Methods

Proposed Model For Neuropsychological Outcomes

In the context of neuropsychological outcomes, the vector μ_{ij} represents the components of subject i's underlying cognition at observation j for $i \in \{1, 2, ..., n\}$ and $j \in \{1, 2, ..., J\}$. The observation y_{ij} are scores from a test aimed to measure the true underlying cognition of μ_{ij}. We propose modeling the underlying cognition μ_{ij} broken into two components, 1.) β, which is a population parameter that captures the linear trajectory based on a subject's measured characteristics and 2.) α, which captures subject specific random variation in cognition. This proposed model is a special case of the SMM called a Local Linear Trend Model (LLT) and can be written as,

$$\begin{aligned} y_{ij} =& \alpha_{ij} + x_{ij}\beta_j + \varepsilon_{ij}, \quad \varepsilon_{ij} \sim N(0, \sigma^2_\varepsilon) \\ \mu_{ij} =& \begin{bmatrix} \alpha_{ij} \\ \beta_j \end{bmatrix} = \begin{bmatrix} \alpha_{i(j-1)} \\ \beta_{j-1} \end{bmatrix} + \begin{bmatrix} \eta_{ij} \\ 0_{p\times 1} \end{bmatrix}, \quad \eta_{ij} \sim N(0, \delta_{ij}\sigma^2_\eta) \\ &\alpha_{i0} \sim N(a_0, P_0), \quad \beta_0 \sim N(\beta, 0) \end{aligned} \tag{1}$$

The vector $x_{ij} \in R^p$ are the independent variables for subject i at the j^{th} observation. Let t_{ij} be the observation time for the j^{th} observation. Similar to what is used in LMEM modeling, we can allow $x_{ij} = t_{ij} * x_{i0}$, representing a time interaction with baseline covariates. Additionally, a single element of x_{ij} can be set to 1 which will correspond to the intercept parameter common in linear modeling. We assume $\beta \in R^p$ remains fixed over time and follows the same linear effect interpretation common with other linear regression methods. The linear effects of β_j is a population parameter as there is no index for subject i. The proposed model differs from traditional models in that there is a randomly varying subject specific α_{ij} that follow a random walk through time. With respect to y_{ij}, α_{ij} follows the random walk about $x_{ij}\beta_j$ and therefore can be interpreted as random variation in cognition not accounted for by the predictors in x_{ij}.

The variance of the underlying state α_{ij} can be time dependent as the j^{th} observation can occur at any time. To accommodate unequal observation times we can adjust the variance of α_{ij} by multiplying the population variance σ^2_η with $\delta_{ij} = t_{ij} - t_{i(j-1)}$. This allows us to estimate the underlying α_{ij} variance σ^2_η, even when each subject's underlying α_{ij} variance is different due to heterogeneous observation times.

The aim of modeling α is to capture unobserved time varying effects on the outcome of interest. Notice, $E(\alpha_{ij}|\alpha_{i(j-1)} = \tilde{a}_{i(j-1)}) = \tilde{a}_{i(j-1)}$. This indicates that the underlying variations in one's cognitive state at observation j, not accounted for by the predictors x_{ij}, is centered on their underlying cognitive state at observation $j - 1$ and can freely vary up or down from that state. The freedom of variation in the subject specific cognition is much more flexible than restriction enforced in commonly used LMEM models.

A simple comparison to the LLT is a random intercept LMEM with AR(1) errors in the model. The LMEM AR(1) can be generalized as,

$$\begin{aligned} y_{ij} &= b_{io} + \beta x_{ij} + e_{ij}, \quad b_{i0} \sim N(0, \sigma^2_b) \\ e_{ij} &= \rho e_{i(j-1)} + \epsilon_{ij}, \quad \epsilon_{ij} \sim N(0, \sigma^2_\epsilon), \quad \rho \in (-1, 1) \end{aligned}$$

The linear effect is represented by β and x_{ij} are the covariates of subject i at observation j. The value e_{ij} can be considered the underlying cognitive state at observation j. If $\rho = 1$ then e_{ij} can vary freely and this model mirrors the LLT without a measurement error on the observation equation. However, if $|\rho| < 1$ then $|E(e_{ij}|e_{i(j-1)} = \tilde{e}_{i(j-1)})| = |\rho\tilde{e}_{i(j-1)}| \leq |\tilde{e}_{i(j-1)}|$, meaning the underlying state is reverting back to the level it held at time 0. The LLT relaxes the restrictive mean reverting assumption and allows the subject specific underlying state to vary more freely.

