Weighted Empirical Baysian Model for Quantifying an Atomic Model

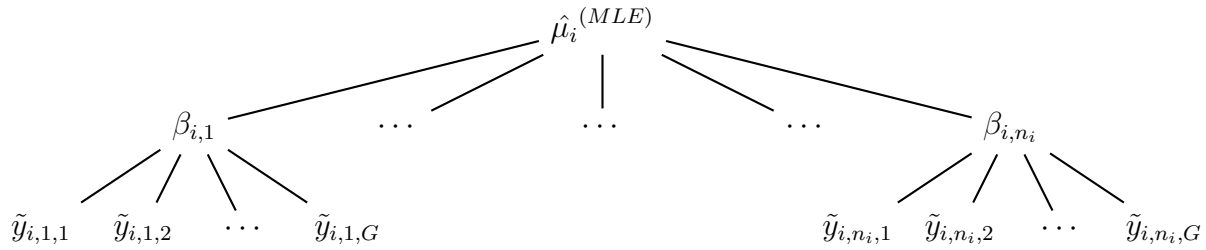
Shu-Cheng Zheng

July 29, 2023

1 Weighted empirical Bayesian Linear Regression Model

To simulate the volume of the structure created by the atomic model and assign it a value, we observe that the shape of the density maps resembles a Gaussian probability density function. Based on this observation, we propose constructing a 3-variate Gaussian kernel function, denoted as ϕ_{ij}, for each type of amino acid (i) and each residue (j). Instead of using a single Gaussian function with a fixed center and variance for all atoms, we suggest assigning distinct variances $\tau_{ij}^2 = \beta_{ij}^{-1}$ to the 20 amino acids. These variances are based on the known charging and polar properties of the amino acids, as these properties are likely to affect their density maps.

To model the variability in the β_{ij} values, we assume that they are drawn from a bivariate Gaussian distribution with a mean of μ_0. To simplify the computation, we adopt an empirical Bayesian approach to estimate the hyperparameters μ_0 using maximum likelihood estimation.



By assigning the k-th grid points x_k to the probability density function ϕ_{ij} with the intensity of atom A_{ij}, we can obtain corresponding voxel values y_{ijk} on those locations.

Accordingly, we derive the following linear regression model

$$
\begin{aligned}
& y_{ijk} = \phi_{ij}(x_k; \tau_{ij}^2) A_{ij} + \epsilon_{ijk} \\
\Rightarrow & y_{ijk} = (2\pi\tau_{ij}^2)^{\frac{-3}{2}} \exp\left[-\frac{1}{2\tau_{ij}^2}|x_k|_2^2\right] A_{ij} + \epsilon_{ijk} \\
\Rightarrow & \log y_{ijk} = -\frac{3}{2}\log 2\pi\tau_{ij}^2 + \log A_{ij} + \frac{1}{\tau_{ij}^2}\left[-\frac{1}{2}|x_k|_2^2\right] + \tilde{\epsilon}_{ijk} \\
\Rightarrow & \tilde{y}_{ijk} = \tilde{x}_k^T \beta_{ij} + \tilde{\epsilon}_{ijk}, \qquad (1)
\end{aligned}
$$

where $\beta_{ij} = [-\frac{3}{2}\log 2\pi\tau_{ij}^2 + \log A_{ij},\ \frac{1}{\tau_{ij}^2}]^T,\ \tilde{x}_k = [1,\ -\frac{1}{2}|x_k|_2^2]^T,$ and $\tilde{\epsilon}_{ijk} \sim N(0, \sigma_{ij}^2)$.

We can represent the logarithm voxel values $\tilde{y}_{ijk}$ on grid points $\tilde{x}_k$ as a G-dimensional vector $\tilde{y}_{ij}$, and we concatenate $\tilde{x}_k$ into a matrix $\tilde{X}$ with dimensions $G \times 2$. The weights for each sample are incorporated into a diagonal matrix W_{ij}, obtained through minimum density power divergence. By considering the weighted linear regression model, we can express the conditional distribution of $\tilde{y}_{ij}$ given β_{ij} as

$$
\tilde{y}_{ij}|\beta_{ij} \sim N_G(\tilde{X}\beta_{ij}, \sigma_{ij}^2 W_{ij}^{-1}) \qquad (2)
$$

Furthermore, we assume that the parameters β_{ij} follow a bivariate Gaussian prior distribution, denoted as

$$
\beta_{ij}|\mu_i \sim N_2(\mu_i, \sigma_{ij}^2 \lambda_{ij}^{-1} I_2). \qquad (3)
$$

Here, λ_{ij} represents the weights between each type of amino acid, which are derived using minimum density power divergence, and I_2 is the 2x2 identity matrix.

1.1 Posterior

To obtain the posterior distribution and derive the empirical Bayesian estimator, we consider the joint probability density function of β_{ij} and $\tilde{y}_{ij}$

$$
\begin{aligned}
\pi(\beta_{ij}, \tilde{y}_{ij}|\mu_i) =& f(\tilde{y}_{ij}|\beta_{ij}, \mu_i)\pi(\beta_{ij}|\mu_i) \\
=& (2\pi\sigma_{ij}^2)^{-\frac{G}{2}} |W_{ij}|^{\frac{1}{2}} \exp[-\frac{1}{2\sigma_{ij}^2}(\tilde{y}_{ij} - \tilde{X}\beta_{ij})^T W_{ij}(\tilde{y}_{ij} - \tilde{X}\beta_{ij})] \\
& (2\pi\sigma_{ij}^2\lambda_{ij}^{-1})^{-1} \exp[-\frac{\lambda_{ij}}{2\sigma_{ij}^2}(\beta_{ij} - \mu_i)^T(\beta_{ij} - \mu_i)] \\
=& (2\pi\sigma_{ij}^2)^{-1} |\tilde{\Sigma}_{ij}|^{\frac{1}{2}} \exp[-\frac{1}{2\sigma_{ij}^2}(\beta_{ij} - \tilde{\mu}_{ij})^T \tilde{\Sigma}_{ij}(\beta_{ij} - \tilde{\mu}_{ij})] \cdot C_{ij}, \qquad (4)
\end{aligned}
$$

where $\tilde{\Sigma}_{ij} = (\tilde{X}^T W_{ij}\tilde{X} + \lambda_{ij} I_2),\ \tilde{\mu}_{ij} = \tilde{\Sigma}_{ij}^{-1}(\tilde{X}^T W_{ij}\tilde{y}_{ij} + \lambda_{ij}\mu_i),$ and
$C_{ij} = (2\pi\sigma_{ij}^2)^{-\frac{G}{2}} |W_{ij}|^{\frac{1}{2}} |\tilde{\Sigma_{ij}}|^{-\frac{1}{2}}$
$\exp\left\{ -\frac{1}{2\sigma_{ij}^2}[-\tilde{\mu}_{ij}^T \tilde{\Sigma}_{ij}\tilde{\mu}_{ij} + \tilde{y}_{ij}^T W_{ij}\tilde{y}_{ij} + \lambda_{ij}\mu_i^T\mu_i] \right\}$

