Linköping University | Department of Computer and Information Science
Master's thesis, 30 ECTS | Statistics and Machine Learning
2020 | LIU-IDA/STAT-A--20/028--SE

Improving predictions of muscle mass from an impedance device

Cross-calibration of bioelectrical impedance analysis and dual X-ray absorbiometry using a Bayesian approach

Alexander Karlsson

Supervisor : Bertil Wegmann
Examiner : Annika Tillander

External supervisor : Ola Wallengren

LINKÖPING UNIVERSITY

Linköpings universitet
SE–581 83 Linköping
+46 13 28 10 00 , www.liu.se

Upphovsrätt

Copyright

Abstract

Assessment of body composition by means of quantitative methods is an important aspect for practitioners of medicine when treating patients, both in terms of longitudinal monitoring to see eventual progression but also in terms of diagnosis with respect to certain clinical parameters. Especially interesting is the amount of appendicular lean soft tissue (ALST), as this particular quantity can be compared to certain clinical cut-offs for classification of disease. There exist many options for assessment of body composition, yet few have as many positives as bioelectrical impedance analysis (BIA). BIA is an attractive option for many practitioners due to the simplicity of the method. However, using BIA for assessment of body composition has it's downsides; unreliable predictions. BIA has been reported to be especially unreliable when applied on subjects that are classified as underweight or obese via body mass index (BMI).

The BIA device outputs several electrical variables in different parts of the body. These electrical variables are highly influenced by anthropometric variables and are adjusted accordingly to reduce variability. Several transformations that are common in similar studies are applied and explained. Additionally, this thesis proposes three novel suggestions of electrical variables for future research. The first is a weight-adjusted angle of two electrical variables, the second is an entropy-based variable of angles and the third is a multivariate distance of the second. All three variables are used in the analysis of this thesis. Further research might reveal a better understanding of how these variables relate to body composition in theory.

This thesis aims to improve the predictions of BIA via by regressing output of electrical variables to another – more reliable – technique; dual x-ray absorbiometry (DXA). Furthermore, this thesis aims to investigate if the selected models can generalize well over all regions of BMI. The methods used are Bayesian hierarchical linear regression and variational Bayesian neural networks. Fitting of the hierarchical models are obtained via Markov chain Monte Carlo (MCMC) and evaluated using the widely applicable information criterion (WAIC) as well as graphical checks of the posterior predictive distribution. Fitting of Bayesian neural networks are obtained using back-propagation of a two-component loss function consisting of a complexity cost and a likelihood cost and evaluated via graphical checks. Prior elicitation and prior sensitivity analysis is performed for the hierarchical linear models and two regularizing prior configurations are tested on the neural networks.

Results show that improvements have been made for both methods, where the neural networks are performing best. In both the linear models and neural networks, a regularizing Laplace prior gave the best results. Graphical checks shows that both methods have good generalizing ability, yet concerns can be raised over subjects with very high ALST. In conclusion, both methods used are adequately able to improve predictions on ALST and generalize well over different ranges of BMI.

Acknowledgments

First of, I would like to thank my supervisor for this project, Bertil Wegmann. You always gave me helpful insights and patiently listened to my ideas, no matter how crazy they were. Our discussions were not only fun, but always resulted in improvements and your support gave me confidence when I was in doubt.

Gratitude is also extended to Ola Wallengren, the external commissioner of this thesis. You always replied quickly to my many questions and always provided me a place to sit when visiting Sahlgrenska. Your feedback was very helpful for my understanding of the subject and is very much appreciated.

I would like to thank my opponent Saewon Jun for a thorough read through of the thesis and insightful comments. Your input most definitely made this thesis better and more comprehensible, for that I am grateful.

Last but not least, I would like to thank my sister Andrea Jareteg and my nephew Ebbe Jareteg Petersson. You gave me a place to live when visiting Gothenburg and took my mind of the thesis even during difficult times. You are both very special people in my life.

Glossary

- **Body composition** - An alternative, more detailed description of the human body than e.g. weight alone.
- **FM** - Fat-Mass. The amount of fat mass.
- **FFM** - Fat-Free Mass. The amount of non-fat mass.
- **BM** - Bone-Mass. The amount of bone mass.
- **MM** - Muscle-Mass. The amount of muscle mass.
- **LST** - Lean Soft Tissue. Fat-free, bone-free mass.
- **ALST** - Appendicular Lean Soft Tissue. Fat-free, bone-free mass in arms and legs.
- **ECW** - Extra-Cellular Water. Amount of water that exist outside the cell membrane.
- **ICW** - Intra-Cellular Water. Amount of water that exist inside the cell membrane.
- **TBW** - Total-Body Water. Sum of ICW and ECW.
- **Body compartment** - A single compartment of the human body, e.g. FFM, FM or similar.
- **Two-compartment model** - Partition of the human body into FM and FFM.
- **Three-compartment model** - Partition of the human body into FM, BM and LST.
- **BIA** - Bioelectrical Impedance Analysis. Technique for body composition assessment, to a degree based on the electrical properties of different compartments in the body.
- **BIA device** - A device that predicts body composition on the basis of BIA.
- **Impedance** - Obstruction of flow to an electrical current, denoted Z. Consist of two components (see below).
- **Resistance** - Component of impedance, denoted R.
- **Reactance** - Component of impedance, denoted Xc.
- **Impedance variables / electrical variables** - Measurements based on the injected current, e.g. resistance, reactance or any transformation of the two.
- **DXA** - Dual X-ray Absorbiometry. Technique for body composition assessment.
- **DXA device** - A device that measures body composition based on two x-ray beams with different energy levels.
- **BIA prediction** - Prediction on body compartment(s) based on a built-in regression algorithm of a BIA device.
- **BIA equation** - An alternative prediction model to body compartment(s) to the BIA prediction. Typically a linear regression of some shape or form. Variables included are up to the user.

Contents

Abstract iii

Acknowledgments v

Glossary vii

Contents viii

List of Figures x

List of Tables xii

1 Introduction 1
- 1.1 Commissioner . . . 1
- 1.2 Background . . . 1
- 1.3 Motivation . . . 2
- 1.4 Aim . . . 2
- 1.5 Related work . . . 3
- 1.6 Ethical aspects . . . 4
- 1.7 Delimitations . . . 4

2 Theory 5
- 2.1 Fundamentals of bioelectrical impedance analysis . . . 5
- 2.2 Shortcomings of bioelectrical impedance analysis . . . 6

3 Data 7
- 3.1 Data source . . . 7
- 3.2 Data description . . . 7
- 3.3 Data cleaning . . . 9
- 3.4 Data wrangling . . . 10
- 3.5 Feature engineering . . . 10
- 3.6 Data partition . . . 16

4 Method 19
- 4.1 Bayesian inference . . . 19
- 4.2 Bayesian hierarchical linear regression . . . 20
  - 4.2.1 Assessing need for hierarchical modelling . . . 21
  - 4.2.2 Model evaluation . . . 21
  - 4.2.3 Markov chain Monte Carlo . . . 22
    - 4.2.3.1 Hamiltonian Monte Carlo . . . 22
    - 4.2.3.2 Chain diagnostics . . . 24
    - 4.2.3.3 Efficient re-parameterizations . . . 25
  - 4.2.4 Stan . . . 25

4.3 Bayesian neural networks . . . 26
4.3.1 Bayes By Backprop . . . 26
4.3.2 Stochastic optimization . . . 28
4.3.3 Activation function . . . 28
4.3.4 Optimizers . . . 28
4.3.5 Model evaluation . . . 29
4.3.6 PyTorch . . . 30

5 Results **31**
5.1 Bayesian hierarchical regression . . . 31
5.1.1 Model specifications . . . 31
5.1.2 Parameter estimates . . . 33
5.1.3 Performance on training data . . . 33
5.1.4 Performance on validation data . . . 36
5.1.5 Model selection . . . 38
5.2 Bayesian neural networks . . . 39
5.2.1 Fit metrics . . . 40
5.2.2 Performance on training data . . . 41
5.2.3 Performance on validation data . . . 42
5.2.4 Model selection . . . 44
5.3 Results summary . . . 44

6 Discussion **45**
6.1 Data . . . 45
6.2 Results . . . 46
6.3 Method . . . 47
6.4 The work in a wider context . . . 48
6.5 Future research . . . 48

7 Conclusion **49**

Bibliography **51**

A Appendix A – Model diagnostic plots **55**

B Appendix B – Code **61**

List of Figures

3.1 Number of patient visits per year. 7
3.2 Issues with BIA predictions. The black line denotes identical predictions and red lines denote ± 1 kg of identical predictions from BIA and DXA. Left: BIA predictions have a tendency to overestimate muscle mass. Right: Tendencies towards overestimation from BIA compared to DXA increases with increasing BMI. 11
3.3 Discrepancies of *Resistance* (left) and *Reactance* (right) in the left leg. Black vertical line separates measurements prior and after 2015. 12
3.4 Functional form between electrical variables and target variable. Green: 5 kHz, orange: 250 kHz. Top row: raw electrical variables, bottom row: electrical variables adjusted with height (left) and weight (right). 13
3.5 Functional form between anthropometric variables and target, colored by gender. Orange: males, green: females. 14

5.1 Bayesian R^2, MSE and ICC for all hierarchical linear models. Left: regularizing model, center: weakly informative model, right elicitated model. 34
5.2 Importance ratios plotted over BMI for all hierarchical linear models. Green points correspond to females, orange points correspond to males. Left: regularizing model, center: weakly informative model, right: elicitated model. 35
5.3 Difference $\tilde{y} - y$ plotted over BMI for all hierarchical linear models. Green points correspond to females, orange points correspond to males. Left: regularizing model, center: weakly informative model, right: elicitated model. 35
5.4 Y-vs-Y plots for posterior predictive means on training data. Green dots represent females and orange dots represent males. Left: regularizing model, center: weakly informative model, right: elicitated model. 36
5.5 MSE for validation data. Left: Regularizing model, center: weakly informative model, right: elicitated model. 36
5.6 Difference $\tilde{y} - y$ plotted over BMI for all hierarchical linear models on validation data. Green points correspond to females, orange points correspond to males. Left: regularizing model, center: weakly informative model, right: elicitated model. 37
5.7 Y-vs-Y plots for posterior predictive means on validation data. Light green dots represent females, dark green dots represent males. Left: Laplace model, center: weakly informative model, right: elicitated model. 37
5.8 Posterior predictive distributions for a selection of subjects with different BMI. From left to right: underweight, normal and obese. Top row: Male subjects, bottom row: female subjects. Gray: regularizing model, orange: weakly informative model, green: elicitated model. 38
5.9 Test set Y-vs-Y (left) and test set MSE (right) for the regularized model. 39
5.10 Variational free energy for the Gaussian NN (orange) and the Laplacean NN (green). Left: training data, right: validation data. 40
5.11 Posterior means and standard deviations for the Gaussian NN (orange) and Laplacean NN (green). Left: means, right: standard deviation. 41

5.12 Y-vs-Y plots for posterior predictive means on training data. Left: Laplacean NN, right: Gaussian NN. 41
5.13 Bayesian MSE (left) and R^2 (right) for the two neural networks on training data. Green: Laplacean NN, orange: Gaussian NN. 42
5.14 Y-vs-Y plots data for posterior predictive means on validation data. Left: Laplacean NN, right: Gaussian NN. 42
5.15 Difference $\tilde{y} - y$ plotted over BMI for Laplacean NN (left) and Gaussian NN (right). 43
5.16 Validation set MSE for Laplacean NN (left) and Gaussian NN (right). 43
5.17 Y-vs-Y-plot (left) and MSE (right) for the Laplacean NN on test data. 44

A.1 Traceplots for relevant parameters of the regularizing hierarchical linear model. . . 55
A.2 Effective samples and Gelman-Rubin diagnostic for the regularizing hierarchical linear model. 56
A.3 Traceplots for relevant parameters of the weakly informative hierarchical linear model. 56
A.4 Effective samples and Gelman-Rubin diagnostic for the weakly informative hierarchical linear model. 57
A.5 Traceplots for relevant parameters of the elicitated hierarchical linear model. . . . 57
A.6 Effective samples and Gelman-Rubin diagnostic for the elicitated hierarchical linear model. 58
A.7 Complexity cost over epochs for both neural networks, green: Laplacean NN, orange: Gaussian NN. 58
A.8 Likelihood cost over epochs for both neural networks, green: Laplacean NN, orange: Gaussian NN. 59

List of Tables

3.1 Variable overview, prior to data manipulation and data cleaning. 8
3.2 Data reduction; number of measurements concerned with a short motivation. . . . 9
3.3 Variable overview, after data manipulation and data cleaning. 15
3.4 Multi-level structure of data. n_v is number of unique patients with v visits in total, $v \cdot n_v$ is the number of measurements for v visits. 16
3.5 Selection of subjects with varying BMI from validation data. 16

5.1 Parameter estimates for hierarchical linear models. 33
5.2 Estimates on out-of-sample predictive abilities of hierarchical linear models. 34
5.3 MSE for the different data sets. 44

1 Introduction

1.1 Commissioner

The commissioner of this thesis is the Clinical Nutrition Unit at Sahlgrenska hospital in Gothenburg, which comprises of dietitians and physicians. The dietitian's role at the Clinical Cutrition Unit is to provide medical nutritional therapy to patients who have nutrition related deceases. Dietitians also take precautionary measures for patients who are at risk with respect to certain clinical parameters. In order to do so, an accurate assessment of body composition may be required.

1.2 Background

Body composition is a way to divide the human body into components which collectively sum up to the total body weight. One way to decompose the human body into components is to consider the body as a sum of fat-mass (FM), bone mass (BM) and lean soft tissue (LST), where lean soft tissue is defined as fat-free, bone-free mass. These components combined constitute a more detailed description of the body than weight or body mass index (BMI), which does not discriminate the components. Measurements of body composition in healthcare enables practitioners to detect indications of disease, detect signs of aging, monitor progression of body composition and give advice on nutrition etc [35]. By measuring body composition using some quantitative technique of choice, a distinct depiction of the different body compartments can be obtained – as opposed to what a visual inspection would yield.

The Clinical Nutrition Unit at Sahlgrenska utilize two techniques for assessment of body composition in patients; bioelectrical impedance analysis (BIA) and dual x-ray absorptiometry (DXA). BIA devices operate by sending electrical currents through the human body via injectors, where a sensor measures the current received in a distant part of the body. A measurement of how well the body is working as a conductor of electricity is telling of what medium the current flows through, i.e. the impedance reveals some indication of body composition [18]. DXA devices operate by sending two beams of different energy levels through the body, where the medium that the beams flow through absorb different amounts of energy. The aforementioned amounts of energy allow for differentiation of different com-

ponents, in an elaborate scheme that is not explained in this thesis.

The BIA device at the Clinical Nutrition Unit outputs several quantities; (1) predictions on body composition in legs, arms and trunk that are based on a multiple regression algorithm [34] and (2) impedance variables that reflect electrical properties of the body. The exact built-in regression algorithm that generates predictions in the BIA device is not disclosed by the device manufacturer. Questions have been raised over the accuracy in measurements of the particular device, and extra caution should be taken for patients with extreme BMI [35], especially for patients with BMI exceeding 34 kg/m^2 [23]. A common approach among researchers aiming to provide more robust outputs is to regress body composition predictions from a reference method (e.g. DXA) to various impedance variables generated from the BIA device in combination with variables such as age, gender and height [18] [1]. The aforementioned BIA equations are generally considered to be sample-specific and thus not advised to use for external samples unless properly validated at such [1].

Evolution of muscle function and muscle mass may be divided into three phases in life with different goals regarding muscle mass; (1) early life where muscle mass peaks, (2) adult life where the aim is to maintain muscle mass and (3) older life where focus is to minimize loss of muscle mass. Low muscle mass is a common factor amongst elderly and malnutritioned patients, and is associated with a higher risk of disease and a deteriorated response to treatment. Formally, the muscle deterioration of muscle mass and muscle function is a disease called Sarcopenia. Sarcopenia is common at high ages and costs society as a whole in terms of both suffering and costs. [6]

Appendicular lean soft tissue (ALST) is a good indicator of total body muscle mass and is one way to quantify muscle quantity [6]. Clinical assessment of ALST are part of confirmation of Sarcopenia and part of the follow-up procedures that is evaluated in patients that are considered to be at risk of Sarcopenia. The Clinical Nutrition Unit wishes to further use the BIA device for assessment of body composition, due to the benefits that BIA offers in terms of prize and simplicity. However, increased precision in the output is sought, as BIA predictions on ALST are not of sufficient standard.

1.3 Motivation

The two body composition techniques provide a quantification of body contents that are not possible to visually detect and should ideally be as precise as possible. Precision in BIA devices are low in comparison to DXA and other reference methods and give especially unstable body composition outputs for patients with extreme BMI. Furthermore, BIA predictions has previously been proved to overestimate muscle mass [35] [27] while underestimating fat-mass. However, BIA devices are comparatively cheap, not restricted to the clinical environment, non-invasive and easy to use, and are thus an attractive option. Further usage of BIA, with a refined output that matches the precision of DXA by means of a predictive equation would give the best of two worlds; a cheap and accurate way to predict body composition.

1.4 Aim

There is a perceived problem at the Clinical Nutrition Unit when BIA devices are used to predict body composition in general due to lack of preciseness in the predictions. ALST is particularly of interest as there exist clear cut-off points for diagnosis of Sarcopenia. A solution to the problem of non-precise outputs from BIA is to use the electrical measurements that the device outputs to create more robust predictions for patients in all ranges of BMI. Thus, the aim can be formulated with the following research questions:

1. Can predictions on ALST from the impedance device be improved upon using a statistical model?
2. Can predictions from the fitted models generalize over all ranges of BMI?

1.5 Related work

Previous work within the field of evaluating accuracy of predictions generated by BIA devices cross-referenced to an alternative "golden standard" method is not so common. Tognon et. al. in [35] compared predictions on FM and ALST in an elderly Swedish sample and concluded that a BIA device is not a valid tool for measurement of body composition due to overestimation of LST and underestimation of FM. The authors argue for a full disclosure of the full built-in algorithm that generates predictions on body composition from the device manufacturer, in order to reliably evaluate any predictions provided. Similar conclusions are provided by Kyle et. al. in [23], whom report that BIA predictions device is to be interpreted cautiously, especially for subjects with abnormal hydration status or extreme BMI. As such, the advice is not to use BIA devices for routine assessment of body composition. The authors do however argue that a generated equation (model) based on the device output may provide reliable predictions, yet with a caution for subject with anthropometric abnormalities in e.g. BMI and body shape.

In contrast to evaluation of the raw predictions generated by BIA devices, the studies that generate an equation based on electrical outputs of the device in combination with one or several other variables are in abundance. Bosy-Westphal et. al. in [4] reported that linear regression with variables selected from a step-wise procedure provide predictions on skeletal muscle-mass that are more reliable than those of DXA, when magnetic resonance imaging (MRI) was used as the reference method (ground truth). The authors used a Caucasian sample for the development of their equation and validated on a multi-ethnic sample consisting of roughly equal proportions of Caucasians, Asians, Afro-Americans and Hispanics. Some differences were reported between the ethnic groups as measured by the golden standard method (MRI), which was reflected by the equations generated. In another study, Bosy-Westphal et. al. [5] conclude that a BIA device may be used to create an equation that can adequately predict two-component body composition, yet with caution towards extreme ranges of BMI, or abnormal states of hydration.

As a testament to the popularity in generating BIA equations, several reviews of such are available. Beaudart et. al. in [1] provide a systematic review of 25 equations independently created by a variety of researchers. The review aims to give "clinicians and researchers the opportunity to verify the existence of a prediction equation when using a BIA device for estimating muscle mass". The review also provides all equations for predictions of muscle mass, what model were used, how variable selection was performed (if any), what frequencies was used and model performance in terms of mean squared error and coefficient of variation. From the review, it becomes apparent that researchers are relying on variable selection via the model itself to a high extent, and that regression equations on average explain approximately 90 % of the variation in the respective regression models.

Although the interest in creating new BIA equations is seemingly an interesting topic for many researchers, nearly all models are linear. However, some more advanced or more elaborate (machine learning) approaches are available. Kuen-Chang et. al. in [16] applied a step-wise variable selection in a multiple linear regression to obtain a baseline performance. The selected variables were then used in a single-layer neural network and a comparison was performed. Results showed that the neural network outperformed the linear regression, and the authors concluded that a neural network is more suitable for estimation of FFM. Lu, Hahn

and Zhang in [24] predict pixel-level body composition obtained by three-dimensional body scans using an elaborate modelling scheme involving Bayesian networks. When considering time-series, Tronstad and Strand-Amundsen in [36] applied both a single-layer artificial neural network and a multi-layer long-short term memory neural network to monitor changes in a biological process over time.

Hinton and van Camp in [14] introduced a theoretical foundation on minimum-description-length (MDL) to impose regularization on the weights of a neural network via Kullback-Leibler (KL) divergence. Graves [13] built upon this framework to formalize a variational inference (VI) approach for Bayesian neural networks, formalising a cost function that is splitted into a KL cost and a likelihood cost. Subsequently, Blundell et. al. in [2] present a framework based on VI that learns a probability distribution over the weights of a neural network using back propagation, referred to as *Bayes By Backprop*. By continuously evaluating an unbiased objective function with samples from the variational posterior, the KL divergence need not be evaluated in closed form, allowing for flexibility in prior and posterior selection as well as speed. The proposed method was tested in a regression setting, where predictions in sparse data regions resulted in uncertainty, as opposed to conventional (frequentist) methods.

1.6 Ethical aspects

The data obtained for this thesis contain measurements on weight, height, BMI etc. as well as information about how many visits to the Clinical Nutrition Unit a given subject has. This may be regarded as sensitive information for the subjects involved. However, the only identification available about subjects are transformed such that the author is not able to obtain for instance city of residence or personal code numbers of the subjects involved. Thus, no sensitive information can be spread on the basis of the data obtained.

1.7 Delimitations

Body composition was measured by DXA (Lunar Prodigy, Scanex, Sweden, software version 8.70.005) and BIA (MC-180MA, Tanita, Japan). The output given by these two devices allows for modelling of lean soft tissue in individual components of the body, yet the delimitation of modelling the sum of LST (i.e. ALST) was decided. The delimitations described essentially means that a one-component output is modelled, instead of a four-component output, as there are four limbs in the human body.

2 Theory

2.1 Fundamentals of bioelectrical impedance analysis

Impedance (Z) is a measurement on the obstruction to alternating electrical currents (AC) flow, which in bioelectrical impedance corresponds to the obstruction of an alternating electrical current caused by the human body. Notably, impedance is a complex equation; $Z \in \mathbb{C}$. An analogy to the Cartesian plane is a valid representation of the two components of impedance; *resistance* (R) and *reactance* (Xc).

$$Z_{(\Omega)} = R_{(\Omega)} + i \cdot Xc_{(\Omega)} \tag{2.1}$$

The two components R and Xc in equation 2.1 are measurements of two different kinds of resistance and have physiological interpretations. Resistance is regulated by the conductive ability of the biological tissue (fluid level) which the current flows through. Reactance is regulated by the capacitive ability of the cell membrane (healthiness of cell membrane). The two components are mathematically defined below. [18]

$$R_{(\Omega)} = \rho_{(\Omega \cdot m)} \cdot \frac{L_{(m)}}{A_{(m^2)}} \tag{2.2}$$

$$Xc_{(\Omega)} = \frac{1}{2 \cdot \pi \cdot f_{(kHz)} \cdot C_{(Farad)}} \tag{2.3}$$

Above, L is length of the cylinder, A is the cylinder cross-sectional area, ρ is a resistivity constant, C denotes capacitance and f denotes the frequency applied. In electrical circuits, some amount of electrical charge can be stored using a capacitor, giving rise to a lag between current and voltage. In the human body, capacitance (C) is caused by the cell membrane holding onto the electrical charge applied from the injector, causing a lag in the voltage applied and the current received by the sensor [18]. The lag between current applied and current measured is referred to as phase shift [18], or an angle in the Cartesian plane;

$$\varphi_{(\circ)} = tan^{-1}\Big(\frac{Xc_{(\Omega)}}{R_{(\Omega)}}\Big) \cdot \Big(\frac{180}{\pi}\Big), \tag{2.4}$$

commonly referred to as *phase angle*, where $(180/\pi)$ converts radians into degrees.

