Approximate Message Passing for Mixed Generalised Linear Models

Philip Sosnin
Department of Engineering
University of Cambridge

30th May 2023

Supervisor: Prof. Ramji Venkataramanan

Approximate Message Passing for Mixed Generalised Linear Models

Philip Sosnin, St. John's College

Technical Abstract

Regression analysis is a fundamental statistical method employed across various disciplines, including economics, engineering, and natural sciences. Common regression models, such as linear, logistic and Poisson regressions, are essential tools in understanding and interpreting the relationships between variables. However, these simple models often fail to capture the complexity of data sampled from *heterogeneous* populations, where multiple unobserved classes contribute to the observed data. Failing to account for the presence of multiple signals within the data can lead to inaccurate conclusions and hinder the understanding of the underlying relationships between variables.

To address the limitations of simple regression models, finite mixture models have been introduced to effectively explain complex data patterns by incorporating latent classes. However, inference in mixed models is challenging, particularly in high-dimensional scenarios and when incorporating prior information. Existing methods, such as Alternating-Minimisation (AM) [11] and Expectation-Maximisation (EM) [3], are often unable to provide satisfactory parameter estimates in these situations.

This project investigates the application of approximate message passing (AMP) to estimation in mixed models. AMP is a class of iterative algorithms that have been applied to a wide range of estimation and learning tasks [4]. AMP algorithms offer the advantages of incorporating prior information, computational efficiency, and precise asymptotic characterisation. Previous work has applied the AMP algorithm to mixtures of linear regressions and mixtures of max-affine regressions [10]. This project builds upon this work by extending the AMP algorithm to encompass the following regression models:

- Rectified linear regression.
- Mixtures of rectified linear regressions.
- Logistic regression.
- Mixtures of logistic regressions.

For each of these models, a Bayes-optimal AMP algorithm is derived, which requires a computable characterisation of the AMP denoising functions. In the case of mixed and non-mixed rectified linear regressions, an exact expression for the denoising functions was obtained. For the logistic models, suitable approximations were employed to obtain an approximate Bayes-optimal AMP recursion.

The AMP algorithms were implemented in python[1] and shown to outperform existing estimators in the high-dimensional regime. Numerical results demonstrate that AMP shows strong recovery of signals in mixed models by exploiting dependencies between the signals. Additionally, GAMP exhibits robustness to noise, fast convergence, and good computational efficiency, making it an attractive option for signal estimation in heterogeneous populations.

