("P", 2:(nLev), sep="")
    vNew.quad2.tlNW <- (log(dati[,NW]))^2 * temp ;
    row.names(vNew.quad2.tlNW)<-NULL
    dimnames(vNew.quad2.tlNW)[[2]] <- paste("P", 2:(nLev), sep="")
    vNew.cub3.tlNW <- (log(dati[,NW]))^3 * temp ;
    row.names(vNew.cub3.tlNW)<-NULL
    dimnames(vNew.cub3.tlNW)[[2]] <- paste("P", 2:(nLev), sep="")
    vNew.lin.trNW <- sqrt(dati[,NW]) * temp ; row.names(vNew.lin.trNW)<-NULL
    dimnames(vNew.lin.trNW)[[2]] <- paste("P", 2:(nLev), sep="")
    vNew.quad2.trNW <- (sqrt(dati[,NW]))^2 * temp ;
    row.names(vNew.quad2.trNW)<-NULL
    dimnames(vNew.quad2.trNW)[[2]] <- paste("P", 2:(nLev), sep="")
    vNew.cub3.trNW <- (sqrt(dati[,NW]))^3 * temp ;
    row.names(vNew.cub3.trNW)<-NULL
    dimnames(vNew.cub3.trNW)[[2]] <- paste("P", 2:(nLev), sep="")
```

```
Cost<-c(rep(1,nrow(dati)))
# dati senza trasf
dati.lin.st <- cbind(dati[,3:nvar],treat,Cost,dati$NW,vNew.lin.st)
dimnames(dati.lin.st)[[2]][(n_var_risp+3)]<-"NW"
dati.quad1.st <-
cbind(dati[,3:nvar],treat,Cost,dati$NW,(dati$NW)^2,vNew.lin.st)
dimnames(dati.quad1.st)[[2]][(n_var_risp+3)]<-"NW"
dimnames(dati.quad1.st)[[2]][(n_var_risp+4)]<-"NW2"
dati.quad2.st <-
cbind(dati[,3:nvar],treat,Cost,dati$NW,(dati$NW)^2,vNew.quad2.st)
dimnames(dati.quad2.st)[[2]][(n_var_risp+3)]<-"NW"
dimnames(dati.quad2.st)[[2]][(n_var_risp+4)]<-"NW2"
dati.cub1.st <- cbind(dati[,3:nvar],treat,Cost,dati$NW,(dati$NW)^2,
(dati$NW)^3,vNew.lin.st)
dimnames(dati.cub1.st)[[2]][(n_var_risp+3)]<-"NW"
dimnames(dati.cub1.st)[[2]][(n_var_risp+4)]<-"NW2"
dimnames(dati.cub1.st)[[2]][(n_var_risp+5)]<-"NW3"
dati.cub2.st <- cbind(dati[,3:nvar],treat,Cost,dati$NW,(dati$NW)^2,
(dati$NW)^3,vNew.quad2.st)
dimnames(dati.cub2.st)[[2]][(n_var_risp+3)]<-"NW"
dimnames(dati.cub2.st)[[2]][(n_var_risp+4)]<-"NW2"
dimnames(dati.cub2.st)[[2]][(n_var_risp+5)]<-"NW3"
dati.cub3.st <- cbind(dati[,3:nvar],treat,Cost,dati$NW,(dati$NW)^2,
(dati$NW)^3,vNew.cub3.st)
dimnames(dati.cub3.st)[[2]][(n_var_risp+3)]<-"NW"
dimnames(dati.cub3.st)[[2]][(n_var_risp+4)]<-"NW2"
dimnames(dati.cub3.st)[[2]][(n_var_risp+5)]<-"NW3"
# dati con trasf log per Y
dati.lin.tlY <- cbind(log(dati[,3:nvar]),treat,Cost,dati$NW,vNew.lin.st)
dimnames(dati.lin.tlY)[[2]][(n_var_risp+3)]<-"NW"
dati.quad1.tlY <-
cbind(log(dati[,3:nvar]),treat,Cost,dati$NW,(dati$NW)^2,vNew.lin.st)
dimnames(dati.quad1.tlY)[[2]][(n_var_risp+3)]<-"NW"
dimnames(dati.quad1.tlY)[[2]][(n_var_risp+4)]<-"NW2"
dati.quad2.tlY <-
cbind(log(dati[,3:nvar]),treat,Cost,dati$NW,(dati$NW)^2,vNew.quad2.st)
dimnames(dati.quad2.tlY)[[2]][(n_var_risp+3)]<-"NW"
dimnames(dati.quad2.tlY)[[2]][(n_var_risp+4)]<-"NW2"
dati.cub1.tlY <- cbind(log(dati[,3:nvar]),treat,Cost,dati$NW,(dati$NW)^2,
(dati$NW)^3,vNew.lin.st)
dimnames(dati.cub1.tlY)[[2]][(n_var_risp+3)]<-"NW"
dimnames(dati.cub1.tlY)[[2]][(n_var_risp+4)]<-"NW2"
dimnames(dati.cub1.tlY)[[2]][(n_var_risp+5)]<-"NW3"
dati.cub2.tlY <- cbind(log(dati[,3:nvar]),treat,Cost,dati$NW,(dati$NW)^2,
```