2.2 Shortcomings of bioelectrical impedance analysis

The four representations of impedance; R, Xc, Z and φ presented in section 2.1 combined constitute the representation of impedance in the Cartesian plane, and have different roles and shortcomings in bioelectrical impedance analysis. Simplification of the human body as a cylinder expressed by equation 2.2 – which is mathematically described as a uniform cylinder multiplied with a resistivity constant ($\rho_{(\Omega \cdot m)}$) – acts as the basis of body composition assessment, using height as a proxy for cylinder length [35]. However, identifying body composition on the basis of resistance alone fails, and only accounts for a small proportion in coefficient of correlation (R^2) in various applied equations [8]. An attractive solution is to define an alternative representation, where equation 2.2 is extended to express volume [18]:

$$V_{(m^3)} = \rho_{(\Omega \cdot m)} \cdot \frac{(L_{(m)})^2}{R_{(\Omega)}} \tag{2.5}$$

Equation 2.5 gives an empirical relationship between subject volume, resistance and height. However, expressing volume through equation 2.5 requires knowledge of the resistivity constant, which is medium-dependent, and likely subject-dependent. A solution is to disregard the constant, and use the remaining parts of equation 2.5 under the assumption of proportionality:

$$V \propto \frac{L^2}{R} \tag{2.6}$$

The representation in equation 2.6 is commonly referred to as resistance index (R^{index}), and is a common variable in BIA equations [1]. Neither equation 2.5 nor R^{index} account for variations in body shapes that can stem from variations in ethnicity or natural variations among individuals of the same ethnicity. The shortcomings of BIA is further complicated as differently shaped objects with an identical height may have identical resistance [8]. To account for the aforementioned variations in body shape, two new variables are proposed by [5], where a fraction of segmental resistance and reactance measurements in trunk and extremities is used to distinguish body shapes of subjects:

$$R^{Shape} = \frac{R_{Trunk}}{mean(R_{Arms}) + mean(R_{Legs})} \tag{2.7}$$

$$Xc^{Shape} = \frac{Xc_{Trunk}}{mean(Xc_{Arms}) + mean(Xc_{Legs})} \tag{2.8}$$

The proposed indices R^{Shape} and Xc^{Shape} was proved in [5] to correlate with circumference of arms, trunk and trunk length. It should be noted that measurements of resistance and reactance in the trunk is unstable, as the trunk constitutes a large proportion of body mass, yet constitutes a small proportion of resistance due to the large cross-sectional area in comparison to its length [22]. Thus, the indices in equation 2.7 and 2.8 may be contain a degree of noise that is undesirable. While the assumption of shape constitutes one major shortcoming of bioelectrical impedance analysis, the assumption of a constant hydration in subjects likely equally as flawed. A BIA device may be able to detect longitudinal changes in a subject [34], yet it is recommended to measure during normal hydration levels [22].

The phase angle given in equation 2.4 is variously suggested as a variable that can determine levels of hydration or malnutrition in subjects [29] [27] [22].

3 Data

3.1 Data source

The data obtained for this thesis is partly collected from routine medical examinations and partly from several research studies ([35], [3], [17], [26], [7]) at the Clinical Nutrition Unit at Sahlgrenska University Hospital between 2007-2016. Below, the total number of patient visits to the Clinical Nutrition Unit are plotted over year.

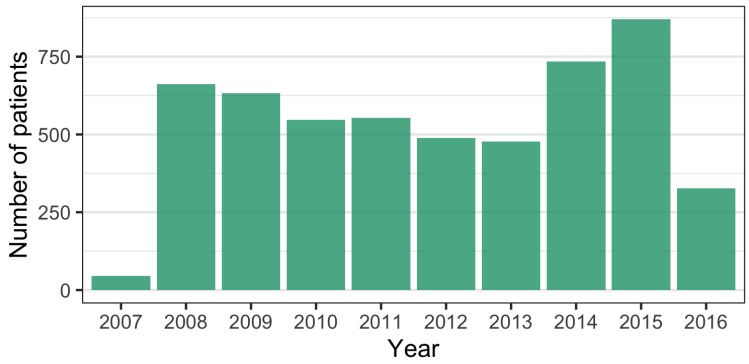


Figure 3.1: Number of patient visits per year.

Note that the sum of patient visits from figure 3.1 corresponds to the cleaned data described in section 3.3. Predictions from the two techniques (BIA and DXA) are not limited to MM and LST. As both techniques attempt to predict the full body composition, there are predictions of the other components of the body as well; bone mass (BM) and fat-mass (FM) in legs, trunk and arms. The predictions of BM and FM are irrelevant for the aims of this thesis and will only be used in the context of filtering for outliers in data, as described in section 3.3.

3.2 Data description

Each patient has undergone assessments of body composition using the two techniques; DXA and BIA. A distinction should be made clear early on about the variables meaning

and their naming conventions to avoid confusion; both techniques *predict* body composition, where predictions from BIA are called *Muscle Mass* (MM) and predictions from DXA are called *Lean Soft Tissue* (LST). In line with the discussion in [35], the predictions from BIA are assumed to be predictions on LST, rather than MM. In other words, despite the confusing naming conventions, the two methods aim to predict the same quantity, i.e. MM and LST are in reality the same thing. The naming conventions provided are kept throughout this chapter as it allows for a clear distinction between quantities generated by the two techniques.

Furthermore, the BIA device outputs two kinds of electrical measurements – resistance and reactance – in different parts of the body. These are different from the predictions mentioned above in the sense that they are purely measured. Exactly how a BIA device transforms a current into electrical variables depends upon the device in question, as there are several methods [27]. Despite being measured, electrical variables are inevitably noisy. Outside factors such as hydration status, contact with the devices electrodes and fasting status of subjects may cause two subjects that are identical w.r.t anthropometry, age and similar to have non-identical measurements of resistance and reactance [23]. Thus, the variations in electrical variables corresponds to a high degree to *epistemic uncertainty*, i.e. "unknown unknowns". It is not possible to completely single out the underlying factors leading to noisy measurements and obtain perfect predictors. Noise can however be reduced slightly, by creating new variables that are based on the confounding factors given by other variables, as will be discussed in section 3.5.

The obtained data is arranged in such a way that one row corresponds to a measurement of one subject. However, subjects are not restricted to have a single measurement (visit), and in cases where several measurements have taken place on one subject, the subject in question will appear in several rows. The total amount of subjects in the data are 3746, and with some subjects re-visiting the Clinical Nutrition Unit, the total number of measurements are 5760. Each row contains outputs from BIA and DXA, combined with information about the patient's anthropometry, electrical variables and date of hospital visit etc. Below, a table of variables are shown, where variables of similar kind are grouped together. The column $\mathbf{n_{group}}$ reveals how many variables that can be found within the corresponding group.

Variables	$\mathbf{n_{group}}$	**Type**
$\text{LST}_{RightArm}$, $\text{LST}_{LeftArm}$, $\text{LST}_{RightLeg}$, $\text{LST}_{LeftLeg}$	4	Continuous
$\text{MM}_{RightArm}$, $\text{MM}_{LeftArm}$, $\text{MM}_{RightLeg}$, $\text{MM}_{LeftLeg}$	4	Continuous
Resistance, Reactance	48	Continuous
Date, Visit, ID	3	Mixed
Age, Height, Weight	3	Continuous
Gender, Model_Gender, Tanita_BodyType	3	Factor

Table 3.1: Variable overview, prior to data manipulation and data cleaning.

With the device operating on 4 frequencies (5 kHz, 50 kHz, 250 kHz, 500 kHz), measuring both resistance and reactance in 6 body sections (right arm, left arm, right leg, left leg, left side, both legs), the total number of variables impedance-related variables in table 3.1 are 48, where half are variables of resistance, half are variables of reactance. Note that the measurement of segment "both legs" is a separate measurement that is not deterministically obtained ny adding "left leg" + "right leg". However, such addition approximates the segment "both legs" fairly well. *Model_Gender* and *Tanita_BodyType* are input variables fed to the impedance device prior to measurement. *Visit* is a counter for the cumulative number of visits each patient has, ordered by the *Date* variable. *Gender* is a variable provided directly to the database

by the practitioner who is overseeing the procedure. *Model_Gender* is provided by selecting either "Male" or "Female" when standing on the BIA device, and *Tanita_BodyType* is provided by selecting one of "Standard", "Athletic" or "Obese" when standing on the BIA device.

3.3 Data cleaning

All data reductions on the original data supplied by the commissioner is performed in the order described in table 3.2 below. All reductions are done by row, i.e. if there is an incorrect or missing value on one variable, then the complete row is removed. Thus, the number of rows removed might give a different outcome if the steps are performed in a different order.

	$\mathbf{n_{removed}}$	**Motivation**
1:	3	*Tanita_Bodytype* $\neq$ "Standard" alters predictions on *MM*
2:	25	*Model_Gender* $\neq$ *Gender* alters predictions on *MM*
3:	6	*LST* contains NA, unknown reason
4:	1	$FM_{LeftArm} > 100kg$ (from BIA), too extreme to be considered realistic
5:	126	*Reactance* > 0, not plausible to have positive *Reactance*
6:	259	$Z(d_M(\textit{Adjusted electrical variables})) > 2$, removal of extreme outliers
7:	1	$ALST - MM > 15kg$, extreme outlier prediction from the BIA device

Table 3.2: Data reduction; number of measurements concerned with a short motivation.

As mentioned in section 3.2, the BIA device outputs a prediction on the different components of the body (where MM is the only one of interest), using a regression algorithm [34]. The exact model and parameters generating predictions is not disclosed by the device manufacturer, but it is believed by the commissioner that *Tanita_BodyType* and *Model_Gender* affect the regression output of MM. Thus, setting *Tanita_BodyType* to "Athletic" is likely to give a different outcome from setting *Tanita_BodyType* to "Standard". Consequently, all instances of the former setting are removed. In cases where *Model_Gender* is different from *Gender*, it means that either a male subject has predictions on MM that is generated with an equation that is constructed for female subjects, or vice versa. Consequently, all instances where the two variables mismatch are removed.

In agreement with the commissioner, any value for *Reactance* on the ”wrong” side of 0 is considered unrealistic. Representing reactance as positive means that the body works as an inductor, effectively meaning that current leads voltage. The human body contain cell membrane, which has electrical properties of a capacitor. Which sign the BIA device chooses to represent *Reactance* in the output is not known for certain, but the vast majority of electrical measurements are negative. Thus, any positive value for any of the 24 available *Reactance* variables for a given individual result in a removal of the row, and consequently 126 rows were removed.

After removal of reactance that was deemed incorrect, an intermediate set was created only using standardized measurements of adjusted electrical variables where adjustments were stratified on gender. Several intermediate steps were performed and empirical (graphical) checks were performed to see that the filtering process was performed accurately. Ultimately, Mahalanobis distance (d_M) was computed for all the involved variables, and any standardized distance larger than a specified limit was removed, resulting in 259 removals. The adjusted measurements are performed with the aim of removing outlier measurements of electrical variables – when extreme or erroneous measurements are present – as opposed to

outlier individuals. A more detailed description of confounders for electrical measurements is given in section 3.5.

Predictions from the BIA device that are considered completely unrealistic are removed. In one subject, predicted FM in the left arm exceeded 140 kg, whereas the subject's total weight was 114 kg. The anomaly mentioned is not directly related to quality the target variable given by the DXA device, nor to the electrical variables from BIA, yet some concerns can be raised with respect to the quality of the measurement as a whole, and the observation was removed. The last removal in table 3.2 was checked several times, and no particular reason to why a prediction difference exceeding 15 kg was found. However, this individual was highly influential in different analyses, and was removed.

3.4 Data wrangling

Outputs on LST are given with unit grams, while outputs from MM are given in kilograms. To enhance comparisons between the two, LST is converted to unit kilograms according to $LST_{(kg)} = LST_{(g)}/1000$. This transformation enables comparisons with similar studies, which represent similar quantities to LST in units of kilograms.

Due to the removals described in section 3.3, the *Visit* and *ID* variables are not monotonically increasing. For programming conveniences (e.g. indexing), these two variables are recomputed, ignoring the removals completely. This means that if an individual has five hospital visits; $Visit = \{1, 2, 3, 4, 5\}$ and the second visit is removed due to inaccuracies in data, then the updated column will appear as $Visit = \{1, 2, 3, 4\}$ rather than $Visit = \{1, 3, 4, 5\}$. Similar logic applies to the ID variable.

Despite the removals of *Reactance* that appeared as positive numbers, described in section 3.3, all remaining reactance measurements are converted to positive. The transformation makes little sense from a logical perspective, yet it is a common transformation in similar studies. As a consequence, the rather illogical transformation allows for a direct comparison of model parameters to other BIA equations.

3.5 Feature engineering

Several new variables are presented in this section and a summary over all variables are given at the end of this section. Appendicular lean soft tissue (ALST) is computed as the sum of LST in extremities and appendicular muscle mass (AMM) is computed as the sum of muscle mass in extremities. To distinguish the sum of these *predictions* generated by the DXA and BIA devices respectively, these will be referred to as y and y_{bia}. Note here that y is seen as the ground truth of $ALST$. The naming conventions are straightforward; on one hand, y is the target variable – and is named as such. On the other hand, y_{bia} is a prediction on the same quantity, yet it only lives within this thesis as a benchmark, as y_{bia} constitutes what should be improved upon. Thus, y_{bia} constitutes a target variable in a sense, and the subscript distinguishes the two.

$$ALST = y = LST_{RightArm} + LST_{LeftArm} + LST_{RightLeg} + LST_{LeftLeg} \tag{3.1}$$

$$AMM = y_{bia} = MM_{RightArm} + MM_{LeftArm} + MM_{RightLeg} + MM_{LeftLeg} \tag{3.2}$$

As outlined in chapter 1, predictions of y_{bia} generated by the BIA device are less precise for extreme ranges of BMI. This issue is visualized in figure 3.2, where a Y-vs-Y plot of y and y_{bia} is plotted side-by-side of a plot where the difference $y - y_{bia}$ is plotted as a function of BMI. BMI is calculated according to standard formula as $BMI_{(kg/m^2)} = Weigth_{(kg)}/Height^2_{(m)}$.

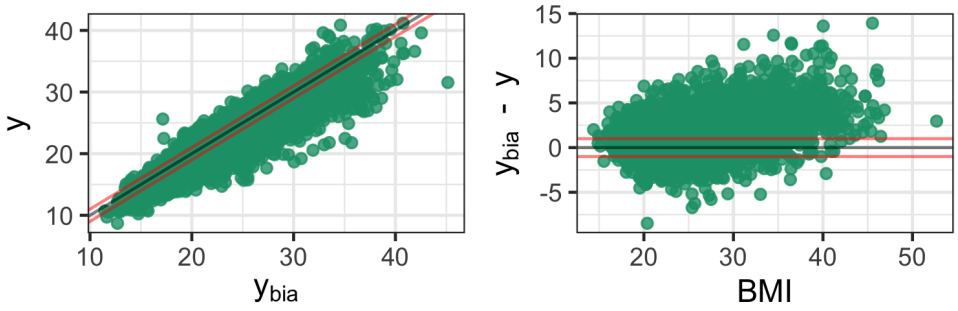


Figure 3.2: Issues with BIA predictions. The black line denotes identical predictions and red lines denote ± 1 kg of identical predictions from BIA and DXA. Left: BIA predictions have a tendency to overestimate muscle mass. Right: Tendencies towards overestimation from BIA compared to DXA increases with increasing BMI.

The left hand side of figure 3.2 may serve as a motivation to create a new equation (e.g. this thesis) as the BIA device has a tendency to overestimate muscle mass. The right hand side of figure 3.2 suggests that BMI is a contributing factor for overestimation of ALST as the overestimation increases along with increased BMI. Note also that the trend of difference $y_{bia} - y$ poses problems of estimation in underweight or severely underweight subjects with $BMI < 20$, as the trend for decreasing BMI indicates that the BIA device underestimates muscle mass for such individuals.

Appendicular *Resistance* (R) is computed as the sum of individual R and Xc in extremities. The motivation behind this transformation is simple; R reveal – to some degree – characteristics of the medium the electrical current flows through. Thus, all components that make up the sum of $ALST$ should be weighed by the contribution of the electrical variables that help explain components of the sum. The computations can then be seen as a weighted average. The algorithm for calculating appendicular resistance is given below.

Algorithm 1 Total appendicular *Resistance*

Require: Appendicular *Resistance* variables: $R_{(RA,f)}$, $R_{(LA,f)}$, $R_{(RL,f)}$, $R_{(BL,f)}$

$$R_{(L,f)} = \left(R_{(RL,f)} + R_{(BL,f)}\right) \cdot \left(\frac{1}{3}\right) \quad \text{(Average leg } \textit{Resistance}\text{)}$$

$$R_{(A,f)} = \left(R_{(RA,f)} + R_{(LA,f)}\right) \cdot \left(\frac{1}{2}\right) \quad \text{(Average arm } \textit{Resistance}\text{)}$$

$$R_f = \left(R_{(L,f)} + R_{(A,f)}\right) \cdot 2 \quad \text{(Total appendicular } \textit{Resistance}\text{)}$$

return R_f

Using algorithm 1, frequency-specific resistance measurements R_5, R_{50}, R_{250}, R_{500} are obtained. Note that the logic in algorithm 1 can be used for reactance measurements Xc_5, Xc_{50}, Xc_{250} and Xc_{500} as well. The averaging of segments ”right leg” and ”both legs” is constructed as to smooth any undesired noise that may occur in the single segment measurements and that ”both legs” was empirically checked to be approximately twofold compared to ”right leg”. Note that measurements on R and Xc for the left leg are not included in algorithm 1, due to a biasing factor prior to the year of 2015. Below, output of R and Xc are plotted over the complete time-span during which data was collected. Note that only the frequency 50 kHz is included, to avoid cluttering.

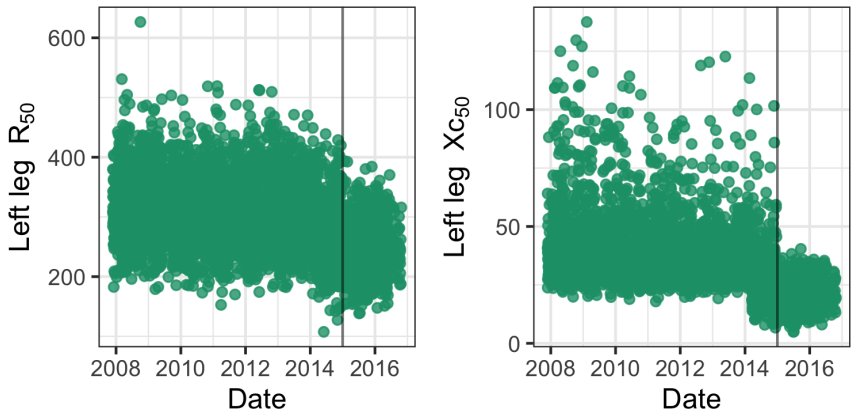


Figure 3.3: Discrepancies of *Resistance* (left) and *Reactance* (right) in the left leg. Black vertical line separates measurements prior and after 2015.

Discussions with the commissioner regarding the seemingly different behaviour of the output plotted in figure 3.3 did not lead to any conclusions why measurements differ before and after 2015. The discrepancies plotted above was only found in the left leg, where the measurements in figure 3.3 after 2015 are in line with the corresponding measurements in the right leg, i.e. more plausible. Similar patterns were found for all frequencies, leading to the conclusion that whatever the underlying bias is, it is only connected to measurements of the left leg. Another interesting factor of biased measurements is that other measurements, such as *Reactance* in the segments ”Left side” or ”Both legs” that in a purely logically sense includes *Reactance* measured in "left leg" did not show any visual signs of the displayed behaviour. Resistance and reactance measurements in the left leg only constitutes a fraction of total appendicular impedance (algorithm 1), but any added inclusion of unnecessary bias is considered as undesirable.

Algorithm 2 Appendicular phase angle

Require: Appendicular resistance variables: $R_{(RA,f)}$, $R_{(LA,f)}$, $R_{(RL,f)}$, $R_{(LL,f)}$, $R_{(BL,f)}$

Require: Appendicular reactance variables: $Xc_{(RA,f)}$, $Xc_{(LA,f)}$, $Xc_{(RL,f)}$, $Xc_{(LL,f)}$, $Xc_{(BL,f)}$

$$\varphi_{RA,f} = tan^{-1}\left(\frac{Xc_{(RA,f)}}{R_{(RA,f)}}\right) \cdot \left(\frac{180}{\pi}\right) \quad \text{(Phase angle: right arm)}$$

$$\varphi_{LA,f} = tan^{-1}\left(\frac{Xc_{(LA,f)}}{R_{(LA,f)}}\right) \cdot \left(\frac{180}{\pi}\right) \quad \text{(Phase angle: left arm)}$$

$$\varphi_{RL,f} = tan^{-1}\left(\frac{Xc_{(RL,f)}}{R_{(RL,f)}}\right) \cdot \left(\frac{180}{\pi}\right) \quad \text{(Phase angle: right leg)}$$

$$\varphi_{LL,f} = tan^{-1}\left(\frac{Xc_{(LL,f)}}{R_{(LL,f)}}\right) \cdot \left(\frac{180}{\pi}\right) \quad \text{(Phase angle: left leg)}$$

$$\varphi_{BL,f} = tan^{-1}\left(\frac{Xc_{(BL,f)}}{R_{(BL,f)}}\right) \cdot \left(\frac{180}{\pi}\right) \quad \text{(Phase angle: both legs)}$$

$$\varphi_f = \frac{1}{5} \cdot \left(\varphi_{RA,f} + \varphi_{LA,f} + \varphi_{RL,f} + \varphi_{LL,f} + \varphi_{BL,f}\right) \quad \text{(Composite phase angle)}$$

return φ_f

Phase angle (φ), as given in equation 2.4 is the angle between R and Xc in the Cartesian plane. Frequency-specific phase angles φ_f may be calculated using composite measurements outputted from algorithm 1 or averaging over all available frequency-specific and

extremities-specific measurements. Empirical experiments proved that averaging over all angles computed in extremities yields a composite angle that is closer to the ground truth (which was constructed to be known). All steps for computations of appendicular phase angle is given in algorithm 2. Note that the algorithm computes appendicular phase angle using the discussed biased measurements of the left leg. As both resistance and reactance are biased in the same "direction" (e.g. higher) prior to 2015, the angle is not believed to be altered to a degree which is unsatisfactory. On the contrary, averaging over more angles on the noisy data proved to smooth out the resulting measurements.

The apparent theoretical dependency between Resistance and subject size as given in equation 2.2 is defined below, based on appendicular resistance from algorithm 1. The variable R_f^{in} in this thesis differ from e.g. [5], where trunk measurements of resistance are included in computations of R_f. Thus, no attempt to approximate whole body volume is sought. However, the (proposed) measurement may be interpreted as *proportional* to appendicular volume. Equation 3.3 (below) offers another – perhaps more interesting – statistical interpretation. As the dependency between resistance and subject height is present, the computed indices can be seen as *interaction*; e.g. the statistical importance of one variable (resistance) is dependent upon another variable (height). Additionally, a second index variable φ_f^{in} is computed by normalizing with subject weight.

$$R_f^{in} = \frac{Height^2}{R_f} \tag{3.3}$$

$$\varphi_f^{in} = \frac{Weight^2}{\varphi_f} \tag{3.4}$$

Equation 3.4 is a variable that – to the authors knowledge – is unseen and serves as a novel suggestion on data transformation. The variable is constructed on a hypothesized dependency between phase angle and cell mass, where cell mass is directly linked to weight. The compound variables of R_f, φ_f, R_f^{in} and φ_f^{in} are plotted versus y in figure 3.4 to display the functional form, where each frequency f receives a unique color. To avoid any excessive cluttering, only two frequencies (5 kHz and 250 kHz) are included.