Moreover, we know that the posterior $\pi(\beta_{ij}|\mu_i, \tilde{y}_{ij}) \propto \pi(\beta_{ij}, \tilde{y}_{ij}|\mu_i)$. Hence,

$$\pi(\beta_{ij}|\mu_i, \tilde{y}_{ij}) = (2\pi\sigma_{ij}^2)^{-1}|\tilde{\Sigma}_{ij}|^{\frac{1}{2}} \exp[-\frac{1}{2\sigma_{ij}^2}(\beta_{ij} - \tilde{\mu}_{ij})^T\tilde{\Sigma}_{ij}(\beta_{ij} - \tilde{\mu}_{ij})]. \tag{5}$$

That is,

$$\beta_{ij}|\mu_i, \tilde{y}_{ij} \sim N_2(\tilde{\mu}_{ij}, \sigma_{ij}^2\tilde{\Sigma}_{ij}^{-1}). \tag{6}$$

By giving quadratic loss, the Bayes estimator is the posterior mean,

$$\hat{\beta}_{ij} = \mathbb{E}(\beta_{ij}|\mu_i, \tilde{y}_{ij}) = \tilde{\mu}_{ij} = \tilde{\Sigma}_{ij}^{-1}(\tilde{X}^T W_{ij}\tilde{y}_{ij} + \lambda_{ij}\mu_i). \tag{7}$$

After incorporating the maximum likelihood estimator $\hat{\mu}^{(MLE)}$ and the weights within groups W_{ij} as well as the weights between groups λ_{ij}, we obtain the final result of our Weighted Empirical Bayesian Estimator.

1.2

Hyperparameter MLE for μ_i After intergrating every β_{ij} in type i, we have the marginal likelihood of μ_i

$$\begin{aligned}
L(\mu_i|\tilde{y}_{ij}) =& \int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty}\prod_{j=1}^{n_i}\pi(\beta_{ij}, \tilde{y}_{ij}|\mu_i)d\beta_{i,1}\cdots d\beta_{i,n_i} \\
=& \prod_{j=1}^{n_i} C_{ij}\int_{-\infty}^{\infty}(2\pi\sigma_{ij}^2)^{-1}|\tilde{\Sigma}_{ij}|^{\frac{1}{2}}\exp[-\frac{1}{2\sigma_{ij}^2}(\tilde{\beta}_{ij} - \tilde{\mu}_{ij})^T\tilde{\Sigma}_{ij}(\tilde{\beta}_{ij} - \tilde{\mu}_{ij})]d\beta_{ij} \\
=& \prod_{j=1}^{n_i}(2\pi\sigma_{ij}^2)^{-\frac{G}{2}}\frac{|W_{ij}|^{\frac{1}{2}}}{|\tilde{\Sigma_{ij}}|^{\frac{1}{2}}}\exp\Big\{-\frac{1}{2\sigma_{ij}^2}[-\tilde{\mu}_{ij}^T\tilde{\Sigma}_{ij}\tilde{\mu}_{ij} + \tilde{y}_{ij}^T W_{ij}\tilde{y}_{ij} + \lambda_{ij}\mu_i^T\mu_i]\Big\}
\end{aligned} \tag{8}$$

since $\tilde{\Sigma}_{ij}$ is a diagonal matrix such that we can seperate $\prod_{j=1}^{n_i}\pi(\beta_{ij}, \tilde{y}_{ij}|\mu_i)$ without interaction. We could get $\hat{\mu_i}^{(MLE)}$ of μ_i by maximizing its log liklihood

$$\begin{aligned}
\frac{\partial \log L(\mu_i|\tilde{y}_{ij})}{\partial\mu_i} =& \frac{\partial}{\partial\mu_i}\sum_{j=1}^{n_i}\log\Big[(2\pi\sigma_{ij}^2)^{-\frac{G}{2}}\frac{|W_{ij}|^{\frac{1}{2}}}{|\tilde{\Sigma_{ij}}|^{\frac{1}{2}}}\Big] - \frac{1}{2\sigma_{ij}^2}[-\tilde{\mu}_{ij}^T\tilde{\Sigma}_{ij}\tilde{\mu}_{ij} + \tilde{y}_{ij}^T W_{ij}\tilde{y}_{ij} + \lambda_{ij}\mu_i^T\mu_i] \\
=& \sum_{j=1}^{n_i} -\frac{1}{2\sigma_{ij}^2}\Big[-\lambda_{ij}\tilde{\Sigma}_{ij}^{-1}\cdot 2\tilde{\Sigma}_{ij}\tilde{\Sigma}_{ij}^{-1}(\tilde{X}^T W_{ij}\tilde{y}_{ij} + \lambda_{ij}\mu_i) + 2\lambda_{ij}\mu_i\Big] \overset{Let}{=} 0
\end{aligned}$$

Threfore,

$$\hat{\mu}_i^{(MLE)} = \Big[\sum_{j=1}^{n_i} p_{ij}\Big]^{-1}\Big[\sum_{j=1}^{n_i} p_{ij}(\tilde{X}^T W_{ij}\tilde{X})^{-1}\tilde{X}^T W_{ij}\tilde{y}_{ij}\Big], \tag{9}$$

where $p_{ij} = \lambda_{ij}\tilde{\Sigma}_{ij}^{-1}\tilde{X}^T W_{ij}\tilde{X}/\sigma_{ij}^2$

1.3 Weights

Suppose we have $\tilde{y}_{ijk}|\beta_{ij} \sim N(\tilde{x}_k^T\beta_{ij}, \sigma_{ij}^2)$ with the empirical density function for $z_{ijk} = \tilde{y}_{ijk} - \tilde{x}_k^T\beta$. We want to minimize the density power divergence $d_\alpha(g_n, f)$

$$d_\alpha(g_n, f) = \int f^{1+\alpha}(z)dz - \left(1 + \frac{1}{\alpha}\right) n^{-1} \sum_{k=1}^{n} f^\alpha(z_{ijk}) + C,$$

Plugging in the result derived in **??**, we obtain the MDPDE estimator for β_{ij} as

$$\hat{\beta}_{ij}^{(MDPDE)} = \left(\sum_{k=1}^{G} w_{ijk}\tilde{x}_k\tilde{x}_k^T\right)^{-1} \left(\sum_{k=1}^{G} w_{ijk}\tilde{x}_k^T\tilde{y}_{ijk}\right) = (\tilde{X}^T W_{ij}\tilde{X})^{-1}(\tilde{X}^T W_{ij}\tilde{y}_{ij}),$$

where $w_{ijk} = \exp\left[-\frac{\alpha}{2\sigma_{ij}^2}(\tilde{y}_{ijk} - \tilde{x}_k^T\beta_{ij})^2\right]$, and W_{ij} is a diagonal matrix with weights w_{ijk}.

