Master's Thesis in Artificial Intelligence

Radboud University

Modeling Multi-Site Neuroimaging Data Using Hierarchical Bayesian Neural Networks to Study Structural Brain Variability

Author:
Hester Huijsdens (s4480562)

Supervisors:
Dr. A.F. Marquand
Dr. S.M. Kia
Dr. U. Güçlü

Radboud University, Nijmegen
Faculty of Social Sciences
October 2020

Abstract

Brain disorders are heterogeneous in their nature, therefore making it problematic to differentiate between multiple brain disorders using a traditional case-control approach. In contrary to case-control studies, where subjects are being classified into distinct groups, normative modeling can be used to model the variation in a large population. Normative modeling is a statistical method aiming to learn both the mean of a response variable and its normative range, therefore being able to handle the heterogeneous nature of highly variant groups. However, large amounts of neuroimaging data are required for normative modeling. Hence, datasets coming from different imaging centers need to be combined, introducing site-related variance generated by among others variability in acquisition devices across centers. We must account for this variability in the analysis, as these site-related variations are usually larger than those that we are interested in when trying to detect individual outliers from the normative range. Hierarchical Bayesian regression (HBR) models have been applied for multi-site normative modeling previously. However, it was limited in that they only had the options to fit either a linear, or a simple parametric, form. To allow for more flexibility and make less assumptions about the data, we will extend the HBR approach with a hierarchical Bayesian neural network. Using three simulated datasets, we show the model's ability to handle site-effects when the estimated effect is non-linear in both the mean and variance, while being computationally scalable. Moreover, the model can also deal with, and hence does not overfit on, sites with limited data. Although the estimates fit well to the data, the parameter sampling is unstable. In future studies, this is something that would need to be solved first, for example by assuming different priors or experimenting with different chain initialization methods. Finally, we fitted and evaluated the model on image-derived phenotypes from the UK Biobank dataset. In this test case, most brain measures showed a linear trend in their mean. Thus, we have observed limited improvements of our model compared to the previously described linear HBR model. However, when investigating the estimated variance, the HBNN showed a good performance in estimating a non-linear heteroscedastic variance in the data. Nevertheless, this non-linearity in the variance was minor, resulting in limited improvements of the neural network in comparison to the linear HBR model. Nevertheless, our model is able to model the mean and variance in datasets, also when there is non-linearity in the mean or variance of the data, or when the variance follows a skew Gaussian distribution. Compared to the linear HBR model, our model offers additional flexibility to handle site-related variation and non-linearity in neuroimaging datasets. Despite the limited improvements that we found, we expect that using a dataset containing more non-linearity in the response variable would allow us to differentiate between both models to a greater extent.

1 Introduction

Brain disorders are highly heterogeneous in terms of their symptoms and underlying biology [1, 2]. Hence, patients who have been diagnosed with the same brain disorder usually show diverse symptom behaviour, and the biology behind the same set of symptoms might vary across patients. Traditionally, to study brain disorders, we try to differentiate between them by classifying each disorder into a separate non-overlapping group. This is referred to as the case-control approach. The method has been successful in some cases, for example when differentiating between patients and control subjects. However, by using this approach, we assume that each brain disorder has a well-defined and distinct set of characteristics. Hence, if the intention is to differentiate between multiple brain disorders based on clinical data, the case-control approach usually results in a low performance due to the heterogeneous nature that characterises many brain disorders [3]. In short, the high variability within brain disorders makes the case-control approach unsuitable for studying these disorders.

Alternatively, brain disorders can be studied by means of normative modeling [4], a statistical method that is different from the traditional case-control approach in that it does not only model the mean of a response variable, but also its variance, therefore taking into account the heterogeneity and allowing inferences at the level of the individual. Instead of focusing on the group level, the learned variance is used to obtain a normative range. Subsequently, this range is used to identify individual participants who deviate from this range. This concept is analogous to a growth chart, where height is mapped as a function of (children's) age, and where deviations from the normative range are seen as deviations from normal growth. However, large amounts of data are required to reliably capture the variation in a response variable across the population. As neuroimaging datasets coming from a single imaging center are not likely to consist of enough participants, multiple datasets from different imaging centers usually need to be combined, bringing new challenges for the successive analysis of the data. Namely, technical differences such as varying procedures and scanner parameters across imaging sites introduce significant site-related variability in the data [5]. Moreover, different imaging centers might introduce demographic biases in the data in for example the gender ratio or age range. Both the technical and demographic differences across imaging centers are referred to as *site-effects* in this thesis. The site-effects are likely to confound the deviations that we are looking for in the normative modeling approach, because the site-related variability is usually larger than the individual deviations that we are interested in. Therefore, it is important to develop a normative modeling method that deals with site-effects, while preserving the other variances that are present in the data.

An approach to deal with site-effects is to regress them out of the data. A well-known method for this is ComBat [6], which was originally designed for harmonizing genomics datasets. The method has proven its ability to harmonize neuroimaging datasets, however, it has an important limitation when studying heterogeneous brain disorders. In order to remove site-effects using ComBat, we would need to specify the sources of variation explicitly. In the case of heterogeneous brain disorders, we do not know the underlying subtypes of a disorder, and therefore we do not know the sources of variation. Moreover, in many cases, site-effects strongly correlate with the covariates. As an illustration, this would be the case if there are large differences in both age ranges as well as scanner variation across imaging sites. Therefore, regressing out site-effects would then also regress out the signal of interest. A different approach is the use of hierarchical Bayesian regres-

sion (HBR) [7], which is a modeling technique that makes use of an overarching prior distribution across all sites and individually specialized parameter settings for each site-effect. Previously, we have implemented several HBR models and applied them on neuroimaging data using the normative modeling method [8]. However, depending on the model settings that are being specified in advance, the approach would try to fit either a linear or a a simple parametric form for the HBR model. Moreover, there was little flexibility in modeling the variance in the data - the variance could only be modeled linearly. Extending the approach to more flexible models would allow us to fit to other types of data, such as datasets that contain non-linearity in the mean, or a non-linear variance.

Here, the aim is to deal with the above described site-effects that are present in combined neuroimaging datasets, while also correctly estimating the mean and variance of the data. we extended the HBR approach from our previous work by a hierarchical Bayesian neural network (HBNN) for normative modeling. The HBNN is based on the previously implemented HBR model, but now it can fit non-linear functions in both the mean and variance. Moreover, the model has been extended with an option to fit either a Gaussian distribution or a skewed Gaussian distribution for the variance. The model is computationally scalable and can handle different numbers of subjects across sites.

First, we will describe the model, along with the normative modeling approach and HBR models in general, in more detail in Section 2.We will also discuss some of the problems that arise when fitting the HBNN. For example, sampling from the HBNN to learn the model's parameters is a complex problem. Moreover, the implementation of the HBNN for different group sizes is not straightforward. Next, the model will be fitted and evaluated using both a simulated dataset, to measure its performance on a case with clear site-effects, and the UK Biobank dataset (`https://www.ukbiobank.ac.uk`) [9], to determine its performance on neuroimaging data. Both datasets will also be described in Section 2. Next, the experimental setup and results will be presented in Section 3. In Section 4, I will end with a discussion where I will also mention limitations of the current model and some possible future directions.

2 Methods

2.1 Data

The model was fitted, evaluated and compared on different datasets: i) three simulated datasets, and ii) data coming from the UK Biobank (`https://www.ukbiobank.ac.uk`) [9]. To be able to evaluate the model's ability to model non-linearity in the mean and variance, we made sure that the simulated data contained non-linearity. For each group or site i, the mean of the simulated dataset is defined as:

$$\boldsymbol{y_i} = a_i \boldsymbol{x_i} + b_i \boldsymbol{x_i}^3 + c_i \tag{1}$$

The input $\boldsymbol{x_i}$, and variables a_i, b_i and c_i are sampled from a uniform distribution in the range of respectively -2 to 6, 0 to 20, 0 to 5, and 10 to 100. Moreover, the variance was normally distributed, centered around the mean, and simulated to be heteroscedastic, meaning that the variance increased as the independent variable increased. This variance is generated by adding it to the mean $\boldsymbol{y_i}$:

$$\boldsymbol{y_i} = \boldsymbol{y_i} + \frac{\boldsymbol{d_i}}{e_i} \log(1 + e^{0.3\boldsymbol{x_i}}) \tag{2}$$

Here, $\boldsymbol{d_i}$ is a vector of the length of number of samples group i. Its values are sampled from a standard Gaussian distribution. e_i is sampled from a uniform distribution in the range of 5 to 10. Three simulated datasets were generated. Further details and the implementation used for generating the data can be found online at `https://github.com/amarquand/PCNtoolkit`, but as an illustration, the three datasets can be seen in Figure 3. In the first two scenarios, there were 1000 data points divided equally over two groups. In the third scenario, one group contained 500 samples, while the other group contained only 50 samples. There was one feature that served as the response variable.