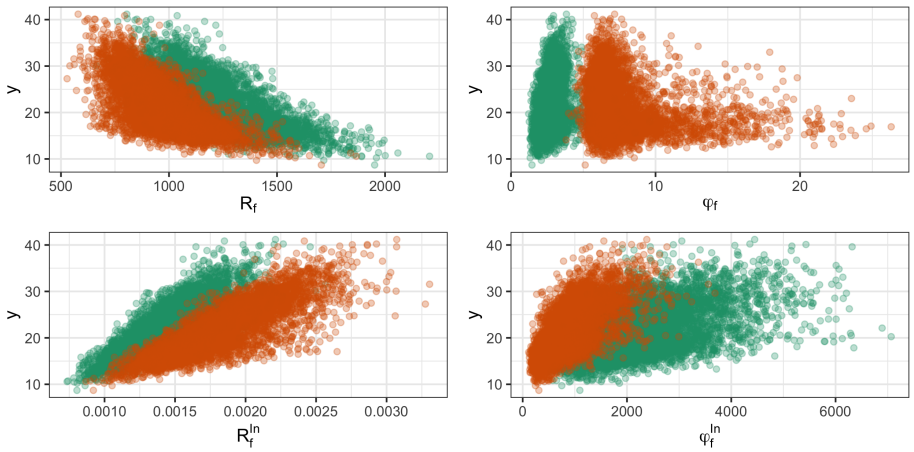


Figure 3.4: Functional form between electrical variables and target variable. Green: 5 kHz, orange: 250 kHz. Top row: raw electrical variables, bottom row: electrical variables adjusted with height (left) and weight (right).

As can be seen in figure 3.4, the adjustments has aligned the functional form of variables and target to resemble more of a linear relationship. A factor that might be of importance is that each subplot of the top row of figure 3.4 corresponds to two distinct clusters (not highlighted); $y \approx 20$ is a separator for female and male patients, where the two clusters have seemingly different "slopes". The transformations to index variables computed and displayed in the bottom row have reduced the gender-specific slopes, yet not completely, introducing what more resembles a linear relationship between $ALST$ and the corresponding variable. The differences seen between different frequencies is reasonably explained by the fact that resistance – and hence phase angle – is to a high degree influenced by the hydration level of the corresponding medium, e.g. muscle. High frequencies identify TBW, while low frequencies are not able to penetrate the cell membrane, essentially measuring ECW. As such, higher frequencies have a higher correlation with $ALST$. In addition to the functional form given in 3.4, the functional form between anthropometric variables and the target is displayed below, colored by gender.

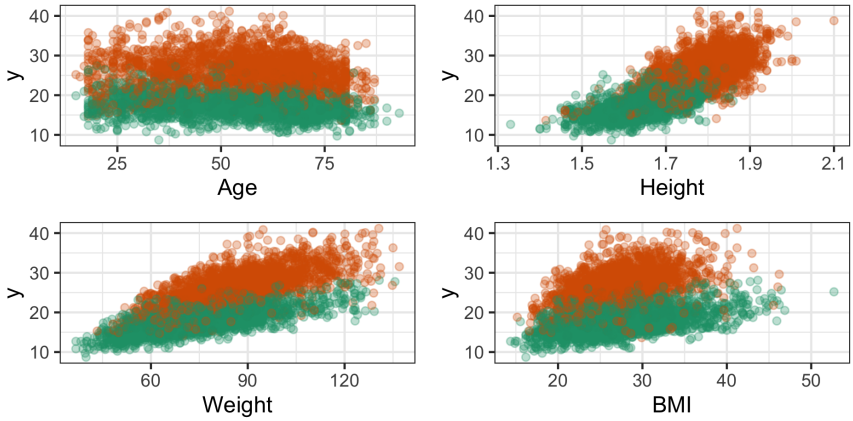


Figure 3.5: Functional form between anthropometric variables and target, colored by gender. Orange: males, green: females.

The difference between male and female subjects in terms of $ALST$ discussed with respect to figure 3.4 is displayed in figure 3.5, where male subjects on average have more $ALST$ than female subjects. The functional form of anthropometric variables suggests that the relationship to $ALST$ is linear, or close to linear, where primary differences can be found between genders. If any non-linearity is to stand out, it is that $ALST$ have a quadratic relationship with Age, where the peak of $ALST$ occurs in mid-life, as is discussed in [6]. In addition to the index variable of phase angle, two forms of entropy-related variables are proposed. Computations involve the empirical entropy and the multivariate Mahalanobis distance:

$$Empirical\ entropy = -\sum_i \mathbf{v}_i \cdot log\left[\mathbf{v}_i\right] \tag{3.5}$$

$$Mahalanobis\ distance = \sqrt{(\mathbf{v} - \bar{\mathbf{v}})^T C^{-1} (\mathbf{v} - \bar{\mathbf{v}})} \tag{3.6}$$

The complete procedure involves several steps and is executed as follows, where the initial phase angles for step 1 are computed according to equation 2.4:

1 Compute component-specific phase angle vectors for right arm, left arm, right leg and left side. Each component receives one vector with angles computed for 5 kHz, 50 kHz, 250 kHz and 500 kHz.

2 Normalise the computed vectors such that each vector sums to one, and each vector elements are bounded in [0,1].

3 Compute entropy as in equation 3.5, one entropy obtained for each normalised vector, denoted φ_f^e.

4 Compute sample mean and sample covariance for the entropy vectors.

5 Compute Mahalanobis distance as in equation 3.6 between the entropy vectors.

6 Compute the natural logarithm of the distance obtained in step 5, denoted as D_φ.

Step one transforms angles into proportions, and the proportions are regarded similar to probabilities. If all frequencies applied gives an angle that is the same, then entropy computed in step 3 will by definition be maximized. The entropy variables describe dispersion that arises from applying different frequencies to the same body component, which is justified as frequencies on different ranges can differentiate between ECW and TBW. The distance-based variable may be able to detect edema, primarily common amongst elderly, where a large amount of water is moved to the legs. As *Resistance* is highly correlated with water content, which is high in muscles, the magnitude of *Resistance* is similar in subjects with edema and muscular subjects. If a subject display electrical measurements that are widely different when comparing legs and arms, such as in subjects with edema, then the distance will be high. This is especially true for legs, which – due to the bigger size – have higher *Resistance* and is the component that is primarily affected by edema.

This section has described how the target variable is computed and several electrical variables have been defined. Many of the electrical variables have connections to the theory described in chapter 2, but this section also introduced three novel suggestions of electrical variables. Similar to table 3.1, relevant variables are displayed below. For notational convenience, the variables *Resistance* and *Reactance* and are shown without sub- and superscripts, although both are measured for all frequencies in all body segments.

Variables	**Notation**	$\mathbf{n_{group}}$	**Type**
ALST	y	1	Continuous
AMM	y_{bia}	1	Continuous
Resistance, Reactance	R, Xc	40	Continuous
Resistance index	R_f^{in}	4	Continuous
Phase angle index	φ_f^{in}	4	Continuous
Appendicular angle entropy	φ_f^e	4	Continuous
Appendicular entropy distance	D_φ	1	Continuous
Visit, ID	–	2	Integer
Age, Height, Weight, BMI	–	4	Continuous
Gender dummies	D_{Male}, D_{Female}	2	Binary

Table 3.3: Variable overview, after data manipulation and data cleaning.

Note that the raw *Resistance* and *Reactance* variables in table 3.3 are fewer than in table 3.1, this is due to removal of measurements in the left leg displayed in figure 3.3. Note also that appendicular resistance (algorithm 1) and appendicular phase angle (algorithm 2) are not included above, since these are only intermediate steps for computations of resistance index (equation 3.3) and phase angle index (equation 3.4).

3.6 Data partition

The data partition is sample-based, where the aim was to have 4000 training data points, 800 validation data points and the remaining 539 points left for test. To keep the multi-level structure that arises from repeated visits intact, a sampling scheme was created. The scheme includes sampling ID's for the corresponding set, and recursively check if target number of points were matched. If not, sampling continued. If the set was too large, a pre-specified number of ID's were removed, and sampling continued until target criteria's were met.

	All		Train		Validation		Test	
Nr of visits (v)	n_v	$v \cdot n_v$	n_v	$v \cdot n_v$	n_v	$v \cdot n_v$	n_v	$v \cdot n_v$
1	2567	2567	1935	1935	386	386	246	246
2	431	862	327	654	57	114	47	94
3	288	864	216	648	44	132	28	84
4	144	576	110	440	20	80	14	56
5	54	270	37	185	10	50	7	35
6	21	126	12	72	5	30	4	24
7	7	49	7	49	0	0	0	0
8	2	16	1	8	1	8	0	0
9	1	9	1	9	0	0	0	0
$\sum =$	3515	5339	2646	4000	523	800	346	539

Table 3.4: Multi-level structure of data. n_v is number of unique patients with v visits in total, $v \cdot n_v$ is the number of measurements for v visits.

The resulting partition in table 3.4 contains 2646 unique patients in the training set, 523 unique patients in the validation set and 346 unique patients in the test set. Divided by the total number of measurements in the full set results in approximately 1.51 observations on average. For the partitioned sets, this computation results in approximately 1.51, 1.52 and 1.56 observations on average, i.e. the multi-level structure is intact and also consistent through partitions. The corresponding MSE is 6.46 for the complete data, 6.55 for the training set, 6.30 for the validation set and 6.05 for the test set.

Due to the issues that BMI poses on predictions outlined in section 3.5, a selection of subjects in different categories of BMI are taken for model evaluation. Note that this particular selection acts as a complementary evaluation to the validation set, for a small selection of subjects and does not replace evaluation of the complete validation set. The selection consists of three males and three females from the validation set and the selection criteria is made primarily by selection of different ranges of BMI, but also by checking how much other key variables deviate from their corresponding mean. Table 3.5 shows the selection.

ALST	Gender	Age	Height	Weight	BMI	R^{in}_{250}	φ^{in}_{250}
24.61	Male	66	1.76	71.40	23.18	0.0019	756.22
25.60	Male	40	1.76	86.10	27.79	0.0020	1025.45
29.78	Male	30	1.78	105.90	33.42	0.0022	1708.05
15.72	Female	59	1.62	55.95	21.36	0.0015	505.61
17.11	Female	71	1.64	66.20	24.76	0.0016	712.42
17.39	Female	51	1.63	88.60	33.35	0.0017	1056.91

Table 3.5: Selection of subjects with varying BMI from validation data.

Table 3.5 indicates that the selection does not include a large variety in *Height* within the gender-specific groups. However, both *Weight* and P_{250}^{in} vary quite a lot. For different categories of BMI – underweight, normal and obese – the above table can be regarded as representative with respect to the other variables since these observations have a small multivariate distance to other subjects with similar BMI. Notably, the selection is not single-handily performed based upon BMI since such a selection may provide a subset with either erroneous measurements or simply based upon measurements that are too extreme, due to the high variability that BMI introduces in electrical measurements.

4 Method

4.1 Bayesian inference

All Bayesian algorithms are – to various degrees – based on the foundation of prior to posterior updating. The updating procedure always includes a prior $p(\theta)$ and a likelihood $p(\mathcal{D}|\theta)$, where θ are the unobserved model parameters which characterize the model one wants to perform inference upon together with the data $\mathcal{D} = \{\mathbf{x}_i, y_i\}$, $i = 1, 2, ..., n$. The mix of a prior and likelihood is a way to quantify how plausible an event of interest is, e.g. a continuous response in regression or similar, and reflects the full uncertainty of the model one performs inference upon. Bayesian prior-to-posterior updating is typically computed via Bayes theorem;

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta) \cdot p(\theta)}{p(\mathcal{D})} \propto p(\mathcal{D}|\theta) \cdot p(\theta), \tag{4.1}$$

where $p(\mathcal{D})$ is the marginal probability of data. The simplification from 4.1 is valid as the marginal probability simply transforms the posterior in such a way that integration sums to one. Regardless of the marginal probability, the posterior still has the same shape. After computing the sought normalized or un-normalized posterior distribution in equation 4.1, the most common interest is usually to average over expectations with respect to the posterior, and obtain a *posterior predictive distribution*:

$$p(\tilde{y}|y) = \int p(\tilde{y}|\theta) \cdot p(\theta|y) \partial\theta = \int p(\tilde{y}|y, \theta) \cdot p(\theta|y) \tag{4.2}$$

where y replaces $\mathcal{D}$ and explanatory variables $\mathbf{x}$ are intentionally left out to avoid cluttering. The posterior predictive distribution answer queries about unseen data $\tilde{y}$ and embodies the full uncertainty about model parameters given in the posterior distribution over θ. Using the expectation $\tilde{y}$ enables computations of model performance, such as (Bayesian) coefficient of variation R^2 [10] and the mean square error (MSE);

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i^s)^2 \tag{4.3}$$

$$R^2 = \frac{V[\tilde{y}_i^s]}{V[\tilde{y}_i^s] + V[\tilde{r}_i^s]}, \tag{4.4}$$

where $\tilde{r}_i^s = y_i - \tilde{y}_i^s$ and $s = 1, 2, ..., S$ denotes posterior draws for model parameters. For both MSE and R^2 in equation 4.3 and 4.4, it is possible to use a single draw of posterior parameters and calculate a single R^2 and a single MSE based on those. By using a full set of posterior draws θ^s, the metrics displayed above would correspond to distributions of retrodictive model performance R^2 and predictive model performance MSE, displaying the full uncertainty in the corresponding metric.

In the context of Bayesian prior-to-posterior updating, key distinctions can be made on the information carried by the prior distribution; where the spectrum includes e.g. *regularizing priors*, *weakly informative priors* and *non-informative priors*. The information carried by regularizing priors places a heavy penalty for posterior mass sufficiently far away from the prior, leading to a posterior distribution that kept within reasonable bounds from the prior. On the contrary, non-informative priors barely affect the posterior distribution at all, leading to a posterior that is highly controlled by the data. Weakly informative priors lie somewhere in between the two. Using a Laplace prior is a way to promote sparsity in parameters, also known as *Lasso regression* in regression settings [9]. A weakly informative prior is one that intentionally contains slightly less information than what the Bayesian practitioner knows a-priori about the parameter of interest [25]. A common choice for weakly informative priors is the *Cauchy* distribution, due to its wide tails.

Priors sometimes play another important role in Bayesian inference with regard to the posterior distribution; in some cases, there exist an analytical solution for how to compute the posterior $p(\theta|\mathcal{D})$ via conjugate prior-likelihood pairs. For the remainder of this chapter, no attempts to display closed form solutions for posteriors are presented (unless explicitly stated otherwise). Instead, the remaining parts of this chapter is focused on explaining models without analytical solutions (intractable posteriors) and various approximations to the desirable posterior distribution are presented.

4.2 Bayesian hierarchical linear regression

Bayesian linear regression aims to relate the target variable y_i to some distribution – typically the Gaussian distribution – parameterized by a mean (μ_i) and a standard deviation (σ), where $i = 1, 2, ..., n$. In itself, the mean μ_i is a deterministic linear function – variously known as the *link function* or *linear model* – consisting of the intercept parameter α, slope parameters β_k and features x_{ik}, where $k = 1, 2, ..., K$. The model parameters $\theta = \{\alpha, \beta_k, \sigma\}$ need be assigned appropriate prior distributions that reflect a-priori beliefs about the phenomena that is modelled. By performing the aforementioned requirements, one achieves Bayesian linear regression. The extension from Bayesian linear regression to Bayesian hierarchical linear regression is achieved by assigning prior distributions to prior parameters using *hyper-priors*, where hyper-priors models specific sub-groups of observations in the data. A general formulation is given below using only Gaussian priors and hyper-priors with zero mean and unit standard deviation:

$$
\begin{aligned}
y_{ij} &\sim \mathcal{N}(\mu_{ij}, \sigma) && [\textit{Likelihood}] \\
\mu_{ij} &= \alpha_j + \beta_k \cdot x_{ik} && [\textit{Hierarchical link function}] \\
\alpha_j &\sim \mathcal{N}_J(\alpha_\mu, \alpha_\sigma) && [\textit{Hierarchical prior}] \\
\alpha_\mu &\sim \mathcal{N}(0, 1) && [\textit{Hyper-prior}] \\
\alpha_\sigma &\sim \mathcal{N}^+(0, 1) && [\textit{Hyper-prior}] \\
\beta_k &\sim \mathcal{N}_K(0, 1 \cdot \mathbb{I}_K) && [\textit{Prior}] \\
\sigma &\sim \mathcal{N}^+(0, 1), && [\textit{Prior}]
\end{aligned}
\tag{4.5}
$$

where $j \in \{1, ..., J\}$ denotes the sub-groups one models using varying intercepts and $\mathcal{N}^+$ denotes a truncated Gaussian with support on positive numbers. The hierarchical link function described in equation 4.5 now consist of a group-specific parameter α_j that is pooled towards the hierarchical mean given by α_μ, where the sizes of each sub-group affects the degree of pooling, or *shrinkage* [25]. In cases when unseen data can be correctly partitioned in the correct group of α_j, it is possible to use the hierarchical parameters for predictions. However, in cases where the aforementioned partition is impossible, the hierarchical mean μ_α and standard deviation σ_α can be used as these parameters describes the *population* of sub-groups [25]. Naturally, other choices of priors may be used in a given problem, and should be *elicitated* using *domain specific expertise*. In settings involving hyper-priors, the degree of regularization imposed on the model is determined by the model itself, which may be of benefit not only in hierarchical settings, but also when there exists ambiguity over appropriate a-priori beliefs about the phenomena modelled [25].

4.2.1 Assessing need for hierarchical modelling

A way to evaluate the gains made by incorporating an hierarchical structure in the data is to compute intraclass-correlation (ICC). Bayesian ICC is computed using the posterior draws of the standard deviation in the hierarchical prior $\tilde{\alpha}_\sigma$ and the standard deviation of the likelihood $\tilde{\sigma}$. Like Bayesian R^2 and Bayesian MSE, calculations on ICC gives a distribution.

$$
ICC = \frac{\tilde{\alpha}_\sigma}{\tilde{\alpha}_\sigma + \tilde{\sigma}} \tag{4.6}
$$

As can be seen from equation 4.6, ICC display the level of correlation within clusters α_σ. If this metric is high then the variation between clusters is high, suggesting that an hierarchical structure is required. Intuitively, if $ICC > 0.5$ then the between groups variation is higher than the variation in the model (likelihood). Such a scenario may suggest the need for a hierarchical model, given the chosen parametric form.

4.2.2 Model evaluation

A way of estimating out-of-sample predictive performance of a hierarchical Bayesian linear models is to compute the widely applicable information criteria (WAIC). This information criteria contains two main components; the *log point-wise predictive density* ($\widehat{lppd}$) and the *effective number of parameters* ($\hat{p}_{waic}$). Formulas for these estimates, for data-points y_i where $i = 1, 2, ...n$ and posterior samples θ^s for $s = 1, 2, ..., S$, are given below :

$$
\widehat{lppd} = \sum_{i=1}^{n} \left[log\Big(\sum_{s=1}^{S} p(y_i|\tilde{\theta}^s) \Big) \right] \tag{4.7}
$$

$$
\hat{p}_{waic} = \sum_{i=1}^{n} \left[\mathbb{V}_{s=1}^{S} \Big(log\ p(y_i|\tilde{\theta}^s) \Big) \right] \tag{4.8}
$$

where $\mathbb{V}_{s=1}^{S}$ is the variance computed for all posterior parameters. From equation 4.7 and 4.8, it is clear that the computations are performed with respect to all posterior parameters for each observation, then summed over all observations. In model comparison, the difference between best model $\widehat{lppd}$ can be computed, referred to as $\nabla\widehat{lppd}$. The best model then receives $\nabla\widehat{lppd} = 0$. To obtain the full WAIC estimate, the following formula is applied:

$$WAIC = -2(\widehat{lppd} - \hat{p}_{waic}) \tag{4.9}$$

Ideally, the WAIC estimates should be as low as possible, where a lower WAIC indicates better out-of-sample fit [37] [25]. However, a single WAIC estimate lacks meaning, as these estimates are made to be compared between models. The model with lowest WAIC has best predictive abilities. A standard error can then be computed on the log-pointwise predictive density for each sample and summed up, referred to as $se(WAIC)$. Two competing models with overlapping $se(WAIC)$ implies ambiguity over which model is best. Another way of checking model performance, that is not directly related to predictive power, is to compute importance ratios. Importance ratios are best used when all n data points are independent [37], which is quite a stretch for hierarchical models. However, these ratios r_i^s can be plotted as a function of some key variable in a model to evaluate if the particular variable poses difficulties in some regions, or to single out observations in a model that is extra hard to fit.

$$r_i^s = \frac{1}{p(y_i|\tilde{\theta}^s)} \tag{4.10}$$

The importance ratio may then be averaged (or summed) over all posterior samples $\tilde{\theta}^s$, providing one aggregated importance ratio per observation, which enables plotting one importance ratio per observation.

4.2.3 Markov chain Monte Carlo

The idea behind Markov chain Monte Carlo (MCMC) algorithms is to iteratively draw samples of all model parameters θ from a posterior distribution $p(\theta|y)$, which as a whole is intractable. The position of model parameters θ at some iteration is referred to as *states*, and any movement between iterations is referred to as a *transition*. As the name suggest, MCMC are Markov chains, where an updating of a state is dependent only on the previous state. On the limit, when the number of iterations converges to infinity, the sampler will have visited all regions of the posterior according to the density of in each region [25]. For smart and efficient exploration of the posterior distribution, or simply in situations where not all types of MCMC algorithms meet the requirements, a special breed of MCMC is required.

4.2.3.1 Hamiltonian Monte Carlo

Hamiltonian Monte Carlo (HMC) gains efficiency by making smart proposals for the transitions and by adaptation to the surroundings. The gains are made by incorporating yet another set of (independent) parameters; the auxiliary *momentum* parameters ρ, parametrized by a zero-mean vector and a *mass-matrix* M, and knowledge of the surrounding environment through the log-probability of data $p(\theta|y)$. Note that the momentum parameter plays no role in inference of the model, it serves only as an auxiliary paramter that increases efficiency [9]. The two sets of parameters then form a joint distribution $p(\rho,\theta|y) = p(\rho)p(\theta|y)$, which is used to define a Hamiltonian [33];

$$\begin{aligned} H(\theta,\rho) &= -log\big[p(\theta,\rho)\big] \\ &= -log\big[p(\rho|\theta)\big] - log\big[p(\theta|y)\big] \\ &= T(\rho|\theta) + V(\theta|y) \end{aligned} \tag{4.11}$$

which describes the total amount of energy in the system. Above, T denotes the kinetic energy and V denotes the potential energy, and together these two forms the total energy H of the system. Having defined the system in accordance with the physics analogy to a Hamiltonian system in equation 4.11, Hamilton's equations describe the transition-generating process in the system as a time-evolution [33]:

$$\frac{\partial \theta}{\partial t} = +\frac{\partial H}{\partial \rho} = +\frac{\partial T}{\partial \rho} \tag{4.12}$$

$$\frac{\partial \rho}{\partial t} = -\frac{\partial H}{\partial \theta} = -\frac{\partial V}{\partial \theta} \tag{4.13}$$

Equation 4.12 and 4.13 defines the evolution of the system in HMC, where updates θ is dependent upon the gradients of ρ and vice versa. As a consequence, the gradients for equation 4.12 and 4.13 are required. Defining ρ as a multivariate Gaussian yields $\partial T / \partial \rho = M\rho$ [28]. The gradients for $\partial \rho / \partial t$ are model-dependent. However, a general formulation may be written as a vector-derivative [9]:

$$\frac{\partial \rho}{\partial t} = \frac{\partial log\big[p(\theta|y)\big]}{\partial \theta} = \left(\frac{\partial log\big[p(\theta|y)\big]}{\partial \theta_1}, ..., \frac{\partial log\big[p(\theta|y)\big]}{\partial \theta_d} \right), \tag{4.14}$$

for a total of d model parameters. As Bayesian models includes multiplication of prior(s) and likelihood(s), taking the log of all products yields a sum. Luckily, the derivative of a sum is equal to the sum of derivatives [25], meaning that a general formulation for equation 4.14 is feasible (but not defined here). The process of updating states θ and momentum ρ is performed using the *leapfrog* algorithm (or leapfrog integrator), which alternates half-updates on ρ and full updates on θ, by taking L discrete leapfrog-steps, with step-size ϵ. The combination of L and step-size ϵ determines how far θ is allowed to move in parameter space [25]. If the product $L \cdot \epsilon$ equals to one, then the leapfrog algorithm is sufficiently able to generate a *trajectory* of L (intermediate) states from one end of the posterior to the other [9]. A pseudo-algorithm for the leapfrog integrator is given below.