Similarly, for the bivariate Gaussian distribution $\beta_{ij}|\mu_i \sim N_2(\mu_i, \sigma_{ij}^2 I_2)$, we denote the empirical density function g_n^* for $z_{ij}^* = \beta ij/\sigma_{ij}^2$, and we want to minimize the density power divergence

$$d_\alpha(g_n^*, h) = \int h^{1+\alpha}(z)dz - \left(1 + \frac{1}{\alpha}\right) n^{-1} \sum_{k=1}^{n} h^\alpha(z_{ij}^*) + C.$$

Then the MDPDE estimator for μ_i is then given by

$$\hat{\mu}_i^{(MDPDE)} = (\sum_{j=1}^{n_i} \lambda_{ij})^{-1} \sum_{j=1}^{n_i} \lambda_{ij}\beta_{ij},$$

where $\lambda_{ij} = \sigma_{ij}^{-(\gamma+2)} \exp\left[-\frac{\gamma}{2\sigma_{ij}^2}(\beta_{ij} - \mu_i)^T(\beta_{ij} - \mu_i)\right]$.

Moreover, in the linear regression empirical Bayesian model, the estimator is given by

$$\hat{\beta}_{ij}^{(EB)} = (\tilde{X}^T\tilde{X} + I_2)(\tilde{X}^T\tilde{y}_{ij} + \hat{\mu_i}^{(MLE)}),$$

where the ratio between the data information $\tilde{X}^T\tilde{y}_{ij}$ and the prior information $\hat{\mu_i}^{(MLE)}$ is $G:1$. To normalize the scale of W_{ij} to be G, we modify the equation as follows:

$$w_{ijk} = \frac{G \times \exp\left[-\frac{\alpha}{2\sigma_{ij}^2}(\tilde{y}_{ijk} - \tilde{x}_k^T\beta_{ij})^2\right]}{\sum_{k=1}^{G} \exp\left[-\frac{\alpha}{2\sigma_{ij}^2}(\tilde{y}_{ijk} - \tilde{x}_k^T\beta_{ij})^2\right]},$$

where w_{ijk} represents the normalized weights now. However, the crucial difference lies in the alteration of the ratio between data information and prior information through the weight between groups λ_{ij}. As a result, we arrive at the Weighted Empirical Bayes Estimator as follows:

$$\hat{\beta}_{ij}^{(WEB)} = (\tilde{X}^T W_{ij}\tilde{X} + \lambda_{ij} I_2)^{-1}(\tilde{X}^T W_{ij}\tilde{y}_{ij} + \lambda_{ij}\hat{\mu_i}^{(MLE)}).$$

2 Algorithm

Here's a simplified explanation of the algorithm:

1. Initialization: Initialize the initial parameter estimates.
2. Parameter Estimation Loop: For each type of amino acid (i), repeat the following steps until convergence or a maximum number of iterations (N) is reached:
   a. Weight Calculation: Calculate the weights ($W_{ij}^{(t)}$) and ($\lambda_{ij}^{(t)}$) based on the current estimates of the variance by Minimum Density Power Divergence(MDPDE).
   b. Mean Estimation: Update the mean estimate (${\hat{\mu}_i}^{(t)}$) using the Maximum Likelihood Estimation (MLE) based on the current weights and hyperparameters.
   c. Parameter Update: Update the parameter estimate ($\hat{\beta}_{ij}^{(t)}$) for each type of amino acid using the empirical Baysian model for weighted linear regression model.
   d. Variance Estimation: Update the variance estimate (${\hat{\sigma}_{ij}^2}^{(t)}$) based on the current parameter estimates.
   e. Convergence Check: Calculate the maximum difference ($S^{(t)}$) between the current and previous parameter estimates. If the difference is within the specified tolerance and patience limit, the algorithm stops for the current amino acid type.
3. Repeat the Parameter Estimation Loop for all 20 types of amino acids.

Algorithm Weighted Empirical Bayesian Model

Initialize: ${\hat{\mu}_i}^{(0)} \leftarrow \underset{k}{med}(\hat{\beta}_{ij}^{(OLS)})$, ${\hat{\sigma}_{ij}^2}^{(0)} \leftarrow \underset{k}{med}(\tilde{y}_{ijk} - \tilde{X}_k^T \hat{\beta}_{ij}^{(OLS)})^2$,
$\hat{\beta}_{ij}^{(0)} \leftarrow \hat{\beta}_{ij}^{(EB)}$, $S^{(0)} \leftarrow -\infty$, $t \leftarrow 1$

for $i \leftarrow 1$ to 20 **do**

while $t \leq N$ and ($S^{(t)} \leq S^{(t-1)}$ with tolerance 10^{-5} and patience 3) **do**

$W_{ij}^{(t)} \leftarrow Normalize\Big\{diag\Big\{ \exp\big[-\frac{\alpha}{2{\hat{\sigma}_{ij}^2}^{(t-1)}}(\tilde{y}_{ijk} - \tilde{X}_k^T \hat{\beta}_{ij}^{(t-1)})^2\big]\Big\}\Big\}$

$\lambda_{ij}^{(t)} \leftarrow {\hat{\sigma}_{ij}^2}^{(t-1)\frac{-\gamma-2}{2}} \exp[-\frac{\gamma}{2{\hat{\sigma}_{ij}^2}^{(t-1)}}(\hat{\beta}_{ij}^{(t-1)} - {\hat{\mu}_i}^{(t-1)})^T(\hat{\beta}_{ij}^{(t-1)} - {\hat{\mu}_i}^{(t-1)}]$

${\hat{\mu}_i}^{(t)} \leftarrow {\hat{\mu}_i}^{(MLE)}(W_{ij}^{(t)}, \lambda_{ij}^{(t)})$

$\hat{\beta}_{ij}^{(t)} \leftarrow (\tilde{X}^T W_{ij}^{(t)} \tilde{X} + \lambda_{ij}^{(t)} I_2)^{-1}(\tilde{X}^T W_{ij}^{(t)} \tilde{y}_{ij} + \lambda_{ij}^{(t)} \hat{\mu}_i^{(t)})$