```
    (dati$NW)^3,vNew.quad2.st)
    dimnames(dati.cub2.tlY)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub2.tlY)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub2.tlY)[[2]][(n_var_risp+5)]<-"NW3"
    dati.cub3.tlY <- cbind(log(dati[,3:nvar]),treat,Cost,dati$NW,(dati$NW)^2,
    (dati$NW)^3,vNew.cub3.st)
    dimnames(dati.cub3.tlY)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub3.tlY)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub3.tlY)[[2]][(n_var_risp+5)]<-"NW3"
    # trasformata log per NW
   dati.lin.tlNW <- cbind(dati[,3:nvar],treat,Cost,log(dati$NW),vNew.lin.tlNW)
    dimnames(dati.lin.tlNW)[[2]][(n_var_risp+3)]<-"NW"
    dati.quad1.tlNW <-
   cbind(dati[,3:nvar],treat,Cost,log(dati$NW),(log(dati$NW))^2,vNew.lin.tlNW)
    dimnames(dati.quad1.tlNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.quad1.tlNW)[[2]][(n_var_risp+4)]<-"NW2"
    dati.quad2.tlNW <-
cbind(dati[,3:nvar],treat,Cost,log(dati$NW),(log(dati$NW))^2,vNew.quad2.tlNW)
    dimnames(dati.quad2.tlNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.quad2.tlNW)[[2]][(n_var_risp+4)]<-"NW2"
    dati.cub1.tlNW <-
    cbind(dati[,3:nvar],treat,Cost,log(dati$NW),(log(dati$NW))^2,
   (log(dati$NW))^3,vNew.lin.tlNW)
    dimnames(dati.cub1.tlNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub1.tlNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub1.tlNW)[[2]][(n_var_risp+5)]<-"NW3"
    dati.cub2.tlNW <-
    cbind(dati[,3:nvar],treat,Cost,log(dati$NW),(log(dati$NW))^2,
    (log(dati$NW))^3,vNew.quad2.tlNW)
    dimnames(dati.cub2.tlNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub2.tlNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub2.tlNW)[[2]][(n_var_risp+5)]<-"NW3"
    dati.cub3.tlNW <-
    cbind(dati[,3:nvar],treat,Cost,log(dati$NW),(log(dati$NW))^2,
    (log(dati$NW))^3,vNew.cub3.tlNW)
    dimnames(dati.cub3.tlNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub3.tlNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub3.tlNW)[[2]][(n_var_risp+5)]<-"NW3"
    # trasformata log per Y e NW
    dati.lin.tlYeNW <-
    cbind(log(dati[,3:nvar]),treat,Cost,log(dati$NW),vNew.lin.tlNW)
    dimnames(dati.lin.tlYeNW)[[2]][(n_var_risp+3)]<-"NW"
```

```
    dati.quad1.tlYeNW <-
cbind(log(dati[,3:nvar]),treat,Cost,log(dati$NW),(log(dati$NW))^2,vNew.lin.tlN
W)
    dimnames(dati.quad1.tlYeNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.quad1.tlYeNW)[[2]][(n_var_risp+4)]<-"NW2"
    dati.quad2.tlYeNW <-
cbind(log(dati[,3:nvar]),treat,Cost,log(dati$NW),(log(dati$NW))^2,vNew.quad2.t
lNW)
    dimnames(dati.quad2.tlYeNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.quad2.tlYeNW)[[2]][(n_var_risp+4)]<-"NW2"
    dati.cub1.tlYeNW <-
cbind(log(dati[,3:nvar]),treat,Cost,log(dati$NW),(log(dati$NW))^2,
(log(dati$NW))^3,vNew.lin.tlNW)
    dimnames(dati.cub1.tlYeNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub1.tlYeNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub1.tlYeNW)[[2]][(n_var_risp+5)]<-"NW3"
    dati.cub2.tlYeNW <-
cbind(log(dati[,3:nvar]),treat,Cost,log(dati$NW),(log(dati$NW))^2,
(log(dati$NW))^3,vNew.quad2.tlNW)
    dimnames(dati.cub2.tlYeNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub2.tlYeNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub2.tlYeNW)[[2]][(n_var_risp+5)]<-"NW3"
    dati.cub3.tlYeNW <-
cbind(log(dati[,3:nvar]),treat,Cost,log(dati$NW),(log(dati$NW))^2,
(log(dati$NW))^3,vNew.cub3.tlNW)
    dimnames(dati.cub3.tlYeNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub3.tlYeNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub3.tlYeNW)[[2]][(n_var_risp+5)]<-"NW3"
    # trasformata radq per Y
    dati.lin.trY <- cbind(sqrt(dati[,3:nvar]),treat,Cost,dati$NW,vNew.lin.st)
    dimnames(dati.lin.trY)[[2]][(n_var_risp+3)]<-"NW"
    dati.quad1.trY <-
cbind(sqrt(dati[,3:nvar]),treat,Cost,dati$NW,(dati$NW)^2,vNew.lin.st)
    dimnames(dati.quad1.trY)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.quad1.trY)[[2]][(n_var_risp+4)]<-"NW2"
    dati.quad2.trY <-
cbind(sqrt(dati[,3:nvar]),treat,Cost,dati$NW,(dati$NW)^2,vNew.quad2.st)
    dimnames(dati.quad2.trY)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.quad2.trY)[[2]][(n_var_risp+4)]<-"NW2"
    dati.cub1.trY <- cbind(sqrt(dati[,3:nvar]),treat,Cost,dati$NW,(dati$NW)^2,
(dati$NW)^3,vNew.lin.st)
    dimnames(dati.cub1.trY)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub1.trY)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub1.trY)[[2]][(n_var_risp+5)]<-"NW3"
```