Autocorrelation

Our proposed model implies a flexible dynamic moving average auto-correlation structure. Suppose $\delta_{ij} = 1$ for all j. The correlation between any two time points for subject i is defined as,

$$corr(y_{ij}, y_{i(j+\tau)}) = \frac{j\sigma^2_\eta + P_0}{\sqrt{\sigma^2_\varepsilon + j\sigma^2_\eta + P_0}\sqrt{\sigma^2_\varepsilon + (j+\tau)\sigma^2_\eta + P_0}}$$

If there is no variation in α over time ($\sigma^2_\eta = 0$) then $corr(y_{ij}, y_{i(j+\tau)}) = \frac{P_0}{\sigma^2_\varepsilon + P_0}$. This situation provides an observationally equivalent parallel to a random intercept LMEM. Consider the reduced model of the LLT,

$$y_{ij} = \alpha_{i0} + \sum_{j=1}^{t} \eta_{ij} + x_{ij}\beta_0 + \varepsilon_{ij}$$

If $\sigma^2_\eta = 0$ then $\eta_j = 0$ for $j \in \{1, 2, ..., J\}$. The model further reduces to,

$$y_{ij} = \alpha_{i0} + x_{ij}\beta_0 + \varepsilon_{ij}$$

Where $\alpha_{i0} \sim N(a_0, P_0)$, which is directly comparable to a linear mixed effect model with a random intercept. This highlights that the proposed model can accommodate the simplistic LMEM while also accommodating more complex temporal auto-correlation.

General Linear Gaussian State Space Model

The proposed LLT model is derived from the general linear state space model. For the general linear SSM we use matrix notation and instead use the bold face vector $y_j = [y_{1j}, y_{2j}, ..., y_{nj}]'$ as the outcome and $\mu_j = [\mu_{1j}, \mu_{2j}, ..., \mu_{qj}]'$ as the latent state. A general linear state space model can be denoted as:

$$\begin{aligned} y_j &= F_j\mu_j + v_j, \quad v_j \sim N(0, V_j) \\ \mu_j &= G_j\mu_{j-1} + w_j, \quad w_j \sim N(0, W_j) \;\;\&\;\; \mu_0 \sim N(u, P) \end{aligned} \tag{2}$$

Both v_j and w_j are independent and mutually uncorrelated. The vector $\mu_j \in R^q$ represents the unobserved true state of the process. Estimating the true underlying process μ_j is the typical aim in SSM modeling. To estimate μ_j we use the observations y_j. The observations y_j are assumed to be a linear combination of μ_j after being transformed by the observation matrix F_j, which is fixed by design, plus the random noise v_j. The latent states follow a Markov Chain process where μ_j is a linear function of μ_{j-1} in the form $G_j\mu_{j-1}$ with the random noise w_j. The matrix G_j is the state transition matrix and is also assumed fixed by design. This state space model can be regarded as a Hidden Markov model with continuous latent states and continuous observations.

Here, we denote $\hat{\mu}_{a|b} = E(\mu_a|y_1, y_2, ..., y_b)$ and $P_{a|b} = \text{Var}(\mu_a|y_1, y_2, ..., y_b)$ for a and b in $\{1, 2, ..., J\}$. With known initial parameters of underlying state μ_0 (mean u and variance P) and variance parameters (V_j and W_j) we can utilize the popular Kalman Filter and Kalman Smoother to calculate $\hat{\mu}_{j|j} = E(\mu_j|y_1, y_2, ..., y_j)$ and $P_{j|j} = \text{Var}(\mu_j|y_1, y_2, ..., y_j)$ for j in $\{1, 2, ..., J\}$. To estimate the unobserved state μ_j conditioned on the observed data $y_{1:t} = \{y_1, y_2, ..., y_j\}$ we can utilize The Kalman Filter [@linFilt, @durbin_koopman_2012]. The expected value and variance of the unobserved states conditioned on all the data can then be calculated by iterating backwards through the Kalman Smoother. The Kalman filter and smoother is a maximum likelihood recursive algorithm that proceeds as shown in algorithm 1.