[1] `https://github.com/psosnin/amp-for-mixed-generalised-linear-models`

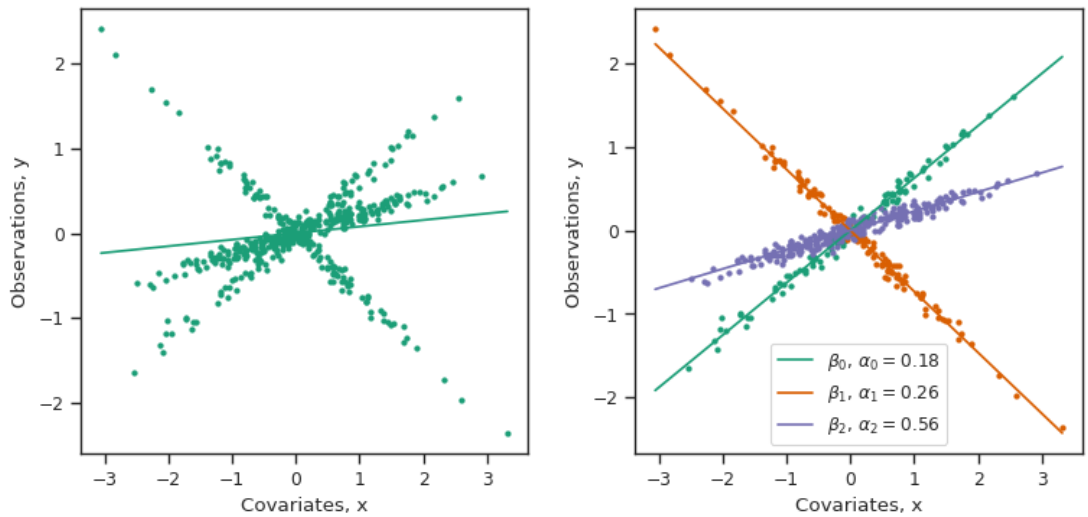
Contents

1 Introduction . . . 3
2 Background and Prior Work . . . 4
  2.1 Generalised Linear Models . . . 4
  2.2 Generalised Approximate Message Passing Algorithm . . . 5
  2.3 Matrix-Valued Generalised Linear Models . . . 6
  2.4 Expectation-Maximisation Algorithm . . . 7
  2.5 Alternating-Minimisation Algorithm . . . 8
  2.6 GAMP for Matrix GLMs . . . 8
3 Rectified Linear Regression . . . 11
  3.1 GAMP for Rectified Linear Regression . . . 11
  3.2 Comparison of Algorithms for Rectified Linear Regression . . . 13
4 Mixed Rectified Linear Regression . . . 14
  4.1 Expectation-Maximisation for Mixed Rectified Linear Regression . . . 14
  4.2 Alternating-Minimisation for Mixed Rectified Linear Regression . . . 15
  4.3 Gradient Descent for Mixed Rectified Linear Regression . . . 15
  4.4 GAMP for Mixed Rectified Linear Regression . . . 15
  4.5 Comparison of Algorithms for Mixed Rectified Linear Regression . . . 19
5 Logistic Regression . . . 20
  5.1 GAMP for Logistic Regression . . . 21
  5.2 Comparison of Algorithms for Logistic Regression . . . 22
6 Mixed Logistic Regression . . . 22
  6.1 Expectation-Maximisation for Mixed Logistic Regression . . . 23
  6.2 Alternating-Minimisation for Mixed Logistic Regression . . . 23
  6.3 GAMP for Mixed Logistic Regression . . . 24
  6.4 Comparison of Algorithms for Mixed Logistic Regression . . . 27
  6.5 Comparison of Classification Performance . . . 27
7 Conclusions and Future Work . . . 28
References . . . 29
Appendix A: Derivation of GAMP for Logistic Regression . . . 30
Appendix B: Derivation of GAMP for Mixed Logistic Regression . . . 31
Appendix C: Derivation of GAMP for Rectified Linear Regression . . . 33
Appendix D: Derivation of GAMP for Mixed Rectified Linear Regression . . . 36

1 Introduction

Regression analysis is a fundamental statistical method employed across various disciplines, including economics, engineering, and natural sciences. The generalised linear model (GLM) serves as a comprehensive framework for regression models, encompassing a wide range of well-known models such as linear, logistic and Poisson regressions. These models have become essential tools in understanding and interpreting the relationships between variables in various domains.

Despite their widespread use, simple regression models fall short in explaining data sampled from *heterogeneous* populations, where multiple latent classes contribute to the observed data. To illustrate this, consider the dataset in Figure 1a, which was generated from a population containing three latent classes. Each class contains its own regression vector but the observations are unlabelled. A single linear regression fails to capture the complexity, and even non-linear regression models struggle due to the multi-modal nature of the data. The inability to account for the presence of multiple signals within the data can lead to inaccurate conclusions and hinder the understanding of the underlying relationships between variables.



(a) Standard linear regression fit (b) Mixtures of linear regressions fit

Fig. 1: Dataset generated from a mixed linear regression model with $L = 3$ components

Heterogeneous data can arise in a variety of applications, including medicine, economics and social sciences. In many cases, there are multiple signals present in data, which cannot be explained by a simple regression model. To address this limitation, finite mixture models have been introduced to account for this heterogeneity. In a finite mixture model, each observation is generated from one of L signals according to a generalised linear model. By incorporating the notion of latent classes, finite mixture models provide a more effective means of explaining complex data patterns. For example, in a mixed linear regression model, each observation is generated from one of L regression vectors at random. Figure 1b shows that such a fit can be much more effective at explaining data than a standard regression model.

Existing methods for inference in mixed models include *Alternating-Minimisation (AM)* [11] and *Expectation-Maximisation (EM)* [3]. While AM and EM have been widely

utilized for inference in mixed models, their performance suffers in high-dimensional scenarios. Moreover, incorporating prior information into these methods is challenging, limiting their applicability. Tailoring the prior over the parameters is important in many contexts, such as enforcing a sparse signal for compressed sensing [7].