${\hat{\sigma}_{ij}^2}^{(t)} \leftarrow \underset{k}{med}(\tilde{y}_{ijk} - \tilde{X}_k^T \hat{\beta}_j^{(t)})^2$

$S^{(t)} \leftarrow \underset{j}{max}\big[(\hat{\beta}_{ij}^{(t)} - \hat{\beta}_{ij}^{(t-1)})^T(\hat{\beta}_{ij}^{(t)} - \hat{\beta}_{ij}^{(t-1)})\big]$

end while

end for=0

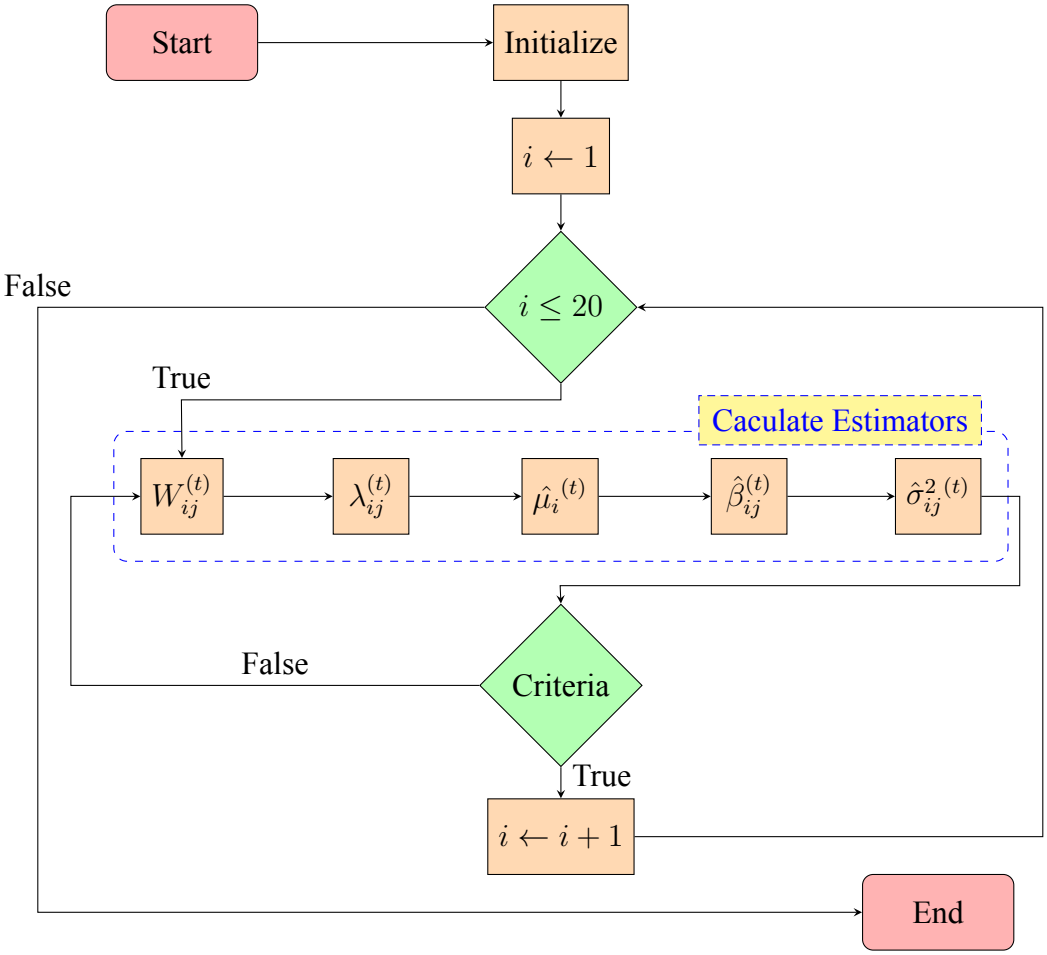


Figure 1: The Flow Chart of Our Algorithm

3 Simulation

To simulate the situation of data contamination, we construct clean data following these steps:

1. Generate clean parameters:
    a. Set the mean vector for each group as $\mu_i = [-6, 6]^T$.
    b. Sample the variance for each group from an inverse gamma distribution, i.e., $\sigma_{ij}^2 \sim \text{InvGamma}(400, 0.1)$.
    c. Generate clean coefficients for each group, i.e., $\beta_{ij} \sim N_2(\mu_i, \sigma_{ij}^2)$, where the total number of groups is denoted by n_i.
    d. Generate clean data points for each group, i.e., $\tilde{y}_{ij} \sim N_G(\tilde{X}\beta_{ij}, \sigma_{ij}^2)$, where $\tilde{X}$ is a matrix with G columns representing the grid points.
2. Introduce outliers between groups:
    a. The outliers between each group account for a ratio R_1 and are generated from another distribution.
    b. Specifically, for each group i, we generate $n_i \times R_1 = n_i'$ outliers according to $\beta_{ij}^{(C1)} \sim N_2(\begin{bmatrix} -8 \\ 8 \end{bmatrix}, \sigma_{ij}^2 I_2)$.
3. Introduce outliers within groups:
    a. The outliers within each group account for a ratio R_2 and are also generated from another distribution.
    b. For all the grid points k, we generate $G \times R_2 = G'$ outliers according to $\tilde{y}_{ij}^{(C2)} \sim N_{G'}(\tilde{X}_k \beta_{ij}^{(C2)}, \sigma_{ij}^2 I_{G'})$, where $\beta_{ij}^{(C2)} \sim N_2(\begin{bmatrix} -4 \\ 4 \end{bmatrix}, \sigma_{ij}^2 I_2)$.
4. Replace the proportion of clean data with contaminated data

By following this procedure, we can simulate a dataset with clean data points and introduce outliers to mimic a real-world scenario with some contaminated data as the below schematic diagram

Initially, we set $i = 2, n_i = 5,$ and $G = 600$ with a contaminated ratio between groups of $R_1 = 0.2$ and a ratio within groups of $R_2 = 0.2$. The following fig. 3 demonstrates that each group fits well even though it is generated from a contaminated distribution.