Algorithm 1 Kalman Filter and Smoother

```
Kalman Filter
For $j$ in $1, 2, ..., J$:
    1. Predicted state: $\hat{\mu}_{j|j-1} = G_j \hat{\mu}_{j-1|j-1}$
    2. Predicted state covariance: $P_{j|j-1} = G_j P_{j-1|j-1} G_j^{'} + W$
    3. Innovation covariance: $S_j = F_j P_{j|j-1} F_j^{'} + V$
    4. Kalman Gain: $K_j = P_{j|j-1} F_j' S_j^{-1}$
    5. Innovation: $\tilde{f}_j = y_j - F_j \hat{\mu}_{j|j-1}$
    6. Updated state estimate: $\hat{\mu}_{j|j} = \hat{\mu}_{j|j-1} + K_j \tilde{f}_j$
    7. Updated state covariance: $P_{j|j} = (I - K_j F_j) P_{j|j-1}$
    8. Updated innovation: $\tilde{f}_{j|j} = y_j - F_j \hat{\mu}_{j|j}$
Where $\hat{\mu}_{0|0} = u$ and $P_{0|0} = P$.

Kalman Smoother
For $j^*$ in J, J-1, ..., 1:
    1. Smoothed predicted state: $\hat{\mu}_{j^*|J} = \hat{\mu}_{j^*|j^*} + L_j^*(\hat{\mu}_{(j^*+1)|J} - \hat{\mu}_{(j^*+1)|j^*})$
    2. Smoothed predicted state covariance: $P_{j^*|J} = P_{j^*|j^*} - L_{j^*} G_{(j^*+1)} P_{j^*|j^*}$
Where $L_{j^*} = P_{j^*|j^*} G^{'}_{(j^*+1)} + P^{-1}_{(j^*+1)|j^*}$.
```

Proper estimates for μ_j depend on correctly specifying parameters the variance parameters V_j and W_j. When V_j and W_j are unknown these parameters can be estimated by maximizing the joint log-likelihood using the limited memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) method [@conOpt, @limBFGS].

$$\ell(V_j, W_j) = -\frac{np}{2} log(2\pi) - \frac{1}{2}\sum_{i=1}^{t}(log|\tilde{S_i}| + \tilde{f_i}\grave{}S_i^{-1}f_i)$$

To get a final estimate of the SSM model, (1) run the Kalman Filter and Smoother, (2) compute and maximize the log-likelihood to estimate variance parameters, (3) iterate through (1) and (2) until convergence criteria is met.

Another important aspect is correctly specifying the initial parameters of μ_0 (mean u and variance P), which are often unknown. However, by giving μ_0 an "infinite prior" of $P = \infty$ and $u = 0$, after only a few iterations of the Kalman Filter our estimates $P_{i|j}$ converge to the values they would have taken if we would have started with the true P and u [@durbin_koopman_2012].

Formulating Proposed Model Into State Space Form

The proposed model in equation 1 can be rewritten to fit in the general matrix state space model framework from equation 2,

$$\begin{aligned}
y_j &= \begin{bmatrix} I_n & X_j \end{bmatrix} \begin{bmatrix} \alpha_j \\ \beta_j \end{bmatrix} + \varepsilon_j \\
\begin{bmatrix} \alpha_j \\ \beta_j \end{bmatrix} &= \begin{bmatrix} I_{(n+p)} \end{bmatrix} \begin{bmatrix} \alpha_{j-1} \\ \beta_{j-1} \end{bmatrix} + \begin{bmatrix} \eta_j \\ 0_{p\times 1} \end{bmatrix}
\end{aligned}$$

where,

$$F_j = \begin{bmatrix} I_n & X_j \end{bmatrix}, \quad \mu_j = \begin{bmatrix} \alpha_j \\ \beta_j \end{bmatrix}, \quad V_j = \sigma_\varepsilon^2 I_n, \quad G_j = I_{n+p}, \quad W_j = \sigma_\eta^2 \begin{bmatrix} diag(\delta_j) & 0 \\ 0 & 0 \end{bmatrix}$$