Algorithm 3 Leapfrog algorithm

Require: L: Number of leapfrog-steps
Require: ϵ: Step-size
Require: θ: States
Require: ρ: Momentum
  for $l \in 1 : L$ **do**
    $\rho \leftarrow \rho - \frac{\epsilon}{2} \frac{\partial log[p(\theta|y)]}{\partial \theta}$
    $\theta \leftarrow \theta + \epsilon M \rho$
    $\rho \leftarrow \rho - \frac{\epsilon}{2} \frac{\partial log[p(\theta|y)]}{\partial \theta}$
  end for
  return θ

In algorithm 3, equation 4.12 and 4.13 are put into context, somewhat bridging the gap between a physics simulation and the model inference that is really of interest. The leapfrog algorithm does however only constitute the part of HMC that generates the trajectory of θ and ρ, i.e. the sequence of samples of length L that can be (and will be) tracked. The parts left to complete HMC are the *accept-reject* step and the *update* step. Let $\theta^{(t-1)}$, $\rho^{(t-1)}$ denote the

states and momentum prior to algorithm 3, and $\theta^{(t)}$, $\rho^{(t)}$ denote the output from algorithm 3, after all leapfrog steps are taken. Then the acceptance probability r is computed as [9]:

$$r = \frac{p(\theta^{(t)}|y)p(\rho^{(t)})}{p(\theta^{(t-1)}|y)p(\rho^{(t-1)})} \tag{4.15}$$

Here, r is the negative change in energy, which is exactly equal to 1 if the energy is preserved throughout the leapfrog steps [9]. Since the trajectory is performed under discretized time via L and ϵ, and not continuously, there is no guarantee that the energy will be preserved [28]. Energy preservation is especially difficult if the curvature of the posterior is steep in some directions [33]. The final update step is shown below:

$$\theta^{(t)} = \begin{cases} \theta^{(t)}, & \text{with probability } min(1, r) \\ \theta^{(t-1)}, & \text{otherwise} \end{cases} \tag{4.16}$$

Intuitively, the parameters θ can be thought of as particles that are flicked back and forth in a friction-less bowl, where the height of the bowl (at arbitrary state) correspond to the negative log-probability of data (through V) and the velocity of which the particle moves is regulated by the auxiliary parameter ρ (through T). If the particle faces an uphill slope, the kinetic energy T decreases and the potential energy V increases. If the uphill slope is steep and long enough, then the particle will eventually halt, and reverse down the slope, to a region that is of higher density, and to it's best ability preserve the total energy [28].

4.2.3.2 Chain diagnostics

In this section, the physical properties described in section 4.2.3.1 are used to evaluate how well the sampler behaves. Having defined HMC as a tool that operates under Hamiltonian laws, the system is mathematically well-defined. If the total energy H is not preserved, then issues such as *divergent transitions* appear. In other words, having defined rigorous physical laws over the environment in which updating is performed, those laws may be used to check if updating is performed adequately. If energy is not maintained before and after the leapfrog algorithm is executed, it simply means that the algorithm does not have optimal performance. [33] [25]

Further diagnostics are the Gelman-Rubin scale reduction statistic $\hat{R}$ [11] and the effective sample size N_{eff} [33]. N_{eff} is calculated as the fraction of the total number of samples drawn S and the inefficiency factor, measuring auto-correlation in the samples drawn from the posterior. Chains of samples that exhibit a high degree of auto-correlation are not efficient and receives low values for N_{eff}, essentially meaning that the samples are not independent. The $\hat{R}$ diagnostic is an estimate of *between-chain variance B* and *within-chain variance W*. These two diagnostic estimates can be computed for multiple parallel chains and are shown in brevity below.

$$\begin{aligned} \widehat{var}^{+}(\theta|y) &= \frac{S-1}{S}W + \frac{1}{S}B \\ \hat{R} &= \sqrt{\frac{\widehat{var}^{+}(\theta|y)}{W}} \\ \rho_t &= \frac{1}{\tau^2}\int_{\Theta} \theta^{(n)}\theta^{(n+t)}p(\theta)\partial\theta \end{aligned} \tag{4.17}$$

$$N_{eff} = \frac{S}{1 + 2 \cdot \sum_{t=1}^{\infty} \rho_t} \tag{4.18}$$

Above, ρ_t is the auto-correlation at lag t, S are the number of samples drawn by the sampler, τ^2 is the variance of a joint probability function $p(\theta)$. There is thus a notational redundancy from previous sections. However, this section can be seen in isolation. Regarding the magnitude of $\hat{R}$ and N_{eff}, there are no universal limits. If the chain(s) have converged to the true posterior, then $\hat{R} \approx 1$ [33]. Various suggestions for the effective number of samples exist, where e.g. 200 effective draws are suggested to be sufficient for describing the posterior [25].

4.2.3.3 Efficient re-parameterizations

One of the major obstacles to efficient sampling of HMC can be found within how the model is parameterized. If one parameter of the model is a function of another parameter, such as in the hierarchical prior presented in 4.5, then the sampler may turn into trouble because the parameters are highly correlated in parameter space. The parameterization in 4.5 is known as *centered parameterization*. A trick to avoid issues with correlations is to use *non-centered parameterization*. The Gaussian hierarchical intercept α_j in 4.5 may be defined by sampling the mean α_μ and the standard deviation α_σ separately and introduce a new parameter; $\eta_j \sim N_J(0, 1)$, where each group in the hierarchy receives a unique parameter [25]:

$$\alpha_j = \alpha_\mu + \alpha_\sigma \cdot \eta_j \tag{4.19}$$

The re-parameterization presented in equation 4.19 is still a Gaussian distribution (as in 4.5), only now it has been "un-normalized". The intuition is to regard η_j as a standardized – or unit Gaussian – version of α_j. In order to retrieve α_j to its original scale, it is shifted via its mean α_μ and scaled via itself and it's standard deviation $\alpha_\sigma \cdot \eta_j$. This reparameterization has introduced $j = 1, 2, ..., J$ new parameters into the sampler and by doing so, a simpler geometry is introduced. The extra parameters introduced places no particular extra burden on the sampler, as each parameter receives it's own momentum parameter and all gradients are used anyhow. [25]

4.2.4 Stan

Stan [33] implements a version of HMC called the No-U-Turn Sampler (NUTS). The special thing about NUTS samplers in Stan are that trajectories for model parameters θ from the leapfrog algorithm are not allowed to turn around and move back towards the origin. This restriction of the trajectories forces the NUTS samples to sample states in distant regions of the posterior and reduces auto-correlations of the posterior samples. Moreover, Stan provides tuning of hyper-parameters (not model parameters) that dictate how the sampler will behave. By feeding a model specification such as 4.5, the following is fine-tuned:

- Leapfrog steps (L).
- Step-size (ϵ).
- Mass matrix (M).

Additionally, Stan will calculate the log-posterior density and log-posterior sum of derivatives of a given model specification. Thus, the user need only specify the number of warm-up iterations during which the tuning will take place and the number of iterations for which to sample from the posterior, i.e. the selected size of posterior samples the user wishes to have. More tuning parameters are available, and up to the user. However, these are not covered in this thesis. Note that the acceptance probability in the NUTS sampler provided by Stan differs from the one presented in equation 4.15, although the same philosophy holds. For exact details, see [15].

4.3 Bayesian neural networks

Bayesian neural networks aim to find the posterior distribution $p(\mathbf{w}|\mathcal{D})$ of the network weights $\mathbf{w}$ conditioned on the data $\mathcal{D}$. Under the Bayesian paradigm, the posterior distribution is calculated according to Bayes theorem. However, this solution is typically intractable for neural networks of practical sizes. One approach to make the problem tractable is to approximate $p(\mathbf{w}|\mathcal{D})$ using a variational posterior $q(\mathbf{w}|\theta)$ whose functional form is known and parameterized by θ. How well the variational posterior $p(\mathbf{w}|\theta)$ fits the true posterior $p(\mathbf{w}|\mathcal{D})$ is measured using the Kullback-Leibler (KL) divergence [21], given below:

$$KL\big(q(\mathbf{w}|\theta)||p(\mathbf{w}|\mathcal{D})\big) = \int q(\mathbf{w}|\theta) \cdot log\bigg(\frac{q(\mathbf{w}|\theta)}{p(\mathbf{w}|\mathcal{D})}\bigg)\partial\mathbf{w} = \mathbb{E}_{q(\mathbf{w}|\theta)} log\bigg(\frac{q(\mathbf{w}|\theta)}{p(\mathbf{w}|\mathcal{D})}\bigg), \tag{4.20}$$

where the intractable posterior $p(\mathbf{w}|\mathcal{D})$ is a component of the expression. The KL divergence in equation 4.20 measures the *average difference in log-probability* between the variational posterior q and the true, intractable posterior p. Moreover, KL divergence does not have the same properties as other distance metrics (e.g. Euclidean), as the distance is asymmetric and dependent on the direction, i.e. $KL(q,p) \neq KL(p,q)$. The metric does however have a lower bound, when the two distributions are equal, then the divergence is the same in both directions; $KL(q,p) = 0 = KL(p,q)$ [25]. Due to the intractability, equation 4.20 needs to be expressed differently. By replacing $p(\mathbf{w}|\mathcal{D})$ with it's corresponding counterpart on the right hand side of Bayes theorem, equation 4.20 can be expressed as:

$$KL\big(q(\mathbf{w}|\theta)||p(\mathbf{w}|\mathcal{D})\big) = \mathbb{E}_{q(\mathbf{w}|\theta)}\Big[log\big(q(\mathbf{w}|\theta)\big) - log\big(p(\mathbf{w})\big) - log\big(p(\mathcal{D}|\mathbf{w})\big)\Big] - log\big(p(\mathcal{D})\big), \tag{4.21}$$

where the term $p(\mathcal{D})$ is not dependent on the parameters θ and can thus be placed outside the expectation term. Moreover, the first two terms within the expectation can – from the definition of KL divergence – be rewritten as a KL divergence; $KL\big(q(\mathbf{w}|\theta)||p(\mathbf{w})\big)$, leaving the third term as is. Thus, to find the optimal parameters θ^* in the variational posterior, optimization is performed by minimizing the parts of equation 4.21 that depends on θ:

$$\theta^* = \arg\min_{\theta}\Big\{KL\big(q(\mathbf{w}|\theta)||p(\mathbf{w})\big) - \mathbb{E}_{q(\mathbf{w}|\theta)} log\big(p(\mathcal{D}|\mathbf{w})\big)\Big\}, \tag{4.22}$$

where all terms are tractable. The terms in the argument on the right-hand side in 4.22 correspond to a prior-dependent part; $KL(q(\mathbf{w}|\theta)||p(\mathbf{w}))$ referred to as the *complexity cost* and a data-dependent part; $log(p(\mathcal{D}|\mathbf{w}))$ referred to as the *likelihood cost*, and is as a whole referred to as the *variational free energy* [2] [14] [13]:

$$\mathcal{F}(\mathcal{D},\theta) = KL\big(q(\mathbf{w}|\theta)||p(\mathbf{w})\big) - \mathbb{E}_{q(\mathbf{w}|\theta)} log\big(p(\mathcal{D}|\mathbf{w})\big) \tag{4.23}$$

The variational free energy – or cost function – in equation 4.23 describes a trade-off between regularization on the weights via the complexity cost and the eagerness to fit the data via the likelihood cost [2]. There is a specific relationship between the variational free energy and the Evidence LOwer Bound (ELBO), such that $ELBO = -\mathcal{F}$. Note that the complexity cost in equation 4.23 is only defined in closed form for some specific prior-posterior combinations and is computationally expensive.

4.3.1 Bayes By Backprop

A specific breed of Bayesian variational neural networks that utilize the theoretical aspects given in section 4.3 are *Bayes By Backprop* neural networks [2]. The aim is to perform

back-propagation on the weights $\mathbf{w}$, which are distributions with parameters mean μ and standard deviation σ. To allow only positive standard deviations, σ is parameterized as $\sigma = log(1 + exp(\rho))$. Note that each weight receives exactly one mean and one standard deviation parameter. The first aim is to formulate an objective function, consisting of the weights and the parameters prone to optimization, such that $f(\mathbf{w}, \theta) \approx \mathcal{F}(\mathcal{D}, \theta)$. The objective function proposed in [2] is defined as:

$$f(\mathbf{w}, \theta) = log\big[q(\mathbf{w}|\theta)\big] - log\big[p(\mathbf{w})p(\mathcal{D}|\mathbf{w})\big] \tag{4.24}$$

The objective function in equation 4.24 denotes an expectation w.r.t the variational posterior, and is subject to minimization. Having defined the objective function, the aim is to perform back-propagation [32] on the neural network weights, where gradients of the objective function directs the optimization procedure towards a minimum. By introducing a stochastic parameter $\epsilon \sim q(\epsilon)$, one per weight, and treating $\theta = \{\mu, \rho\}$ as deterministic, a diagonal Gaussian variational posterior can be described as a deterministic mapping $q(\mathbf{w}, \theta) = t(\theta, \epsilon) = \mu + \sigma \odot \epsilon$ and $q(\mathbf{w}|\theta)\partial\theta = q(\epsilon)\partial\epsilon$, where $\odot$ is element-wise multiplication. The particular re-parameterization is known as the *re-parameterization trick* [19], and is similar to the non-centered paramterization in equation 4.19. Re-parameterizing the variational posterior this way allows to express the derivative of the variational posterior $q(\theta|\mathbf{w})$ as the expectation of the parameter-free noise $q(\epsilon)$:

$$\begin{aligned}
\frac{\partial}{\partial\theta}\mathbb{E}_{q(\mathbf{w}|\theta)}\big[f(\mathbf{w}, \theta)\big] &= \frac{\partial}{\partial\theta}\int f(\mathbf{w}, \theta) \cdot q(\mathbf{w}|\theta)\partial\mathbf{w} \\
&= \frac{\partial}{\partial\theta}\int f(\mathbf{w}, \theta) \cdot p(\epsilon)\partial\epsilon \\
&= \mathbb{E}_{q(\epsilon)}\left[\frac{\partial f(\mathbf{w}, \theta)}{\partial\mathbf{w}}\frac{\partial\mathbf{w}}{\partial\theta} + \frac{\partial f(\mathbf{w}, \theta)}{\partial\theta}\right]
\end{aligned} \tag{4.25}$$

The re-parameterization trick in combination with the specific objective function from equation 4.24 have resulted in an unbiased gradient expression for the objective function, referred to as *unbiased Monte Carlo gradients* [2]. Furthermore, as equation 4.25 clearly suggests; taking a sample from the variational posterior $q(\mathbf{w}|\theta)$ can be performed by shifting and scaling a sample from the parameter-free noise $q(\epsilon)$ and this particular sample is exactly the gradient that is sought for back-propagation. With unbiased gradients, a Monte Carlo approach is proposed as an approximation of the exact cost in equation 4.23 [2]. For $i = 1, 2, ..., S$ samples, the expression becomes:

$$\mathcal{F}(\mathcal{D}, \theta) \approx \sum_{i=1}^{S} log\big(q(\mathbf{w}^{(i)}|\theta)\big) - log\big(p(\mathbf{w}^{(i)})\big) - log\big(p(\mathcal{D}|\mathbf{w}^{(i)})\big) \tag{4.26}$$

Note that all terms of the approximated cost function depends upon weights drawn from the variational posterior $\mathbf{w}^{(i)}$ and that the expression clearly avoids computing the complexity cost from equation 4.22 in closed form. Having liberated the neural network from closed form solutions of the complexity cost, choice of priors for the weights $p(\mathbf{w})$ are less constrained. A full overwiew of the optimization procedure proposed by [2] is given in algorithm 4, where α denotes the learning rate.

The parts $\frac{\partial f(\mathbf{w}, \theta)}{\partial\mathbf{w}}$ in step 5 and 6 of algorithm 4 are shared among the network parameters and correspond exactly to the gradients of a standard (frequentist) neural network [2]. Scaling and shifting the gradients as in the algorithm above thus allows to learn both the mean μ and the (parametrized) standard deviation $log(1 + exp(\rho))$. Additionally, adding noise to the weights via ϵ is one way of obtaining regularization and is a technique that encourages stability in the function that is being evaluated [12]. Thus, Bayesian neural networks of the

Algorithm 4 Bayes By Backprop optimization procedure

1. $\epsilon \sim \mathcal{N}(0,1)$
2. $\mathbf{w} = \mu + log(1 + exp(\rho)) \odot \epsilon$
3. $\theta = \{\mu, \rho\}$
4. $f(\mathbf{w}, \theta) = log\ q(\mathbf{w}|\theta) - log\ p(\mathbf{w})p(\mathcal{D}|\mathbf{w})$
5. $\nabla_\mu = \frac{\partial f(\mathbf{w},\theta)}{\partial \mathbf{w}} + \frac{\partial f(\mathbf{w},\theta)}{\partial \mu}$
6. $\nabla_\rho = \frac{\partial f(\mathbf{w},\theta)}{\partial \mathbf{w}} \frac{\epsilon}{1+exp(-\rho)} + \frac{\partial f(\mathbf{w},\theta)}{\partial \rho}$
7. $\mu = \mu + \alpha \cdot \nabla_\mu$
   $\rho = \rho + \alpha \cdot \nabla_\rho$

kind presented in this section are potentially regularizing in two ways; via the priors and and via the random noise associated with the weights.

4.3.2 Stochastic optimization

Stochastic gradient descent may be regarded as an improved way of learning the gradient found in back-propagation, where small chunks of data called *mini-batches* are used to evaluate the gradients. As a consequence of the usage of mini-batches, the gradient is noisy, which can lead to faster learning [12]. Stochastic gradient descent on the approximated cost from equation 4.26 may be used by splitting the data $\mathcal{D}$ into $m^{'}$ equally sized subsets $\mathcal{D}_j$ and assign each partition a partition-weight π_j, where $\sum_j \pi_j = 1$ and $\pi_j \in [0,1]$ [2]. Note that only the complexity cost is scaled with the partition-weights, as given below:

$$\mathcal{F}(\mathcal{D}, \theta) \approx \pi_j \Big[\sum_{i=1}^{S} log\big(q(\mathbf{w}^{(i)}|\theta)\big) - log\big(p(\mathbf{w}^{(i)})\big) \Big] - log\big(p(\mathcal{D}_j|\mathbf{w}^{(i)})\big) \tag{4.27}$$

Note that by assigning the partition-weights decreasingly, the scochastic optimization mimics Bayesian inference; when little data is seen, the priors plays a significant role and as data increases the prior becomes less influential. Using a decreasing scheme for the partition-weights is suggested as a way to avoid getting stuck in local minima and avoid poor parameters [2].

4.3.3 Activation function

Bayesian feed-forward networks of the kind presented here use – like ordinary neural networks – activation function(s) between the input and hidden layer as well as between hidden layers. Choice of activation function is dependent upon the given problem. However, some activation functions have attractive properties. The rectified linear unit (ReLU) is a non-linear activation with close to linear properties, and thus the optimization surface may be easier to optimize [12]. A formula for the ReLU is given below.

$$h(z) = max(0, z) \tag{4.28}$$

In equation 4.28, the input z could either denote the input vector $\mathbf{x}$, multiplied by the neural network weights $\mathbf{w}$, or the output from a previous layer.

4.3.4 Optimizers

The last step of the optimization procedure given in algorithm 4 corresponds to the standard gradient descent, where learning is scaled by some pre-selected learning-rate (or stepsize) α. The learning rate itself is hard to fine-tune, i.e. challenging to specify for the user. Using fixed

stepsizes in all directions may be a drawback, as the cost function may be more sensitive to steps in some directions of a high-dimensional optimization surface than it is to others. As such, it may be beneficial to use an optimization algorithm that adapts the momentum in different directions according to the corresponding sensitivity. [12]

The Adam algorithm [20] is an alternative to standard gradient descent that utilizes additional information on the first-order and second-order momentum for each parameter separately and computes bias-corrections of such before taking an optimization step. Usage of the Adam optimizer requires a stepsize, and several other parameters that can be tweaked to other values than the suggested defaults. In this thesis, the default suggestions for the additional parameters are kept. The full Adam optimizer presented in [20] is presented in algorithm 5.

Algorithm 5 Adam optimizer

Require: α: Stepsize
Require: ϵ: Constant for numerical stabilization (default = 10^{-8})
Require: $\beta_1 \in [0, 1)$: Exponential decay-rate for first moment (default = 0.9)
Require: $\beta_2 \in [0, 1)$: Exponential decay-rate for second moment (default = 0.999)
Require: $f(\theta)$: Objective function
Require: θ_0: Initial parameter vector

$m_0 \leftarrow 0$
$v_0 \leftarrow 0$
$t \leftarrow 0$
while θ_t not converged **do**
  $t \leftarrow t + 1$
  $g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$ (Gradients w.r.t objective function, at timestep t)
  $m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Biased first moment estimate)
  $v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ (Biased second moment estimate)
  $\hat{m}_t \leftarrow m_t / (1 - \beta_1^t)$ (Bias-corrected first moment estimate)
  $\hat{v}_t \leftarrow v_t / (1 - \beta_2^t)$ (Bias-corrected second moment estimate)
  $\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \hat{m}_t / (\sqrt{\hat{v}_t} + \delta)$ (Update step)
end while
return θ_t (Final parameters)

Note that evaluation of g_t in algorithm 5 above corresponds to algorithm 4, excluding the updating step (step 7). With individual treatment for all parameters, g_t, m_t etc. are vectors, and g_t^2 denotes the element-wise square of gradients for all parameters [20]. Using mini-batch optimization, the Adam algorithm is evaluated $m^{'}$ times before the complete data is back-propagated. Successfully updating all $m^{'}$ mini-batches corresponds to one *epoch*.

4.3.5 Model evaluation

Model evaluation can be performed by monitoring the cost function of the network for each epoch, for both training data and evaluation data simultaneously. The fitting procedure is performed by minimizing a cost function at each epoch, which – as discussed – includes regularization on the weights. By only storing parameters of the network on the basis of the validation set cost at each epoch, then *early stopping* is achieved [12]. This kind of stopping criteria always ensures the best generalizing ability of a neural network. If several models are

fitted on the same data, then the cost function over epochs can be compared, in order to see which model performs best. Additionally, the output can be used to compute MSE and R^2 in accordance with equation 4.3 and 4.4.

4.3.6 PyTorch

The machine learning framework PyTorch [31] allows for simple specification of the feed-forward structure applied in neural networks, and back-propagation through *automatic differentiation* [30] of the objective function. In other words, the program computes gradients of any given objective function, meaning that no closed form gradients are required to learn the neural networks. Furthermore, optimizers such as Adam and activation functions such as ReLU are pre-programmed.