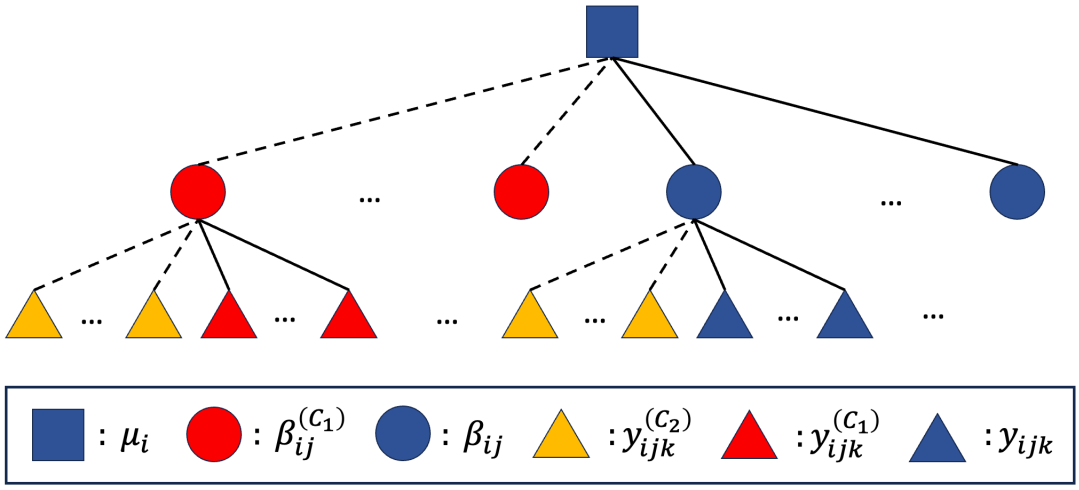


Figure 2: Contamniated Data Schematic Diagram

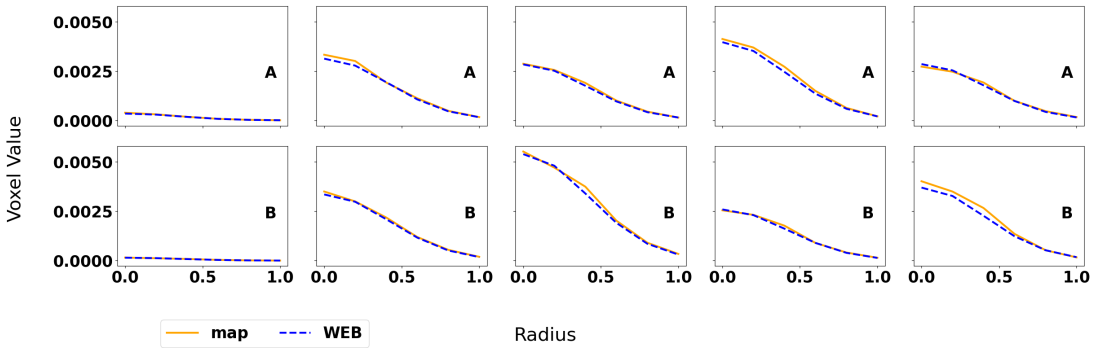


Figure 3: The MSE of the true hyperparameter μ_i and the MLE $\hat{\mu}_i^{(MLE)}$

Now, we extend our simulation to include a single group consisting of 50 samples, each with 600 data points. Additionally, we vary the ratio of points and betas from 0 to 0.45 with an increment of 0.05. As a result, we have a total of 81 experimental outcomes to analyze. The fixed values for α and γ were set at 0.5.

Figure 4 demonstrates that, in general, the mean squared error of the hyperparameter μ_i increases as the contamination ratio of points and betas increase. However, the influence of the contamination ratio of betas is slightly more significant than that of the contamination ratio of points. On the other hand, Figure 5 illustrates that the contamination ratio of betas has less impact on the empirical Bayesian estimator. The mean squared error of betas β_{ij} drops smoothly as the contamination ratio of points increases but increases dramatically after reaching a ratio of 0.4. This behavior may be attributed to our robust estimator, which maintains robustness in the presence of outliers. However, it results in a slight loss of efficiency, and the weight may not handle an excessive number of outliers. Fine-tuning the values of α and γ can potentially address this issue.

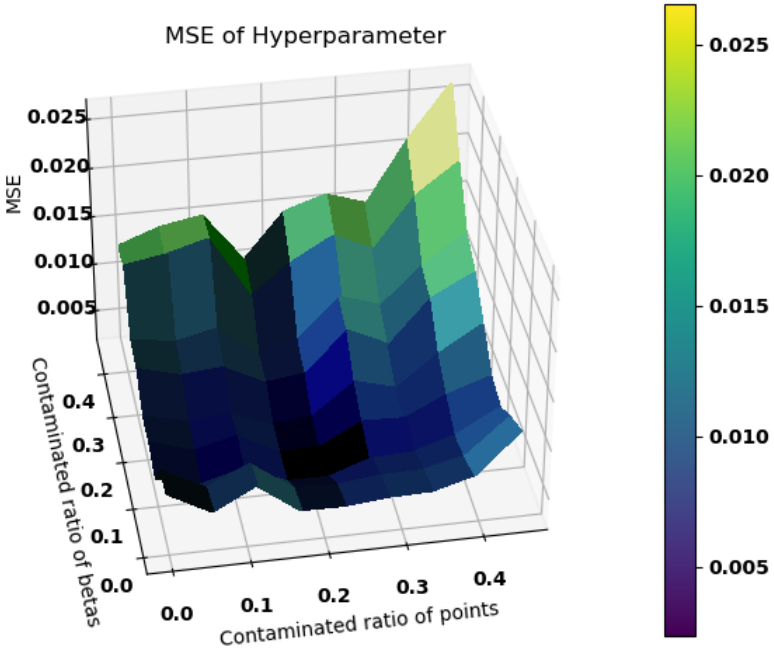


Figure 4: The MSE of the true hyperparameter μ_i and the MLE $\hat{\mu}_i^{(MLE)}$

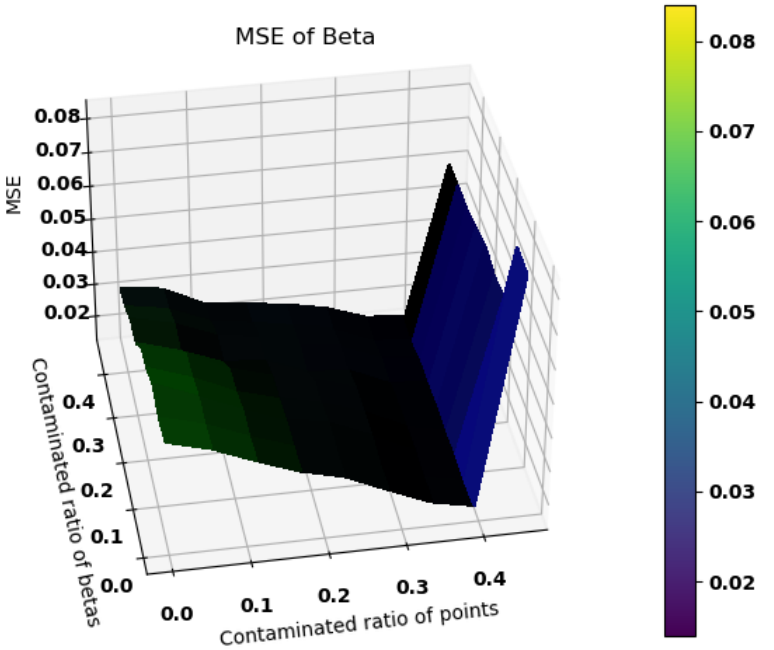


Figure 5: The MSE of the true parameter β_{ij} and the WEB esitmator $\hat{\beta_{ij}}^{(WEB)}$

