Calibrating Surgical SmartForceps^{TM} Using Bootstrap and Multilevel Modeling Techniques

by

Parisa Azimaee

A Thesis submitted to the Faculty of Graduate Studies of
The University of Manitoba
in partial fulfilment of the requirements of the degree of

Master of Science

Department of Statistics
University of Manitoba
Winnipeg

Abstract

Knowledge of forces, exerted on brain tissues during the performance of neurosurgical tasks, is critical for quality assurance, case rehearsal, and training purposes. Quantifying the interaction forces has been made possible by developing SmartForceps^{TM}, a bipolar forceps retrofitted by a set of strain gauges. The unknown values of implemented forces are estimated using voltages read from strain gauges. To this end, one needs to quantify the force-voltage relationship to estimate the interaction forces during microsurgery. This problem has been addressed in the literature by following the physical and deterministic properties of the force-sensing strain gauges without obtaining the precision associated with each estimate. In this thesis, we employed different probabilistic methodologies such as bootstrapping, weighted least squares regression, Bayesian regression and multi-level modeling in order to estimate the implemented force on tissue using voltages read from strain gauges. We obtain both point and interval estimates of the applied forces at the tool tips and calculate the precision associated with each point estimate. As a proof-of-concept, the proposed techniques were then employed to estimate unknown forces, and construct necessary confidence intervals using observed voltages in data sets that are measured from conducting surgical tasks on a cadaveric brain. Results indicated that the proposed techniques are capable of estimating tool-tissue interaction forces with acceptable level of accuracy.

Acknowledgment

I wish to express my sincere gratitude to my supervisor, Dr. Mohammad Jafari Jozani for his precious advises, patience and guidance throughout this research. I have learned a great deal under your supervision and I am forever grateful for all your support.

I would also like to thank my co-advisor, Dr. Yaser Maddahi, who helped me understand and acquire the necessary technical knowledge about the interesting field of neurosurgery with SmartForceps. Thank you for sharing your valuable experiences and engineering skills with me. Furthermore, I am thankful to the project NeuroArm at the University of Calgary for providing the medical data for this research.

I am thankful to the members of my thesis committee, Drs. Lisa Lix, and Saman Muthukumarana, for taking the time and reading my thesis. The valuable comments you provided improved this work immensely. I gratefully acknowledge the funding I received towards my Master's degree from the Department of Statistics, the Faculty of Science and the Faculty of Graduate Studies at the University of Manitoba, as well as NSERC.

Finally, I must express my very profound gratitude to my parents for always being there when I needed them and providing me with infinite support and courage to pursue my dreams. I am also indebted to my uncle for his encouragement and guidance throughout this journey. This accomplishment would not have been made possible without you by my side.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This chapter provides background information on the importance of tool-tissue force interaction measurement in neurosurgery. The chapter elaborates on the underlying motivation of this research, that is, to help neurosurgical trainees and residents have knowledge of the amount of forces required to complete a surgical task by providing information on implemented force on tissues during the performance of microsurgery. A complete list of research objectives and questions is also presented in this chapter. Finally, the research plan section describes how each research objective could be addressed.

1.1 Background and Motivation of the Problem

Knowledge of the interaction forces in neurosurgery is essential in training process of novice neurosurgeons and neurosurgical trainees. According to statistics, significant amount of errors in neurosurgeries (more than 50%), are due to applying excessive force to the brain tissue, that may result in tissue injury (Maddahi et al., 2016). This clearly demonstrates the necessity of avoiding the application of excessive forces.

Knowing the amount of interaction forces, would not only reduce damages caused by excessive applied forces, but would also help neurosurgeons learn about safe margins of forces when dealing with brain tissue. The safe level of applied force is considered as a value between the maximum and minimum effective forces, and may be very useful in the training of novice surgeons and trainees. Determination of safe limit of forces help residents in neurosurgery acquire surgical skills by practicing surgery on cadaveric brains, and even in clinical trials. Knowledge of force values may also improves the learning curve significantly, since the residents can *"learn-by-doing"*, instead of only *"observing"* experienced surgeons performing surgical tasks.

On the other hand, novice neurosurgeons require years of experience as well as multiple pre-clinical and clinical trials to become educated in dealing appropriately with the brain tissues. Nevertheless, as the hours of surgeries are significantly decreasing, the opportunities for gaining the appropriate experience for a novice surgeon are decreasing (Reznick and MacRae, 2006). As a result, there will be an increasing demand for improving the efficiency of the learning process and providing trainees with quantitative tools to assess their surgical skills (Gan et al., 2015).

Nowadays, simulation-based training of neurosurgeons is becoming an integrated part of the neurosurgical training modules (Clark et al., 2017). Simulations provide surgeons the opportunity to rehearse the neurosurgical case so that they can learn, practice, and acquire the experience in a significantly shorter period of time (Zareinia et al., 2015). However, to do this, it is necessary to find a reasonable relationship between organ tissue and applied forces to the surgical tool. Simulators are mostly based on simulation techniques, such as Finite Element method and Boundary

Element method, that are not able to provide very realistic information compared to results obtained using an experimental study. There are several literature addressing the design of simulators for training neurosurgeons (Rosseau et al., 2013; Marcus et al., 2016). Nevertheless, the effectiveness of simulation-based training highly depends on the safe margins measured during an actual surgery. For example, if the maximum (safe) force exerted to the brain tissue performing task **A** is 0.3 N (where N is Newton), then the maximum allowable force during the simulation of task **A** should always be less than 0.3 N.

Although, in microsurgery[1] and especially for training purposes, it is often necessary for neurosurgeons to quantitatively measure the amount of force applied to the brain tissue. However, there are not many appropriate tools to properly and precisely quantify the technical aspects of surgical skills. Therefore, this training process has remained mainly qualitative.

In order to provide force feedback to the surgeon, conventional surgical tools have to be modified. This modification occasionally involves the addition of force sensors to conventional surgical tools. As an example, Tanimoto et al. (1998) proposed a micro force sensor for a catheter, that is, a surgical tool for endovascular surgeries. The force sensor measures interaction forces between catheter and blood vessels. Authors claim that the sensor has reasonably high sensitivity and it is able to measure even small interaction forces.

Several other studies were conducted to design surgical tools that are able to efficiently measure the interaction forces in surgeries. For example, a team of researchers at the University of Calgary, Department of Clinical Neuroscience,

[1]The term "microsurgery" refers to a class of surgical operations performed by an operating medical-grade microscope (Yaşargil, 2013).

performed series of research on force sensing, real-time force measurement and robot assisted surgeries (neuroArm, 1997). This group of researchers, under the project *neuroArm*, addressed the problem of interaction force measurement and introduced the idea of a *SmartForceps*™ (Zareinia et al., 2015). The *SmartForceps*™ is a bipolar forceps equipped with several strain gauges that enables neurosurgeons to keep track of the amount of applied forces. In *SmartForceps*™, the applied forces to the brain tissue generates a change in the electrical resistance of the strain gauge, and as a result, the voltage would change.

The goal in this thesis is to develop and validate several statistical tools in order to find proper models that are capable of predicting the applied forces during neurosurgery, using the observed voltages in the strain gauges of *SmartForceps*™ developed by scientists in project neuroArm at the University of Calgary (Zareinia et al., 2015). This prediction capability not only enables neurosurgeons to apply safe amount of forces during neurosurgery, but also accelerates the process of training novice neurosurgeons.

1.2 Research Objectives

The main objectives of this research are as follow:

- Preliminary study to obtain high-level understanding of the practice of neurosurgery, including terminologies, definitions, surgical tasks, and surgical tools.
- Identifying and developing statistical methods in order to find a reasonable statistical model between forces applied to the brain tissue and voltages read

from strain gauges.

- Investigating advantages and disadvantages of some important and commonly-used applicable statistical methods.

- Developing necessary techniques to perform calibration 2-dimensional (2D) and 3-dimensional (3D) *SmartForceps*$^{\text{TM}}$ using techniques such as multilevel modeling, Bayesian analysis, and the bootstrap methodology.

- Validating the application of the proposed models in a real-world scenario.

- Comparing the performance of suggested statistical methods via simulation studies as well as real data applications.

1.3 Research Question

In order to meet the above research objectives, among others, the following questions should be addressed:

- What is the application of predicted forces in the real field?

- What are the statistical models that are more suitable for force prediction in neurosurgery?

- How asymmetric loss functions can be used to provide more reliable force predictions in the Bayesian context?

- Is it feasible to use a combination of statistical methods to further improve the estimation accuracy?

- What are the differences between all available tools for 1D, 2D, and 3D forces? And, what are the advantages/disadvantages of each method for each tool (*i.e.*, 1D, 2D, 3D)?
- Is there any obligation to filter the calibration data set?

1.4 Research Plan

1.4.1 Preliminary Studies on Neurosurgery

Understanding the underlying concept of this research requires some preliminary knowledge about the performance of neurosurgery. This knowledge can be classified into several aspects listed bellow:

- Gaining knowledge on instrument handling and becoming familiar with different components of the forceps.
- Understanding important forces in a neurosurgical operation, such as dissection (opening) and coagulation (closing) forces.
- Obtaining information about the range of effective forces in neurosurgery to avoid injury or incomplete task.
- Studying the structure and applications of force-sensing strain gauges, as well as information about the position of the configuration/position of installed strain gauges.

1.4.2 Identifying and Developing Statistical Methods

The main goal of this research is to first identify and compare appropriate statistical tools in order to predict applied forces during neurosurgery and possibly, develop new techniques or use a hybrid of them in order to address challenges associated with our real data problem. In the first step, it is essential to find proper statistical models and quantify the relationship between applied forces and output voltages obtained from strain gauges. The underlying problem of interest can be classified as an inverse regression problem, that is called calibration. In calibration problems, the response variable is observed and the explanatory variable needs to be predicted. The main challenge is to use and develop statistical approaches that are easy to understand by practitioners and are capable of constructing efficient point and interval estimates of implemented force by surgeons using the observed voltages in stain gauges mounted on the prongs of the medical forceps.

1.4.3 Important Features of a Proposed Statistical Model

Several factors are involved when selecting the statistical methodology for our underlying calibration problem. Some of the key factors when selecting a proper statistical method include:

- **Time Complexity:** This factor considers the time required to run a method and obtain force estimations. For instance, some methodologies, such as bootstrapping, require long run simulations, whereas other methods (*e.g.* , Eisenhart's method) do not need any simulation and are therefore more time efficient.

- **Computational Complexity:** This is a function of the number of steps and complexity of the estimation required by the proposed method in order to obtain the results.
- **Accuracy:** The accuracy, in the context of statistical estimation is defined in terms of the bias, and the prediction's Mean Squared Errors (MSE). Bias and the MSE are inversely proportional to the accuracy.
- **Available Options to Modify the Results:** Some methodologies are able to provide point estimation only (*e.g.*, the naive approach), whereas other methods are capable of providing interval estimation (*e.g.*, bootstrapping).

The most important factor for this research is accuracy, since even small values of excessive forces in this problem could be resulted in serious injuries. After this factor, we should take time complexity into account to provide real-time results. The last but not least significant factors are available options to modify the results and computational complexity.

1.4.4 Applying Proposed Methods to the Calibration Data Set

After a thorough investigation into available statistical methods and determining which methodologies have the potential of further improvements, a few are chosen based on their advantages and disadvantages. Selected methods should be applied to the calibration data set. By the calibrations data set, we mean the training data set obtained in the lab, that includes both applied forces to the forceps tips and also read voltages from the strain gauges. This process helps us identify a reasonable

relationship between voltages and forces, and enables us to predict the amount of force in the future neurosurgical operations.

1.4.5 Application in Real Practice

Estimation of applied forces to the brain tissue during the performance of neurosurgery, not only improves the quality and efficiency of novice neurosurgeons and surgical residents training, but also prevents damages caused by excessive forces to the brain tissue (Zareinia et al., 2015). However, to ensure that the proposed statistical methods are capable of estimating true amount of applied forces, we will implement the proposed methods through conducting surgical tasks on a cadaveric brain.

1.4.6 Comparing All Suggested Statistical Methods

A comparison of all statistical tools, employed in the thesis, is required to identify the most applicable method. This comparison is mainly based on the trade-off between the items explained in Section 1.4.3. A summary of this comparison will be provided in Chapter 5

1.5 Organization of the Thesis

The organization of the thesis is as follow. Chapter 1 provides detailed description of the problem and explains the significance of force estimation in neurosurgery. Identified research objectives and questions as well as a plan to scientifically address each question are also described in Chapter 1.

Terminologies, definitions, and concepts commonly encountered in the field of neurosurgery (*e.g.*, surgical equipment, surgical tasks, force measurement devices) are briefly described in Chapter 2. This provides the necessary basis for the rest of the thesis. Furthermore, a review of statistical calibration methods is also presented in Chapter 2.

Theory, application, and results of the bootstrap technique and Multi-level modeling, are presented in Chapters 3 and 4, respectively. Some concluding remarks as well as a few future research plans are presented in Chapter 5.

Chapter 2

Literature Review

In this chapter, background information on force sensing, in the context of neurosurgery is presented. We explain the process of obtaining force data using *Smart-Forceps*™ and provide some preliminary information on the statistical tools that are often used in calibration problems.

2.1 Force Sensing in Neurosurgery

Measuring the amount of interaction forces implemented with a surgical tool by surgeons provides a unique opportunity in order to better handle surgical tools and more appropriately interact with tissues. To this end, several new methodologies are developed to employ force sensors and measure the interaction forces during the surgical operations. One solution is to perform operation using a computerized and sensorized machine such as surgical robotic manipulators. Early traces of research in the area of robot assisted surgery and *intelligent* surgical tools, date back to the late 1990's and early 2000. Taylor et al. (1995) and Berkelman et al. (2003) are examples of such attempts. A more recent example is Üneri et al. (2010), in which a

microsurgery robot with micro-force sensors was used for an eye surgery application. However, due to obvious risks and higher safety margins associated with neurological surgery (also known as neurosurgery), there has been less attempt and willingness to perform experiments with technology-assisted tools for these types of surgeries. Nevertheless, there are multiple ongoing research on this topic. Examples of such studies can be found in neuroArm (1997), Beretta et al. (2016) and Yin et al. (2016).

Recently, Gan et al. (2015) conducted a pilot study to develop the *SmartForceps*™, by adding force sensors to regular surgical forceps. Authors evaluated the functionality of their developed *SmartForceps*™ by performing three different neurosurgeries on cadaveric brains. The pilot study provided surgeons with real-time dissection and coagulation force data. The study also claimed that more than 70% of the interaction forces are between 0 N and 0.3 N. It is also mentioned that applied forces are dependent on a variety of factors such as tissue type, region of the brain and the surgical task itself.

Researchers of the *NeuroArm* group focused on methodologies to enhance safety in neurosurgery (neuroArm, 1997). In particular, they conducted several research on robot-assisted surgeries and the use of *SmartForceps*™. For instance, in Maddahi et al. (2016), the *SmartForceps*™ was employed to perform fifty different neurosurgical tasks. The goal was to measure the peak force values during the performance of a set of coagulation and dissection tasks. This work also presented the limitations of different sensory systems such as altering the shape of the forceps that is not of interest since it would change instrument handling as well as the amount of the force needed to be applied.

In another research by the *NeuroArm* group (Marcus et al., 2014), obtaining

quantitative data on microneurosurgery with tele-operated robotic system was explained. While this data was measured with a blade and Rhoton dissector that was held by a robotic arm, it was not very helpful in presenting true forces, because it was dependent on the arm feedback mechanism.

Furthermore, in Maddahi et al. (2015), four different robot-assisted neurosurgeries were conducted using a haptic hand-controller, and the amount of forces, exerted to the brain tissue, were measured. In Maddahi et al. (2015), *NeuroArm* surgical system, an image-guided computer-assisted manipulator, was employed to perform those surgical procedures. Maximum amounts of interaction forces were reported as 1.67 N, 1.65 N and 1.68 N along x, y, and z axes, respectively.

In Wang et al. (2015), development of commercially-available SmartForceps$^{\text{TM}}$ (Codman & Shurtleff Inc., MA, USA) that is capable of measuring the interaction forces and quantifying the displacements, was explained thoroughly. This development follows several steps that are summarized as follow:

- Three pairs of strain gauges was installed on the two prongs of the SmartForceps$^{\text{TM}}$, 3 on each prong, in order to keep track of forces along x, y and z axes. As shown in Figure 2.1, forces along x axis are measured through strain gauges S_1 and S_2. Strain gauges S_3 and S_4 are in charge of force measurement in y direction, and the last two strain gauges, S_5 and S_6, keep track of forces along z axis. Performance of all pairs of strain gauges was examined by implementing different surgeries on cadaveric brains. Zareinia et al. (2015), compared the results obtained through installing only one pair of strain gauges versus mounting two pairs, having the same experimental conditions. The comparison

showed that having a pair of strain gauge on each prong (having two pairs in total) would result in more accurate measurement. An specific type of an electric resistance wire strain gauge was used in this project (CEA-13-125UN-350, Micro-Measurements, Wendell, NC, USA), and the output obtained from all these strain gauges was voltage.

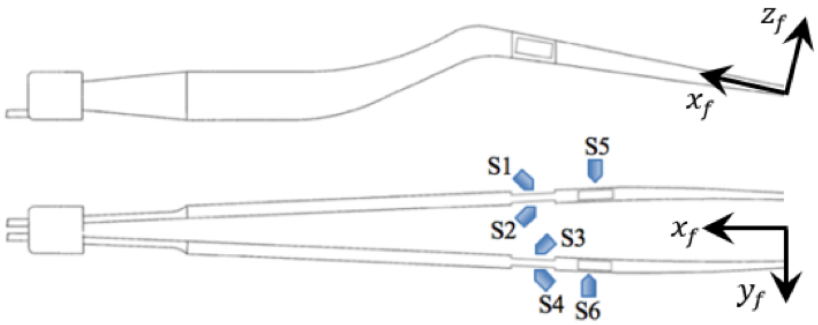
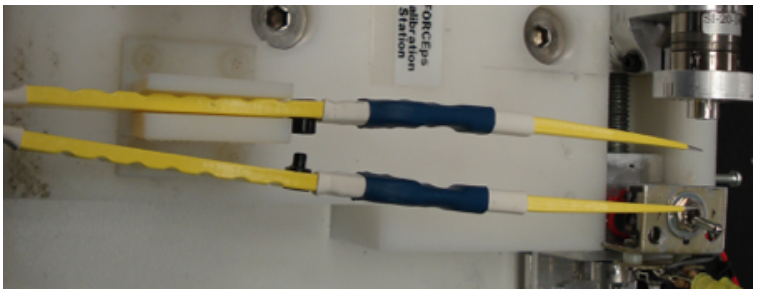
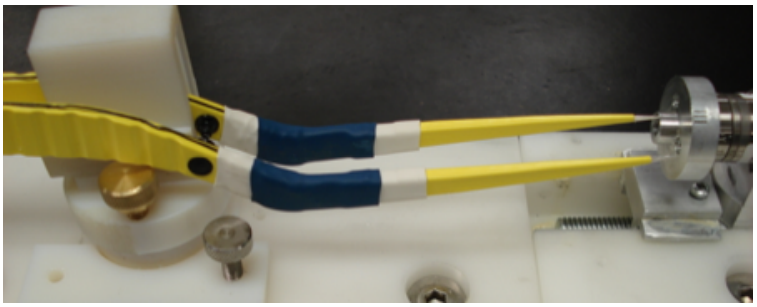


Figure 2.1: Graphical illustration of 6 strain gauges mounted on the prongs of SmartForceps^{TM}. Where, applied forces along x and y axes are measured through strain gauges S_1, S_2, S_3, S_4, and forces along z axis are measured by strain gauges S_5 and S_6. Source: Maddahi et al., "Quantifying workspace and forces of surgical dissection during robot-assisted neurosurgery." The International Journal of Medical Robotics and Computer Assisted Surgery 12, no. 3 (2016): 528-537.

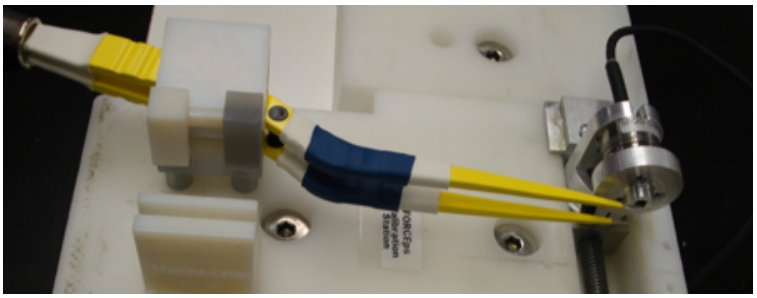
- The Fastrak software (Fastrack, 2017) was employed to record the data.
- Coordinate system considered for calibrating the SmartForceps^{TM} is shown in Figure 2.2 (x, y or z).
- A force sensor, Titanium Nano 17, was also employed to measure the amount of force in Newton. Force reading is required to find reasonable relationships between output voltages read from strain gauges and the observed forces obtained from the force sensor.



(a) Calibration along x_f axis



(b) Calibration along y_f axis



(c) Calibration along z_f axis

Figure 2.2: Coordinate system considered on the bipolar forceps tips during the process of calibration. Source: Wang et al.,"Development of an instrumented surgical setup for quantifying displacement and force in surgical dissection." In Proceedings of ASME international mechanical engineering congress and exposition. Houston, Texas, USA. 2015.

Variety of techniques as well as different sensory systems, as mentioned in Zareinia et al. (2015), were employed to keep track of the amount of exerted forces during surgeries. Two of those applicable techniques are strain gauge (load cells) and

pressure-based tactile sensors. According to Zareinia et al. (2015), among all types of sensory systems, strain gauges are the most popular. This is mainly because strain gauges are relatively cost efficient and provide reasonable robustness. Furthermore, there are normally minimum complication in the implementation and operation when using strain gauges.

Employing the SmartForceps^{TM} equipped with strain gauges and using the Nano 17 sensory system, the *NeuroArm* group was able to calibrate the SmartForceps^{TM}. A calibration data set provided by performing different surgical tasks on cadaveric brains. The calibration data is crucial to find proper statistical model between read voltages and exerted forces. The model should be capable of predicting the amount of interaction forces for future surgical operations. The governing equations and operational principle of strain gauges are briefly explained here. However, we should point out that comprehensive explanation of concepts such as mechanical stress, strain, tension, and their relationships is available in Shigley et al. (2004). Most of the equations in this section are presented without details and proofs. Interested reader may refer to available handbooks on Mechanical Engineering with focus on mechanics of material for further explanations (*e.g.* see Potma (1967) and Window et al. (1982)). For the purpose of our work, it is sufficient to mention that these equations are obtained using the mechanical characteristics of the material and structures of the sensors.

The design of the strain gauge is based on the Wheatstone bridge idea originally introduced by an English Physicist, Sir Charles Wheatstone (1802-1875) (Hoffmann, 1974). A simplified version of the Wheatstone bridge circuit is shown in Figure 2.3.

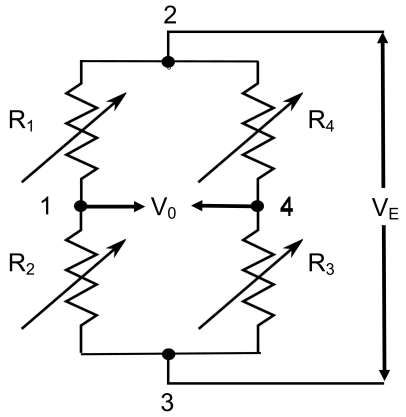


Figure 2.3: Wheatstone bridge circuit.

In Figure 2.3, R_1 through R_4 denote resistors in the bridge circuit. If the corner points (sometimes referred to as the nodes) 2 and 3 are connected to a known voltage source V_E (known as the excitation voltage), an output voltage V_0 appears between nodes 1 and 4. The magnitude of that voltage depends on the ratio of the resistors $R_1 : R_2$ and $R_3 : R_4$ (Dally et al., 1983). Using a similar concept, in a strain gauge, applied mechanical strain is transformed into a proportional change in the resistance. Let us briefly explain the role of strain gauge in the context of Wheatstone bridge circuit.

Consider the Wheatstone bridge circuit shown in Figure 2.4. In this configuration, one of the resistors in Figure 2.3 (R_3), is replaced with a strain gauge (Hereafter is called Rg). A particular class of strain gauges use a wire (uniform conductor) of electric resistivity R, length l, and cross-section A. The resistance of the wire is a function of the geometry and can be obtained by $R = \frac{\rho A}{l}$, where ρ is the resistivity of the material of the wire. Figure 2.4, presents the strain gauge that captures forces in x direction, configuration for the strain gauge that captures forces in y direction

is similar to this.

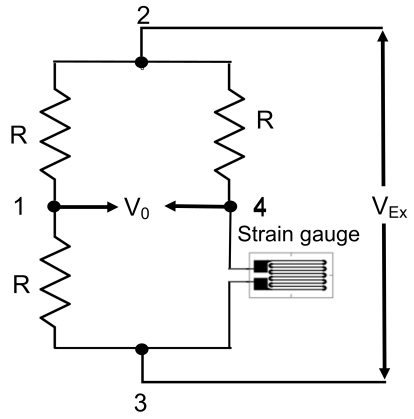


Figure 2.4: Simplified strain gauge in Wheatstone bridge configuration.

Furthermore, assume that the strain gauge is perfectly attached and bounded on the surface of an object. The object in this case is the smart forceps. In this scenario the strain of the strain gauge wire is proportional to the strain of the object. Therefore, alternation in strain (deflection in wire), changes the electrical resistance of the circuit. The rate of change of the resistance can be defined as (Zareinia et al., 2015)

$$\frac{dR}{R} = \frac{dl}{l} - \frac{dA}{A} + \frac{d\rho}{\rho}, \tag{2.1}$$

Note that other resistors in the circuit are chosen to be R (all with equal resistivity). In this configuration, if the nominal resistance of the strain gauge is R_g, the output voltage of the circuit (V_0) would be zero, only if $R = R_g$. The bridge in this situation is said to be in *balance*. However, if the resistance of the strain gauge changes due to strain, V_0 would not be zero. By measuring the change in the electrical resistance of the strain gauge (ΔR), average value of strain for the object can be obtained.

As shown in Hoffmann (1974) and Zareinia et al. (2015), for a constant strain gauge factor (S) and a constant excitation voltage (V_{Ex} and V_{Ey}) there exists linear relationships between axial and lateral strain (ϵ_x and ϵ_y) and the output voltage of the Wheatstone bridge (V_0) given by

$$\epsilon_x \approx \frac{4V_0}{SV_{Ex}}, \qquad \epsilon_y \approx \frac{4V_0}{SV_{Ey}}. \tag{2.2}$$

According to (2.2), the amount of mechanical strain is measured given that S, V_{Ex} and V_{Ey} are known. S is provided by strain gauge manufacturer and excitation voltages are obtained from a voltage regulator. In addition, the amount of stress is proportional to the amount of strain by the so-called modulus of elasticity, also known as the Young's modulus (E_{ym}), presented by

$$\begin{aligned} \sigma_x = E_{ym}\epsilon_x \approx \frac{4E_{ym}V_0}{SV_{Ex}}, \\ \sigma_y = E_{ym}\epsilon_y \approx \frac{4E_{ym}V_0}{SV_{Ey}}, \end{aligned} \tag{2.3}$$

where σ_x and σ_y are the amount of stress, and E_{ym} is the modulus of elasticity (Zareinia et al., 2015).

In Zareinia et al. (2015), it is explained that by considering the configuration similar to the one in Figure 2.1, when a tension loading is applied to the wire, strain in both the axial, and the lateral direction is observed. The so-called Poisson's ratio (ν) defined the correlation between the axial and the lateral strains of the wire. Zareinia et al. (2015) defined the relationship between input stress and output strain

in a forceps equipped with two pairs of strain gauges by

$$\begin{pmatrix}\sigma_x\\ \sigma_y\end{pmatrix} = \frac{E_{ym}}{(1+\nu)(1-2\nu)}\begin{pmatrix}1-\nu & \nu\\ \nu & 1-\nu\end{pmatrix}\begin{pmatrix}\epsilon_x\\ \epsilon_y\end{pmatrix}, \tag{2.4}$$

where σ_x and σ_y are the input stresses and ϵ_x and ϵ_y are output strains along x and y axes, respectively. Metals have a very small Poisson's ratio; thus the diagonal values in (2.4) are expected to be significantly greater than off-diagonal values (Zareinia et al., 2015).

Based on (2.4), having an applied force in x direction, $\sigma_x > 0$ and $\sigma_y = 0$, since the amount of ν is considerably small, the amounts of strains would be:

$$\epsilon_x = \frac{-\sigma_x}{E_{ym}}, \quad \epsilon_y = \frac{-\sigma_x \nu}{E_{ym}}. \tag{2.5}$$

Therefore, force in x direction would create strains in both x and y axes.

2.2 Statistical Calibration

This section describes the necessary statistical definitions and terms constantly used throughout this thesis. A brief overview of the methods used in the statistical calibration is also explained.

2.2.1 Necessary Definitions

In simple linear regression, values of the response variable y, also known as the *response* variable, are regressed on x values, or *explanatory* variables. The goal is to

predict the value of the response based on the observed x value. However, in the calibration, this trend is reverse. In other words, we are interested in predicting x values, where y values are observed.

In calibration studies, there are two main strategies: *controlled calibration*, and *random calibration*. In random calibration, researchers do not have any control over the data set, and they only observe and record data. However, in controlled calibration, researchers design experiments such that they are able to control which values of explanatory variables should be included in the experiment. This would enable them to, for example, control the explanatory variable values to be in an specific range. This study falls into the category of controlled calibration as in the process of collecting necessary training data sets, explanatory variables (*i.e.*, forces in each direction) are fixed, starting from a minimum value in a range and gradually increasing by a fixed amount to reach the maximum force. For each value of the applied force in different directions, voltages are recorded from different strain gauges that are implemented in the forceps. The test data consists of observed voltages in practice, and the goal is to estimate the implemented force by the surgeon associated with observed voltages.

Calibration has widely been used in areas such as univariate and multivariate linear regression (Osborne, 1991, Besalú, 2013), nonlinear models (Ni et al., 2014, Schwartz, 1977) as well as spline regression(Carey and Yee, 1992). As explained in Osborne (1991), statistical calibration, also known as inverse prediction, is highly related to instrument calibration. Nevertheless, statistical calibration is more complicated than instrument calibration.

There are two types of calibration: comparative versus absolute calibration. It is

necessary to distinguish between these two types since they are conceptually different. While in comparative calibration two instruments or methods are calibrated with respect to each other, absolute calibration mainly deals with one measurement approach that is not standard, and comes with minor errors. From this point forwards, the term calibration refers to the absolute calibration for simplicity.

Calibration approaches can be performed using univariate calibration and multivariate calibration. Considering each category, and as we explain later, there are different statistical methods that are applicable in univariate and multivariate calibration problems. Some of the well-known methods in univariate calibration include: classical regression (Besalú, 2013), inverse regression (Parker et al., 2010), Bayesian methods(Hawkins-Daarud et al., 2013), and non-parametric approaches (Rueda et al., 2010). The first three methods will be explained in Sections 2.2.2 and 2.2.4. The non-parametric approach was not part of this study and as such will not be explained here. Interested readers are referred to Lwin and Maritz (1980) for detailed explanation of this method.

In univariate calibration, we deal with one explanatory variable "x_i" and one response variable, "y_i", where x_i values are measured without any error, while y_i's are calculated having some errors, ϵ_i's. The linear relationship between the response and explanatory variables is defined as follow:

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i, \quad i = 1, 2, ..., n, \tag{2.6}$$

where $\epsilon_i \sim N(0, \sigma_1^2)$. Having (2.6), $\hat{\beta}_0$ and $\hat{\beta}_1$ are obtained using the training data set, that contains (x_i, y_i)'s pairs and following either the maximum likelihood or least squares approaches.. Then, observing the y values in the test data set (that

includes only y values) x values can be estimated as:

$$\hat{x}_j = \frac{y_j - \hat{\beta}_0}{\hat{\beta}_1}. \tag{2.7}$$

Here, one needs to assume the test data follows the linear model assumption used in the training step, that is $y_j = \hat{\beta}_0 + \hat{\beta}_1 x_j + \epsilon_j$ with $\epsilon_j \sim N(0, \sigma_2^2)$. This is the assumption that does not necessarily hold in real application.

2.2.2 Classical and Inverse Approaches in Calibration

In this section, two of the most important predictors that are mainly used in statistical calibration are presented, namely classical and inverse predictors. Statistical characteristics and comparison of these two methods are provided.

The following notations are frequently used throughout this section:

$$S_{xy} = \sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}),\ S_{xx} = \sum_{i=1}^{n}(x_i - \bar{x})^2,\ S_{yy} = \sum_{i=1}^{n}(y_i - \bar{y})^2, \tag{2.8}$$

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i. \tag{2.9}$$

The Classical Predictor

In Eisenhart (1939), the following estimator, which is obtained through regression of y on x, is denoted as the *classical estimator*. Mathematical formulation of the classical predictor is as follows. Let X and Y be two random variables, such that

$$E(Y \mid X = x) = \beta_0 + \beta_1 x. \tag{2.10}$$

Suppose we have the training data set as $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$. The least squares estimators of model (2.10) is obtained by minimizing the Sum of Squared Errors (SSE), that is $\sum_{i=1}^{n}(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2$, resulting in

$$\hat{\beta}_1 = \frac{S_{xy}}{S_{xx}} \quad \text{and} \quad \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}.$$

Suppose we observe $y_1, y_2, ..., y_m$ without knowing their associated x_i values. one can rewrite (2.10) to obtain the classical predictors of $x_i's$. To this end, working with

$$\hat{y} = \bar{y} + \frac{S_{xy}}{S_{xx}}(x - \bar{x}), \tag{2.11}$$

results in

$$\begin{aligned} \hat{x}_c &= \bar{x} + \frac{S_{xx}}{S_{xy}}(\bar{y}^{'} - \bar{y}) \\ &= \frac{(\bar{y}^{'} - \hat{\beta}_0)}{\hat{\beta}_1}. \end{aligned} \tag{2.12}$$

where $\bar{y}^{'}$ is the mean value of the observations $y_1, y_2, ..., y_m$, and S_{xy}, S_{xx}, $\bar{x}$ and $\bar{y}$ are defined in (2.8), (2.9), and it is assumed that $\hat{\beta}_1 \neq 0$. It is easy to show that, the classical predictor, $\hat{x}_c$, is also the maximum likelihood estimator of x if the errors follow a normal distribution as in (2.6) and (2.7).

The classical estimator has two major drawbacks (under the commonly used normality assumption). First, its MSE is infinite for fixed x_i's and finite n. Because when $\hat{\beta}_1$ in (2.12) is normally distributed, the probability of $\hat{\beta}_1 = 0$ is not zero (n is the number of calibration data points). Second, its mean value is undefined. These drawback are due to the fact that $\hat{\beta}_1$ is normally distributed independently of y_i's,

and the variance of $\frac{q}{\hat{\beta}_1}$ is infinite. Therefore, $\hat{x}_c$ has undefined mean and infinite variance and consequently infinite MSE (Williams, 1969).