From the UK Biobank dataset, we selected all participants with available demographic information and image-derived phenotypes (IDPs), which are measures of brain structure and function derived from raw imaging data [9, 10]. These IDPs were extracted using the FUNPACK package (`https://git.fmrib.ox.ac.uk/fsl/funpack`). Examples of IDPs include the volume of a specific brain structure and the strength of connectivity between two brain structures. This resulted in a dataset containing demographic information and 891 different IDPs from 16916 participants. The dataset came from two imaging centers and participants' age ranged from 45 to 80 years for both sites. The first site consisted of 14186 participants (7521 males), whereas the other site included 2730 participants (1494 males).

2.2 Normative modeling

The first step of the normative modeling pipeline is to estimate a model that maps clinical or demographic data ($\boldsymbol{x}$) to biological response data ($\boldsymbol{y}$, in the current project these are the IDPs). This mapping function estimates both the mean and the variance, which indicates how well each data point fits to the normative model (as described in Section 1). In other words, we want to find two functions: $f_\mu(\boldsymbol{x}, \theta_\mu)$ for estimating the mean, and $f_\sigma^+(\boldsymbol{x}, \theta_\sigma)$ for estimating the standard deviation. θ_μ and θ_σ are the parameters for f_μ and f_σ respectively. The variance is modeled by a non-negative function, as it represents the normative range. Next, the predicted mean and variance

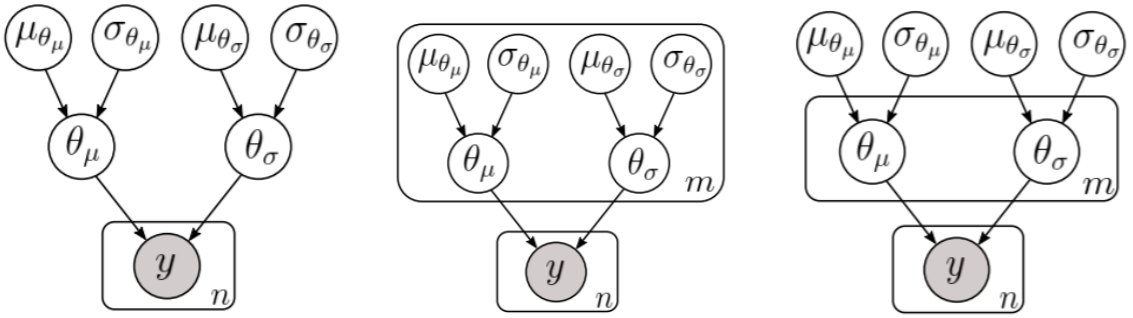
are being used for identifying individual deviations from this normative range. The deviations are quantified as Z-scores:

$$\boldsymbol{z} = \frac{\boldsymbol{y} - f_{\mu}(\boldsymbol{X}, \theta_{\mu})}{f_{\sigma}^{+}(\boldsymbol{X}, \theta_{\sigma})} \tag{3}$$

The Z-scores are computed for each IDP and subject individually. Next, the Z-scores that extremely deviate from the distribution are being identified as outliers, because outliers are expected to extremely deviate from the normative range. Moreover, the Z-scores can be combined into a summary measure of abnormality for each subject individually. To do this, we would first fit an extreme value distribution. Usually, the largest 1% of the Z-scores is selected for fitting this distribution. Next, we compute the corresponding cumulative distribution function and use it to estimate a probabilistic index of abnormality for each subject [4].

2.3 Hierarchical Bayesian regression

The reason for using a HBR model to estimate the mean and variance in a neuroimaging dataset, while also modeling site-effects, can best be explained by comparing them with two extremes: complete pooling and un-pooling models, both visualized in Figure 1. In complete pooling models (Figure 1a), the site-effect is overlooked. Hence, only one model will be fitted for each biological response variable. In other words, when using a pooling model, we are not modeling site-effects that are being present in the data. As explained in Section 1, this would be problematic due to the site-effects being larger than the deviations that we are interested in. Using an un-pooling model (Figure 1b) on the other hand would mean that we fit a separate model for each response variable and site. There is no common factor overarching the site-specific models, which would be problematic when there are sites with, for example, a very limited number of samples. Fitting the model for a site with a small number of subjects available would not be very successful, as the site-specific model is expected to overfit on the limited data.



a) Complete pooling b) Un-pooling c) HBR

Figure 1: A visualization (from Kia et al., 2020 [8]) of the structure of a) complete pooling models, b) un-pooling models, and c) partial-pooling models with hierarchical Bayesian regression (HBR). The number of sites and subjects are denoted by m and n, respectively.

HBR models (Figure 1c) provide an elegant trade-off between complete pooling and un-pooling models. Just like when fitting un-pooling models, a separate model is estimated for each site i:

$$\boldsymbol{y}_i = f_{\mu i}(\boldsymbol{x}, \theta_{\mu i}) + \epsilon_i \tag{4}$$

where ϵ_i is a zero-mean error for site i with standard deviation $f_\sigma^+(\boldsymbol{x}, \theta_\sigma)$. However, unlike the un-pooling models, the parameters $\theta_{\mu i}$ and $\theta_{\sigma i}$ are being sampled from an overarching prior distribution that is estimated from all data being pooled together. We assume an overarching prior distribution for all sites, and sample from this shared distribution for each site individually. In other words, $\forall_i, \theta_{\mu i} \sim \mathcal{N}(\mu_{\theta_\mu}, \sigma^2_{\theta_\mu})$. The variance is modeled to be either normally distributed, or follow a skew Gaussian distribution, which needs to be specified in advance. In the latter case, an additional parameter α_i is learned for each site individually, indicating the skewness of the distribution. Hence, this skewness parameter does not have an overarching prior distribution. In other words, $\theta_{\sigma i} \sim \mathcal{N}(\mu_{\theta_\sigma}, \sigma^2_{\theta_\sigma})$ or $\theta_{\sigma i} \sim \text{SkewNormal}(\mu_{\theta_\sigma}, \sigma^2_{\theta_\sigma}, \alpha_i)$. Due to the hierarchical structure, HBR models allow us build the hierarchical structure that is present in combined neuroimaging datasets, namely that they share an overall similar structure, but are likely to have individual differences in the mean and variance across sites. Moreover, when there is limited data available for a site, the corresponding site-specific model can learn from the data from other sites via the overarching prior distributions. This is different from the un-pooling model, where the individual models would not be able to learn from other sites' data.

2.3.1 Linear hierarchical Bayesian regression implementation

The linear HBR model has been described and implemented previously [8] and was used in this project for comparison purposes. The model allowed for either a fixed or varying slope and intercept across sites. Due to the characteristics of the datasets as described in Section 2.1, we fitted a linear model with a varying intercept, slope and model error across sites. Moreover, the variance was modeled as a heteroscedastic linear function. Hence, both the predicted mean and variance were dependent on $\boldsymbol{x}$. For estimating $f_\mu(\boldsymbol{x}, \theta_\mu)$ from Equation 3, a uniform distribution in the range of -100 to 100 was used for the prior intercept. A Gaussian and half-Cauchy distribution were used for modeling the prior mean and standard deviation of the slope, respectively. These model settings were the same for estimating $f_\sigma^+(\boldsymbol{x}, \theta_\sigma)$, however, the uniform distribution for the prior intercept was only allowed to be positive - a range from 0 to 100 was used. Moreover, the variance was modeled to follow either a Gaussian or skew Gaussian distribution. When estimating the skewness parameter α, we sampled from a uniform distribution in the range of -10 to 10.