The Kalman Filter and Kalman Smoother along with the variance optimization as described above may be calculated to get an estimate of $\mu_{j|J} = [\alpha'_{j|J}\ \beta'_{j|J}]'$. Recall, we assume that β_j remains constant over time, so all $\beta_{j|J}$ for $j \in \{1, 2, ..., J\}$ are equal. Our estimate for the linear effect of our predictors is $\hat{\beta} = \beta_{J|J}$ which has variance $P_{\hat{\beta}} = [P_{J|J}]_{(n+1:n+p),(n+1:n+p)}$. If modeling assumptions are met, $\hat{\beta} \sim N(B, P_{\hat{\beta}})$ which can be used for hypothesis testing [@linFilt, @durbin_koopman_2012].

Computational Considerations

Recall, state vector initial parameters are unknown, putting a diffuse prior on the variances will quickly converge to the same variances as if we correctly specified the initial conditions. To estimate our proposed model we set a diffuse prior on the vector $\mu_0 = [\alpha_0{}'\ \beta_0']'$. Each subject shares β, which is now assumed random as we let $\beta_0 \sim N(0, \infty)$, therefore the observations y_j are no longer treated as independent in the Kalman Filter and Kalman Smoother. This leads to the inverse of a possibly large non-sparse matrix $\text{Var}(Y_j|y_{1:j-1}) = S \in R^{n\times n}$ in the Kalman Filter process for $j \in \{1, 2, ..., J\}$. The S_j is also needed in the calculation of the log-likelihood of the variance parameters σ_ε^2 and σ_η^2, which makes it difficult to avoid the inversion at each iteration. This computational burden is then amplified as the Kalman Filter needs to be run multiple iterations for the maximum likelihood estimation calculation of σ_ε^2 and σ_η^2 to converge. In addition to considerable computation time, inverting a large matrices S_t can also lead to numerical inaccuracies. To overcome the issues of the full likelihood estimation process, we propose either partitioning or a Bayesian's Gibb's Sampling approach.

Partitioning

To decrease the computational burden, we propose randomly and equally partitioning the n subjects into k groups then run the Kalman Filter and Smoother on each group independently. This will result in $\widehat{\beta}^{(i)}$ and $P^{(i)}_{\widehat{\beta}}$ for $i \in \{1, 2, ..., k\}$ independent groups. We then use $\bar{\beta} = k^{-1} \sum_{i=1}^{k} \widehat{\beta}^{(i)}$ as our estimate for B. If modeling assumptions are met, $\bar{\beta} \sim N(B, k^{-2} \sum_{i=1}^{k} P_{\widehat{\beta}^{(i)}})$ which can be used for hypothesis testing.

The partitioning method enforces a linear increase in the Kalman Filter computation time as n increases.

Bayesian Gibb's Sampling

Computational issues arise in the full likelihood estimation because β is not treated as fixed in the Kalman Filter. This issue can be avoided by treating β as fixed when estimating α, then estimating β outside of the Kalman Filter using a Bayesian Gibb's Sampler. By following this methodology, the elements of y_j will be treated as independent in the Kalman Filter and Kalman Smoother process. Using the Bayesian Gibb's Sampling framework, under regularity conditions, a sample from the joint distribution of the unknown parameters can be created. Inference can then be made using the samples of the respective parameters.