The Inverse Predictor

Krutchkoff (1967) introduced another estimator for the calibration problem, that is called the *inverse predictor*. While in the classical predictor it is assumed that x_i's are fixed, in the inverse approach, it is assumed that x_i's are also random. Therefore, instead of regressing y_i's on x_i's (as was the case for the classical predictor), in the inverse prediction approach, x_i's are regressed on y_i's. The model in this case is given by:

$$E(X = x \mid y) = \gamma_0 + \gamma_1(y - \bar{y}), \tag{2.13}$$

and the least squares estimators of the model (2.13) is obtained by minimizing the SSE, with a minimum value given by $\sum_{i=1}^{n}(x_i - \hat{\gamma}_0 - \hat{\gamma}_1(y_i - \bar{y}))^2$, where,

$$\hat{\gamma}_1 = \frac{S_{xy}}{S_{yy}} \quad \text{and} \quad \hat{\gamma}_0 = \bar{x},$$

and as before, S_{xy}, S_{xx}, $\bar{x}$ and $\bar{y}$ are defined in (2.8) and (2.9).

Considering above formula, if we observe $y_1, y_2, ..., y_m$ without knowing their associated x_I values, the inverse predictor would be

$$\begin{aligned} \hat{x}_I &= \bar{x} + \frac{S_{xy}}{S_{yy}}(\bar{y}^{'} - \bar{y}) \\ &= \hat{\gamma}_0 + \hat{\gamma}_1 \bar{y}^{'}, \end{aligned} \tag{2.14}$$

where $\bar{y}'$ is the mean value of $y_1, y_2, ..., y_m$ at the estimation stage, and $\bar{y}$ is the mean value of $y_1, y_2, ..., y_n$ at the calibration stage.

Inverse predictor might initially appear to be more straightforward than the classical predictor, because it does not need inverting the regression model to provide an estimate for the explanatory variable. Nonetheless, the basic regression assumption that insists on independency of errors ($\epsilon_i = x_i - \hat{\gamma}_0 - \hat{\gamma}_1 y_i$) and y_i values, is not valid in inverse predictor. Also, in many experimental settings such as the one we consider in this thesis, the explanatory variables are fixed and do not satisfy the required assumption in the inverse approach.

2.2.3 Comparison of the Classical and Inverse Predictors

Krutchkoff (1967) conducted a Monte Carlo simulation study to support the idea that inverse predictor is better than classical predictor, since its MSE was uniformly less than the MSE of classical estimator. However, in Berkson (1969) it is shown that when S_{xx} is not very small (or x_i's are not very close to $\bar{x}$ in (2.8)) and $\frac{\sigma}{\beta_1}$ is small, the asymptotic MSE of the $\hat{x}_c$ is smaller than that of $\hat{x}_I$, where σ is the variance of the errors.

As mentioned in Section 2.24, the MSE and the mean value for the classical estimator ($\hat{x}_c$) are not finite, but by truncating $\hat{\beta}_1$ such that the probability of $\hat{\beta}_1 = 0$ is close to zero, one can make the MSE and the mean finite. Considering Tchebycheff's inequality given by

$$P(|\hat{\beta}_1 - \beta_1| \geq K) \leq \frac{\sigma^2}{K^2 \beta_1^2 S_{xx}}, \tag{2.15}$$

where $K > 0$, the probability of $\hat{\beta}_1$ being equal, or close to zero would be very small if S_{xx} is large and $|\frac{\sigma}{\beta_1}|$ is not large.

Lemma 2.2.1. *Suppose $Y_i \sim N(\beta_0 + \beta_1 x_i, \sigma^2)$, and we observe y_0. Assume that we are interested in estimating its corresponding X value denoted by x_0. The asymptotic Bias, MSE, and variance for the classical predictor $\hat{x}_c$, are obtained as follow:*

$$Bias(\hat{x}_{0c}) \approx \frac{\sigma^2}{\beta_1^2 S_{xx}}(x_0 - \bar{x}), \tag{2.16}$$

$$Var(\hat{x}_{0c}) \approx \frac{\sigma^2}{\beta_1^2}\big(1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{S_{xx}}\big), \tag{2.17}$$

$$MSE(\hat{x}_{0c}) \approx \frac{\sigma^2}{\beta_1^2}\big(1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{S_{xx}} + \frac{(x_0 - \bar{x})^2\sigma^2}{S_{xx}^2\beta_1^2}\big). \tag{2.18}$$

Similarly, for the inverse predictor $\hat{x}_{0I}$ we have

$$Bias(\hat{x}_{0I}) \approx \frac{-(x_0 - \bar{x})}{1 + \frac{\beta_1^2 S_{xx}}{(n-1)\sigma^2}}, \tag{2.19}$$

$$Var(\hat{x}_{0I}) \approx \frac{\sigma^2 S_{xx}}{\beta_1^2 S_{xx} + (n-1)\sigma^2}\big(1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{S_{xx}}\big), \tag{2.20}$$

$$MSE(\hat{x}_{0I}) \approx \frac{\sigma^2 S_{xx}}{\beta_1^2 S_{xx} + (n-1)\sigma^2}\big(1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{S_{xx}}\big) + \frac{(x_0 - \bar{x})^2}{\big(1 + \frac{\beta_1^2 S_{xx}}{(n-1)\sigma^2}\big)^2}. \tag{2.21}$$

Proof. In the regression model (2.10), estimates of coefficients are normally distributed as follow

$\hat{\beta}_1 \sim N(\beta_1, \frac{\sigma^2}{S_{xx}})$, and $\hat{\beta}_0 \sim N(\beta_0, \sigma^2(\frac{1}{n} + \frac{\bar{x}^2}{S_{xx}}))$, and $Cov(\hat{\beta}_0, \hat{\beta}_1) = \frac{-\bar{x}\sigma^2}{S_{xx}}$,

where σ^2 is the variance of errors, S_{xx} and $\bar{x}$ are defined in (2.8) and (2.9), respectively.

Since the classical and inverse predictors are the ratio of two dependent normal random variables, to calculate above formulas, we employ the δ-method to find below expressions for the expectation as well as the variance of the ratio of two arbitrary variables U and V with finite variances. (Casella and Berger, 2002, Parker et al., 2010)

$$E(\frac{U}{V}) \approx \frac{E(U)}{E(V)} + \frac{E(U)}{E^3(V)}Var(V) - \frac{Cov(U,V)}{E^2(V)}, \tag{2.22}$$

$$Var(\frac{U}{V}) \approx \frac{Var(U)}{E^2(V)} + \frac{E^2(U)}{E^4(V)}Var(V) - 2\frac{E(U)}{E^3(V)}Cov(U,V). \tag{2.23}$$

According to (2.12), the classical estimator for estimating $\hat{x}_0$ is

$$\hat{x}_{0c} = \frac{y_0 - \hat{\beta}_0}{\hat{\beta}_1}. \tag{2.24}$$

The Bias, variance, and MSE for the classical estimator is calculated as follow:

$$\begin{aligned} Bias(\hat{x}_{0c}) &= E(\hat{x}_{0c} - x_0) \\ &= E(\hat{x}_{0c}) - x_0 \\ &= E(\frac{y_0 - \hat{\beta}_0}{\hat{\beta}_1}) - x_0, \end{aligned} \tag{2.25}$$

where, $E(y_0 - \hat{\beta}_0) = \beta_1 x_0$ and $Var(y_0 - \hat{\beta}_0) = \sigma^2\big(1 + \frac{1}{n} + \frac{\bar{x}^2}{S_{xx}}\big)$.

Now, using (2.22)

$$Bias(\hat{x}_{0c}) \approx x_0 + \frac{\sigma^2}{\beta_1^2 S_{xx}}(x_0 - \bar{x}) - x_0$$

$$\approx \frac{\sigma^2}{\beta_1^2 S_{xx}}(x_0 - \bar{x}).$$

Also, following (2.23), we get

$$Var(\hat{x}_{0c}) = Var(\frac{y_0 - \hat{\beta}_0}{\hat{\beta}_1})$$

$$\approx \frac{\sigma^2}{\beta_1^2}\big(1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{S_{xx}}\big).$$

Finally, the MSE for the inverse predictor is

$$MSE(\hat{x}_{0c}) = Var(\hat{x}_{0c}) + Bias^2(\hat{x}_{0c})$$

$$\approx \frac{\sigma^2}{\beta_1^2}\big(1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{S_{xx}} + \frac{(x_0 - \bar{x})^2\sigma^2}{S_{xx}^2\beta_1^2}\big).$$

For the inverse estimator

$$\hat{x}_{0I} = \hat{\gamma}_0 + \hat{\gamma}_1(y_0 - \bar{y}), \tag{2.26}$$

the Bias, variance, and MSE are calculated as follow. For the Bias, we have

$$Bias(\hat{x}_{0I}) = E(\hat{\gamma}_0 + \hat{\gamma}_1(y_0 - \bar{y}) - x_0), \tag{2.27}$$

where, $E(\hat{\gamma}_0) = \bar{x}$, and since $\hat{\gamma}_1 = S_{xy}(S_{yy})^{-1}$ is independent from y_0, we can rewrite (2.27) as

$$Bias(\hat{x}_{0I}) = \bar{x} + \beta_1(x_0 - \bar{x})E(\hat{\gamma}_1) - x_0.$$

Considering that $\hat{\gamma}_1$ is the ratio of two random variables and using (2.22) and (2.23), given below expressions

$$E(S_{xy}) = \beta_1 S_{xx}, \quad E(S_{yy}) = (n-1)\sigma^2 + \beta_1^2 S_{xx},$$

$$Var(S_{yy}) = 2(n-1)\sigma^4 + 4\sigma^2\beta_1^2 S_{xx}, \;\; Cov(S_{xy}, S_{yy}) = 2\beta_1\sigma^2 S_{xx},$$

it is easy to show that

$$E(\hat{\gamma}_1) \approx \frac{\beta_1 S_{xx}}{(n-1)\sigma^2 + \beta_1^2 S_{xx}} + o(\frac{1}{n}) \approx \frac{1}{\frac{(n-1)\sigma^2}{\beta_1 S_{xx}} + \beta_1},$$

and

$$Var(\hat{x}_{0I}) \approx \frac{\sigma^2 S_{xx}}{\beta_1^2 S_{xx} + (n-1)\sigma^2}\big(1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{S_{xx}}\big).$$

Finally, we are able to calculate the Bias and MSE as

$$Bias(\hat{x}_{0I}) \approx \bar{x} + \frac{(x_0 - \bar{x})\beta_1}{\frac{(n-1)\sigma^2}{\beta_1 S_{xx}} + \beta_1} - x_0$$

$$\approx \frac{-(x_0 - \bar{x})}{1 + \frac{\beta_1^2 S_{xx}}{(n-1)\sigma^2}},$$

and

$$MSE(\hat{x}_{0I}) = Var(\hat{x}_{0I}) + Bias^2(\hat{x}_{0I})$$

$$\approx \frac{\sigma^2 S_{xx}}{\beta_1^2 S_{xx} + (n-1)\sigma^2}\big(1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{S_{xx}}\big) + \frac{(x_0 - \bar{x})^2}{\big(1 + \frac{\beta_1^2 S_{xx}}{(n-1)\sigma^2}\big)^2}.$$

□

2.2.4 The Bayesian Method in Prediction

The Bayesian method in calibration has two common approaches, namely the *Bayesian calibration* and the *Bayesian regression*. While the former employs Bayesian concept directly to estimate the explanatory variable, the latter provides estimation for the coefficients first, using the Bayesian method, and then uses the classical predictor (2.12), to predict the explanatory variable.

In this section, first the idea of Bayesian calibration is explained followed by a review of some of the commonly used Bayesian models for the calibration problem. Then, a brief description of the Bayesian regression approach is presented.

Bayesian Calibration

Bayesian calibration is one of the well known methods in calibration problems which has been extensively studied in the past (Osborne, 1991, Hoadley, 1970, Dunsmore, 1968). Suppose that $Data = \{x_i, y_i, \ \ i = 1, 2, ..., n\}$ is the calibration data set, and we observe y_0 at the prediction stage and we are interested in estimating its corresponding x_0. The Bayesian formula that is employed in the Bayesian calibration problem is defined by:

$$P(x_0 \mid Data, y_0) = \frac{f(Data, y_0 \mid x_0)P(x_0)}{\int_{S_x} f(Data, y_0 \mid x_0)P(x_0)} \propto f(Data, y_0 \mid x_0)P(x_0). \qquad (2.28)$$

where $P(x_0 \mid Data, y_0)$ is called the *posterior distribution*, $P(x_0)$ is known as the *prior density*, and $f(Data, y_0 \mid x_0) = L(x_0)$ is the *Likelihood function* of x_0 and is defined as the predictive function of observations.

According to (2.28), $Posterior \propto Prior \times Likelihood$. Because, the denominator is the *scale factor* and does not have any effect on making inference.

In classical estimator, $\hat{\beta}_1$ cannot be equal to zero, since it appears in the denominator of $\hat{x}_c$ (2.12). Therefore, one needs to *test* whether or not $\beta_1 = 0$. Testing this hypothesis is achieved through the F-Statistic defined by

$$F = \frac{\hat{\beta}_1^2 S_{xx}}{\hat{\sigma}^2}, \tag{2.29}$$

where,

$$\hat{\sigma}^2 = \frac{1}{(n-2)} \left\{ \sum_{i=1}^{n} (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2 \right\}. \tag{2.30}$$

According to (2.29), one can reject the null hypothesis ($H_0 : \beta_1 = 0$), if $F < F_{\alpha;1;(n+1-2)}$, where α is the significance level of the test. The rejection of the null hypothesis means that the classical predictor is not accurate enough. As Hoadley (1970) explains, it can be concluded that the calibration data set (x_i, y_i) maintains some information about the accuracy of the classical predictor, and we can give less weight to the classical estimator when we fail to reject the null hypothesis, and more weight when the null hypothesis is rejected.

Therefore, the Bayesian method would be beneficial in this case, since it summarizes the available information on x_0 through observing the calibration data set. In other words, the posterior distribution of x_0 would provide estimation through conditioning the prediction on the calibration data set.

In the Bayesian approach, the first step towards prediction is to find the appropriate prior density for the parameter. Hoadley (1970) proposed a theorem to find

the marginal posterior distribution of x_0.

Theorem 2.2.2 (Application of the Hoadley's Theorem for non-centered data)**.** *Suppose x_0 is independent of β_0, β_1, and σ^2, and consider a noninformative prior density for the parameters, that is*

$$P(\beta_0, \beta_1, \sigma^2) \propto \frac{1}{\sigma^2}. \tag{2.31}$$

Then, the marginal posterior distribution for x_0 would be

$$P(x_0 \mid Data, y_0) \propto L(x_0)P(x_0), \tag{2.32}$$

where,

$$L(x_0) = \frac{1}{((n-1)\hat{\sigma}^2\sigma_0^2)^{\frac{1}{2}}(1 + \frac{(y_0-\hat{\beta}_0-\hat{\beta}_1x_0)^2}{((n-1)\hat{\sigma}^2(\sigma_0^2)})^{\frac{n-1}{2}}}, \quad \sigma_0^2 = (1 + \frac{1}{n} + \frac{(x_0-\bar{x})^2}{S_{xx}}),$$

and $\hat{\sigma}^2$ is defined in (2.30)*.*

Proof. Following Hickey (2006) and under the assumed model, we first write

$$\ell(\beta_0, \beta_1, \sigma^2, x_0 \mid y_0, Data) \propto \ell(\beta_0, \beta_1, \sigma^2 \mid Data)\ell(\beta_0, \beta_1, \sigma^2, x_0 \mid y_0). \tag{2.33}$$

Now, the marginal posterior distribution of x_0 given y_0 and Data can be written as:

$$\begin{aligned} P(x_0 \mid y_0, Data) &\propto P(x_0 \mid Data)\, P(y_0 \mid x_0, Data) \\ &\propto P(x_0)\, P(y_0 \mid x_0, Data). \end{aligned} \tag{2.34}$$

Suppose $y_0 = \beta_0 + \beta_1 x_0 + \epsilon_0$, and,

$$\epsilon_0 \mid \sigma^2 \sim N(0, \sigma^2), \quad \hat{\beta}_0 \mid \sigma^2 \sim N(\beta_0, \sigma^2(\frac{1}{n} + \frac{\bar{x}^2}{S_{xx}})),$$

$$\hat{\beta}_1 \mid \sigma^2 \sim N(\beta_1, \frac{\sigma^2}{S_{xx}}), \quad Cov(\hat{\beta}_0, \hat{\beta}_1) = \frac{-\bar{x}\sigma^2}{S_{xx}}.$$

One can show that

$$(y_0 \mid \sigma^2, x_0, Data) \propto N\left(\hat{\beta}_0 + \hat{\beta}_1 x_0, \sigma^2(1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{S_{xx}})\right),$$

Now, under chosen noninformative prior for parameters and considering the commonly used inverse gamma distribution for σ^2,

$$\sigma^{-2} \mid Data \sim \chi^2(n-2, \hat{\sigma}^2),$$

where, $\hat{\sigma}^2$ was defined earlier in (2.30), we will get

$$\begin{aligned} P(x_0 \mid Data) &\propto P(x_0)P(y_0 \mid x_0, Data) \\ &= \frac{P(x_0)}{((n-1)\hat{\sigma}^2\sigma_0^2)^{\frac{1}{2}}(1 + \frac{(y_0 - \hat{\beta}_0 - \hat{\beta}_1 x_0)^2}{((n-1)\hat{\sigma}^2(\sigma_0^2)})^{\frac{n-1}{2}}}. \end{aligned}$$

□

Remark: This is easily obtained as, if $y_0 \sim N(\hat{\beta}_0 + \hat{\beta}_1 x_0\ , \sigma^2(1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{S_{xx}}))$, and $\sigma^{-2} \sim \chi^2(n-2, \hat{\sigma}^2)$, then $y_0 \mid x_0, Data \sim t_{n-2}(\hat{\beta}_0 + \hat{\beta}_1 x_0, \hat{\sigma}^2(1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{S_{xx}}))$.

Theorem 2.2.3 (Hoadley's Theorem)**.** *Suppose we have the exact same conditions as in Theorem 2.2.2, except x_i's are standardized. The posterior distribution of x_0 given y_0 and Data is defined by*

$$P(x_0 \mid Data, y_0) \propto L(x_0)P(x_0), \tag{2.35}$$

where,

$$L(x_0) = \frac{(1+n+x_0^2)^{\frac{(n-2)}{2}}}{(1+n+R\hat{x}_{0c}^2+(\frac{F'}{n-2}+1)(x_0-R\hat{x}_{0c}^2)^2)^{\frac{n-1}{2}}}, \quad R = \frac{F'}{F'+n-2}, \quad F' = \frac{n\hat{\beta}_1^2}{\hat{\sigma}^2}.$$

Note that the parameter F in (2.29)*, is changed to F' after standardizing x_i's.*

Proof. Proof is similar to the proof of Theorem 2.2.2, hence omitted. However, one can see Hoadley (1970) for a complete account for the derivation of the result. □

In Aitchison and Dunsmore (1980) it has been stated that if the prior distribution for x_0 be considered as $t_{n-3}(\bar{x}, (1+\frac{1}{n})\frac{S_{xx}}{n-3})$, then $\hat{x}_{0I}$ is equal to the mean of the posterior distribution.

In Osborne (1991), the inverse estimator (2.13), in the Bayesian terminology has been interpreted as a movement from the classical predictor (2.12), towards the mean value of prior distribution. Consequently, the more accurate the classical estimator, the less shift towards the inverse predictor.

Bayesian Regression

In regression analysis, *least squares* and *maximum likelihood* approaches are the most commonly used methods in estimating the regression coefficients. Another well-established methodology that considers prior on the parameters of the model, is based on the Bayesian regression approach that can also be used hopping that the extra assumptions in this approach result in more precision in estimating the coefficients.

In linear regression model, observations include response variable, y, and one or more explanatory variables, x. Suppose $y = X\beta + \epsilon$, where $\epsilon \sim N(0, \sigma^2)$, X is called the design matrix, and the parameters of the model are β and σ^2. The posterior distribution in this case is defined as follow:

$$P(\beta, \sigma^2 \mid y) \propto P(y \mid \beta, \sigma^2)P(\beta)P(\sigma^2), \tag{2.36}$$

where, $P(\beta)$ and $P(\sigma^2)$ are prior distributions, and $P(y \mid \beta, \sigma^2)$ is the likelihood function. As in (2.36), posterior distribution is proportional to product of the likelihood function and prior distribution. Therefore by obtaining the posterior distribution, we can make inference on the parameters of the model.

For instance, suppose σ^2 is known. Considering non-informative prior on β, as $\beta \sim N(0, \sigma_\beta^2)$, where σ_β^2 is known and considerably large such that the underlying prior distribution can be interpreted as a flat prior, the likelihood function is as follow:

$$P(y \mid \beta) \propto \exp(-\frac{a}{2}(y - X\beta)^\top (y - X\beta)), \tag{2.37}$$

where, $a = \frac{1}{\sigma^2}$ is called the precision. Furthermore, the posterior distribution is:

$$P(\beta \mid y) \propto \exp(-\frac{a}{2}(y - X\beta)^\top (y - X\beta) \times \frac{b}{2}(\beta^\top \beta)), \tag{2.38}$$

where, $b = \frac{1}{\sigma_\beta^2}$ is also called the precision. In order to find the posterior distribution we proceed as follow:

$$\begin{aligned} P(\beta \mid y) &\propto \exp\left(-\frac{1}{2}\left[a(y - X\beta)^\top (y - X\beta) + b(\beta^\top \beta)\right]\right) \\ &\propto \exp\left(-\frac{1}{2}\left[ay^\top y - 2a\beta X^\top y + \beta^\top a X^\top X\beta + b\beta^\top \beta)\right]\right). \end{aligned} \tag{2.39}$$

Since, $ay^\top y$ is constant with respect to β, we can rewrite (2.39) as

$$\begin{aligned} P(\beta \mid y) &\propto \exp\left(-\frac{1}{2}\left[\beta^\top(aX^\top X + bI)\beta - 2\beta^\top(aX^\top y) + constant\right]\right) \\ &\propto N(\mu, \Sigma), \end{aligned} \tag{2.40}$$

where, $\mu = a\Sigma^{-1}X^\top y$, $\Sigma = aX^\top X + bI$, and I is the identity matrix. Therefore, we obtained the posterior distribution.

More details on this method is provided in Chapter 4.

Chapter 3

Bootstrap Technique in Statistical Calibration

In this chapter we present the theory and application of the nonparametric bootstrap technique in the statistical calibration of an instrumented surgical forceps, *i.e.* SmartForceps^{TM}. Furthermore, an extension of the Eisenhart's method and it's application in our calibration problem is presented. Finally, we compare the performance of our proposed approach with the *Naïve* method, proposed in Zareinia et al. (2015) to predict the amount of interaction forces.

3.1 Why Bootstrapping?

As mentioned, there is a relationship between the external force applied to the brain tissue and the read voltages from the mounted strain gauges on the prongs of the surgical forceps. Therefore, we require to appropriately model the output voltages (*response variable*) and the force components (*explanatory variable*), to estimate the resultant force. A methodology based on the deterministic and physical

properties of the force-sensing strain gauges is employed in Zareinia et al. (2015). In the proposed method, estimates of the force is obtained using equations that relate recorded voltages from strain gauges to the forces exerted on the brain tissue (See Section 2.1).

While this technique, which is called the *Naïve* method hereafter, provides the first step towards estimating the exerted force on the brain tissue, it does not allow us to obtain the precision associated with each estimate, and hence construct necessary confidence intervals. In addition, it does not properly use the information of the training data set to fit the calibration model that is required for estimating force. The reason is that the proposed method in Zareinia et al. (2015) uses the information of the training data set to obtain the calibration model through a deterministic approach, which does not allow to study statistical properties of the estimates such as unbiasedness or construct confidence intervals for the unknown forces given the observed voltages. Also, in Zareinia et al. (2015), one needs to assume that the distribution of the voltages in calibration stage is similar to the distribution of the observed voltages in the real surgery for the estimation step. However, this is most likely not the case, as test data are obtained under real surgery situation and it is highly a function of the surgeon who is performing the surgery.

To address the above issues, in this chapter, we employ a probabilistic methodology by using a nonparametric bootstrap approach to obtain both point and interval estimates of the applied forces to the forceps during the performance of neurosurgery. Detailed explanation of the bootstrap technique is given in Efron and Tibshirani (1986). We also provide the precision associated with each estimate. To this end, we use a *multivariate calibration* technique for calibrating the voltage-force model.

This is done by first fitting a linear regression model between the voltage and force, followed by implementing a least squares method without assuming any parametric assumption, such as normality, for the distribution of the voltages obtained from four strain gauges mounted on the two prongs of the bipolar forceps. The bootstrap technique is then used to estimate the unknown forces and construct necessary confidence intervals using observed voltages in test data sets and following the inverse of the calibration model. We intentionally use the *calibration* terminology as our main goal is to predict the explanatory variable (*force*) by observing the response variable (*voltage*).

There exist several techniques to estimate the explanatory variable, other than the bootstrap method using the inverse of the calibration model. For more details see Chapter 2 as well as Jones and Rocke (1999). As we explained in Section 2.2.2, the most popular approach is probably the one based on the reverse regression by modeling the explanatory variable (x) on the response variable (y) using $x = \alpha y + \text{error}$ (Krutchkoff, 1967). In this approach, the goal is to estimate x based on the observed y values and find the coefficient α by reversing the role of the variables in the model. However, the structure of our data does not allow to use this approach, since x values in the training data are not random.

Another well-known estimation method is the *maximum likelihood* approach. This method uses the profile likelihood function for unknown x variable (Brown and Sundberg, 1987). This approach measures the mutual inconsistency, which would help to understand the differences between the likelihood-based and Bayes confidence regions with other unconditional sampling approaches. However, this approach requires some parametric and sometimes unverifiable assumptions about

the distribution of the errors of the calibration model. For example, one needs to assume that the underlying distribution of the errors for the test and training data sets are the same. An obvious problem is that the information available at the prediction step is usually limited, and the distribution of the observed voltages in the real surgery depends on the surgeon's experience and surgical skills.

In contrast, our proposed bootstrap approach is rather straightforward, and does not require such assumptions and the model adjusts itself by taking into account the effect of the surgeon using the bipolar forceps in the estimation process using a pooling approach during a required resampling step.

Another method that is able to provide point and interval estimation of the explanatory variable is known as the Eisenhart's method. Detailed explanation of this method is available in Eisenhart (1939). However, so far, the Eisenhart's method has been only used to address univariate problems. An extension to the original Eisenhart's method is proposed in Section 3.4.1. This extension allows us to employ this method for our multivariate problem. The outline of this chapter is as follow:

First, we address the statement of the problem, then the concept of the bootstrap technique with univariate as well as multivariate linear models is explained in Sections 3.2 and 3.3. Eisanhart's method in calibration is also explained in Section 3.4. Section 3.5 explains how the calibration data set is obtained and results of bootstrapping are reported in Section 3.6. Results of implementing the developed model in the real field are reported in Section 3.7 by obtaining the amount of interaction forces and corresponding intervals using data set quantified during the performance of neurosurgical tasks on a cadaveric brain. Finally, concluding remarks

are presented in Section 3.8

Throughout this chapter, we use two types of data sets: (i) *Standards* or *Training* data set, in which both response and explanatory variables are observed, that is $(F_1, V_1), (F_2, V_2), ..., (F_n, V_n)$ and (ii) *Unknowns*, which involve only response variables $(V_{01}, V_{02}, ..., V_{0r})$. In the simulation study and to evaluate the performance of our proposed method, we will also have *Test* data sets where we observe both the response and explanatory variables. These data sets are then treated as *Unknowns* by discarding their true values of the forces and predicting them using our proposed method to measure its accuracy in predicting the true forces.

3.2 Bootstrapping in Controlled Calibration With Univariate Linear Models

Suppose we observe a training data set of size n, $(F_1, V_1), (F_2, V_2), ..., (F_n, V_n)$, from the calibration station and assume that the relationship between the response variable (observed voltage, V) and the explanatory variable (applied force, F) is given by the following *calibration curve*:

$$V_i = \beta F_i + \epsilon_i, \tag{3.1}$$

with ϵ_i's being independent random errors having $E(\epsilon_i) = 0$, and $Var(\epsilon_i) = \sigma^2$. This model is easily justified through the model proposed in Zareinia et al. (2015), following the physical and deterministic properties associated with the surgical tool.

A common practice with model (3.1) is to predict a future value of the voltage from an observed value of the force. In the proposed application, however, the

interest lies in doing the reverse, that is, given $(V_{01}, V_{02}, ..., V_{0r})$, we want to estimate corresponding values of applied forces.

The bootstrap method could be a very helpful tool to estimate the force in our calibration setting. It not only helps to predict the force, but also enables us to obtain the precision of the estimates and construct confidence intervals. To implement the bootstrap technique using the measured data from the calibration station and under the linear model (3.1), we used both the *Training* and *Unknowns* data sets. The training data set was obtained under a control setting that covers the required range of the forces that could be observed in a practical situation after discretizing the force range with enough resolution. However, for the *Unknowns*, we only observed the voltages and the goal was to estimate their associated forces.

In order to estimate the amount of force F_0 associated with an observed voltage V_0, following a least squares method, the calibration curve $\hat{V} = \hat{\beta}F$ was first obtained from the *Training* data set, where $\hat{\beta} = \frac{\sum_{i=1}^{n} V_i F_i}{\sum_{i=1}^{n} F_i^2}$. Then, an estimate of the force, denoted by $\hat{F}_0$ was obtained as follow:

$$\hat{F}_0 = \frac{V_0}{\hat{\beta}}, \tag{3.2}$$

given that $\hat{\beta} \neq 0$. Note that if we observe several values of the voltage associated with a fix value of force, one could simply replace V_0 in (3.2) by $\bar{V}_0$, the average of the observed voltages.

One way to construct the bootstrap data set, that is required for the bootstrap calibration, is to obtain the residuals from both *Training* and *Unknowns*. In real application, this will help to adjust the estimation method and account for the effect

of the surgeon through combining the data sets from the calibration station and the real operation to form the bootstrap residual pool. To this end, for the *Training* data, we set,

$$\epsilon_i = V_i - \hat{\beta}F_i,\ i = 1, 2, ..., n, \tag{3.3}$$

and, for *Unknowns*,

$$\epsilon_j = V_{0j} - \bar{V}_0,\ j = 1, 2, ..., r. \tag{3.4}$$

Since the variation of the residuals around the mean is very small, there is a need to adjust residuals by the adjustment factor, $\sqrt{\frac{n}{n-p}}$, where n is the number of data points and p is the number of parameters (Jones and Rocke, 1999) .

The next step is to place the residuals in the residual pool. Therefore, the bootstrap data set for *Training* is given by:

$$V_i^* = \hat{\beta}F_i + R_i^*,\ i = 1, 2, ..., n \Longrightarrow (F_i, V_i^*), \tag{3.5}$$

and for *Unknown* is defined as:

$$V_{0j}^* = \bar{V}_0 + R_j^*,\ j = 1, 2, ..., r \Longrightarrow F_0^* = \frac{\bar{V}_0^*}{\hat{\beta}^*}, \tag{3.6}$$

where R_i^*, and R_j^* are random samples from the residual pool, and $\hat{\beta}^*$ is obtained by fitting a linear regression model to the bootstrap data set. Here, we consider a general case where we assume r values of the voltages are observed for each unknown force. In our setting, we simply take $r = 1$, and the bootstrap estimate of the corresponding force in one run of the procedure is given by $F_0^* = \frac{V_0^*}{\hat{\beta}^*}$. One needs to repeat this process B times, with large B, to obtain the bootstrap estimates of the

force associated with an observed voltage, denoted by $(F_{01}^*, F_{02}^*, ..., F_{0B}^*)$. The mean of these estimates is then used as an estimate of the force. One can also construct confidence intervals for the true but unknown value of the force F_0.

There exist several methods to obtain $100(1-\alpha)\%$ confidence intervals for $\hat{F}_0$, where α is usually taken $\alpha = 0.05$ resulting in a 95% confidence interval. Examples are:

i. *Percentile bootstrap*, that involves finding quantiles $q_{\frac{\alpha}{2}}$ and $q_{1-\frac{\alpha}{2}}$ of the bootstrap values $(F_{01}^*, F_{02}^*, ..., F_{0B}^*)$ and to constructing a confidence interval of the form $(q_{\frac{\alpha}{2}}, q_{1-\frac{\alpha}{2}})$.

ii. *bootstrap-t*, which is computationally expensive, and it is based on approximating the distribution of $\sqrt{n}(\hat{F}_0 - F_0)/se(\hat{F}_0)$ by $\sqrt{n}(F_0^* - \hat{F}_0)/\hat{se}(\hat{F}_0^*)$, and constructing the confidence intervals.

$$(\hat{F}_0 - c^*_{(n,\alpha/2)}\hat{se}(\hat{F}_0), \hat{F}_0 + c^*_{(n,1-\alpha/2)}\hat{se}(\hat{F}_0)), \tag{3.7}$$

where $c^*_{(n,\alpha/2)}$ and $c^*_{(n,1-\alpha/2)}$ are the $\alpha/2$-th and $(1-\alpha/2)$-th quantiles of the empirical distribution of $\sqrt{n}(F_0^* - \hat{F}_0)$, and

$$\hat{se}(\hat{F}_0) \simeq \frac{s}{\hat{\beta}}\sqrt{\frac{1}{r} + \frac{1}{n} + \frac{(\hat{F}_0 - \bar{F})^2}{SS_F}}. \tag{3.8}$$

Here s represents the estimate of the standard deviation of the errors, $\hat{\beta}$ is the slope of the regression line, SS_F is the force variance, and finally n and r denote the number of data points and replications, respectively.

3.3 Bootstrapping in Calibration with Multivariate Linear Models

There is a methodological difference between bootstrapping univariate and multivariate linear models for the force-voltage problem in our setting. In the univariate setting, which is only used for the illustration purpose, we assume that there is only one voltage associated with each force. However, in our real data set, and compared with the *univariate model*, there are two dimensional voltages associated with each force component. Suppose F_x and F_y denote the applied forces along the x and y directions, respectively. For each force we measure two sets of voltages. For example, if F_x is the amount of force in x direction, we observe a vector of voltages (V_{x_1}, V_{x_2}), where V_{x_1} is obtained from the strain gauge 1 and V_{x_2} from strain gauge 2, respectively. To implement the bootstrap technique for estimating the unknown values of F_{x_0} and F_{y_0}, we fit the following model to the *Training* data set:

$$\begin{pmatrix} V_{x_1} & V_{y_1} \\ V_{x_2} & V_{y_2} \end{pmatrix} = \begin{pmatrix} \alpha_1 & \beta_1 \\ \alpha_2 & \beta_2 \end{pmatrix} \begin{pmatrix} F_x & 0 \\ 0 & F_y \end{pmatrix} + \begin{pmatrix} \epsilon_{x_1} & \epsilon_{y_1} \\ \epsilon_{x_2} & \epsilon_{y_2} \end{pmatrix}, \tag{3.9}$$

where, $(V_{x_1}, V_{x_2})^\top$ is the observed voltages when the surgeon applies the force F_x along the x direction. Similarly, $(V_{y_1}, V_{y_2})^\top$ denotes the observed voltages when a force F_y is applied along the y direction. The error is represented by $\epsilon = \begin{pmatrix} \epsilon_{x_1} & \epsilon_{y_1} \\ \epsilon_{x_2} & \epsilon_{y_2} \end{pmatrix}$.