2.3.2 Hierarchical Bayesian neural network implementation

The hierarchical architecture of the HBNN is visualized in Figure 2. It is important to realise that this architecture is being used twice. Namely, once for estimating $f_\mu(\boldsymbol{x}, \theta_\mu)$ and once for estimating the prediction variance $f_\sigma^+(\boldsymbol{x}, \theta_\sigma)$. Following the main idea of hierarchical Bayesian models, we learned separate sets of parameters for each site to estimate the mean for a given IDP. This means that there was one neural network for each site. Moreover, two additional neural networks were fitted to learn the prior distribution's mean and variance. The weights for the individual neural networks were being sampled from their corresponding prior weights. In other words, for a weight going from node a to node b, coming from the network of, for example, the first site, we had $w_{ab1} \sim \mathcal{N}(f_{\mu ab}, f^+_{\sigma ab})$. Moreover, for estimating $f_\mu(\boldsymbol{x}, \theta_\mu)$, a separate intercept was being modeled.

The prior intercept was modeled as a uniform distribution in the range of -100 to 100. Note that the intercept structure is not shown in Figure 2, as, unlike the network for $f_\mu(\boldsymbol{x}, \theta_\mu)$, the function $f_\sigma^+(\boldsymbol{x}, \theta_\sigma)$ did not fit an intercept.

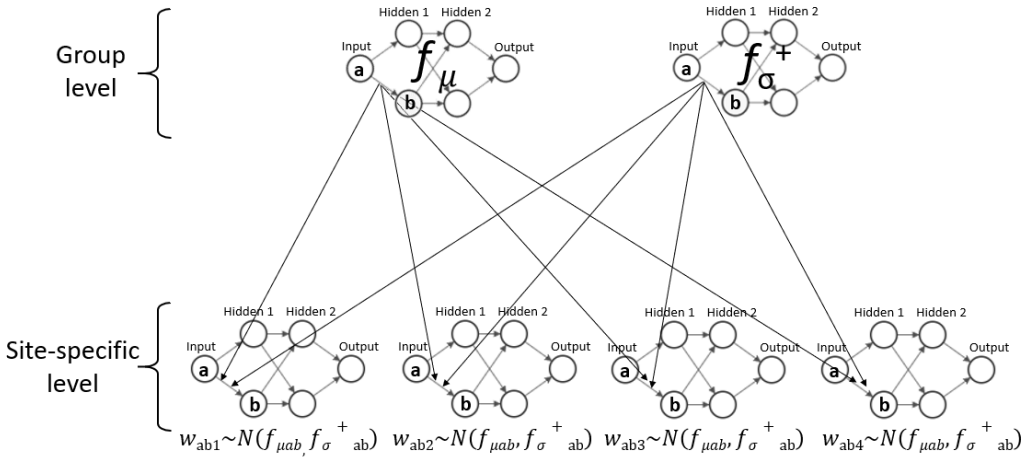


Figure 2: The hierarchical Bayesian neural network architecture. For each weight connecting node a to node b, two overarching functions f_μ and f_σ^+ are learned from pooling all the data. On the site-specific level, the weights are sampled from this overarching distribution (for the first site: $w_{ab1} \sim \mathcal{N}(f_{\mu ab}, f_{\sigma ab}^+)$). This architecture is used twice: once for estimating the mean, and once for estimating the noise term.

We used a simple two-layer architecture with two neurons in each layer. This architecture was chosen because we expect to model simple functions. However, the optimal network might depend on the dataset that is being used. When estimating $f_\mu(\boldsymbol{x}, \theta_\mu)$, hyperbolic activation functions were applied as transfer functions in the two hidden layers. As a consequence, weights for the neural network modeling the prior mean distribution were initialized using Xavier initialization [11], as this helps to avoid slow convergence and makes sure the gradients do not vanish or explode too quickly. Using Xavier initialization, the weights are initialized as:

$$\boldsymbol{w}_{\text{layer}} = \boldsymbol{R}\sqrt{\frac{1}{\text{size(layer-1)}}} \tag{5}$$

With $\boldsymbol{R}$ being a matrix of size(layer) x size(layer-1) samples drawn from a standard Gaussian distribution. When estimating the variance $f_\sigma^+(\boldsymbol{x}, \theta_\sigma)$, sigmoid activations were used instead. However, the weight initialization process was the same. The remainder of the weights for the prior neural networks were drawn randomly from a standard Gaussian distribution. Gaussian and half-Cauchy distributions are being used as priors for the mean and standard deviations of $f_\mu(\boldsymbol{x}, \theta_\mu)$ and $f_\sigma^+(\boldsymbol{x}, \theta_\sigma)$. Moreover, the prediction variance is dependent on $\boldsymbol{x}$ and therefore it was modeled as a heteroscedastic function, assuming either a Gaussian distribution or skew Gaussian distribution. When estimating the skewness parameter α, we sampled from a uniform distribution in the range of -10 to 10.

2.4 Experiments

2.4.1 Inference of the posterior distributions and implementation details

Both models were implemented using the PyMC3 package [12]. The inference of the posterior distributions of each parameter was performed using a No-U-Turn sampler (NUTS) [13], which is an extension of Hamilton Monte Carlo (HMC), a Markov chain Monte Carlo (MCMC) method. The HMC algorithm does not take random steps after initialization, but instead uses the derivative of the posterior to sample the posterior more efficiently. It adds an extra dimension called momentum to the sampler. This is expected to improve the convergence of the sampler. However, when using the HMC sampler, you would need to specify the step size in advance. If the step size is set too low, the sampler would show random walk behaviour. If the step size is set too large, the sampler might not find the optimal parameter values. NUTS overcomes this limitation by determining the optimal step size during the tuning phase. Moreover, each sampling chain is initialized to an identity mass matrix and then adapted to a diagonal based on the variance of the tuning samples. Hence, first the sampler starts with a tuning phase, and subsequently each chain is initialized to the variance of the posterior, as learned during the tuning phase. Next, NUTS begins the actual sampling of the posteriors. Additionally, we add or subtract a random number to the initialization, drawn from a uniform distribution in the range of -1 to 1. This initialization method was chosen because, as the posterior is being used for initialization, it is expected to help the sampling process by focusing on the space of accepted samples for future samples.

Nevertheless, the learning of optimal parameters for a hierarchical Bayesian neural network is a complex problem for multiple reasons. First of all, there are many parameters that need to be sampled, making it a complex task for the sampler to find the optimal combination of parameters. Additionally, there are many dependencies between the different parameters. Changing the value of one parameter, might influence the values of many other parameters in the model. Moreover, the performance of NUTS is influenced by the priors that were assumed, and by the initialization method used for the sampler.

The number of subjects coming from each site will vary, which is why the models need to be able to handle different group sizes. Fitting the HBNN for different group sizes was not straightforward, as in this case the input would not be in the form of a matrix. Therefore, the HBNN loops over all the sites and estimates each model within this loop. For both the simulated data and the UK Biobank dataset, we first used 500 iterations to tune each sampling chain. Next, 4000 samples were drawn for inferring the posterior distributions, using four independent sampling chains. Hence, we sampled 1000 times per chain. All model implementations are available online at `https://github.com/amarquand/PCNtoolkit`.

2.4.2 Data pre-processing and experimental setup

Two experiments were performed in order to measure the HBNN's ability to model the normative range of large datasets, while being able to handle site-effects. In the first experiment, three simulated datasets were used for comparing the HBNN to the linear HBR model. These datasets contained clear site-effects and non-linearity in the mean and variance. The simulated data was partitioned in a training (50 %) and test (50%) sets. The data was standardized so that the values of each feature have zero-mean and unit-variance. Next, both models were compared on neuroimag-

ing data from the UK Biobank (Section 2.1). From this dataset, we randomly selected 50% of the participants to train the model, whereas the remainder 50% was used for model evaluation. Age was treated as a covariate, whereas gender and scanner information were used as batch-effects. The IDPs served as the response variables and were first scaled to a range of zero to one, using a robust min-max feature standardization procedure. The minimum value used for this standardization method was defined as the average of the lowest 1% of the values of an IDP. The maximum value was defined in the same way, but using the highest 1% of the IDP values.

Moreover, we experimented with two different ways of modeling the variance in the UK Biobank dataset. As was explained in Section 2.3, both the HBNN and linear HBR model have the option to assume either a normally distributed variance, or a skewed normally distributed variance. We first modeled all the IDPs assuming a normally distributed variance, and subsequently selected multiple IDPs for modeling a skewed variance.