```
    dati.cub2.trY <- cbind(sqrt(dati[,3:nvar]),treat,Cost,dati$NW,(dati$NW)^2,
(dati$NW)^3,vNew.quad2.st)
    dimnames(dati.cub2.trY)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub2.trY)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub2.trY)[[2]][(n_var_risp+5)]<-"NW3"
    dati.cub3.trY <- cbind(sqrt(dati[,3:nvar]),treat,Cost,dati$NW,(dati$NW)^2,
(dati$NW)^3,vNew.cub3.st)
    dimnames(dati.cub3.trY)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub3.trY)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub3.trY)[[2]][(n_var_risp+5)]<-"NW3"
    # trasformata radq per NW
    dati.lin.trNW <-
cbind(dati[,3:nvar],treat,Cost,sqrt(dati$NW),vNew.lin.trNW)
    dimnames(dati.lin.trNW)[[2]][(n_var_risp+3)]<-"NW"
    dati.quad1.trNW <-
cbind(dati[,3:nvar],treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,vNew.lin.trNW)
    dimnames(dati.quad1.trNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.quad1.trNW)[[2]][(n_var_risp+4)]<-"NW2"
    dati.quad2.trNW <-
cbind(dati[,3:nvar],treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,vNew.quad2.trNW
)
    dimnames(dati.quad2.trNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.quad2.trNW)[[2]][(n_var_risp+4)]<-"NW2"
    dati.cub1.trNW <-
cbind(dati[,3:nvar],treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,
(sqrt(dati$NW))^3,vNew.lin.trNW)
    dimnames(dati.cub1.trNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub1.trNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub1.trNW)[[2]][(n_var_risp+5)]<-"NW3"
    dati.cub2.trNW <-
cbind(dati[,3:nvar],treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,
(sqrt(dati$NW))^3,vNew.quad2.trNW)
    dimnames(dati.cub2.trNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub2.trNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub2.trNW)[[2]][(n_var_risp+5)]<-"NW3"
    dati.cub3.trNW <-
cbind(dati[,3:nvar],treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,
(sqrt(dati$NW))^3,vNew.cub3.trNW)
    dimnames(dati.cub3.trNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub3.trNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub3.trNW)[[2]][(n_var_risp+5)]<-"NW3"
    # trasformata radq Y e NW
    dati.lin.trYeNW <-
cbind(sqrt(dati[,3:nvar]),treat,Cost,sqrt(dati$NW),vNew.lin.trNW)
```

```
    dimnames(dati.lin.trYeNW)[[2]][(n_var_risp+3)]<-"NW"
    dati.quad1.trYeNW <-
cbind(sqrt(dati[,3:nvar]),treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,vNew.lin.
trNW)
    dimnames(dati.quad1.trYeNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.quad1.trYeNW)[[2]][(n_var_risp+4)]<-"NW2"
    dati.quad2.trYeNW <-
cbind(sqrt(dati[,3:nvar]),treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,vNew.quad
2.trNW)
    dimnames(dati.quad2.trYeNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.quad2.trYeNW)[[2]][(n_var_risp+4)]<-"NW2"
    dati.cub1.trYeNW <-
cbind(sqrt(dati[,3:nvar]),treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,
(sqrt(dati$NW))^3,vNew.lin.trNW)
    dimnames(dati.cub1.trYeNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub1.trYeNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub1.trYeNW)[[2]][(n_var_risp+5)]<-"NW3"
    dati.cub2.trYeNW <-
cbind(sqrt(dati[,3:nvar]),treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,
(sqrt(dati$NW))^3,vNew.quad2.trNW)
    dimnames(dati.cub2.trYeNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub2.trYeNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub2.trYeNW)[[2]][(n_var_risp+5)]<-"NW3"
    dati.cub3.trYeNW <-
cbind(sqrt(dati[,3:nvar]),treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,
(sqrt(dati$NW))^3,vNew.cub3.trNW)
    dimnames(dati.cub3.trYeNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub3.trYeNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub3.trYeNW)[[2]][(n_var_risp+5)]<-"NW3"
    # trasformata radq Nw log Y
    dati.lin.tlYrNW <-
cbind(log(dati[,3:nvar]),treat,Cost,sqrt(dati$NW),vNew.lin.trNW)
    dimnames(dati.lin.tlYrNW)[[2]][(n_var_risp+3)]<-"NW"
    dati.quad1.tlYrNW <-
cbind(log(dati[,3:nvar]),treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,vNew.lin.t
rNW)
    dimnames(dati.quad1.tlYrNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.quad1.tlYrNW)[[2]][(n_var_risp+4)]<-"NW2"
    dati.quad2.tlYrNW <-
cbind(log(dati[,3:nvar]),treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,vNew.quad2
.trNW)
    dimnames(dati.quad2.tlYrNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.quad2.tlYrNW)[[2]][(n_var_risp+4)]<-"NW2"
```