Therefore, the values of F_x, and F_y are estimated using the fitted models,

$$S_1 = \begin{pmatrix} \alpha_1 & \beta_1 \end{pmatrix} \begin{pmatrix} F_x & 0 \\ 0 & F_y \end{pmatrix} + \epsilon_1, \tag{3.10}$$

$$S_2 = \begin{pmatrix} \alpha_2 & \beta_2 \end{pmatrix} \begin{pmatrix} F_x & 0 \\ 0 & F_y \end{pmatrix} + \epsilon_2, \tag{3.11}$$

where $S_1 = \begin{pmatrix} V_{x_1} \\ V_{y_1} \end{pmatrix}$ and $S_2 = \begin{pmatrix} V_{x_2} \\ V_{y_2} \end{pmatrix}$. Here, S_i, refers to the voltages obtained from strain gauge $i, i = 1, 2,$. Also, $\epsilon_1 = \begin{pmatrix} \epsilon_{x_1} \\ \epsilon_{y_1} \end{pmatrix}$ and $\epsilon_2 = \begin{pmatrix} \epsilon_{x_2} \\ \epsilon_{y_2} \end{pmatrix}$. We use ϵ_i, $i = 1, 2,$ to show the error terms associated with the models that are used to fit linear relationships between the forces in x and y directions and voltages that are obtained from each strain gauge.

After fitting the necessary regression models, the following steps are used in order to obtain bootstrap estimates of the forces F_{x_0} and F_{y_0}.

1. Calculate $\hat{\alpha}_1, \hat{\beta}_1$, and $\hat{\alpha}_2, \hat{\beta}_2$, from (3.10) and (3.11) using the *Training* data set.

2. Compute the residuals using,

$$\epsilon_{1i} = S_{1i} - \begin{pmatrix} \hat{\alpha}_1 & \hat{\beta}_1 \end{pmatrix} \begin{pmatrix} F_{xi} & 0 \\ 0 & F_{yi} \end{pmatrix}, \tag{3.12}$$

$$\epsilon_{2i} = S_{2i} - \begin{pmatrix} \hat{\alpha}_2 & \hat{\beta}_2 \end{pmatrix} \begin{pmatrix} F_{xi} & 0 \\ 0 & F_{yi} \end{pmatrix}, \tag{3.13}$$

   where, $i = 1, 2, \ldots, n$.

3. Obtain the *bootstrap* data set by first forming the residual pool $\{\epsilon_{1i}, \epsilon_{2i},\ i = 1, 2, \cdots, n\}$ and then resampling it to obtain:

$$Training \begin{cases} S^*_{1i} = \begin{pmatrix} \hat{\alpha}_1 & \hat{\beta}_1 \end{pmatrix} \begin{pmatrix} F_{xi} & 0 \\ 0 & F_{yi} \end{pmatrix} + \epsilon^*_{1i}, \\ \\ S^*_{2i} = \begin{pmatrix} \hat{\alpha}_2 & \hat{\beta}_2 \end{pmatrix} \begin{pmatrix} F_{xi} & 0 \\ 0 & F_{yi} \end{pmatrix} + \epsilon^*_{2i}, \end{cases} \tag{3.14}$$

and,

$$Unknowns \begin{cases} V^*_{0j1} = V_{0j1} + \epsilon^*_{01}, \\ \\ V^*_{0j2} = V_{0j2} + \epsilon^*_{02}, \end{cases} \tag{3.15}$$

where, ϵ^*_{1i}, ϵ^*_{2i}, ϵ^*_{01}, and ϵ^*_{02} are random samples with replacement from the *residual pool*. Note that, we obtain separate residual pools from models (3.10) and (3.11). V_{0j1} and V_{0j2} are considered as the observed response vectors, and are used to predict the amount of relatively F_x, F_y.

4. Fit new models (3.14) to bootstrap data sets and obtain corresponding values of $\hat{\alpha}^*_1$, $\hat{\beta}^*_1$ and $\hat{\alpha}^*_2$,$\hat{\beta}^*_2$.

5. Estimate $\hat{F}_x$, and $\hat{F}_y$ using,

$$\begin{cases} V^*_{0J1} = \begin{pmatrix} \hat{\alpha}^*_1 & \hat{\beta}^*_1 \end{pmatrix} \begin{pmatrix} \hat{F}_{xi} & 0 \\ 0 & \hat{F}_{yi} \end{pmatrix}, \\ \\ V^*_{0J2} = \begin{pmatrix} \hat{\alpha}^*_2 & \hat{\beta}^*_2 \end{pmatrix} \begin{pmatrix} \hat{F}_{xi} & 0 \\ 0 & \hat{F}_{yi} \end{pmatrix}. \end{cases} \tag{3.16}$$

6. Repeat steps 3 to 5 B times.

7. Quantify confidence intervals for estimated forces.

Remark: In practice, when we employ the task data, the force is not necessarily applied in one direction, and one may expect to have force in both x and y directions. In this case, the regression model (3.16) in step 5 is replaced with the following equation to obtain $(\hat{F}_{xi}, \hat{F}_{yi})$ given the voltages that are observed:

$$\begin{cases} V^*_{0J1} = \hat{\alpha}^*_1 \hat{F}_{xi} + \hat{\beta}^*_1 \hat{F}_{yi}, \\ \\ V^*_{0J2} = \hat{\alpha}^*_2 \hat{F}_{xi} + \hat{\beta}^*_2 \hat{F}_{yi}. \end{cases} \tag{3.17}$$

where V^*_{0J1} and V^*_{0J2} represent the recorded voltages from strain gauges 1 and 2, respectively.

3.4 Eisenhart's Method in Calibration

Eisenhart (1939) suggested a methodology based on Student-t distribution to obtain confidence intervals for classical predictor (2.12), where we are dealing with univariate calibration problem. Suppose y_0 is observed, the confidence bounds for x_0 can be obtained by

$$\bar{x} + \frac{\hat{\beta}_1(y_0 - \bar{y})}{\hat{\beta}_1^2 - \frac{\hat{\sigma}^2 t^2}{S_{xx}}} \pm \frac{\hat{\sigma} t}{\hat{\beta}_1^2 - \frac{\hat{\sigma}^2 t^2}{S_{xx}}} \sqrt{(1 + \frac{1}{n})(\hat{\beta}_1^2 - \frac{\hat{\sigma}^2 t^2}{S_{xx}}) + \frac{(y_0 - \bar{y})^2}{S_{xx}}}, \tag{3.18}$$

where, S_{xx}, $(\bar{x}, \bar{y})$, and $\hat{\sigma}$ are defined in (2.8), (2.9), and (2.30), respectively. And $t = (1 - \frac{\alpha}{2})\%$ percentile of the Student-t distribution with $(n - 2)$ DoF.

3.4.1 Multivariate Eisenhart's Method for Calibration

In this section, we provide a multivariate extension to the Eisenhart's calibration method. The major impact of this extension is that it provides confidence intervals

without running any simulations, therefore, the results are obtained much faster. Suppose we observe $V_0 = (V_{01}, V_{02})$. The extension of the Eisenhart's point estimates and confidence intervals of F_x and F_y are obtained by solving the following system of equations

$$\begin{cases} (\hat{\alpha}_1 \hat{F}_{0x} + \hat{\beta}_1 \hat{F}_{0y} - V_{01})^2 = t^2 MSE_1(1 + B_{xy}), \\ (\hat{\alpha}_2 \hat{F}_{0x} + \hat{\beta}_2 \hat{F}_{0y} - V_{02})^2 = t^2 MSE_2(1 + B_{xy}), \end{cases} \tag{3.19}$$

where, MSE_1 and MSE_2 are the mean squared errors obtained from S_1 (3.10) and S_2 (3.11), respectively. Moreover,

$$B_{xy} = \begin{pmatrix} \hat{F}_{0x} & \hat{F}_{0y} \end{pmatrix} (F^\top F)^{-1} \begin{pmatrix} \hat{F}_{0x} \\ \hat{F}_{0y} \end{pmatrix}, \; F = (F_{1x}, \cdots, F_{nx}, F_{1y}, \cdots, F_{ny})^\top, \tag{3.20}$$

In (3.19), $\hat{\alpha}_1$, $\hat{\alpha}_2$, $\hat{\beta}_1$, $\hat{\beta}_2$ are also obtained through (3.10) and (3.11), and $t = (1 - \frac{\alpha}{2})\%$ percentile of the Student-t distribution with $(n-2)$ degree of freedom (DoF). We first obtain the point estimates $\hat{F}_{0x}$ and $\hat{F}_{0y}$ from (3.19), and then calculate the roots of the first and second quadratic equations to construct confidence intervals for $\hat{F}_{0x}$ and $\hat{F}_{0y}$, respectively.

Lemma 3.4.1. *Asymptotic approximations of the MSE and the Bias of $\hat{F}_{0x}$ are given by*

$$Bias(\hat{F}_{0x}) = \frac{V_{01} - \beta_1 F_{0y} - \alpha_1 F_{0x}}{\alpha_1} + (V_{01} - \beta_1 F_{0y}) \frac{MSE_1 \sum_{i=1}^{n} F_{iy}^2}{\alpha_1^3 A_{xy}}$$

$$- \frac{F_{0y} MSE_1 \sum_{i=1}^{n} F_{ix} F_{iy}}{A_{xy}}, \tag{3.21}$$

$$MSE(\hat{F}_{0x}) = \frac{MSE_1}{A_{xy}} \sum_{i=1}^{n} \left(\frac{F_{0y} F_{ix}}{\alpha_1} - \frac{(V_{01} - \beta_1 F_{0y}) F_{iy}}{\alpha_1^2} \right)^2 + Bias^2(\hat{F}_{0x}),$$

where, $A_{xy} = \sum_{i=1}^{n} F_{ix}^2 \sum_{i=1}^{n} F_{iy}^2 - (\sum_{i=1}^{n} F_{ix} F_{iy})^2$.

Proof. Suppose, $V_{01} = \hat{\alpha}_1 F_{0x} + \hat{\beta}_1 F_{0y}$, then we can estimate F_{0x} by

$$\hat{F}_{0x} = \frac{V_{01} - \hat{\beta}_1 F_{0y}}{\hat{\alpha}_1}.$$

Note that as we observe V_{01}, it is considered to be constant. Here, first we obtain variances and covariance for the coefficients through calculating $MSE_1 (F^\top F)^{-1}$, where F is given in (3.20). Therefore

$$\hat{\alpha}_1 \sim N\big(\alpha_1, \frac{MSE_1}{A_{xy}} \sum_{i=1}^{n} F_{iy}^2\big), \quad \hat{\beta}_1 \sim N\big(\beta_1, \frac{MSE_1}{A_{xy}} \sum_{i=1}^{n} F_{ix}^2\big),$$

$$Cov(\hat{\alpha}_1, \hat{\beta}_1) = \frac{-MSE_1}{A_{xy}} \sum_{i=1}^{n} F_{ix} F_{iy}.$$

We obtain expectation and variance of $\hat{F}_{0x}$ using the expressions given in (2.22) and (2.23), where

$$E(V_{01} - \hat{\beta}_1 F_{0y}) = V_{01} - \beta_1 F_{0y}, \quad Var(V_{01} - \hat{\beta}_1 F_{0y}) = \frac{F_{0y}^2 MSE_1}{A_{xy}} \sum_{i=1}^{n} F_{ix}^2. \quad (3.22)$$

To this end,

$$\begin{aligned} E(\hat{F}_{0x}) &= E(\frac{V_{01} - \hat{\beta}_1 \hat{F}_{0y}}{\hat{\alpha}_1}) \\ &= F_{0x} + \frac{V_{01} - \beta_1 F_{0y} - \alpha_1 F_{0x}}{\alpha_1} + MSE_1 \sum_{i=1}^{n} F_{iy}^2 \frac{(V_{01} - \beta_1 F_{0y})}{\alpha_1^3 A_{xy}} \\ &\quad - MSE_1 \sum_{i=1}^{n} F_{ix} F_{iy} \frac{F_{0y}}{\alpha_1^2 A_{xy}}. \end{aligned} \quad (3.23)$$

Also,

$$\begin{aligned} Var(\hat{F}_{0x}) &= Var(\frac{V_{01} - \hat{\beta}_1 \hat{F}_{0y}}{\hat{\alpha}_1}) \\ &= \frac{MSE_1}{A_{xy}} \sum_{i=1}^{n} \left(\frac{F_{0y} F_{ix}}{\alpha_1} - \frac{(V_{01} - \beta_1 F_{0y}) F_{iy}}{\alpha_1^2} \right)^2. \end{aligned} \quad (3.24)$$

Using (3.23) and(3.24), we can obtain expressions for the Bias and MSE, as follow:

$$\begin{aligned} Bias(\hat{F}_{0x}) &= E(\hat{F}_{0x}) - F_{0x} \\ &= F_{0x} + \frac{V_{01} - \beta_1 F_{0y} - \alpha_1 F_{0x}}{\alpha_1} + (V_{01} - \beta_1 F_{0y}) \frac{MSE_1 \sum_{i=1}^{n} F_{iy}^2}{\alpha_1^3 A_{xy}} \\ &\quad - \frac{F_{0y} MSE_1 \sum_{i=1}^{n} F_{ix} F_{iy}}{A_{xy}} - F_{0x}, \end{aligned} \quad (3.25)$$

and

$$MSE(\hat{F}_{0x}) = Var(\hat{F}_{0x}) + Bias^2(\hat{F}_{0x})$$

$$= \frac{MSE_1}{A_{xy}} \sum_{i=1}^{n} \left(\frac{F_{0y}F_{ix}}{\alpha_1} - \frac{(V_{01} - \beta_1 F_{0y})F_{iy}}{\alpha_1^2} \right)^2 + Bias^2(\hat{F}_{0x}). \quad (3.26)$$

□

In order to obtain the confidence bounds for F_{0x}, first we obtain $\hat{F}_{0x}$ and $\hat{F}_{0y}$ from (3.19), then we assume $\hat{F}_{0y}$ is known in $(\hat{\alpha}_1\hat{F}_{0x} + \hat{\beta}_1\hat{F}_{0y} - V_{01})^2 = t^2 MSE_1(1 + B_{xy})$. Therefore, our problem reduced to a quadratic formula with respect to $\hat{F}_{0x}$, that is unknown. One can use the *quadratic formula* to find the roots of this equation, that are upper and lower bounds of desired confidence interval. . One can obtain confidence bound for $\hat{F}_{0y}$ in a similar way, by assuming $\hat{F}_{0x}$ is known and using $(\hat{\alpha}_2\hat{F}_{0x} + \hat{\beta}_2\hat{F}_{0y} - V_{02})^2 = t^2 MSE_2(1 + B_{xy})$.

3.5 Obtaining Calibration Data Through Experiment

In order to obtain the data set for *Training*, the data from 20 trial runs were used during calibration of the bipolar forceps measured using the developed automatic calibration station, (see Figure 3.1). The motor, connected to the force sensor on the calibration station, was programmed to move in two directions: forward (moving towards the point o in Figure 3.2) and backward (moving away from o). Data measurement was performed for 10 times under the same test conditions along each direction. Therefore, in total, 40 sets of data were collected when the force was

applied along x axis (10 trial runs for forward and 10 trials for backward, for the left and right prongs) and 40 data sets for when the force was applied along y axis (10 trials for forward and 10 trial runs for backward, for the left and right prongs). In forward motion, the tips were applied a force of 0 N to 2 N, and in backward, the force reduced from 2 N to 0 N. Note that the force of 2 N is the peak force that we can expect during the performance of a neurosurgery (Zareinia et al., 2015).

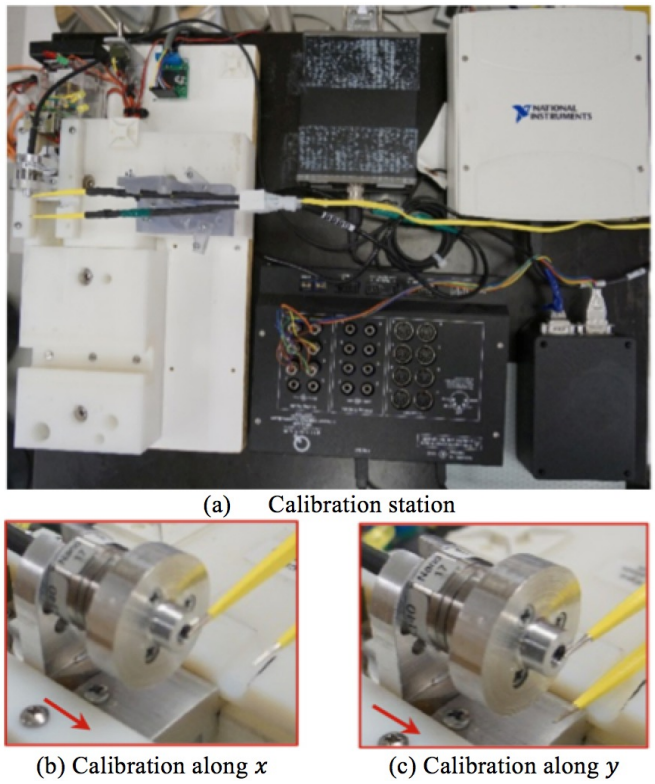


Figure 3.1: (a) Calibration station and the setup used to calibrate the instrumented bipolar forceps along (b) x axis and (c) y axis. Arrows show the direction of the applied force by the motorized system connected to the force sensor.Source: Azimaee et al., "Nonparametric bootstrap Technique for Calibrating Surgical SmartForceps: Theory and Application." Revision Submitted (2017).

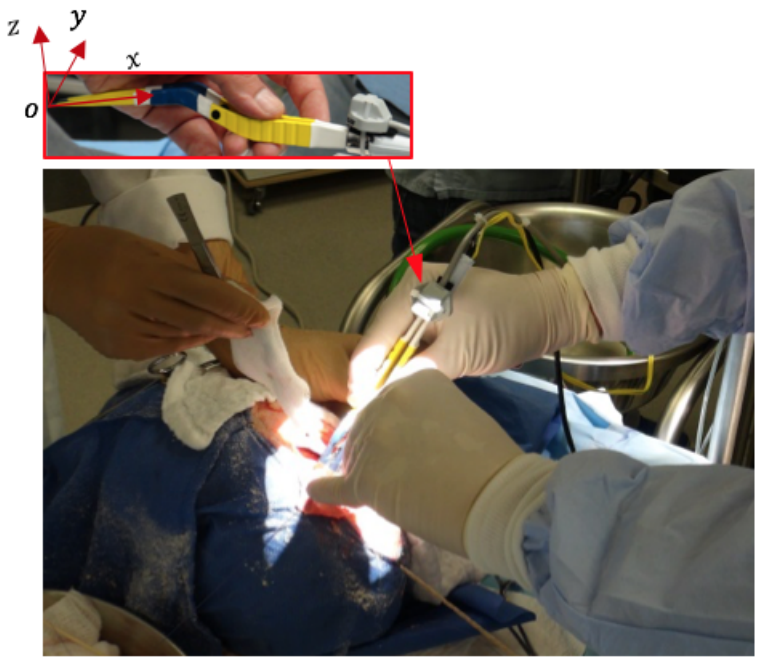


Figure 3.2: Surgeons are performing neurosurgery by using SmartForceps^{TM} to measure applied forces along x, y, and z axes. Source: Maddahi et al., "Quantifying workspace and forces of surgical dissection during robot-assisted neurosurgery." The International Journal of Medical Robotics and Computer Assisted Surgery 12, no. 3 (2016): 528-537.

3.6 Results from the Bootstrap Method

This section presents the results obtained by implementing the methodologies explained through this chapter. A comprehensive graphical and numerical analysis of results is also presented.

3.6.1 Univariate Calibration

As a typical example, suppose we aim at predicting the amount of *force* for a given voltage of $V_1 = 0.26$ V. Results of this examination, using two methods, under 95%

bootstrap confidence intervals, are presented in Table 3.1.

Table 3.1: 95% confidence intervals obtained using different methods of univariate calibration when $V_1 = 0.26$ V is observed, and the true force of $F_x = 1.5$ N is given.

Method	True Force	Confidence Interval
Percentile Bootstrap	1.5	(1.466 , 1.539)
Bootstrap t	1.5	(1.460 , 1.541)

3.6.2 Multivariate Calibration

Force along x direction

In this part, predictions of applied forward and backward forces to the right and left forceps tips along x direction are presented as confidence intervals. As observed from Table 3.2, the length of force intervals obtained for the right tip in forward direction are narrower than the backward direction. For instance, when the true force of F_x is equal to 1.7 N, the force interval in forward direction for the right tip is (1.684 N, 1.730 N), while, in backward is (1.698 N, 1.750 N). This trend is also observed along left tip as can be seen in Table 3.3. For instance, when the true force of $F_x = 1.7$ N is considered, the force interval in the forward direction of the left tip is (1.672 N, 1.745 N), that is again narrower than interval in the backward direction, (1.654 N, 1.741 N). As expected, all intervals contain the true values of force. The bounds for forward and backward $F_x \in [0\text{ N}, 1\text{ N}]$ are plotted in dotted and dashed lines, respectively, as shown in Figs. 3.3 and 3.4.

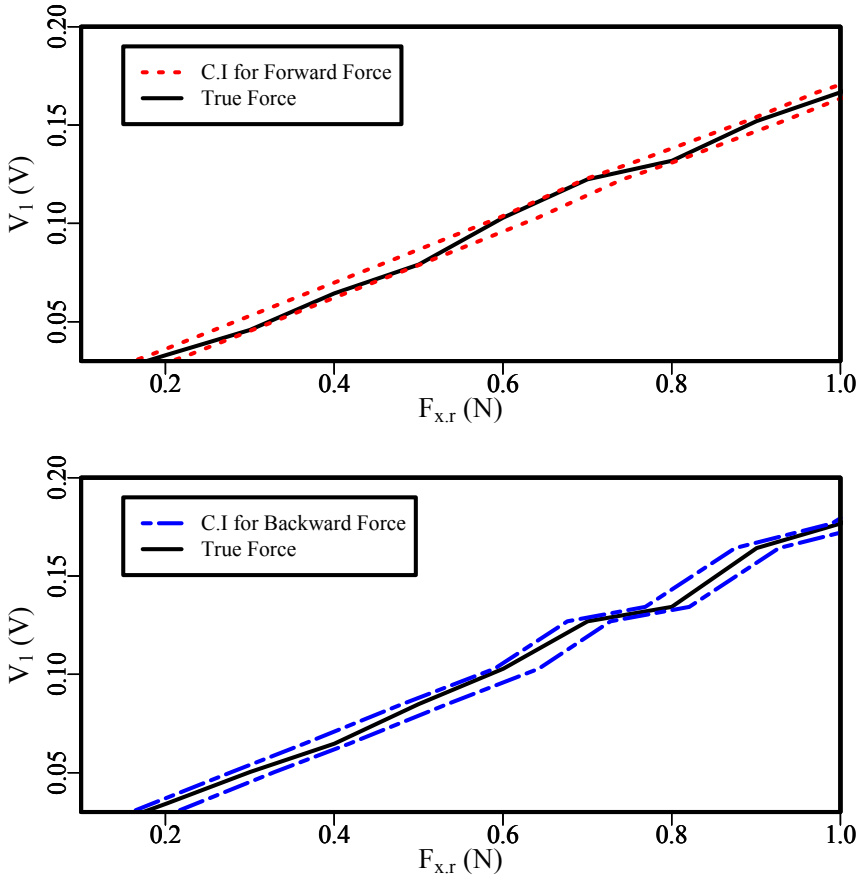


Figure 3.3: Confidence intervals of the forces at the right tip, obtained using the bootstrap method in forward (top) and backward (bottom) directions, when force along x is applied to the right prong ($\hat{F}_x$).

Table 3.2: Estimated confidence intervals for different amounts of F_x applied to right forceps tip. To examine results of the bootstrap method, amounts of true forces were considered as unknown values.

Forces Applied to Right Tip			
Forward Force		Backward Force	
True F_x	C.I	True F_x	C.I
0.1	(0.095 , 0.139)	2.0	(1.950 , 2.002)
0.2	(0.181 , 0.227)	1.9	(1.861 , 1.912)
0.3	(0.258 , 0.303)	1.8	(1.784 , 1.836)
0.4	(0.367 , 0.413)	1.7	(1.698 , 1.750)
0.5	(0.455 , 0.502)	1.6	(1.590 , 1.640)
0.6	(0.595 , 0.641)	1.5	(1.461 , 1.516)
0.7	(0.697 , 0.742)	1.4	(1.378 , 1.429)
0.8	(0.760 , 0.807)	1.3	(1.286 , 1.338)
0.9	(0.888 , 0.932)	1.2	(1.199 , 1.252)
1.0	(0.972 , 1.019)	1.1	(1.056 , 1.107)
1.1	(1.087 , 1.132)	1.0	(0.990 , 1.043)
1.2	(1.185 , 1.230)	0.9	(0.875 , 0.927)
1.3	(1.266 , 1.311)	0.8	(0.769 , 0.821)
1.4	(1.389 , 1.436)	0.7	(0.676 , 0.727)
1.5	(1.464 , 1.509)	0.6	(0.544 , 0.642)
1.6	(1.579 , 1.623)	0.5	(0.481 , 0.534)
1.7	(1.684 , 1.730)	0.4	(0.364 , 0.418)
1.8	(1.799 , 1.844)	0.3	(0.279 , 0.330)
1.9	(1.883 , 1.931)	0.2	(0.184 , 0.236)
2.0	(1.991 , 2.038)	0.1	(0.093 , 0.146)

Table 3.3: Estimated confidence intervals for different amounts of F_x applied to left forceps tips. To examine results of the bootstrap method, amounts of true forces were considered as unknown values.

Forces Applied to Left Tip			
Forward Force		Backward Force	
True F_x	C.I	True F_x	C.I
0.1	(0.069 , 0.142)	2.0	(1.921 , 2.008)
0.2	(0.136 , 0.208)	1.9	(1.837 , 1.922)
0.3	(0.269 , 0.341)	1.8	(1.755 , 1.839)
0.4	(0.361 , 0.433)	1.7	(1.654 , 1.741)
0.5	(0.459 , 0.532)	1.6	(1.559 , 1.646)
0.6	(0.533 , 0.606)	1.5	(1.490 , 1.577)
0.7	(0.677 , 0.750)	1.4	(1.352 , 1.437)
0.8	(0.773 , 0.846)	1.3	(1.256 , 1.342)
0.9	(0.861 , 0.933)	1.2	(1.171 , 1.259)
1.0	(0.989 , 1.062)	1.1	(1.083 , 1.171)
1.1	(1.075 , 1.147)	1.0	(0.969 , 1.056)
1.2	(1.156 , 1.231)	0.9	(0.860 , 0.947)
1.3	(1.283 , 1.355)	0.8	(0.797 , 0.882)
1.4	(1.372 , 1.444)	0.7	(0.680 , 0.767)
1.5	(1.470 , 1.543)	0.6	(0.570 , 0.658)
1.6	(1.588 , 1.662)	0.5	(0.450 , 0.536)
1.7	(1.672 , 1.745)	0.4	(0.329 , 0.416)
1.8	(1.772 , 1.845)	0.3	(0.255 , 0.340)
1.9	(1.893 , 1.966)	0.2	(0.163 , 0.249)
2.0	(1.958 , 2.030)	0.1	(0.076 , 0.162)

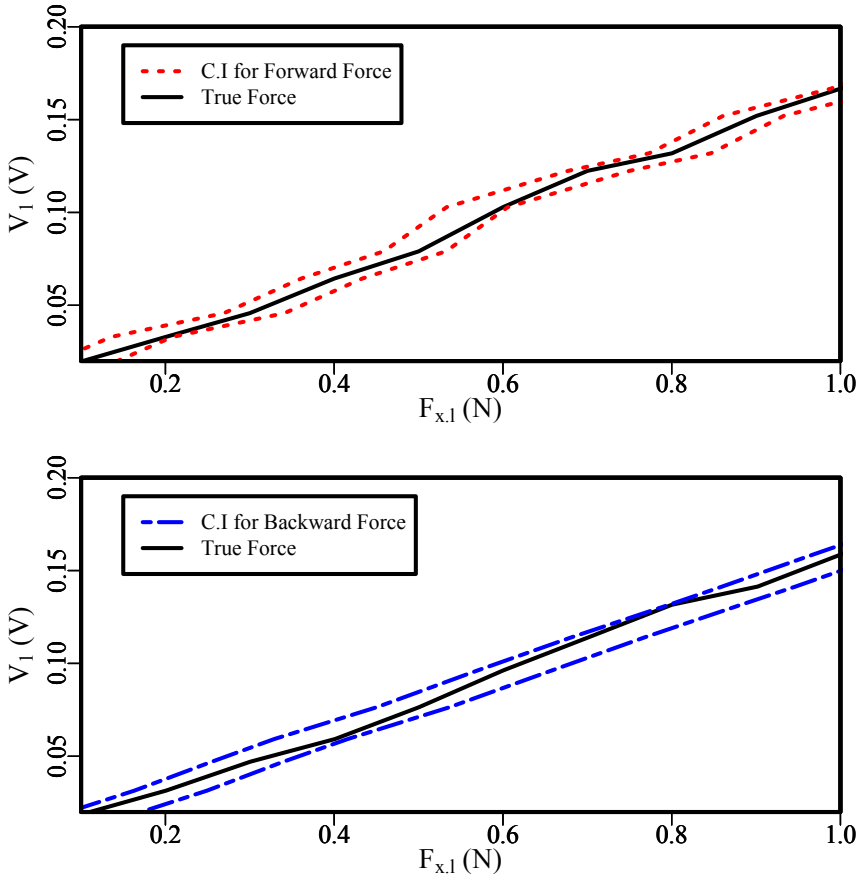


Figure 3.4: Confidence intervals of forces at the left tip, obtained using the bootstrap method in forward (top) and backward (bottom) directions when F_x is applied to the left prong.

Force along y direction

Similar forces as calibration along x axis were applied to the forceps tips along y direction, first to the right tip and then to the left tip. Here again, lengths of force intervals, obtained in backward direction along right tip, were broader than intervals in forward motion (see Table 3.4). However, we could not see this trend for the forces along left tip; it means that the obtained intervals for forces in forward direction

are broader than the ones for backward direction along left tip (see Table 3.5). For instance, when the true force of 1 N was given, the estimated force intervals of the right tip in backward motion was (0.933 N, 1.047 N) compared to (0.992 N, 1.087 N) in forward direction. But the force intervals of the left tip were (0.909 N, 1.063 N) in backward motion and (0.900 N , 1.058 N) in forward direction. Figs. 3.5 and 3.6 illustrate the confidence force bounds in forward and backward directions, respectively.

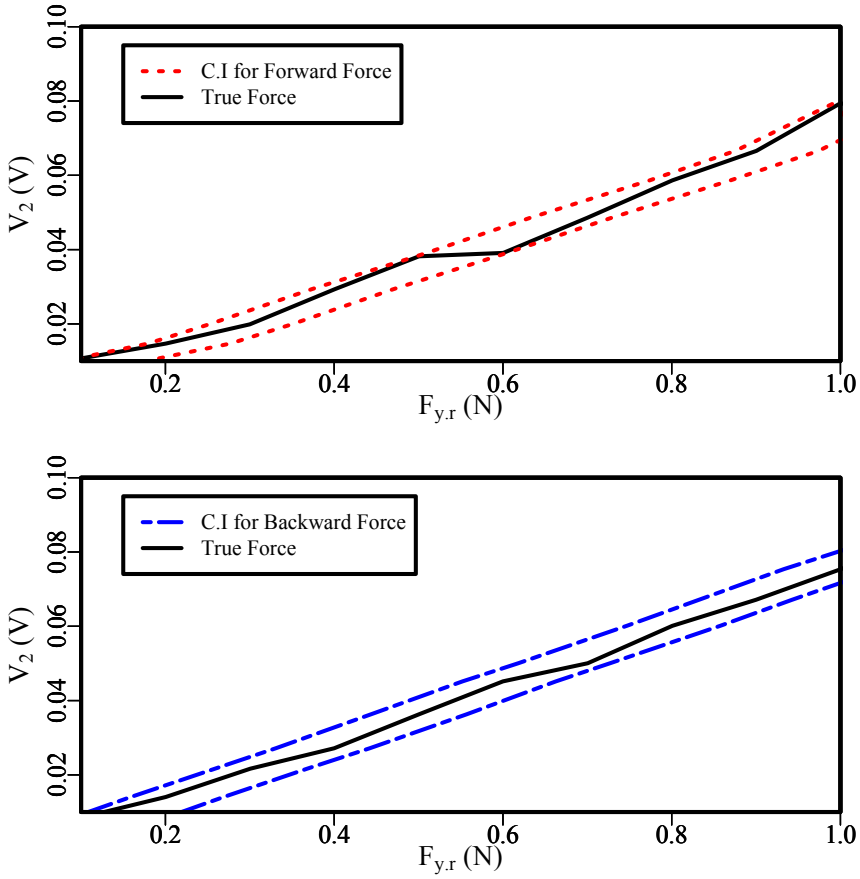


Figure 3.5: Confidence intervals of forces exerted on the right tip, obtained using the bootstrap method for the forward (top) and backward (bottom) directions when the force is applied along y axis.

Table 3.4: Estimated confidence intervals for different amounts of F_y applied to right tip. To examine results of the bootstrap method, amounts of true forces were considered as unknown values.

Forces Applied to Right Tip			
Forward Force		Backward Force	
True F_y	C.I	True F_y	C.I
0.1	(0.096 , 0.192)	2.0	(1.861 , 1.975)
0.2	(0.179 , 0.277)	1.9	(1.805 , 1.914)
0.3	(0.252 , 0.350)	1.8	(1.696 , 1.807)
0.4	(0.373 , 0.470)	1.7	(1.594 , 1.702)
0.5	(0.497 , 0.592)	1.6	(1.585 , 1.692)
0.6	(0.510 , 0.605)	1.5	(1.428 , 1.539)
0.7	(0.632 , 0.729)	1.4	(1.300 , 1.411)
0.8	(0.774 , 0.870)	1.3	(1.222 , 1.331)
0.9	(0.875 , 0.974)	1.2	(1.146 , 1.258)
1.0	(0.992 , 1.087)	1.1	(1.058 , 1.172)
1.1	(1.005 , 1.101)	1.0	(0.933 , 1.047)
1.2	(1.147 , 1.244)	0.9	(0.832 , 0.943)
1.3	(1.229 , 1.324)	0.8	(0.747 , 0.858)
1.4	(1.349 , 1.446)	0.7	(0.618 , 0.727)
1.5	(1.494 , 1.591)	0.6	(0.552 , 0.663)
1.6	(1.554 , 1.650)	0.5	(0.444 , 0.557)
1.7	(1.637 , 1.734)	0.4	(0.331 , 0.442)
1.8	(1.791 , 1.889)	0.3	(0.259 , 0.368)
1.9	(1.874 , 1.971)	0.2	(0.159 , 0.269)
2.0	(1.910 , 2.007)	0.1	(0.090 , 0.201)

Table 3.5: Estimated confidence intervals for different amounts of F_y applied to left tip. To examine results of the bootstrap method, amounts of true forces were considered as unknown values.