5 Results

5.1 Bayesian hierarchical regression

To ensure that the hierarchical linear models included in this thesis are adequately fitted, the HMC sampler is checked according to the chain diagnostics presented in section 4.2.3.1. During model prototyping, the models were fitted using several chains to ensure that potential different starting positions gave similar results. Note that prototyping models are not included in the report. The models presented in this chapter are fitted using one chain each. The results presented fulfils the following requirements: no divergent transitions, no iterations that exceed maximum tree-depth and trace plots indicating that the chains have mixed well. All models are fitted using 500 warm-up iterations, and 1500 sampling iterations. All N_{eff} are greater than 200, but all $\hat{R}$ are not within $[0.99, 1.01]$. Careful checks of the chains indicate that despite the somewhat large $\hat{R}$ for some parameters, the chains seem to be sampling sufficiently well for inference. For specifics of trace plots, effective draws and the Gelman-Rubin diagnostics for relevant parameters, see Appendix A.

5.1.1 Model specifications

All models are fitted using the same prior-setup for the varying intercepts (α_j), consisting of a weakly informative, non-centered prior as in equation 4.19. The prior mean (20) for α is chosen empirically from the data mean and given wide tails with a high standard deviation (5). The motivation is to gently nudge the posterior mean $\tilde{\alpha}$ using an empirically reasonable value, since no relevant prior for the intercept could be elicitated. Identical weakly informative priors for the varying intercept standard deviation parameter α_σ and the regression noise σ are chosen not to steer ICC calculated by equation 4.6 in any particular direction. For predictions, the varying intercept mean parameter α_μ is used as partitioning of new data (patients) into the correct cluster is not possible. This thesis evaluates three different priors for slope parameters – leading to three models – with all other priors shared. Variable selection is performed according to common variables in BIA equations in [1], in combination with the suggested phase angle (equation 3.4). The frequency of 250 kHz proved during prototyping to give comparatively better results to other frequencies and is hence the only frequency utilized. A general model specification for the parametric form is given below, followed by the three specifications that separates the models.

$$
\begin{aligned}
y_{ij} &\sim \mathit{Normal}(\mu_{ij}, \sigma) \\
\mu_{ij} &= \alpha_j + \gamma \cdot R^{in}_{250} + \beta_3 \cdot D_{Male} + \beta_4 \cdot \varphi^{in}_{250} + \beta_5 \cdot \mathit{Weight} + \beta_6 \cdot \mathit{Height} + \beta_7 \cdot \mathit{Age} \\
\gamma &= \beta_1 + \beta_2 \cdot D_{Male} \\
\alpha_j &= \alpha + \sigma_\alpha \cdot \eta_j \\
\alpha &\sim \mathit{Normal}(20, 5) \\
\sigma_\alpha &\sim \mathit{Half\text{-}Normal}(0, 2) \\
\eta_j &\sim \mathit{Normal}(0, 1) \\
\beta_k &\sim \mathit{Model\ specific} \\
\sigma &\sim \mathit{Half\text{-}Normal}(0, 2)
\end{aligned}
$$

Regularizing priors for slope parameters

A Laplace prior is used to penalize large coefficients for slope parameters. Thus, completing the previous model specifications is performed using a Laplace prior with zero mean and standard deviation $\sigma = 0.1$. This prior is heavily regularizing, placing negligible mass outside of ± 0.2, meaning that for any slope coefficient to be large, the corresponding variable needs to have a really high influence on the target variable. The model specification is completed with:

$$\beta_k \sim \mathit{Laplace}_K(0, 0.1) \tag{5.1}$$

Weakly informative priors for slope parameters

In models with standardized features and non-standardized target, a slope parameter $\beta_k = 1$ is readily interpreted as; *a one standard deviation increase in variable k influences the target variable with one unit increase*. With that in mind, independent Cauchy priors for slope parameters with mean 0 and standard deviation 2 allows for parameters to move free due to the wide tails of a Cauchy distribution. The model specification is thus completed with:

$$\beta_k \sim \mathit{Cauchy}_K(0, 2) \tag{5.2}$$

Elicitated priors for slope parameters

Based on the commissioner's domain specific knowledge, a set of priors for the slope parameters were elicitated. Elicitation was performed in such a way that a list of model variables was included, and with comments regarding each variable, a relevant prior was created. The comments were given as a range from highly positive influence ($++$) to highly negative influence ($--$). To formulate priors over the specified range, only Gaussian priors was considered, with the following translation scheme from the given comment into priors:

Comment	**Prior translation**
$++$	$\mu = 4$
$+$	$\mu = 2$
0	$\mu = 0$
$-$	$\mu = -2$
$--$	$\mu = -4$

No correlations between parameters or variables are hypothesized, meaning that each slope parameter receives a prior independent from all others. Unit variance of a Gaussian distribution in combination with the chosen means results in an informative prior, on the boundary to regularizing. The missing part of the model specification is filled below:

$$\begin{aligned}
\beta_1 &\sim Normal(2, 1) \\
\beta_2 &\sim Normal(4, 1) \\
\beta_3 &\sim Normal(2, 1) \\
\beta_4 &\sim Normal(-2, 1) \\
\beta_5 &\sim Normal(4, 1) \\
\beta_6 &\sim Normal(2, 1) \\
\beta_7 &\sim Normal(-2, 1)
\end{aligned} \tag{5.3}$$

5.1.2 Parameter estimates

Below, posterior parameter estimates that are part of the linear model (link function) are displayed for all fitted models. From here on, the models will be referred to as *regularized model*, *weakly informative model* and *elicitated model*, according to the specifications given in 5.1 to 5.3.

		Regularized		Weakly informative		Elicitated	
Parameter	Variable	Mean	Sd	Mean	Sd	Mean	Sd
α	$Intercept$	19.54	0.06	19.51	0.06	19.51	0.06
β_1	R^{in}_{250}	1.19	0.06	1.17	0.06	1.16	0.06
β_2	$R^{in}_{250} \cdot D_{Male}$	1.28	0.07	1.31	0.07	1.33	0.07
β_3	D_{Male}	3.01	0.10	3.07	0.10	3.07	0.11
β_4	φ^{in}_{250}	0.46	0.06	0.46	0.06	0.44	0.06
β_5	$Weight$	0.64	0.07	0.65	0.08	0.68	0.07
β_6	$Height$	1.52	0.05	1.51	0.05	1.51	0.05
β_7	Age	-0.59	0.03	-0.60	0.03	-0.60	0.03

Table 5.1: Parameter estimates for hierarchical linear models.

In table 5.1, different prior specifications did not result in much of a difference between models, with all parameters within one standard deviation from the corresponding parameter in the other models. The linear interaction suggests that males have a steeper slope for $ALST$ than females, as the linear interaction parameter β_2 for $R^{in}_{250} \cdot D_{Male}$ differs from 0 when $D_{Male} \neq 0$, which happens for males. The most notable difference found between the three models presented is that the regularized model trades a higher intercept α for less differences between genders β_3.

5.1.3 Performance on training data

In this section, all three models are evaluated on various checks on the training set. Below, plots displaying Bayesian R^2, MSE and ICC for all three fitted models are presented. The center line in each subplot below denotes the median, and the corresponding value on the x-axis displays the median rounded to the third decimal. Similarly, the left and right lines denote the 5th and 95th percentile, rounded to the third decimal.

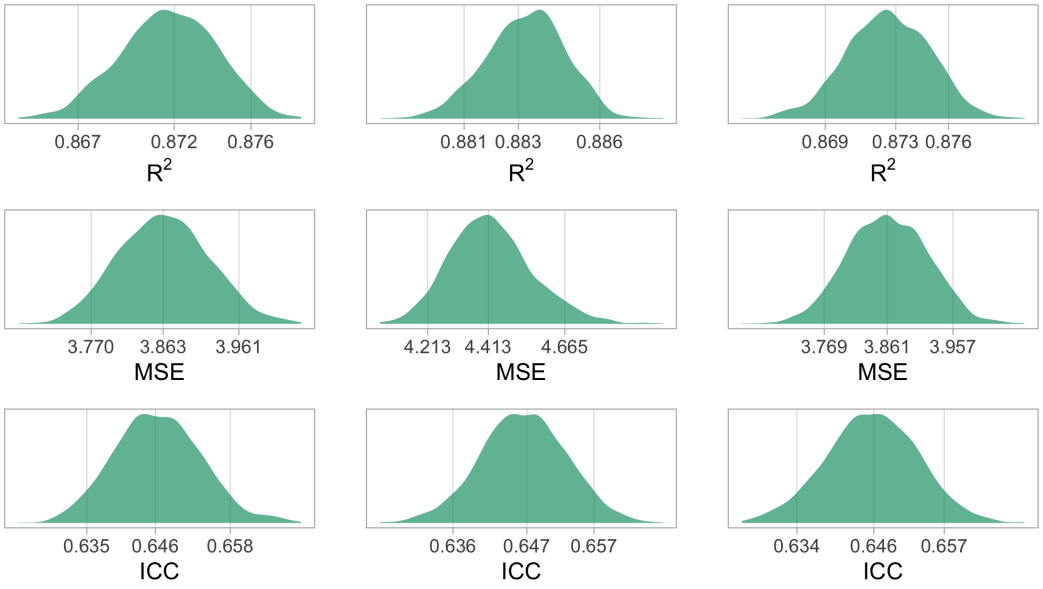


Figure 5.1: Bayesian R^2, MSE and ICC for all hierarchical linear models. Left: regularizing model, center: weakly informative model, right elicitated model.

The differences between models in figure 5.1 are sparse, as all three metrics are similar for all three models. If any metric is to stand out, it is the difference in MSE, which is higher for the weakly informative model, and possibly the wider range the three metrics display for the weakly informative model. Posterior metrics for ICC indicates that the posterior variation-between ($\tilde{\sigma}_\alpha$) is larger than the posterior additive noise ($\tilde{\sigma}$), suggesting that a multilevel-model is a necessity. As the ICC metric is a posterior metric, based on the particular parameters of the model, and all mass is located to the right of 0.5, there is a 100 % probability that $\tilde{\sigma}_\alpha > \tilde{\sigma}$, according to all three models. To estimate out-of-sample fit, the estimates discussed in section 4.2.2 are computed on the log-likelihood matrix for each posterior draw of the posterior model parameters $\tilde{\theta}$.

Model	$\nabla\widehat{lppd}$	$\widehat{lppd}$	$\hat{p}_{waic}$	$WAIC$	$se(WAIC)$
Elicitated	0.00	-12080.31	355.33	24160.63	517.03
Regularizing	-27.66	-12107.98	346.12	24215.95	519.35
Weakly informative	-1839.88	-13920.20	778.45	27840.39	677.00

Table 5.2: Estimates on out-of-sample predictive abilities of hierarchical linear models.

In table 5.2, $WAIC$, as calculated by equation 4.9 is a measure for expected deviance on new data, $\hat{p}\_waic$, as calculated by equation 4.8 is the number of effective parameters and $\nabla\widehat{lppd}$ is the expected log-predictive density as calculated by equation 4.7. The $WAIC$ estimate suggest that the model with elicitated priors for the slope parameters yields best out-of-sample performance and $\hat{p}\_waic$ suggests that the model using weakly informative priors have the highest number of effective model parameters. As the standard error $se(WAIC)$ for the elicitated model is wide enough for the elicitated model $WAIC$ to cover the regularizing model $WAIC$, care should be taken as regarding the elicitated model to have the best out-of-sample performance.

To provide a visual inspection of model performance on training data, two different checks are performed for BMI: (1) the aggregated relative effective sizes presented in equation 4.10 for each model and plotted as a function of BMI and (2) the difference $\tilde{y} - y$ is computed and plotted as a function of BMI, similar to the right side of figure 3.2. The check is performed to see if the various underlying causes of high and low BMI have a confounding effect on predictions generated by the three fitted models.

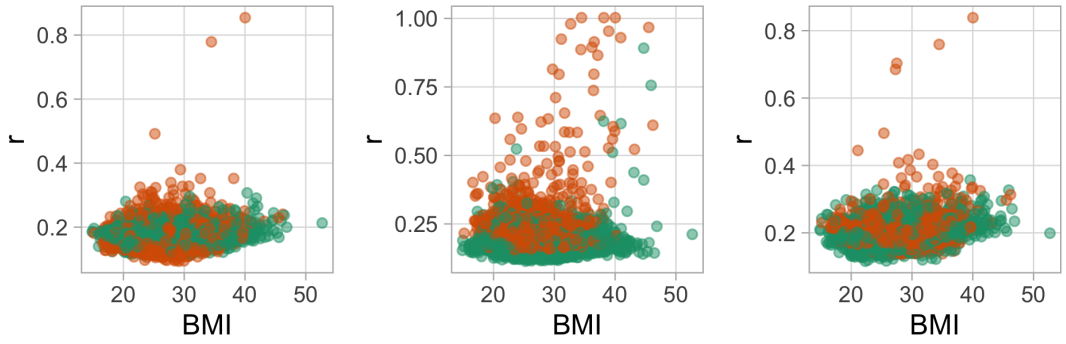


Figure 5.2: Importance ratios plotted over BMI for all hierarchical linear models. Green points correspond to females, orange points correspond to males. Left: regularizing model, center: weakly informative model, right: elicitated model.

In figure 5.2, the difference between models become apparent. The aggregated importance ratios r is not systematically increasing with increasing BMI yet mostly obese subjects receive high values of such. This is especially true for the weakly informative model, which contain more problematic observations. Notably, high ratios is primarily concerned with male subjects, for all models. This is an indication that the model faces a tougher task when predicting males. The difference between posterior predictions and actual data plotted over BMI is shown below.

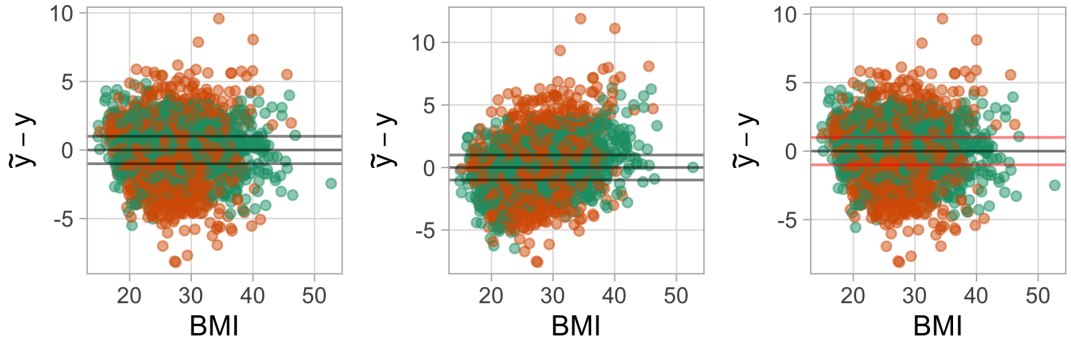


Figure 5.3: Difference $\tilde{y} - y$ plotted over BMI for all hierarchical linear models. Green points correspond to females, orange points correspond to males. Left: regularizing model, center: weakly informative model, right: elicitated model.

No pathological behaviours associated with increased or decreased BMI can be found in figure 5.3 for the regularizing and elicitated model, which are visually close to identical. The weakly informative model shows signs of increasing difference with increasing BMI, i.e. a tendency to over-estimate $ALST$. Most predictions lie within ± 5 kg from the target variable.

However, some concerns can be raised towards all three models ability to accurately estimate $ALST$, as there are a few quite distinct outliers. Further investigations on the posterior predictions $\tilde{y}$ generated by the three models are displayed in figure 5.4.

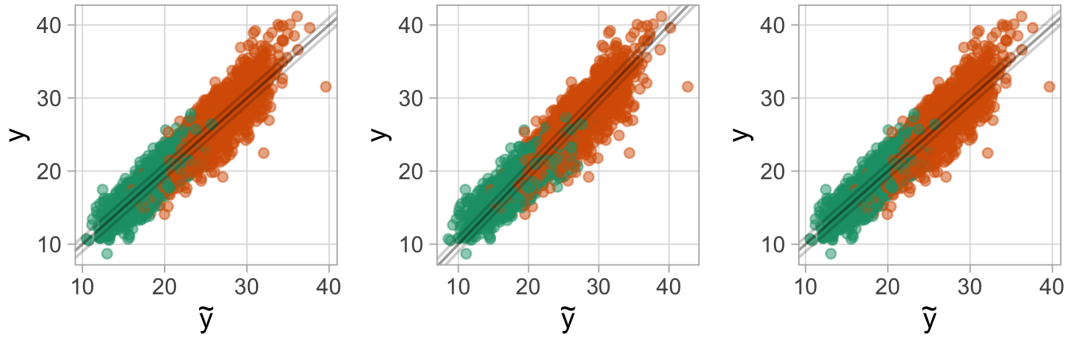


Figure 5.4: Y-vs-Y plots for posterior predictive means on training data. Green dots represent females and orange dots represent males. Left: regularizing model, center: weakly informative model, right: elicitated model.

In figure 5.4, the three models again tend to be very similar, and contain no major differences in terms of predictions. However slight differences are found in the top-right region, where the weakly informative model tends to perform best while the other two models tend to under-estimate $ALST$. It should be noted that the top-right region of muscular subjects contains few observations, so there is a degree of caution that should be taken when regarding the elicitated and regularized models as incapable of predicting high $ALST$.

5.1.4 Performance on validation data

Below, validation data MSE is plotted for all three models. Similar to figure 5.1, the lines denotes 5th, 50th and 95th percentile and the x-axis values correspond to those quantiles, rounded to three decimals.

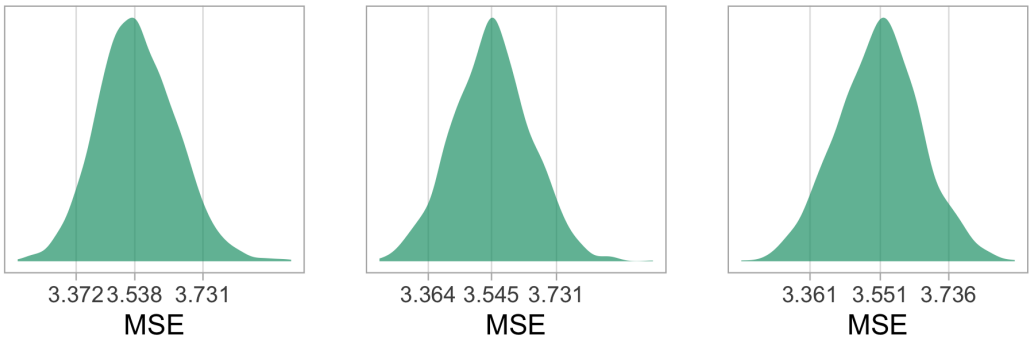


Figure 5.5: MSE for validation data. Left: Regularizing model, center: weakly informative model, right: elicitated model.

Despite the differences found in the WAIC estimates given in table 5.2, figure 5.5 indicates that the weakly informative model has similar predictive performance to the other two models as there is not much difference in MSE. In fact, the weakly informative model displays the second lowest (median) validation set MSE, by a very small margin. The procedure of

plotting BMI as a function of the difference between predictions as in figure 5.3 is repeated for the validation set in figure 5.6.

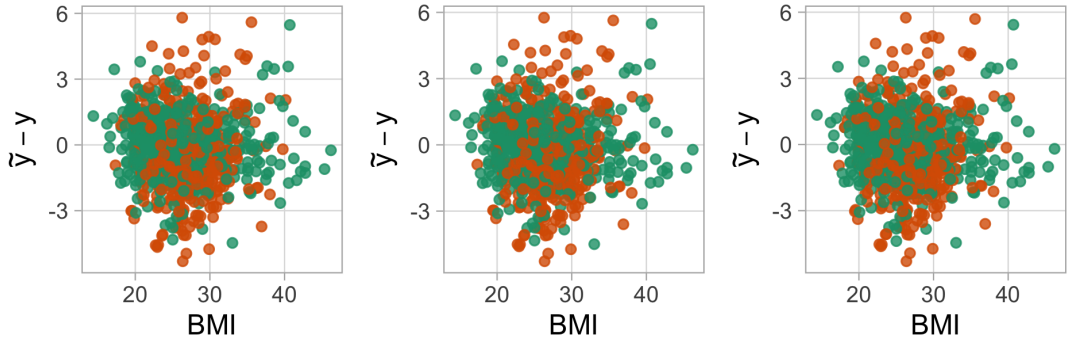


Figure 5.6: Difference $\tilde{y} - y$ plotted over BMI for all hierarchical linear models on validation data. Green points correspond to females, orange points correspond to males. Left: regularizing model, center: weakly informative model, right: elicitated model.

In figure 5.6, no pathological behaviours of the differences between target and prediction over BMI is visible. All models display similar behaviours. Similar to figure 5.4, plots of posterior predictions for all models are computed for the validation set, presented in figure 5.7.

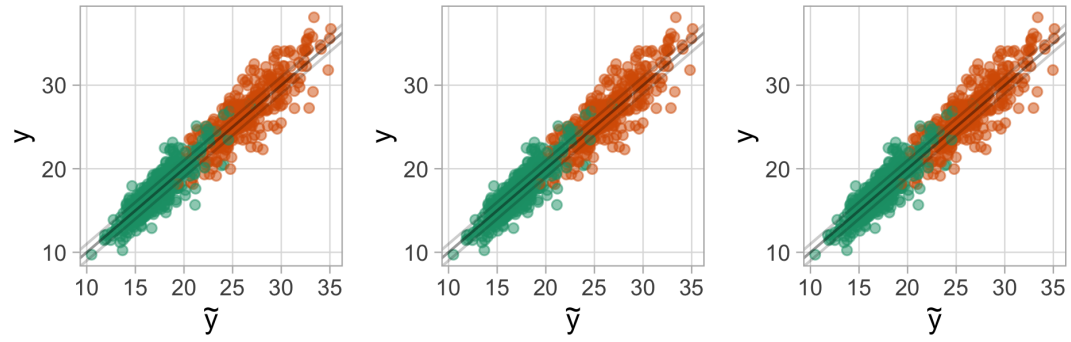


Figure 5.7: Y-vs-Y plots for posterior predictive means on validation data. Light green dots represent females, dark green dots represent males. Left: Laplace model, center: weakly informative model, right: elicitated model.

The predictions given in figure 5.7 indicates that all three fitted models follow the validation data in similar fashion to how the models behaved on training data. The same pattern regarding individuals with high $ALST$ in figure 5.4 is however not present above to the same extent; predicting high $ALST$ does not impose additional difficulties for the hierarchical linear models on validation data.

To see how the predictions perform for underweight, normal weight and obese subjects, the full posterior predictive distribution presented in equation 4.2 is computed for the selected subjects presented in table 3.5. This is presented in figure 5.8, where black dots correspond to the y value for each subject and the distributions show the posterior predictive distribution $\tilde{y}$ given by equation 4.2 for that specific subject.

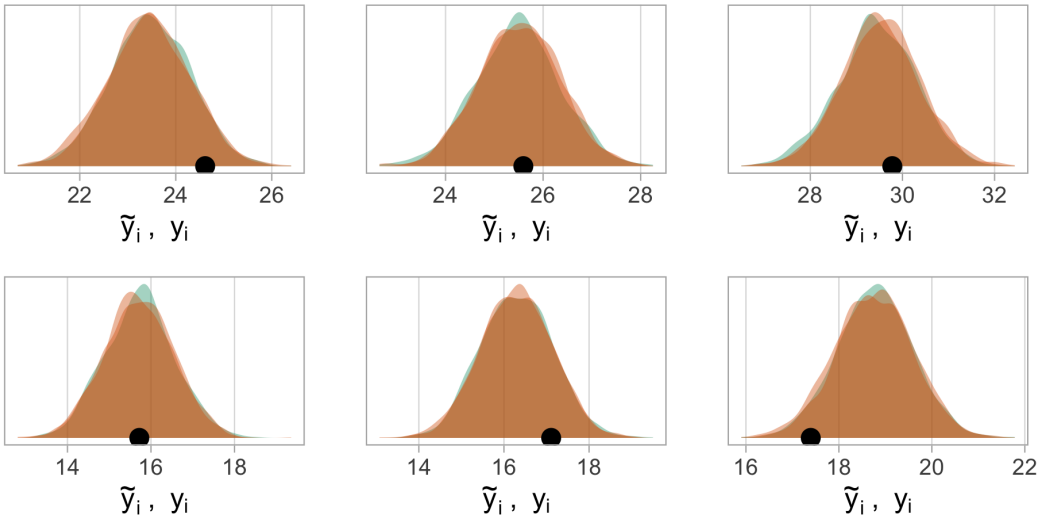


Figure 5.8: Posterior predictive distributions for a selection of subjects with different BMI. From left to right: underweight, normal and obese. Top row: Male subjects, bottom row: female subjects. Gray: regularizing model, orange: weakly informative model, green: elicitated model.