```
    dati.cub1.tlYrNW <-
cbind(log(dati[,3:nvar]),treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,
(sqrt(dati$NW))^3,vNew.lin.trNW)
    dimnames(dati.cub1.tlYrNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub1.tlYrNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub1.tlYrNW)[[2]][(n_var_risp+5)]<-"NW3"
    dati.cub2.tlYrNW <-
cbind(log(dati[,3:nvar]),treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,
(sqrt(dati$NW))^3,vNew.quad2.trNW)
    dimnames(dati.cub2.tlYrNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub2.tlYrNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub2.tlYrNW)[[2]][(n_var_risp+5)]<-"NW3"
    dati.cub3.tlYrNW <-
cbind(log(dati[,3:nvar]),treat,Cost,sqrt(dati$NW),(sqrt(dati$NW))^2,
(sqrt(dati$NW))^3,vNew.cub3.trNW)
    dimnames(dati.cub3.tlYrNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub3.tlYrNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub3.tlYrNW)[[2]][(n_var_risp+5)]<-"NW3"
    # trasformata log Nw radq Y
    dati.lin.trYlNW <-
cbind(sqrt(dati[,3:nvar]),treat,Cost,log(dati$NW),vNew.lin.tlNW)
    dimnames(dati.lin.trYlNW)[[2]][(n_var_risp+3)]<-"NW"
    dati.quad1.trYlNW <-
cbind(sqrt(dati[,3:nvar]),treat,Cost,log(dati$NW),(log(dati$NW))^2,vNew.lin.tl
NW)
    dimnames(dati.quad1.trYlNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.quad1.trYlNW)[[2]][(n_var_risp+4)]<-"NW2"
    dati.quad2.trYlNW <-
cbind(sqrt(dati[,3:nvar]),treat,Cost,log(dati$NW),(log(dati$NW))^2,vNew.quad2.
tlNW)
    dimnames(dati.quad2.trYlNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.quad2.trYlNW)[[2]][(n_var_risp+4)]<-"NW2"
    dati.cub1.trYlNW <-
cbind(sqrt(dati[,3:nvar]),treat,Cost,log(dati$NW),(log(dati$NW))^2,
(log(dati$NW))^3,vNew.lin.tlNW)
    dimnames(dati.cub1.trYlNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub1.trYlNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub1.trYlNW)[[2]][(n_var_risp+5)]<-"NW3"
    dati.cub2.trYlNW <-
cbind(sqrt(dati[,3:nvar]),treat,Cost,log(dati$NW),(log(dati$NW))^2,
(log(dati$NW))^3,vNew.quad2.tlNW)
    dimnames(dati.cub2.trYlNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub2.trYlNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub2.trYlNW)[[2]][(n_var_risp+5)]<-"NW3"
```

```
    dati.cub3.trYlNW <-
cbind(sqrt(dati[,3:nvar]),treat,Cost,log(dati$NW),(log(dati$NW))^2,
(log(dati$NW))^3,vNew.cub3.tlNW)
    dimnames(dati.cub3.trYlNW)[[2]][(n_var_risp+3)]<-"NW"
    dimnames(dati.cub3.trYlNW)[[2]][(n_var_risp+4)]<-"NW2"
    dimnames(dati.cub3.trYlNW)[[2]][(n_var_risp+5)]<-"NW3"
return(list(dati.lin.st=dati.lin.st,dati.quad1.st=dati.quad1.st,
dati.quad2.st=dati.quad2.st,dati.cub1.st=dati.cub1.st,
dati.cub2.st=dati.cub2.st,dati.cub3.st=dati.cub3.st,dati.lin.tlY=dati.lin.tlY,
dati.quad1.tlY=dati.quad1.tlY,
dati.quad2.tlY=dati.quad2.tlY,dati.cub1.tlY=dati.cub1.tlY,
dati.cub2.tlY=dati.cub2.tlY,dati.cub3.tlY=dati.cub3.tlY,
dati.lin.tlNW=dati.lin.tlNW,dati.quad1.tlNW=dati.quad1.tlNW,
dati.quad2.tlNW=dati.quad2.tlNW,dati.cub1.tlNW=dati.cub1.tlNW,
dati.cub2.tlNW=dati.cub2.tlNW,dati.cub3.tlNW=dati.cub3.tlNW,
dati.lin.tlYeNW=dati.lin.tlYeNW,dati.quad1.tlYeNW=dati.quad1.tlYeNW,
dati.quad2.tlYeNW=dati.quad2.tlYeNW,dati.cub1.tlYeNW=dati.cub1.tlYeNW,
dati.cub2.tlYeNW=dati.cub2.tlYeNW,
dati.cub3.tlYeNW=dati.cub3.tlYeNW,dati.lin.trY=dati.lin.trY,
dati.quad1.trY=dati.quad1.trY,
dati.quad2.trY=dati.quad2.trY,dati.cub1.trY=dati.cub1.trY,
dati.cub2.trY=dati.cub2.trY,dati.cub3.trY=dati.cub3.trY,
dati.lin.trNW=dati.lin.trNW,dati.quad1.trNW=dati.quad1.trNW,
dati.quad2.trNW=dati.quad2.trNW,dati.cub1.trNW=dati.cub1.trNW,
dati.cub2.trNW=dati.cub2.trNW,dati.cub3.trNW=dati.cub3.trNW,
dati.lin.trYeNW=dati.lin.trYeNW,dati.quad1.trYeNW=dati.quad1.trYeNW,
dati.quad2.trYeNW=dati.quad2.trYeNW,dati.cub1.trYeNW=dati.cub1.trYeNW,
dati.cub2.trYeNW=dati.cub2.trYeNW,dati.cub3.trYeNW=dati.cub3.trYeNW,
dati.lin.tlYrNW=dati.lin.tlYrNW,dati.quad1.tlYrNW=dati.quad1.tlYrNW,
dati.quad2.tlYrNW=dati.quad2.tlYrNW,
dati.cub1.tlYrNW=dati.cub1.tlYrNW,dati.cub2.tlYrNW=dati.cub2.tlYrNW,
dati.cub3.tlYrNW=dati.cub3.tlYrNW,
dati.lin.trYlNW=dati.lin.trYlNW,dati.quad1.trYlNW=dati.quad1.trYlNW,
dati.quad2.trYlNW=dati.quad2.trYlNW,
dati.cub1.trYlNW=dati.cub1.trYlNW,dati.cub2.trYlNW=dati.cub2.trYlNW,
dati.cub3.trYlNW=dati.cub3.trYlNW))
}#end-trasforma
```

- ***Create vector of contrasts for calculating F-test***