Forces Applied to Left Tip			
Forward Force		Backward Force	
True F_y	C.I	True F_y	C.I
0.1	(0.025 , 0.180)	2.0	(1.906 , 2.061)
0.2	(0.137 , 0.299)	1.9	(1.776 , 1.925)
0.3	(0.216 , 0.372)	1.8	(1.748 , 1.900)
0.4	(0.307 , 0.467)	1.7	(1.655 , 1.808)
0.5	(0.411 , 0.568)	1.6	(1.478 , 1.633)
0.6	(0.493 , 0.652)	1.5	(1.424 , 1.575)
0.7	(0.633 , 0.788)	1.4	(1.387 , 1.540)
0.8	(0.694 , 0.848)	1.3	(1.213 , 1.368)
0.9	(0.820 , 0.977)	1.2	(1.156 , 1.309)
1.0	(0.900 , 1.058)	1.1	(1.031 , 1.183)
1.1	(1.021 , 1.179)	1.0	(0.909 , 1.063)
1.2	(1.145 , 1.304)	0.9	(0.826 , 0.978)
1.3	(1.203 , 1.361)	0.8	(0.698 , 0.854)
1.4	(1.373 , 1.529)	0.7	(0.636 , 0.798)
1.5	(1.404 , 1.558)	0.6	(0.495 , 0.649)
1.6	(1.462 , 1.620)	0.5	(0.410 , 0.564)
1.7	(1.634 , 1.794)	0.4	(0.300 , 0.459)
1.8	(1.726 , 1.884)	0.3	(0.212 ,0.364)
1.9	(1.753 , 1.906)	0.2	(0.135 , 0.289)
2.0	(1.881 , 2.038)	0.1	(0.017 , 0.171)

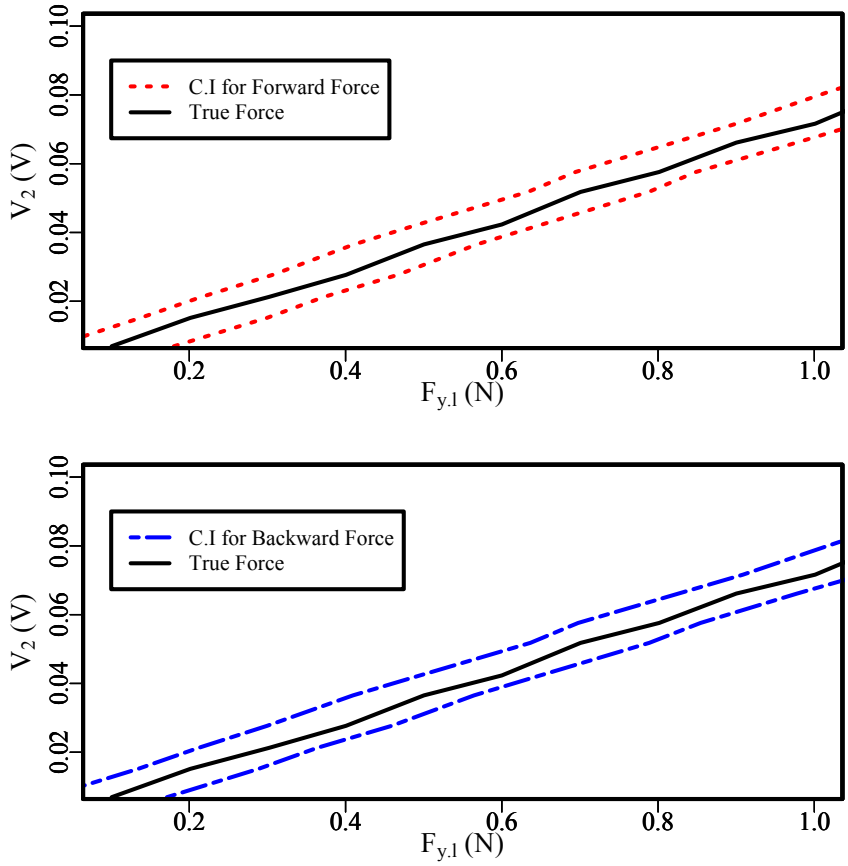


Figure 3.6: Confidence intervals of forces applied to the left tip, calculated using the bootstrap method for the forward (top) and backward (bottom) directions when F_y is applied.

3.7 Real Field Application and Further Discussions

The forces of tool tissue interaction, while providing a force feedback to the surgeon, is an important information source for the ongoing development of techniques to train neurosurgery residents and novice surgeons, as well as assess their surgical skills. Specifically, the quantification of tool-tissue interaction forces will allow

implementation of a high force warning system, such that surgeons are warned when safety threshold for forces of tool-tissue interaction is reached (Payne et al., 2015, Payne and Yang, 2014). In this section, we provide a real field application of our proposed methods and discuss the accuracy of each approach.

3.7.1 Replications of the Voltages

Results presented in this study were calculated using one single replication of unknown voltages. However, it is also important to investigate the effect of using r replications of the voltages for the same amount of force, and investigate the effect of the number of replications on the size of force interval.

As a typical test, we used replications of the voltages for $F_x = 1.5$ N and $F_y = 1.5$ N in both left and right tips. The replications were extracted from the real data set measured using the calibration station. Figure 3.7 illustrates the confidence intervals for different number of replications (r). As observed, an increase in the number of replications results in a narrower force interval.

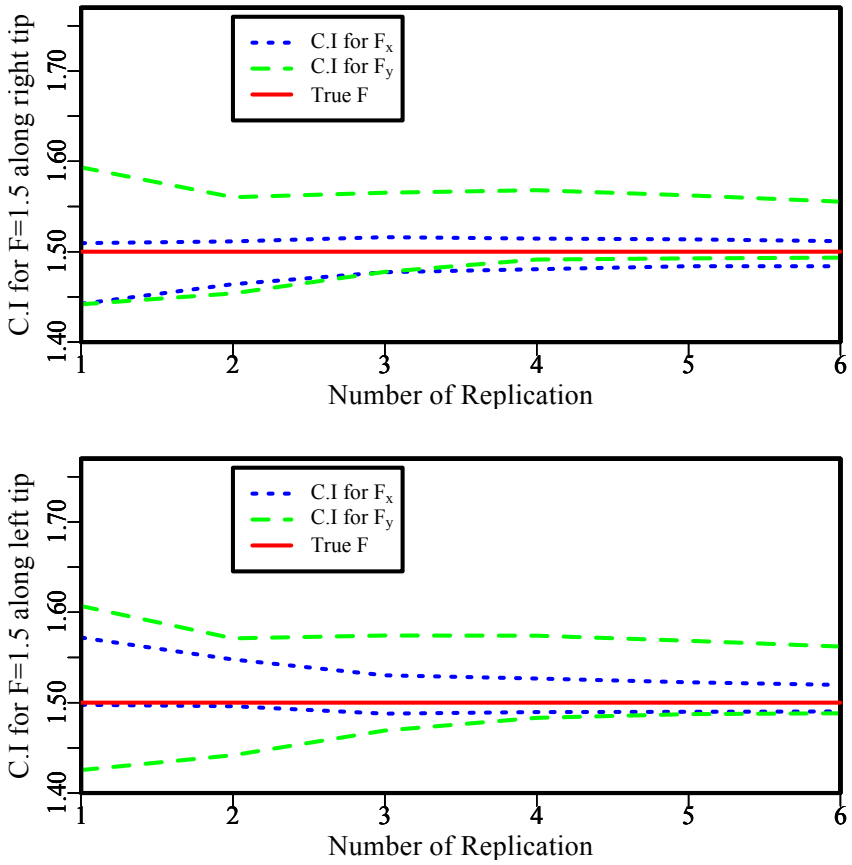


Figure 3.7: Confidence intervals obtained for different numbers of replications, when the true $F_x = 1.5$ N, and $F_y = 1.5$ N for the right tip (top) and the left tip (bottom). While dotted and dashed lines show the C.I for F_x and F_y applied to the tips, respectively. Solid line represents the true force value.

3.7.2 Accuracy of the Bootstrap Method

Comparison of the estimation based on the bootstrap method with the deterministic method was also performed. Considering the deterministic *Naïve* method, we estimated the matrix of coefficients with the data set obtained from calibration station in forward and backward direction to put in (3.27) and estimate the amount

Table 3.6: Comparison of F_x and F_y estimates using the bootstrap method, the Naïve technique, and the extension of Einsenhart's method for several true forces measured using calibration station.

Bootstrap Method					**Naïve** Method			**Extension of the Eisenhart's Method**			
True F_x	$\hat{F}_x$	C.I for F_x	Bias [a]	RMSE [b]	$\hat{F}_x$	Bias[c]	RMSE [b]	$\hat{F}_x$	C.I for F_x	Bias [d]	RMSE [b]
1.100	1.098	(1.087 , 1.133)	-0.002	0.017	1.107	0.007	-	1.128	(1.063 , 1.128)	-0.003	0.003
1.200	1.210	(1.198 , 1.244)	0.010	0.020	1.212	0.012	-	1.174	(1.174 , 1.238)	0.009	0.009
1.301	1.285	(1.274 , 1.319)	-0.015	0.023	1.287	-0.013	-	1.315	(1.251 , 1.315)	-0.015	0.015
1.400	1.400	(1.388 , 1.434)	0.000	0.017	1.401	0.001	-	1.363	(1.363 , 1.428)	0.001	0.001
1.500	1.476	(1.465 , 1.512)	-0.024	0.029	1.477	-0.023	-	1.506	(1.441 , 1.506)	-0.029	0.029
1.601	1.596	(1.584 , 1.630)	-0.005	0.018	1.597	-0.004	-	1.626	(1.561 , 1.626)	-0.004	0.004
1.700	1.699	(1.687 , 1.733)	-0.002	0.017	1.701	0.001	-	1.729	(1.664 , 1.729)	-0.001	0.001
1.800	1.810	(1.798 , 1.845)	0.010	0.020	1.789	0.011	-	1.784	(1.784 , 1.848)	0.011	0.011
1.900	1.911	(1.899 , 1.944)	0.010	0.020	1.927	0.027	-	1.885	(1.885 , 1.949)	0.013	0.013
1.991	1.987	(1.975 , 2.023)	-0.004	0.018	1.932	-0.059	-	2.018	(1.952 , 2.017)	-0.004	0.004
Bootstrap Method					**Naïve Method**			**Extension of the Eisenhart's Method**			
True F_y	$\hat{F}_y$	C.I for F_y	Bias[a]	RMSE[b]	$\hat{F}_y$	Bias[c]	RMSE[b]	$\hat{F}_y$	C.I for F_y	Bias [d]	RMSE [b]
1.100	1.152	(1.127 , 1.225)	0.052	0.064	1.122	0.022	-	1.065	(1.065 , 1.206)	0.045	0.045
1.201	1.242	(1.217 , 1.314)	0.041	0.056	1.170	-0.031	-	1.157	(1.157 , 1.298)	0.039	0.039
1.300	1.322	(1.297 , 1.395)	0.021	0.043	1.338	0.038	-	1.261	(1.261 , 1.402)	0.022	0.022
1.400	1.403	(1.379 , 1.476)	0.003	0.037	1.432	0.032	-	1.343	(1.343 , 1.485)	0.013	0.013
1.500	1.541	(1.516 , 1.612)	0.041	0.055	1.543	0.043	-	1.455	(1.455 , 1.597)	0.033	0.033
1.601	1.587	(1.562 , 1.658)	-0.014	0.040	1.636	0.035	-	1.673	(1.530 , 1.671)	-0.009	0.010
1.700	1.715	(1.690 , 1.787)	0.015	0.040	1.672	-0.028	-	1.631	(1.631 , 1.773)	0.015	0.0151
1.800	1.824	(1.799 , 1.896)	0.024	0.044	1.845	0.045	-	1.765	(1.765 , 1.906)	0.027	0.027
1.901	1.940	(1.914 , 2.013)	0.039	0.054	1.886	-0.015	-	1.881	(1.881 , 2.022)	0.042	0.042
1.990	2.025	(2.000 , 2.100)	0.035	0.051	1.968	-0.022	-	1.966	(1.966 , 2.107)	0.043	0.043

[a]Bias in Bootstrap method is the difference between expectation of Bootstrap's estimates and the true force value.

[b]MSE = square root of mean squared errors

[c]Bias in Navie method is defined as the difference between expectation of estimation based on the Naïve method and the true force.

[d]Bias in extension of the Eisenhart's method is the difference between expectation of estimation based on the extension of the Eisenhart's method and the true value of force.

of F_x and F_y,

$$\begin{pmatrix} V_1 \\ V_2 \end{pmatrix} = \begin{pmatrix} f_{1x} & f_{1y} \\ f_{2x} & f_{2y} \end{pmatrix} \begin{pmatrix} F_x \\ F_y \end{pmatrix}, \tag{3.27}$$

where f_{ij} is the slope of the fitted regression line of voltages obtained from strain gauge i as a result of applied force in j direction. The coefficients matrices for right and left prongs were equal to (3.28) and (3.29), respectively, where the matrices on the right side are for backward forces and the ones on the left are assigned to forward forces. Results of the calibration in forward and backward direction are shown in Figure 3.8.

$$\text{(Right, Forward)} = \begin{pmatrix} 0.16876 & 0.01222 \\ 0.02816 & 0.07195 \end{pmatrix}, \text{(Right,Backward)} = \begin{pmatrix} 0.17198 & 0.01393 \\ 0.02829 & 0.07839 \end{pmatrix}, \tag{3.28}$$

$$\text{(Left, Forward)} = \begin{pmatrix} 0.15565 & 0.01048 \\ 0.02657 & 0.07398 \end{pmatrix}, \text{ (Left, Backward)} = \begin{pmatrix} 0.15765 & 0.01290 \\ 0.02738 & 0.07310 \end{pmatrix}. \tag{3.29}$$

As listed in Table 3.6, the bias of the *Naïve* method in most cases is greater than the bias for the bootstrap method. Moreover, the amount of the force estimated using the mean value of the bootstrap force estimates is closer to the true force than *Naïve* method. However, overall there is not that much difference between bootstrap point estimations and the *Naïve* method estimates, which can be explained with the small variation of residuals obtained from fitting models S_1 (3.10) and S_2 (3.11) in bootstrapping, which means that the values of residuals are relatively small.

In next step, the confidence bounds obtained from the bootstrap technique were

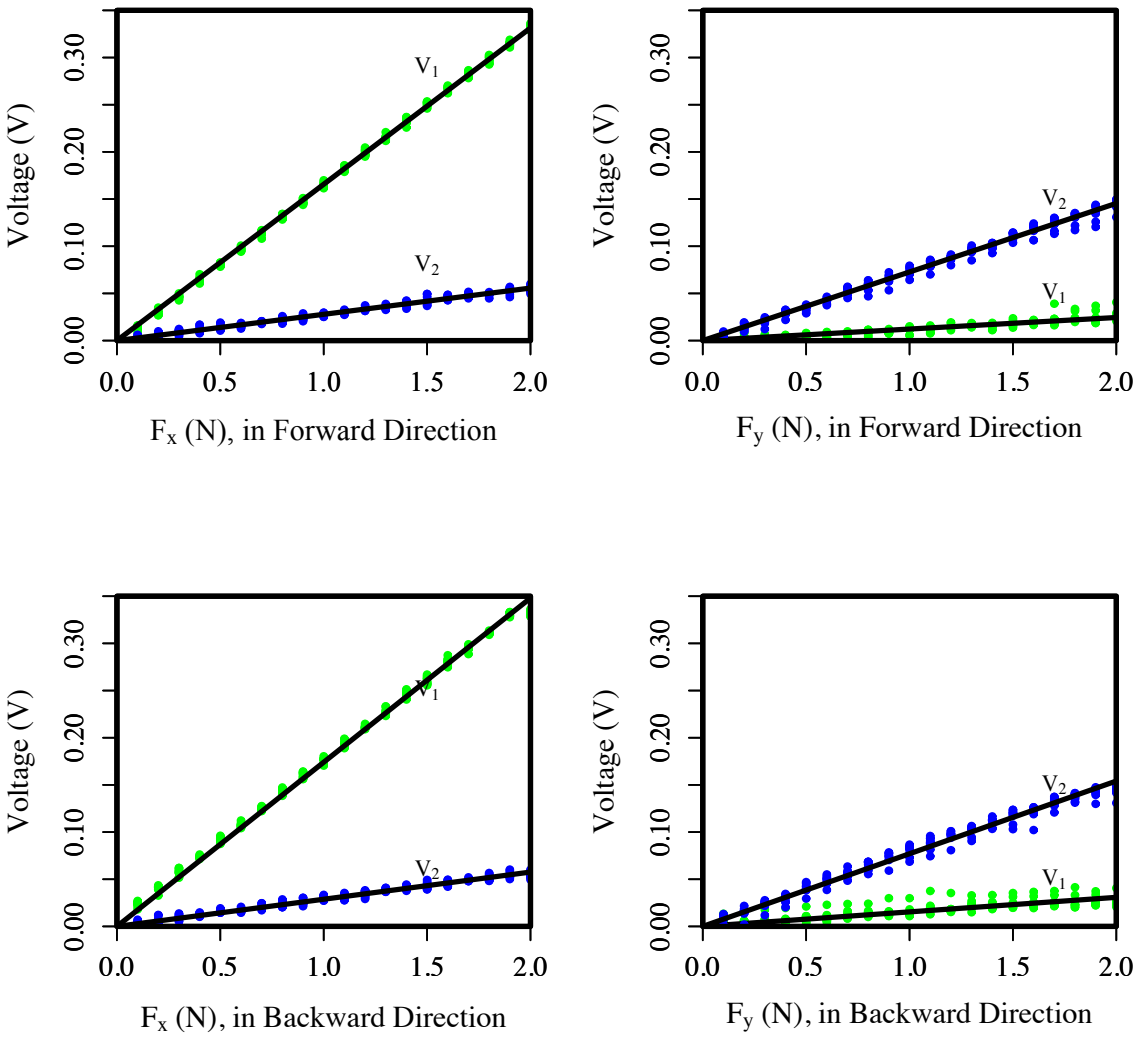


Figure 3.8: Calibration forces applied to the right tip in x and y directions *vs.* voltages read from strain gauges for 10 trials in forward (top) and backward (bottom) directions. Fitted regression lines are also presented for V_1 and V_2 *vs.* F_x and F_y, respectively.

compared against the ones obtained from the *extension of the Eisenhart's* method for multiple regression models(Eisenhart, 1939).

As observed in Table 3.6, the length of the intervals obtained from the bootstrap method are significantly narrower than those obtained by the extension of the Eisenhart's method. For example, when $F_x = 1.8$ N is applied, the confidence bound that is given by the bootstrap method is (1.798 N , 1.845 N), while the interval that is obtained by extension of Eisenhart's method is (1.780 N , 1.844 N). The difference in the confidence interval bounds is even more evident for F_y.

3.8 Concluding Remarks

In this chapter, the bootstrap technique was used to calibrate the interaction forces between a surgical tool (SmartForceps) and the brain tissue. The instrumented bipolar forceps is generally employed to conduct a variety of neurosurgical tasks including dissection, coagulation and squeezing the brain tissue. Two sets of strain gauges were mounted on each prong of the forceps, away from tips to measure the components of the interaction force in four directions. In practice, the interaction forces are calculated using reading the voltages of the strain gauges during surgery. Therefore, a model was required to map the voltages read from strain gauges and the forces applied to the forceps tips. The *Naïve* model has previously been employed by Zareinia et al. (2015) to determine the force. In this study, the bootstrap technique was used to calibrate the forces.The bootstrap method takes into account effects of the surgeon who is using the bipolar forceps in the estimation process through a pooling stage during required resampling steps in the procedure. Obtained results

indicated that the bootstrap technique provides a more accurate estimate of the force value, compared to the *Naïve* method. More precisely, the bias obtained from the bootstrap technique was smaller. Furthermore, the *Naïve* method only provides point estimates while the bootstrap provided confidence intervals, and hence, the bootstrap technique provided a more reliable estimation.

Chapter 4

Multilevel Modeling and Bayesian Approach for Calibrating SmartForceps^{TM}

In this chapter, limitations and difficulties associated with mounting the third pair of strain gauges on the smartForceps^{TM} are discussed. Furthermore, several statistical approaches such as weighted least squares, multilevel modeling, and Bayesian approach in multi-level modeling are suggested to address these problems. We consider the calibration problem under the Bayesian approach using both symmetric and asymmetric loss functions. To this end, LINEX loss function is used as an alternative to the usual squared error loss (SEL) in order to asymmetrically penalize over and under estimation in the underlying calibration problem.

4.1 Problem with 3-D Data

In this section, first we describe the problem of predicting the amount of forces, while observed voltages from all three pairs of strain gauges are considered. Moreover,

graphical illustrations and tables are used to clarify this problem. Then, bootstrap method that is proposed in Chapter 3, is employed to predict the amount of forces, and results are obtained through weighted least squares as well as ordinary least squares prediction approaches. First, we briefly explain the challenges involved in using 3-D data.

4.1.1 Data Set Specification

Calibration data set for the z direction is obtained in a similar fashion explained in Section 3.5. Another pair of strain gauges were mounted on the prongs of the *SmartForceps*$^{\text{TM}}$ to record voltages in z direction. Therefore, calibration data set is obtained by applying force along x, y, and z directions, where each pair of the strain gauges measures the output voltages in corresponding direction (S_1, S_2, and S_3 are used to obtain voltages when force is applied in x, y, and z directions, respectively).

When force is applied in a certain direction, we expect to observe substantial amount of output voltage from the strain gauge that is installed to measure voltages along that particular direction. For instance, applied force in x direction, should result in large output voltages from S_1. Similarly, applied force in z direction should result in output voltages in S_3 that are significantly higher than those recorded in S_1 and S_2. However, an anomaly occurs in the y direction *i.e.* when force is applied in the y direction, considerable amount of voltages are observed from S_3 instead of S_2, and observed voltages in S_2 is negligible. This problem is illustrated in Figure 4.1: when force is applied in x direction, observed voltages in S_2 and S_3 are relatively small and close to zero, while S_1 voltages are increasing corresponding to the increase in F_x. However, when F_y is applied, observed voltages in S_2 and

S_1 are close to zero, but voltages in S_3 are increasing remarkably with increasing the amount of applied F_y's. Nevertheless, we expected to observe voltages in S_2 rise with increasing F_y. The last plot of Figure 4.1, shows the behavior of the three strain gauges when force is applied in z direction.

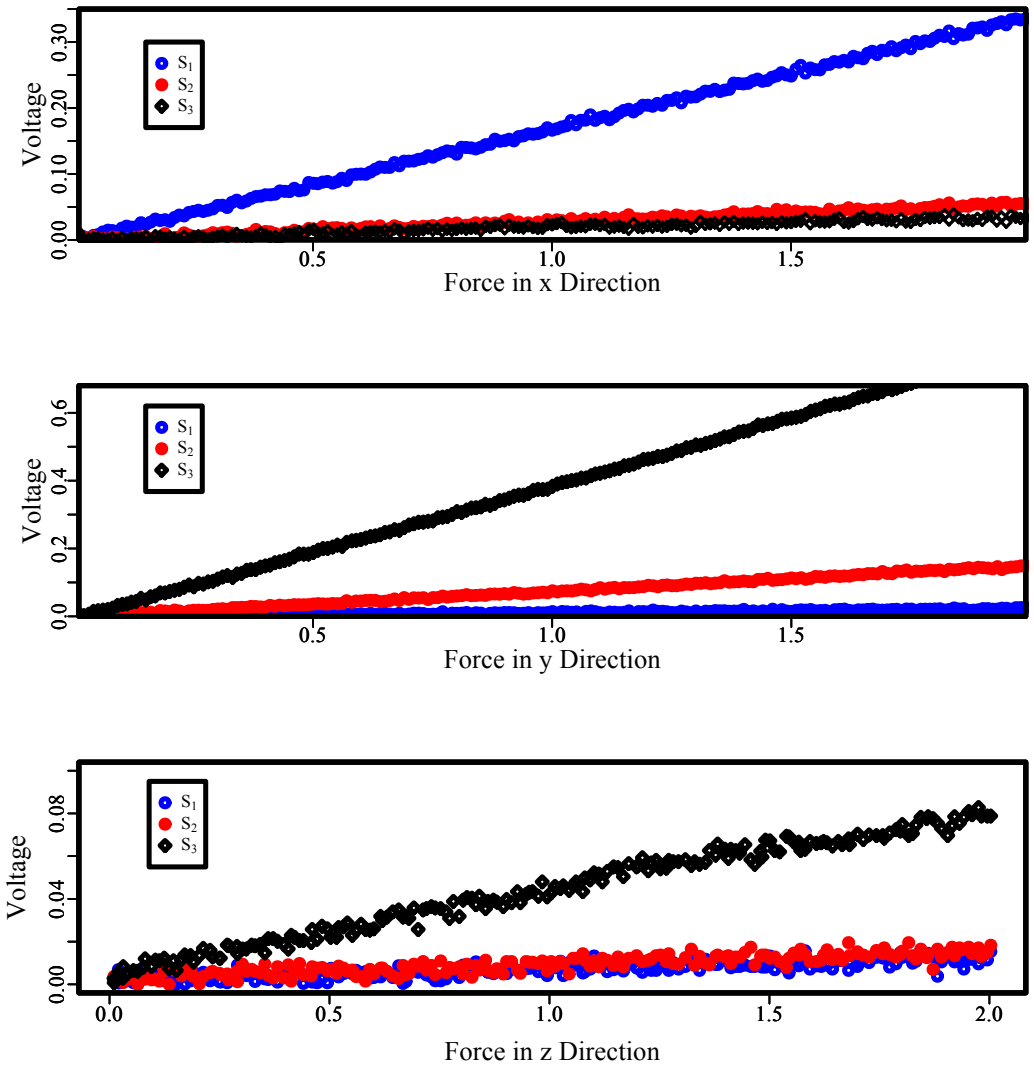


Figure 4.1: Observed voltages from 3 different strain gauges (S_1, S_2, and, S_3) are presented when force is applied in x (top), y (middle), and z (bottom) directions.

This problem is probably caused by improper position of the third pair of strain gauges. In other words, when force is applied in Y direction, first it deforms S_3 instead of S_2. And this is mainly because there should not be any alteration in the shape of conventional neurosurgical forceps.

Another issue that affects force prediction having 3 pairs of strain gauges is the variability of observed voltages in each strain gauge, when force is applied along different directions. In other words, variance of the observed voltages obtained from 3 strain gauges are significantly different given applied forces along each axis. Figure 4.2, shows the box plots of the voltages in S_1, S_2, and S_3 when force is in x, y, and z directions, respectively. Plot on the left panel of Figure 4.2, shows that variance of S_1 is between 0 V and 0.35 V, and as it is shown in the middle panel, when force is applied along y direction, variability of voltages in S_3 changes from 0 V to 0.8 V. However, as can be seen in the last panel, variance of S_3 is remarkably lower than other strain gauges, that is between 0 V and 0.08 V.

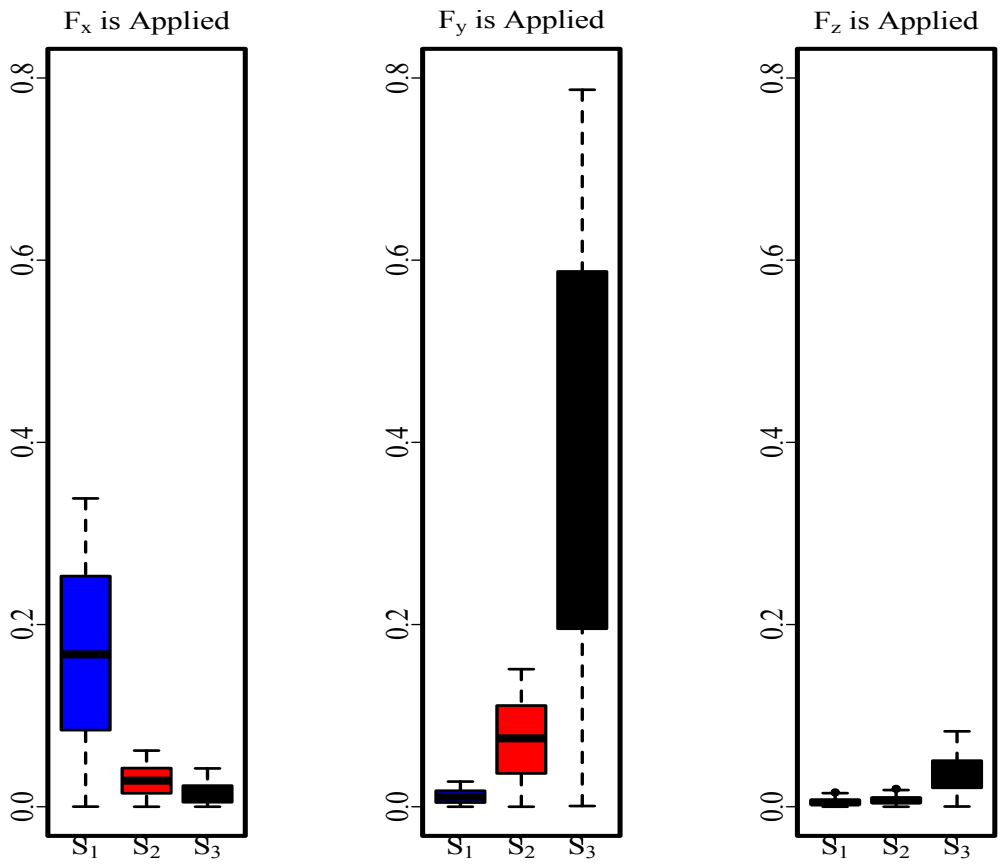


Figure 4.2: Box plots of observed voltages in strain gauges 1,2, and 3 (S_1, S_2, and, S_3), when force is applied in 3 directions. Different strain gauges are specified with different colors.

So far, two main issues that directly affect force prediction in 3 directions are explained. These problems are due to installing the third pair of strain gauges and they make the force prediction along 3 directions rather difficult. Another difficulty of force estimation considering 3 pairs of strain gauges is shown in the following section where bootstrap method that was proposed in Chapter 3, is employed to obtain interval estimation for applied forces along x, y, and z axes.

4.1.2 Bootstrap Calibration for the 3-D Case

In this section, we have employed the bootstrap method (proposed in Chapter 3) for the calibration data set that includes the amount of forces between 0.1 N and 2 N, in 3 directions and also observed voltages in S_1, S_2, and S_3. Through out this chapter, as true force values are between 0 N and 2 N, confidence intervals are truncated when the lower bound is negative or the upper bound is bigger than 2 N.

The procedure of the bootstrap method is similar to the one explained in Section 3.3, except that we have also considered observations obtained from S_3. Consequently, residuals are obtained from 3 different regression models that are listed below:

$$
\begin{aligned}
S_1 &= \begin{pmatrix} \alpha_1 & \beta_1 & \gamma_1 \end{pmatrix} \begin{pmatrix} F_x & 0 & 0 \\ 0 & F_y & 0 \\ 0 & 0 & F_z \end{pmatrix} + \epsilon_1, \\
S_2 &= \begin{pmatrix} \alpha_2 & \beta_2 & \gamma_2 \end{pmatrix} \begin{pmatrix} F_x & 0 & 0 \\ 0 & F_y & 0 \\ 0 & 0 & F_z \end{pmatrix} + \epsilon_2, \\
S_3 &= \begin{pmatrix} \alpha_3 & \beta_3 & \gamma_3 \end{pmatrix} \begin{pmatrix} F_x & 0 & 0 \\ 0 & F_y & 0 \\ 0 & 0 & F_z \end{pmatrix} + \epsilon_3,
\end{aligned}
\tag{4.1}
$$

where, $S_1 = (V_{x1}, V_{y1}, V_{z1})^\top$, $S_2 = (V_{x2}, V_{y2}, V_{z2})^\top$, and $S_3 = (V_{x3}, V_{y3}, V_{z3})^\top$ are observed voltages obtained from each strain gauge, where V_{xi}, V_{yi}, and V_{zi} are observed voltages from strain gauge i, $i = 1, 2, 3$, when force is applied in x, y, and z directions, respectively.

Suppose we observe $(V_{0J1}, V_{0J2}, V_{0J3})$ voltages from 3 strain gauges, and our goal is to predict the amount of applied forces in x, y, and z direction. First, we need

to fit models in (4.1), and then we will proceed to the following steps to obtain bootstrap predictions:

1. Obtaining $\hat{\alpha}_1, \hat{\beta}_1, \hat{\gamma}_1$, $\hat{\alpha}_2, \hat{\beta}_2, \hat{\gamma}_2$, and $\hat{\alpha}_3, \hat{\beta}_3, \hat{\gamma}_3$ from (4.1) using the *Training* data set.

2. Calculating the residuals using,

$$\epsilon_{1i} = S_{1i} - \begin{pmatrix} \hat{\alpha}_1 & \hat{\beta}_1 & \hat{\gamma}_1 \end{pmatrix} \begin{pmatrix} F_{xi} & 0 & 0 \\ 0 & F_{yi} & 0 \\ 0 & 0 & F_{zi} \end{pmatrix}, \tag{4.2}$$

$$\epsilon_{2i} = S_{2i} - \begin{pmatrix} \hat{\alpha}_2 & \hat{\beta}_2 & \hat{\gamma}_2 \end{pmatrix} \begin{pmatrix} F_{xi} & 0 & 0 \\ 0 & F_{yi} & 0 \\ 0 & 0 & F_{zi} \end{pmatrix}, \tag{4.3}$$

$$\epsilon_{3i} = S_{3i} - \begin{pmatrix} \hat{\alpha}_3 & \hat{\beta}_3 & \hat{\gamma}_3 \end{pmatrix} \begin{pmatrix} F_{xi} & 0 & 0 \\ 0 & F_{yi} & 0 \\ 0 & 0 & F_{zi} \end{pmatrix}, \tag{4.4}$$

   where, $i = 1, 2, \ldots, n$.