Overall, the scale of both mean squared error figures is not too high, indicating that our estimation is robust in downweighting outliers and producing stable results even in the presence of contamination. This robustness is a desirable property, as it ensures that our model is less affected by extreme observations and provides reliable parameter estimates even in challenging data scenarios.

4 Real Data

Here, we are focusing on analyzing the protein structure of 6Z6U, which represents the human apoferritin. The structure was obtained from Titan Mono-BCOR microscope and has a resolution of 1.25Å.

In the atomic model, precise positions are defined for each atom within the protein, with a particular focus on the C_α atom, which is the C atom connected to the side chain. To study the protein's density, we employ a grid of points surrounding each C_α coordinate. The grid is designed to maintain equal density and spans a distance of 2Å , with a gap of 0.2Å between each point. Initially, we start with 40 points within a 0.2Å radius from each atomic coordinate. To control computational complexity, we set a maximum of 1000 points for the densest regions. By setting up this grid, we can analyze the protein's density in different regions and gain insights into its structural properties.

The fig. 6 illustrates that the density map exhibits characteristics resembling half of a Gaussian kernel function when the radius is less than 1 Å for each C_α of every amino acid. Consequently, we opt to fit the model using the data within a radius of 1 Å.

To illustrate the impact of our weighting approach, we initially observe that the empirical Bayesian estimator performs poorly in **??** due to the presence of outliers between and within groups. However, after employing our weighted empirical Bayesian model, we obtain the maximum likelihood estimator $\hat{\mu}_i^{(MLE)}$ for each amino acid hyperparameter, as depicted in Figure 8. Additionally, we can notice that even though we use weight W_{ijk} to downweight the outliers within groups, we still need to assign weight λ_{ij} to control the influence of the outliers between groups, which is particularly evident for the "SER" and "PRO" amino acids. This highlights the effectiveness of our model in reducing the impact of outliers in each group.