The Model

In this representation we no longer index the linear effect β with a time component.

$$\begin{aligned} y_{ij} &= \alpha_{ij} + x_{ij}\beta + \varepsilon_{ij}, \quad \varepsilon_{ij} \sim N(0, \sigma^2_\varepsilon) \\ \alpha_{ij} &= \alpha_{i(j-1)} + \eta_{ij}, \quad \eta_{ij} \sim N(0, \delta_{ij}\sigma^2_\eta) \end{aligned}$$

We also set the following prior distributions for the unknown parameters,

$$\begin{aligned} \alpha_0 &\sim N(u, P_0) \\ \beta &\sim N(\theta, I_p \sigma^2_\beta) \\ \sigma^2_\eta &\sim IG(a_0/2, b_0/2) \\ \sigma^2_\varepsilon &\sim IG(c_0/2, d_0/2) \end{aligned}$$

Posterior Distributions

The conditional distribution $\alpha_{1:n}|y_{1:n}, X_{1:n}, \beta, \sigma^2_\eta, \sigma^2_\varepsilon$ can be estimated directly from the the Kalman Filter. By conditioning on x_{ij} and β, each y_{ij} is independent for $i \in \{1, 2, ..., n\}$ and fixed j. Thus, we can run the Kalman Filter with $\tilde{y}_{ij} = y_{ij} - x_{ij}\beta$ as the outcome in the model,

$$\begin{aligned} \tilde{y}_{ij} &= \alpha_{ij} + \varepsilon_{ij} \\ \alpha_{ij} &= \alpha_{ij-1} + \eta_{ij} \end{aligned}$$

Because each element of $\tilde{y}_j$ are treated as independent, it is equivalent to estimating each element α_j independently for each subject, resulting in a computational efficient Kalman Filter where,

$$\begin{aligned} \alpha_{j|j-1} &= \alpha_{j-1|j-1}, \quad P_{j|j-1} = P_{j-1|j-1} + \sigma^2_\eta \\ \alpha_{j|j} &= \alpha_{j|j-1} + K_j(\tilde{y}_j - \alpha^{j-1}_j), \quad P_{j|j} = (1 - K_j)P_{j|j-1} \\ K_j &= \frac{P_{j|j-1}}{P_{j|j-1}} + \sigma^2_\varepsilon, \quad L_{j-1}(\alpha_{j|J} - \alpha_{j|j-1}) \end{aligned}$$

Although we are using vectors, because of the independence each operation is done element-wise. Let $\psi = \{\sigma_\eta^2, \sigma_\varepsilon^2, \beta\}$, the vector of the other unknown parameters. For the posterior of α we seek to find the distribution defined by,

$$\begin{aligned}
P_\psi(\alpha_{0:J}|y_{1:J}) &= P_\psi(\alpha_J|y_{1:J})P_\psi(\alpha_{j-1}|\alpha_J, y_{1:J})...P_\psi(\alpha_0|\alpha_{1:J}, y_{1:J}.) \\
&= P_\psi(\alpha_J|y_{1:J})P_\psi(\alpha_{j-1}|\alpha_n, y_{1:(J-1)})...P_\psi(\alpha_0|\alpha_1)
\end{aligned}$$

Therefore we need the following densities for j in $1, 2, ..., J-1$:

$$P_\psi(\alpha_j|\alpha_{j+1}, y_{1:j}) \propto P_\psi(\alpha_j|y_{1:j})P_\psi(\alpha_{j+1}|\alpha_j)$$

From the Kalman Filter we calculated $\alpha_j|y_{1:j} \sim N_\psi(\alpha_{j|j}, P_{j|j})$ and $\alpha_{j+1}|\alpha_j \sim N_\psi(\alpha_j, \sigma_\eta^2)$. After combining the two densities $m_j = E_\psi(\alpha_j|\alpha_{j+1}, y_{1:j}) = \alpha_{j|j} + L_j(\alpha_{j+1} - \alpha_{j+1|j})$ and $R_j = \text{Var}_\psi(\alpha_j|\alpha_{j+1}, y_{1:j}) = P_{j|j} - L_j^2 P_{j+1|j}$ [@shumway_stoffer_2017]. Because of normality, the posterior distribution for α_j is $N(m_j, R_j)$.