3. Forming the *bootstrap* data set by constructing the residual pool $\{\epsilon_{1i}, \epsilon_{2i}, \epsilon_{3i} \quad i = 1, 2, \cdots, n\}$ and obtain the following by resampling it.

$$
\textit{Training}\begin{cases}
S^*_{1i} = \begin{pmatrix}\hat{\alpha}_1 & \hat{\beta}_1 & \hat{\gamma}_1\end{pmatrix}\begin{pmatrix}F_{xi} & 0 & 0\\ 0 & F_{yi} & 0\\ 0 & 0 & F_{zi}\end{pmatrix} + \epsilon^*_{1i},\\
S^*_{2i} = \begin{pmatrix}\hat{\alpha}_2 & \hat{\beta}_2 & \hat{\gamma}_2\end{pmatrix}\begin{pmatrix}F_{xi} & 0 & 0\\ 0 & F_{yi} & 0\\ 0 & 0 & F_{zi}\end{pmatrix} + \epsilon^*_{2i},\\
S^*_{3i} = \begin{pmatrix}\hat{\alpha}_3 & \hat{\beta}_3 & \hat{\gamma}_3\end{pmatrix}\begin{pmatrix}F_{xi} & 0 & 0\\ 0 & F_{yi} & 0\\ 0 & 0 & F_{zi}\end{pmatrix} + \epsilon^*_{3i},
\end{cases} \tag{4.5}
$$

and,

$$
\textit{Unknowns}\begin{cases}
V^*_{0j1} = V_{0j1} + \epsilon^*_{01},\\
V^*_{0j2} = V_{0j2} + \epsilon^*_{02},\\
V^*_{0j3} = V_{0j3} + \epsilon^*_{03},
\end{cases} \tag{4.6}
$$

where, ϵ^*_{1i}, ϵ^*_{2i}, ϵ^*_{3i}, ϵ^*_{01}, ϵ^*_{02} and ϵ^*_{03} are random samples with replacement from the *residual pool*. Note that, we obtain separate residual pools from each model of (4.1).

4. Fitting new models (4.5) to bootstrap data sets and obtaining corresponding values of $\hat{\alpha}^*_1$, $\hat{\beta}^*_1$,$\hat{\gamma}^*_1$, $\hat{\alpha}^*_2$,$\hat{\beta}^*_2$, $\hat{\gamma}^*_2$, and $\hat{\alpha}^*_3$,$\hat{\beta}^*_3$, $\hat{\gamma}^*_3$.

5. Calculating $\hat{F}_x$, $\hat{F}_y$, and $\hat{F}_z$ using,

$$\begin{cases} V^*_{0J1} = \begin{pmatrix} \hat{\alpha}^*_1 & \hat{\beta}^*_1 & \hat{\gamma}^*_1 \end{pmatrix} \begin{pmatrix} \hat{F}_{xi} & 0 & 0 \\ 0 & \hat{F}_{yi} & 0 \\ 0 & 0 & \hat{F}_{zi} \end{pmatrix}, \\ V^*_{0J2} = \begin{pmatrix} \hat{\alpha}^*_2 & \hat{\beta}^*_2 & \hat{\gamma}^*_2 \end{pmatrix} \begin{pmatrix} \hat{F}_{xi} & 0 & 0 \\ 0 & \hat{F}_{yi} & 0 \\ 0 & 0 & \hat{F}_{zi} \end{pmatrix}, \\ V^*_{0J3} = \begin{pmatrix} \hat{\alpha}^*_3 & \hat{\beta}^*_3 & \hat{\gamma}^*_3 \end{pmatrix} \begin{pmatrix} \hat{F}_{xi} & 0 & 0 \\ 0 & \hat{F}_{yi} & 0 \\ 0 & 0 & \hat{F}_{zi} \end{pmatrix}. \end{cases} \tag{4.7}$$

6. Repeat steps 3 to 5 B times.

7. Quantify confidence intervals for estimated forces.

In the following table, estimated bootstrap confidence bounds in each direction for the amount of forces between 0.1 N and 2 N are presented.

Table 4.1: Estimated bootstrap confidence intervals for different amounts of forces applied to right forceps tip, in x, y, and z directions. To examine results of the bootstrap method, amounts of true forces were considered as unknown values.

Estimated Forces by Bootstrap Approach			
True Force	F_x C.I	F_y C.I	F_z C.I
0.1	(0.000 , 0.920)	(0.154 , 2.000)	(0.000 , 2.000)
0.2	(0.000 , 0.893)	(0.000 , 2.000)	(0.000 , 2.000)
0.3	(0.008 , 2.000)	(0.000 , 2.000)	(0.000 , 2.000)
0.4	(0.275 , 2.000)	(0.000 , 2.000)	(0.000 , 2.000)
0.5	(0.257 , 2.000)	(0.000 , 2.000)	(0.000 , 2.000)
0.6	(0.447 , 2.000)	(0.777 , 2.000)	(0.000 , 2.000)
0.7	(0.000 , 1.301)	(0.000 , 2.000)	(0.000 , 2.000)
0.8	(0.994 , 2.000)	(0.851 , 2.000)	(0.000 , 2.000)
0.9	(1.000 , 2.000)	(0.991 , 2.000)	(0.000 , 2.000)
1.0	(0.000 , 1.105)	(1.370 , 2.000)	(0.000 , 2.000)
1.1	(0.184 , 1.762)	(0.000 , 2.000)	(0.000 , 2.000)
1.2	(0.314 , 1.691)	(0.000 , 2.000)	(0.000 , 2.000)
1.3	(0.687 , 2.000)	(0.000 , 2.000)	(0.000 , 2.000)
1.4	(0.890 , 2.000)	(0.442 , 2.000)	(0.000 , 2.000)
1.5	(0.000 , 1.375)	(0.000 , 2.000)	(0.000 , 2.000)
1.6	(0.728 , 2.000)	(0.000 , 2.000)	(0.000 , 2.000)
1.7	(1.026 , 2.000)	(0.325 , 2.000)	(0.000 , 2.000)
1.8	(1.263 , 2.000)	(0.974 , 2.000)	(0.000 , 2.000)
1.9	(1.521 , 2.000)	(1.824 , 2.000)	(0.000 , 2.000)
2.0	(1.269 , 2.000)	(1.324 , 2.000)	(0.000 , 2.000)

As it can be seen from Table 4.1, not only the lengths of intervals are too wide, but also the estimated force are very different from the true force values. Although constructed bootstrap confidence intervals are very wide, still in certain situations

the true value of the force is not within the interval. As an example, when the true amount of applied force is 1.5 N, the estimated bootstrap confidence intervals in x, y, and z directions are (0.000 N, 1.375 N), (0 N, 2.000 N), and (0.000 N, 2.000 N), respectively, where 1.5 N is not within the interval constructed for force in x direction.

In the left panel of Fig 4.3, observed voltages from each strain gauge are plotted, and the direction of the applied force is specified with different colours. 3 different residuals (4.2), (4.3), and (4.4) obtained from different models in (4.1), are also illustrated in the right panel of Figure 4.3. One can see that in the right panel of Figure 4.3, voltages in S_3 are not close to zero when force is applied in the y direction. Nevertheless, we expected significant amount of voltages to be observed when force is applied in the z direction. Therefore, the residuals are also significant in S_3 only when force is excerted in the y direction.

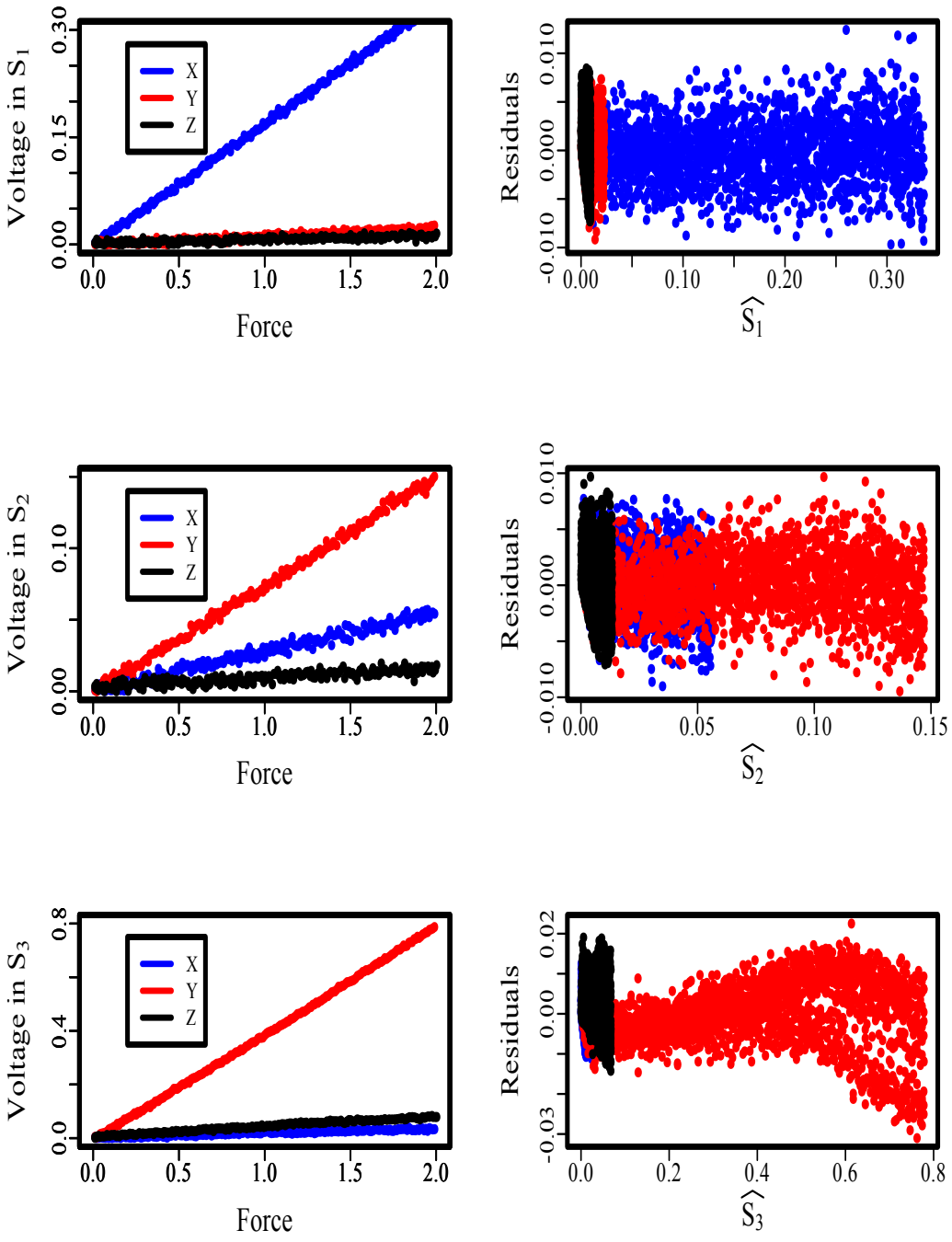


Figure 4.3: In the left panel, observed voltages from each strain gauge is presented versus the amount of force in 3 directions. In the right panel, residuals obtained from fitting models in each strain gauge is illustrated versus the fitted values of voltages for all 3 directions. Different directions are specified with different colours.

In what follows, several methods are proposed in order to address the issue with estimating the force in z direction. For each proposed method results are provided for the comparison purposes.

4.2 Weighted Least Squares Regression

According to Draper and Smith (2014), when the variances of all observations on different variables are not equal, or when observations are highly correlated, one can apply the Weighted Least Squares (WLS) approach to estimate the regression coefficients instead of the commonly used Ordinary Least Squares (OLS) method. Considering Figure 4.3, it can be seen that the variability of the residuals obtained from fitting regression models using an OLS approach is changing within the third strain gauge (S_3). This suggests using a WLS approach by considering suitable *weights* in the regression models (4.1) in order to obtain more suitable regression models to be used in our proposed bootstrap method. To this end, we use the inverse of the variances of observations in each strain gauge to construct the necessary weights in the WLS approach to run the bootstrap calibration procedure.

To be more specific, for each regression model (4.1), 3 different weights are considered:

$$\begin{aligned} W_1 &= (Var^{-1}(V_{1x}), Var^{-1}(V_{1y}), Var^{-1}(V_{1z})), \\ W_2 &= (Var^{-1}(V_{2x}), Var^{-1}(V_{2y}), Var^{-1}(V_{2z})), \\ W_3 &= (Var^{-1}(V_{3x}), Var^{-1}(V_{3y}), Var^{-1}(V_{3z})), \end{aligned} \tag{4.8}$$

where W_1, W_2, and W_3 are the associated weights for S_1, S_2, and S_3 in models (4.1), respectively. After implementing the necessary expressions for WLS estimates

of regression model parameters (*e.g.*, Draper and Smith (2014)) the bootstrap calibration will be implemented as before.

4.2.1 Results Obtained from Bootstrapping with WLS

The results that are obtained through employing WLS with weights (4.8), are peresented in Table 4.2. As we observe, estimated confidence intervals are significantly improved compared to the results in Table. 4.1. For most cases, the true amount of forces in the x and y directions are within the estimated confidence intervals. However, estimated confidence intervals for forces in z direction are not accurate enough, and the length of the intervals are considerably wide. For example, when the true force is 0.7 N (in x, y or z directions), estimated intervals for forces in x and y directions are (0.697 N , 0.742 N) and (0.657 N , 0.750 N), respectively, while the obtained interval in z direction is (0.650 N , 1.129 N). As we observed, although WLS results in more accurate estimates of true forces compared with the OLS approach, still results for the z direction are not as precise as they need to be in order to be used in practice. Part of the difficulty related with estimation in the z direction is associated with the technical problem with improper implementation of the censors in the device. In the next section we proceed the underlying estimation process using the Bayesian methodology hoping that this will results in more accurate estimation of the true force values in the z direction.

Table 4.2: Estimated confidence intervals for different amounts of forces applied to right forceps tip, in x, y, and z directions, using WLS approach. To examine results of the bootstrap method, amounts of true forces were considered as unknown values.

Estimated Forces by WLS Approach			
True Force	F_x C.I	F_y C.I	F_z C.I
0.1	(0.095 , 0.138)	(0.094 , 0.190)	(0.228 , 0.715)
0.2	(0.156 , 0.199)	(0.169 , 0.264)	(0.290 , 0.766)
0.3	(0.299 , 0.344)	(0.272 , 0.368)	(0.379 , 0.860)
0.4	(0.394 , 0.438)	(0.347 , 0.442)	(0.378 , 0.867)
0.5	(0.498 , 0.541)	(0.471 , 0.566)	(0.542 , 1.014)
0.6	(0.580 , 0.625)	(0.589 , 0.682)	(0.657 , 1.143)
0.7	(0.697 , 0.742)	(0.657 , 0.750)	(0.650 , 1.129)
0.8	(0.760 , 0.805)	(0.791 , 0.886)	(1.050 , 1.530)
0.9	(0.881 , 0.925)	(0.887 , 0.981)	(1.060 , 1.553)
1.0	(0.975 , 1.020)	(0.998 , 1.090)	(1.153 , 1.622)
1.1	(1.087 , 1.130)	(1.121 , 1.218)	(1.432 , 1.913)
1.2	(1.199 , 1.243)	(1.216 , 1.310)	(1.636 , 2.000)
1.3	(1.275 , 1.319)	(1.298 , 1.394)	(1.526 , 1.994)
1.4	(1.390 , 1.436)	(1.389 , 1.483)	(1.739 , 2.000)
1.5	(1.461 , 1.505)	(1.510 , 1.605)	(1.716 , 2.000)
1.6	(1.586 , 1.630)	(1.566 , 1.659)	(1.859 , 2.000)
1.7	(1.688 , 1.733)	(1.691 , 1.785)	(1.947 , 2.000)
1.8	(1.800 , 1.846)	(1.801 , 1.897)	(2.000 , 2.000)
1.9	(1.902 , 1.947)	(1.918 , 2.000)	(2.000 , 2.000)
2.0	(1.976 , 2.000)	(2.000 , 2.000)	(2.000 , 2.000)

4.3 Multi-level Modeling

Multi-level models are generalizations of the regression models in which a set of data is structured in different categories based on their characteristics and the model coefficients can vary within each different category. There are three commonly used formulations in multi-level modeling. These formulations are briefly explained as follow:

1. Models with varying slope, $Y_i = \alpha + \beta_{j[i]} X_i + \epsilon_i$.

2. Models with varying intercept, $Y_i = \alpha_{j[i]} + \beta X_i + \epsilon_i$.

3. Models with varying slope and intercept, $Y_i = \alpha_{j[i]} + \beta_{j[i]} X_i + \epsilon_i$.

Here, $j = 1, 2, .., J$ indicates the group factors with J levels, and $i = 1, 2, ..., n$ represents the data points. Figure. 4.4, illustrates these models for the case of a simple linear regression model with three group levels. The left, middle, and right plots represent model 1, 2, and 3, respectively.

Multi-level models appear with different names in the literature. For example, in Ruppert et al. (2003), these models are referred to as *mixed models*. Mixed models or mixed-effects models are comprised of fixed, as well as random coefficients, where multi-level models include these two types of coefficients as well. In this regard, random effects are defined as varying coefficients (α_j's and β_j's), while the fixed effects are fixed for different group levels in these models.

As in Gelman and Hill (2006), multi-level modeling could be defined as an adjustment between completely excluding the effects of different group levels (com-

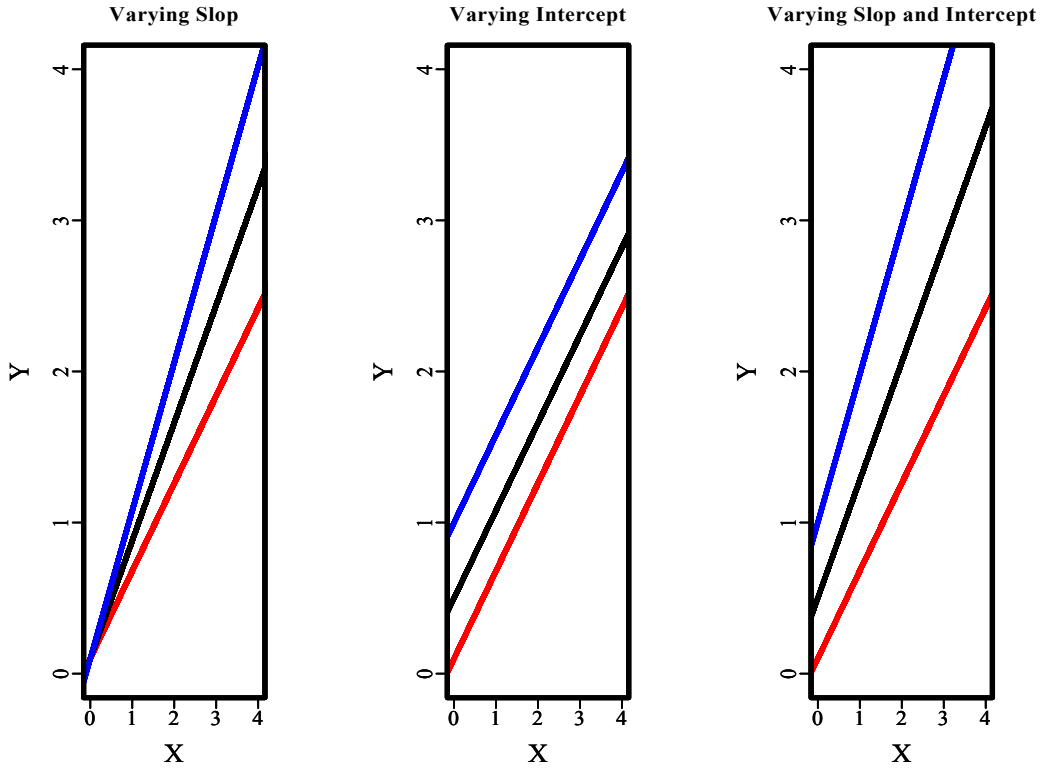


Figure 4.4: Three possible models in multi-level modeling are illustrated. First plot from the left, presents a model with varying slop, middle plot is showing the model with varying intercept, while the last plot illustrates the case where both slope and intercepts are varying for different groups.

plete pooling) and considering separate models for different groups (no pooling). Therefore, multi-level models are referred to as *partial-pooling* estimation models, since they consider the average of the available observations at each group level to estimate the coefficients. While *no-pooling* estimation results in overfitting the data, *complete-pooling* ignores the variation between different groups. Such undesirable effects can be avoided (or minimized) using multi-level modeling.

Multi-level modeling approach is applicable to our calibration problem as we

are dealing with categorical data set. As mentioned earlier, explanatory variable (force) in this problem includes 3 different directions (x, y, z), that is considered as a group factor. Having calibration data set, varying slope without intercept models are fitted to obtain relative coefficients:

$$S_{kij} = F_i\beta_k + Z_i u_{jk} + \epsilon_{kij}, \tag{4.9}$$

where, $k = 1, 2, 3$ is the strain gauge number, $i = 1, 2, ..., 3n$ is calibration data point number, $F = (F_{1x}, \cdots, F_{nx}, F_{1y}, \cdots, F_{ny}, F_{1z}, \cdots, F_{nz})^\top$,

$$j = \begin{cases} 1, & x \text{ direction,} \\ 2, & y \text{ direction,} \\ 3, & z \text{ direction,} \end{cases} \quad \text{and,} \quad Z = \begin{pmatrix} F_{1x} & \cdots & F_{nx} & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & F_{1y} & \cdots & F_{ny} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 & F_{1z} & \cdots & F_{nz} \end{pmatrix}^\top$$

Furthermore, $E\begin{pmatrix} u_k \\ \epsilon_k \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$, and $Cor\begin{pmatrix} u_k \\ \epsilon_k \end{pmatrix} = \begin{pmatrix} G_k & 0 \\ 0 & R_k \end{pmatrix}$, where $G_k = \sigma^2_{u_k}$, $R_k = \sigma^2_{\epsilon_k}$. Therefore,

$$Var(S_k) = ZG_kZ^\top + R_k = \Sigma_k. \tag{4.10}$$

In model (4.9) β_k's are called *fixed effects* and u_k's that are varying for different j's (direction) are called *random effects*.

4.3.1 Estimation of Fixed and Random Effects Coefficients

Suppose Σ_k (4.10) is known, fixed effects coefficients are estimated through either the ML, or Generalized Least Squares (GLS) approach. Under the multivariate normality assumption, the log-likelihood function under the model (4.2) is,

$$\ell(\beta_k) = -\frac{1}{2}\{n\log(2\pi) + \log|\Sigma_k| + (S_k - F\beta_k)^\top\Sigma_k^{-1}(S_k - F\beta_k)\}. \tag{4.11}$$

Consequently, $\hat{\beta}_k = (F^\top \Sigma_k^{-1} F)^{-1} F^\top \Sigma_k^{-1} S_k$, that is the Best Linear Unbiased Prediction (BLUP).

To estimate random effects, Best Linear Unbiased Prediction (BLUP) approach under Squared Error Loss (SEL) is employed. Let $\delta_k(S_k)$ be an estimation for u_k. Then,

$$\delta_{BLUP} = \underset{\delta_k}{\operatorname{argmin}} E\left[(u_k - \delta_k(S_k))^2 \mid S_k\right]. \tag{4.12}$$

In order to obtain δ_{BLUP}, let

$$f(\delta_k) = E[u_k^2 \mid S_k] - 2\delta_k(S_k) E[u_k \mid S_k] + \delta_k^2(S_k). \tag{4.13}$$

One can obtain δ_{BLUP}, by taking the derivative of $f(\delta_k)$, as follow:

$$\begin{aligned} \frac{d}{d\delta_k} f(\delta_k) &= -2E[u_k \mid S_k] + 2\delta_k(S_k) = 0 \\ &\Rightarrow \delta_k = E[u_k \mid S_k]. \end{aligned} \tag{4.14}$$

Therefore, the BLUP estimation of u_k under SEL for a given β_k is obtained, assuming that S_k and u_k are normally distributed, as follow:

$$\begin{aligned} \hat{u}_k &= E(u_k | S_k) \\ &= E(u_k) + E[(u_k - E(u_k))(S_k - E(S_k))^\top] Var^{-1}(S_k)(S_k - E(S_k)) \\ &= E(u_k S_k^\top) \Sigma_k^{-1} (S_k - F\beta_k) \\ &= Z_k^\top E(u_k^2) \Sigma_k^{-1} (S_k - F\beta_k) \\ &= Z_k^\top G_k \Sigma_k^{-1} (S_k - F\beta_k), \end{aligned} \tag{4.15}$$

However, in practice, β_k is replaced by its estimation, that is $\hat{\beta}_k$, and G_k and R_k have to be estimated.

In what follows we extend our results under SEL to LINEX loss function. The LINEX loss is an asymmetric function, that enables us to use different penalties for over and under estimation of the parameter of interest. In our calibration problem, underestimating the amount of forces is more serious than overestimating as excessive force on brain tissues can cause serious damages and hence it is recommended to protect against under estimation. To address this problem, LINEX loss function has been also applied in our prediction process, to obtain more reliable results. LINEX loss function is defined as

$$L_\alpha(\hat{u}_k, u_k) = e^{\alpha(\hat{u}_k - u_k)} - \alpha(\hat{u}_k - u_k) - 1.$$

In order to use LINEX loss function, we first need an extension of the notation of unbiasedness to take into account the role of the loss function in the estimation (prediction) process. To this end, we use the following theoretic definition of unbiasedness.

Definition: A predictor $\hat{u}_k$ of u_k is said to be loss-unbiased if it satisfies

$$E[L(\hat{u}_k, u_k)] \leq E[L(\nu, u_k)], \quad \text{for any} \quad \nu \neq \hat{u}_k. \tag{4.16}$$

In particular, when $L(\hat{u}_k, u_k) = (\hat{u}_k - u_k)^2$, then the notation of unbiasedness reduces to $E(\hat{u}_k - u_k) = 0$.

Under the Linex loss $L_\alpha(\hat{u}_k, u_k)$, it is easy to see that

$$E[L_\alpha(\hat{u}_k, u_k)] \leq E[L_\alpha(\nu, u_k)], \quad \forall \nu \neq \hat{u}_k.$$

Now, the problem reduces to finding the Best Linex Unbiased Predictor (BLinexUP). To obtain such a predictor, we need to minimize $E[L_\alpha(\hat{u}_k, u_k)]$, with respect to $\hat{u}_k$

by minimizing the left-side of the following equation with respect to $\hat{u}_k$:

$$E[L_\alpha(\hat{u}_k, u_k)] = E\big[E[L_\alpha(\hat{u}_k, u_k)] \mid S_k\big],$$

that is equivalent to minimizing $E[L_\alpha(\hat{u}_k, u_k) \mid S_k]$, with respect to $\hat{u}_k$. Under the LINEX loss function, let

$$Q(\hat{u}_k) = E[e^{\alpha(\hat{u}_k - u_k)} - \alpha(\hat{u}_k - u_k) - 1 \mid S_k], \tag{4.17}$$

taking the derivative of $Q(\hat{u}_k)$ with respect to $\hat{u}_k$, one can obtain BLinexUP as follow:

$$\frac{d}{d\hat{u}_k} Q(\hat{u}_k) = E[\alpha e^{\alpha(\hat{u}_k - u_k)} - \alpha] = 0,$$

which results in

$$\hat{u}_{\alpha,k} = \frac{1}{\alpha} \log E[e^{-\alpha u_k} \mid S_k]. \tag{4.18}$$

Assuming $u_k \mid S_k \sim N(\mu_*, \sigma_*^2)$, where $\mu_* = E(u_k \mid S_k) = \hat{u}_k$, then,

$$\hat{u}_{\alpha,k} = \mu_* - \frac{\sigma_*^2 \alpha}{2} = \hat{u}_k - \frac{\sigma_*^2 \alpha}{2}. \tag{4.19}$$

4.3.2 Estimation of Covariance Matrices, G_k and R_k

In Ruppert et al. (2003), the ML approach and Restricted (or Residual) ML (REML) strategies are proposed to estimate G_k and R_k. We first establish ML approach by writing the log-likelihood function as follow:

$$\ell(\beta_k, \Sigma_k) = -\frac{1}{2}\{n \log(2\pi) + \log|\Sigma_k| + (S_k - F\beta_k)^\top \Sigma_k^{-1} (S_k - F\beta_k)\}. \tag{4.20}$$

We then substitute $\hat{\beta}_k$ in (4.20) to get

$$\ell_p(\Sigma_k) = -\frac{1}{2}\{\log|\Sigma_k| + S_k^\top \Sigma_k^{-1}(I - F(F^\top \Sigma_k^{-1} F)^{-1} F^\top \Sigma_k^{-1}) S_k\} - \frac{n}{2}\log(2\pi), \quad (4.21)$$

where, ℓ_p stands for *profile log-likelihood*, and Σ_k is defined in (4.10). At this point, we can obtain ML estimates of G_k and R_k in Σ_k, by maximizing (4.21). Since, Σ_k is the function of (R_k, G_k), estimation of them involves maximizing $\ell_p(G_k Z Z^\top + R_k)$

The likelihood function to obtain REML estimates for G_k and R_k is

$$\ell_{REML}(\Sigma_k) = \ell_p(\Sigma_k) - \frac{1}{2}\log|F^\top \Sigma_k^{-1} F|. \quad (4.22)$$

According to Ruppert et al. (2003), the REML is a more accurate method to obtain the estimates than ML method, specially when we are dealing with small sample sizes. Nevertheless, when the sample size is large enough, ML and REML estimates are very close.

The reason that REML is preferable over ML is the fact that the influence of fixed effects's degrees of freedom has not been considered in the ML approach (Searle et al., 2009). As it is mentioned in Ruppert et al. (2003), REML estimation is more complicated since it requires the estimates by maximizing the linear combination of components of likelihood function that are independent of β_k. More details on REML approach is given in Searle et al. (2009).

4.3.3 Results Obtained from Multi-level Modeling

In this part, multi-level modeling approach is employed to obtain point and interval estimates of applied forces in x, y, and z directions. Results are obtained using **R**

built-in function *'lmer'*, within *'lme4'* package (De Boeck et al., 2011).

In theory, we should not consider intercept in the model, since voltages are not observed when applied force is 0 N. However, we violated the model and considered intercepts as bias in the model to get better predictions. Also, we have provided the results under both SEL and LINEX loss functions. For obtaining the results under LINEX loss, we use $\alpha \in (0.1, 0.5, 1)$ to provide different measures of penalty for overprediction compared with underprediction. Tables 4.3 and 4.4 present the results obtained under SEL and LINEX loss, respectively.

Table 4.3: Point estimates obtained by multi-level modeling for different amounts of forces applied to right forceps tip, in x, y, and z directions under SEL loss function. As before, to examine results, amounts of true forces were considered as unknown values.

	Models Without Intercept			**Models With Intercept**		
True Force	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$
0.1	0.086	0.096	0.216	0.094	0.090	0.237
0.2	0.169	0.188	0.303	0.176	0.183	0.322
0.3	0.302	0.297	0.515	0.308	0.292	0.530
0.4	0.405	0.448	0.525	0.411	0.444	0.540
0.5	0.495	0.477	0.612	0.501	0.473	0.625
0.6	0.573	0.587	0.698	0.577	0.583	0.710
0.7	0.652	0.763	0.833	0.656	0.760	0.843
0.8	0.798	0.733	0.853	0.801	0.731	0.862
0.9	0.899	0.863	0.882	0.902	0.861	0.890
1.0	0.980	0.968	1.094	0.982	0.967	1.099
1.1	1.079	1.077	1.239	1.081	1.076	1.241
1.2	1.221	1.224	1.267	1.222	1.224	1.269
1.3	1.308	1.308	1.354	1.308	1.308	1.355
1.4	1.418	1.396	1.412	1.418	1.396	1.411
1.5	1.514	1.560	1.422	1.513	1.561	1.421
1.6	1.582	1.648	1.557	1.581	1.649	1.553
1.7	1.730	1.698	1.615	1.727	1.700	1.610
1.8	1.827	1.849	1.702	1.825	1.852	1.696
1.9	1.910	1.959	1.875	1.907	1.961	1.866
2.0	2.022	2.055	1.943	2.019	2.058	1.932

Table 4.4: Estimated forces in x, y, and z directions under LINEX loss function, considering three different values for α. Top part of the table presents the obtained force estimates for models without considering intercept, and the bottom part presents the estimation for models with intercept.

	Estimated Forces in Models without Intercept								
	$\alpha_1 = 0.1$			$\alpha_2 = 0.5$			$\alpha_3 = 1$		
True Force	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$
0.1	0.086	0.096	0.221	0.087	0.096	0.244	0.087	0.096	0.281
0.2	0.170	0.188	0.310	0.170	0.188	0.342	0.171	0.188	0.394
0.3	0.302	0.297	0.527	0.303	0.297	0.582	0.305	0.298	0.670
0.4	0.406	0.448	0.537	0.407	0.449	0.593	0.409	0.449	0.683
0.5	0.496	0.478	0.626	0.498	0.478	0.692	0.500	0.479	0.796
0.6	0.573	0.587	0.715	0.575	0.587	0.790	0.578	0.588	0.909
0.7	0.653	0.763	0.853	0.655	0.764	0.942	0.658	0.765	1.084
0.8	0.798	0.734	0.873	0.801	0.734	0.964	0.804	0.735	1.109
0.9	0.900	0.864	0.903	0.903	0.865	0.997	0.906	0.866	1.147
1.0	0.981	0.969	1.120	0.984	0.970	1.237	0.988	0.971	1.423
1.1	1.080	1.078	1.268	1.084	1.079	1.401	1.088	1.080	1.612
1.2	1.222	1.225	1.298	1.226	1.226	1.433	1.231	1.228	1.649
1.3	1.309	1.309	1.386	1.313	1.310	1.532	1.319	1.312	1.762
1.4	1.419	1.397	1.446	1.424	1.398	1.597	1.430	1.400	1.838
1.5	1.515	1.560	1.456	1.520	1.562	1.608	1.527	1.564	1.850
1.6	1.584	1.649	1.594	1.589	1.650	1.761	1.596	1.652	2.026
1.7	1.731	1.699	1.653	1.737	1.701	1.826	1.744	1.703	2.101
1.8	1.829	1.850	1.742	1.835	1.852	1.924	1.843	1.854	2.214
1.9	1.912	1.959	1.920	1.918	1.961	2.121	1.926	1.964	2.440
2.0	2.024	2.056	1.989	2.031	2.058	2.197	2.039	2.061	2.528
	Estimated Forces in Models with Intercept								
	$\alpha_1 = 0.1$			$\alpha_2 = 0.5$			$\alpha_3 = 1$		
True Force	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$
0.1	0.094	0.090	0.242	0.094	0.090	0.267	0.095	0.090	0.306
0.2	0.177	0.183	0.329	0.177	0.183	0.363	0.178	0.183	0.417
0.3	0.309	0.292	0.543	0.310	0.293	0.598	0.311	0.293	0.686
0.4	0.411	0.444	0.552	0.413	0.445	0.609	0.414	0.445	0.699
0.5	0.501	0.474	0.640	0.503	0.474	0.705	0.505	0.475	0.809
0.6	0.578	0.583	0.727	0.580	0.584	0.801	0.582	0.585	0.919
0.7	0.657	0.760	0.863	0.659	0.761	0.951	0.662	0.762	1.091
0.8	0.802	0.731	0.882	0.804	0.732	0.972	0.808	0.732	1.115
0.9	0.902	0.861	0.911	0.905	0.862	1.004	0.909	0.863	1.152
1.0	0.983	0.967	1.124	0.986	0.968	1.240	0.990	0.969	1.422
1.1	1.082	1.076	1.270	1.086	1.078	1.400	1.090	1.079	1.606
1.2	1.223	1.224	1.299	1.227	1.225	1.432	1.232	1.227	1.643
1.3	1.309	1.308	1.386	1.314	1.310	1.528	1.319	1.311	1.753
1.40	1.419	1.397	1.444	1.424	1.398	1.592	1.430	1.400	1.827
1.5	1.514	1.561	1.454	1.520	1.563	1.603	1.526	1.565	1.839
1.6	1.582	1.650	1.590	1.588	1.652	1.753	1.594	1.654	2.011
1.7	1.729	1.700	1.648	1.735	1.702	1.817	1.742	1.704	2.084
1.8	1.826	1.852	1.735	1.832	1.854	1.913	1.840	1.857	2.194
1.9	1.908	1.962	1.910	1.915	1.964	2.105	1.923	1.966	2.415
2.0	2.020	2.059	1.977	2.027	2.061	2.180	2.035	2.064	2.501

As can be seen in Tables 4.3 and 4.4, force estimates in x and y directions are very close to the true amount of applied forces for models without intercept as well as models with intercept, but estimation in z directions is not accurate enough irrespective of the values of α. The problem with force estimation along the z axis is caused by adding the third pair of strain gauges as was discussed in Section 4.1. For instance, when the true force is 0.3 N, estimation of models without intercept in x and y directions under SEL and also LINEX loss are 0.302 N and 0.297 N, respectively. However, estimated force in z direction is 0.515 N and 0.527 N under SEL and LINEX loss for $\alpha = 0.1$, respectively, that is still far from the true amount of force.