```
vett_confr_test<-function(n_prodotti){
confronti<-combn(n_prodotti,2)
```

```
n_confr<-(n_prodotti*(n_prodotti-1))/2
# confronti modello lineare
n_col.lin<-2+n_prodotti
matr_confr.lin<-matrix(0,ncol=n_col.lin,nrow=n_confr)
for(i in 1:n_confr){
a<-confronti[1,i]
matr_confr.lin[i,a+2]<-1
b<-confronti[2,i]
matr_confr.lin[i,b+2]<--1}
# confronti modello quadratico
n_col.quad<-3+n_prodotti
matr_confr.quad<-matrix(0,ncol=n_col.quad,nrow=n_confr)
for(i in 1:n_confr){
a<-confronti[1,i]
matr_confr.quad[i,a+3]<-1
b<-confronti[2,i]
matr_confr.quad[i,b+3]<--1}
# confronti modello cubico
n_col.cub<-4+n_prodotti
matr_confr.cub<-matrix(0,ncol=n_col.cub,nrow=n_confr)
for(i in 1:n_confr){
a<-confronti[1,i]
matr_confr.cub[i,a+4]<-1
b<-confronti[2,i]
matr_confr.cub[i,b+4]<--1}
return(list(lin=matr_confr.lin,quad=matr_confr.quad,cub=matr_confr.cub))
} #end-vett_confr_test
```

- ***F-test calculation***

```
testF_nested_models_new<-
function(dati=dati.ok,out1=out1,matr_confr=matr_confr,Res.Df=Res.Df,n_var_risp
){
Z<-dati[,-c(1:n_var_risp)]
Z<-as.matrix(Z)
colnames(Z)<-NULL
err.mod.compl<-out1$residuals
err.mod.compl2<-t(err.mod.compl)%*%err.mod.compl
beta<-out1$coefficients[,1]
valori_testF<-NULL
p_values_testF<-NULL
Res.Df<-as.numeric(Res.Df)
```

```
for(i in 1:nrow(matr_confr)){
beta.vinc<-beta-
solve(t(Z)%*%Z)%*%matr_confr[i,]%*%solve(t(matr_confr[i,])%*%solve(t(Z)%*%Z)%*
%matr_confr[i,])%*%(t(matr_confr[i,])%*%beta)
err.mod.rid<-err.mod.compl-Z%*%(beta.vinc-beta)
err.mod.rid.2<-err.mod.compl2+t(beta.vinc-beta)%*%(t(Z)%*%Z)%*%(beta.vinc-
beta)
valori_testF[i]<-((err.mod.rid.2-err.mod.compl2)/err.mod.compl2)*Res.Df
p_values_testF[i]<-1-pf(valori_testF[i],1,Res.Df)}
return(list(valori_test=valori_testF,p_values=p_values_testF))}
```

– ***Calculating permutation tests***