2.4.3 Performance evaluation

In both experiments, the estimated functions from age to IDPs were compared using three metrics: i) the Pearson correlation coefficient (RHO); ii) the standardized mean squared error (SMSE), defined as the mean squared error between the true and predicted IDP divided by their variance; and iii) the mean standardized log loss (MSLL) [14], defined as:

$$\text{MSLL} = \frac{1}{n}\sum_{j=1}^{n}\left(\frac{1}{2}\log(2\pi\sigma_*^2) + \frac{(y_j-\hat{y}_j)^2}{2\sigma_*^2} - \frac{1}{2}\log(2\pi\sigma_t^2) - \frac{(y_j-y_t)^2}{2\sigma_t^2}\right) \tag{6}$$

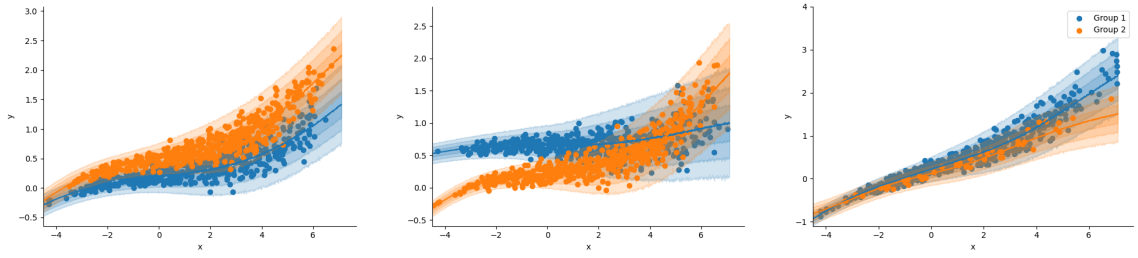
The MSLL was acquired for each IDP individually. We first computed the log loss with the true response (y_j), predicted response ($\hat{y}_j$), and σ_*^2 - the sum of the prediction variance (σ_j^2) and noise variance (σ_{noise}^2). Next, the log loss was standardized by subtracting the training log loss, computed with the mean (y_t) and variance (σ_t^2) of the training data t. This standardized log loss was averaged over the number of subjects n. The Pearson correlation coefficient and SMSE were used for evaluating the predicted mean of each normative model. However, the MSLL also takes into account the predicted variance of a normative model. This becomes especially important when estimating deviations from the normative range, as computed in Equation 3. We are only interested in positive values for the Pearson correlation coefficient and the reference values for the SMSE and MSLL are 1 and 0, respectively. Lower values of SMSE and MSLL indicate a better fit of the model to the data.

3 Results

3.1 Modeling non-linearity and site-effects

3.1.1 Fitting performance on the simulated datasets

The performance of the HBNN in estimating a non-linear mean and variance function, and modeling site-effects, was evaluated using simulated datasets, as described in Section 2.1. The individual groups all shared an overall similar non-linear structure, with site-specific variability in the mean and variance. Three scenarios were trialed. The first two scenarios had the same number of samples within each group, whereas the number of samples in the groups of the third scenario were different for the two groups.



a) Scenario 1 b) Scenario 2 c) Scenario 3

Figure 3: Model fits and their normative ranges for the hierarchical Bayesian neural network on three simulated datasets. The data within each scenario contained an overarching non-linear structure, but had individual differences between the groups.

Figure 3 shows the simulated data and resulting model fit for each scenario. For all three scenarios and both groups of the data, the normative models showed a good fit. This was confirmed by the corresponding performance metrics, presented in Table 1. Note that the reference values for SMSE and MSLL are 1 and 0 respectively, and that a lower value indicates a better fit. Moreover, we are only interested in positive Pearson correlation coefficients. In the first scenario (Figure 3a), the site-effects were relatively small. The SMSE value for this model was 0.1210, indicating a good fit to the dataset. The MSLL of -1.2415 and Pearson correlation coefficient of 0.9377 both confirmed this finding. Moreover, we computed the SMSE, MSLL and Pearson correlation for each group individually. The group-specific metrics for this scenario, also presented in Table 1, suggested a good fit for both groups. The second scenario (Figure 3b), contained larger site-effects. There was a large difference between the data coming from each group. The performance metrics for the previous scenario were slightly better, however the metrics for the second scenario still indicated a good model fit (SMSE = 0.2410, MSLL = -0.8998 and RHO = 0.8712). This was also the case for the group-specific SMSE, MSLL and Pearson correlation coefficient.

Table 1: The standardized mean-squared error (SMSE), mean standardized log loss (MSLL) and Pearson correlation coefficient (RHO) for three distinct simulated data scenarios.

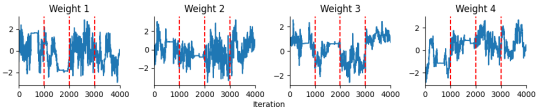
Scenario	Overall			Group 1			Group 2		
	SMSE	MSLL	RHO	SMSE	MSLL	RHO	SMSE	MSLL	RHO
1	0.1210	-1.2415	0.9377	0.2230	-0.9537	0.8827	0.1321	-1.1781	0.9319
2	0.2410	-0.8998	0.8712	0.8007	-0.3183	0.4579	0.1691	-1.0569	0.9144
3	0.0604	-1.6235	0.9703	0.0614	-1.6090	0.9696	0.0515	-1.7232	0.9807

The simulated dataset for the third scenario (Figure 3c) was somewhat different from the first two scenarios, because the second group contained a limited number of samples (10% of the number of samples in the first group). Again, the normative model showed a good performance: of all three scenarios, this resulting normative model best fitted to the simulated data based on the performance metrics (SMSE = 0.0604, MSLL = -1.6235 and RHO = 0.9703). Additionally, using this scenario, we aimed to show the advantage of using a hierarchical model over an un-pooling model, as it might be possible to achieve similar results on the first two scenarios with the use of un-pooling models. However, this is not expected for the third scenario, as it is expected that a model without hierarchical structure would overfit on the second group. When looking at the performance metrics, the model fit for the second group was still good (SMSE = 0.0515, MSLL = -1.7232 and RHO = 0.9807), indicating that the second group learned by using the networks for the prior distributions.

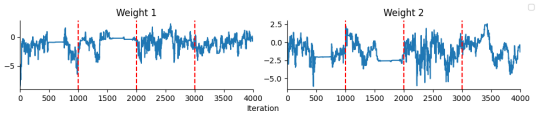
3.1.2 Inferring the posterior parameter distributions

To assess the stability of the model fits, we looked at the sampling traces for different parameter values. The second scenario was selected to further explore the behaviour of NUTS for sampling different parameter values, because as mentioned in Section 2, sampling the parameters for a hierarchical Bayesian neural network can be difficult. This was confirmed by the trace plots in Figure 4, showing the behaviour of NUTS for four independent chains and 1000 samples in each chain. Note that this figure shows the trace of a few parameters only, however, the behaviour observed in the trace plots was similar for the other sampled parameters. Ideally, the sampled values should first cover the whole parameter space, and subsequently converge as the number of iterations increases. When sampling the parameters for the log-transformed group-level standard deviations from the input to the first hidden layer (Figure 4b), we saw this behaviour in the last two chains. Moreover, a similar pattern was found in the trace plots corresponding to the site-specific mean between the same two layers (Figure 4c).

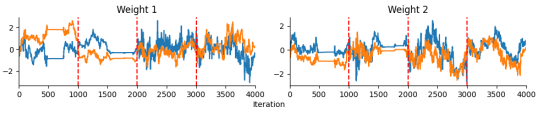
Although the sampled traces looked like how we wanted it to look for some chains and some parameters, we were able to identify three problems from the trace plots. First of all, the sampled parameter values varied across sampling chains. This behaviour could best be observed in Figure 4a, which shows the mean parameter on the overarching group level from the first to the second hidden layer. It could be that there are multiple optimal parameter combinations that result in the same output. NUTS can find a different set of optimal parameters depending on the initialization of each chain. The second problem can be observed by the flat horizontal lines in all four trace plots. Namely, the sampler got stuck. Hence, NUTS rejected many samples, as explained in Section 2.4.1. Finally, the sampler was inefficient for some parameters. This can best be observed by looking at the sampling behaviour for the site-specific intercepts (Figure 4d). Here, the parameter values might eventually converge, but it will take many iterations.



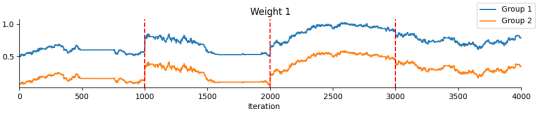
a) Group level mean from the first to the second hidden layer. The figure shows a plot for each weight going from the first to the second hidden layer.



b) Log-transformed values of the group level standard deviation from the input to the first hidden layer. There are two weights between the layers.



c) Site-specific mean parameters from the input to the first hidden layer. There are two weights, and a separate parameter is sampled for each group (site).