To address the issues with force estimates along the z direction, we decided to apply weights in the multi-level modeling approach (WLS approach was explained in Section 4.2). Since the problem was only in estimating the forces along z direction, we applied weight only for $k = 3$ in model (4.9) (third strain gauge).

Considering the calibration data set, S_3 includes three different types of voltages, which are observed voltages when force is applied in x, y, and z directions. We therefore considered three different weights associated with each part of the S_3. Table 4.5 presents six different weighting schemes applied to the S_3 models. We applied all six weights presented in Table. 4.5, in our models to obtain force estimation along z axis. Table. 4.6, presents the results for models with and without intercept.

Table 4.5: Six different proposed weights to use in multi-level modeling approach.

	Weights					
S_3 **Components**	W_1	W_2	W_3	W_4	W_5	W_6
Type I [a]	1	1	1	v_{1y} [b]	v_{1y}	1
Type II [c]	1	1	1	v_{2y} [d]	v_{2y}	1
Type III [e]	$\frac{1}{\sigma^2(v_{3z})}$ [f]	$\frac{1}{\sigma^2(v_{2y})}$	$\frac{1}{(v_{2y})}$	v_{3y} [g]	v_{1x} [h]	$\frac{1}{\mu(v_{3z})}$

[a] Observed voltages when force is applied in x direction
[b] Observed voltages in S_1 when F_y is applied
[c] Observed voltages when force is applied in y direction
[d] Observed voltages in S_2 when F_y is applied
[e] Observed voltages when force is applied in z direction
[f] Observed voltages in S_3 when F_z is applied
[g] Observed voltages in S_3 when F_y is applied
[h] Observed voltages in S_1 when F_x is applied

As can be seen in Tables 4.6 and 4.7, models with intercept provide better force estimation compared to models without intercept. For instance, when the true F_z is 0.1 N, models without intercepts under SEL do not provide accurate estimation (all estimated the force as 0.22 N), while models with intercept provide accurate force prediction as 0.11 N, when W_3 is applied in the model. This is also true for results under LINEX loss, considering all three values of α. As it is presented in Table 4.7, when $\alpha = 1$, forces are overestimated, and the best results are obtained for $\alpha = 0.5$ considering W_1 in the model. Since σ_*^2 is very small, the differences between force predictions obtained under SEL and LINEX loss functions are not significant.

Table 4.6: Estimated forces in z direction, considering six different weights under SEL. Top part of the table presents the obtained force estimates for models without considering intercept, and the bottom part presents the estimation for models with intercept.

	Estimated F_z in Models without Intercept					
True F_z	W_1	W_2	W_3	W_4	W_5	W_6
0.1	0.22	0.22	0.21	0.22	0.22	0.22
0.2	0.30	0.30	0.29	0.31	0.31	0.30
0.3	0.52	0.52	0.50	0.52	0.52	0.52
0.4	0.52	0.52	0.51	0.53	0.53	0.52
0.5	0.61	0.61	0.59	0.62	0.62	0.61
0.6	0.70	0.70	0.67	0.71	0.71	0.70
0.7	0.83	0.83	0.80	0.85	0.85	0.83
0.8	0.85	0.85	0.82	0.87	0.87	0.85
0.9	0.88	0.88	0.85	0.89	0.89	0.88
1.0	1.09	1.09	1.05	1.11	1.11	1.09
1.1	1.24	1.24	1.19	1.26	1.26	1.24
1.2	1.27	1.27	1.22	1.29	1.29	1.27
1.3	1.35	1.35	1.30	1.37	1.37	1.35
1.4	1.41	1.41	1.36	1.43	1.43	1.41
1.5	1.42	1.42	1.37	1.44	1.44	1.42
1.6	1.56	1.56	1.50	1.58	1.58	1.56
1.7	1.61	1.61	1.55	1.64	1.64	1.61
1.8	1.70	1.70	1.64	1.73	1.73	1.70
1.9	1.88	1.88	1.81	1.90	1.90	1.88
2.0	1.94	1.94	1.87	1.97	1.97	1.94
	Estimated F_z in Models with Intercept					
True F_z	W_1	W_2	W_3	W_4	W_5	W_6
0.1	0.05	0.05	**0.11**	0.07	**0.09**	0.07
0.2	0.15	0.15	**0.21**	0.16	**0.19**	0.17
0.3	0.39	0.39	0.43	0.40	0.42	0.41
0.4	**0.41**	**0.41**	0.44	**0.42**	0.44	**0.42**
0.5	**0.50**	**0.51**	0.54	**0.51**	0.53	**0.52**
0.6	**0.60**	**0.60**	0.63	**0.61**	0.63	**0.62**
0.7	0.76	0.76	0.78	0.77	0.78	0.77
0.8	**0.78**	**0.78**	**0.80**	**0.79**	**0.80**	**0.79**
0.9	0.81	0.81	0.83	0.82	0.83	0.82
1.0	1.06	1.06	1.06	1.06	1.07	1.06
1.1	1.22	1.22	1.21	1.22	1.23	1.22
1.2	1.25	1.25	**1.24**	1.26	1.26	1.26
1.3	1.35	1.35	**1.34**	1.36	1.36	1.35
1.4	**1.42**	**1.42**	**1.40**	**1.42**	**1.42**	**1.42**
1.5	1.43	1.43	1.41	1.43	1.43	1.43
1.6	**1.58**	**1.58**	1.55	**1.58**	**1.58**	**1.58**
1.7	1.65	1.65	1.62	1.65	1.65	1.65
1.8	1.75	1.75	1.71	1.75	1.75	1.74
1.9	1.95	1.95	1.90	1.95	**1.94**	**1.94**
2.0	**2.00**	**2.00**	**1.97**	**2.00**	**2.00**	**2.00**

Table 4.7: Estimated forces in z direction, considering six different weights under LINEX loss. Top part of the table presents the obtained force estimates for models without considering intercept, and the bottom part presents the estimation for models with intercept.

Estimated F_z in Models without Intercept

	$\alpha_1 = 0.1$						$\alpha_2 = 0.5$						$\alpha_3 = 1$					
True F_z	W_1	W_2	W_3	W_4	W_5	W_6	W_1	W_2	W_3	W_4	W_5	W_6	W_1	W_2	W_3	W_4	W_5	W_6
0.1	0.22	0.24	0.28	0.22	0.24	0.28	0.21	0.23	0.27	0.22	0.25	0.29	0.22	0.25	0.29	0.22	0.24	0.28
0.2	0.31	0.34	0.39	0.31	0.34	0.39	0.30	0.33	0.37	0.31	0.35	0.40	0.31	0.35	0.40	0.31	0.34	0.39
0.3	0.53	0.58	0.67	0.53	0.58	0.67	0.51	0.56	0.64	0.54	0.59	0.68	0.54	0.59	0.68	0.53	0.58	0.67
0.4	0.54	0.59	0.68	0.54	0.59	0.68	0.52	0.57	0.65	0.55	0.60	0.70	0.55	0.60	0.70	0.54	0.59	0.68
0.5	0.63	0.69	0.80	0.63	0.69	0.80	0.60	0.66	0.76	0.64	0.70	0.81	0.64	0.70	0.81	0.63	0.69	0.80
0.6	0.71	0.79	0.91	0.71	0.79	0.91	0.69	0.76	0.86	0.73	0.80	0.93	0.73	0.80	0.93	0.71	0.79	0.91
0.7	0.85	0.94	1.08	0.85	0.94	1.08	0.82	0.90	1.03	0.87	0.96	1.11	0.87	0.96	1.11	0.85	0.94	1.08
0.8	0.87	0.96	1.11	0.87	0.96	1.11	0.84	0.92	1.06	0.89	0.98	1.13	0.89	0.98	1.13	0.87	0.96	1.11
0.9	0.90	1.00	1.15	0.90	1.00	1.15	0.87	0.96	1.09	0.92	1.01	1.17	0.92	1.01	1.17	0.90	1.00	1.15
1.0	1.12	1.24	1.42	1.12	1.24	1.42	1.08	1.18	1.35	1.14	1.26	1.45	1.14	1.26	1.45	1.12	1.24	1.42
1.1	1.27	1.40	1.61	1.27	1.40	1.61	1.22	1.34	1.53	1.29	1.42	1.64	1.29	1.42	1.64	1.27	1.40	1.61
1.2	1.30	1.43	1.65	1.30	1.43	1.65	1.25	1.37	1.57	1.32	1.46	1.68	1.32	1.46	1.68	1.30	1.43	1.65
1.3	1.39	1.53	1.76	1.39	1.53	1.76	1.33	1.47	1.68	1.41	1.56	1.80	1.41	1.56	1.80	1.39	1.53	1.76
1.4	1.45	1.60	1.84	1.45	1.60	1.84	1.39	1.53	1.75	1.47	1.62	1.87	1.47	1.62	1.87	1.45	1.60	1.84
1.5	1.46	1.61	1.85	1.46	1.61	1.85	1.40	1.54	1.76	1.48	1.63	1.89	1.48	1.64	1.89	1.46	1.61	1.85
1.6	1.59	1.76	2.03	1.59	1.76	2.03	1.53	1.69	1.93	1.62	1.79	2.07	1.62	1.79	2.07	1.59	1.76	2.03
1.7	1.65	1.83	2.10	1.65	1.83	2.10	1.59	1.75	2.00	1.68	1.86	2.14	1.68	1.86	2.14	1.65	1.83	2.10
1.8	1.74	1.92	2.21	1.74	1.92	2.21	1.68	1.84	2.11	1.77	1.96	2.26	1.77	1.96	2.26	1.74	1.92	2.21
1.9	1.92	2.12	2.44	1.92	2.12	2.44	1.85	2.03	2.32	1.95	2.16	2.49	1.95	2.16	2.49	1.92	2.12	2.44
2.0	1.99	2.20	2.53	1.99	2.20	2.53	1.91	2.10	2.41	2.02	2.23	2.58	2.02	2.24	2.58	1.99	2.20	2.53

Estimated F_z in Models with Intercept

	$\alpha_1 = 0.1$						$\alpha_2 = 0.5$						$\alpha_3 = 1$					
True F_z	W_1	W_2	W_3	W_4	W_5	W_6	W_1	W_2	W_3	W_4	W_5	W_6	W_1	W_2	W_3	W_4	W_5	W_6
0.1	0.05	**0.06**	**0.07**	**0.06**	**0.06**	**0.07**	**0.12**	**0.13**	0.15	**0.07**	**0.08**	**0.09**	**0.09**	**0.11**	**0.12**	**0.07**	**0.08**	**0.10**
0.2	**0.16**	**0.18**	**0.21**	**0.16**	**0.18**	**0.21**	**0.21**	**0.24**	0.27	**0.17**	**0.19**	**0.22**	**0.19**	**0.22**	**0.25**	**0.18**	**0.20**	**0.23**
0.3	0.40	0.45	0.54	0.41	0.46	0.54	0.45	0.50	0.58	0.42	0.47	0.55	0.44	0.49	0.57	0.42	0.47	0.55
0.4	**0.42**	0.47	0.55	**0.42**	0.47	0.55	0.46	0.51	0.59	**0.43**	0.48	0.56	**0.45**	0.50	0.59	**0.43**	0.48	0.57
0.5	**0.52**	0.58	0.69	**0.52**	0.58	0.69	**0.55**	0.61	0.72	**0.53**	0.59	0.70	**0.55**	0.61	0.72	**0.53**	0.60	0.70
0.6	**0.62**	0.70	0.82	**0.62**	0.70	0.82	**0.65**	0.72	0.84	**0.63**	0.71	0.83	**0.65**	0.72	0.85	**0.63**	0.71	0.83
0.7	0.78	0.87	1.03	0.78	0.87	1.03	0.80	0.89	1.03	0.79	0.88	1.04	0.80	0.89	1.05	0.79	0.88	1.04
0.8	**0.80**	0.90	1.06	**0.80**	0.90	1.06	**0.82**	0.91	1.06	**0.81**	0.91	1.07	**0.82**	0.92	1.08	**0.81**	0.91	1.07
0.9	0.83	**0.94**	1.10	0.84	**0.94**	1.11	**0.85**	**0.95**	1.10	0.84	**0.94**	1.11	**0.95**	0.96	1.12	0.84	**0.94**	1.11
1.0	1.08	1.22	1.43	1.08	1.22	1.43	1.08	1.21	1.41	1.09	1.22	1.44	1.10	1.23	1.44	1.09	1.22	1.43
1.1	1.25	1.41	1.66	1.25	1.41	1.66	1.24	1.38	1.61	1.26	1.41	1.66	1.26	1.41	1.66	1.26	1.41	1.65
1.2	1.29	1.44	1.70	1.29	1.44	1.70	1.27	1.42	1.65	1.29	1.45	1.70	1.30	1.45	1.70	1.29	1.44	1.70
1.3	1.39	1.56	1.84	1.39	1.56	1.84	1.37	1.53	1.78	1.39	1.56	1.84	1.39	1.56	1.83	1.39	1.56	1.83
1.4	1.46	1.63	1.93	1.46	1.63	1.93	**1.43**	1.60	1.86	1.46	1.64	1.93	1.46	1.63	1.92	1.46	1.63	1.92
1.5	**1.47**	1.65	1.94	**1.47**	1.65	1.94	**1.45**	1.61	1.88	**1.47**	1.65	1.94	**1.47**	1.65	1.93	**1.47**	1.64	1.93
1.6	**1.63**	1.83	2.15	**1.63**	1.82	2.15	**1.59**	1.78	2.07	**1.63**	1.82	2.15	**1.63**	1.82	2.14	**1.62**	1.82	2.14
1.7	**1.69**	1.90	2.24	**1.69**	1.90	2.24	**1.66**	1.85	2.15	**1.69**	1.90	2.24	**1.69**	1.89	2.22	**1.69**	1.89	2.22
1.8	**1.80**	2.02	2.38	**1.80**	2.01	2.38	**1.75**	1.95	2.28	**1.80**	2.01	2.37	**1.79**	2.00	2.35	**1.79**	2.00	2.36
1.9	2.00	2.24	2.65	2.00	2.24	2.65	**1.95**	2.17	2.52	2.00	2.24	2.64	1.99	2.23	2.61	1.99	2.23	2.62
2.0	2.08	2.33	2.75	2.08	2.33	2.75	**2.02**	2.25	2.62	2.08	2.33	2.74	2.07	2.31	2.71	2.07	2.32	2.72

In order to provide interval estimation, we employed our proposed bootstrap technique in conjunction with multi-level models. Tables 4.8 and 4.9 present the estimated confidence bounds for applied forces in x, y, and z directions, as well as Root Mean Square Error (RMSE) and Bias values. bootstrap results are obtained from models with considering weights (WLS). The bootstrap procedure is the same as what we explained in Section 4.1.2, except that instead of linear regression models, we fitted multi-level models as presented in (4.9). Based on the results presented in Tables 4.6 and 4.7, we selected W_3 to be used in the model for estimating forces in z direction under SEL and W_1 under LINEX loss in the bootstrapping procedure. We also considered $\alpha = 0.1$, since the results under this value was considerably better that other values.

Once again, we considered both models with and without intercept for obtaining the results. As can be seen in Tables 4.8 and 4.9, in most cases true value of the applied forces are within the estimated intervals and the Bias is very low. However, there are some differences between the results obtained for applied forces in z direction from models with intercept and models without intercept. While there are wider confidence bounds for models with intercepts, the bias and sometimes RMSE is lower for these models and most of the times true value of the force is within the estimated interval. For instance, when the applied force in z direction, F_z, is 0.1 N, calculated confidence interval by a model without intercept under SEL is (0.11 N , 0.74 N), compared to (0.167 N , 0.987 N) that is obtained based on a model with intercept. Furthermore, calculated confidence interval under LINEX loss is (0.134 N , 0.789 N) for models without intercept, and (0.000 N , 0.778 N) for models with intercept.

Table 4.8: Point and interval estimation of applied forces in x, y, and z directions using proposed bootstrap method with using multi-level models under SEL function.

Estimation of Models Without Intercept												
	Force in x Direction				**Force in y Direction**				**Force in z Direction**			
True Force	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**RMSE**
0.1	(0.07 , 0.12)	0.08	-0.02	0.02	(0.08 , 0.18)	0.10	0.00	0.05	(0.11 , 0.74)	0.18	0.08	0.38
0.2	(0.16 , 0.20)	0.17	-0.03	0.04	(0.16 , 0.27)	0.19	-0.01	0.04	(0.20 , 0.80)	0.29	0.09	0.33
0.3	(0.29 , 0.33)	0.30	0.00	0.02	(0.28 , 0.38)	0.30	0.00	0.04	(0.41 , 0.99)	0.50	0.20	0.38
0.4	(0.39 , 0.43)	0.40	0.00	0.02	(0.43 , 0.53)	0.45	0.05	0.07	(0.42 , 1.03)	0.50	0.10	0.42
0.5	(0.48 , 0.53)	0.49	-0.01	0.02	(0.46 , 0.55)	0.48	-0.02	0.05	(0.51 , 1.11)	0.58	0.08	0.35
0.6	(0.56 , 0.60)	0.57	-0.03	0.03	(0.56 , 0.66)	0.59	-0.01	0.05	(0.59 , 1.17)	0.67	0.07	0.53
0.7	(0.64 , 0.68)	0.65	-0.05	0.05	(0.74 , 0.84)	0.77	0.07	0.08	(0.73 , 1.34)	0.82	0.12	0.52
0.8	(0.78 , 0.83)	0.80	0.00	0.02	(0.71 , 0.81)	0.73	-0.07	0.08	(0.73 , 1.36)	0.82	0.02	0.37
0.9	(0.89 , 0.93)	0.90	0.00	0.02	(0.84 , 0.94)	0.86	-0.04	0.06	(0.77 , 1.35)	0.83	-0.07	0.53
1.0	(0.97 , 1.01)	0.98	-0.02	0.03	(0.94 , 1.05)	0.97	-0.03	0.05	(0.98 , 1.59)	1.07	0.07	0.35
1.1	(1.07 , 1.11)	1.08	-0.02	0.03	(1.05 , 1.15)	1.08	-0.02	0.05	(1.12 , 1.71)	1.19	0.09	0.54
1.2	(1.21 , 1.25)	1.22	0.02	0.03	(1.20 , 1.30)	1.23	0.03	0.05	(1.13 , 1.75)	1.20	0.00	0.37
1.3	(1.30 , 1.34)	1.31	0.01	0.02	(1.29 , 1.39)	1.31	0.01	0.05	(1.22 , 1.87)	1.32	0.02	0.39
1.4	(1.41 , 1.45)	1.42	0.02	0.03	(1.37 , 1.47)	1.40	0.00	0.04	(1.28 , 1.90)	1.37	-0.03	0.54
1.5	(1.50 , 1.55)	1.51	0.01	0.02	(1.54 , 1.63)	1.56	0.06	0.08	(1.30 , 1.92)	1.40	-0.10	0.33
1.6	(1.57 , 1.61)	1.58	-0.02	0.02	(1.63 , 1.72)	1.65	0.05	0.07	(1.42 , 2.00)	1.49	-0.11	0.72
1.7	(1.72 , 1.76)	1.73	0.03	0.04	(1.68 , 1.78)	1.70	0.00	0.04	(1.48 , 2.00)	1.55	-0.15	0.72
1.8	(1.82 , 1.86)	1.83	0.03	0.03	(1.83 , 1.92)	1.85	0.05	0.06	(1.57 , 2.00)	1.65	-0.15	0.37
1.9	(1.90 , 1.94)	1.91	0.01	0.02	(1.93 , 2.00)	1.96	0.06	0.07	(1.75 , 2.00)	1.81	-0.09	0.37
2.0	(2.00 , 2.00)	2.00	0.02	0.03	(2.00 , 2.00)	2.00	0.05	0.07	(1.81 , 2.00)	1.86	-0.14	0.43
Estimation of Models With Intercept												
	Force in x Direction				**Force in y Direction**				**Force in z Direction**			
True Force	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**RMSE**
0.1	(0.08 , 0.13)	0.09	-0.01	0.02	(0.07 , 0.17)	0.09	-0.01	0.05	(0.00 , 0.74)	0.10	0.00	0.41
0.2	(0.16 , 0.21)	0.18	-0.02	0.03	(0.16 , 0.26)	0.18	-0.02	0.05	(0.10 , 0.81)	0.21	0.01	0.36
0.3	(0.30 , 0.34)	0.31	0.01	0.02	(0.27 , 0.37)	0.29	-0.01	0.05	(0.34 , 1.04)	0.44	0.14	0.39
0.4	(0.40 , 0.44)	0.41	0.01	0.02	(0.42 , 0.52)	0.44	0.04	0.07	(0.33 , 1.05)	0.43	0.03	0.41
0.5	(0.49 , 0.53)	0.50	0.00	0.02	(0.45 , 0.55)	0.47	-0.03	0.05	(0.44 , 1.15)	0.53	0.03	0.38
0.6	(0.57 , 0.61)	0.58	-0.02	0.03	(0.56 , 0.66)	0.58	-0.02	0.05	(0.53 , 1.24)	0.62	0.02	0.40
0.7	(0.64 , 0.69)	0.66	-0.04	0.05	(0.74 , 0.84)	0.76	0.06	0.08	(0.69 , 1.39)	0.78	0.08	0.37
0.8	(0.79 , 0.83)	0.80	0.00	0.02	(0.71 , 0.81)	0.73	-0.07	0.08	(0.70 , 1.41)	0.79	-0.01	0.40
0.9	(0.89 , 0.93)	0.90	0.00	0.02	(0.84 , 0.94)	0.86	-0.04	0.06	(0.73 , 1.42)	0.82	-0.08	0.42
1.0	(0.97 , 1.01)	0.98	-0.02	0.02	(0.94 , 1.04)	0.97	-0.03	0.06	(0.97 , 1.65)	1.07	0.07	0.39
1.1	(1.07 , 1.11)	1.08	-0.02	0.03	(1.05 , 1.15)	1.08	-0.02	0.05	(1.11 , 1.80)	1.22	0.12	0.39
1.2	(1.21 , 1.25)	1.22	0.02	0.03	(1.20 , 1.31)	1.23	0.03	0.05	(1.14 , 1.83)	1.22	0.02	0.39
1.3	(1.30 , 1.34)	1.31	0.01	0.02	(1.28 , 1.39)	1.31	0.01	0.05	(1.24 , 2.00)	1.35	0.05	0.39
1.4	(1.41 , 1.45)	1.42	0.02	0.03	(1.37 , 1.47)	1.40	0.00	0.04	(1.30 , 2.00)	1.39	-0.01	0.42
1.5	(1.50 , 1.54)	1.51	0.01	0.02	(1.54 , 1.64)	1.56	0.06	0.08	(1.33 , 2.00)	1.44	-0.06	0.36
1.6	(1.57 , 1.61)	1.58	-0.02	0.03	(1.63 , 1.72)	1.65	0.05	0.07	(1.46 , 2.00)	1.55	-0.05	0.39
1.7	(1.72 , 1.76)	1.73	0.03	0.03	(1.68 , 1.78)	1.70	0.00	0.04	(1.53 , 2.00)	1.63	-0.07	0.40
1.8	(1.81 , 1.85)	1.82	0.02	0.03	(1.83 , 1.92)	1.85	0.05	0.07	(1.62 , 2.00)	1.72	-0.08	0.39
1.9	(1.89 , 1.94)	1.91	0.01	0.02	(1.94 , 2.00)	1.96	0.06	0.07	(1.80 , 2.00)	1.89	-0.01	0.41
2.0	(2.00 , 2.00)	2.00	0.02	0.03	(2.00 , 2.00)	2.00	0.06	0.07	(1.87 , 2.00)	1.98	-0.02	0.40

Table 4.9: Point and interval estimation of applied forces in x, y, and z directions using proposed bootstrap method with using multi-level models under LINEX loss function.

Estimation of Models Without Intercept												
	Force in x Direction				Force in y Direction				Force in z Direction			
True Force	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**RMSE**
0.1	(0.072 , 0.116)	0.085	-0.015	0.024	(0.076 , 0.179)	0.098	-0.002	0.046	(0.134 , 0.789)	0.212	0.112	0.413
0.2	(0.156 , 0.202)	0.169	-0.031	0.036	(0.165 , 0.267)	0.188	-0.012	0.045	(0.240 , 0.856)	0.329	0.129	0.368
0.3	(0.290 , 0.333)	0.302	0.002	0.018	(0.277 , 0.376)	0.300	-0.000	0.044	(0.461 , 1.069)	0.549	0.249	0.429
0.4	(0.394 , 0.435)	0.404	0.004	0.017	(0.429 , 0.527)	0.450	0.050	0.069	(0.469 , 1.091)	0.484	0.084	1.957
0.5	(0.485 , 0.527)	0.496	-0.004	0.018	(0.458 , 0.554)	0.480	-0.020	0.045	(0.563 , 1.191)	0.640	0.140	0.394
0.6	(0.562 , 0.605)	0.573	-0.027	0.032	(0.564 , 0.664)	0.586	-0.014	0.048	(0.650 , 1.252)	0.719	0.119	0.491
0.7	(0.641 , 0.684)	0.652	-0.048	0.051	(0.743 , 0.840)	0.766	0.066	0.079	(0.801 , 1.445)	0.949	0.249	1.940
0.8	(0.786 , 0.829)	0.798	-0.002	0.017	(0.710 , 0.810)	0.733	-0.067	0.081	(0.814 , 1.465)	0.892	0.092	0.418
0.9	(0.888 , 0.931)	0.900	-0.000	0.016	(0.840 , 0.942)	0.864	-0.036	0.056	(0.848 , 1.454)	0.912	0.012	0.463
1.0	(0.970 , 1.011)	0.981	-0.019	0.026	(0.945 , 1.047)	0.969	-0.031	0.054	(1.072 , 1.711)	1.161	0.161	0.405
1.1	(1.069 , 1.115)	1.081	-0.019	0.026	(1.054 , 1.153)	1.078	-0.022	0.049	(1.215 , 1.840)	1.298	0.198	0.472
1.2	(1.211 , 1.255)	1.223	0.023	0.029	(1.204 , 1.305)	1.227	0.027	0.052	(1.233 , 1.880)	1.312	0.112	0.429
1.3	(1.298 , 1.341)	1.310	0.010	0.020	(1.286 , 1.387)	1.309	0.009	0.047	(1.331 , 2.016)	1.438	0.138	0.462
1.4	(1.409 , 1.455)	1.421	0.021	0.028	(1.374 , 1.468)	1.397	-0.003	0.042	(1.397 , 2.050)	1.503	0.103	1.002
1.5	(1.506 , 1.548)	1.517	0.017	0.025	(1.537 , 1.635)	1.560	0.060	0.076	(1.417 , 2.075)	1.521	0.021	0.337
1.6	(1.575 , 1.617)	1.585	-0.015	0.023	(1.627 , 1.723)	1.649	0.049	0.066	(1.545 , 2.171)	1.625	0.025	0.473
1.7	(1.722 , 1.765)	1.733	0.033	0.038	(1.677 , 1.780)	1.699	-0.001	0.043	(1.611 , 2.257)	1.693	-0.007	0.480
1.8	(1.819 , 1.862)	1.831	0.031	0.035	(1.827 , 1.918)	1.849	0.049	0.065	(1.712 , 2.345)	1.794	-0.006	0.366
1.9	(1.902 , 1.946)	1.914	0.014	0.022	(1.936 , 2.033)	1.958	0.058	0.073	(1.889 , 2.548)	1.966	0.066	0.396
2.0	(2.015 , 2.060)	2.027	0.027	0.033	(2.030 , 2.132)	2.055	0.055	0.071	(1.959 , 2.629)	2.041	0.041	0.593
Estimation of Models With Intercept												
	Force in x Direction				Force in y Direction				Force in z Direction			
True Force	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**MSE**	**C.I**	$\hat{F}$	**Bias**	**MSE**
0.1	(0.082 , 0.126)	0.094	-0.006	0.019	(0.067 , 0.173)	0.091	-0.009	0.048	(0.000 , 0.778)	0.072	-0.028	0.452
0.2	(0.165 , 0.207)	0.177	-0.023	0.029	(0.158 , 0.262)	0.182	-0.018	0.047	(0.063 , 0.851)	0.193	-0.007	0.401
0.3	(0.298 , 0.339)	0.310	0.010	0.020	(0.270 , 0.372)	0.294	-0.006	0.045	(0.333 , 1.110)	0.449	0.149	0.430
0.4	(0.401 , 0.440)	0.411	0.011	0.020	(0.424 , 0.522)	0.445	0.045	0.065	(0.327 , 1.119)	0.427	0.027	0.455
0.5	(0.491 , 0.531)	0.502	0.002	0.017	(0.453 , 0.550)	0.475	-0.025	0.048	(0.438 , 1.236)	0.543	0.043	0.425
0.6	(0.568 , 0.609)	0.579	-0.021	0.027	(0.559 , 0.660)	0.582	-0.018	0.050	(0.546 , 1.320)	0.640	0.040	0.441
0.7	(0.646 , 0.688)	0.657	-0.043	0.046	(0.740 , 0.840)	0.763	0.063	0.077	(0.709 , 1.501)	0.822	0.122	0.420
0.8	(0.790 , 0.833)	0.802	0.002	0.017	(0.705 , 0.807)	0.729	-0.071	0.084	(0.723 , 1.523)	0.822	0.022	0.442
0.9	(0.892 , 0.932)	0.903	0.003	0.016	(0.838 , 0.939)	0.862	-0.038	0.058	(0.767 , 1.528)	0.862	-0.038	0.450
1.0	(0.972 , 1.012)	0.984	-0.016	0.024	(0.943 , 1.045)	0.967	-0.033	0.056	(1.020 , 1.780)	1.131	0.131	0.442
1.1	(1.072 , 1.116)	1.083	-0.017	0.025	(1.052 , 1.151)	1.077	-0.023	0.049	(1.181 , 1.954)	1.296	0.196	0.460
1.2	(1.213 , 1.254)	1.224	0.024	0.029	(1.203 , 1.306)	1.227	0.027	0.052	(1.202 , 1.972)	1.300	0.100	0.446
1.3	(1.299 , 1.340)	1.310	0.010	0.020	(1.285 , 1.389)	1.309	0.009	0.047	(1.319 , 2.220)	1.445	0.145	0.452
1.4	(1.409 , 1.454)	1.420	0.020	0.027	(1.374 , 1.469)	1.397	-0.003	0.042	(1.386 , 2.207)	1.491	0.091	0.474
1.5	(1.505 , 1.545)	1.516	0.016	0.023	(1.538 , 1.638)	1.561	0.061	0.077	(1.412 , 2.228)	1.543	0.043	0.398
1.6	(1.573 , 1.613)	1.583	-0.017	0.024	(1.627 , 1.724)	1.651	0.051	0.067	(1.560 , 2.322)	1.666	0.066	0.434
1.7	(1.719 , 1.760)	1.730	0.030	0.035	(1.678 , 1.783)	1.701	0.001	0.043	(1.637 , 2.452)	1.748	0.048	0.441
1.8	(1.816 , 1.857)	1.827	0.027	0.032	(1.830 , 1.922)	1.852	0.052	0.067	(1.734 , 2.535)	1.848	0.048	0.427
1.9	(1.898 , 1.940)	1.910	0.010	0.020	(1.939 , 2.038)	1.961	0.061	0.076	(1.924 , 2.757)	2.038	0.138	0.472
2.0	(2.010 , 2.053)	2.022	0.022	0.029	(2.034 , 2.136)	2.058	0.058	0.074	(2.005 , 2.852)	2.131	0.131	0.460

4.4 Bayesian Approach

Bayesian methodology could be employed in multi-level modeling regression to obtain better and more accurate estimation. According to Ruppert et al. (2003), Bayesian statistics is distinct from other statistics (*e.g.* ML statistics), because of the essence

of the prior and posterior distributions. By specifying prior in the estimation process, we consider some information about parameters of our model even before looking at calibration data set.

In the Bayesian analysis, one can choose *informative* or *noninformative* priors. When there are some available information on the parameters, we can employ those information to choose an informative prior; however, sometimes we do not know anything about our data set, in this case, one decides to use noninformative prior. Noninformative (or flat) priors are mainly chosen as uniform distributions or normal densities with considerably large variances. As in Ruppert et al. (2003), even improper priors can lead to proper posterior distributions. The posterior distribution includes all information about our parameter of interest.

In order to improve our estimation in terms of accuracy, we decided to employ Bayesian statistic instead of ML and REML procedures. Considering the model (4.9), β_k, u_k, G_k, R_k are the parameters of interest that we want to estimate using the Bayesian methodology. The posterior distribution for this problem is defined as:

$$P(\beta_k, u_k, G_k, R_k \mid S_k) \propto P(S_k \mid \beta_k, u_k, R_k)\, P(u_k \mid G_k)\, P(R_k)\, P(\beta_k)\, P(G_k). \quad (4.23)$$

Suppose,

$$u_k \sim N(0, G_k) \quad \text{and} \quad S_k \mid u_k \sim N(F\beta_k + Zu_k,\ R_k).$$

Therefore,

$$P(S_k \mid \beta_k, u_k, R_k)P(u_k \mid G_k) \propto \exp\left\{-\frac{1}{2}\Big((S_k - F\beta_k - Zu_k)^\top R_k^{-1}(S_k - F\beta_k - Zu_k)\Big)\right\} \times \exp\left\{-\frac{1}{2}\Big(u_k^\top G_k^{-1} u_k\Big)\right\}. \quad (4.24)$$

Since (4.24) is a nonnegative function of (β_k, u_k), after some mathematical manipulations, we have

$$P(S_k \mid \beta_k, u_k, R_k)P(u_k \mid G_k) \propto N((C^\top C + R_k D_k)^{-1}C^\top S_k, R_k(C^\top C + R_k D_k)^{-1}),$$

where, $C \equiv \begin{pmatrix} F & Z \end{pmatrix}$ and $D_k = \begin{pmatrix} 0 & 0 \\ 0 & G_k^{-1} \end{pmatrix}$.