For the backward sampling procedure we start by sampling a α_J^* from a $N_\psi(m_J, R_J)$, then setting $\alpha_J^* = \alpha_J$ for the calculation of m_{T-1} to then sample α_{T-1}^* from a $N_\psi(m_{T-1}, R_{T-1})$. This process continues until a whole chain $\alpha_{0:T}^*$ has been sampled.

The posterior distribution β follows a pattern more similar to multivariate Bayesian regression. With all the other parameters fixed,

$$P(\beta|Y, \alpha, \sigma_\eta^2, \sigma_\varepsilon^2) = \frac{P(Y, \alpha, \sigma_\eta^2, \sigma_\varepsilon^2|\beta)P(\beta)}{P(Y, \alpha, \sigma_\eta^2, \sigma_\varepsilon^2)}$$

Because $P(Y, \alpha, \sigma_\eta^2, \sigma_\varepsilon^2)$ is constant with respect to β and $P(\beta)$ is already defined we focus our attention on $P(Y, \alpha, \sigma_\eta^2, \sigma_\varepsilon^2|\beta)$.

$$\begin{aligned}
P(Y, \alpha, \sigma_\eta^2, \sigma_\varepsilon^2|\beta) =& P(y_1, ..., y_J, \alpha_1, ..., \alpha_J, \sigma_\eta^2, \sigma_\varepsilon^2|\beta) \\
=& P(y_J|y_1, ..., y_{T-1}, \alpha_1, ..., \alpha_J, \sigma_\eta^2, \sigma_\varepsilon^2\beta) \\
& \times P(y_1, ..., y_{T-1}, \alpha_1, ..., \alpha_J, \sigma_\eta^2, \sigma_\varepsilon^2|\beta) \\
=& P(y_J|\alpha_J, \sigma_\varepsilon^2\beta)P(\alpha_{T-1}|y_1, ..., y_{T-1}, \alpha_1, ..., \alpha_{T-1}, \sigma_\eta^2, \sigma_\varepsilon^2, \beta) \\
& \times P(y_1, ..., y_{T-1}, \alpha_1, ..., \alpha_{T-1}, \sigma_\eta^2, \sigma_\varepsilon^2|\beta) \\
=& P(y_J|\alpha_J, \sigma_\varepsilon^2\beta)P(\alpha_{T-1}|\alpha_{T-1}, \sigma_\eta^2) \\
& \times P(y_1, ..., y_{T-1}, \alpha_1, ..., \alpha_{T-1}, \sigma_\eta^2, \sigma_\varepsilon^2|\beta) \\
=& P(\sigma_\varepsilon^2)P(\sigma_\eta^2)\Big(\prod_{k=0}^{T} P(\alpha_k|\alpha_{k-1}, \sigma_\eta^2)\Big)\prod_{j=1}^{T} P(y_j|\alpha_j, \sigma_\varepsilon^2, \beta) \\
\propto& \prod_{j=1}^{T} P(y_j|\alpha_j, \sigma_\varepsilon^2, \beta)
\end{aligned}$$

For simplicity we can further write,

$$\begin{aligned}
-2logP(Y, \alpha, \sigma_\eta^2, \sigma_\varepsilon^2|\beta) &\propto \sum_{j=1}^{T} \frac{(y_j - \alpha_j - X_j\beta)'(y_j - \alpha_j - X_j\beta)}{\sigma_\varepsilon^2} \\
&\propto \sum_{j=1}^{T} \frac{-2y_j'X_j\beta + 2\alpha_j'X_j\beta + \beta'X_j'X_j\beta}{\sigma_\varepsilon^2} \\
&\propto \frac{\beta'\big(\sum_{j=1}^{T} X_j'X_j\big)\beta - 2\big(\sum_{j=1}^{T} y_j - \alpha_j\big)'X_j\beta}{\sigma_\varepsilon^2}
\end{aligned}$$