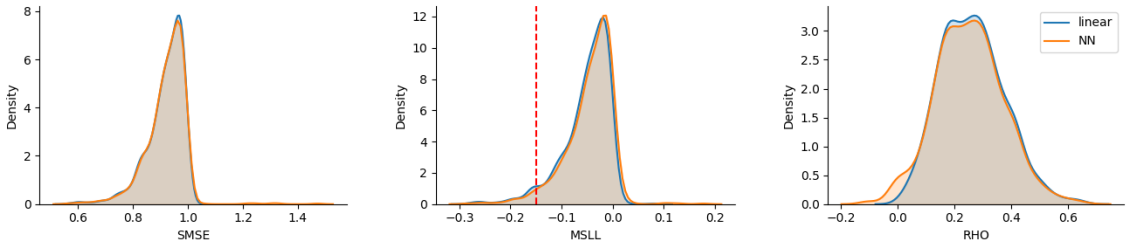
d) Site-specific intercept parameters for each group (site).

Figure 4: Trace plots for estimating the parameters of the second scenario. The No-U-Turn Sampler with four independent sampling chains (indicated by the red lines) and 1000 samples per chain were used for obtaining the posterior distributions.

3.2 Performance on a neuroimaging dataset

3.2.1 Assuming a Gaussian distribution on the likelihood

To evaluate the HBNN's performance on a neuroimaging dataset, the HBNN and linear HBR model were fitted on the IDPs coming from the UK Biobank dataset. Figure 5 shows a comparison of the densities of the SMSE, MSLL and Pearson correlation coefficient for the two models. The densities of all three metrics largely overlap, indicating that the HBNN performed as good as the linear HBR model.



a) SMSE b) MSLL c) RHO

Figure 5: A comparison between performances of the linear HBR model and hierarchical Bayesian neural network for all image-derived phenotypes, measured by a) standardized mean-squared error (SMSE), b) mean standardized log loss (MSLL), and c) Pearson correlation coefficient (RHO) between predicted and true values for the image-derived phenotypes. Phenotypes having a log loss value below 0.15 for the hierarchical Bayesian neural network were selected for further analyses.

The advantages of the HBNN over the linear HBR model did seem negligible, however this was as expected, since the effect of age will be mainly linear for most IDPs. The linear HBR model was already able to fit well to the data of these IDPs, therefore making it hard for the HBNN to improve this performance. In total, 155 IDPs showing a better data fit for the HBNN based on all three metrics were found. However, when looking at the model fits, no clear differences in performance could be seen, as both models fitted well to the data.

Moreover, multiple IDPs with a non-linear heteroscedastic variance were found by looking at the data. It is expected that the HBNN would be better able to fit to these IDPs than the linear HBR model, as the latter method can only fit a linear heteroscedastic noise term, whereas the HBNN can estimate a non-linear variance. To study the variance estimates for both models in depth, several IDPs were selected based on their MSLL values, because, as can be seen from Equation 6, this metric takes into account the predicted variance. The selection could be performed on two different criteria: 1) the IDPs where the difference in MSLL values for the HBNN and linear HBR model is largest, or 2) the IDPs that show a good model fit for the HBNN. A comparison between both options is visualized in Figure 6, showing that a large difference in MSLL between the two models does not necessarily mean that the HBNN fits well to the data. Here, we decided to select based on the MSLL values themselves instead of on the largest differences between the two models, as we wanted the model to fit well to the data. Thirty IDPs were selected to study the variance in the data in more detail. Those IDPs were selected based on MSLL values below -0.15 ($\text{MSLL}_{\text{HBNN}} < -0.15$), as indicated by the red line in Figure 5b.

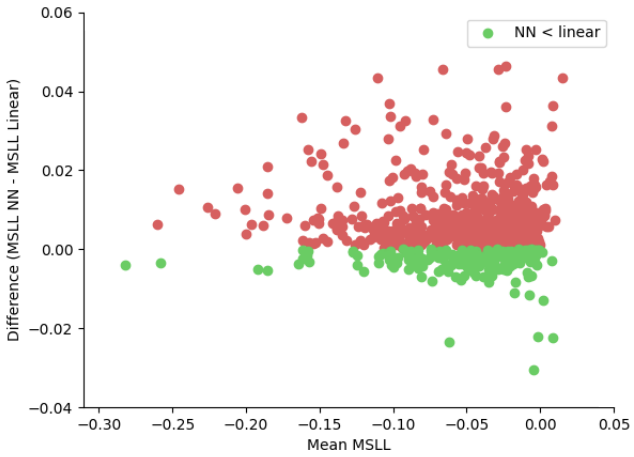
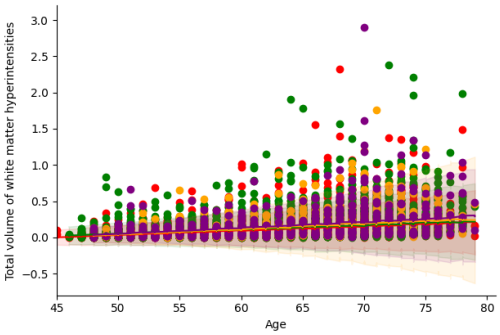
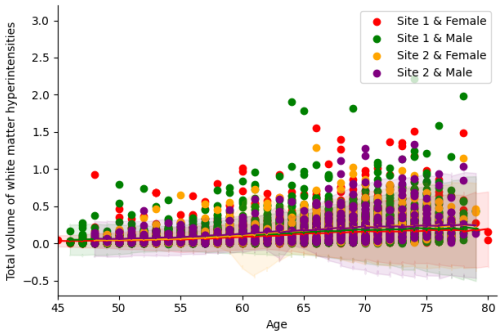


Figure 6: A comparison between good model fit, indicated by a low mean standardized log loss (MSLL), and large difference in the performance between the two models that were evaluated. The two measures are plotted against each other for each image-derived phenotype, showing that a large difference in mean standardized log loss between the two models, does not necessarily mean a good fit to the data. Green dots represent the image derived phenotypes for which the hierarchical Bayesian neural network performed better than the linear HBR model.

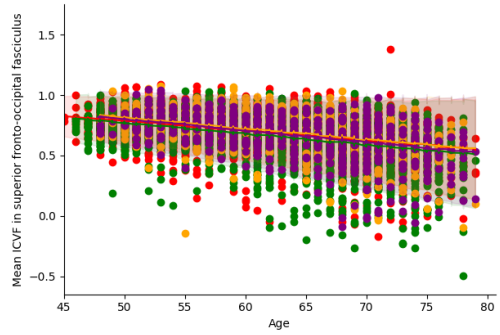
Assuming a Gaussian distribution for the variance, the normative models for three of the selected IDPs are shown in Figure 7. Figures 7a and 7b show the effect of age on the development of white matter hyperintensities for the linear HBR model and the HBNN. Both models fitted well to the mean of the data, with the linear HBR model ($\text{MSLL}_{\text{linear}} = -0.2675$ performing slightly better than the HBNN ($\text{MSLL}_{\text{HBNN}} = -0.2218$). However, it can be seen from both figures that the predicted variance did not fit the data very well. The variance in the data follows a skewed distribution, while the estimated model assumed a Gaussian distribution. Figures 7c and 7d show, again for both models, the estimated effect of age on the intracellular volume fraction (ICVF), which is an estimate of neurite density, in the superior fronto-occipital fasciculus. The HBNN estimated a small non-linearity for the mean, whereas of course the linear HBR model did not have this possibility. Nevertheless, the difference between both models was minor ($\text{MSLL}_{\text{linear}} = -0.1539, \text{MSLL}_{\text{HBNN}} = -0.1522$). Another IDP, indicating the mean L2 in the superior fronto-occipital fasciculus, can be seen in Figure 7e and 7f. The linear HBR model ($\text{MSLL}_{\text{linear}} = -0.1960$) performed slightly better than the HBNN ($\text{MSLL}_{\text{HBNN}} = -0.1749$), but when looking at the estimated variances, we observed a similar pattern as in the top IDP. Namely, the variance was not normally distributed, but the model estimated a normally distributed variance. All the other selected IDPs, along with their MSLL values for both models, can be found in the supplementary materials.