Following Ruppert et al. (2003), and assuming that we do not have enough information on β_k, we use an improper prior on β_k as, $P(\beta_k) \equiv 1$, or $\beta_k \sim N(0, \sigma^2_{\beta_k})$, where, $\sigma^2_{\beta_k}$ is considerably large.

Furthermore, it has been suggested in Ruppert et al. (2003), to use inverse gamma densities as priors for R_k and G_k, with parameters (A_{r_k}, B_{r_k}) and (A_{g_k}, B_{g_k}), that are listed below:

$$\begin{cases} P(R_k) = \dfrac{B_{r_k}^{A_{r_k}}}{\Gamma(A_{r_k})}(R_k)^{-(A_{r_k}+1)} \exp\left(-\dfrac{B_{r_k}}{R_k}\right), \\ \\ P(G_k) = \dfrac{B_{g_k}^{A_{g_k}}}{\Gamma(A_{g_k})}(G_k)^{-(A_{g_k}+1)} \exp\left(-\dfrac{B_{g_k}}{G_k}\right), \end{cases} \tag{4.25}$$

where, (A_{r_k}, B_{r_k}) and (A_{g_k}, B_{g_k}) are called *hyper parameters*, that are positive. If we consider hyper parameters to be equal to zero, then the priors for R_k and G_k would be improper and equal to $\frac{1}{R_k}$ and $\frac{1}{G_k}$, respectively. It has been recommended in Ruppert et al. (2003), to consider hyper parameters close to zero (*e.g.* 0.1), to come up with noninformative, but proper priors. Based on 4.24,

$$R_k \propto (R_k)^{-(\frac{3}{2}n + A_{r_k}+1)} \exp\left\{-\frac{1}{2}\left((S_k - F\beta_k - Zu_k)^\top R_k^{-1}(S_k - F\beta_k - Zu_k) + B_{r_k}\right)\right\}. \tag{4.26}$$

Comparing 4.25 with 4.26, it is straightforward to show that

$$[R_k \mid S_k, \beta_k, u_k, G_k] \sim IG(A_{r_k} + \frac{3}{2}n, B_{r_k} + \frac{1}{2}(S_k - F\beta_k - Zu_k)^\top (S_k - F\beta_k - Zu_k)). \tag{4.27}$$

An argument similar to the one used in 4.26 for G_k shows that

$$[G_k \mid S_k, \beta_k, u_k, R_k] \sim IG(A_{g_k} + \frac{3}{2}, B_{g_k} + \frac{1}{2}u_k^\top u). \tag{4.28}$$

Finally, in order to make statistical inference, we use Markov Chain Monte Carlo (MCMC) to sample from the posterior distribution based on the following algorithm (Ruppert et al., 2003):

1. Sample (β, u) from the following multivariate normal distribution

$$N((C^\top C + R_k D_k)^{-1} C^\top S_k, R_k (C^\top C + R_k D_k)^{-1}).$$

2. Sample G_k from $IG(A_{g_k} + \frac{3}{2}, B_{g_k} + \frac{1}{2}u_k^\top u)$.

3. Sample R_k from $IG(A_{r_k} + \frac{3}{2}n, B_{r_k} + \frac{1}{2}(S_k - F\beta_k - Zu_k)^\top (S_k - F\beta_k - Zu_k))$.

4. Repeat steps 1 - 3, M times, where, M is the number of iterations.

In step 1 of the MCMC algorithm, we sample (β_k, u_k) given $(R_k^{(0)}, G_k^{(0)})$ values, while in steps 2 and 3, (G_k, R_k) are sampled when $(\beta_k^{(0)}, u_k^{(0)})$ are given. Given (β_k, u_k), R_k and G_k are independent, therefore, steps 2 and 3 of the MCMC algorithm are exchangeable. The values of $(\beta_k^{(0)}, u_k^{(0)}, R_k^{(0)}, G_k^{(0)})$ to start the MCMC algorithm are estimated using the ML or REML approaches as explained in Sections 4.3.1

and 4.3.2. According to Ruppert et al. (2003), this MCMC algorithm is flexible in terms of the values of $(\beta_k^{(0)}, u_k^{(0)}, R_k^{(0)}, G_k^{(0)})$. This is because the Marcov Chain would gradually converge to the stationary distribution and it discards the starting.

For the comparison purposes, we obtain the predictions under SEL, as well as LINEX loss functions. Through the Markov Chain Monte Carlo, we obtain samples with length of M for β and u, where under the SEL, we consider the mean value of these samples $\hat{u}$ and $\hat{\beta}$ as follow:

$$\begin{aligned} \hat{u} &= \frac{1}{M}\sum_{i=1}^{M} u_i, \\ \hat{\beta} &= \frac{1}{M}\sum_{i=1}^{M} \beta_i. \end{aligned} \tag{4.29}$$

However, based on (4.19), $\hat{u}$ and $\hat{\beta}$ under the LINEX loss function are obtained as follow:

$$\begin{aligned} \hat{u}_\alpha &= \frac{1}{M}\sum_{i=1}^{M} u_i - \frac{\sigma_{u_i}^2 \alpha}{2}, \\ \hat{\beta}_\alpha &= \frac{1}{M}\sum_{i=1}^{M} \beta_i - \frac{\sigma_{\beta_i}^2 \alpha}{2}. \end{aligned} \tag{4.30}$$

4.4.1 Results Obtained from Bayesian Approach

In this section, we report predicted values of the applied forces along x, y, and z directions that are obtained using the Bayesian approach under both SEL and LINEX

loss functions. To obtain the results, we used *lmer-stan* function of *"rstanarm"* package, in R programming language (Gabry and Goodrich, 2016).

Here again, to obtain more precise predictions we violated the physical model and considered intercepts in the model. Therefore, results are reported for models with, and without intercept. Point estimates of applied forces along x, y, and z directions are listed in Tables 4.10 and 4.11. Results show that there is not a significant difference between predictions obtained through models with intercept and models without intercept. For instance, when the true force is equal to 1.0 N, the estimated forces along x, y, and z directions based on models without intercept under SEL are 0.980 N, 0.969 N, 1.094 N, respectively. And estimated forces along x, y, and z directions through models with intercept under SEL are 0.982 N, 0.967 N, 1.097 N, respectively. This is also true for results obtained under the LINEX loss function.

As can be seen in Table 4.11, estimated forces along x and y axes are not altering based on different values of α. However, estimated F_z's are changing considerably with different values of α. For instance, when $F_z = 2$ N, estimated forces with $\alpha = (0.1, 0.5, 1)$ using models without intercept and models with intercept are $(1.966, 2.065, 2.203)$ N and $(1.953, 2.040, 2.161)$ N, respectively.

Table 4.10: Point estimates obtained by Bayesian approach in multi-level models considering models with and without intercept under SEL function for different amounts of forces applied to the right forceps tip, in x, y, and z directions.

	Models Without Intercept			**Models With Intercept**		
True Force	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$
0.1	0.086	0.096	0.216	0.094	0.090	0.237
0.2	0.169	0.188	0.303	0.176	0.183	0.322
0.3	0.302	0.297	0.515	0.308	0.292	0.530
0.4	0.405	0.448	0.525	0.411	0.444	0.539
0.5	0.495	0.478	0.612	0.501	0.474	0.624
0.6	0.573	0.587	0.699	0.578	0.583	0.709
0.7	0.652	0.763	0.834	0.656	0.760	0.842
0.8	0.798	0.734	0.853	0.801	0.731	0.861
0.9	0.899	0.864	0.882	0.902	0.862	0.889
1.0	0.980	0.969	1.094	0.982	0.967	1.097
1.1	1.079	1.078	1.239	1.081	1.077	1.239
1.2	1.221	1.225	1.268	1.222	1.224	1.267
1.3	1.308	1.308	1.355	1.308	1.309	1.352
1.4	1.418	1.397	1.413	1.418	1.397	1.409
1.5	1.514	1.560	1.422	1.513	1.562	1.418
1.6	1.582	1.648	1.557	1.581	1.650	1.551
1.7	1.730	1.699	1.615	1.728	1.701	1.607
1.8	1.827	1.850	1.702	1.825	1.853	1.692
1.9	1.910	1.959	1.876	1.907	1.962	1.862
2.0	2.022	2.055	1.944	2.019	2.059	1.929

Table 4.11: Point estimates obtained by Bayesian approach in multi-level models considering models with and without intercept under LINEX function for different amounts of forces applied to the right forceps tip, in x, y, and z directions.

	Estimated Forces in Models without Intercept								
	$\alpha_1 = 0.1$			$\alpha_2 = 0.5$			$\alpha_3 = 1$		
True Force	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$
0.1	0.086	0.096	0.219	0.086	0.096	0.230	0.087	0.096	0.245
0.2	0.170	0.188	0.306	0.170	0.188	0.322	0.170	0.188	0.343
0.3	0.302	0.297	0.521	0.303	0.297	0.547	0.303	0.298	0.584
0.4	0.405	0.448	0.531	0.406	0.448	0.558	0.406	0.449	0.595
0.5	0.496	0.478	0.619	0.496	0.478	0.650	0.497	0.478	0.694
0.6	0.573	0.587	0.707	0.574	0.587	0.742	0.574	0.588	0.792
0.7	0.652	0.763	0.843	0.653	0.763	0.886	0.654	0.764	0.945
0.8	0.798	0.734	0.863	0.799	0.734	0.906	0.800	0.735	0.967
0.9	0.899	0.864	0.892	0.900	0.864	0.937	0.901	0.865	1.000
1.0	0.980	0.969	1.107	0.981	0.969	1.163	0.983	0.970	1.240
1.1	1.080	1.078	1.253	1.081	1.078	1.316	1.082	1.079	1.405
1.2	1.222	1.225	1.283	1.223	1.225	1.347	1.225	1.226	1.437
1.3	1.308	1.308	1.370	1.310	1.309	1.439	1.311	1.310	1.536
1.4	1.419	1.397	1.429	1.420	1.398	1.501	1.422	1.399	1.602
1.5	1.515	1.560	1.439	1.516	1.561	1.511	1.518	1.563	1.612
1.6	1.583	1.648	1.575	1.584	1.649	1.655	1.587	1.651	1.766
1.7	1.730	1.699	1.634	1.732	1.700	1.716	1.734	1.701	1.831
1.8	1.828	1.850	1.722	1.830	1.851	1.808	1.832	1.853	1.930
1.9	1.911	1.959	1.898	1.913	1.960	1.993	1.915	1.962	2.127
2.0	2.023	2.055	1.966	2.025	2.057	2.065	2.028	2.059	2.203
	Estimated Forces in Models with Intercept								
	$\alpha_1 = 0.1$			$\alpha_2 = 0.5$			$\alpha_3 = 1$		
True Force	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$
0.1	0.094	0.090	0.240	0.094	0.090	0.250	0.094	0.090	0.265
0.2	0.176	0.183	0.326	0.177	0.183	0.340	0.177	0.183	0.360
0.3	0.308	0.292	0.536	0.309	0.292	0.560	0.309	0.293	0.593
0.4	0.411	0.444	0.546	0.411	0.444	0.570	0.412	0.445	0.604
0.5	0.501	0.473	0.632	0.501	0.474	0.660	0.502	0.474	0.699
0.6	0.578	0.583	0.718	0.578	0.583	0.750	0.579	0.584	0.795
0.7	0.656	0.760	0.852	0.657	0.761	0.890	0.658	0.761	0.943
0.8	0.801	0.731	0.871	0.802	0.731	0.910	0.803	0.732	0.964
0.9	0.902	0.861	0.900	0.903	0.862	0.940	0.904	0.863	0.996
1.0	0.982	0.967	1.110	0.984	0.967	1.160	0.985	0.968	1.229
1.1	1.081	1.076	1.254	1.083	1.077	1.310	1.084	1.078	1.388
1.2	1.222	1.224	1.283	1.224	1.225	1.340	1.225	1.226	1.419
1.3	1.309	1.308	1.369	1.310	1.309	1.430	1.312	1.310	1.515
1.4	1.418	1.397	1.426	1.420	1.398	1.490	1.422	1.399	1.578
1.5	1.514	1.561	1.436	1.515	1.562	1.500	1.517	1.564	1.589
1.6	1.581	1.650	1.570	1.583	1.651	1.640	1.585	1.652	1.737
1.7	1.728	1.700	1.627	1.730	1.701	1.700	1.732	1.703	1.801
1.8	1.825	1.852	1.713	1.827	1.853	1.790	1.829	1.855	1.896
1.9	1.907	1.962	1.886	1.910	1.963	1.970	1.912	1.965	2.087
2.0	2.019	2.059	1.953	2.021	2.060	2.040	2.024	2.062	2.161

Once again, since the point estimates of applied forces along z axis were not accurate enough, we decided to apply weights in the model (WLS). Therefore, weights that are introduced in Table 4.5 were applied to the models (4.9), and results are reported for models with considering intercept, as well as models without considering intercept. Tables 4.12 and 4.13 present the results for models under SEL and LINEX loss, respectively.

According to Tables 4.12 and 4.13, estimated forces using models with intercept are closer to the true values of forces. Among all weights that have been employed, W_4 provides more accurate results and most predictions are very close to the true amount of applied forces for models under both SEL and LINEX loss functions. For instance, there is no bias in estimating the true force of 2 N, under SEL when W_4 is applied.

Results under LINEX loss function are reported for $\alpha \in (0.1, 0.5, 1)$, and as can be seen more accurate results are provided when $\alpha = 1$.

Table 4.12: Estimated forces in z direction using Bayesian methodology in multi-level modeling, considering six different weights under SEL. Top table presents the obtained force estimates for models without considering intercept, and bottom table presents the estimation for models with intercept.

			Estimated F_z in Models without Intercept			
True F_z	W_1	W_2	W_3	W_4	W_5	W_6
0.1	0.22	0.22	0.21	0.22	0.22	0.22
0.2	0.30	0.30	0.29	0.31	0.30	0.30
0.3	0.52	0.52	0.49	0.53	0.52	0.52
0.4	0.52	0.52	0.50	0.54	0.53	0.53
0.5	0.61	0.61	0.59	0.62	0.61	0.61
0.6	0.70	0.70	0.67	0.71	0.70	0.70
0.7	0.83	0.83	0.80	0.85	0.84	0.83
0.8	0.85	0.85	0.82	0.87	0.86	0.85
0.9	0.88	0.88	0.85	0.90	0.88	0.88
1.0	1.09	1.09	1.05	1.12	1.10	1.10
1.1	1.24	1.24	1.19	1.26	1.24	1.24
1.2	1.27	1.27	1.22	1.29	1.27	1.27
1.3	1.35	1.35	1.30	1.38	1.36	1.36
1.4	1.41	1.41	1.36	1.44	1.42	1.41
1.5	1.42	1.42	1.37	1.45	1.43	1.42
1.6	1.56	1.56	1.50	1.59	1.56	1.56
1.7	1.61	1.61	1.55	1.65	1.62	1.62
1.8	1.70	1.70	1.64	1.74	1.71	1.70
1.9	1.88	1.88	1.80	1.91	1.88	1.88
2.0	1.94	1.94	1.87	1.98	1.95	1.94
			Estimated F_z in Models with Intercept			
True F_z	W_1	W_2	W_3	W_4	W_5	W_6
0.1	**0.05**	**0.05**	**0.11**	**0.07**	**0.09**	**0.07**
0.2	**0.15**	**0.15**	**0.21**	**0.17**	**0.19**	**0.17**
0.3	0.39	0.39	0.43	0.41	0.43	0.41
0.4	**0.41**	**0.41**	**0.44**	**0.42**	**0.44**	**0.42**
0.5	**0.50**	**0.50**	**0.54**	**0.51**	**0.53**	**0.52**
0.6	**0.60**	**0.60**	**0.63**	**0.61**	**0.63**	**0.61**
0.7	0.76	0.76	0.78	0.76	0.78	0.76
0.8	**0.78**	**0.78**	**0.80**	**0.78**	**0.80**	**0.79**
0.9	0.81	0.81	0.83	0.82	**0.84**	0.82
1.0	1.06	1.06	1.06	**1.05**	1.07	1.06
1.1	1.22	1.22	1.21	1.22	1.24	1.22
1.2	**1.25**	**1.25**	**1.24**	**1.25**	1.27	**1.25**
1.3	**1.35**	**1.35**	**1.34**	**1.34**	1.37	**1.35**
1.4	**1.42**	**1.42**	**1.40**	**1.41**	**1.43**	**1.41**
1.5	1.43	1.43	1.41	1.42	**1.44**	1.42
1.6	**1.58**	**1.58**	**1.55**	**1.57**	**1.59**	**1.58**
1.7	**1.65**	**1.65**	1.62	1.64	**1.66**	1.64
1.8	**1.75**	**1.75**	1.71	1.73	**1.76**	1.74
1.9	**1.95**	**1.95**	**1.90**	**1.93**	**1.95**	**1.93**
2.0	**2.03**	**2.02**	**1.97**	**2.00**	**2.03**	**2.01**

Another point to consider is the robustness of the algorithm with respect to the choice of different priors for (β, u). To check this, we obtained the force prediction in x, y, and z directions under SEL function considering **noninformative**, **normal**, and **student-t** priors, and results are presented in Table 4.14. As demonstrated, there is not any significant difference between estimates under different priors.

Table 4.13: Estimated forces in z direction, considering six different weights under LINEX loss function. Top table presents the obtained force estimates for models without considering intercept, and bottom table presents the estimation for models with intercept.

Estimated F_z in Models without Intercept

	$\alpha_1 = 0.1$						$\alpha_2 = 0.5$						$\alpha_3 = 1$					
True F_z	W_1	W_2	W_3	W_4	W_5	W_6	W_1	W_2	W_3	W_4	W_5	W_6	W_1	W_2	W_3	W_4	W_5	W_6
0.1	0.221	0.242	0.276	0.218	0.224	0.233	0.210	0.216	0.226	0.222	0.233	0.249	0.222	0.238	0.260	0.218	0.224	0.233
0.2	0.309	0.339	0.386	0.305	0.314	0.326	0.294	0.303	0.316	0.311	0.327	0.349	0.312	0.333	0.364	0.305	0.315	0.327
0.3	0.526	0.577	0.657	0.519	0.534	0.555	0.500	0.516	0.538	0.528	0.556	0.594	0.530	0.566	0.619	0.519	0.535	0.557
0.4	0.536	0.588	0.670	0.529	0.544	0.565	0.509	0.526	0.548	0.538	0.566	0.605	0.540	0.577	0.630	0.529	0.545	0.567
0.5	0.625	0.686	0.780	0.616	0.634	0.658	0.594	0.613	0.639	0.627	0.660	0.705	0.630	0.672	0.735	0.616	0.635	0.661
0.6	0.714	0.783	0.891	0.703	0.724	0.752	0.678	0.700	0.729	0.717	0.753	0.805	0.719	0.768	0.839	0.704	0.725	0.755
0.7	0.852	0.934	1.063	0.839	0.864	0.897	0.809	0.835	0.870	0.855	0.899	0.961	0.858	0.916	1.001	0.840	0.866	0.901
0.8	0.871	0.956	1.088	0.859	0.884	0.918	0.828	0.854	0.890	0.875	0.920	0.983	0.878	0.938	1.024	0.859	0.886	0.922
0.9	0.901	0.988	1.125	0.888	0.914	0.949	0.856	0.883	0.921	0.905	0.951	1.016	0.908	0.969	1.059	0.888	0.916	0.953
1.0	1.118	1.226	1.396	1.102	1.134	1.178	1.062	1.096	1.142	1.122	1.180	1.261	1.126	1.203	1.314	1.102	1.136	1.182
1.1	1.265	1.388	1.580	1.248	1.284	1.334	1.202	1.241	1.293	1.271	1.336	1.428	1.275	1.362	1.488	1.248	1.287	1.339
1.2	1.295	1.421	1.617	1.277	1.314	1.365	1.230	1.270	1.324	1.300	1.367	1.461	1.305	1.394	1.523	1.277	1.317	1.370
1.3	1.384	1.518	1.728	1.364	1.404	1.458	1.314	1.357	1.414	1.390	1.461	1.561	1.395	1.489	1.627	1.364	1.407	1.464
1.4	1.443	1.583	1.802	1.422	1.464	1.521	1.371	1.415	1.475	1.449	1.523	1.628	1.454	1.553	1.697	1.423	1.467	1.526
1.5	1.453	1.594	1.814	1.432	1.474	1.531	1.380	1.425	1.485	1.459	1.534	1.639	1.464	1.563	1.708	1.433	1.477	1.537
1.6	1.591	1.745	1.987	1.568	1.615	1.676	1.511	1.560	1.626	1.597	1.679	1.794	1.603	1.712	1.870	1.569	1.617	1.683
1.7	1.650	1.810	2.061	1.626	1.675	1.739	1.567	1.618	1.686	1.657	1.742	1.861	1.663	1.776	1.940	1.627	1.677	1.745
1.8	1.739	1.908	2.171	1.714	1.765	1.832	1.651	1.705	1.777	1.746	1.835	1.961	1.752	1.871	2.044	1.714	1.768	1.839
1.9	1.916	2.102	2.393	1.889	1.945	2.019	1.820	1.879	1.958	1.924	2.023	2.161	1.931	2.062	2.253	1.889	1.948	2.027
2.0	1.985	2.178	2.479	1.957	2.015	2.092	1.886	1.947	2.029	1.993	2.096	2.239	2.001	2.136	2.334	1.957	2.018	2.100

Estimated F_z in Models with Intercept

	$\alpha_1 = 0.1$						$\alpha_2 = 0.5$						$\alpha_3 = 1$					
True F_z	W_1	W_2	W_3	W_4	W_5	W_6	W_1	W_2	W_3	W_4	W_5	W_6	W_1	W_2	W_3	W_4	W_5	W_6
0.1	0.055	0.064	0.081	0.055	0.058	0.062	0.114	0.117	0.122	0.069	0.074	0.080	0.095	0.103	0.114	0.074	0.076	0.080
0.2	0.157	0.184	0.232	0.155	0.164	0.178	0.208	0.214	0.224	0.169	0.180	0.195	0.193	0.209	0.233	0.172	0.178	0.186
0.3	0.408	0.477	0.602	0.401	0.425	0.459	0.438	0.452	0.472	0.413	0.438	0.475	0.434	0.470	0.523	0.413	0.427	0.446
0.4	0.420	0.490	0.619	0.412	0.436	0.472	0.448	0.463	0.483	0.424	0.450	0.488	0.445	0.481	0.536	0.424	0.438	0.458
0.5	0.522	0.610	0.770	0.512	0.543	0.587	0.542	0.560	0.584	0.523	0.556	0.602	0.543	0.588	0.655	0.522	0.540	0.564
0.6	0.625	0.729	0.922	0.613	0.649	0.702	0.636	0.657	0.686	0.623	0.662	0.717	0.642	0.694	0.774	0.620	0.642	0.671
0.7	0.785	0.916	1.157	0.769	0.815	0.881	0.783	0.808	0.843	0.778	0.826	0.895	0.795	0.860	0.958	0.773	0.800	0.836
0.8	0.808	0.943	1.191	0.791	0.839	0.906	0.804	0.830	0.866	0.800	0.850	0.921	0.817	0.884	0.985	0.795	0.823	0.860
0.9	0.842	0.982	1.242	0.825	0.874	0.945	0.835	0.863	0.900	0.833	0.885	0.959	0.850	0.919	1.024	0.828	0.857	0.895
1.0	1.093	1.275	1.612	1.070	1.134	1.226	1.065	1.100	1.147	1.077	1.144	1.239	1.090	1.180	1.314	1.069	1.106	1.155
1.1	1.264	1.475	1.864	1.238	1.312	1.418	1.222	1.262	1.316	1.243	1.320	1.430	1.255	1.357	1.512	1.233	1.275	1.333
1.2	1.298	1.515	1.915	1.271	1.347	1.456	1.253	1.295	1.350	1.276	1.355	1.469	1.287	1.393	1.552	1.266	1.309	1.368
1.3	1.401	1.635	2.066	1.372	1.454	1.571	1.347	1.392	1.452	1.376	1.461	1.583	1.386	1.499	1.670	1.364	1.411	1.475
1.4	1.469	1.715	2.167	1.439	1.525	1.648	1.410	1.457	1.519	1.442	1.532	1.660	1.452	1.570	1.750	1.430	1.479	1.546
1.5	1.481	1.728	2.184	1.450	1.537	1.661	1.421	1.467	1.530	1.453	1.543	1.672	1.462	1.582	1.763	1.441	1.490	1.557
1.6	1.641	1.914	2.419	1.606	1.702	1.840	1.567	1.619	1.688	1.609	1.708	1.851	1.616	1.748	1.947	1.594	1.649	1.723
1.7	1.709	1.994	2.520	1.673	1.773	1.916	1.630	1.683	1.756	1.675	1.778	1.927	1.681	1.819	2.026	1.659	1.717	1.794
1.8	1.812	2.114	2.672	1.774	1.880	2.031	1.724	1.781	1.857	1.775	1.884	2.042	1.780	1.926	2.145	1.758	1.818	1.900
1.9	2.017	2.354	2.975	1.974	2.093	2.262	1.912	1.975	2.060	1.974	2.096	2.271	1.977	2.139	2.382	1.955	2.022	2.113
2.0	2.097	2.447	3.092	2.053	2.175	2.351	1.985	2.051	2.139	2.051	2.178	2.360	2.053	2.221	2.475	2.031	2.101	2.196

Table 4.14: Point estimates of applied forces in x, y, and z directions using Bayesian approach in multi-level modeling using different priors for models with and without intercept.

Estimation of Models Without Intercept									
Prior	**Noninformative**			**Normal**			**Student-t**		
True Force	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$
0.1	0.086	0.096	0.216	0.086	0.096	0.215	0.086	0.096	0.216
0.2	0.169	0.188	0.303	0.170	0.188	0.301	0.169	0.188	0.303
0.3	0.302	0.297	0.516	0.302	0.297	0.512	0.302	0.297	0.515
0.4	0.405	0.448	0.525	0.405	0.448	0.522	0.405	0.448	0.525
0.5	0.495	0.478	0.612	0.496	0.477	0.608	0.495	0.478	0.611
0.6	0.573	0.587	0.699	0.573	0.586	0.695	0.573	0.587	0.698
0.7	0.652	0.763	0.834	0.652	0.763	0.829	0.652	0.763	0.833
0.8	0.797	0.734	0.853	0.798	0.733	0.848	0.797	0.734	0.853
0.9	0.899	0.864	0.882	0.899	0.863	0.877	0.899	0.864	0.881
1.0	0.980	0.968	1.095	0.980	0.968	1.088	0.980	0.969	1.094
1.1	1.079	1.078	1.240	1.080	1.077	1.232	1.079	1.078	1.238
1.2	1.221	1.224	1.269	1.222	1.224	1.261	1.221	1.225	1.267
1.3	1.307	1.308	1.356	1.308	1.308	1.347	1.307	1.309	1.354
1.4	1.418	1.397	1.413	1.419	1.396	1.405	1.418	1.397	1.412
1.5	1.514	1.560	1.423	1.515	1.560	1.415	1.514	1.560	1.422
1.6	1.582	1.648	1.558	1.583	1.648	1.549	1.582	1.648	1.557
1.7	1.729	1.699	1.616	1.730	1.698	1.606	1.729	1.699	1.614
1.8	1.827	1.850	1.703	1.828	1.849	1.693	1.827	1.850	1.701
1.9	1.910	1.959	1.877	1.911	1.958	1.866	1.910	1.959	1.875
2.0	2.022	2.055	1.945	2.023	2.055	1.933	2.022	2.056	1.942
Estimation of Models With Intercept									
Prior	**Noninformative**			**Normal**			**Student-t**		
True Force	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$	$\hat{F}_x$	$\hat{F}_y$	$\hat{F}_z$
0.1	0.094	0.090	0.238	0.094	0.090	0.236	0.094	0.090	0.235
0.2	0.176	0.183	0.324	0.176	0.183	0.321	0.176	0.183	0.320
0.3	0.308	0.292	0.533	0.308	0.292	0.530	0.308	0.292	0.528
0.4	0.411	0.444	0.543	0.411	0.444	0.539	0.411	0.444	0.537
0.5	0.500	0.474	0.629	0.501	0.473	0.625	0.500	0.474	0.622
0.6	0.577	0.583	0.715	0.578	0.583	0.710	0.577	0.584	0.707
0.7	0.656	0.760	0.848	0.656	0.760	0.843	0.656	0.761	0.839
0.8	0.801	0.731	0.867	0.801	0.730	0.862	0.801	0.731	0.858
0.9	0.901	0.861	0.896	0.902	0.861	0.890	0.901	0.862	0.886
1.0	0.982	0.967	1.106	0.982	0.966	1.099	0.982	0.967	1.094
1.1	1.081	1.076	1.249	1.081	1.076	1.241	1.081	1.077	1.235
1.3	1.308	1.308	1.363	1.308	1.308	1.355	1.308	1.309	1.348
1.4	1.418	1.397	1.420	1.418	1.396	1.412	1.417	1.398	1.405
1.5	1.513	1.561	1.430	1.513	1.561	1.421	1.513	1.562	1.414
1.6	1.581	1.650	1.563	1.581	1.649	1.554	1.580	1.651	1.546
1.7	1.727	1.700	1.620	1.728	1.700	1.611	1.727	1.701	1.603
1.8	1.824	1.852	1.706	1.825	1.851	1.696	1.824	1.853	1.688
1.9	1.906	1.962	1.878	1.907	1.961	1.867	1.906	1.963	1.857
2.0	2.018	2.059	1.945	2.019	2.058	1.933	2.018	2.060	1.923

Now, we are going to employ the proposed bootstrap method for Bayesian approach, that was explained in Section 4.1.2. The only difference between the procedure explained in Section 4.1.2, and what we used here is that instead of (4.1), we used the multi-level model (4.9) and a Bayesian methodology to estimate the coefficients (β, u). As we concluded before, we have applied W_4 and $\alpha = 1$ in the models to obtain the following results.

Results of bootstrapping under SEL and LINEX loss functions are presented in Tables 4.15 and 4.16, respectively. As it is shown, calculated confidence bounds for applied forces along x and y axes under SEL, are not very different between models with intercept or models without intercept. However, confidence bounds that we obtained using models with intercept for F_z are more precise and almost always contain the true values of force, nevertheless the bounds are relatively wide. Point estimates of applied forces in z directions are much more better than other estimates and the bias is negligible under SEL. For example, when force of 0.9 N is applied along z axis, bootstrap approach gives the point estimate of 0.892 N, and confidence bound of (0.797 N,1.345 N).

However, results obtained under the LINEX loss function are not very favorable. As can be seen, estimated confidence intervals for F_x and F_y are very narrow, and often does not contain the true values of force. Although, the calculated bounds for F_z are also narrower, true values of force are not included within the bounds. The advantage of results obtained under the LINEX loss is that the point estimates with considering intercept in the models are relatively accurate.

Table 4.15: Point and interval estimation of applied forces in x, y, and z directions using proposed bootstrap method with using Bayesian approach in multi-level modeling under SEL function.