We can then find the proportionality of the prior of β,

$$\begin{aligned}-2logP(\beta) \propto &\frac{(\beta-\theta)'(\beta-\theta)}{\sigma^2_\beta}\\ \propto &\frac{\beta'\beta-2\theta\beta}{\sigma^2_\beta}\end{aligned}$$

Thus the -2log posterior of β is proportional to,

$$\begin{aligned}-2logP(\beta|Y,\alpha,\sigma^2_\eta,\sigma^2_\varepsilon) \propto &\frac{\beta'(\sum_{j=1}^T X_j'X_j)\beta - 2(\sum_{j=1}^T y_j-\alpha_j)'X_j\beta}{\sigma^2_\varepsilon} + \frac{\beta'\beta-2\theta\beta}{\sigma^2_\beta}\\ \propto &\frac{\beta'(\sigma^2_\beta\sum_{j=1}^T X_j'X_j)\beta - 2\sigma^2_\beta(\sum_{j=1}^T y_j-\alpha_j)'X_j\beta + \beta'\sigma^2_\varepsilon I_p\beta - 2\sigma^2_\varepsilon\theta\beta}{\sigma^2_\varepsilon\sigma^2_\beta}\\ \propto &\frac{\beta'(\sigma^2_\beta\sum_{j=1}^T X_j'X_j)\beta + \beta'\sigma^2_\varepsilon I_p\beta - 2\sigma^2_\beta(\sum_{j=1}^T y_j-\alpha_j)'X_j\beta - 2\sigma^2_\varepsilon\theta\beta}{\sigma^2_\varepsilon\sigma^2_\beta}\\ \propto &\frac{\beta'\Bigg((\sigma^2_\beta\sum_{j=1}^T X_j'X_j)+\sigma^2_\varepsilon I_p\Bigg)\beta - 2\Bigg(\sigma^2_\beta(\sum_{j=1}^T y_j-\alpha_j)'X_j - \sigma^2_\varepsilon\theta'\Bigg)\beta}{\sigma^2_\varepsilon\sigma^2_\beta}\\ \propto &\frac{(\beta-\Sigma^{-1}B)'\Sigma(\beta-\Sigma^{-1}B)}{\sigma^2_\varepsilon\sigma^2_\beta}\end{aligned}$$

Where $B = \sigma^2_\beta(\sum_{j=1}^T y_j-\alpha_j)'X_j - \sigma^2_\varepsilon\theta'$ and $\Sigma = (\sigma^2_\beta\sum_{j=1}^T X_j'X_j)+\sigma^2_\varepsilon I_p$. Therefore, $\beta|Y,\alpha,\sigma^2_\eta,\sigma^2_\varepsilon \sim N(\Sigma^{-1}B, \sigma^2_\varepsilon\sigma^2_\beta\Sigma^{-1})$.

Note, Σ is a $p\times p$ matrix that needs to be inverted. If p is large, this can greatly slow down the Gibb's Sampler, especially when considering we may do several thousand iterations. However, Σ can be broken down to increase computation speed. Recall, $\Sigma = ((\sigma^2_\beta\sum_{j=1}^T X_j'X_j)+\sigma^2_\varepsilon I_p)$. The term $(\sigma^2_\beta\sum_{j=1}^T X_j'X_j)$ will not change at each iteration because it does not contain unknown parameters. By calculating the eigenvalue decomposition on $\sigma^2_\beta\sum_{j=1}^T X_j'X_j$ we can rewrite Σ as follows,

$$\begin{aligned}((\sigma^2_\beta\sum_{j=1}^T X_j'X_j)+\sigma^2_\varepsilon I) &= (Q\Lambda Q' + \sigma^2_\varepsilon I)\\ &= (Q\Lambda Q' + \sigma^2_\varepsilon QQ')\\ &= Q(\Lambda+\sigma^2_\varepsilon I)Q'\end{aligned}$$

then,

$$((\sigma^2_\beta\sum_{j=1}^T X_j'X_j)+\sigma^2_\varepsilon I)^{-1} = Q(1/(\Lambda+\sigma^2_\varepsilon I))Q'$$

We only need to calculate the eigen vectors Q and eigen values Λ of $\sigma^2_\beta\sum_{j=1}^T X_j'X_j$ once, then simply update σ^2_ε before calculating the inverse.