```
test.perm.univ<-function(dati, C <- 5, n <- 1, nStrata <- 5, p <- 3,
B <- 1e3){
nDati <- nStrata*C
strata <- as.factor(dati[,2,1])
labs.strata <- levels(strata) ; levels(strata) <- 1:nStrata
vett.strata <- as.integer(labs.strata)
gr <- as.factor(dati[,1,1])
K <- C*(C-1)/2
labs <- character(K)
ris<-list()
for(i in 1:100){
T.temp <- array(NA, dim=c(B, nStrata, 2*K ))
dati.mod<-array(NA, dim=c(nDati,p,100))
dati.mod[,,i] <- dati[,,i][,-c(1,2)]
temp <- array(NA, dim=c( p, nStrata, 2*K))
for(ss in 1:nStrata){
    dat <- as.matrix(dati.mod[(strata == ss),,i])
    for(j in 1:p){
        mu <- as.numeric(tapply( dat[,j], gr[(strata == ss)], FUN=mean ))
        pc <- ( t(replicate(C, cbind(mu))) - mu ) * vett.strata[ss]
        temp[j,ss,] <- c( pc[lower.tri(pc)], - pc[lower.tri(pc)] )
    }#end_for-p
}#end_for-strata
T.temp[1,,] <- apply(temp, c(2,3), sum)
set.seed(seed)
tempInd <- array(1:C, dim=c(C, B-1, nStrata))
for(ss in 1:nStrata) tempInd[,,ss] <- tempInd[,,ss] + (ss-1)*C
indMat  <- apply(tempInd, c(2,3), FUN=sample, replace=FALSE)
for(bb in 2:B){
```

```
    dati.perm <- dati.mod[c(indMat[,(bb - 1),]),,i]
    temp <- array(NA, dim=c( p, nStrata, 2*K ))
    for(ss in 1:nStrata){
        dat <- as.matrix(dati.perm[(strata == ss),])
        for(j in 1:p){
            mu <- as.numeric(tapply( dat[,j], gr[(strata == ss)], FUN=mean ))
            pc <- ( t(replicate(C, cbind(mu))) - mu ) * vett.strata[ss]
            temp[j,ss,] <- c( pc[lower.tri(pc)], - pc[lower.tri(pc)] )
        }#end_for-p
    }#end_for-strata
    T.temp[bb,,] <- apply(temp, c(2,3), sum)
}#end_for-bb
#- passaggio ai p-value
P <- apply(T.temp, c(2,3), FUN= stat2p, B=B)
risF <- t( apply(P, 1, FUN=Fcomb, fun=Fish, ng=C) )
attr(risF, "dimnames")[[2]] <- paste(levels(gr),"> All")
ris[[i]]<-risF
print(i)
}
source("funzioni_mie//confInt5.r")
source("funzioni_mie//myBoxcox.r")
source("funzioni_mie//ranking4.r")
ris.fin<-list()
for(i in 1:100){
ris.fin[[i]] <- confInt5(ris[[i]], lam.range=c(0,4), alpha=0.05,
logLikCI=FALSE, showCI=FALSE, plotIt=FALSE)}
tab<-
table(factor(paste(1:C,"°",sep="")),factor(paste(1:C,"°",sep=""),ordered=TRUE)
,dnn=c("True","Observed"))-diag(C)
for(i in 1:100){
tab<-
tab+table(factor(1:C,ordered=TRUE),factor(ris.fin[[i]],levels=1:C,ordered=TRUE
))}
tab.new<-
table(factor(paste(1:C,"°",sep="")),factor(paste(1:C,"°",sep=""),ordered=TRUE)
,dnn=c("True","Observed"))-diag(C)
for(i in 1:C){
tab.new[i,]<-tab[(C+1)-i,]}
k=0
for(i in 1:100){
if( identical(as.integer(ris.fin[[i]]), as.integer(C:1)) ) k<-k+1}
}
```

– ***C.I. of pairwise differences and ranking***

```
confInt7 <- function(X, lam.range, alpha=0.05, logLikCI=TRUE, showCI=FALSE,
plotIt=FALSE){
#- Omitting rows containing non-finite values
if( any(!is.finite(X)) ){
    inf.omit <- function(ff){ all(is.finite(ff)) }#
    ind.ok <- apply(X, 1, FUN= inf.omit )
    X <- X[ind.ok,]
}#end_if-not-finite
#--
B <- nrow(X) ; C <- ncol(X) ; K <- C*(C-1)/2
Z <- array(NA, dim=c( B, K ))
X.t <- array(NA, dim=dim(X))
lam <- rep( NA,C ) ; lam.pd <- rep( NA,K )
CI.pd <- CIa <- array(NA,dim=c( 3, K ))
CI.s <- CIb <- array(NA,dim=c( 3, C ))
labs <- character(K)
#- Box-Cox transformation
transf <- function(y,la){ return( (y^la - 1)/la )}#
#- Inverse of Box-Cox transformation
transf.inv <- function(z,la){ return( (z * la + 1)^(1/la) ) }#
#- Shift for distribution of differences
shift <- rep(NA, K)
#- Ordering by observed scores
ord <- order(X[1,], decreasing=TRUE)
X <- X[,ord]
cont <- 1
for(i in 1:(C-1)){
    aa <- X[,i]
    for(j in (i+1):C){
        diff.temp <- (aa - X[,j])
        shift[cont] <- abs(min(diff.temp)) + 1
        Z[,cont] <- (diff.temp + shift[cont])
        labs[cont] <- paste(ord[i], ">", ord[j])
        cont <- cont+1
    }#end_for-j
}#end_for-i
#- optimum lambda's for 1st transformation
lam    <- apply( X,2, FUN=myBoxcox, interval=lam.range, loglikCI=logLikCI )
lam.pd <- apply( Z,2, FUN=myBoxcox, interval=lam.range, loglikCI=logLikCI )
#- Pairwise Data tranformation
Y <- array(NA,dim=dim(Z))
```