Estimation of Models Without Intercept												
	Force in x Direction				Force in y Direction				Force in z Direction			
True Force	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**RMSE**
0.1	(0.071 , 0.115)	0.083	-0.017	0.024	(0.073 , 0.166)	0.100	0.000	0.035	(0.047 , 0.478)	0.113	0.013	0.301
0.2	(0.154 , 0.198)	0.166	-0.034	0.038	(0.166 , 0.260)	0.192	-0.008	0.036	(0.203 , 0.636)	0.272	0.072	0.309
0.3	(0.287 , 0.331)	0.299	-0.001	0.017	(0.275 , 0.369)	0.301	0.001	0.035	(0.359 , 0.795)	0.430	0.130	0.328
0.4	(0.390 , 0.434)	0.402	0.002	0.017	(0.426 , 0.520)	0.452	0.052	0.063	(0.672 , 1.113)	0.748	0.348	0.460
0.5	(0.481 , 0.525)	0.492	-0.008	0.018	(0.455 , 0.550)	0.482	-0.018	0.040	(0.515 , 0.954)	0.589	0.089	0.314
0.6	(0.558 , 0.602)	0.570	-0.030	0.034	(0.565 , 0.659)	0.590	-0.010	0.036	(0.682 , 1.123)	0.758	0.158	0.340
0.7	(0.637 , 0.682)	0.649	-0.051	0.053	(0.741 , 0.835)	0.767	0.067	0.075	(0.594 , 1.034)	0.668	-0.032	0.303
0.8	(0.786 , 0.827)	0.799	-0.001	0.016	(0.709 , 0.805)	0.733	-0.067	0.075	(0.770 , 1.213)	0.847	0.047	0.305
0.9	(0.884 , 0.929)	0.896	-0.004	0.017	(0.842 , 0.936)	0.867	-0.033	0.048	(0.878 , 1.326)	0.956	0.056	0.307
1.0	(0.965 , 1.010)	0.977	-0.023	0.028	(0.946 , 1.041)	0.972	-0.028	0.045	(1.094 , 1.551)	1.174	0.174	0.349
1.1	(1.065 , 1.110)	1.077	-0.023	0.028	(1.056 , 1.150)	1.081	-0.019	0.040	(1.044 , 1.500)	1.124	0.024	0.303
1.2	(1.207 , 1.252)	1.219	0.019	0.025	(1.202 , 1.297)	1.228	0.028	0.045	(1.153 , 1.613)	1.233	0.033	0.305
1.3	(1.293 , 1.339)	1.306	0.006	0.018	(1.285 , 1.380)	1.311	0.011	0.037	(1.231 , 1.695)	1.313	0.013	0.303
1.4	(1.404 , 1.449)	1.416	0.016	0.023	(1.373 , 1.469)	1.399	-0.001	0.035	(1.408 , 1.880)	1.491	0.091	0.317
1.5	(1.500 , 1.545)	1.512	0.012	0.021	(1.536 , 1.632)	1.563	0.063	0.072	(1.536 , 2.014)	1.620	0.120	0.327
1.6	(1.568 , 1.614)	1.580	-0.020	0.026	(1.624 , 1.720)	1.651	0.051	0.062	(1.555 , 2.034)	1.640	0.040	0.307
1.7	(1.715 , 1.761)	1.728	0.028	0.033	(1.674 , 1.771)	1.701	0.001	0.035	(1.614 , 2.096)	1.699	-0.001	0.305
1.8	(1.813 , 1.859)	1.826	0.026	0.031	(1.825 , 1.922)	1.852	0.052	0.063	(1.693 , 2.178)	1.779	-0.021	0.306
1.9	(1.896 , 1.942)	1.909	0.009	0.019	(1.934 , 2.031)	1.961	0.061	0.070	(1.812 , 2.302)	1.898	-0.002	0.306
2.0	(2.008 , 2.054)	2.021	0.021	0.027	(2.031 , 2.127)	2.057	0.057	0.067	(1.822 , 2.312)	1.908	-0.092	0.319
Estimation of Models With Intercept												
	Force in x Direction				Force in y Direction				Force in z Direction			
True Force	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**RMSE**
0.1	(0.080 , 0.120)	0.092	-0.008	0.018	(0.067 , 0.162)	0.093	-0.007	0.036	(0.000 , 0.419)	0.000	-0.124	0.354
0.2	(0.162 , 0.203)	0.175	-0.025	0.030	(0.161 , 0.257)	0.185	-0.015	0.038	(0.074 , 0.593)	0.148	-0.052	0.335
0.3	(0.294 , 0.335)	0.307	0.007	0.017	(0.271 , 0.367)	0.295	-0.005	0.036	(0.245 , 0.768)	0.321	0.021	0.331
0.4	(0.397 , 0.437)	0.409	0.009	0.018	(0.423 , 0.518)	0.447	0.047	0.059	(0.565 , 1.116)	0.666	0.266	0.424
0.5	(0.486 , 0.527)	0.499	-0.001	0.016	(0.452 , 0.548)	0.476	-0.024	0.043	(0.409 , 0.942)	0.493	-0.007	0.330
0.6	(0.563 , 0.603)	0.576	-0.024	0.029	(0.562 , 0.658)	0.586	-0.014	0.038	(0.575 , 1.127)	0.677	0.077	0.339
0.7	(0.642 , 0.682)	0.655	-0.045	0.048	(0.738 , 0.835)	0.763	0.063	0.072	(0.489 , 1.029)	0.580	-0.120	0.352
0.8	(0.786 , 0.827)	0.799	-0.001	0.016	(0.709 , 0.805)	0.733	-0.067	0.075	(0.666 , 1.225)	0.774	-0.026	0.332
0.9	(0.887 , 0.928)	0.900	0.000	0.016	(0.839 , 0.936)	0.864	-0.036	0.050	(0.797 , 1.345)	0.892	-0.008	0.331
1.0	(0.967 , 1.008)	0.981	-0.019	0.025	(0.945 , 1.042)	0.969	-0.031	0.047	(1.037 , 1.584)	1.129	0.129	0.356
1.1	(1.066 , 1.107)	1.080	-0.020	0.026	(1.054 , 1.151)	1.079	-0.021	0.041	(0.981 , 1.530)	1.076	-0.024	0.332
1.2	(1.207 , 1.249)	1.221	0.021	0.026	(1.201 , 1.299)	1.227	0.027	0.044	(1.105 , 1.649)	1.194	-0.006	0.332
1.3	(1.293 , 1.335)	1.307	0.007	0.018	(1.286 , 1.384)	1.311	0.011	0.037	(1.195 , 1.737)	1.280	-0.020	0.333
1.4	(1.403 , 1.445)	1.417	0.017	0.023	(1.374 , 1.473)	1.399	-0.001	0.036	(1.390 , 1.933)	1.474	0.074	0.341
1.5	(1.499 , 1.540)	1.512	0.012	0.020	(1.538 , 1.637)	1.564	0.064	0.073	(1.526 , 2.074)	1.615	0.115	0.353
1.6	(1.566 , 1.608)	1.579	-0.021	0.026	(1.626 , 1.726)	1.652	0.052	0.063	(1.546 , 2.096)	1.636	0.036	0.336
1.7	(1.713 , 1.755)	1.726	0.026	0.031	(1.676 , 1.777)	1.703	0.003	0.036	(1.611 , 2.161)	1.701	0.001	0.335
1.8	(1.810 , 1.852)	1.823	0.023	0.028	(1.827 , 1.929)	1.855	0.055	0.065	(1.698 , 2.248)	1.787	-0.013	0.336
1.9	(1.893 , 1.934)	1.905	0.005	0.017	(1.936 , 2.039)	1.964	0.064	0.074	(1.824 , 2.379)	1.917	0.017	0.337
2.0	(2.004 , 2.046)	2.017	0.017	0.024	(2.033 , 2.136)	2.061	0.061	0.071	(1.834 , 2.390)	1.927	-0.073	0.344

Table 4.16: Point and interval estimation of applied forces in x, y, and z directions using proposed bootstrap method with using Bayesian approach in multi-level modeling under LINEX loss function.

Estimation of Models Without Intercept												
	Force in x Direction				**Force in y Direction**				**Force in z Direction**			
True Force	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**RMSE**
0.1	(0.087 , 0.087)	0.083	-0.017	0.024	(0.096 , 0.096)	0.097	-0.003	0.042	(0.192 , 0.218)	0.111	0.011	0.361
0.2	(0.170 , 0.170)	0.166	-0.034	0.038	(0.188 , 0.189)	0.189	-0.011	0.043	(0.373 , 0.422)	0.268	0.068	0.368
0.3	(0.303 , 0.304)	0.299	-0.001	0.017	(0.297 , 0.299)	0.298	-0.002	0.042	(0.553 , 0.626)	0.426	0.126	0.383
0.4	(0.406 , 0.408)	0.402	0.002	0.017	(0.448 , 0.451)	0.449	0.049	0.064	(0.914 , 1.034)	0.741	0.341	0.498
0.5	(0.497 , 0.498)	0.492	-0.008	0.018	(0.477 , 0.480)	0.478	-0.022	0.047	(0.733 , 0.830)	0.584	0.084	0.372
0.6	(0.575 , 0.576)	0.570	-0.030	0.034	(0.586 , 0.590)	0.587	-0.013	0.044	(0.925 , 1.046)	0.751	0.151	0.393
0.7	(0.654 , 0.656)	0.649	-0.051	0.053	(0.762 , 0.767)	0.763	0.063	0.076	(0.824 , 0.932)	0.663	-0.037	0.364
0.8	(0.800 , 0.802)	0.795	-0.005	0.017	(0.733 , 0.738)	0.734	-0.066	0.078	(1.026 , 1.161)	0.840	0.040	0.365
0.9	(0.902 , 0.904)	0.896	-0.004	0.017	(0.863 , 0.868)	0.864	-0.036	0.055	(1.150 , 1.302)	0.948	0.048	0.366
1.0	(0.983 , 0.986)	0.977	-0.023	0.028	(0.968 , 0.974)	0.969	-0.031	0.052	(1.398 , 1.582)	1.165	0.165	0.400
1.1	(1.083 , 1.086)	1.077	-0.023	0.028	(1.077 , 1.084)	1.078	-0.022	0.047	(1.342 , 1.518)	1.116	0.016	0.364
1.2	(1.225 , 1.228)	1.219	0.019	0.025	(1.223 , 1.231)	1.224	0.024	0.048	(1.466 , 1.659)	1.224	0.024	0.365
1.3	(1.312 , 1.315)	1.306	0.006	0.018	(1.307 , 1.316)	1.308	0.008	0.043	(1.556 , 1.761)	1.303	0.003	0.364
1.4	(1.423 , 1.427)	1.416	0.016	0.023	(1.395 , 1.404)	1.396	-0.004	0.042	(1.759 , 1.990)	1.480	0.080	0.374
1.5	(1.519 , 1.523)	1.512	0.012	0.021	(1.559 , 1.569)	1.560	0.060	0.073	(1.906 , 2.156)	1.608	0.108	0.381
1.6	(1.587 , 1.591)	1.580	-0.020	0.026	(1.647 , 1.657)	1.648	0.048	0.063	(1.928 , 2.181)	1.628	0.028	0.367
1.7	(1.735 , 1.740)	1.728	0.028	0.033	(1.697 , 1.708)	1.698	-0.002	0.042	(1.996 , 2.258)	1.687	-0.013	0.366
1.8	(1.833 , 1.838)	1.826	0.026	0.031	(1.848 , 1.860)	1.849	0.049	0.064	(2.086 , 2.360)	1.766	-0.034	0.368
1.9	(1.916 , 1.921)	1.909	0.009	0.019	(1.957 , 1.970)	1.958	0.058	0.071	(2.221 , 2.513)	1.884	-0.016	0.367
2.0	(2.029 , 2.034)	2.021	0.021	0.027	(2.053 , 2.067)	2.054	0.054	0.068	(2.232 , 2.526)	1.894	-0.106	0.382
Estimation of Models With Intercept												
	Force in x Direction				**Force in y Direction**				**Force in z Direction**			
True Force	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**RMSE**	**C.I**	$\hat{F}$	**Bias**	**RMSE**
0.1	(0.093 , 0.097)	0.091	-0.009	0.017	(0.088 , 0.098)	0.090	-0.010	0.043	(0.052 , 0.209)	-0.040	-0.140	0.415
0.2	(0.176 , 0.180)	0.176	-0.024	0.028	(0.181 , 0.190)	0.183	-0.017	0.045	(0.271 , 0.406)	0.134	-0.066	0.396
0.3	(0.309 , 0.312)	0.306	0.006	0.015	(0.291 , 0.300)	0.292	-0.008	0.042	(0.487 , 0.603)	0.309	0.009	0.391
0.4	(0.412 , 0.415)	0.409	0.009	0.016	(0.443 , 0.451)	0.444	0.044	0.061	(0.910 , 1.034)	0.658	0.258	0.468
0.5	(0.502 , 0.505)	0.498	-0.002	0.014	(0.473 , 0.481)	0.473	-0.027	0.049	(0.699 , 0.810)	0.483	-0.017	0.391
0.6	(0.580 , 0.583)	0.576	-0.024	0.028	(0.582 , 0.590)	0.583	-0.017	0.045	(0.924 , 1.049)	0.669	0.069	0.397
0.7	(0.659 , 0.662)	0.655	-0.045	0.048	(0.760 , 0.767)	0.760	0.060	0.073	(0.807 , 0.919)	0.571	-0.129	0.412
0.8	(0.804 , 0.807)	0.800	-0.000	0.014	(0.731 , 0.737)	0.730	-0.070	0.081	(1.041 , 1.176)	0.767	-0.033	0.393
0.9	(0.906 , 0.909)	0.901	0.001	0.014	(0.861 , 0.868)	0.861	-0.039	0.057	(1.178 , 1.324)	0.887	-0.013	0.392
1.0	(0.987 , 0.989)	0.984	-0.016	0.022	(0.967 , 0.974)	0.966	-0.034	0.054	(1.457 , 1.643)	1.126	0.126	0.413
1.1	(1.086 , 1.089)	1.080	-0.020	0.024	(1.076 , 1.083)	1.078	-0.022	0.047	(1.391 , 1.569)	1.072	-0.028	0.394
1.2	(1.228 , 1.231)	1.222	0.022	0.026	(1.224 , 1.232)	1.224	0.024	0.048	(1.536 , 1.731)	1.192	-0.008	0.394
1.3	(1.312 , 1.315)	1.306	0.006	0.018	(1.308 , 1.317)	1.308	0.008	0.043	(1.635 , 1.849)	1.279	-0.021	0.395
1.4	(1.424 , 1.428)	1.418	0.018	0.023	(1.396 , 1.406)	1.396	-0.004	0.042	(1.857 , 2.120)	1.475	0.075	0.403
1.5	(1.520 , 1.523)	1.514	0.014	0.020	(1.560 , 1.572)	1.560	0.060	0.073	(2.019 , 2.310)	1.617	0.117	0.414
1.6	(1.588 , 1.592)	1.582	-0.018	0.023	(1.649 , 1.661)	1.648	0.048	0.063	(2.044 , 2.339)	1.639	0.039	0.399
1.7	(1.735 , 1.739)	1.728	0.028	0.032	(1.699 , 1.712)	1.698	-0.002	0.042	(2.117 , 2.427)	1.704	0.004	0.397
1.8	(1.833 , 1.837)	1.826	0.026	0.029	(1.851 , 1.865)	1.849	0.049	0.064	(2.218 , 2.544)	1.791	-0.009	0.398
1.9	(1.916 , 1.920)	1.908	0.008	0.016	(1.961 , 1.976)	1.958	0.058	0.071	(2.370 , 2.719)	1.922	0.022	0.400
2.0	(2.028 , 2.033)	2.020	0.020	0.025	(2.057 , 2.074)	2.054	0.054	0.068	(2.383 , 2.734)	1.933	-0.067	0.405

Chapter 5

Conclusion and Future Work

Our goal in this thesis was to propose statistical methodologies to provide accurate point and interval estimates of applied forces to brain tissues during a neurosurgery. To meet this goal, we studied different approaches and developed a few ones. These include the bootstrap technique, weighted least squares in linear regression, multi-level modeling and Bayesian approach to estimate parameters of multi-level models.

To be more specific, we first employed nonparametric bootstrap technique in statistical calibration, that provides accurate point and interval force estimation along x and y axes. We compared the results obtained from the bootstrapping approach with those reported in Zareinia et al. (2015) (which in this work was referred to as the *Naïve* method). Furthermore, we developed a multivariate extension of Eisenhart's methodology for calibration, that provides point and interval estimation of applied forces. Results indicated that the bootstrap approach provides more accurate estimation compared to the *Naïve* method.

Since the proposed bootstrap methodology fails to provide force estimation in z direction, we applied different weights as introduced in Table 4.5 in the model

to come up with more accurate estimation (WLS). After applying these weights, bootstrap results were improved as well, but still more accuracy was required.

As the next stage of this research, we introduced multi-level modeling that is applicable when data is structured in groups like this case that calibration data set is categorized in three different directions of x, y, and z. We realized that multi-level models yield a better estimation compared with WLS, especially when weights are applied in the models. Then, we assumed that some information about the distribution of data set is available and we employed Bayesian approach to estimate the coefficients in multi-level models. In multi-level models, results are reported under SEL, as well as LINEX loss functions.

We also employed the proposed bootstrap methodology in conjunction with multi-level models to provide interval estimation and enhance the accuracy of point estimation. The overall results suggest that estimations obtained using multi-level models are more accurate, however, Bayesian approach provides narrower confidence bounds and less Bias and RMSE in multi-level models. Another important conclusion is that using intercept in the model helps obtain more accurate results.

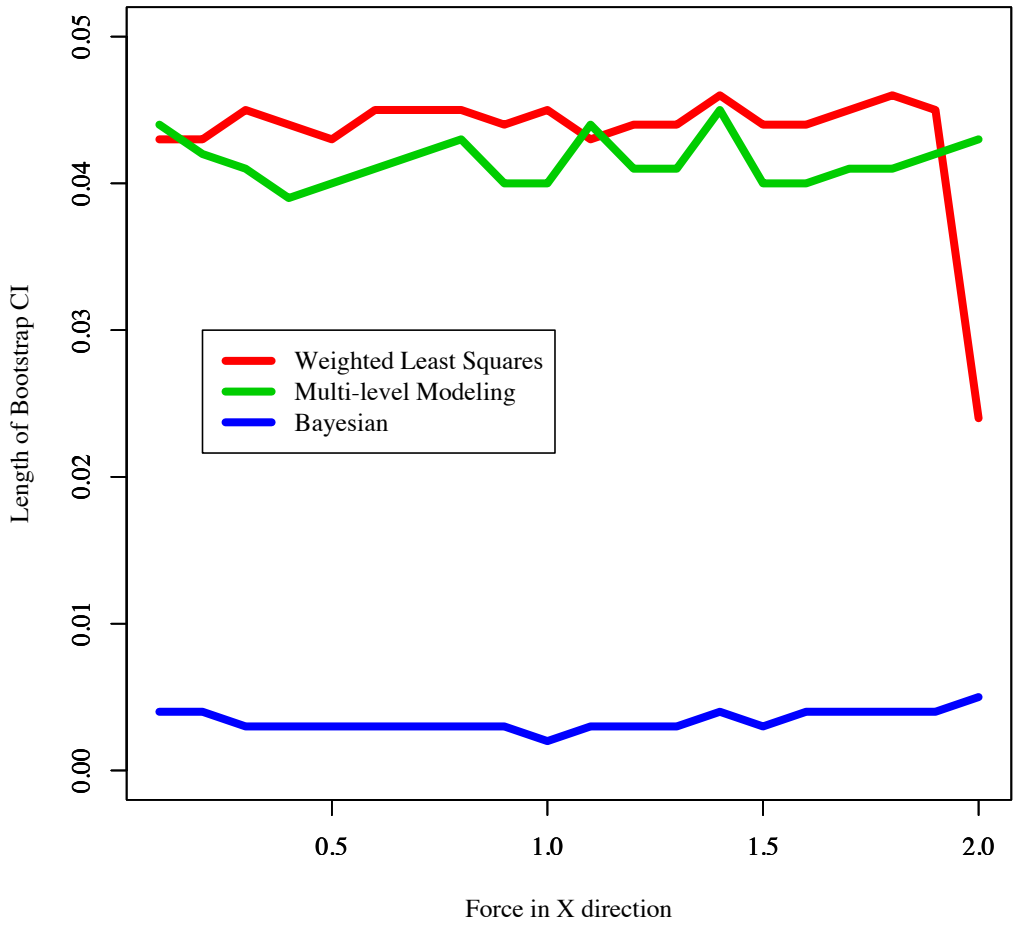


Figure 5.1: Comparison between length of intervals obtained by Weighted least squares, multi-level modeling, and Bayesian approaches, when force is applied in x direction.

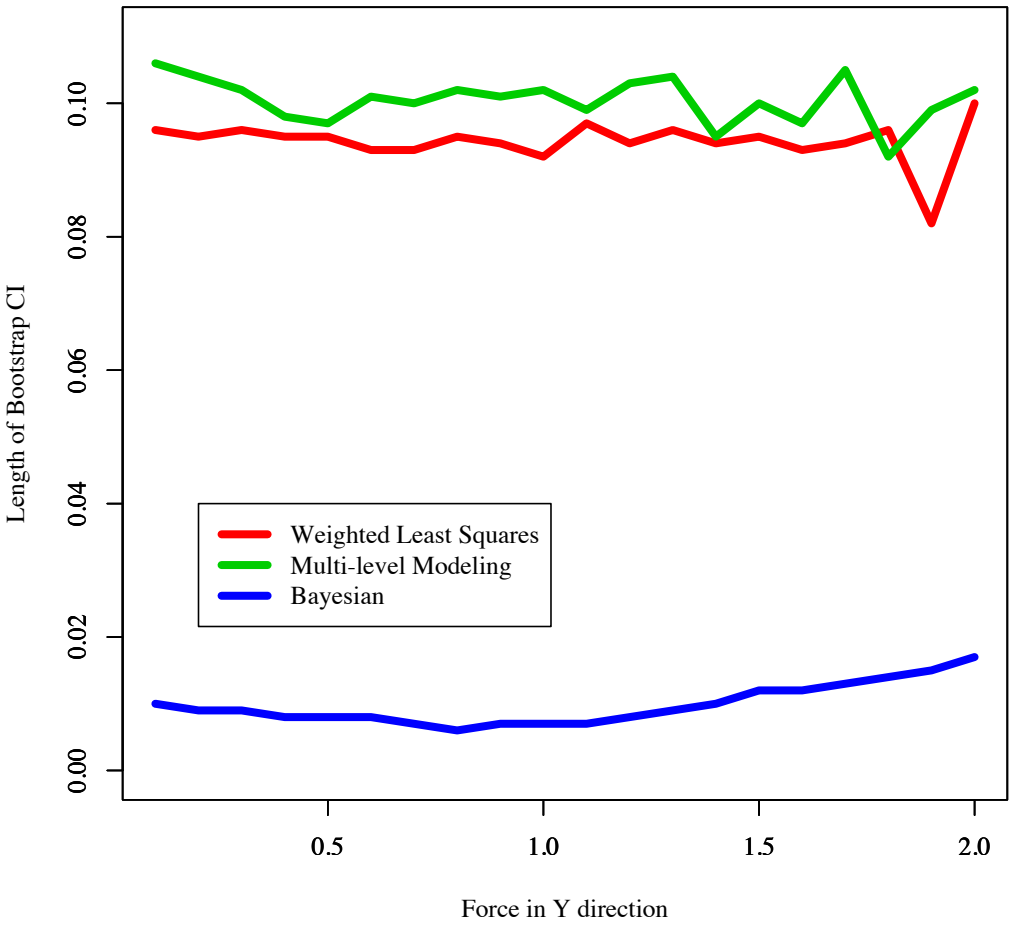


Figure 5.2: Comparison between length of intervals obtained by Weighted least squares, multi-level modeling, and Bayesian approaches, when force is applied in y direction.

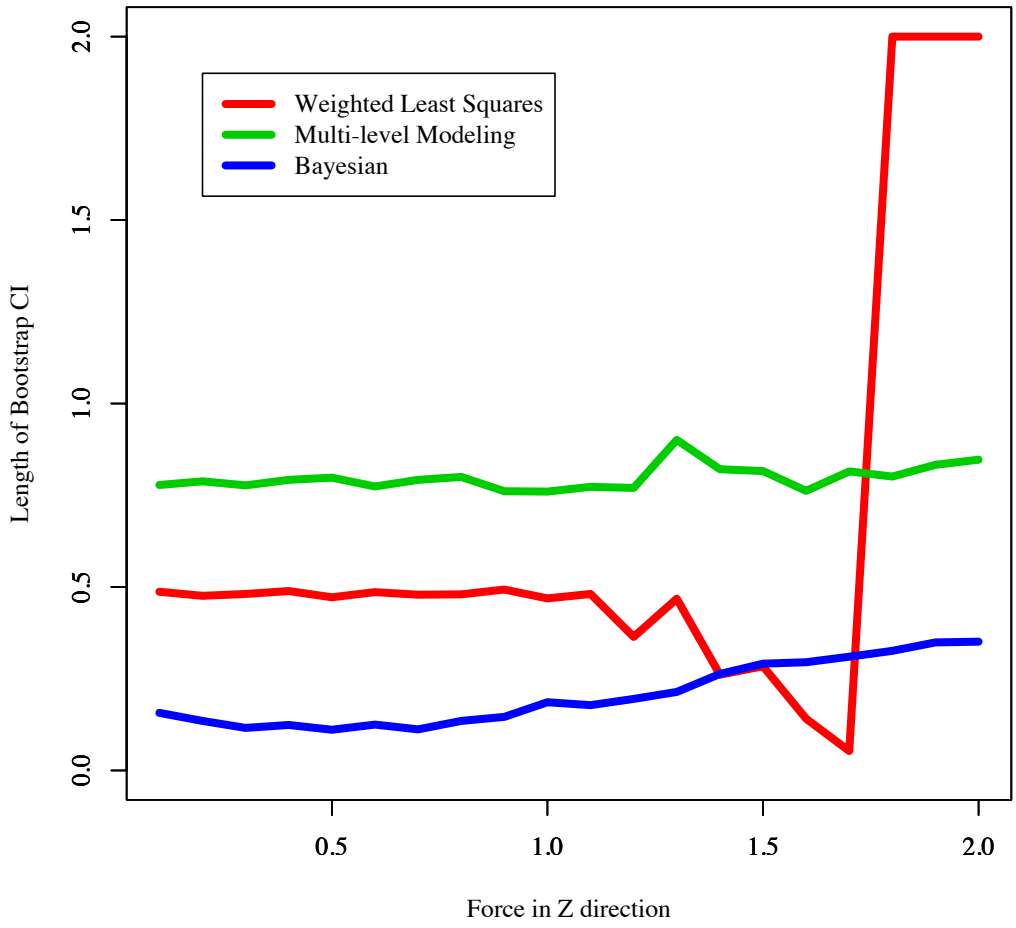


Figure 5.3: Comparison between length of intervals obtained by Weighted least squares, multi-level modeling, and Bayesian approaches, when force is applied in z direction.

Figures 5.1, 5.2, and 5.3 provide the comparison between length of intervals obtained by WLS, multi-level modeling, and Bayesian approaches. For multi-level models and Bayesian confidence intervals, we considered results obtained with models with intercept and under LINEX loss function. As can be seen from all three plots,

length of intervals obtained by Bayesian methodology is always far less than other methodologies.

There were several limitations associated with this study, that added to the challenge of proposing predictions with less bias. The most significant limitation was the fact that the output voltages read from the third pair of strain gauges, were not accurate enough. This issue is caused by non-optimal positioning of the strain gauges on the prongs of neurosurgical forceps.

One interesting direction for the future work is to consider measurement error in the model, specifically, multi-level models (Goldstein, Goldstein). One should consider the effect of measurement errors in the model, since there are errors in measuring the applied forces using the read voltages from the strain gauges. Another method that would be of interests to provide more accurate estimation is meta analysis (Brockwell and Gordon, 2001). Meta analysis provides pooled estimate of several separate studies that would be closer estimate to the true value.

Bibliography

Aitchison, J. and I. R. Dunsmore (1980). *Statistical prediction analysis*. CUP Archive. 35

Beretta, E., G. Ferrigno, and E. De Momi (2016). Nonlinear force feedback enhancement for cooperative robotic neurosurgery enforces virtual boundaries on cortex surface. *Journal of Medical Robotics Research 1*(02), 1650001. 12

Berkelman, P. J., L. L. Whitcomb, R. H. Taylor, and P. Jensen (2003). A miniature microsurgical instrument tip force sensor for enhanced force feedback during robot-assisted manipulation. *IEEE Transactions on Robotics and Automation 19*(5), 917–921. 11

Berkson, J. (1969). Estimation of a linear function for a calibration line; consideration of a recent proposal. *Technometrics 11*(4), 649–660. 26

Besalú, E. (2013). The connection between inverse and classical calibration. *Talanta 116*, 45–49. 21, 22

Brockwell, S. E. and I. R. Gordon (2001). A comparison of statistical methods for meta-analysis. *Statistics in medicine 20*(6), 825–840. 125

Brown, P. J. and R. Sundberg (1987). Confidence and conflict in multivariate calibration. *Journal of the Royal Statistical Society. Series B (Methodological)*, 46–57. 40

Carey, W. P. and S. S. Yee (1992). Calibration of nonlinear solid-state sensor arrays using multivariate regression techniques. *Sensors and Actuators B: Chemical 9*(2), 113–122. 21

Casella, G. and R. L. Berger (2002). *Statistical inference*, Volume 2. Duxbury Pacific Grove, CA. 28

Clark, A. D., D. G. Barone, N. Candy, M. Guilfoyle, K. Budohoski, R. Hoffmann, T. Santarius, R. Kirollos, and R. A. Trivedi (2017). The effect of 3-dimensional simulation on neurosurgical skill acquisition and surgical performance: A review of the literature. *Journal of Surgical Education*. 2

Dally, J. W., W. Riley, and K. G. McConnell (1983). *Instrumentation for engineering measurements*. 17

De Boeck, P., M. Bakker, R. Zwitser, M. Nivard, A. Hofman, F. Tuerlinckx, I. Partchev, et al. (2011). The estimation of item response models with the lmer function from the lme4 package in r. *Journal of Statistical Software 39*(12), 1–28. 95

Draper, N. R. and H. Smith (2014). *Applied regression analysis*. John Wiley & Sons. 85, 86

Dunsmore, I. (1968). A bayesian approach to calibration. *Journal of the Royal Statistical Society. Series B (Methodological) 30*(2), 396–405. 31

Efron, B. and R. Tibshirani (1986). Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical science*, 54–75. 39

Efron, B. and R. J. Tibshirani (1994). *An introduction to the bootstrap*. CRC press.

Eisenhart, C. (1939). The interpretation of certain regression methods and their use in biological and industrial research. *The Annals of Mathematical Statistics 10*(2), 162–186. 23, 41, 49, 71

Fastrack (2017). Tracking device. 14

Gabry, J. and B. Goodrich (2016). rstanarm: Bayesian applied regression modeling via stan. r package version 2.10. 0. 109

Gan, L. S., K. Zareinia, S. Lama, Y. Maddahi, F. W. Yang, and G. R. Sutherland (2015). Quantification of forces during a neurosurgical procedure: A pilot study. *World neurosurgery 84*(2), 537–548. 2, 12

Gelman, A. and J. Hill (2006). *Data analysis using regression and multilevel/hierarchical models*. Cambridge university press. 88

Goldstein, H. Measurement errors in multilevel models. *Multilevel Statistical Models, 4th Edition*, 267–284. 125

Hawkins-Daarud, A., S. Prudhomme, K. G. van der Zee, and J. T. Oden (2013). Bayesian calibration, validation, and uncertainty quantification of diffuse interface models of tumor growth. *Journal of mathematical biology 67*(6-7), 1457–1485. 22

Hickey, G. L. (2006). *The linear calibration problem: a Bayesian analysis*. Ph. D. thesis, PhD dissertation, University of Durham. 1–148. 33

Hoadley, B. (1970). A bayesian look at inverse linear regression. *Journal of the American Statistical Association 65*(329), 356–369. 31, 32, 35

Hoffmann, K. (1974). *Applying the wheatstone bridge circuit*. HBM. 16, 19

Jones, G. and D. M. Rocke (1999). Bootstrapping in controlled calibration experiments. *Technometrics 41*(3), 224–233. 40, 44

Krutchkoff, R. (1967). Classical and inverse regression methods of calibration. *Technometrics 9*(3), 425–439. 25, 26, 40

Lwin, T. and J. Maritz (1980). A note on the problem of statistical calibration. *Applied Statistics*, 135–141. 22

Maddahi, Y., L. S. Gan, K. Zareinia, S. Lama, N. Sepehri, and G. R. Sutherland (2015). Quantifying workspace and forces of surgical dissection during robot-assisted neurosurgery. *The International Journal of Medical Robotics and Computer Assisted Surgery*. 13

Maddahi, Y., J. Huang, J. Huang, L. S. Gan, H. Hoshyarmanesh, K. Zareinia, and G. R. Sutherland (2016). Real-time measurement of tool-tissue interaction forces in neurosurgery: Quantification and analysis. In *Advanced Intelligent Mechatronics (AIM), 2016 IEEE International Conference on*, pp. 1405–1410. IEEE. 1, 12

Marcus, H. J., C. J. Payne, A. Kailaya-Vasa, S. Griffiths, J. Clark, G.-Z. Yang, A. Darzi, and D. Nandi (2016). A "smart" force-limiting instrument for microsurgery: Laboratory and in vivo validation. *PloS one 11*(9), e0162232. 3

Marcus, H. J., K. Zareinia, L. S. Gan, F. W. Yang, S. Lama, G.-Z. Yang, and G. R. Sutherland (2014). Forces exerted during microneurosurgery: a cadaver study. *The International Journal of Medical Robotics and Computer Assisted Surgery 10*(2), 251–256. 12

neuroArm (1997). Latex — neuroarm,. [Online; accessed 07-December-2016]. 4, 12

Ni, W., L. Nørgaard, and M. Mørup (2014). Non-linear calibration models for near infrared spectroscopy. *Analytica chimica acta 813*, 1–14. 21

Osborne, C. (1991). Statistical calibration: a review. *International Statistical Review/Revue Internationale de Statistique*, 309–336. 21, 31, 35

Parker, P. A., V. G. Geoffrey, S. R. Wilson, J. L. Szarka III, and N. G. Johnson (2010). The prediction properties of inverse and reverse regression for the simple linear calibration problem. 22, 28

Payne, C. J., H. J. Marcus, and G.-Z. Yang (2015). A smart haptic hand-held device for neurosurgical microdissection. *Annals of biomedical engineering 43*(9), 2185–2195. 66

Payne, C. J. and G.-Z. Yang (2014). Hand-held medical robots. *Annals of biomedical engineering 42*(8), 1594–1605. 66

Potma, T. (1967). *Strain gauges: Theory and application.* Centrex Publishing Company. 16

Reznick, R. K. and H. MacRae (2006). Teaching surgical skills—changes in the wind. *New England Journal of Medicine 355*(25), 2664–2669. 2

Rosseau, G., J. Bailes, R. del Maestro, A. Cabral, N. Choudhury, O. Comas, P. Debergue, G. De Luca, J. Hovdebo, D. Jiang, et al. (2013). The development of a virtual simulator for training neurosurgeons to perform and perfect *Neurosurgery 73*, S85–S93. 3

Rueda, M., I. Sánchez-Borrego, A. Arcos, and S. Martínez (2010). Model-calibration estimation of the distribution function using nonparametric regression. *Metrika 71*(1), 33–44. 22

Ruppert, D., M. P. Wand, and R. J. Carroll (2003). *Semiparametric regression*. Number 12. Cambridge university press. 88, 93, 94, 104, 105, 106, 107, 108

Schwartz, L. M. (1977). Nonlinear calibration. *Analytical chemistry 49*(13), 2062–2069. 21

Searle, S. R., G. Casella, and C. E. McCulloch (2009). *Variance components*, Volume 391. John Wiley & Sons. 94

Shigley, J. E., C. R. Mischke, and R. G. Budynas (2004). *Mechanical engineering design*. McGraw-Hill,. 16

Tanimoto, M., F. Arai, T. Fukuda, H. Iwata, K. Itoigawa, Y. Gotoh, M. Hashimoto, and M. Negoro (1998). Micro force sensor for intravascular neurosurgery and in vivo experiment. In *Micro Electro Mechanical Systems, 1998. MEMS 98. Proceedings., The Eleventh Annual International Workshop on*, pp. 504–509. IEEE. 3

Taylor, R. H., J. Funda, B. Eldridge, S. Gomory, K. Gruben, D. LaRose, M. Talamini,

L. Kavoussi, and J. Anderson (1995). A telerobotic assistant for laparoscopic surgery. *IEEE Engineering in Medicine and Biology Magazine 14*(3), 279–288. 11

Üneri, A., M. A. Balicki, J. Handa, P. Gehlbach, R. H. Taylor, and I. Iordachita (2010). New steady-hand eye robot with micro-force sensing for vitreoretinal surgery. In *Biomedical Robotics and Biomechatronics (BioRob), 2010 3rd IEEE RAS and EMBS International Conference on*, pp. 814–819. IEEE. 11

Wang, K., K. Grant, Q. Sun, L. Gan, K. Zareinia, S. Lama, G. Sutherland, and Y. Maddahi (2015). Development of an instrumented surgical setup for quantifying displacement and force in surgical dissection. In *ASME 2015 International Mechanical Engineering Congress and Exposition*, pp. V003T03A032–V003T03A032. American Society of Mechanical Engineers. 13

Williams, E. (1969). A note on regression methods in calibration. *Technometrics 11*(1), 189–192. 25

Window, A., G. Holister, et al. (1982). *Strain gauge technology*. Applied science Barking. 16

Yaşargil, M. G. (2013). *Microsurgery: applied to neurosurgery*. Elsevier. 3

Yin, X., S. Guo, N. Xiao, T. Tamiya, H. Hirata, and H. Ishihara (2016). Safety operation consciousness realization of a mr fluids-based novel haptic interface for teleoperated catheter minimally invasive neurosurgery. *IEEE/ASME Transactions on Mechatronics 21*(2), 1043–1054. 12

Zareinia, K., Y. Maddahi, L. S. Gan, A. Ghasemoonia, S. Lama, T. Sugiyama, F. W.

Yang, and G. R. Sutherland (2015). A force-sensing bipolar forceps to quantify tool-tissue interaction forces in microsurgery. 2, 4, 9, 13, 15, 16, 18, 19, 20, 38, 39, 42, 54, 71, 120