This project investigates the application of *approximate message passing (AMP)* to estimation in mixed models. AMP is a class of iterative algorithms that can be applied to a wide range of estimation and learning tasks. AMP algorithms have several properties that make them attractive. Firstly, prior information about the quantity of interest can easily be incorporated into the algorithm, allowing locally Bayes-optimal inference in a variety of settings, such as under regularisation or sparsity constraints [4]. Additionally, the iterative updates of AMP are computationally efficient and low complexity, allowing them to scale well to high dimensional problems. Finally, under certain conditions on the design matrix, it can be shown that the distribution of the AMP iterates have a precise asymptotic characterisation [1].

Recent advancements have introduced an AMP variant designed for mixed models, specifically addressing mixtures of linear regressions and mixtures of max affine regressions [10]. This project builds upon prior work by extending the application of the AMP algorithm to include logistic and rectified regression models in both mixed and non-mixed cases. In Section 2, we begin by introducing the standard AMP algorithm for generalised linear models. We then describe how mixed models can be represented by a matrix form of generalised linear model. The EM, AM and AMP algorithms for the matrix GLM are presented, using the mixed linear regression model as an example. Sections 3-6 contain the main contributions of this project, which are

- Deriving the Bayes-optimal AMP algorithm for the rectified and logistic regression models, in both the standard and mixed settings.
- Comparing the performance of AMP with existing estimation algorithms such as EM and AM.

Numerical results show that AMP is able to significantly outperform existing estimators for the mixed rectified and logistic regression models.

2 Background and Prior Work

2.1 Generalised Linear Models

Generalised linear models extend standard linear models by introducing a (possibly non-linear) link function. Consider a design matrix $X = (x_1, \dots, x_n)^T \in \mathbb{R}^{n \times p}$ and observations $y = (y_1, \dots, y_n) \in \mathbb{R}^n$ satisfying

$$y_i = q\left(x_i^\top \beta, \epsilon_i\right) \text{ for } i = 1, \dots, n, \tag{1}$$

where the signal $\beta \in \mathbb{R}^p$ is the target of inference, $\epsilon_i \sim \pi_\epsilon$ are independent noise variables and $q : \mathbb{R}^2 \to \mathbb{R}$ is the known link function.

The GLM framework is flexible and can be used to express a wide range of regression models by choosing the link function q, such as

$$\begin{aligned}
&y_i = \beta^T x_i + \epsilon_i && \text{(linear regression)}\\
&y_i \in \{0, 1\} \text{ with probability } \sigma\left(\beta^T x_i\right) && \text{(logistic regression)}\\
&y_i = \max\left(0, \beta^T x_i\right) + \epsilon_i && \text{(rectified linear regression)}
\end{aligned}$$

where $\sigma(\cdot)$ is the logistic function. Note that in the case of logistic regression the noise is implicit in the probabilistic assignment of values to y_i.

In this project, we consider a high dimensional Gaussian design setting where the covariates $x_1, \ldots, x_n \overset{\text{iid}}{\sim} \mathcal{N}_p(0, I_p/n)$ and n, p tend to infinity with fixed proportion $\delta = n/p$. Each component of the parameters β is distributed iid according to prior $\pi_{\bar{\beta}}$, which is assumed to be known.

2.2 Generalised Approximate Message Passing Algorithm

The generalised approximate message passing (GAMP) algorithm is an iterative algorithm designed for inference in GLMs. The algorithm produces iterative estimates of $\hat{\beta}^k, \theta^k$ of $\beta \in \mathbb{R}^p$ and $\theta := X\beta \in \mathbb{R}^n$ respectively [4].