First of, figure 5.8 shows again that in terms of predictions, all three models are very similar. Almost nothing separates the three different (almost indistinguishable) colors for the three models. The over-estimation of $ALST$ in the bottom-right sub-figure and under-estimation of $ALST$ in the top-left sub-figure are problematic signs for the linear models. All six selected individuals are as representative of the respective region of BMI with respect to a couple of other variables. However, the two mentioned subjects correspond to an underweight male with relatively high $ALST$ and an overweight female with relatively low $ALST$. That means the under- and over-estimation is likely (but not guaranteed) to be problematic for underweight males and overweight females, if calculated according to BMI.

5.1.5 Model selection

Based on the out-of-sample deviances presented in table 5.2, the model with elicitated priors on the slope parameters is estimated to have the best predictive ability, although the standard error of WAIC (`se_waic`) is large enough to cover the WAIC estimate of the regularized model. All fitted models have very similar parameter estimates, with primary differences found in the intercept and dummy-variable for males. The weakly informative model display worrying signs for increasing BMI, which is a direct contradiction to the aim of this thesis and is ruled out. For the elicitated and regularized model, no differences in visual performance is detected other than for the relative effective sizes in figure 5.2, which is in favour of the regularized model due to the lower amount of problematic observations. Hence, the regularized model is considered the most suitable, and selected. In figure 5.9, an Y-vs-Y plot is displayed on the left-hand side in combination with test set MSE on the right-hand side for the test set.

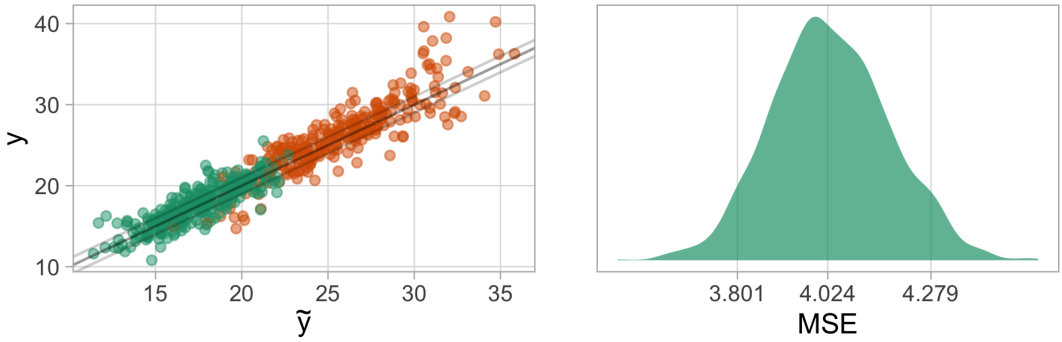


Figure 5.9: Test set Y-vs-Y (left) and test set MSE (right) for the regularized model.

In figure 5.9, problematic predictions can again be found in subjects with much muscles. The large deviations have resulted in MSE that is slightly higher than that of the validation set. Also, as there are fewer observations in the test set, large deviations become more influential. The average MSE of the posterior distribution is approximately 4.0, with an approximate 90 % probability interval; $3.8 < MSE < 4.28$. I.e. it is 9 times more probable that test set MSE is within the intervals than outside.

5.2 Bayesian neural networks

The neural networks presented in this section share variable selection and hyper-parameter settings. The variables used are can all be found in table 3.3; all 20 resistance (R), all 20 reactance (Xc), all 4 entropy-based phase angles (φ_f^e), the entropy distance (D_φ), both gender dummies (D_{Male}, D_{Female}) and finally the anthropometric variables Age, $Height$, $Weight$ and BMI. No standardization is performed on the target (y). The Adam optimizer is used in combination with the ReLU activation function, and all networks have three hidden layers with 8 hidden units each. All networks have a maximum training of 1000 epochs, where early stopping is used according to the optimal validation set cost. A mini-batch configuration where each mini-batch of size 100 is sampled uniformly is chosen, which results in 40 mini-batches for the training set and 8 for the validation set. Thus, the natural ordering that occurs due to repeated visits is most likely removed and evaluation of the likelihood cost may be evaluated under the assumption of i.i.d observations. The particular weight-scheme for each mini-batch discussed in section 4.3.2 is set according to the proposed scheme in [2]:

$$\pi_i = \frac{2^{m'-i}}{2^{m'}-1}, \tag{5.4}$$

where $i = 1, 2, ..., m'$. For the likelihood cost presented in equation 4.23, the squared loss is used. Both neural networks are initialized with means (μ_j) for biases and weights sampled from a Gaussian distribution with zero-mean and standard deviation of 0.1, i.e. tightly concentrated around 0. Log standard deviation parameters (ρ_j) are initialized uniformly in the interval $[-6, -5]$. Empirical experiments proved that the initialization described gave consistent results, under repeated trials with different random seeds. The parameter-free noise is distributed as $p(\epsilon) \sim \mathcal{N}(0, 1)$. Differences in the neural networks can be found within the prior-specification for the weights. Two different set-ups are tested, one Gaussian prior and one Laplace prior:

$$p_L(\mathbf{w}) \sim \prod_{i=1}^{N} Laplace(\mathbf{w}^{(i)}|0, 0.5)$$
$$p_N(\mathbf{w}) \sim \prod_{i=1}^{N} Normal(\mathbf{w}^{(i)}|0, 0.5)$$

where N is the total number of weights and biases in the network. Henceforth, the neural network with Gaussian prior for the weights will be referred to as the *Gaussian NN* and the neural network with Laplace priors for the weights will be referred to as the *Laplacean NN*.

5.2.1 Fit metrics

The variational free energy for both the Gaussian NN and Laplacean NN are plotted below. Here, the first 100 epochs are trimmed, as the loss drastically decreases during these epochs and deteriorate any visual interpretations due to the difference in loss function, which decreases with several orders of magnitude only a couple of epochs after initialization.

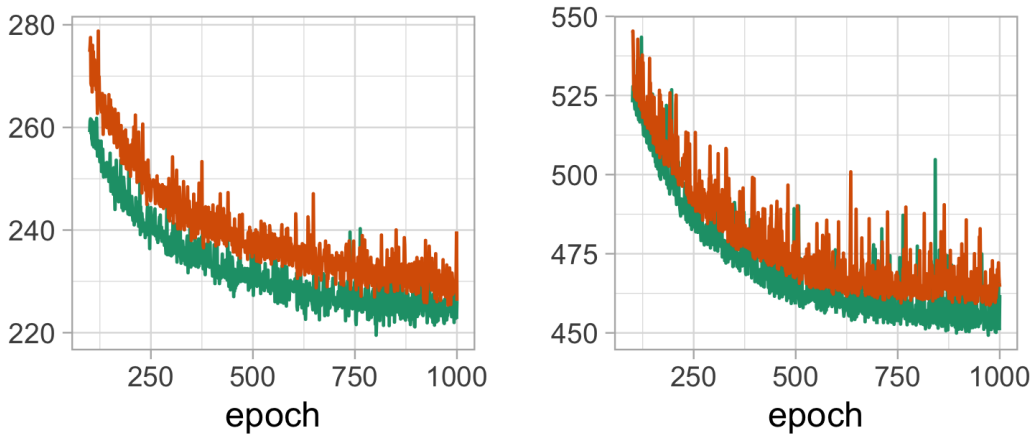


Figure 5.10: Variational free energy for the Gaussian NN (orange) and the Laplacean NN (green). Left: training data, right: validation data.

Throughout most displayed epochs in figure 5.10, the Laplace NN outperforms the Gaussian NN with respect to the cost function. Early stopping was initiated at epoch 974 for the Gaussian NN and at epoch 971 for the Laplacean NN. Both neural networks display a trend that indicates the cost function keeps decreasing, which may be interpreted as more iterations are required. In appendix A, the corresponding plots for complexity cost and likelihood cost can be found, which suggest that the likelihood cost has reached its minimum, and that only the complexity cost decreases. The likelihood cost is most important for predictions, and thus the number of epochs that are run to obtain the above results are considered enough. In figure 5.11, distributions for point estimates of means and standard deviations for the weights are plotted.

In figure 5.11, most weights are dispersed around 0, which is where both models have the highest mass for posterior means $\tilde{\mu}$, and the standard deviations shows that most standard deviations for the weights are found around $\tilde{\sigma} \approx 0.7$. Differences between parameters in between the are sparse, as the difference between the different distributions in figure 5.11 are sparse.

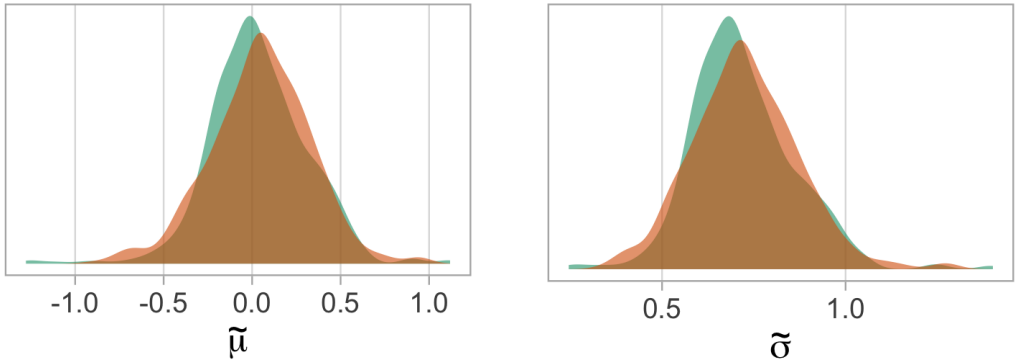


Figure 5.11: Posterior means and standard deviations for the Gaussian NN (orange) and Laplacean NN (green). Left: means, right: standard deviation.

5.2.2 Performance on training data

Similar to the plots presented throughout section 5.1, predictions from the neural networks are fitted versus the ground truth, to see if the network is able to predict adequately in all regions. However, the plots presented in this section does not provide different colors for gender. Both neural networks can adequately make predictions for both genders, so including this variable only contributes to more cluttering, and is hence scrapped.

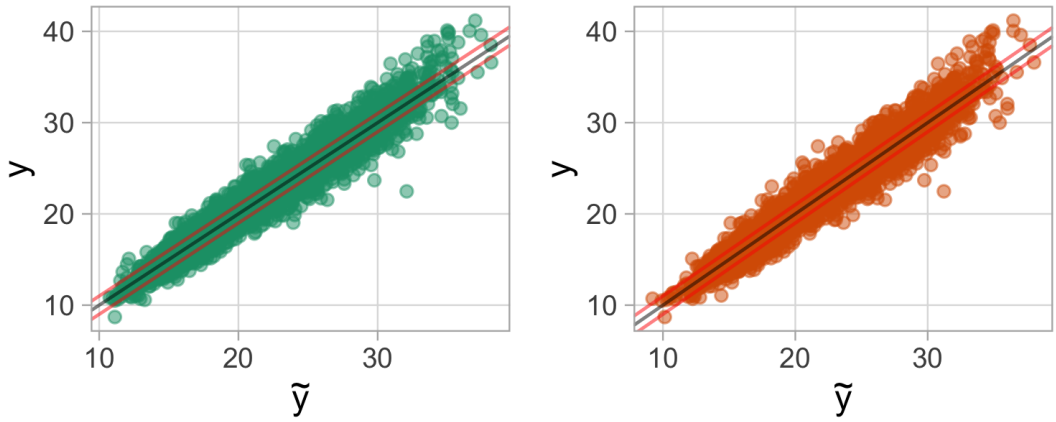


Figure 5.12: Y-vs-Y plots for posterior predictive means on training data. Left: Laplacean NN, right: Gaussian NN.

Both models display high similarities in terms of predictions, as seen in figure 5.12. Both models predict well in low to high ranges of $ALST$, with slight concerns for very muscular individuals in the top-right corner. The predictions above is considerably closer to the truth than in the hierarchical linear models presented in figure 5.4, practically reduced in half. Below, distributions of R^2 and MSE are plotted for both neural networks.

In figure 5.13, differences between the Gaussian NN and Laplacean NN becomes more apparent, as both MSE and R^2 differs between models. Both metrics are calculated based

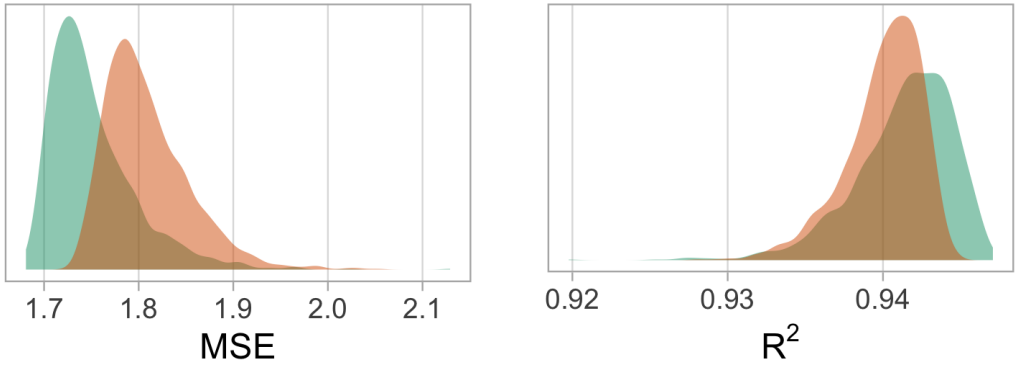


Figure 5.13: Bayesian MSE (left) and R^2 (right) for the two neural networks on training data. Green: Laplacean NN, orange: Gaussian NN.

on 1500 draws from the posterior distribution of each weight. The Laplacean NN produces higher R^2 and lower MSE, yet the differences are quite subtle. It should be noted that the Laplacean NN seemingly has a lower bound for MSE and upper bound for R^2, as the distributions look almost truncated in the tails.

5.2.3 Performance on validation data

Similar to figure 5.7, the posterior predictions from both neural networks are plotted versus y in figure 5.14 for the validation data.

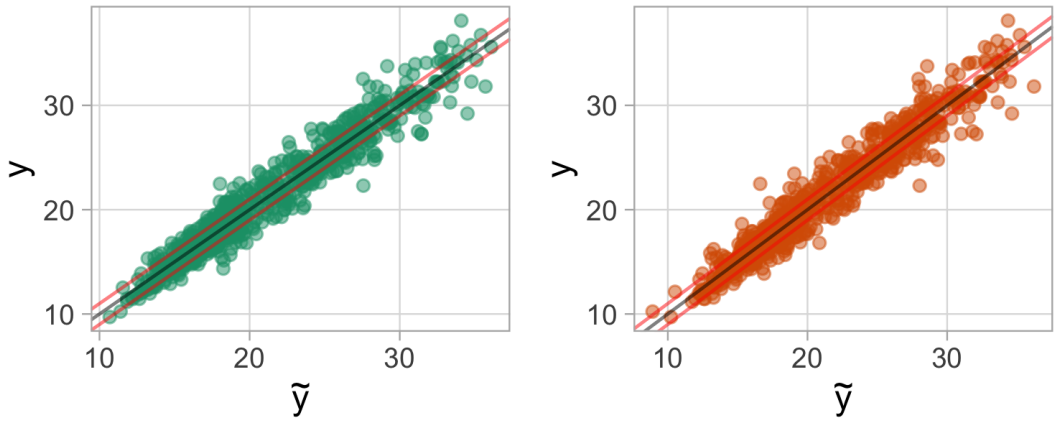


Figure 5.14: Y-vs-Y plots data for posterior predictive means on validation data. Left: Laplacean NN, right: Gaussian NN.

Again, the predictions reveal little difference between models, as both the left-hand side and right-hand side of figure 5.14 look identical. Both models hold the ability to predict all regions of $ALST$ for unseen data. Similar to figure 5.8, the difference between posterior predictions and target variable is plotted as a function of BMI below.

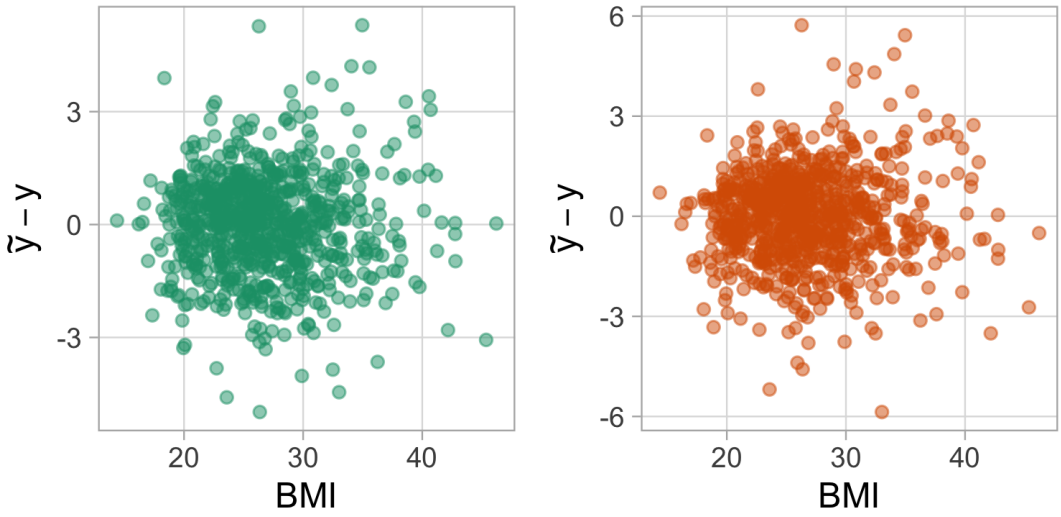


Figure 5.15: Difference $\tilde{y} - y$ plotted over BMI for Laplacean NN (left) and Gaussian NN (right).

In figure 5.15, no pathological behaviours for predictions are associated with low or high BMI is displayed. However, the largest outliers are primarily found in mid-to-high regions of BMI (> 30). On average, both neural networks predict $ALST$ that matches the target for all ranges. Below, MSE is computed for both neural networks on validation data, this time using 1500 new samples from the posterior distribution of each weight.

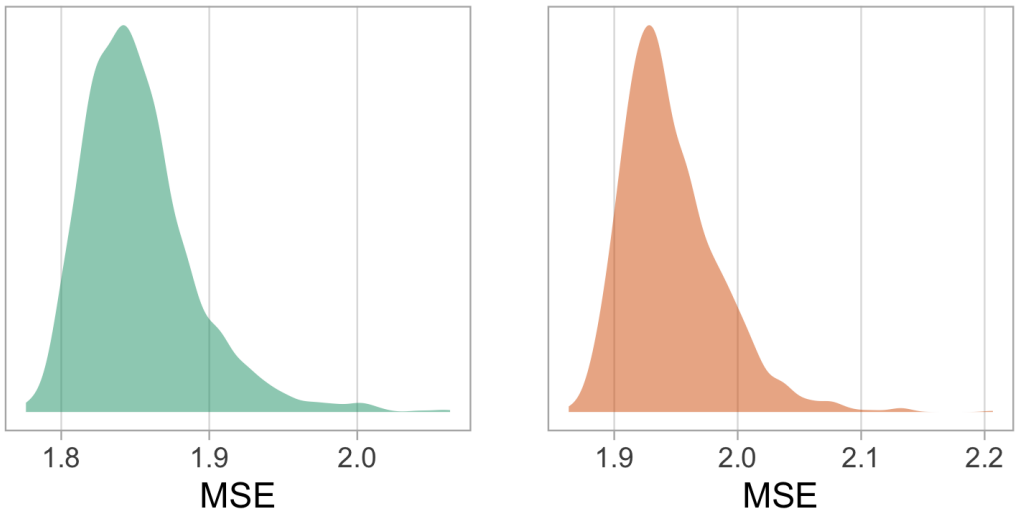


Figure 5.16: Validation set MSE for Laplacean NN (left) and Gaussian NN (right).

In figure 5.16, the differences between the two neural networks once again becomes apparent. Similar to figure 5.13, the Laplacean NN outperforms the Gaussian NN in terms of MSE, with approximately 0.1 units. The lower bound which was apparent for the Laplacean NN in figure 5.13 is not as evident as previously.

5.2.4 Model selection

As both neural networks display similar predictions for all ranges of BMI, and no clear visual differences in Y-vs-Y plots are apparent, the choice comes down to selecting the model with best predictive abilities and lowest cost function. In this case, the Laplacean NN performs best, and is hence selected. Test set MSE and test set Y-vs-Y plot is displayed below, where MSE calculations are based on 1500 samples from the posterior of each weight.

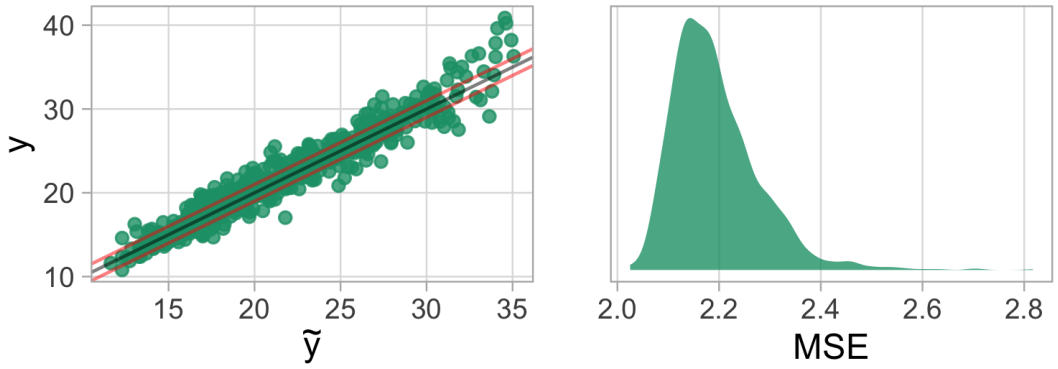


Figure 5.17: Y-vs-Y-plot (left) and MSE (right) for the Laplacean NN on test data.

In figure 5.17 test set MSE has it's mode around 2.15, with high uncertainty displayed in the right tail. The Y-vs-Y-plot displays that some subjects with high $ALST$ remains hard to predict well and receives an under-estimation of $ALST$. Similar to the linear models, the test set MSE is higher than validation MSE.

5.3 Results summary

A table of MSE for BIA predictions, the selected hierarchical linear model and the selected neural network is given below for training data, validation data and test data. Since MSE from the fitted models are distributions, the posterior mode of MSE is taken as a point estimate.

Model	Training	Validation	Test
BIA prediction	6.55	6.30	6.05
Hierarchical linear	3.86	3.51	4.00
Neural network	1.73	1.84	2.14

Table 5.3: MSE for the different data sets.

Table 5.3 suggests that the BIA device predicts MM that is closer to $ALST$ in the test set as compared to the validation set. The neural network and hierarchical linear model suggest otherwise. For the neural network, the fact that generalization error for the validation set would be lower than test set is not strange, as the optimal weights are chosen with the lowest variational free energy on the validation set. However, as the hierarchical linear models suggest, the test set seems to include subjects that are hard to predict. In all cases, the results show that the corresponding error for the BIA device is highest, whereas the neural network error is lowest, by quite some margin.

6 Discussion

6.1 Data

Accuracy of the target variable is only as accurate as the quality of the golden standard method allows it to be, i.e. predictions on ALST given by the DXA device. While DXA is a reliable tool for assessment of ALST, it is not it's primary strength. Therefore, some caution should be taken as to how certain the measurements of the target variable is in reality. For instance, different DXA devices from different manufacturers have been reported to give inconsistent results [6]. It is mentioned several times in this thesis that DXA gives predictions on ALST, seen as the ground truth quantity rather than the exact quantity. Considering the previously mentioned, what is the point of creating a predictive model if the machine that generates the target variable in itself displays uncertainty? Considering that the DXA predictions are stable when compared to the BIA predictions, any upgrade in terms of precision in the target variable allows to create a model that can outshine the device output.