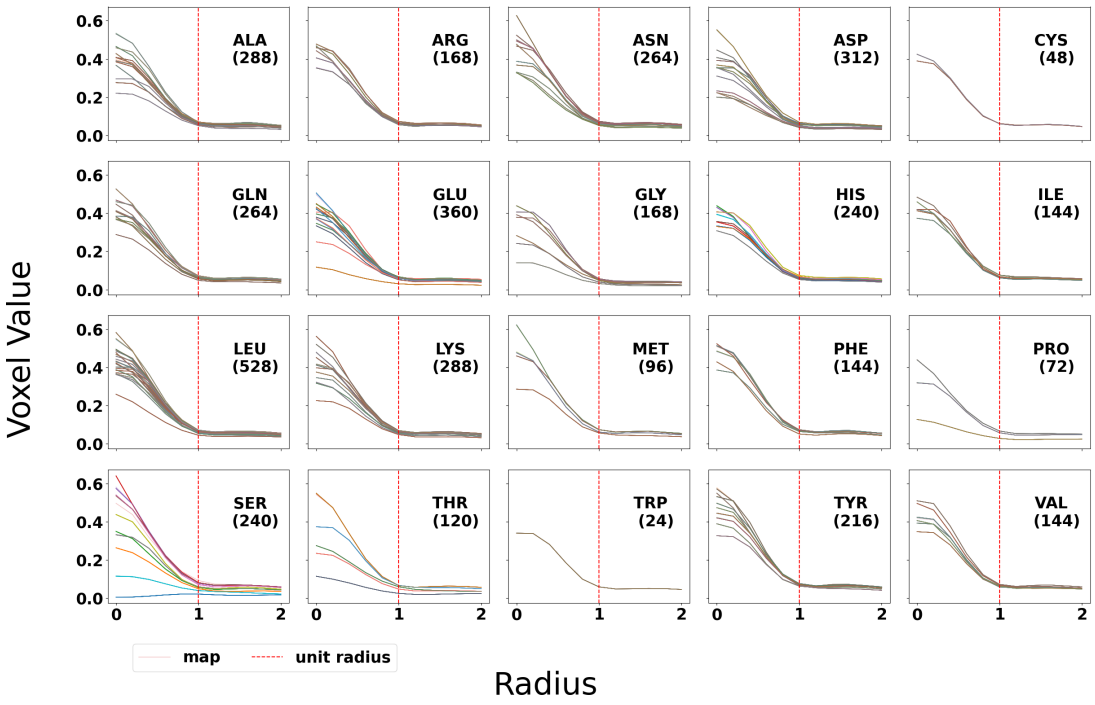


Figure 6: Density Maps of 6Z6U for 4128 residues

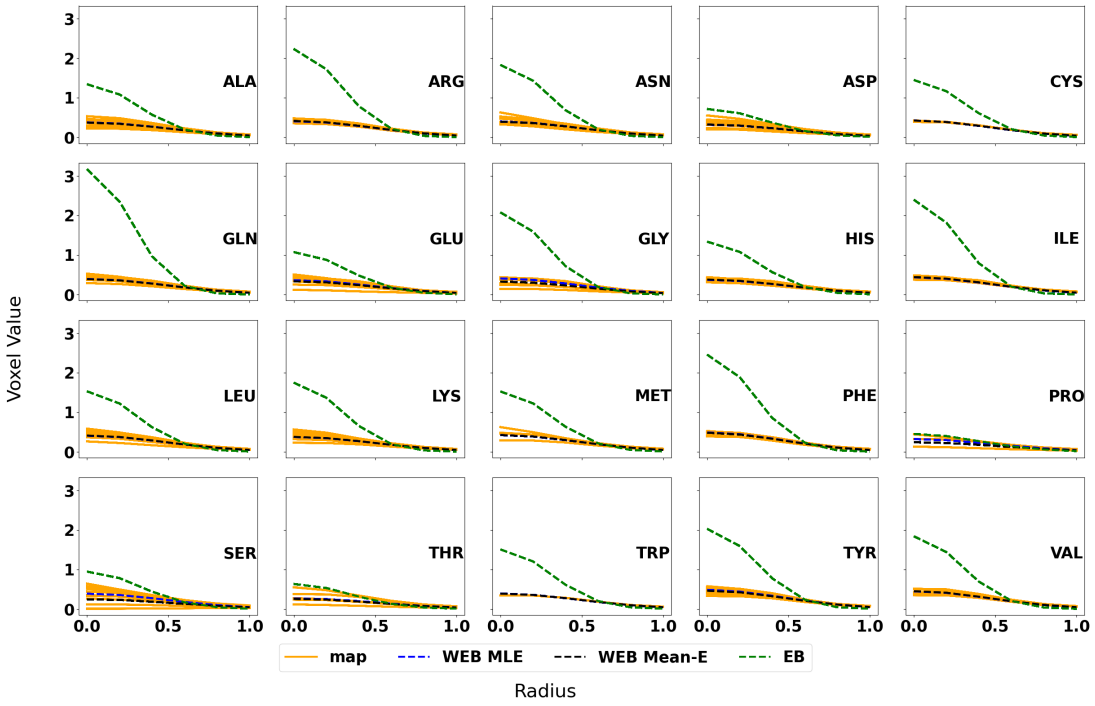


Figure 7: Fitted Result with $\alpha = 0.1$ and $\gamma = 0.1$

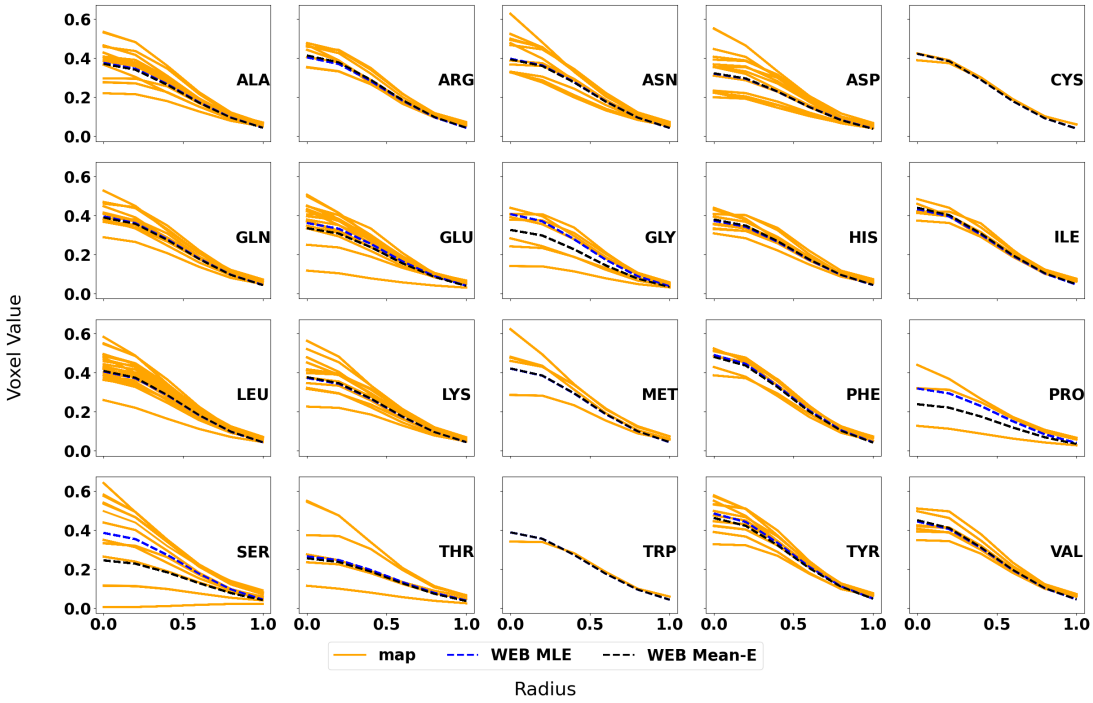


Figure 8: Fitted Result with $\alpha = 0.1$ and $\gamma = 0.1$

Moreover, to quantify outliers, we use the maximum likelihood estimator $\hat{\mu}_i^{(MLE)}$ to represent the hyperparameter for each group and compute the statistical distances between the $\hat{\beta}_{ij}^{(WEB)}$ and $\hat{\mu}_i^{(MLE)}$

$$d(\hat{\beta}_{ij}^{(WEB)}, \hat{\mu}_i^{(MLE)}) = (\hat{\beta}_{ij}^{(WEB)} - \hat{\mu}_i^{(MLE)})^T \Sigma_i^{-1} (\hat{\beta}_{ij}^{(WEB)} - \hat{\mu}_i^{(MLE)}) \tag{10}$$

with $\Sigma_i = \sum_{j=1}^{n_i} \lambda_{ij} (\hat{\beta}_{ij}^{(WEB)} - \hat{\mu}_i^{(MLE)})(\hat{\beta}_{ij}^{(WEB)} - \hat{\mu}_i^{(MLE)})^T / \sum_{j=1}^{n_i} \lambda_{ij}$.

Exactly, to visualize whether they are outliers, we display 3 standard deviation confidence regions with 99.73% for all amino acids in fig. 9. Any betas that fall outside this region are identified as outliers. To quantify this, we calculate the statistical distances for each residue, which follows a χ_2^2 distribution. If the statistical distance for any residue exceeds the critical value $\chi_2^2(0.0027)$, it is classified as an outlier, as fig. 10.

After identifying outliers, they account for about 3% of the residues. As shown in Figure 11, these outliers significantly deviate from the corresponding $\hat{\mu}_i^{(MLE)}$ in each amino acid. It's important to note that if the standard deviation of betas in each amino acid is too small, even a slight deviation from the mean may be classified as an outlier.