Given initialisers $\hat{\beta}^0 \in \mathbb{R}^p$, $\hat{r}^{-1} := 0 \in \mathbb{R}^n$ and $b_0 \in \mathbb{R}$, GAMP recursively applies the following update steps:

$$\begin{aligned} \theta^k &:= X\hat{\beta}^k - b_k \hat{r}^{k-1}, & \hat{r}^k &:= g_k\left(\theta^k, y\right), & c_k &:= n^{-1} \textstyle\sum_{i=1}^n g_k'\left(\theta_i^k, y_i\right), \\ \beta^{k+1} &:= X^\top \hat{r}^k - c_k \hat{\beta}^k, & \hat{\beta}^{k+1} &:= f_{k+1}\left(\beta^{k+1}\right), & b_{k+1} &:= n^{-1} \textstyle\sum_{j=1}^p f_{k+1}'\left(\beta_j^{k+1}\right), \end{aligned} \tag{2}$$

for $k \in \mathbb{N}_0$. The functions $g_k : \mathbb{R}^2 \to \mathbb{R}$ and $f_{k+1} : \mathbb{R} \to \mathbb{R}$ act component-wise on their vector arguments. The derivatives $g_k' : \mathbb{R}^2 \to \mathbb{R}$, $f_{k+1}' : \mathbb{R} \to \mathbb{R}$ are the partial derivatives of g_k, f_{k+1} with respect to their first arguments.

The memory terms $b_k \hat{r}^{k-1}$ and $c_k \hat{\beta}^k$ in the update steps de-bias the iterates such that their distribution can be described by a "state evolution" in the high dimensional regime. Consequently, the functions f_k and g_k can be viewed as de-noising functions given the distribution of the iterates from the state evolution and prior.

2.2.1 State Evolution

The state evolution recursion provides an asymptotic characterisation of the distribution of the entries of the iterates [4]. In particular, each component of β^k is distributed according to $\mu_k \bar{\beta} + \sigma_k \xi$, where $\xi \sim \mathcal{N}(0,1)$ is independent Gaussian noise and $\bar{\beta} \in \mathbb{R}$ is a random variable distributed according to the component-wise distribution of the signal β. Therefore, it is clear that the estimation problem of $\hat{\beta}^{k+1} := f_{k+1}\left(\beta^{k+1}\right)$ reduces to estimation in Gaussian noise. Similarly, the distribution of the components of θ^k converges to the distribution of $\mu_{Z,k} Z + \sigma_{Z,k} \zeta$, where $\zeta \sim \mathcal{N}(0,1)$ and $Z \sim \mathcal{N}\left(0, \frac{1}{\delta}\mathbb{E}\left[\bar{\beta}^2\right]\right)$.

The state evolution defines the constants $\mu_k, \sigma_k, \mu_{Z,k}, \sigma_{Z,k}$ according to a deterministic recursive sequence. Given an initialisation $\Sigma^0 \in \mathbb{R}^{2\times 2}$, the state evolution parameters are defined as

$$\begin{aligned} \mu_{k+1} &= \mathbb{E}\left(\partial_z g_k\left(Z^k, \bar{Y}\right)\right) & \sigma_{k+1}^2 &= \mathbb{E}\left(g_k\left(Z^k, \bar{Y}\right)^2\right) \\ \mu_{Z,k} &= \tfrac{\Sigma_{21}^k}{\Sigma_{11}^k}, & \sigma_{Z,k}^2 &= \Sigma_{22}^k - \tfrac{\left(\Sigma_{12}^k\right)^2}{\Sigma_{11}^k} \\ \Sigma^{k+1} &= \tfrac{1}{\delta}\begin{pmatrix} \mathbb{E}\left[\bar{\beta}^2\right] & \mathbb{E}\left[\bar{\beta} f_{k+1}\left(\mu_{k+1}\bar{\beta} + \sigma_{k+1}\xi\right)\right] \\ \mathbb{E}\left[\bar{\beta} f_{k+1}\left(\mu_{k+1}\bar{\beta} + \sigma_{k+1}\xi\right)\right] & \mathbb{E}\left[f_{k+1}\left(\mu_{k+1}\bar{\beta} + \sigma_{k+1}\xi\right)^2\right] \end{pmatrix} \end{aligned} \tag{3}$$

where the random variables $(Z, Z^k) \sim \mathcal{N}\left(0, \Sigma^k\right)$, $\bar{\epsilon} \sim \pi_\epsilon$ and $\overline{Y} = q\left(Z, \bar{\epsilon}\right)$.