The majority of data points in this thesis are collected while performing routine medical examinations at Sahlgrenska hospital. Although patients have received instructions of how to correct the diet and exercise prior to measurements, no meta-data of such information is accessible. Since the BIA device is recommended to use under very strict conditions [34], and no such conditions can be guaranteed, many of the data points may contain an undesirable degree of noise that have arised from sub-optimal measurement conditions. As a consequence, the degree of noise may be quantified, yet the underlying factors for why the noise is present will remain unknown.

The data cleaning described in section 3.3 included removals of both NA's and measurements that was considered unrealistic, or outliers. The total amount of rows removed from the original set was 421 observations, which corresponds to approximately 7.3 percent of the total data. Essentially, this is information lost. Imputation was considered as an alternative to the plain removals. For instance, HMC can be used for imputation of continuous variables [25]. Imputing NA's would have little effect, as there are only six missing data points. However, for the outlier cases, using imputation of the complex dependencies that exist within the variables would be tough due to the confounding effect that many variables have. There is no point in imputing values if the imputed values are no good. The idea was scrapped

due to the complexity of the task, in comparison to the information that could potentially be gained and the potential errors that the imputation could infer in the data.

Many of the variable transformations in this thesis are related to the variable transformations in previous studies. For instance, the index variable of resistance seems to be the most common transformation [1]. The proposed variable of phase angle index is – to the authors knowledge – not present in the literature and serves as a novel suggestion of data transformation. The entropy-based variables and the (log-) Mahalanobis distance variable are also – to the authors knowledge – unseen in previous literature in the way they are set up. However, using entropy variables in itself is not new, and has been applied to BIA equations before. The reasoning in creating these variables was to quantify the *quality* of a measurement, with the hypothesis that low entropy and high distances would result in low quality measurements. In the neural network setting where these variables were applied, it is hard to investigate the effect which these variables have, but experiments with and without these variables proved that small gains in both MSE and R^2 were made.

6.2 Results

The data set used in this thesis contains more observations than most previous studies and contains a degree of diversity that is representative of an adult Swedish population, including patients with diverse medical conditions. Most data materials in similar studies are much narrower, which allows modelling a specific group of individuals that is of clinical interest. Using a diverse sample as the one obtained in this thesis may lead to lower model performance (e.g. MSE, R^2) compared to similar studies yet result in models with better ability to generalize to a whole population. The results indicate that despite the wide selection, the retrodictive (R^2) and predictive (MSE) metrics for both the hierarchical linear models and the neural networks are comparative to those seen in [1]. These metrics are naturally highly dependent on the parametric form, and no other study models the exact same parametric form as in this thesis.

The Laplace and Cauchy prior specifications were tested to see how sensitive the analysis is to prior information, a form of sensitivity analysis. The elicitated prior was tested as it is the most reasonable method for choosing a prior and to see if domain specific expertise would affect inference. In all three cases, predictions were seemingly identical. The main difference was found in the weakly informative specification, which performed slightly worse to the evaluations that were performed. It seems as the models fitted allows for very little flexibility. A reason for this might be the highly influential resistance index variable that was modelled as an interaction. The results suggest that the models face a tough task of predicting the very highest ranges of ALST. However, in these regions there are very little data.

The neural networks are implemented in PyTorch [31], which does not include pre-programmed probabilistic (Bayesian) layers. Thus, the neural networks were implemented by the author, using relevant parts of the PyTorch framework. For model code, see Appendix B. The hierarchical linear models are coded in Stan [33], which is a highly flexible modelling language. Code for the linear models can be found in Appendix B. Several alternative options to PyTorch and Stan exist, but was not tested. Exactly how much the usage of alternative frameworks could potentially change inference remains unknown.

6.3 Method

The linear models presented in this thesis may not have the predictive power of other non-linear models. Linear models may be tweaked to display non-linear predictions, by incorporating one or several higher-order polynomials (or similar) while still fulfilling the definition of linear models; linearity in the parameters. From a theoretical point of view, there are several approaches that could be tried. For instance, as ALST has a theoretical established quadratic relationship with age [6], such a configuration would be feasible. Potential modelling choices are perhaps in abundance, it is not possible to include all. The variable selection and the linear models presented in this thesis may be justified from what exists in the literature, as variable selection to a high degree was founded upon previous research. The predictive metrics of linear models reported in this thesis places a direct comparison to other models available and as a reference for future studies in the field.

Inclusion of neural networks was to a degree chosen as to see if a more advanced and complex model structure would be able to improve predictions while generalization to unseen data is maintained. The multi-layer setting of three layers was used with the aim of finding the complex patterns that exist in the data via deep learning. Empirical testing proved that a three-layer neural network performed well on training data with sufficient generalization to unseen data. Empirical testing of different structures also proved that a single-layer neural network was enough to easily outperform linear regression, yet not with the same amount of reduced variability in predictions. Exploring deep learning is likely an approach that can reduce variability even further, to what extent remains unknown.

Although regarded as benchmark models, the linear models fulfil an interesting purpose; the hierarchical setup. The ICC metric was used in this thesis to check if a multi-level structure was necessary – and indeed it was. As all ICC distributions presented in this thesis have all posterior mass over 0.5, there is thus a 100 percent probability that hierarchical models are necessary (assuming 0.5 is used as a threshold), according to the models and the parametric form that they are founded upon. Although changing the parametric form may lead to contradicting results regarding ICC, the results are somewhat telling. The fitting of neural networks may then be questioned, as the NN's presented do not incorporate the hierarchical structure in the data. However, since the NN's have a completely different parametric form, it remains unknown if ICC would be of any importance in a hierarchical NN setting.

The neural networks fitted in this thesis are built on the Bayesian foundation, using variational inference. The diagonal Gaussian representation of the weight posterior distribution is a considerable simplification. There are likely dependencies that exist between the weights that this particular setup does not account for. However, the fitted Bayesian neural networks bring the benefit of displaying the full uncertainty in models' predictions, and simultaneously bring regularization on the weights via the complexity cost. The hierarchical linear models were primarily chosen with the domain specific expertise of the external commissioner in mind. Although the elicitated linear model was not selected, the prior elicitation is an interesting contribution to this thesis. Bayesian inference might be of extra relevance for dietitians, as the research is performed in an area or their expertise. Often, the number of subjects is few, in that sense priors are of importance, especially using proper elicitation.

As discussed in chapter 3, there is a dependency between features, that to various (and unknown) degrees affect the outcome of e.g. reactance. If a directed acyclic graph (DAG) was to be created, then there would be no theoretical justification to draw a directed arrow from e.g. $Reactance \rightarrow ALST$. In fact, it would be directed in the other direction; $ALST \rightarrow Reactance$, as reactance clearly does not cause muscle mass. With such a contradicting statement laid out, it seems counter intuitive to model ALST using electrical variables as features, as both

neural networks and hierarchical linear models are essentially DAG's that point in the wrong direction. A family of models that could be considered are Bayesian networks, with directed arrows that better rhymes with the theoretical causal reasoning of how BIA works. However, with the aim of this thesis – to improve BIA predictions – modelling BN's may be an inferior choice. Bayesian networks model the joint distribution, which is of little interest in terms of predictions.

With the aim of finding a functional mapping from features to target, with seemingly high dependencies between features, hierarchical linear models and simple feed-forward neural networks may not be the best option. A way to incorporate both the hierarchical setup and a powerful predictive machine are Gaussian processes (GP). The "heart" of GP's is to model the correlations – or covariance – between individuals with the kernel function. The kernel function computes similarities between individuals and may be tweaked by using more than one kernel to express structures in data. Furthermore, using GP's allows for a full Bayesian treatment, with priors, hyper-priors and complete modelling of the uncertainty. As such, the choice of GP's seems ideal, but there are some significant drawbacks. GP's need to calculate the inverse of a covariance matrix with dimensions $[n \times n]$ at each iteration of e.g. MCMC, which makes computations extremely expensive. Also, computing the predictive mean and predictive variance of a fitted model requires a grid of all possible combinations of features, which is unfeasible if too many features are included.

6.4 The work in a wider context

The results of the models fitted in this thesis suggest that an improvement has been made on predictions. While not exactly as accurate as those of the DXA device, the models can be used by the practitioners at the Clinical Nutrition Unit for future patients. The data filtering has however filtered out the most influential (outlier) measurements, meaning that only observations that fulfil specific quality measures are left. In order to safely use the results in the future, careful analysis of which observations were removed need be investigated, to have full control over which future patients are safe to predict. This is true because the most influential observations are not part of the inference, and thus not part of the evaluation presented in this thesis either.

6.5 Future research

Knowing the data is key to building a good model. It is theoretically clear why some variables influence other variables, but not to what extent. Performing an analysis that describes the data and the relationships would likely be a good choice in order to understand the mechanisms of BIA. One such model is Bayesian networks. This kind of model could be used to describe the conditional dependencies that exist between variables in a probabilistic way. The conclusions drawn from such an analysis could be used to de-mystify some aspects of BIA and allow for reasonable variable selections and/or model choices in the future. Another approach that would most likely increase the predictive results is to include at least one additional anthropometric variable, e.g. arm length or waist girth. Knowing for sure how a subject is shaped would likely explain some variability in electrical variables. However, using one such variable in a regression setting requires measurement of the same variable for all future patients, if the model is used for prediction. That might be a less attractive option. In conclusion, future research should be divided in two; (1) use a descriptive model to quantify conditional dependencies in BIA variables and (2) use a non-linear model based on knowledge gained from (1) as a predictive machine.

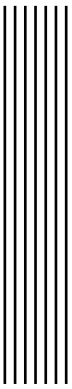
7 Conclusion

Can predictions on appendicular lean soft tissue from the impedance device be improved upon using a statistical model?

The results from both the hierarchical linear models and neural networks show that improvements have been made in terms of MSE. The MSE metric does however only indicate that improvements are made on average. In the different Y-vs-Y plots produced for all models, some concerns can be raised towards very muscular individuals, in these regions the models are prone to under-estimation of $ALST$, although these predictions are definitely closer to the ground truth than those of the BIA device. In conclusion, the predictions are improved using both modelling approaches.

Can predictions from the fitted models generalize over all ranges of BMI?

When plotting the difference between prediction and target as a function of BMI, the fitted models display no systematic errors associated with either increased or decresased BMI. Some concerns can however be raised over higher regions of BMI, which contain more outliers. In conclusion, both models generalize well over all ranges of BMI.



Bibliography

[1] Charlotte Beaudart, Olivier Bruyère, Anton Geerinck, Manon Hajaoui, Aldo Scafoglieri, Stany Perkisas, Ivan Bautmans, Evelien Gielen, Jean-Yves Reginster, and Fanny Buckinx. "Equation models developed with bioelectric impedance analysis tools to assess muscle mass: A systematic review". In: *Clinical Nutrition ESPEN* 35 (2020), pp. 47–62. DOI: 10.1016/j.clnesp.2019.09.012. URL: https://doi.org/10.1016/j.clnesp.2019.09.012.

[2] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. "Weight Uncertainty in Neural Networks". In: (2015). cite arxiv:1505.05424Comment: In Proceedings of the 32nd International Conference on Machine Learning (ICML 2015). URL: http://arxiv.org/abs/1505.05424.

[3] Marja Bosaeus, Therese Karlsson, Agneta Holmäng, and Lars Ellegård. "Accuracy of quantitative magnetic resonance and eight-electrode bioelectrical impedance analysis in normal weight and obese women". In: *Clinical Nutrition* 33.3 (2014), pp. 471–477. DOI: 10.1016/j.clnu.2013.06.017. URL: https://doi.org/10.1016/j.clnu.2013.06.017.

[4] A. Bosy-Westphal, B. Jensen, W. Braun, M. Pourhassan, D. Gallagher, and M J Müller. "Quantification of whole-body and segmental skeletal muscle mass using phase-sensitive 8-electrode medical bioelectrical impedance devices". In: *European Journal of Clinical Nutrition* 71.9 (2017), pp. 1061–1067. DOI: 10.1038/ejcn.2017.27. URL: https://doi.org/10.1038/ejcn.2017.27.

[5] A Bosy-Westphal, Britta Schautz, Wiebke Later, Joseph John Kehayias, Donal Gallagher, and Manfred James Müller. "What makes a BIA equation unique? Validity of eight-electrode multifrequency BIA to estimate body composition in a healthy adult population." In: *European Journal of Clinical Nutrition* 67 (2013), S14–S21.

[6] Alfonso Cruz-Jentoft, Gulistan Bahat, Juergen Bauer, Yves Boirie, Olivier Bruyère, Tommy Cederholm, Cyrus Cooper, Francesco Landi, Yves Rolland, Avan Aihie Sayer, Stéphane Schneider, Cornel Sieber, Eva Topinková, Maurits Vandewoude, Marjolein Visser, and Mauro Zamboni. "Sarcopenia: revised European consensus on definition and diagnosis". In: *Age and ageing* 48 (Oct. 2018). DOI: 10.1093/ageing/afy169.

[7] L. Ellegård, F. Bertz, A. Winkvist, I. Bosaeus, and H K Brekke. "Body composition in overweight and obese women postpartum: bioimpedance methods validated by dual energy X-ray absorptiometry and doubly labeled water". In: *European Journal of Clinical*

Nutrition 70.10 (2016), pp. 1181–1188. DOI: 10.1038/ejcn.2016.50. URL: https://doi.org/10.1038/ejcn.2016.50.

[8] K R Foster and H C Lukaski. "Whole-body impedance–what does it measure?" In: *The American Journal of Clinical Nutrition* 64.3 (Sept. 1996), 388S–396S. ISSN: 0002-9165. DOI: 10.1093/ajcn/64.3.388S. eprint: https://academic.oup.com/ajcn/article-pdf/64/3/388S/24035357/388s.pdf. URL: https://doi.org/10.1093/ajcn/64.3.388S.

[9] Andrew Gelman, John B. Carlin, Hal S. Stern, and Donald B. Rubin. *Bayesian Data Analysis*. 3rd ed. Chapman and Hall/CRC, 2014.

[10] Andrew Gelman, Ben Goodrich, Jonah Gabry, and Aki Vehtari. "R-squared for Bayesian Regression Models". In: *The American Statistician* 73.3 (2019), pp. 307–309. DOI: 10.1080/00031305.2018.1549100.

[11] Andrew Gelman and Donald B. Rubin. "Inference from Iterative Simulation Using Multiple Sequences". In: *Statistical Science* 7.4 (1992), pp. 457–472. ISSN: 08834237. URL: http://www.jstor.org/stable/2246093.

[12] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[13] Alex Graves. "Practical Variational Inference for Neural Networks". In: *Advances in Neural Information Processing Systems 24* (2011), pp. 2348–2356. URL: http://papers.nips.cc/paper/4329-practical-variational-inference-for-neural-networks.pdf.

[14] Geoffrey E. Hinton and Drew van Camp. "Keeping the Neural Networks Simple by Minimizing the Description Length of the Weights". In: New York, NY, USA: Association for Computing Machinery, 1993, pp. 5–13. ISBN: 0897916115. DOI: 10.1145/168304.168306. URL: https://doi.org/10.1145/168304.168306.

[15] Matthew Hoffman and Andrew Gelman. "The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo". In: *Journal of Machine Learning Research* 15 (Nov. 2011).

[16] Kuen-Chang Hsieh, Yu-Jen Chen, Hsueh-Kuan Lu, Ling-Chun Lee, Yong-Cheng Huang, and Yu-Yawn Chen. "The novel application of artificial neural network on bioelectrical impedance analysis to assess the body composition in elderly". In: *Nutrition Journal* 12.1 (Sept. 2013), p. 21. DOI: 10.1186/1475-2891-12-21. URL: https://doi.org/10.1186/1475-2891-12-21.

[17] Therese Karlsson, Amra Osmancevic, Nina Jansson, Lena Hulthén, Agneta Holmäng, and Ingrid Larsson. "Increased vitamin D-binding protein and decreased free 25(OH)D in obese women of reproductive age". In: *European Journal of Nutrition* 53.1 (2014), pp. 259–267. DOI: 10.1007/s00394-013-0524-8. URL: https://doi.org/10.1007/s00394-013-0524-8.

[18] Sami F. Khalil, Mas S. Mohktar, and Fatimah Ibrahim. "The theory and fundamentals of bioimpedance analysis in clinical status monitoring and diagnosis of diseases." In: *Journal of sensors* 14 (2014).

[19] Diederik P. Kingma and Max Welling. "Auto-Encoding Variational Bayes". In: *CoRR* abs/1312.6114 (2014).

[20] Diederik Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *International Conference on Learning Representations* (Dec. 2014).

[21] S. Kullback and R. A. Leibler. "On Information and Sufficiency". In: *The Annals of Mathematical Statistics* 22.1 (1951), pp. 79–86. ISSN: 00034851. URL: http://www.jstor.org/stable/2236703.

[22] Ursula G. Kyle, Ingvar Bosaeus, Antonio D. De Lorenzo, Paul Deurenberg, Marinos Elia, JoséManuel Gómez, Berit Lilienthal Heitmann, Luisa Kent-Smith, Jean-Claude Melchior, Matthias Pirlich, Hermann Scharfetter, Annemie M. W. J. Schols, and Claude Pichard. "Bioelectrical impedance analysis – part I: review of principles and methods". In: *Clinical Nutrition* 23 (2004), pp. 1226–1243. DOI: 10.1016/j.clnu.2004.06.004.

[23] Ursula G. Kyle, Ingvar Bosaeus, Antonio D. De Lorenzo, Paul Deurenberg, Marinos Elia, José Manuel Gómez, Berit Lilienthal Heitmann, Luisa Kent-Smith, Jean-Claude Melchior, Matthias Pirlich, Hermann Scharfetter, Annemie M. W. J Schols, and Claude Pichard. "Bioelectrical impedance analysis – part II: utilization in clinical practice". In: *Clinical Nutrition* 23 (2004), pp. 1430–1453. DOI: 10.1016/j.clnu.2004.09.012.

[24] Y. Lu, J. K. Hahn, and X. Zhang. "3D Shape-Based Body Composition Inference Model Using a Bayesian Network". In: *IEEE Journal of Biomedical and Health Informatics* 24.1 (2020), pp. 205–213.

[25] Richard McElreath. *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*. CRC Press, 2016. URL: http://xcelab.net/rm/statistical-rethinking/.

[26] K. Mehlig, E. Freyer, G. Tognon, V. Malmros, L. Lissner, and I. Bosaeus. "Body composition by dual-energy X-ray spectrometry and bioelectrical impedance spectroscopy in a healthy population at age 75 and 80". In: *Clinical Nutrition ESPEN* 10.1 (2015), e26–e32. DOI: 10.1016/j.clnme.2014.11.001. URL: https://doi.org/10.1016/j.clnme.2014.11.001.

[27] David Naranjo, Javier Reina-Tosina, and Mart Min. "Fundamentals, Recent Advances, and Future Challenges in Bioimpedance Devices for Healthcare Applications". In: *Journal of Sensors* (2019). DOI: 10.1155/2019/9210258.

[28] Radford M. Neal. "MCMC Using Hamiltonian Dynamics". In: *Handbook of Markov Chain Monte Carlo* 54 (2010), pp. 113–162.

[29] Kristina Norman, Nicole Stobäus, Matthias Pirlich, and Anja Bosy-Westphal. "Bioelectrical phase angle and impedance vector analysis - Clinical relevance and applicability of impedance parameters". In: *Clinical nutrition (Edinburgh, Scotland)* 31 (2012). DOI: 10.1016/j.clnu.2012.05.008.

[30] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. "Automatic differentiation in PyTorch". In: (2017).

[31] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[32] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. "Learning representations by back-propagating errors". In: *Nature* 323.6088 (1986), pp. 533–536. URL: https://doi.org/10.1038/323533a0.

[33] Stan Development Team. *Stan Modeling Language Users Guide and Reference Manual*. Version Version 2.18.0. 2018. URL: http://mc-stan.org.

[34] U.K. Tanita. *Multi-Frequency Body Composition Analyzer MC-180MA. Instruction manual.* English. 2005. URL: http://www.agenteksport.co.il/files/catalog/1372229239q39Th.pdf.

[35] Gianliuca Tognon, Vibeke Malmros, Elisa Freyer, Ingvar Bosaeus, and Kirsten Mehlig. "Are segmental MF-BIA scales able to reliably assess fat mass and lean soft tissue in an elderly Swedish population?" In: *Elsevier* (2015).

[36] Christian Tronstad and Runar Strand-Amundsen. "Possibilities in the application of machine learning on bioimpedance time-series". In: *Journal of Electrical Bioimpedance* 10 (2019), pp. 24–33.

[37] Aki Vehtari, Andrew Gelman, and Jonah Gabry. "Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC". In: *Statistics and Computing* 27.5 (2017), pp. 1413–1432. DOI: 10.1007/s11222-016-9696-4.

A Appendix A – Model diagnostic plots

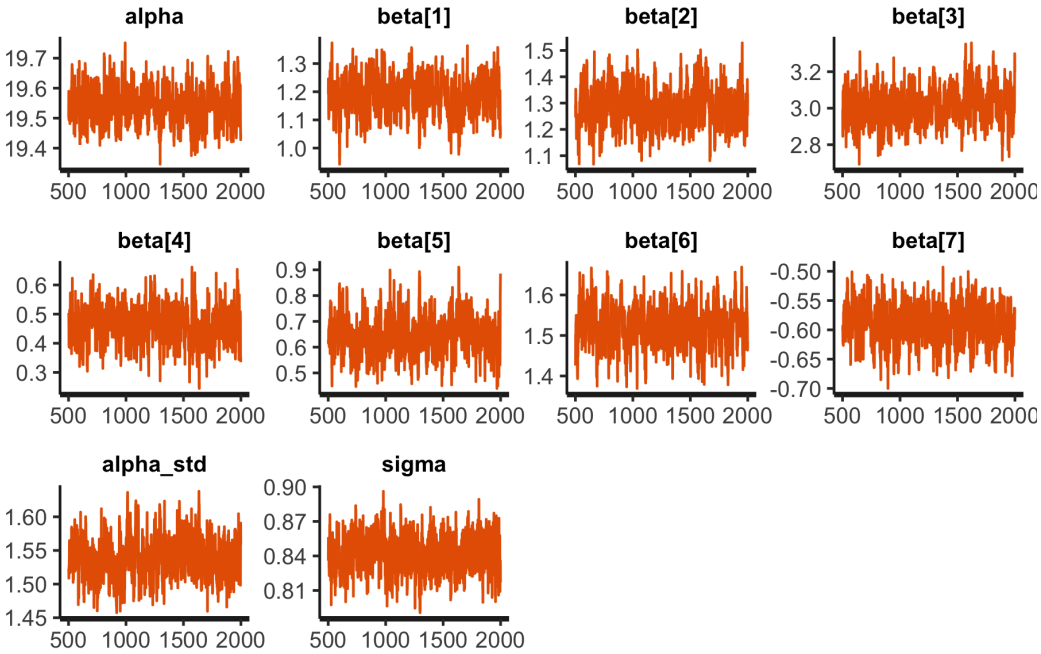


Figure A.1: Traceplots for relevant parameters of the regularizing hierarchical linear model.