For the variance paramter σ^2_η, we apply the same Bayes' rule rationale,

$$
\begin{aligned}
P(\sigma_\eta^2|Y,\beta,\alpha,\sigma_\varepsilon^2) =& \frac{P(Y,\beta,\alpha,\sigma_\varepsilon^2|\sigma_\eta^2)P(\sigma_\eta^2)}{P(Y,\beta,\alpha,\sigma_\varepsilon^2)}\\
\propto& P(Y,\beta,\alpha,\sigma_\varepsilon^2|\sigma_\eta^2)P(\sigma_\eta^2)\\
\propto& (\delta_j\sigma_\eta^2)^{-nT/2}e^{\sum_{j=1}^T(\alpha_j-\alpha_{j-1})^2/2\delta_j\sigma_\eta^2}(\sigma_\eta^2)^{-a_0/2-1}e^{-b_0/2\sigma_\eta^2}\\
\propto& (\sigma_\eta^2)^{-(nT+a_0)/2-1}e^{(\sum_{j=1}^T(\alpha_j-\alpha_{j-1})^2/\delta_j+b_0)/2\sigma_\eta^2}
\end{aligned}
$$

Therefore, $\sigma_\eta^2|Y,\beta,\alpha,\sigma_\varepsilon^2 \sim IG(\frac{nT+a_0}{2},\frac{\sum_{j=1}^T(\alpha_j-\alpha_{j-1})^2/\delta_j+b_0}{2})$.

In a very similar fashion we can show $\sigma_\varepsilon^2|Y,\beta,\alpha,\sigma_\eta^2 \sim IG(\frac{nT+c_0}{2},\frac{d_0+\sum_{j=1}^T(y_j-X_j\beta-\alpha_j)^2}{2})$.

Algorithm 2 Gibb's Sampling Algorithm

```
1. Select prior parameters $\theta,\sigma_\beta^2,a_0,b_0,c_0,d_0$.
2. Let $\beta^{(0)}=\theta$, $\sigma_\eta^{2(0)}=\frac{d_0/2}{1+c_0/2}$, and $\sigma_\varepsilon^{2(0)}=\frac{b_0/2}{1+a_0/2}$.
3. Repeat steps 4-7 for $i\in\{1,2,...,M\}$
4. Run a forward-filtering backward sampling procedure, conditioning on $\beta=\beta^{i-1},\sigma_\eta^2=\sigma_\eta^{2(i-1)},\sigma_\varepsilon^2=\sigma_\varepsilon^{2(i-1)}$ to get samples $\alpha^*$, then set $\alpha^{(i)}=\alpha^*$.
5. Sample $\sigma_\eta^{2*}$ from $IG(\frac{nT+a_0}{2},\frac{\sum_{j=1}^T(\alpha_j^{(i)}-\alpha_{j-1}^{(i)})^2+b_0}{2})$ and set $\sigma_\eta^{2(i)}=\sigma_\eta^{2*}$.
6. Sample $\sigma_\varepsilon^{2*}$ from $IG(\frac{nT+c_0}{2},\frac{d_0+\sum_{j=1}^T(y_j-X_j\beta^{(i-1)}-\alpha_j^{(i)})^2}{2})$ and set $\sigma_\varepsilon^{2(i)}=\sigma_\varepsilon^{2*}$.
7. Sample $\beta^*$ from $N(\Sigma^{-1}B,\sigma_\varepsilon^2\sigma_\beta^2\Sigma^{-1})$ where $\alpha=\alpha^{(i)},\sigma_\eta^2=\sigma_\eta^{2(i)},\sigma_\varepsilon^2=\sigma_\varepsilon^{2(i)}$ and set $\beta^{(i)}=\beta^*$.
```

After steps 1-7 are completed posterior samples can be used as an empirical distribution to make parameter inference.