2.2.2 Choosing the functions f_k and g_k

Given the state evolution as described above, it is possible to derive Bayes-optimal settings of the functions f_k and g_k. In particular, the function f_k becomes the MMSE estimator in Gaussian noise, with the exact form depending on the prior π_β. Likewise, the de-noising function g_k depends on the form the noise π_ϵ and link function q.

$$f_k^*\left(\mu_k\bar{\beta}+\sigma_k\xi\right)=\mathbb{E}\left(\bar{\beta}\mid\mu_k\bar{\beta}+\sigma_k\xi\right) \tag{4}$$

$$g_k^*\left(Z^k,\bar{Y}\right)=\frac{\mathbb{E}\left(Z\mid Z^k,\bar{Y}\right)-\mathbb{E}\left(Z\mid Z^k\right)}{\mathrm{Var}\left(Z\mid Z^k\right)} \tag{5}$$

Note that under a Bayes-optimal setting of the functions g_k and f_k, the state evolution in equation (3) simplifies to

$$\mu_{k+1}=\sigma_{k+1}^2=\mathbb{E}\left[g_k^*(Z^k,\bar{Y})^2\right] \tag{6}$$

2.2.3 Performance measures

It has been shown that under suitable conditions the following holds with respect to any function ψ [4]

$$\frac{1}{p}\sum_{j=1}^{p}\psi\left(\hat{\beta}_j^k,\beta_j\right)\to\mathbb{E}\left\{\psi\left(f_k\left(\mu_k\bar{\beta}+\sigma_k\xi\right),\bar{\beta}\right)\right\} \tag{7}$$

Thus, by setting ψ to be any loss function, the state evolution gives a prediction of the loss at every iteration of the GAMP algorithm. For example, the normalised squared correlation of the GAMP signal estimate after k iterations converges to

$$\frac{\left\langle\hat{\beta}^k,\beta\right\rangle^2}{\|\hat{\beta}^k\|_2^2\|\beta\|_2^2}\to\frac{\left(\mathbb{E}\left[f_k\left(\mu_k\bar{\beta}+\sigma_k\xi\right)\bar{\beta}\right]\right)^2}{\mathbb{E}\left[f_{k,\ell}\left(\mu_k\bar{\beta}+\sigma_k\xi\right)^2\right]\mathbb{E}\left[\bar{\beta}^2\right]} \tag{8}$$

The form of GAMP described above applies to standard (non-mixed) GLMs. To extend the algorithm to perform inference in finite mixture models, we first introduce the matrix GLM.

2.3 Matrix-Valued Generalised Linear Models

Finite mixture models can be described using a matrix form of generalised linear model. The signal of each of the L components of the mixture form the columns of a signal matrix $B=(\beta_1,\dots\beta_L)\in\mathbb{R}^{p\times L}$. As before, the mapping from the design matrix $X\in\mathbb{R}^{n\times p}$ to observations $y\in\mathbb{R}^n$ depends on the link function:

$$y_i=q\left(B^Tx_i,\Psi_i\right),\quad i=1,\dots,n \tag{9}$$

where x_i denotes the i-th feature vector and Ψ_i is a vector of unobserved auxiliary variables, which include any latent or noise variables. The matrix B is distributed row-wise according to prior $\pi_{\bar{B}}$.

In the case of a mixture model, Ψ_i contains a categorical latent variable that describes which of the L components generated the observation. For example, a mixed linear regression model can be described using the following link function

$$y_i = \sum_{j=1}^{L} \mathbb{I}\left(c_i = j\right) \left[B^T x_i\right]_j + \epsilon_i, \quad c_i \sim \text{cat}(\alpha), \quad \epsilon_i \sim \mathcal{N}(0, \sigma_\epsilon^2). \tag{10}$$

Note that the maximum-likelihood solution for such a model is given by the least squares estimator

$$\widehat{B} = \underset{\substack{B \in \mathbb{R}^{p \times L} \\ c \in \{1,\ldots,L\}^n}}{\text{argmin}} \sum_{i=1}^{n} \left(y_i - \mathbb{I}\left(c_i = j\right) \left[B^T x_i\right]_j \right)^2 \tag{11}$$

which in general can be non-convex and NP-hard to compute [11]. