Auteursrechterlijke overeenkomst

Opdat de Universiteit Hasselt uw eindverhandeling wereldwijd kan reproduceren, vertalen en distribueren is uw akkoord voor deze overeenkomst noodzakelijk. Gelieve de tijd te nemen om deze overeenkomst door te nemen, de gevraagde informatie in te vullen (en de overeenkomst te ondertekenen en af te geven).

Ik/wij verlenen het wereldwijde auteursrecht voor de ingediende eindverhandeling met

Titel: Statistical analysis of haptic data
Richting: master in Biostatistics   Jaar: 2008

in alle mogelijke mediaformaten, - bestaande en in de toekomst te ontwikkelen - , aan de Universiteit Hasselt.

Niet tegenstaand deze toekenning van het auteursrecht aan de Universiteit Hasselt behoud ik als auteur het recht om de eindverhandeling, - in zijn geheel of gedeeltelijk -, vrij te reproduceren, (her)publiceren of distribueren zonder de toelating te moeten verkrijgen van de Universiteit Hasselt.

Ik bevestig dat de eindverhandeling mijn origineel werk is, en dat ik het recht heb om de rechten te verlenen die in deze overeenkomst worden beschreven. Ik verklaar tevens dat de eindverhandeling, naar mijn weten, het auteursrecht van anderen niet overtreedt.

Ik verklaar tevens dat ik voor het materiaal in de eindverhandeling dat beschermd wordt door het auteursrecht, de nodige toelatingen heb verkregen zodat ik deze ook aan de Universiteit Hasselt kan overdragen en dat dit duidelijk in de tekst en inhoud van de eindverhandeling werd genotificeerd.

Universiteit Hasselt zal mij als auteur(s) van de eindverhandeling identificeren en zal geen wijzigingen aanbrengen aan de eindverhandeling, uitgezonderd deze toegelaten door deze overeenkomst.

Ik ga akkoord,

EKWEMPE EBWEKOH, Clifford

Datum: 20.11.2008

Statistical analysis of haptic data

Clifford Ekwempe Ebwekoh

promotor :
Prof. dr. Chris RAYMAEKERS

universiteit
►►hasselt

Statistical analysis of haptic data

by

Ekwempe Ebwekoh Clifford

Internal Supervisor**:** Philippe Haldermans

External Supervisors: Prof Dr Chris Raymaekers & Dr Joan De Boeck

Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Biostatistics.

7 November 2007

Statistical analysis of haptic data

by

Ekwempe Ebwekoh Clifford

Accepted by:

Internal Supervisor: Philippe Haldermans

Signature ……………………..   Date ………………………

External Supervisor: Prof. Dr. Chris Raymaekers & Dr.Joan De Boeck.

Signature ……………………..   Date ….……………………

Thesis submitted in partial fulfillment of the requirements for the degree of
Master of Science in Biostatistics.

7 November 2007

ACKNOWLEDGEMENT

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

The science of applying touch (tactile) sensation and control to interaction with computer applications is called haptic. In this study interest was to measure the effect of haptic sensation on the evaluation of virtual prototypes. In this context, forces are generated and felt by the user and this is done through haptic algorithms. The aim of this report was to investigate which statistics should be used to compare algorithms.

The data is collected from an experiment conducted in the laboratories of University of Hasselt more precisely at the EDM (Expertise Centrum for Digital Media). Three virtual objects (an orb, fish and a cube) were used. First a reference algorithm is used where the position and velocity are recorded, secondly a set of 6 algorithms are used producing outputs on which comparisons can be made. Each algorithm is used on each object 50000 times resulting to 900 000 observations in total. Performance of each algorithm was characterized using execution time and the surface contact points (SCP).

Analysis of variance (ANOVA) was implemented to the data set in order to investigate difference in the performance of the different algorithms on the different objects. Multiple comparisons were performed to investigate differences between all 6 algorithms. Multivariate normal regression was also implemented to investigate the performance based on both the SCP and execution time together, thereby taking into account the correlation between the multiple responses. Due to departures from normality, generalized estimating equations with a gamma distribution for the errors were further applied.

All models indicated that the algorithms performed differently, and the performance depends on the object on which the algorithm is being applied on.

Key Words: *Haptic algorithm, Analysis of variance, Multivariate analysis of variance, multiple comparisons, generalized estimating equations.*

1. INTRODUCTION

1.1 Background

In humans and many other primates, the visual system plays the major role in object recognition. But objects can also be recognized through haptic exploration: Haptic is the science of applying touch (tactile) sensation and control to interaction with computer applications. The evaluation of new computer devices can provide information for the future development of innovations, as well as for integration of these new devices into existing systems and practices [9].

This study measures the effect of haptic sensation on the evaluation of virtual prototypes. Virtual Reality provides new options for conducting motor assessment and training within computer-generated 3 dimensional environments [10]. To date very little has been reported about normal performance in virtual environments which uses our sense of touch. It has been argued that the haptic system makes use of 'visual' processing to construct a representation of the object. Now our days, research has been carried out to materialize haptic world: When playing some video games for example, these devices give users feedbacks as vibrations when users shot the gun or being hit by the enemy. Force feedback is produced by force feedback devices which give us some feedback as a type of force during the interaction between the pointer and the virtual object. There are many types of devices: game pads, joysticks, gloves, steering wheels, trackball, mice, phantom etc [7].

To investigate how the haptic device Phantom in this case allows users to experience the feel of force feedback in order to convey information between the user and the virtual environment, tools have been used and one of the commonly used tool is the haptic algorithm or force feedback algorithm [5].

Haptic algorithm is a two-fold algorithm. Its input is the current position and velocity of the pointer, as defined by the force-feedback device, and a virtual object, which has to be

rendered. The first part is a collision-detection step, which calculates whether the pointer position is located inside the object. The second step, called the render step, calculates the surface contact point (SCP) and the force that should be exerted by the force feedback device. These algorithms are highly interactive: the forces that are sent back commonly called force feed back must be calculated in real time by the user in order to achieve realistic results [9].

Haptic algorithms are been evaluated by techniques such as GHOST haptic API used by Acosta and Temkin: they looked at the implementation of polygonal meshes by loading objects with as many triangles as possible, until the system was not able to process the information within the haptic loop. Another commonly means of measuring an algorithm's performance is to use a tool which measures the haptic load (this is the processor time spent in the haptic loop to calculate the forces) [6].

Over the past few years, haptic algorithms have advanced considerably and it should be noticed that no standard evaluation method exits, which allows researchers to compare their results in a clear and objective manner. Nevertheless, researchers are allowed to apply evaluation methods that help them to develop these haptic algorithms.

In this study, three virtual objects were used: fish, cube and orb. First, a reference algorithm was used to record the haptic interaction with these virtual objects. The reference algorithm records the position of the cursor and the velocity of the cursor. A second algorithm is given the position and velocity of the reference algorithm which produces as output: surface contact point, whether there is collision or not, that is whether the pointer held by the user is being located inside the object or not. Also the calculation time (how long it takes for a calculation) called the rendering (or execution) time is determined and last but not the least as output we have the force exerted by the force feedback device and the speed of the surface contact point.

Performance of each algorithm was characterized using rendering time and the surface contact points (SCP). The basic interest of this study was based on the outputs produced

by each algorithm separately and to compare the results produced by these algorithms to evaluate the performance and correctness of force feedback and to formulate statistics to compare these individual algorithms and their components. A secondary objective was the implementation of a small function in R/SPLUS for comparing the performance of 2 algorithms.

The rest of the report is structured as follows: a brief description of the data is presented in Section 2, the statistical methodology used to analyze the data is described in Section 3. The results are presented in Section 4 followed by Discussion and Conclusion in Section 5.

2.   DATA

2.1   Experimental setting

The experiment was performed in the laboratories of University of Hasselt more precisely at the EDM (Expertise Centrum for Digital Media). Three virtual objects (an orb, fish and a cube) are measured with several algorithms. First a reference algorithm is used where the position and velocity are recorded. This reference algorithm produces the position of the cursor and the velocity (speed) of the cursor.

Secondly, a different algorithm is used where the previous path is being replayed follow by recording the data that is being produced. The latter algorithm produces as output Surface Contact Point (SCP) which is determined using three dimensions (*SCPx, SCPy* and *SCPz)*. The second algorithm also calculates whether the pointer position is located inside the object (*Colfound)*. In case the pointer is located inside the object, it is assigned the value 1, otherwise it is recorded as 0. The time needed for an algorithm to perform the calculation known as the rendering time is determined (*Time*). The force produced by the device called force feedback and which is perceived by the user is also measured (*Force*). Finally the velocity of the surface contact point is determined.

2.2   Variables

2.2.1 Original variables

The following variables were measured during or after the experiment.
- *Loop*: this is an identity variable representing each experience carried out.
- *Node ID:* is the person carrying out the experience
- *Time* :also called the rendering time, it is the time needed to execute one haptic loop.
- *Colfound:* which is a binary variable calculates whether the pointer position is located inside the object or not. In case it is located inside, colfound is marked as 1 and zero otherwise.

- *Surface Contact Point* (**SCP**): is the point on the object's surface to which the user pointer should be brought, by applying forces on the haptic device in order to feel a realistic force it is in 3 D.
- *ForceX*, *ForceY* and *ForceZ:* represent the effect produced by the device. This force is also in 3D.
- *Algo:* is the algorithm studied, there are 6 of them. These algorithms are coded as follows:
    1. *Log-HAL*
    2. *Log-HAL-AABB*
    3. *Log-HAL-SPATTLASH*
    4. *Log-HAL-SPHERECELL*
    5. *Log-CHAI*
    6. *Log-CHAI AABB*

- *Object*: which is the virtual object (Cube, fish and Orb) been studied. These objects are coded as:
    1. *Cube*
    2. *Fish*
    3. *Orb*

According to the objective of the study, certain variables were created:
- *Company:* is the company from which these algorithms are issued. We have 2 companies (**HAL** and **CHAI** companies) involved in this study and 4 algorithms coming from HAL-Company and 2 from CHAI-Company.
- *M_SCP:* is the pooled SCPs (*SCPx*, *SCPy* and *SCPz*) and represents the magnitude of the three SCPs. It is obtained as follows:

$$M\_SCP = \sqrt{(SCPx)^2 + (SCPy)^2 + (SCPz)^2}$$

- *Force:* represents the pooled forces *ForceX*, *ForceY*, and *ForceZ.*

3. METHODOLOGY

In this experiment on haptic algorithms, the interest was to determine and to compare algorithms based on two characteristics: the time for calculation (velocity) for these algorithms and their correctness which is determined through the surface contact point (position). After a reference algorithm is been used to record the data, a second one is used with the velocity and position determined by the first one. The interest is on the output of the second one which produces amongst other: the rendering time, the force produced by the Phantom device, the surface contact point, whether the pointer is positioned inside the object or not…The main issue of this analysis is to compare between algorithms based on the performance (time) and correctness (SCP). Before the implementation of statistical models, it is first of all advisable to explore the data and from the results obtained from this data exploration, a model or series of models will be deduced.

3.1 Exploratory Data Analysis

3.1.1 Summary Statistics

Due to the large nature of the data set and with the intention to communicate as much as possible and as simply as possible the data was summarized and described through measures of location (mean, median …). Also to have an idea on the dispersion of the values in data, standard deviation, variance, and range were determined.

3.1.2 Correlation

The knowledge of correlation structure is fundamental when dealing with models in which the response variables are correlated. The correlation between variables was also studied to measure the association or relation between these variables.

3.2   Statistical Models

In this analysis, focus was made on the variables related to the research question in order to give an attempt of answers. In this perspective, we considered the Analysis of Variance (ANOVA) model since the comparison of the algorithms was based on the performance (*time*) and correctness (*SCP*). The consideration of ANOVA was to answer the question on how time and the Surface Contact Point (*SCP*) could be influenced by different algorithms and other factors such as the force produced by the device or the object used.

The multivariate nature of the principal response variables was also considered since these variables were assumed to be correlated. This characteristic was taken into consideration by implementing models that take into account such pattern. We considered multivariate analysis of variance (MANOVA) for the 3 SCP variables only, then later the 3 SCP variables together with the *Time* variable. Finally we considered generalized estimating equations.

3.2.1 Analysis of Variance

The research question is to investigate if there is a significant difference amongst the 6 implemented algorithms. Comparisons were based first on the time needed for algorithms to perform calculations. Here we are dealing mainly with factors such as: *Algo, Object, Company* (with respectively 6, 3, and 2 levels) and to some extend the variable *Colfound*. Generally when we are dealing with a single factor with *r* levels, one approach is to construct a linear statistical model with *r-1* indicator variables as predictors. In the multi-factor model, there exist two ways of representing ANOVA models: cell means model and factor effects model. Depending on the objective preference is made of which method to choose [16]. In our case since we are investigating on the differences of algorithms we will focus our attention on the cell means methods and this method can be resumed as follows:

$$Y_{ijklm} = u_{ujklm} + \varepsilon_{ijklm} \qquad (1)$$

where:

Y_{ijklm} denote the observation for the mth case ($m = 1,...,n$) for the treatment consisting of the ith level of algorithm ($i = 1,...,a;$), the jth level of Object ($j = 1,...,b$), the kth level of Company ($k = 1,...,c;$) and the lth level of Colfound ($l = 1,...,d;$).

u_{ijklm} are parameters

ε_{ijklm} are independent $N(0,\sigma^2)$

3.3 Diagnostics

3.3.1 Normality

Many data analysis methods (t-test, ANOVA, normal-regression) depend on the assumption that data were sampled from a Gaussian distribution. One of the ways to evaluate how far data are from Gaussian is to look at a graph and see if the distribution deviates grossly from a bell-shaped normal distribution. This can be achieved using normal quantile plots (univariate normality). Also some tests exist to investigate normality for instance, the Kolmogorov-Smirnov test. This test enables one to check if a sample is from a given distribution [16].

3.3.2 Constancy of Variance

In general several tests are available for studying the constancy of the error variance as required by the ANOVA model we can name: the Harvey test and the Levene test. Although the Hartley test is simple it presents some drawbacks; it is applicable only when sample sizes are equal and the error terms normally distributed. In this work, the Levene test for constancy of variance was used, since it is less dependent on the normality assumption [16].

This test split the sample in two groups then compute $d_{i1} = | e_{i1} - \bar{e}_1 |$ and $d_{i2} = | e_{i2} - \bar{e}_2 |$ with $\bar{e}_1$ and $\bar{e}_2$ the medians. The statistics is a two sample t-test

$$t^*_L = \frac{\bar{d}_1 - \bar{d}_2}{S_P \sqrt{1/n_1 + 1/n_2}} \sim t(n-2) \qquad (2)$$

Where

$$S_P = \frac{\sum(d_{i1} - \bar{d}_1)^2 + \sum(d_{i2} - \bar{d}_2)^2}{n - 2} \qquad (3)$$

3.4 Remedial Measures: Box-Cox Transformation

The aim of the Box-Cox transformations is to ensure that the usual assumptions for Linear Model hold [13], $y \sim N(X\beta, \sigma^2 I_n)$. Clearly not all data could be power-transformed to have a normal distribution. Draper and Cox (1969) studied this problem and concluded that even in cases that no power-transformation could bring the distribution to exactly normal the usual estimates of λ will lead to a distribution that satisfies certain restrictions on the first 4 moments, thus will be usually symmetric. The original form of the Box Cox transformation here has the following form :

$$y(\lambda) = \begin{cases} \dfrac{y^\lambda - 1}{\lambda} & if \ \lambda \neq 0 \\ \log(\lambda) & if \ \lambda = 0 \end{cases} \qquad (4)$$

3.5 Multiple Comparisons

Generally, estimating and testing factor level means present some limitations. The specified Type I error rate, α is applied uniquely to a particular test but to a series of tests. In analysis of variance models, this limitation is crucial because frequently many different comparisons are of interest. Often, it is of interest to piece the different findings together. The second limitation while processing to test factor levels with one single test is that the confidence coefficient and 1-α and the significance level α are valid only if the estimate or test was not suggested by the data. A suggested solution to overcome these limitations is the use of multiple comparison procedure where the family of inferences includes all the possible inferences that can be estimated as important [16].

Generally when interest of analysis is on the mean difference between more than two items, the analysis of variance is the model which is implemented. When the ANOVA model has fulfilled the assumptions of normality and constancy of variance, to determine which pairs are significantly different, multiple comparisons is the candidate model to

further the analysis. This technique permits to compute all pair wise comparisons and determine which pairs are significantly different. There exist many type of multiple comparison method: Tukey, Bonferroni and Scheffe among other.

Tukey multiple comparison, is applied when the family of interest is a set of all pair wise comparisons of factor level means. The family consists of estimates of all pairs $D = \mu_i - \mu_{i'}$ or of all tests of the form [16]:

$$H_0 : \mu_i - \mu_{i'} = 0$$
$$H_a : \mu_i - \mu_{i'} \neq 0$$

The Tukey multiple comparison confidence limits for all pairwise comparison $D = \mu_i - \mu_{i'}$ with family confidence coefficient of at least $1 - \alpha$ are as follows:

$$D \pm Ts\{D\}$$

Where: $D = \bar{Y}_{i.} - \bar{Y}_{i'.}$

$$s^2\left\{\bar{Y}_{i.}\right\} + s^2\left\{\bar{Y}_{i'.}\right\} = MSE\left(\frac{1}{n_i} + \frac{1}{n_{i'}}\right) \quad (5)$$

$$T = \frac{1}{\sqrt{2}} q(1 - \alpha; n_T - r) \quad (6)$$

Scheffe Multiple Comparison procedure is applied when the set of all possible contrasts among the factor level means [16]:

$$L = \sum c_i \mu_i \text{ where } \sum c_i = 0$$

The Scheffe confidence intervals for the family of contrasts L are of the form:

$$L \pm Ss\left\{\bar{L}\right\} \quad (7)$$

Where: $s^2 = (r - 1)F(1 - \alpha; r - 1, n_T - r)$ and $s^2\left\{\bar{S}\right\} = MSE\sum \frac{c^2_i}{n_i}$ (8)

Bonferroni multiple comparison: the interest is in a particular (finite) set of differences between two factor level mean. The set is specified by the user, before the start of the statistical analysis. If there are g parameters in the set, each parameter is written as [16]:

$$L_k = \sum_{j=1}^{r} d_{k_j} \mu_j \text{ , where } (d_{k_1},...,d_{k_r}) \text{ is a given vector, k=1,...,g.}$$

It is good to note that:

$$\bar{L} = \sum_{j=1}^{r} d_{k_j} \bar{Y}_{j.} \text{ and } S^2(\bar{L}_k) = MSE \sum_{j=1}^{r} \frac{d^2_{k_j}}{n_j} \qquad (9)$$

The Bonferroni inequality implies that the confidence limits to be:

$$\bar{l}_k \pm B.s\left(\bar{l}_k\right), \text{ where } \bar{l}_k, \ S^2(\bar{l}) \text{ are the actual values of } \bar{L}_k, \ S^2(\bar{L}) \text{ with}$$

$$B = t(1 - \frac{\alpha}{2g}; n_T - r) \qquad (10)$$

where g is the number of comparisons to be made.

3.6 Multivariate Regression

3.6.1 Multivariate Model

The responses been correlated, the multivariate family of models considered here is allowed to be implemented to this data set of the haptic algorithm. This multivariate family of model can be formulated as follows:

$$g(y) = X\beta + \Sigma \qquad (11)$$

where g is the link function, X is the matrix of covariates, β is the vector of parameter estimates and Σ is the covariance matrix of the error terms.

3.6.1.1 *Multivariate Analysis of Variance*

Since the responses were measured for the same run of the experiment (*loop*), there were likely to be correlated. PROC CORR in SAS was used to obtain the Spearman correlation matrix as a measurement of the sample association of the response variables. To handle correlated data, Multivariate Analysis of variance (MANOVA), a generalization of ANOVA to the case of multiple response or dependent variables would be a better choice since MANOVA permits to take into account relations present among the response variables, whereas ANOVA does not allow such relationships/complexities. MANOVA

was employed to investigate whether algorithms differ by considering firstly the three *SCPs* and secondly the three *SCPs* and *time* together.

3.6.1.2 Diagnostic for Multivariate Normality

To assess the multivariate normality, a chi-square or gamma plot was implemented to the data to see if it deviates or not from normality. On this plot, normality is detected if approximately there is a straight line that passes through the origin with slope 1. To construct the Chi-square plot, first the square distance are determined then ordered from smallest to largest as: $d^2_{(1)} \leq d^2_{(2)} \leq ... \leq d^2_{(n)}$

Followed by plotting the graph of the pairs $(q_{c,p}(j-\frac{1}{2})/n), d_{(j)}^2)$, where $q_{c,p}((j-\frac{1}{2})/n)$

is the $100(1-\frac{1}{2})/n)$ quantile of the chi-square distribution with p degrees of freedom [15].

3.6.1.3 Generalized Estimating Equations (GEE)

The general linear model (GLM) used here had as estimating procedure the generalized estimating equations (GEE) [17,12] and systematic component:

$$\eta_{ik} = \sum_j \beta_j x_{ij}, \qquad\qquad i = 1,...,N. \qquad\qquad (11)$$

In matrix form,

$$\eta = X\beta \qquad\qquad (12)$$

Where $\eta = (\eta 1,...,\eta_N)'$, $\beta = (\beta_1,...,\beta_p)'$ column vectors of model parameters, and X is the $N \times p$ matrix of values of the explanatory variables for the *N* subjects.

The GLM links η_i to $\mu_i = E(Y_i)$ by a link function g(.). Thus μ_i relates to the explanatory variables by:

$$\eta_i = g(\mu_i) = \sum_j \beta_j x_{ij}, \quad i = 1.,,,.N. \qquad\qquad (13)$$

$$Y_{ik} = \mu_{ik} + \varepsilon_{ik} \qquad\qquad (14)$$

$$g(\mu_{ik}) = \eta_{ik} = \sum_j \beta_j x_{ijk} \qquad\qquad (15)$$

First we assume normal distribution and identical link to obtain the function

$$g(\mu_{ik}) = \mu_{ik}. \qquad (16)$$

Secondly, the gamma distribution is assumed together with inverse (power-1) link to obtain:

$$g^{*}(\mu_{ik}) = \eta_{ik} \qquad (17)$$

where g^{*} = inverse (power (-1)). For the generalized estimating equation, misspecification of the variance structure is allowed because this estimating equation has the possibility to correct for it as far as the mean structure is correctly specified [17,12].

4. RESULTS

4.1 Exploratory Data Analysis

4.1.1 Summary Statistics

From Table 1, we observe that the mean rendering time (*M_Time*) varies from one algorithm to another within a specific object. In other words, the rendering time for a given algorithm appears to depend on a particular object. When we consider for example the cube and fish virtual objects, the fastest algorithm for the Cube virtual object is algorithm 5 (with the smallest time) representing the Log-Chai algorithm, while algorithm 2 have the smallest mean rendering time when it comes to the Fish object. It should be noted that an algorithm is better if it is faster i.e has a smaller rendering time.

Table 1: Mean time, mean SCP and mean force by Algorithm and object

Object	Algo	M_SCP	M_Time	M_Force	M_SCPX	M_SCPY	M_SCPZ
Cube	1	0.073218	0.01981	1.00612	-0.011345	0.024285	0.016248
Cube	2	0.073218	0.01575	1.00612	-0.011345	0.024285	0.016248
Cube	3	0.073218	0.01659	1.00612	-0.011345	0.024285	0.016248
Cube	4	0.073218	0.02003	1.00612	-0.011345	0.024285	0.016248
Cube	5	0.072848	0.00705	9.32948	-0.011231	0.02432	0.016888
Cube	6	0.073054	0.00832	9.45408	-0.011375	0.024084	0.017
Fish	1	0.039969	0.15979	0.53507	0.002625	0.013142	0.016289
Fish	2	0.039969	0.02035	0.53507	0.002625	0.013142	0.016289
Fish	3	0.039969	0.0204	0.53507	0.002625	0.013142	0.016289
Fish	4	0.039933	0.02429	0.51243	0.002685	0.01315	0.016277
Fish	5	0.039773	0.07539	9.5016	0.002574	0.013239	0.015694
Fish	6	0.039773	0.02206	9.5016	0.002574	0.013239	0.015694
Orb	1	0.043856	0.94642	0.89441	-0.005996	0.002667	-0.00267
Orb	2	0.043856	0.02382	0.89441	-0.005996	0.002667	-0.00267
Orb	3	0.043856	0.02368	0.89441	-0.005996	0.002667	-0.00267
Orb	4	0.04251	0.01622	0.33383	-0.006189	0.002657	-0.002524
Orb	5	0.043352	0.24417	9.77764	-0.005391	0.002618	-0.002164
Orb	6	0.043299	0.01651	9.8736	-0.005633	0.002332	-0.002537

When it comes to the SCP, an algorithm is better if it has the highest value of M_SCP. This depends also on the algorithm and object being studied. When we take the Cube and the Orb objects, algorithms 1, 2, 3 and 4 have the same and the highest value of M_SCP while for the Orb object, algorithms 1, 2 and 3 have the same and highest value of

M_SCP. Thus, it appears that the value of M_SCP for each algorithm depends on the object on which the algorithm is being applied on.

4.1.2 Partial Correlation of (*SCPx, SCPy, SCPz, Time*) Residuals

This correlation matrix between the responses shows that these four variables are associated through their pair-wise correlation [Appendix Table A.2]. In this case, univariate regression may not be an appropriate procedure, to fully characterize the performance of the algorithms i.e. considering both *SCP* and *time*, since it ignored correlation between the four response variables. In this set up, multivariate analysis of variance (MANOVA) is an alternative to investigate primarily the algorithm effect on the response variables. Since in this study, interest was also to evaluate the impact of company and object on the response variable, a multi-way MANOVA will be suitable statistical technique in this study.

4.2   Statistical Analysis

4.2.1 General Linear Models (Multi-way ANOVA)

We can use either PROC GLM or PROC ANOVA to analyze a balance data indiscriminately. However, when the data is unbalanced PROC GLM is required because its efficiency is higher compared to PROC ANOVA. PROC ANOVA has a major draw back, it does not take the continuous covariates into account. It means continuous variables have to be dropped out of the model before implementing this procedure on data sets. This would lead to a loss of information contained in the original data.

Modeling factors and the continuous variables at the same time is well handled by the PROC GLM statement. The general F test of the model shows a *p*-value which is significant (<0.0001), suggesting that at least one of the variables is significant. We therefore obtain the following model:

Table 2: Type 3 tests of covariates from GLM, with Time as response

Variable	Df	MS	*p*-value
Algo	5	3669.8703	<0.0001
Object	1	3262.5979	<0.0001
Company	1	7.9265	<0.0001
Colfound	1	6.0103	<0.0001
Force	1	0.3744	0.0114
Algo*Object	10	155.7042	<0.0001

All these variables are significant including the *Algo*Object* interaction, which represents the interaction between *Algorithm* and *Object*. Thus, the speed of an algorithm depends on the object being used. Furthermore, all independent variables of interest play a major role in the determination of the rendering time.

Before any statistical inference or interpretation could be applied, it is necessary to check if this model fulfills the requirements of the analysis of variance model. That is to check if the error terms are normally distributed and also to check if there is constancy of variance. The normality assumption is investigated through the Kolmogorov-Smirnov test. The KS-test has the advantage of making no assumption about the distribution of data (Technically speaking it is non-parametric and distribution free.). This test gives a p-value (<0.0100) suggesting that the error terms are not normally distributed. This can also be seen from the Q-Q plot in Figure 1. Figure 1 shows a non-linear pattern in contrast to the linear pattern expected for normal distributed errors.
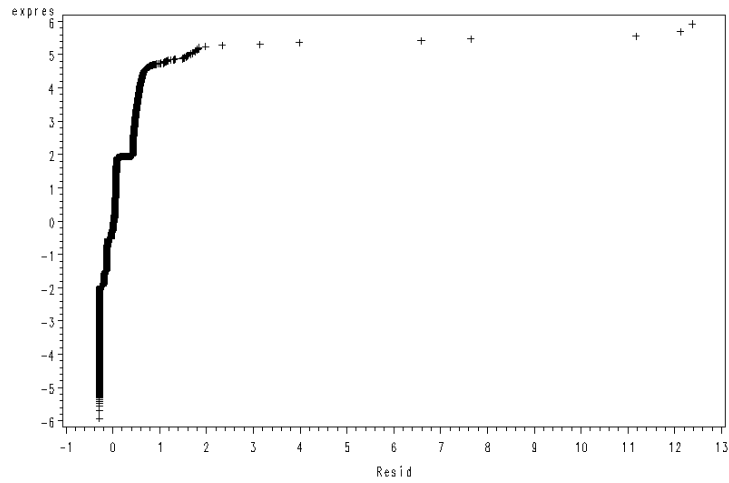
Figure 1: Normal probability plot for the model with Time as response

The constancy of variance will be investigated through the Levene test which is robust against departures of normality and does not requires balanced design.

A plot of residuals against predicted values can be seen in Figure 2, which shows a non constant pattern for all factors and the continuous variable *Force*. This test shows very significant p-values implying that the variance among the *Company*, *Colfound*, *Force* and *Object*Algo* groups are not constant [Appendix Table A.3]. The null hypothesis of equal variances is rejected and it is concluded that there is a difference between the variances in the population.
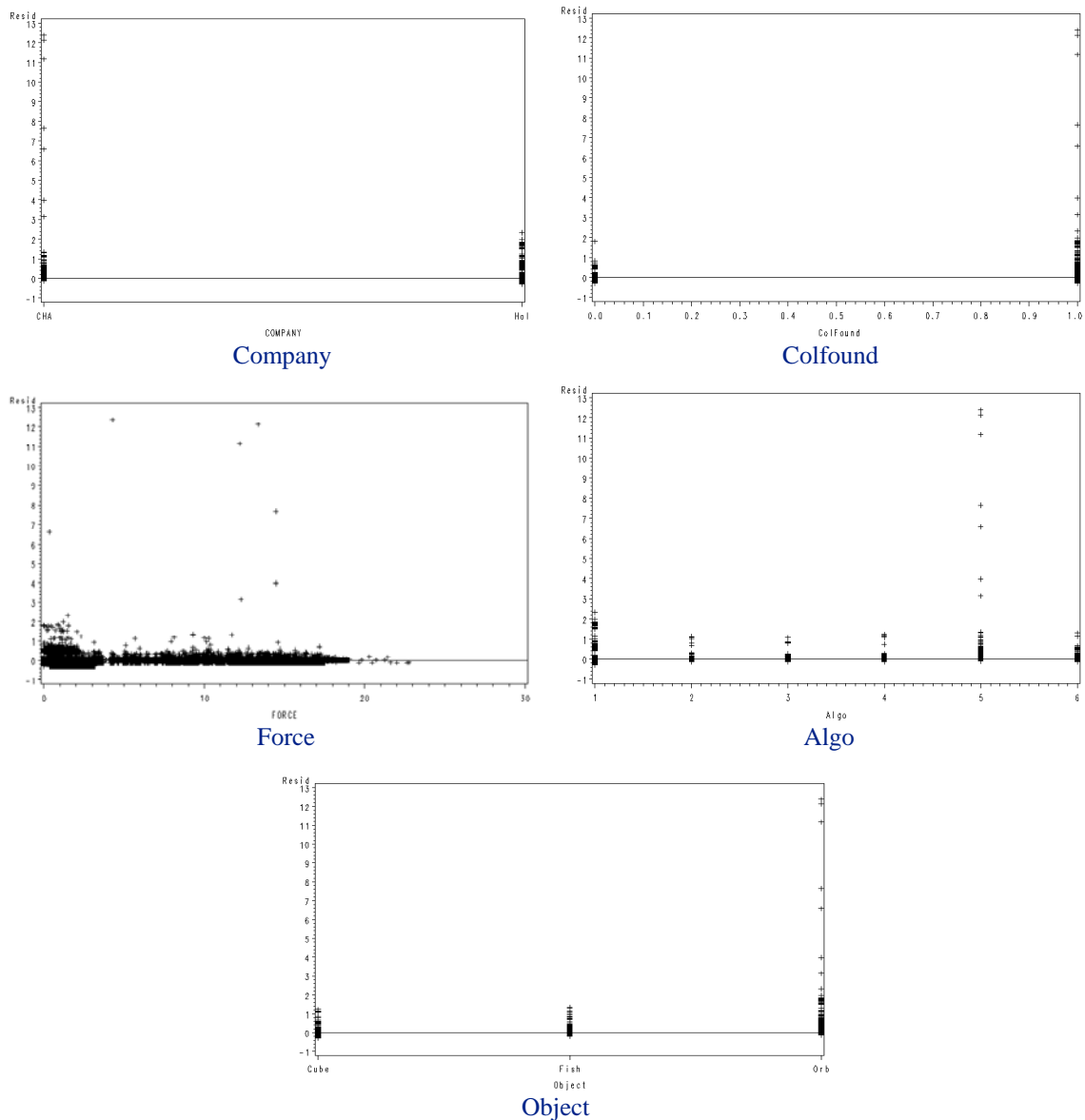
Figure 2: Plots of residuals vs. Predicted for non transformed Time

The error terms are not normal and do not have constant variance. It is of great importance to try to overcome these handicaps and one of the remedial measures to this situation is the transformation approach. The idea is to replace Y with transformed values, which will yield approximate normal errors with constant variance, then to model the transformed response in the ANOVA (GLM) model. Further test of significance and confidence intervals are performed on the transformed values scale.

The values of lambda (λ) family enabled us to derive the appropriate transformation to carry on. These values range between [-2, 2] and a general rule of thumb dictates us to choose the λ that minimizes SSE and since SSE depends on the value of λ; a "standardized" variable is used (that is the one that has the same SSE for all λ). The λ chosen is equals to zero (0) and implies to transform the response *Time* to *log(Time)* [Appendix, Figure A.4].

Table 3: Type 3 test for independent variables from GLM with transformed time response

Variable	Df	MS	*p*-value
Algo	5	141579.8700	<0.0001
Object	2	166582.1376	<0.0001
Company	1	4492.4700	<0.0001
Force	1	0.3981	0.0054
Colfound	1	77800.2062	<0.0001
Object*Algo	10	79137.8950	<0.0001

The model for the transformed response (*log Time)* shows that all the variables are again significant (Table 3). However, from Figure A.1 in the Appendix, we observe that even after transformation the *Force* shows a pattern indicating non-constancy of variance. The Levene test indicates that errors are heteroscedastic (Table 4). Also, the error terms are not normally distributed according to Kolmogorov-Smirnov, Cramer-von Mises, and Anderson-Darling tests (Table 4).

Table 4: Constancy of Variance and Normality tests for the transformed Time (log Time) model

Test	DF/Statistic	*p*-value
Algo	5	0
Object	2	0
COMPANY	1	0
Colfound	1	0
Force	1	0
Algo*Object	10	0
Kolmogorov-Smirnov	0.24	0.01
Cramer-von Mises	13305.98	0.005
Anderson-Darling	69368.45	0.005

Although transformation may lead to a model that satisfies model assumptions, it usually leads to difficulty in model interpretation. Since models based on the original response and transformed responses do not satisfy the model assumptions, it is therefore advisable to interpret the non transformed model.

According to this model, there is a significant difference between algorithms from the different companies. More importantly algorithm performance is conditioned on the virtual object. The force perceived by the user when using the device has an effect on the time used by the algorithm for the haptic device to generate the force. The time used by algorithm also depends if there is collision or not. From the parameter estimates, we can say that the force leads to more time (positive estimate).

In the research world of the haptic algorithm, the focus is not only to investigate which algorithm or set of algorithms performs best by relying only on the rendering time as it is known that the algorithm which requires less time to perform a calculation is the best: interest is also to determine the correctness of the algorithms. To check if an algorithm is correct, the surface contact point is most of the time use as principal tool. In the following analysis we will try to determine which algorithm is better, in terms of correctness. We will also be interested on factors that influence the correctness of an algorithm: the surface contact point (SCP).

Surface contact point is measured on a three dimensions, that is has three measures on the x, y and z axes. To perform analysis, first the magnitude of the three SCP was evaluated. So the three dimensions of a pointer (*SCPx, SCPy* and *SCPz*) were combined into one representing the magnitude, *M_SCP*, as indicated in Section 2.2. A general linear model shows that all the variables were significant (<.0001). The interaction *Algo*Object* is significant also (Table 5). Similary to rendering time, the correctness of an algorithm depends on the object being used. Furthermore, all independent variables of interest play a major role in the determination of SCP.

Table 5: Type 3 test results from GLM for the magnitude of SCP.

Variable	Df	MS	*p*-value
Algo	5	0.0060	<0.0001
Object	2	99.8760	<0.0001
Force	1	1.1759	<0.0001
Company	1	0.0093	<0.0001
Colfound	1	8.9007	<0.0001
Algo*Object	10	0.1253	<0.0001

To adapt the interpretation of this model, it is wise to investigate if it fulfils the conditions of normality of the error terms and constancy of variance of the error terms as well. The Q-Q plot of Figure 3 shows a non linear trend of the residuals suggesting a deviation from normality of the model.
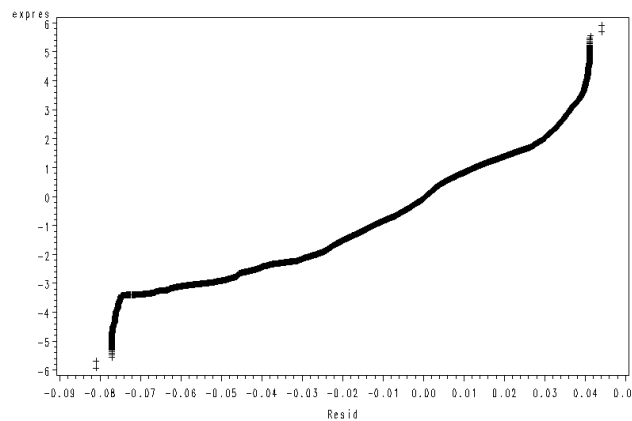
Figure 3: Normal probability plot for the model with magnitude of SCP as response

The errors terms from the M_SCP models were tested for normality and constancy of variance assumptions (Appendix Table A.4). This table shows that the null hypothesis of constancy of variance for *Algo, Object*, *Company*, *Colfound, Force* and *Algo*Object* are rejected therefore suggesting there is no constancy of variance of the error terms of these variables. This can also be observed from Figure A.2 in Appendix.

To overcome the non normality issue and non constancy of variance, an attempt to remediate to the situation will be done through Box Cox transformation that will permit us to generate a family of lambda with the smallest SSE. We obtain a $\lambda=0$ that

21

minimizes SSE [Appendix Table A.5]. From this $\lambda =0$, the response variable: magnitude of SCP is transformed to *Log(M_SCP)* followed by implementation of the ANOVA model through the general linear model. The obtained transformed model is as follows:

Table 6: ANOVA (Log(MagnitudeSCP) as response).

Variable	Df	MS	*p*-value
Algo	5	37.9691	<0.0001
Object	2	32059.8814	<0.0001
Company	1	18014.6154	<0.0001
Colfound	1	3849.6482	<0.0001
Force	1	716.4893	<0.0001
Algo*Object	10	4853.0386	<0.0001

Also here all of the obtained variables are significant, leading to similar inference as the previous model. However, the errors are not normally distributed and not heteroscedastic (Table 7). This transformed model does not fulfill the assumption of normality and constancy of variance [Appendix Figure A.3] therefore it is advisable to consider the non transformed model.

Table 7: Constancy of variance and normality tests for log (M_SCP) model

Test	Stat	*p*-value
Algo	5	0
Object	2	<0.0001
Company	1	0
Colfound	1	0
Force	1	0
Object*Inter	10	0
Kolmogorov-Smirnov	0.26	0.005
Cramer-von Mises	22420.07	0.001
Anderson-Darling	113889.69	0.0001

The objective of the research question was to compare if the different algorithms are different based on the summarized SCP. Although multiple comparisons are applicable when all of these assumptions hold, inferences can still be made because of the large nature of this sample. However inference should not be made on the model with the

transformed variable because its transformation did not remediate the situation of non normality and non constancy of variance. It should be made on the model that was not transformed. Another alternative would be to go for a non parametric analysis through the Kruskal-Wallis test. Non parametric methods are most of the time considered in situations were normality assumptions don't hold. Unfortunately it has draw backs when considering large sample size which is the case here.

The fundamental preoccupation now is to know how these algorithms differ amongst each other. An attempt of answer can be given by implementing multiple comparisons to the Analysis of variance model this will be done in the following section. It is worth to remind that the multiple comparison method is based on the non transformed ANOVA model although it does satisfied the assumptions of ANOVA model.

4.2.2 Multiple Comparisons

4.2.2.1 Multiple Comparisons taking Time as Response

In this section, an attempt will be done to see how these algorithms differ taken into account the object used to carry on the experience and also investigation on the difference between companies.

In a one-way or two-way analysis of variance, to compare the means of several groups to test the hypothesis that they are all the same, against the general alternative that they are not all the same. Sometimes this alternative may be too general. So we may need information about which pairs of means are significantly different, and which are not. A test that can provide such information is called a *multiple comparison procedure*. It should be mentioned that there are many procedures for multiple comparison amongst which we can name the Scheffe, Tukey, Bonferroni. We will use in this analysis the Scheffe procedure because of the fact that it presents certain advantages over the others when we stick on the research questions which is to investigate if there is difference between the 6 algorithms being studied in the first place then secondary to investigate this difference when considering the virtual objects and companies designing these algorithms. Based on these objectives, the choice is carried on Scheffe because the interest here is on pairwise comparisons. Although Tukey is a possible candidate here

because it gives narrower confidence limits but has one major inconvenient: it looses power when it comes to data snooping whereas the Scheffe procedure is robust against this issue. [16].

We obtain from the multiple comparison output, the table comparing companies. From this outputs when considering the variable company we see that there is a significant difference between the two companies. According to *Table 8*, there is a significant difference between the two companies producing algorithms. The CHA-company produces algorithms with lesser time of calculation comparing to the HAL-company suggesting that algorithms from CHA-company is better than the algorithm from HAL company.

Table 8: Multiple comparison between companies base on Time

Company comparison	Difference between means	Simultaneously 95% confidence interval	
HAL-CHA	*0.04668*	(0.04668; 0.04668)	***

*** comparisons significant at the 5% level of significance

Comparing between objects, when Time is the response, we observe according to the output that all of the 3 pairwise comparisons are significant. The amount of time of calculation by Algorithm depends on the object used for this haptic device experiment. Through the output of the Scheffe multiple comparison [Appendix Table A.10]. This table shows that objects are significantly different from each other. The following ordering was deduced : The object Cube was the object with less time of calculation by algorithm, then followed by the object fish lastly the Orb object when used required more time for algorithm to perform calculation [Appendix Table A.9].

The main focus of this study was to compare algorithm performance based on the time of calculation. From the output of the Scheffe multiple comparison and the mean calculation of time for each algorithm [Appendix Tables A.7] the following ordering was possible when considering time as response. Table 9 shows that globally the six algorithms are

subdivided into 3 subgroups. The first consist of one algorithm representing the one with smallest rendering time followed by the second group which has 3 algorithms with closely similar rendering time then the last group with 2 algorithms representing algorithm with the highest amount of time of calculation therefore the worst algorithms.

Table 9: Ordering for algorithms with Time as response

Name of algorithm	Ordering from best to worst
6-Log-Chai AABB	Smallest time of calculation
2-Log-Hal-AABB 4-Log-Hal-SphereCell 3-Log-Hal-Spattlash	Second best
5-Log-Chai 1-Log-Hal	The worst of the 6 algorithms

4.2.2.2 Multiple Comparisons with Magnitude of SCP as Response

The main focus in this study was to compare between algorithms based on the rendering time performed by each algorithm; interest was also to determine the correctness through the SCP. Based on the surface contact point, the two companies are different and the HAL company is the best company because has algorithms that permit to attain the bigger surface contact point (see Table 10).

Table 10: Multiple comparisons between companies based on magnitude of SCP

Company comparison	Difference between means	Simultaneously 95% confidence interval		
HAL-CHA	6.52651E-7	5.22043E-7	7.8326E-7	***

*** Comparisons significant at the 5% level of significance

While considering the surface contact point, the Cube object was the best because required much surface followed by Orb and lastly the object Fish was the one with smallest SCP [Appendix Table A.9].

Comparison of the algorithms based on the surface contact point, was performed with the help of Scheffe multiple comparisons and the table of the mean SCP [Appendix Tables A.8 & A.13] give the following ordering.

Table 11: Multiple comparisons between Algorithms based on magnitude of SCP

Name of algorithm	Ordering from best to worst
4-Log-Hal-Sphere 5-Log-Chai 6-Log-Chai-AABB	Biggest Magnitude of SCP
3-Log-Hal-Spattlash 2-Log-Hal-AABB 1-Log-Hal	Smallest Magnitude of SCP

We have two sets of algorithms, the first group representing the ones which have approximately the same SCP they consist of Log-Hal-Sphere, Log-Chai, Log-Chai-AABB: they perform better than the second set of algorithms who also have on a general note the same SCP they consist of Log-Hal-Spattlash, Log-hal-AABB and Log-Hal (see Table 11).

4.2.3 Multivariate Analysis

4.2.3.1 Considering SCPx, SCPy, SCPz as Multivariate Responses

When there are multiple dependent variables in a design, the design is said to be multivariate. Multivariate measures of association are by nature and require more complex analysis than their univariate counterparts (such as the correlation coefficient, for example). This is because multivariate measures of association should be taken into account. However, these measures of association provide information about the strength of the relationships between predictor and dependent variables independent of the dependent variable interrelationships.

Using the GLM procedure and based on both the within and between Sum of Squares and Cross Products, the eigenvalues (roots) and eigenvectors are determined. Based on the

eigenvalues, roots statistics can be computed. These statistics test the effects of all the factors that affect the multivariate response.

It is worth to recall that the four root statistics have exact or approximate F statistics and according to the experimental settings, some are preferred than others. In the contest of a multidimensional set up, Wilks' lamda, Pillai's Trace, and Hotelling-Lawley's Trace are preferred whereas Roy's Greatest Root is preferred in cases where the comparison is one dimension. Nevertheless, all 4 statistics are used in this analysis.

Frequently, when interactions appear to exist, the interpretation for this interaction is carried out before the test for the main effects, otherwise we have to concentrate on the main effects. The root statistics indicate that *Algo*Object* has an impact on the three dimension response (SCPx, SCPy, SCPz) (see Table 12)

Table 12: Root statistics for Algo*Object interaction effect

Root Statistic	Value	*p*-value
Wilks' Lambda	0.9998	<.0001
Pillai's Trace	0.0001	<.0001
Hotelling-Lawley Trace	0.0001	<.0001
Roy's Greatest Root	0.0001	<.0001

To test for the hypothesis of no effect for *Algo* Table 13, the same test as above is performed and the null hypothesis of no effect is rejected: therefore *Algo* has also an effect on the surface contact points. For the other variables, that is *Object*, *Colfound* and *Force,* their statistics present the same highly significant p-value (<0.0001). We therefore observe that these results obtained from MANOVA are consistent with what was observed in the univariate setting.

Table 13: Root statistics for Algo

Variable	Statistics	*p*-value
Algo	Wilks' Lambda	<.0001
	Pillai's Trace	<.0001
	Hotelling-Lawley Trace	<.0001
	Roy's Greatest Root	<.0001

In the univariate and bivariate cases the assumption of normality is easily verified or checked but when we are dealing with cases where we have more than 2 dimensions, normality diagnostic is not easy to examine. However techniques are encounter to investigate on the normality status if we are dealing with multivariate models. One of such is the gamma plot which is obtained by computing a chi-square plot of the ordered distances.
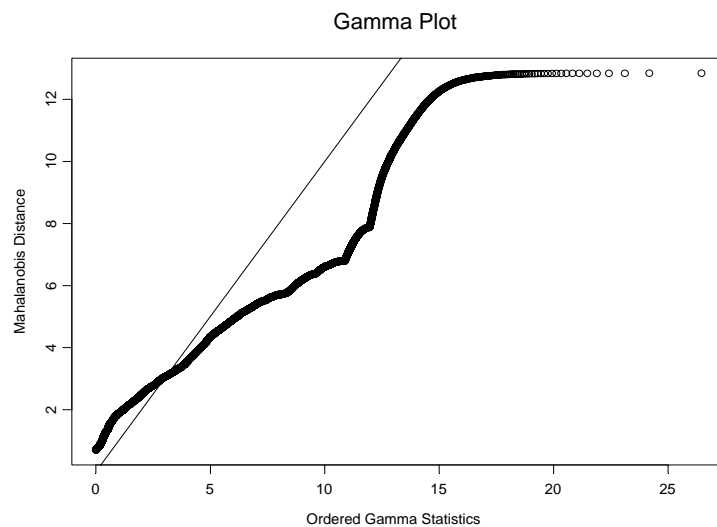
Gamma Plot

Figure 4: Chi-square plot of the ordered distances for SCPx, SCPy, SCPz

In the previous analysis, assumptions for the univariate normality were found not to hold. Multivariate normality on its part was asses by the gamma plot. From this plot (Figure 4), it appears that the points of this data set do not lie along the line with slope 1. Therefore we can say that this data do not appear to deviate from multivariate normality although the sample is large.

All the variables are significant. That is when considering the Surface Contact Points, Algorithms differ when taking object into account. We observe almost the same conclusion as the univariate analysis when the three correlated variables (SCPs) were combine to form the magnitude of SCP.

4.2.3.2 Considering SCPx, SCPx,SCPz and Time as Multivariate Responses

When we consider all of the responses, all of the above mention statistics are also significant at 5% level of significance [Appendix Table A.15]. To test for the effect of the

variables on the multivariate response (*SCPx, SCPx,SCPz and Time)*, the power of the test, Wilk's Lambda, Pillais' Trace, Hotelling-Lawley Trace is best as compared to Roy's Greatest Root who is preferred when we are on one dimension.

Table 14: Results for four-way MANOVA

Factors (overall effect)	Wilks' lambda statistics	
	F value	*p*-value
Algo*Object	126989	<0.0001
Object	427169	<0.0001
Algo	246212	<0.0001
Colfound	7997	<0.0001
Force	733	<0.0001

We present, the results for the Wilk's lambda (Table 14), since all the roots statistics yield the same result. These variables are all significant with (*p*-value<0.0001). Therefore they have an effect on the four correlated responses. Multivariate normality was investigated through Gamma plot (see Figure 5).
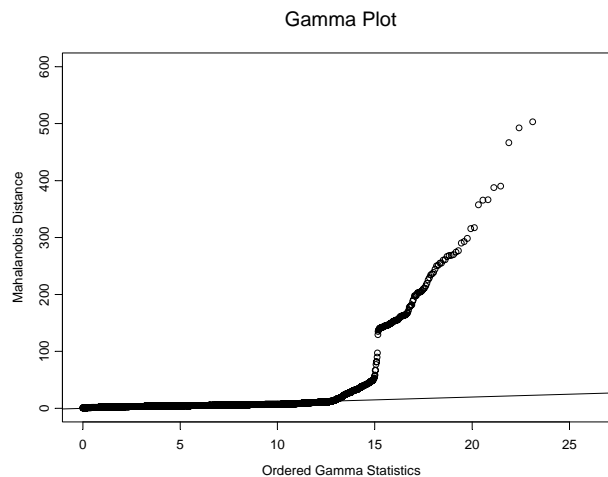
Figure 5: Chi-square plot of the ordered distances for SCPx, SCPy, SCPz and Time

The Chi-square or gamma plot to asses the multivariate normality reveals that the plot do not resembled a straight line that passes through the origin with slope 1. Suggesting a deviation from multivariate normality distribution.

The force feedback influences the performance (correctness and rendering time together) of the algorithms. There are differences in the performances of the algorithms used in the experimental (haptic algorithm) set-up and this is controlled or determined by the virtual object to touch. Another factor that has an effect on the performance is the detection step, which calculates whether the pointer is located inside the object or not (*Colfound*).

4.2.3.3 General Estimating Equation (GEE)

Both univariate and multivariate models employed were based on the normality assumptions. However, tests for normality indicated that the error terms (residuals) were non-normal. Also, it should be noted that for very large datasets, small deviations from normality may lead the tests to indicate non-normality. Therefore results obtained from the previous analysis could be considered valid. Nevertheless, the subsequent model assumes that the errors follow a gamma distribution.

However, to adopt a secured approach, it is interesting to consider model that encompasses a wide variety of distribution and compare its results to those obtained from the previous models. This was done with the help of the generalized estimating equation (GEE) through the GENMOD procedure. The model was implemented to the data with the gamma distribution and using as link the inverse power (-1). The results of this model [Appendix Table A.16] yield results which were in the same direction as the previous results obtain from the ANOVA and MANOVA models.

All of the variables are significant as in the previous multivariate model. We observe from the results that when considering a variable there exist a significant difference among the levels suggesting that the variable is significant therefore has an effect on the correlated responses. Therefore results from the previous models may be considered valid although we realized departure from normality.

From the estimates [Appendix Table A.16], it can be seen that force has a significant positive parameter. Thus, the SCP values as well as the rendering time increases with the

force applied. This is expected, as more force will lead to more 'volume' of the object being in contact with the pointer and more time will be needed for the pointer to cover more of the object. Also, Colfound was specified as a class variable in the model with a negative estimate. This implies that larger SCP values and rendering time are observed when *Colfound* is 1 (i.e. collision between pointer and object) compared to when *Colfound* is zero. This conclusion is similar to those of the previous models.

5. DISCUSSION AND CONCLUSION

In the last few years, haptic algorithms have advanced considerably. However, no standard evaluation method exists, which allows researchers to compare their results in an objective manner. Nevertheless based on certain aspects, research has been carried on in the field of comparisons between algorithms. These comparisons are based on the performance and correctness of the algorithms. The data at hand come from a haptic experience where two devices where used: one was used as input to record and the other was used as output.

The objective of the study was from the experimental set up to investigate which statistics can be applied for the comparison. Based on the objectives of the research question, statistical models were used to analyse the data. To stick on the primary interest (performance) and the secondary interest (correctness) of the research question, a simple univariate was first performed considering first the rendering time as the continuous response and secondly considering also the magnitude of the surface contact point for the correctness of the algorithm as response. This ANOVA applied to the data revealed that the variables, Force, Colfound, the interaction Object and Algo have effects on the univariate responses.

To take into account the correlation pattern of the 3 SCP (SCPx, SCPy, SCPz), a multivariate analysis was considered. The analysis revealed that all of the parameters were significant at 5% level of significance confirming the findings in the ANOVA models. Also a multivariate model was conducted taking into account the 3 dimensional SCPs and time variables. The results still in this case were consistent with the previous ones.

Analysis of variance and multivariate analysis of variance are analyses that require that assumptions like normality should be fulfil: checking these assumptions in both the univariate setting and multivariate setting did show deviation from normality therefore to account for that, the generalized estimating equation (GEE) was considered. This

estimating procedure uses the inverse power (-1) as link function and the gamma distribution which covers the more restrictive normal distribution. Here specifying the covariance structure was not of interest because GEE allows us to mispecify the covariance structure as long as the mean structure is specified [12]. Though the results didn't give the type 3 , they were however on the same direction as in the ANOVA and MANOVA models. So, results from the previous models could be consider valid and generalized.

As conclusion, we observe in one word that the variables in interest were all significant that is: the performance and the correctness of algorithms is determined by the type of algorithms. Algorithms coming from the same company have the same performance and also have approximately the same level of correctness. Based on the performance we can say that the algorithm Log-CHAI-AABB performs better while Log-HAL performs worst. Considering the SCP, Log-HAL, Log-CHAI, Log-CHAI-AABB have higher magnitude of SCP and the same level of correctness while: Log-HAL-Spattlash, Log-HAL-AABB and Log-HAL have smaller magnitude of SCP.

As recommendation we will say that it would have been nice to have differently skilled subjects (such as novice, experienced and expert users) furthermore, they could split the large size of the data to many individuals to have as much as possible individuals so to give more general conclusion on the process.

A secondary objective of this project was to develop a small function in R/SPLUS, which can be used to compare the performance of 2 algorithms. A function was developed in R, which compares the performance of 2 algorithms based on their SCP values and or their execution time. The comparison can be based on generalized linear models or bivariate Gaussian model. A brief definition of the function, its arguments and outputs, as well as exemplary sample codes is presented in the Appendix.

6.   REFERENCE

[1] **Agresti, A**. (2002), Categorical Data Analysis, Second Edition, New York: John Wiley & Sons Inc

[2] **Kirkpatrick, A.E., Douglas, S.A**. Application-based evaluation of haptic inter-Faces : Proceedings of the 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, Orlando, FL, USA (2002) 32-39.

[3] **Sutcliffe, A., Gault, B.**: Heuristic evaluation of virtual reality applications. Inter-acting with Computers 16 (2004) 631-849.

[4] **Oakley, I., McGee, M.R., Brewster, S., Gray, P**. Putting the feel in 'look and feel': Proceedings of CHI 2000, The Hague, NL (2000) 415-422.

[5] **Raymaekers, C., Beets, K., Van Reeth, F**.: Fast haptic rendering of complex objects using subdivision surfaces: Proceedings of the sixth PHANToM Users Group Workshop, Aspen, CO, USA (2001).

[6] **Acosta, E., Temkin, B**. Scene complexity: A measure for real-time stable haptic applications: Proceedings of the sixth PHANToM Users Group Workshop, Aspen, CO, USA (2001).

[7] **Anderson, T., Brown, N**. The activepolygon polygonal algorithm for haptic force generation: Proceedings of the sixth PHANToM Users GroupWorkshop, Aspen, CO, USA (2001).

[8] **De Boeck, J., Raymaekers, C., Coninx,** K.: Assessing the increase in haptic load.

[9] **De Boeck, J. Raymaekers, C., Coninx, K**.: A Method for the Verification of Haptic Algorithms. Hasselt University, Expertise Centre for Digital Media (EDM) and transnationale Universiteit Limburg

[10] **Basdogan, C., Ho C., Slater, M. & Srinivasan, M.A**. (1998) The role of haptic communication in shared virtual environments: Proceedings of the PHANToM Users Group Workshop, (http://www.sensable.com/community/PUG98/19_basdogan.pdf).

[11] **Durlach, M. & Slater, M**. (1998) Presence in shared virtual environments and virtual togetherness: Proceedings of the BT Presence Workshop BT Labs (http://www.cs.ucl.ac.uk/staff/m.slater/BTWorkshop/TouchExp/index.html.

[12] **Verbeke, G. & Molenberghs, G**. (2005). Models for Discrete Longitudinal Data. Springer Series in statistics. New-York: Springer-verlag

[13] **Molenberghs, G.** (2007), Multivariate Data Analysis, for Biostatistics Uhasselt press.

[14] **Verbeke, G. & Molenberghs, G.** (2000) Linear Mixed Model for Longitudinal Data Springer Series in Statistics. New York: Springer-Verlag.

[15] **Richard, A. & Dean, W, Wichern.** (2002) Applied Multivariate Statistical Analysis (5th Edition). Pearson Education International.

[16] **Neter, J., Kutner, MH, Nachtsheim, CJ, and Wasserman, W. (2005).** Applied Linear Statistical Models (5th edition). McGraw-Irwin series.

[17] **Laird, N. M. and Ware, J. H.** (1982) Random effects models for longitudinal data. Biometrics, **38**, 963-974.

7. APPENDIX

Table A 1: Intereaction between Algo and Object

objects	Algo	Inter
1	1	1
1	2	2
1	3	3
1	4	4
1	5	5
1	6	6
2	1	7
2	2	8
2	3	9
2	4	10
2	5	11
2	6	12
3	1	13
3	2	14
3	3	15
3	4	16
3	5	17
3	6	18

Table A 2: Correlation of SCPx,SCPy,SCPz and time Residuals

	Resid1	Resid2	Resid3	Time
Resid1	1.00000	-0.12080	0.06878	0.06878
Resid2	-0.12080	1.00000	0.18038	0.00214
Resid3	0.06878	0.18038	1.00000	-0.00727
Time	-0.00081	0.00214	-0.00727	1.00000

Table A 3: Constancy of Variance and Normality tests for the model with time as response

Test	Stat	Pvalue
Algo	5	0
Object	2	0
Company	1	0
Colfound	1	3.012848E-45
Force	1	0
Algo*Object	10	0
Kolmogorov-Smirnov	0.34	0.01
Cramer-von Mises	40216.76	0.005
Anderson-Darling	200095.38	0.005

Table A 4: Constancy of Variance and Normality tests for magnitude of (SCP) model

Test	Stat	Pvalue
Algo	5	0
Object	3	0
Company	1	0
Colfound	1	2.0224848E-27
Force	1	0
Algo*Object	10	0
Kolmogorov-Smirnov	0.26	0.005
Cramer-von Mises	22420.07	0.001
Anderson-Darling	113889.69	0.0001

Company

Colfound
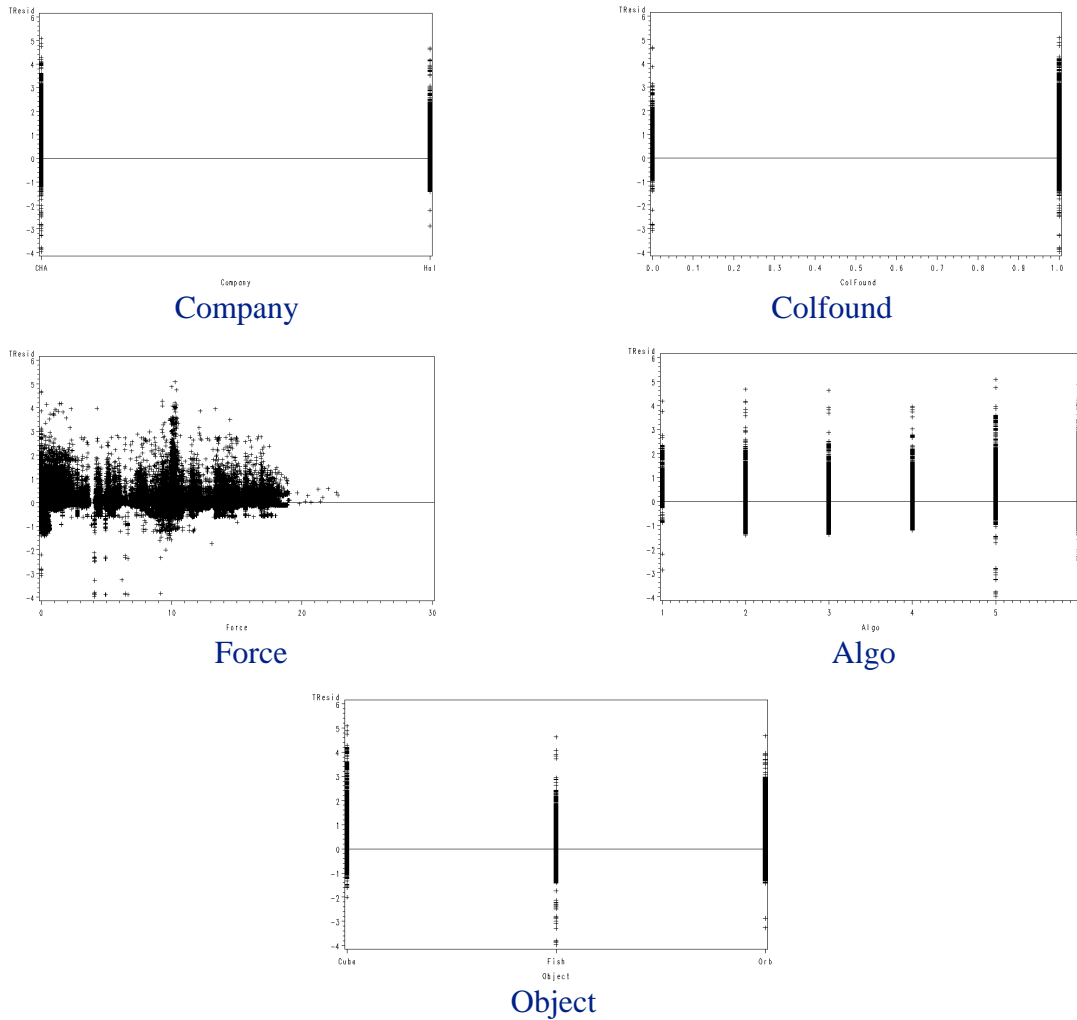
Force

Algo

Object

Figure A 1: Plots of residuals vs. Predicted for transformed Time
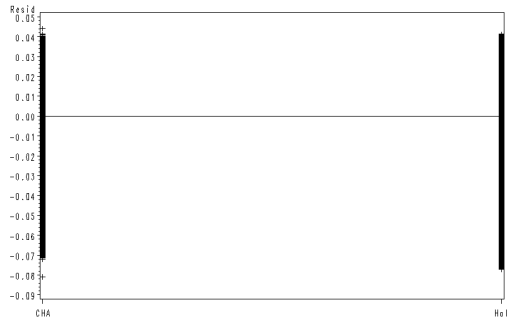
Company
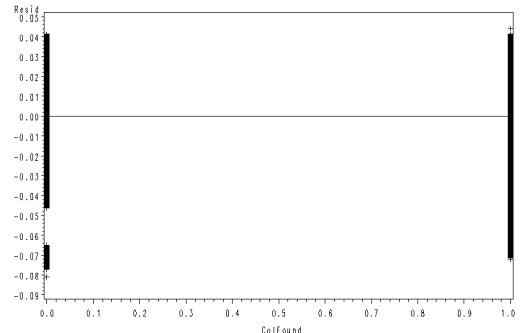
Colfound
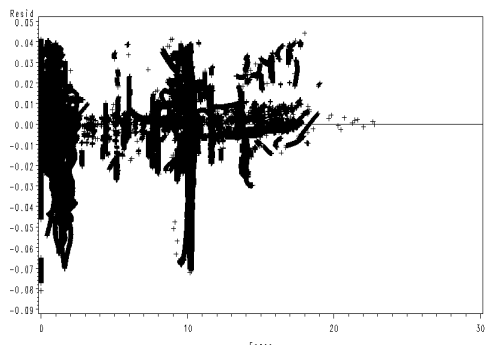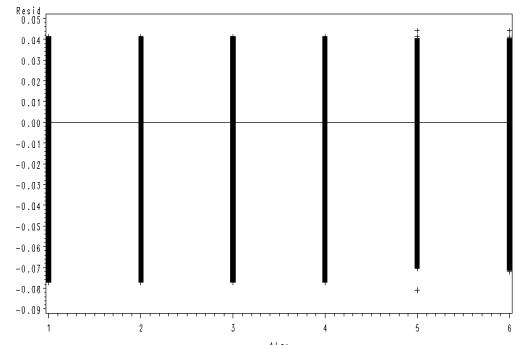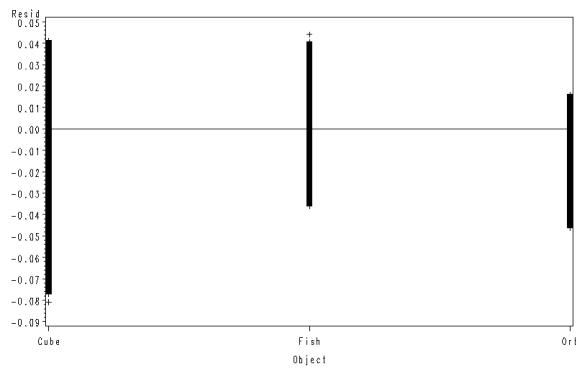
Force

Algo

Object

Figure A 2: Plots of residuals vs .Predicted for magnitude of SCP

Company
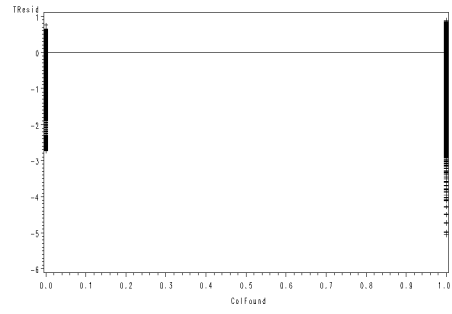
Colfound
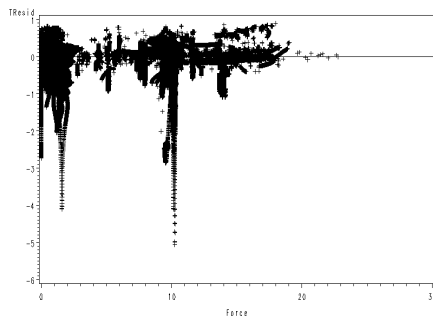
Force
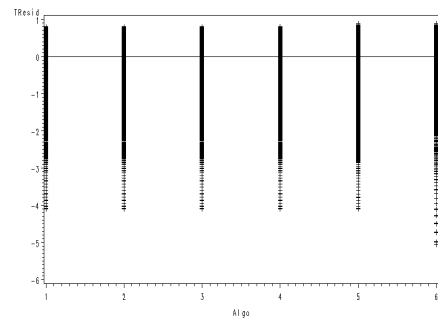
Algo
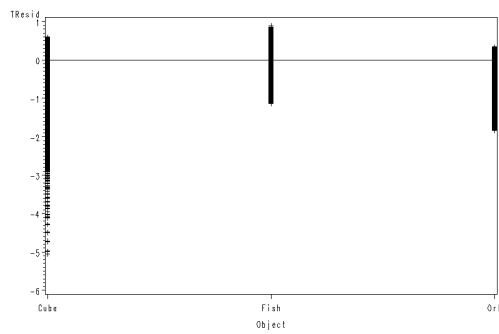
Object

Figure A 3: Plots of residuals vs . Predicted for transformed SCP
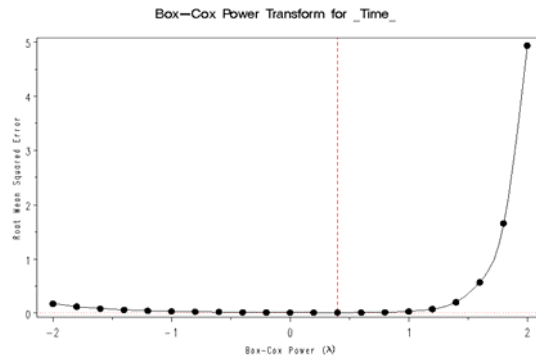
Figure A 2: Family of lambda for the model with time as response

Table A 5: Family of lambda for the model with SCP as response

Box power_LAMBDA_	_SSE_	_RMSE_	_EDF_	_LIKE_	CONF
-2	0.000797699	0.000029772	899980	9379758.13	*
-1.8	0.000797725	0.000029772	899980	9379743.72	
-1.6	0.00079775	0.000029773	899980	9379729.32	
-1.4	0.000797776	0.000029773	899980	9379714.91	
-1.2	0.000797801	0.000029774	899980	9379700.5	
-1	0.000797827	0.000029774	899980	9379686.1	
-0.8	0.000797852	0.000029775	899980	9379671.69	
-0.6	0.000797878	0.000029775	899980	9379657.28	
-0.4	0.000797903	0.000029775	899980	9379642.88	
-0.2	0.000797929	0.000029776	899980	9379628.47	
0	0.000797955	0.000029776	899980	9379614.06	
0.2	0.00079798	0.000029777	899980	9379599.65	
0.4	0.000798006	0.000029777	899980	9379585.24	
0.6	0.000798031	0.000029778	899980	9379570.83	
0.8	0.000798057	0.000029778	899980	9379556.43	
1	0.000798082	0.000029779	899980	9379542.02	
1.2	0.000798108	0.000029779	899980	9379527.61	
1.4	0.000798133	0.00002978	899980	9379513.2	
1.6	0.000798159	0.00002978	899980	9379498.79	
1.8	0.000798184	0.000029781	899980	9379484.38	
2	0.00079821	0.000029781	899980	9379469.97	

Table A 6: Newly created data combining Algorithm and Object variables

objects	Algo	Inter
1	1	1
1	2	2
1	3	3
1	4	4
1	5	5
1	6	6
2	1	7
2	2	8
2	3	9
2	4	10
2	5	11
2	6	12
3	1	13
3	2	14
3	3	15
3	4	16
3	5	17
3	6	18

Table A 7: Mean Time by Algorithm

Algo	M_Time
6	0.01563
2	0.01998
4	0.02018
3	0.02022
5	0.10887
1	0.37534

Table A 8: Mean Magnitude of SCP by Algorithm

Algo	M_SCP
4	0.051887
5	0.051991
6	0.052042
3	0.052347
2	0.052347
1	0.052347

Table A 9: Mean Time and Mean magnitude of SCP by virtual object

Object	M_SCP	M_Time
Cube	0.073129	0.01459
Fish	0.039898	0.05371
Orb	0.043455	0.2118

Table A 10: Multiple comparison between Object when time is the response

Object Comparison	Difference Between Means	Simultaneous 95% Confidence Limits		
Orb - Fish	0.15809175	0.15789989	0.15828361	***
Orb - Cube	0.19721294	0.19702108	0.19740480	***
Fish - Orb	-.15809175	-.15828361	-.15789989	***
Fish - Cube	0.03912119	0.03892933	0.03931305	***
Cube - Orb	-.19721294	-.19740480	-.19702108	***
Cube - Fish	-.03912119	-.03931305	-.03892933	***

*** *Comparison significant at the 5% level*

Table A 11: Multiple comparison between Object when magnitude of SCP is the response

Object Comparison	Difference Between Means	Simultaneous 95% Confidence Limits		
Cube - Orb	0.02967420	0.02956461	0.02978380	***
Cube - Fish	0.03323118	0.03312159	0.03334078	***
Orb - Cube	-.02967420	-.02978380	-.02956461	***
Orb - Fish	0.00355698	0.00344738	0.00366657	***
Fish - Cube	-.03323118	-.03334078	-.03312159	***
Fish - Orb	-.00355698	-.00366657	-.00344738	***

*** *Comparison significant at the 5% level*

Table A 12: Multiple comparison between Algorithms when time is the response

Algo Comparison	Difference Between Means	Simultaneous 95% Confidence Limits		
1 - 5	0.2664689	0.2661000	0.2668377	***
1 - 3	0.3551134	0.3547446	0.3554822	***
1 - 4	0.3551558	0.3547870	0.3555247	***
1 - 2	0.3553625	0.3549937	0.3557314	***
1 - 6	0.3597088	0.3593400	0.3600776	***
5 - 1	-0.2664689	-0.2668377	-0.2661000	***
5 - 3	0.0886445	0.0882757	0.0890134	***
5 - 4	0.0886870	0.0883181	0.0890558	***
5 - 2	0.0888937	0.0885249	0.0892625	***
5 - 6	0.0932399	0.0928711	0.0936088	***
3 - 1	-0.3551134	-0.3554822	-0.3547446	***
3 - 5	-0.0886445	-0.0890134	-0.0882757	***
3 - 4	0.0000424	-0.0003264	0.0004113	
3 - 2	0.0002491	-0.0001197	0.0006180	
3 - 6	0.0045954	0.0042266	0.0049642	***
4 - 1	-0.3551558	-0.3555247	-0.3547870	***
4 - 5	-0.0886870	-0.0890558	-0.0883181	***
4 - 3	-0.0000424	-0.0004113	0.0003264	
4 - 2	0.0002067	-0.0001621	0.0005755	
4 - 6	0.0045530	0.0041841	0.0049218	***
2 - 1	-0.3553625	-0.3557314	-0.3549937	***
2 - 5	-0.0888937	-0.0892625	-0.0885249	***
2 - 3	-0.0002491	-0.0006180	0.0001197	
2 - 4	-0.0002067	-0.0005755	0.0001621	
2 - 6	0.0043463	0.0039774	0.0047151	***
6 - 1	-0.3597088	-0.3600776	-0.3593400	***
6 - 5	-0.0932399	-0.0936088	-0.0928711	***
6 - 3	-0.0045954	-0.0049642	-0.0042266	***
6 - 4	-0.0045530	-0.0049218	-0.0041841	***
6 - 2	-0.0043463	-0.0047151	-0.0039774	***

*** Comparison significant at the 5% level

Table A 13: Multiple comparison between Algorithms when magnitude of SCP is the response

Algo Comparison	Difference Between Means	Simultaneous 95% Confidence Limits		
1 - 2	0.00000000	-.00021068	0.00021068	
1 - 3	0.00000000	-.00021068	0.00021068	
1 - 6	0.00030556	0.00009488	0.00051624	***
1 - 5	0.00035665	0.00014597	0.00056733	***
1 - 4	0.00046071	0.00025003	0.00067139	***
2 - 1	0.00000000	-.00021068	0.00021068	
2 - 3	0.00000000	-.00021068	0.00021068	
2 - 6	0.00030556	0.00009488	0.00051624	***
2 - 5	0.00035665	0.00014597	0.00056733	***
2 - 4	0.00046071	0.00025003	0.00067139	***
3 - 1	0.00000000	-.00021068	0.00021068	
3 - 2	0.00000000	-.00021068	0.00021068	
3 - 6	0.00030556	0.00009488	0.00051624	***
3 - 5	0.00035665	0.00014597	0.00056733	***
3 - 4	0.00046071	0.00025003	0.00067139	***
6 - 1	-.00030556	-.00051624	-.00009488	***
6 - 2	-.00030556	-.00051624	-.00009488	***
6 - 3	-.00030556	-.00051624	-.00009488	***
6 - 5	0.00005109	-.00015959	0.00026177	
6 - 4	0.00015515	-.00005553	0.00036583	
5 - 1	-.00035665	-.00056733	-.00014597	***
5 - 2	-.00035665	-.00056733	-.00014597	***
5 - 3	-.00035665	-.00056733	-.00014597	***
5 - 6	-.00005109	-.00026177	0.00015959	
5 - 4	0.00010406	-.00010662	0.00031474	
4 - 1	-.00046071	-.00067139	-.00025003	***
4 - 2	-.00046071	-.00067139	-.00025003	***
4 - 3	-.00046071	-.00067139	-.00025003	***
4 - 6	-.00015515	-.00036583	0.00005553	
4 - 5	-.00010406	-.00031474	0.00010662	

Table A 14: Mean time per object and algorithm

Object	Algo	M_Time
Cube	5	0.00705
Cube	6	0.00832
Cube	2	0.01575
Orb	4	0.01622
Orb	6	0.01651
Cube	3	0.01659
Cube	1	0.01981
Cube	4	0.02003
Fish	2	0.02035
Fish	3	0.0204
Fish	6	0.02206
Orb	3	0.02368
Orb	2	0.02382
Fish	4	0.02429
Fish	5	0.07539
Fish	1	0.15979
Orb	5	0.24417
Orb	1	0.94642

Table A 15: Root statistics when considering SCPx, SCPy, SCPz and time

Variable	Statistics	p-value
Algo	Wilks' Lambda	<.0001
	Pillai's Trace	<.0001
	Hotelling-Lawley Trace	<.0001
	Roy's Greatest Root	<.0001
Object	Wilks' Lambda	<.0001
	Pillai's Trace	<.0001
	Hotelling-Lawley Trace	<.0001
	Roy's Greatest Root	<.0001
Colfound	Wilks' Lambda	<.0001
	Pillai's Trace	<.0001
	Hotelling-Lawley Trace	<.0001
	Roy's Greatest Root	<.0001
Force	Wilks' Lambda	<.0001
	Pillai's Trace	<.0001
	Hotelling-Lawley Trace	<.0001
	Roy's Greatest Root	<.0001
Algo*Force	Wilks' Lambda	<.0001
	Pillai's Trace	<.0001
	Hotelling-Lawley Trace	<.0001
	Roy's Greatest Root	<.0001

Table A 16: Parameter estimate of the GEE model

Parameter			DF	Estimate	Standard Error	Wald 95% Confidence Limits		Chi-Square	p-value
Intercept			1	52.5651	0.1474	52.2761	52.8541	127106	<.0001
ALGO	1		1	-50.1503	0.1449	-50.4343	-49.8663	119805	<.0001
ALGO	2		1	-7.0023	0.1893	-7.3734	-6.6312	1367.66	<.0001
ALGO	3		1	-6.8852	0.1895	-7.2567	-6.5137	1319.50	<.0001
ALGO	4		1	1.7837	0.2060	1.3798	2.1875	74.94	<.0001
ALGO	5		1	-44.5990	0.1448	-44.8827	-44.3153	94920.0	<.0001
ALGO	6		0	0.0000	0.0000	0.0000	0.0000	.	.
OBJECT	Cube		1	-18.5636	0.1688	-18.8944	-18.2328	12096.5	<.0001
OBJECT	Fish		1	-7.6667	0.1803	-8.0202	-7.3132	1807.11	<.0001
OBJECT	Orb		0	0.0000	0.0000	0.0000	0.0000	.	.
ALGO*OBJECT	1	Cube	1	46.2485	0.1859	45.8842	46.6127	61919.2	<.0001
ALGO*OBJECT	1	Fish	1	20.0945	0.1839	19.7340	20.4550	11935.8	<.0001
ALGO*OBJECT	1	Orb	0	0.0000	0.0000	0.0000	0.0000	.	.
ALGO*OBJECT	2	Cube	1	4.5802	0.2236	4.1419	5.0185	419.47	<.0001
ALGO*OBJECT	2	Orb	0	0.0000	0.0000	0.0000	0.0000	.	.
ALGO*OBJECT	3	Cube	1	4.1471	0.2235	3.7090	4.5852	344.27	<.0001
ALGO*OBJECT	3	Fish	1	8.7663	0.2449	8.2862	9.2463	1281.23	<.0001
ALGO*OBJECT	3	Orb	0	0.0000	0.0000	0.0000	0.0000	.	.
ALGO*OBJECT	4	Cube	1	-5.7640	0.2364	-6.2274	-5.3006	594.27	<.0001
ALGO*OBJECT	4	Orb	0	0.0000	0.0000	0.0000	0.0000	.	.
ALGO*OBJECT	5	Cube	1	50.1539	0.1962	49.7695	50.5384	65371.3	<.0001
ALGO*OBJECT	5	Fish	1	24.0179	0.1917	23.6422	24.3935	15703.2	<.0001
ALGO*OBJECT	5	Orb	0	0.0000	0.0000	0.0000	0.0000	.	.
ALGO*OBJECT	6	Cube	0	0.0000	0.0000	0.0000	0.0000	.	.
ALGO*OBJECT	6	Fish	0	0.0000	0.0000	0.0000	0.0000	.	.
ALGO*OBJECT	6	Orb	0	0.0000	0.0000	0.0000	0.0000	.	.
COLFOUND			1	-0.0195	0.0291	-0.0766	0.0376	0.45	0.5024
FORCE			1	0.1982	0.0054	0.1877	0.2088	1352.03	<.0001
Scale			1	1.1547	0.0010	1.1528	1.1567		

==
To Compare algorithms

Description: A function for comparing the performance of two algorithms, based on their SCP values and their execution time. It can be done using Generalized Linear Models (GLM) or Bivariate normal models.

CompAlgo(Algo1,Algo2,univariate=T,Family=gaussian,Link="identity",type="deviance",plot.em=FALSE)

Arguments:

Algo1: A dataframe with the SCP (SCP_X, SCP_Y, SCP_Z) values and execution time from the first algorithm

Algo2: A dataframe with the SCP values (in the same order as in Algo1) and execution time from the second algorithm.

univariate: A logical value, if TRUE univariate generalized linear models are used otherwise a bivariate normal model is used. Default is TRUE

Family: Name of the error distribution used in the model. Default is 'gaussian'.

Link: An appropriate link for the error distribution used in the model. Default is "identity".

type: specifies the type of residuals obtained from the univariate GLM.

plot.em: A logical value indicating if the normal quantile plots should be plotted.

Resid: A logical value indicating if the residuals should be presented in the output.

The residuals are not presented by default, i.e. default is FALSE

The options 'Family', 'Link', 'type', 'plot.em' are only available for the GLM models.

Value

For univariate models, the mean difference in the SCP norm, its standard error are presented. Similarly to SCP, the mean difference in execution time, its corresponding standard error and pvalue are presented.

For bivariate models, the mean difference, corresponding standard errors and pvalues of SCP and execution time between both algorithms are presented, together with the 2 by 2
 variance covariance matrix (Sigma) of SCP and execution time.

For both **univariate** and **bivariate** options, the residuals are also presented if Resid is TRUE.