```
for(i in 1:K){
    if(lam.pd[i] == 0){
        Y[,i] <- log(Z[,i])
    } else{
        Y[,i] <- transf( y=Z[,i] , la=lam.pd[i])
    }#end-if_else
}#end-for-1:K
for(i in 1:C){
    if(lam[i] == 0){
        X.t[,i] <- log(X[,i])
    } else{
        X.t[,i] <- transf(y=X[,i], la=lam[i])
    }#end_if-else
}#end_for-1:C
#- confidence intervals
for(i in 1:K){
    #-> Pairwise Differences C.I.
    z.temp <- Y[,i] ; z0 <- z.temp[1]
    if(lam.pd[i] == 0){
        var.z <- var(z.temp) #
        z.err <- qnorm(1 - alpha/(2 * K)) * sqrt(var.z) #
        CIa[,i] <- z0 + c(-1, 0, 1) * z.err
        CI.pd[,i] <- exp( CIa[,i] ) - shift[i]
    } else{
        var.z <- var(z.temp) #
        z.err <- qnorm(1 - alpha/(2 * K)) * sqrt(var.z) #
        CIa[,i] <- ci.temp <- z0 + c(-1, 0, 1) * z.err
        ci.temp[(ci.temp * lam.pd[i] < -1)] <- -1/lam.pd[i]
        CI.pd[,i] <- transf.inv(z=ci.temp, la=lam.pd[i]) - shift[i]
    }#end-if_else-p.d.
    #-> Single scores C.I.
    if(i <= C){
        z.temp <- X.t[,i] ; z0 <- z.temp[1]
        if(lam[i] == 0){
            var.z <- var(z.temp) #
            z.err <- qnorm(1 - alpha/(2 * C)) * sqrt(var.z) #
            CIb[,i] <- z0 + c(-1, 0, 1) * z.err
            CI.s[,i] <- exp( CIb[,i] )
        } else{
            var.z <- var(z.temp) #
            z.err <- qnorm(1 - alpha/(2 * C)) * sqrt(var.z) #
            CIb[,i] <- ci.temp <- z0 + c(-1, 0, 1) * z.err
            ci.temp[(ci.temp * lam[i] < -1)] <- -1/lam[i]
            CI.s[,i] <- transf.inv(z=ci.temp , la=lam[i])
```

```
        }#end_if-else-single
    }#end_if-C
}#end-for
resCI <- list( CI=CI.pd, C=C )
res <- ranking4(resCI)
res[ord] <- res
CI.s[,ord] <- CI.s
#- ranking of conf.intervals
#res <- as.integer( (CI.pd[1,] < 0)&(CI.pd[2,] > 0) )
#- plotting conf.intervals
if(plotIt == TRUE){
    #-> C.I. for p.d.
    x11()
    cols1 <- rainbow(K) ; cols1[3] <- "darkgray"
    yLim1 <- range(CI.pd)
    plot(CI.pd[1,], ylim=yLim1, xlim=c(1,K+(K %/% 3)), type="n",
         main = "C.I. for pairwise differences of scores", xaxp=c(1, K, K-1))
    for(i in 1:K){
        lines(x=c(i,i), y=CI.pd[c(1,3),i], lwd=2, col=cols1[i])
        points(x=c(i,i,i), y=CI.pd[,i],pch="-", cex=2.5, col=cols1[i])
        #abline(h=CI.pd[2,i], lty=3, col=cols1[i])
    }#end-for
    abline(h=0, col=1, lwd=2)
    legend(x="topright", legend=labs, col=cols1, lty=1, lwd=2)
    #-> C.I. for single scores
    x11()
    cols2 <- c( 1, seq(2, 2 * C-1, by=2) )
    yLim2 <- range(CI.s) ; yLim2[2] <- yLim2[2] + 0.1*(yLim2[2]-yLim2[1])
    plot(CI.s[1,], ylim=yLim2, type="n", xlim=c(1,C+(C %/% 3)),
         main = "C.I. for scores and their ranking", xaxp=c(1,C,C-1))
    for(i in 1:C){
        lines(x=c(i,i), y=CI.s[c(1,3),i], lwd=2, col=cols2[i])
        points(x=c(i,i,i), y=CI.s[,i], pch="-", cex=2.5, col=cols2[i])
        abline(h=CI.s[2,i], lty=3, col=cols2[i])
    }#end-for
    labs.rank <- paste( as.character(res), "°", sep="" )
    text(x = 1:C, y = CI.s[3,], labels = labs.rank, pos=3)
    legend(x = "topright",
        legend = paste( as.character(1:C), " = ", labs.rank, sep=""),
        col = cols2, lty=1, lwd=2)
}#end-if_plotIt
if(showCI == TRUE){
    return( list( Ranking = res, CI.pd = CI.pd, lambda.pd = lam.pd,
                  CI.s = CI.s, lambda = lam, labs = labs) )
```

```
}else{
    return(Ranking=res)
}#end-if_showCI
}
```

– *NPC score-statistics calculation with synchronized permutation approach*