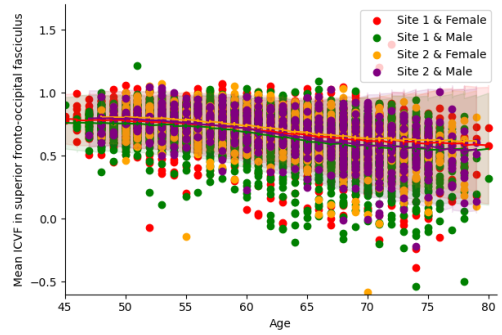
a) $\text{MSLL}_{\text{linear}} = -0.2675$



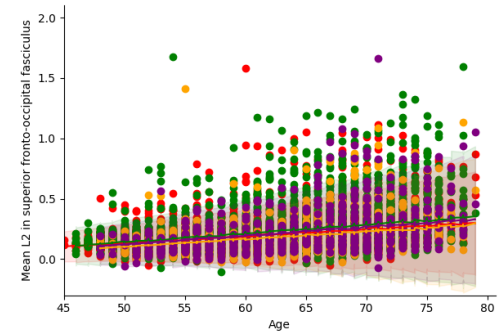
b) $\text{MSLL}_{\text{HBNN}} = -0.2218$



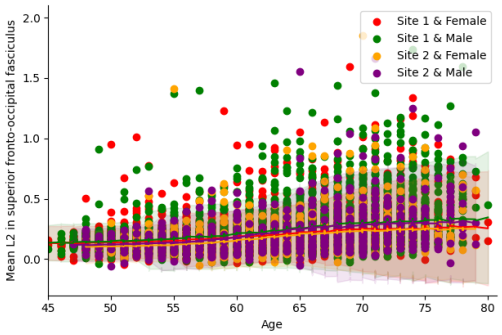
c) $\text{MSLL}_{\text{linear}} = -0.1539$



d) $\text{MSLL}_{\text{HBNN}} = -0.1522$



e) $\text{MSLL}_{\text{linear}} = -0.1960$



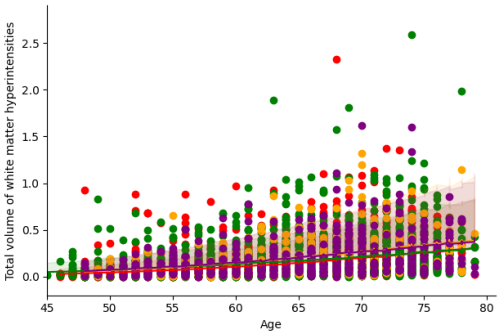
f) $\text{MSLL}_{\text{HBNN}} = -0.1749$

Figure 7: Normative models, along with their mean standardized log losses, for three of the selected image-derived phenotypes. The top two figures show the effect of age on white matter hyperintensity development for a) the linear HBR model and b) the hierarchical Bayesian neural network. The figures in the middle show c) the effect of age on the intra-cellular volume fraction (ICVF) in the superior fronto-occipital fasciculus, estimated by a linear HBR model, and d) the same effect, but estimated by the hierarchical Bayesian neural network. The bottom two model fits show the estimated effect of age on the mean L2 in this same brain region for e) the linear HBR model and f) the hierarchical Bayesian neural network.

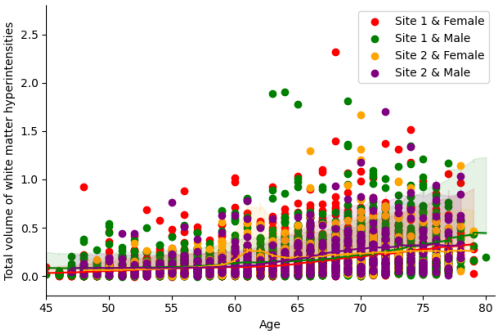
3.2.2 Assuming a skew Gaussian distribution on the likelihood

From the shape of the variance in the IDPs (Figure 7), we can see a clear pattern. Namely, the variances in the data were not only heteroscedastic, but also suggested a non-zero skewness in its distribution. The above results were generated assuming a Gaussian distribution on the likelihood. However, as described in Section 2, both models have the option to sample a skewness parameter for each site-specific model. Here, we repeated the experiments, this time assuming a skew Gaussian distribution on the model likelihood.

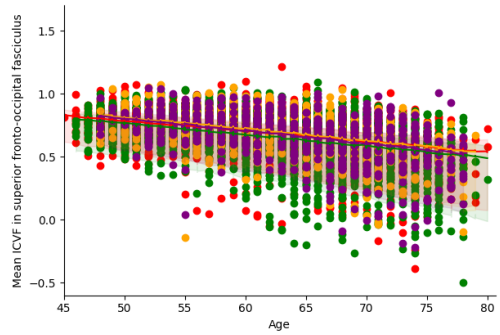
The resulting normative models, using the same IDPs as above, are presented in Figure 8. In Figure 8a and 8b, we see that the HBNN ($\text{MSLL}_{\text{HBNN}} = -0.1102$) had a better performance in estimating the effect of age on the total volume of white matter hyperintensities than the linear HBR model ($\text{MSLL}_{\text{linear}} = -0.0780$). It can be seen from the figures that the skewness of the noise in this IDP was clearly not non-zero, which was being confirmed by the α values, as these were between 9 and 10. Nevertheless, although the models seemed to fit better to the data, it should be noted that the MSLL values assuming a Gaussian distribution suggested a better fit. Figures 8c and 8d show the effect of age on the mean ICVF in the superior fronto-occipital fasciculus. The difference in performance between both models was small under the assumption that the noise was normally distributed (Figure 7c and 7d), and was minor again when assuming a skew Gaussian distribution on the likelihood. However, in this case the HBNN ($\text{MSLL}_{\text{HBNN}} = -0.1600$) performed slightly better than the linear HBR model ($\text{MSLL}_{\text{linear}} = -0.1671$). Moreover, the MSLL values suggested a better model fit when assuming a skew Gaussian distribution. For the third IDP, the mean L2 in the same brain region, the difference in MSLL values for the HBNN ($\text{MSLL}_{\text{HBNN}} = -0.1825$) and the linear HBR model ($\text{MSLL}_{\text{linear}} = -0.1826$) was negligible. As can be seen from both the plots and the α values, the skewness parameter was estimated to be around six.



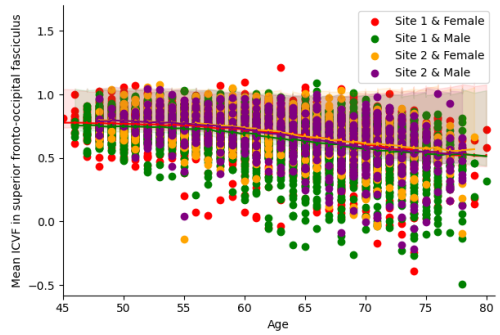
a) $\text{MSLL}_{\text{linear}} = -0.0780$
$\alpha_{10} = 9.9462$, $\alpha_{11} = 9.9482$
$\alpha_{20} = 9.7711$, $\alpha_{21} = 9.7442$



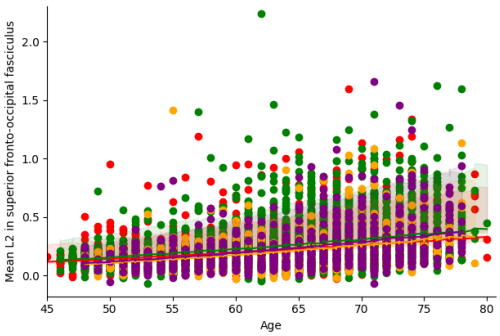
b) $\text{MSLL}_{\text{HBNN}} = -0.1102$
$\alpha_{10} = 9.9436$, $\alpha_{11} = 9.9495$
$\alpha_{20} = 9.7700$, $\alpha_{21} = 9.6951$



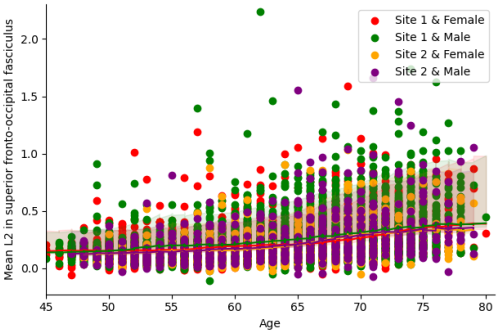
c) $\text{MSLL}_{\text{linear}} = -0.1600$
$\alpha_{10} = -2.2832$, $\alpha_{11} = -2.3113$
$\alpha_{20} = -2.7239$, $\alpha_{21} = -2.0885$



d) $\text{MSLL}_{\text{HBNN}} = -0.1671$
$\alpha_{10} = -2.2623$, $\alpha_{11} = -2.2685$
$\alpha_{20} = -2.6911$, $\alpha_{21} = -2.0383$



e) $\text{MSLL}_{\text{linear}} = -0.1825$
$\alpha_{10} = 4.3485$, $\alpha_{11} = 6.0535$
$\alpha_{20} = 5.6238$, $\alpha_{21} = 5.4800$



f) $\text{MSLL}_{\text{HBNN}} = -0.1826$
$\alpha_{10} = 4.3965$, $\alpha_{11} = 5.7887$
$\alpha_{20} = 6.0104$, $\alpha_{21} = 5.1085$

Figure 8: The same image-derived phenotypes as in Figure 7, but using a skew Gaussian distribution for estimating the prediction variance. The values for skewness parameter α are presented below each MSLL value.