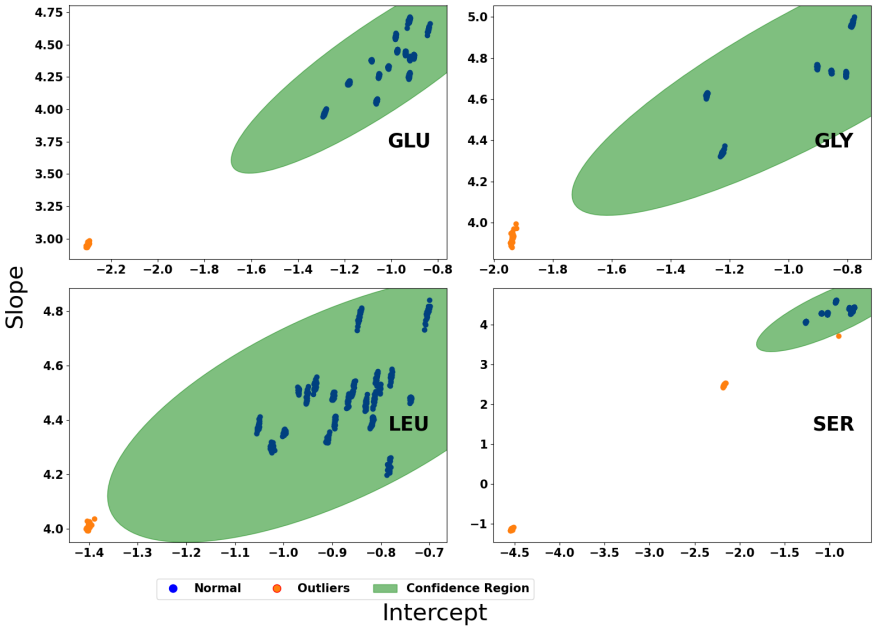


Figure 9: The scatter of $\hat{\beta}_{ij}^{(WEB)}$ and the Confidence Region

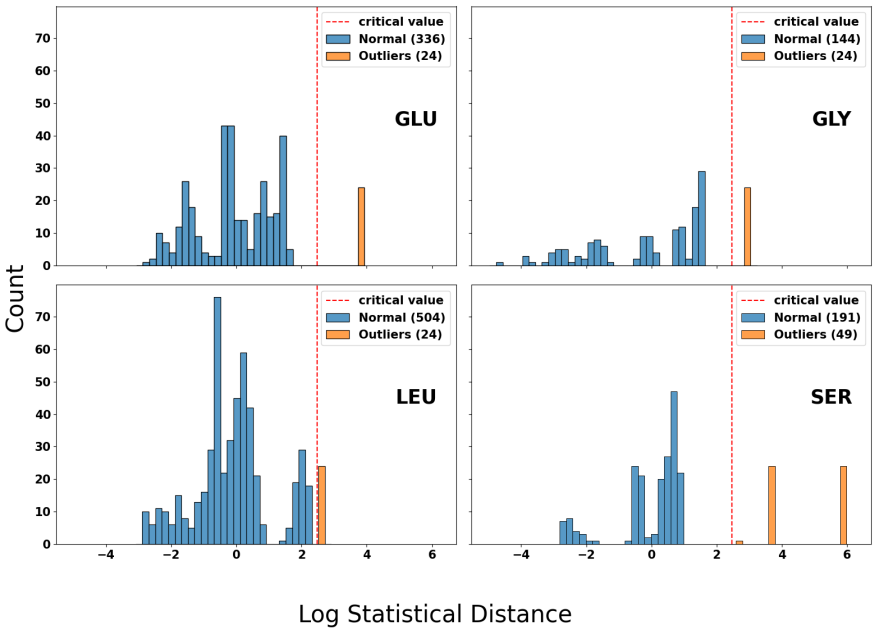


Figure 10: The histogram of $\hat{\beta}_{ij}^{(WEB)}$ and the critical vaue line

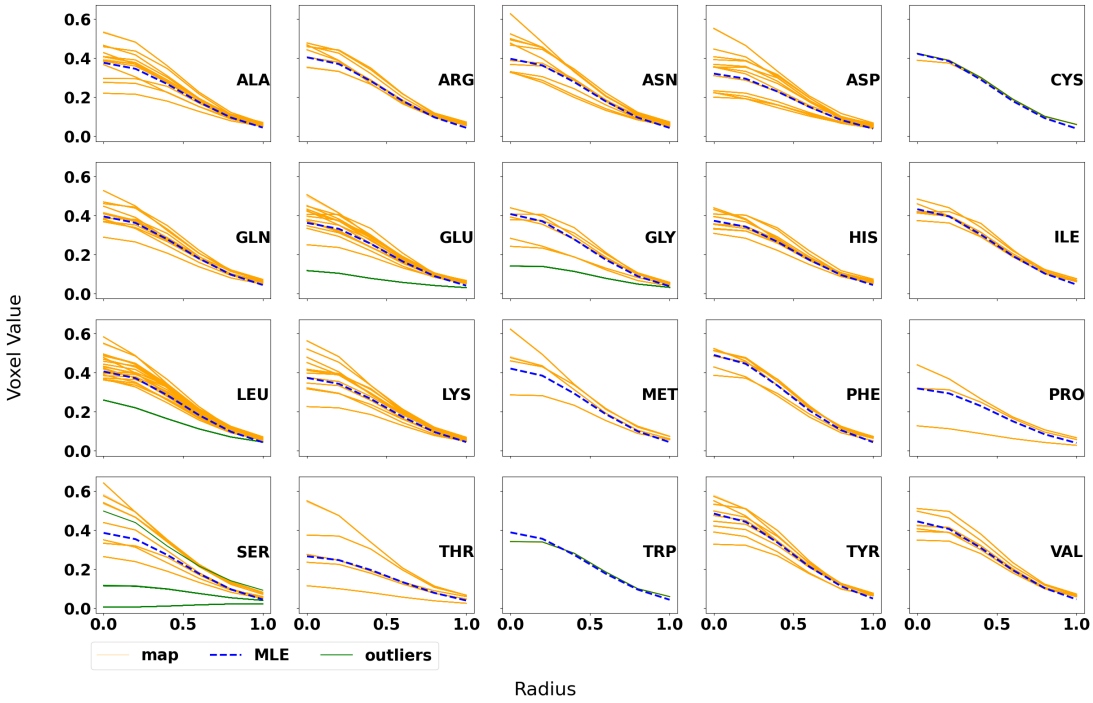


Figure 11: The density maps that are identified as outliers by $\hat{\beta}_{ij}^{(WEB)}$

References

[1] Andrew Gelman, John Carlin, Hal Stern, David Dunson, Aki Vehtari, and Donald Rubin. *Bayesian Data Analysis, 3rd Ed.* 01 2014.

[2] Ayanendranath Basu, Ian R Harris, Nils L Hjort, and MC Jones. Robust and efficient estimation by minimising a density power divergence. *Biometrika*, 85(3):549–559, 1998.

[3] Abhik Ghosh and Ayanendranath Basu. Robust estimation for independent non-homogeneous observations using density power divergence with applications to linear regression. 2013.

[4] Grigore Pintilie, Kaiming Zhang, Zhaoming Su, Shanshan Li, Michael F Schmid, and Wah Chiu. Measurement of atom resolvability in cryo-em maps with q-scores. *Nature methods*, 17(3):328—334, March 2020.