```
perm.npc1 <- function(datas, gr, randSeed=123, B=1000){
gr <- as.factor(gr) # if it's still not a factor
tab <- table(gr) ; C <- length(tab) ; p <- ncol(datas)
nData <- nrow(datas) ; n <- tab[[1]]
K <- C*(C-1)/2
sig <- rep(NA,p) ; mu <- rep(NA, C)
labs <- character(K)

#-> Matrixes of the p.values resulting from t.test, results of NPC and
#   datas re-allocated for pairwise comparisons
#   every layer contains all informations of a single comparison
T <- array(NA, dim=c( B+1, p, 2*K ))
X <- array(NA, dim=c( 2*n, p, K ))

#-> Fisher combining function
Fish <- function(x){ -2 * sum(log(x), na.rm=TRUE) }

#-> Function for applying Fisher's NPC to the
#   pairwise comparisons p.values matrix
Fcomb <- function(Pv, ng = C, fun = Fish){
    p <- nrow(Pv) ; K <- ncol(Pv)/2
    P.f <- array(NA, dim=c(ng, ng, p))
    M <- array(NA, dim=c(ng, ng))
    for(j in 1:p){
        M[lower.tri(M)] <- Pv[j, 1:K]
        M <- t(M)
        M[lower.tri(M)] <- Pv[j, (K + 1):(2 * K)]
        P.f[,,j] <- M
    }#end_for-j
    res <- apply(P.f, 1, FUN=fun)
    return(res)
}#end_Fcomb

#-> Contrasts vector
vContr = c(rep(1/n,n),rep(-1/n,n))

#-> Matrix permutated indices
set.seed(randSeed)
indMat <- apply(matrix(1:(2*n), nrow=2*n, ncol=B-1), 2, FUN=sample,
replace=FALSE)

#-> Observed statistics and reallocation of datas
cont <- 1
for(i in 1:(C-1)){
    dd1 <- datas[(gr == i),]
    for(k in (i+1):C){
        dd2 <- datas[(gr == k),]
        for(j in 1:p){
            X[,j,cont] <- c( dd1[,j], dd2[,j] )
        }#end_for-p
        cont <- cont+1
    }#end_for-k
}#end_for-i
Ttemp <- apply(vContr * X, c(2,3), FUN=sum)
T[1,,] <- cbind(Ttemp, -Ttemp)
```

```
#-> Permutation statistics
# set the seed for rand. num. gen.
for(bb in 2:B){
    ind <- indMat[,(bb - 1)]
    temp <- X[ind,,]
    Ttemp <- apply(vContr * temp, c(2,3), FUN=sum)
    T[bb,,] <- cbind(Ttemp, -Ttemp)
}#end for-bb

#-> Last permutation = observed
T[(B + 1),,] <- T[1,,]

#-> Passing to p.values
P <- t2p(T)

#-> Re-aggregating results for Fisher's NPC
risF <- t( apply(P, 1, FUN=Fcomb, fun=Fish, ng=C) )

#-> Non-Parametric Combination with Fisher
attr(risF, "dimnames")[[2]] <- paste(levels(gr),"> All")

return( Fish.bs = risF[(1:B),] )
}
```

Bibliography

AISE (2009), A.I.S.E. detergent test protocol 2009. WWW page, URL www.aise.eu/aisedetergenttestprotocol2009/.

Arboretti Giancristofaro R., Corain L., Gomiero D., Mattiello F. (2010), Nonparametric Multivariate Ranking Methods for Global Performance Indexes, *Quaderni di Statistica*, to appear.

Bonnini S., Corain L., Cordellina A., Crestana A., Musci R., Salmaso L., (2009), A Novel Global Performance Score with Application to the Evaluation of New Detergents, in *Statistical methods for the evaluation of educational services and quality of products*, Editors: M. Bini, P. Monari, D. Piccolo, L. Salmaso, Contribution to Statistics, Physica-Verlag: Heidelberg, pp. 161–179.

Box G.E.P., Cox D.R. (1964),. An Analysis of Transformations. *Journal of the Royal Statistical Society-Series B*, 26:211–252.

Draper N.R., Cox D.R. (1969), On Distributions and their Transformation to Normality. *Journal of the Royal Statistical Society-Series B*, 31:472–476.

Draper N.R., Smith H. (1998), *Applied Regression Analysis* – 3rd Edition. Wiley Series in Probability and Statistics, New York.

Fayers P.M.,. Hand D.J. (2002), Casual Variables, Indicator Variables and Measurement Scales: an Example from Quality of Life. *Journal of the Royal Statistical Association*, 165:1–21.

Folks J.L. (1984), Combination of Independent Tests. In *Handbook of Statistics*, 4, Krishnaiah P. R. and Sen P. K. eds. 113-121, Elsevier Sciences.

Gupta S.S., Panchapakesan S. (2002), *Multiple Decision Procedures* – 2nd Edition. Classics In Applied Mathematics. Siam, New York.

Hinkley D.V. (1975), On Power Transformations to Symmmetry. *Biometrika*, 62:101–111.

Pesarin F., Salmaso L. (2010), *Permutation tests for complex data: theory, applications and software*. Wiley, Chichester.

Westfall P.H., Tobias R.D., Rom D., Wolfinger R.D., Hochberg Y. (1999), *Multiple Comparisons and Multiple Tests using the SAS*® (SAS Books by Users).