4 Discussion

In this project, we aimed to study brain disorders by means of normative modeling. We extended our previous work on linear HBR models and provided a fully Bayesian neural network. Although the HBR models were able to fit well to the mean and variance of linear data, and model site-effects, they were limited because they could not fit non-linearity. Here, we addressed these limitations by means of a HBNN. This model allowed us to not only fit linear functions in the mean and variance, but also estimate functions with a non-linear effect in the mean or variance. Moreover, the linear HBR models assumed a Gaussian distribution on the likelihood. The HBNN model is more flexible with regards to modeling non-symmetric variance distributions, because it has the possibility to fit an additional skewness parameter, indicating the skewness of the Gaussian likelihood distribution. Besides addressing the limitations of our linear HBR models, we wanted the HBNN to handle site-effects that usually arise when combining multiple neuroimaging datasets from different imaging sites. Site-effects are especially undesired when studying brain disorders, and when trying to distinguish between different groups of patients, because the effect that we are interested in might be much smaller than the site-effects, therefore confounding the deviations that we are looking for in a normative modeling approach. Furthermore, the method had to be computationally scalable to large datasets. The HBNN's ability to estimate the prediction mean and variance, and model site-effects, was compared to the linear HBR model.

First, the performance of the HBNN was evaluated on three simulated datasets. These simulations consisted of two different groups of data, but the groups shared an overarching hierarchy. The simulations contained non-linearity in the mean, and a normally distributed non-linear variance. The resulting models showed a good fit, even when relatively large site-effects were simulated. This finding was confirmed by the model's SMSE and MSLL values, and the Pearson correlation coefficients. Moreover, the model's ability to handle small groups was tested. Here, it is expected that the individual site models will learn from the other groups via the hierarchical structure. Namely, we wanted the individual site models to learn from their shared prior, which was learned from all the data. We tested this using a simulated dataset with one group containing only a limited number of data points. The resulting model fits, SMSE and MSLL values, and the Pearson correlation coefficients showed a good model fit for all groups. To summarise, these results suggest that the HBNN is able to model non-linearity in the mean and variance, and handle site-effects. Furthermore, the hierarchical structure helps estimating a model for groups with little data. The latter can be considered as an advantage for hierarchical models over un-pooling models, because un-pooling models are likely to overfit on small groups.

Even though the model fits to all three simulated data scenarios suggested a good fit, which was confirmed by the performance metrics, there were some problems with the parameter sampling. Observing the sampling traces for each parameter is important, as it can help identify instabilities in the model. Sampling the parameters in a HBNN can be difficult, because there are many parameters to be sampled and they are highly correlated. The sampling traces confirmed this, as we found three distinct problems. First of all, there might be multiple optimal parameter configurations, therefore converging to different values across chains. Narrowing down the priors that we set on the sampling is expected to overcome this limitation. Next, the sampler rejects many samples, therefore resulting in a slow convergence. Last, for some parameters, the sampler has a very small step size, again resulting in a slow convergence of the sampler. These last two problems could be

solved by increasing the number of sampling steps.

Next, the HBNN was tested on a neuroimaging dataset, using IDPs of the UK Biobank dataset. A separate model was fitted for each IDP, and only a few of them showed a better fitting performance of the HBNN than the linear HBR model. However, this was expected because the effect of age on an IDP does not have to be non-linear. If there is only a linear effect in the data, the linear HBR model might already be able to fit well to the data, and therefore the HBNN might not be able to further improve this performance. Instead, for the current project, we were interested in the IDPs that had some non-linear effect of age. IDPs with a good model fit (an MSLL value below -0.15) for the HBNN were selected for an in-depth study, resulting in 30 IDPs. Assuming a Gaussian distribution on the likelihood, the differences in fitting performance were negligible. Nevertheless, these results showed that the HBNN can fit to the mean of the IDPs. Next, we observed that some IDPs showed a skewed variance distribution, and therefore also fitted a skew Gaussian distribution on the likelihood. When using a skew Gaussian distribution for predicting the variance, the performance of the HBNN slightly improved compared to the linear HBR model. However, for IDPs containing a large skewness in the variance, the metrics and visualized model fits suggested a better fit when assuming zero skewness. The skewness parameters might be too extreme. In future work, we propose to decrease the range of the skewness parameters in the sampler to a uniform distribution between -5 and 5.

The results of the test case using neuroimaging data indicate that the HBNN performs as good as, but not better than, the linear HBR model. However, this does not suggest that the HBNN does not have any benefits compared to the already existing hierarchical methods. As the results of the simulated data have shown, the model can fit well to non-linear data containing a hierarchical structure, while it is obvious that the linear HBR model would not be able to fit to these simulated datasets. An explanation of the results on the neuroimaging data could be that the UK Biobank dataset mainly contained a linear effect of age in both the mean and prediction variance, and only some very small non-linearity. To show the benefits of the HBNN over existing linear HBR models, different neuroimaging datasets could be used. An example of such a dataset would be the effect of age on the head circumference or brain size, which is known to be non-linear from birth to 3 years old [15].

We identified two additional improvements for future studies, mainly on the technical aspects. First of all, the MSLL might not be the best measure in this case, because as can be seen in Equation 6, the MSLL is computed by averaging over all participants. The MSLL values can help identifying IDPs where a model fits well to the data to some extend, however, here we were interested in finding not only non-linearity in the mean, but also in the variance. For future work, it is advised that a different metric, which could measure the non-linearity in the variance, and the performance of each model to estimating this variance, would be used. Moreover, we used the priors as described in Section 2. However, depending on the data, these priors might be too uninformative, resulting in regions with a gradient of zero. In our study, this happened to only a few of the image-derived phenotypes, and re-running the model fitting procedure usually solved the problem for our data. However, ideally the priors would be adjusted to the data, and the priors being too uninformative would be prevented. In a next study, this is something that could be improved.

In conclusion, a HBNN was implemented to model non-linearity in the mean and variance, and

to handle site-effects in neuroimaging datasets. The results showed that the model is able to model both the non-linearity and site-effects in a simulated dataset, and that the hierarchical structure of the model allows the individual groups to learn from each other through an overarching prior. Despite these findings, in a future study we would first need to address the sampling problems that were observed, as these problems make the model unstable. Subsequently, on neuroimaging data, using IDPs from the UK Biobank dataset, the HBNN performed as good as the linear HBR model. When using a skewed Gaussian distribution for the model likelihood, we should limit the sampling of the skewness parameter to a smaller range. Our HBNN allows for more flexibility in terms of modeling than the linear HBR model, and therefore it is expected that when using different datasets, containing more non-linearity, would show the advantage of using the HBNN.

References

[1] T. Wolfers, C. F. Beckmann, M. Hoogman, J. K. Buitelaar, B. Franke, and A. F. Marquand, "Individual differences v. the average patient: mapping the heterogeneity in adhd using normative models," *Psychological medicine*, pp. 1–10, 2019.

[2] T. Wolfers, N. T. Doan, T. Kaufmann, D. Alnæs, T. Moberget, I. Agartz, J. K. Buitelaar, T. Ueland, I. Melle, B. Franke, *et al.*, "Mapping the heterogeneous phenotype of schizophrenia and bipolar disorder using normative models," *JAMA psychiatry*, vol. 75, no. 11, pp. 1146–1155, 2018.