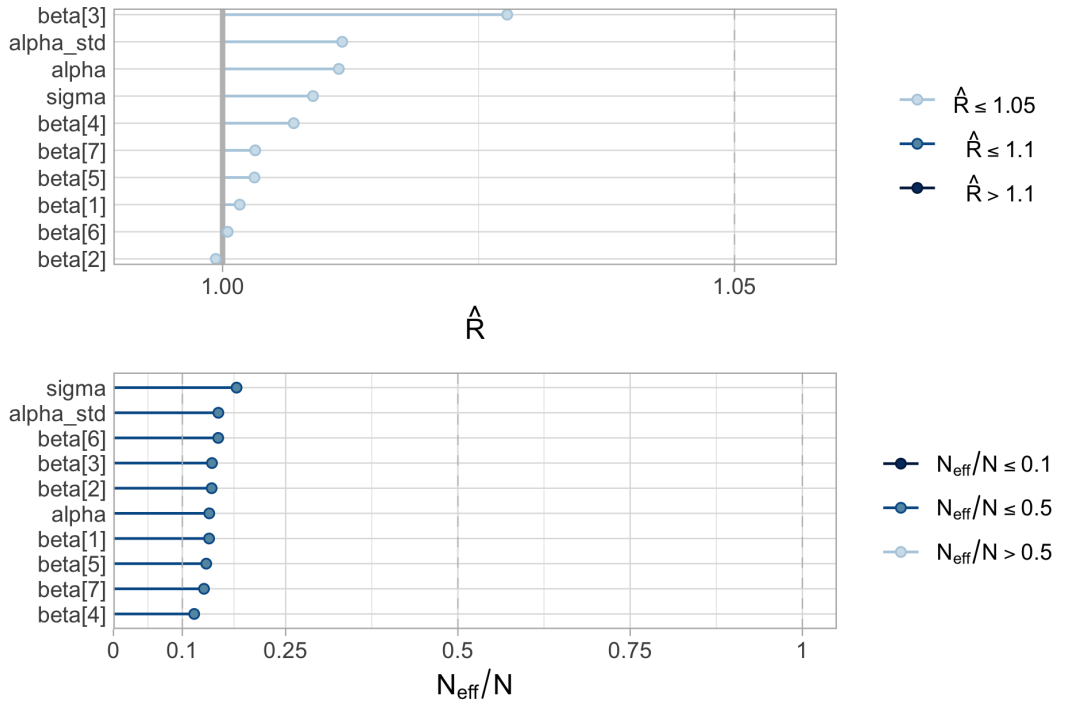


Figure A.2: Effective samples and Gelman-Rubin diagnostic for the regularizing hierarchical linear model.

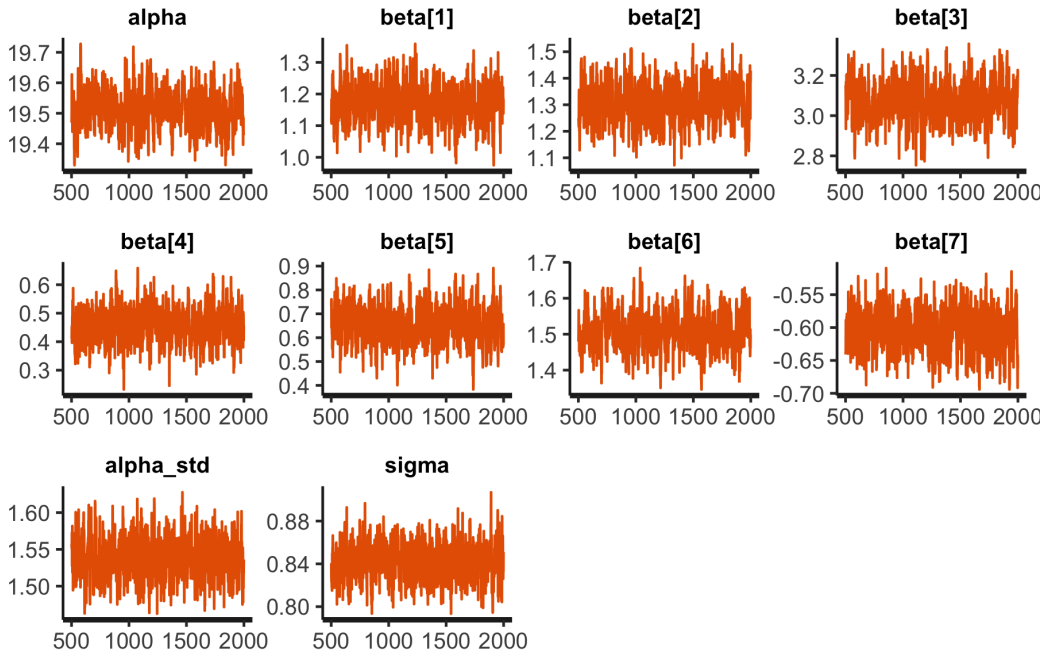


Figure A.3: Traceplots for relevant parameters of the weakly informative hierarchical linear model.

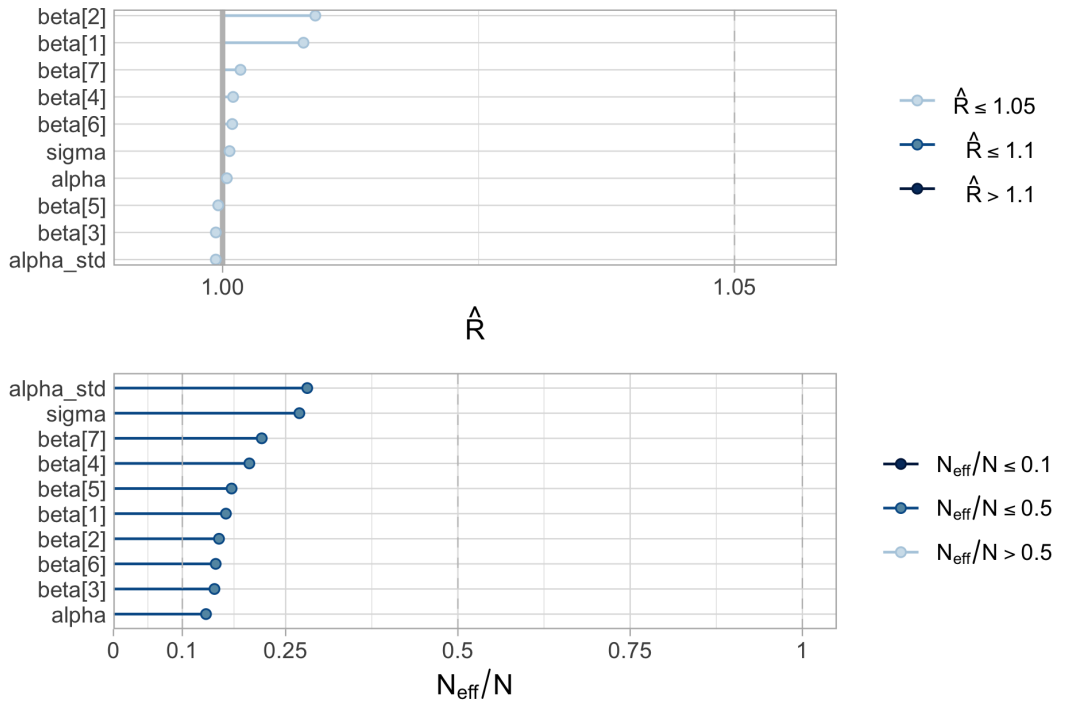


Figure A.4: Effective samples and Gelman-Rubin diagnostic for the weakly informative hierarchical linear model.

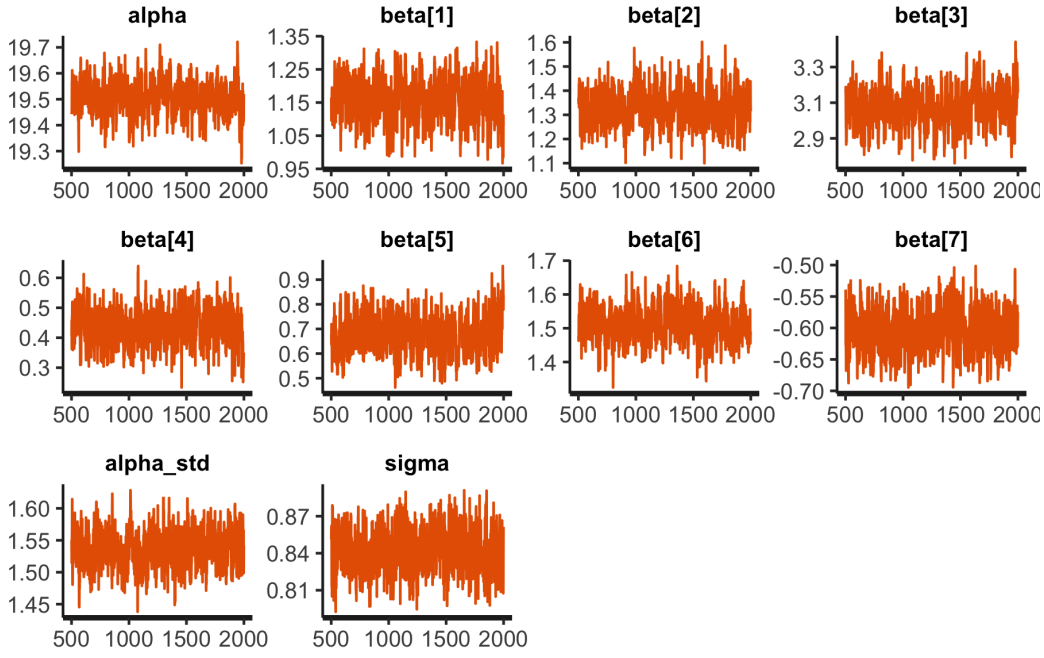


Figure A.5: Traceplots for relevant parameters of the elicitated hierarchical linear model.

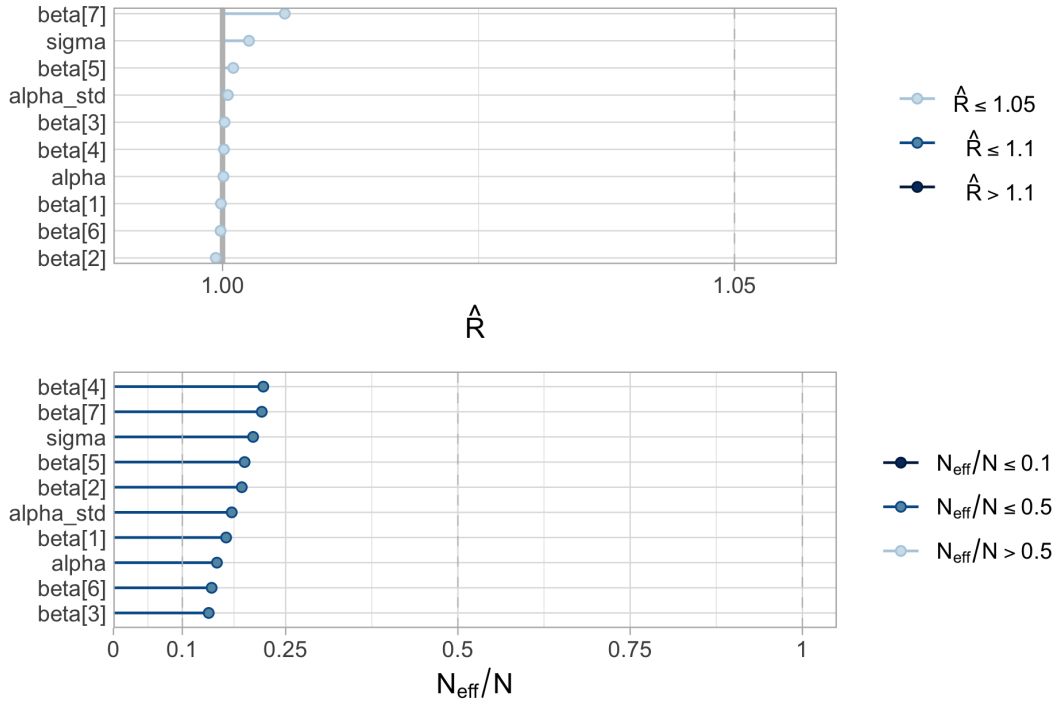


Figure A.6: Effective samples and Gelman-Rubin diagnostic for the elicitated hierarchical linear model.

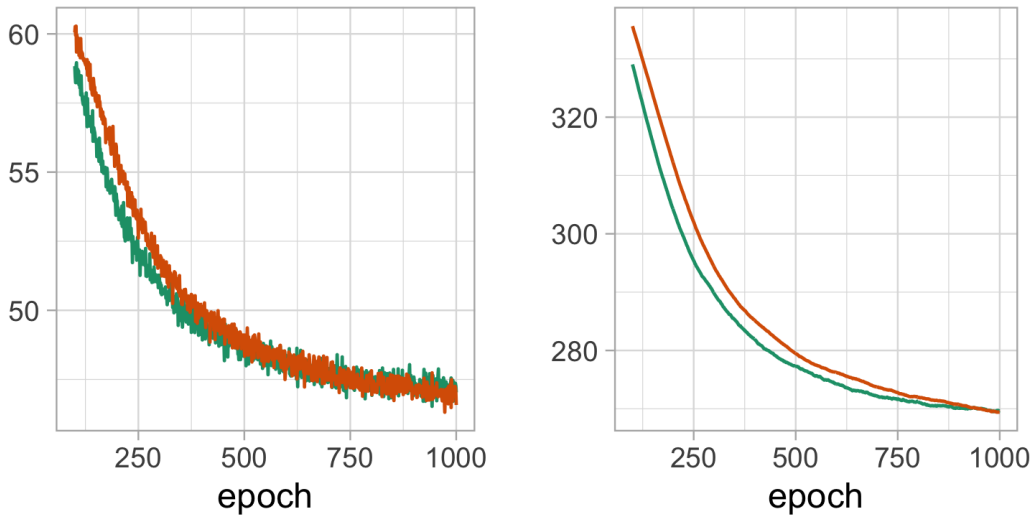


Figure A.7: Complexity cost over epochs for both neural networks, green: Laplacean NN, orange: Gaussian NN.

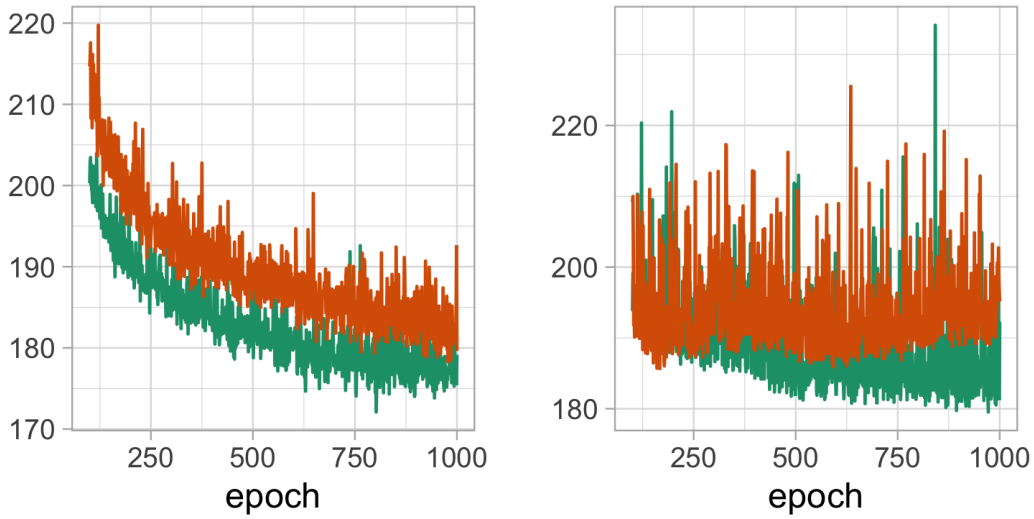


Figure A.8: Likelihood cost over epochs for both neural networks, green: Laplacean NN, orange: Gaussian NN.

B Appendix B – Code

Stan code for hierarchical linear model with Laplace regularization

```
data {
  int<lower=1> N;
  int<lower=1> J;
  int<lower=1> id[N];

  vector[N] Ri;
  vector[N] Pi;
  int<lower=0> Dm[N];
  vector[N] W;
  vector[N] H;
  vector[N] A;
  vector[N] y;
  real lambda;
}
parameters {
  real alpha;
  real<lower=0> alpha_std;
  vector[N] alpha_ncp;

  vector[7] beta;
  real<lower=0> sigma;
}
transformed parameters {
  real alpha_id[J];

  for (j in 1:J){
    alpha_id[j] = alpha + alpha_std * alpha_ncp[j];
  }
}
model {
```

```
  vector[N] mu;
  vector[N] gamma1;
  vector[N] gamma2;

  alpha     ~ normal(20, 10);
  alpha_std ~ normal(0,1);
  alpha_ncp ~ normal(0,1);
  beta      ~ double_exponential(0,lambda);
  sigma     ~ normal(0,2);

  for (n in 1:N){
    // Linear interaction part; D_male and R_index:
    gamma1[n] = beta[1] + beta[2] * Dm[n];
    // Link function:
    mu[n] = alpha_id[id[n]] +
            gamma1[n] * Ri[n] + beta[3] * Dm[n] +
            beta[4] * Pi[n] + beta[5] * W[n] +
            beta[6] * H[n] + beta[7] * A[n];
    // Log-likelihood:
    y[n] ~ normal(mu[n], sigma);
  }
}
generated quantities{
  real ICC;
  real R2;
  real MSE;
  vector[N] y_s;
  vector[N] log_lik;
  ICC = alpha_std / (alpha_std + sigma);

  {
    vector[N] mu_s;
    vector[N] r_s;
    vector[N] gamma1;

    for (n in 1:N){
      gamma1[n] = beta[1] + beta[2] * Dm[n];
      // Note: Using the population mean parameter alpha here.
      mu_s[n] = alpha +
                gamma1[n] * Ri[n] + beta[3] * Dm[n] +
                beta[4] * Pi[n] + beta[5] * W[n] +
                beta[6] * H[n] + beta[7] * A[n];
      y_s[n] = normal_rng(mu_s[n], sigma);
      log_lik[n] = normal_lpdf(y[n] | mu_s[n], sigma);
      r_s[n] = y[n] - y_s[n];
    }
    MSE = mean(square(r_s));
    R2  = variance(y_s) / (variance(y_s) + variance(r_s));
  }
}
```

Python code for diagonal variational posterior

```
class Gaussian(object):
    def __init__(self, mu, rho, eps_sigma):
        super().__init__()
        self.mu  = mu
        self.rho = rho
        self.eps = torch.distributions.Normal(0, eps_sigma)

    @property
    def sigma(self):
        return torch.log(1 + torch.exp(self.rho))

    def reparam(self):
        epsilon = self.eps.sample(self.rho.size())
        return self.mu + self.sigma * epsilon

    def log_prob(self, w):
        return torch.distributions.Normal(self.mu, self.sigma).log_prob(w).sum()

    def rng(self, sampleBias = False):
        s = torch.zeros(self.mu.shape)
       M = self.mu.detach().numpy()
        S = self.sigma.detach().numpy()

        if sampleBias == True:
            for i in range(M.shape[0]):
                s[i] = torch.distributions.Normal(M[i], S[i]).sample()
            return s

        if sampleBias == False:
            for i in range(s.shape[0]):
                for j in range(s.shape[1]):
                    s[i,j] = torch.distributions.Normal(M[i,j], S[i,j]).sample()
            return s
```

Python code for Laplace prior with fixed parameters

```
class Laplace(object):
    def __init__(self, sigma):
        super().__init__()
        self.laplace = torch.distributions.Laplace(0,sigma)

    def log_prob(self, w):
        return (self.laplace.log_prob(w)).sum()
```

Python code for Bayesian feed-forward linear layer

```
class linearBayesLaplace(nn.Module):
    def __init__(self, n_In, n_Out, eps_noise, prior_sigma):
        super().__init__()

        self.w_mu = nn.Parameter(torch.Tensor(n_Out, n_In).normal_(0.0, 0.1))
        self.w_rho = nn.Parameter(torch.Tensor(n_Out, n_In).uniform_(-6.0,-5.0))

        self.b_mu = nn.Parameter(torch.Tensor(n_Out).normal_(0.0, 0.1))
        self.b_rho = nn.Parameter(torch.Tensor(n_Out).uniform_(-6.0,-5.0))

        self.w = Gaussian(self.w_mu, self.w_rho, eps_noise)
        self.b = Gaussian(self.b_mu, self.b_rho, eps_noise)

        self.w_prior = Laplace(prior_sigma)
        self.b_prior = Laplace(prior_sigma)

        self.log_prior = 0.
        self.log_variational_posterior = 0.

    def forward(self, x, training = True):

        if training:
            w = self.w.reparam()
            b = self.b.reparam()
        else:
            w = self.w_mu
            b = self.b_mu

        self.log_prior = self.w_prior.log_prob(w) + self.b_prior.log_prob(b)
        self.log_variational_posterior = self.w.log_prob(w) + self.b.log_prob(b)

        return nn.functional.linear(x, w, b)

    def predict(self, x):
        w = self.w.rng(sampleBias = False)
        b = self.b.rng(sampleBias = True)
        return nn.functional.linear(x, w, b)
```

Python code for complete structure of Laplacean NN

```
class BayesByBackprop(nn.Module):
    def __init__(self, nIn, h1, h2, h3, nOut, eps_noise, sigma_prior):
        super().__init__()
        self.l1    = linearBayesLaplace(nIn, h1,   eps_noise, sigma_prior)
        self.l2    = linearBayesLaplace(h1,  h2,   eps_noise, sigma_prior)
        self.l3    = linearBayesLaplace(h2,  h3,   eps_noise, sigma_prior)
        self.l4    = linearBayesLaplace(h3,  nOut, eps_noise, sigma_prior)
        self.act   = nn.ReLU()

    def forward(self, x, training = True):
        x = self.l1(x, training)
        x = self.act(x)
        x = self.l2(x, training)
        x = self.act(x)
        x = self.l3(x, training)
        x = self.act(x)
        x = self.l4(x, training)
        return x

    def predict(self, x):
        x = self.l1.predict(x)
        x = self.act(x)
        x = self.l2.predict(x)
        x = self.act(x)
        x = self.l3.predict(x)
        x = self.act(x)
        x = self.l4.predict(x)
        return x
```

R code for partitioning of data into training, validation and test sets

```
partitionData = function(id, trainTarget, validTarget, partitionSeed){
  set.seed(partitionSeed)
  trainSet = c()
  trainIDs = c()

  idVec    = id
  idSet    = unique(id)

  idDraw   = sample(idSet, as.integer(validTarget*0.5))
  trainIDs = union(trainIDs, idDraw)
  idSet    = setdiff(idSet, idDraw)
  trainSet = idVec[idVec %in% trainIDs]

  while(length(trainSet) != trainTarget){
    idDraw   = sample(idSet, 1)
    trainIDs = union(trainIDs, idDraw)
    idSet    = setdiff(idSet, idDraw)
    trainSet = which(idVec %in% trainIDs)
    if(length(trainSet) > trainTarget){
      idDraw   = sample(trainIDs, 5)
      trainIDs = setdiff(trainIDs, idDraw)
      idSet    = union(idSet, idDraw)
      trainSet = which(idVec %in% trainIDs)
    }
  }

  validSet = c()
  validIDs = c()

  idDraw   = sample(idSet, as.integer(validTarget*0.5))
  validIDs = union(validIDs, idDraw)
  idSet    = setdiff(idSet, idDraw)
  validSet = idVec[idVec %in% validIDs]

  while(length(validSet) != validTarget){
    idDraw   = sample(idSet, 1)
    validIDs = union(validIDs, idDraw)
    idSet    = setdiff(idSet, idDraw)
    validSet = which(idVec %in% validIDs)
    if(length(validSet) > validTarget){
      idDraw   = sample(validIDs, 5)
      validIDs = setdiff(validIDs, idDraw)
      idSet    = union(idSet, idDraw)
      validSet = which(idVec %in% validIDs)
    }
  }

  testIDs = setdiff(idVec, union(trainIDs, validIDs))
  testSet = which(idVec %in% testIDs)

  partitionVector = vector(length = length(idVec))
```

```
  partitionVector[trainSet] = 1
  partitionVector[validSet] = 2
  partitionVector[testSet]  = 3
  return(partitionVector)
}

partitionVector = partitionData(id            = dtp$ID,
                                trainTarget   = 4000,
                                validTarget   = 800,
                                partitionSeed = 123456789)
```