[3] T. Wolfers, J. K. Buitelaar, C. F. Beckmann, B. Franke, and A. F. Marquand, "From estimating activation locality to predicting disorder: a review of pattern recognition for neuroimaging-based psychiatric diagnostics," *Neuroscience & Biobehavioral Reviews*, vol. 57, pp. 328–349, 2015.

[4] A. F. Marquand, I. Rezek, J. Buitelaar, and C. F. Beckmann, "Understanding heterogeneity in clinical cohorts using normative models: beyond case-control studies," *Biological psychiatry*, vol. 80, no. 7, pp. 552–561, 2016.

[5] J.-P. Fortin, N. Cullen, Y. I. Sheline, W. D. Taylor, I. Aselcioglu, P. A. Cook, P. Adams, C. Cooper, M. Fava, P. J. McGrath, *et al.*, "Harmonization of cortical thickness measurements across scanners and sites," *Neuroimage*, vol. 167, pp. 104–120, 2018.

[6] W. E. Johnson, C. Li, and A. Rabinovic, "Adjusting batch effects in microarray expression data using empirical bayes methods," *Biostatistics*, vol. 8, no. 1, pp. 118–127, 2007.

[7] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin, *Bayesian data analysis*. CRC press, 2013.

[8] S. M. Kia, H. Huijsdens, R. Dinga, T. Wolfers, M. Mennes, O. A. Andreassen, L. T. Westlye, C. F. Beckmann, and A. F. Marquand, "Hierarchical bayesian regression for multi-site normative modeling of neuroimaging data," *arXiv preprint arXiv:2005.12055*, 2020.

[9] K. L. Miller, F. Alfaro-Almagro, N. K. Bangerter, D. L. Thomas, E. Yacoub, J. Xu, A. J. Bartsch, S. Jbabdi, S. N. Sotiropoulos, J. L. Andersson, *et al.*, "Multimodal population brain imaging in the uk biobank prospective epidemiological study," *Nature neuroscience*, vol. 19, no. 11, pp. 1523–1536, 2016.

[10] L. T. Elliott, K. Sharp, F. Alfaro-Almagro, S. Shi, K. L. Miller, G. Douaud, J. Marchini, and S. M. Smith, "Genome-wide association studies of brain imaging phenotypes in uk biobank," *Nature*, vol. 562, no. 7726, pp. 210–216, 2018.

[11] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256, 2010.

[12] J. Salvatier, T. V. Wiecki, and C. Fonnesbeck, "Probabilistic programming in python using pymc3," *PeerJ Computer Science*, vol. 2, p. e55, 2016.

[13] M. D. Hoffman and A. Gelman, "The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo.," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1593–1623, 2014.

[14] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*, vol. 2. MIT press Cambridge, MA, 2006.

[15] H. C. Hazlett, M. Poe, G. Gerig, R. G. Smith, J. Provenzale, A. Ross, J. Gilmore, and J. Piven, "Magnetic resonance imaging and head circumference study of brain size in autism: birth through age 2 years," *Archives of general psychiatry*, vol. 62, no. 12, pp. 1366–1376, 2005.

Supplementary Materials

Performance metrics of selected image-derived phenotypes

Table S 1: The mean standardized log loss (MSLL) of the image-derived phenotypes where the HBNN performs better than the linear HBR model using both a Gaussian and skew Gaussian distribution for the variance, sorted by $\text{MSLL}_{\text{HBNN}}$. The differences were computed as $\text{MSLL}_{\text{linear}} - \text{MSLL}_{\text{HBNN}}$.

Image-derived phenotypes	Gaussian distribution		Skewed distribution	
	$\text{MSLL}_{\text{HBNN}}$	Difference	$\text{MSLL}_{\text{HBNN}}$	Difference
Volume of grey matter (normalised for head size) (5)	-0.2839	0.0039	-0.2932	0.0001
Volume of peripheral cortical grey matter (normalised for head size) (1)	-0.2599	0.0034	-0.2688	0.0013
Volume of brain, grey+white matter (normalised for head size) (9)	-0.1945	0.0050	-0.1937	0.0008
Mean ISOVF in fornix on FA skeleton (445)	-0.1882	0.0054	-0.1885	0.0003
Mean L1 in fornix on FA skeleton (205)	-0.1659	0.0036	-0.1644	0.0011

Table S 2: The mean standardized log loss (MSLL) of the image-derived phenotypes where the HBNN performs better than the linear HBR model using a Gaussian distribution for the variance. The image-derived phenotypes were sorted by $\text{MSLL}_{\text{HBNN}}$ assuming a Gaussian distribution.

Image-derived phenotype	Gaussian distribution		Skewed distribution	
	$\text{MSLL}_{\text{HBNN}}$	Difference	$\text{MSLL}_{\text{HBNN}}$	Difference
Mean MD in fornix on FA skeleton (109)	-0.1627	0.0032	-0.1609	-0.0014
Volume of ventricular cerebrospinal fluid (normalised for head size) (3)	-0.1614	0.0002	-0.1599	-0.0032
Mean L3 in fornix on FA skeleton (301)	-0.1596	0.0015	-0.1533	-0.0045
Mean FA in fornix on FA skeleton (61)	-0.1591	0.0023	-0.1627	-0.0007
Mean L2 in fornix on FA skeleton (253)	-0.1579	0.0005	-0.1537	-0.0010

Table S 3: The mean standardized log loss (MSLL) of the image-derived phenotypes where the HBNN performs better than the linear HBR model using a skewed Gaussian distribution for the variance. The values were sorted by $\text{MSLL}_{\text{HBNN}}$ using a skewed distribution.

Image-derived phenotype	Gaussian distribution		Skewed distribution	
	$\text{MSLL}_{\text{HBNN}}$	Difference	$\text{MSLL}_{\text{HBNN}}$	Difference
Volumetric scaling from T1 head image to standard space	-0.2383	-0.0152	-0.2515	0.0007
Volume of ventricular cerebrospinal fluid	-0.1981	-0.0038	-0.1979	0.0018
Discrepancy between tfMRI brain image and T1 brain image	-0.1801	-0.0088	-0.1894	0.0001
Mean L2 in superior fronto-occipital fasciculus on FA skeleton (left)	-0.1749	-0.0211	-0.1835	0.0010
Mean L2 in fornix cres+stria terminalis on FA skeleton (right)	-0.1562	-0.0018	-0.1672	0.0102
Mean ICVF in superior fronto-occipital fasciculus (left) (387)	-0.1522	-0.0017	-0.1671	0.0071
Discrepancy between T1 and standard-space brain template (732)	-0.1505	-0.0075	-0.1614	0.0009
Total volume of white matter hyperintensities (781)	-0.2218	-0.0456	-0.1102	0.0322

Table S 4: The mean standardized log loss (MSLL) of the image-derived phenotypes where the linear HBR model performs better than the HBNN in terms of MSLL.

Image-derived phenotype	Gaussian distribution		Skewed distribution	
	$\text{MSLL}_{\text{HBNN}}$	Difference	$\text{MSLL}_{\text{HBNN}}$	Difference
Volume of peripheral cortical grey matter (2)	-0.1932	-0.0062	-0.1921	-0.0011
Volume of grey matter (6)	-0.1956	-0.0102	-0.1971	-0.0004
Volume of white matter (8)	-0.2163	-0.0090	-0.2215	-0.0012
Volume of brain, grey and white matter (10)	-0.2209	-0.0106	-0.2243	-0.0012
Volume of thalamus (left) (11)	-0.1606	-0.0024	-0.1607	-0.0009
Volume of thalamus (right) (12)	-0.1684	-0.0079	-0.1722	-0.0002
Volume of putamen (right) (16)	-0.1538	-0.0015	-0.1564	-0.0003
Mean MD in superior fronto-occipital fasciculus on FA skeleton (left) (147)	-0.1979	-0.0154	-0.1955	-0.0077
Mean L3 in superior fronto-occipital fasciculus on FA skeleton (left) (339)	-0.1786	-0.0141	-0.1841	-0.0028
Discrepancy between dMRI brain image and T1 brain image (737)	-0.1572	-0.0060	-0.1674	-0.0002
Discrepancy between rfMRI brain image and T1 brain image (739)	-0.1854	-0.0062	-0.1938	-0.0015
Scanner transverse (Y) brain position (757)	-0.2569	-0.0062	-0.2613	-0.0021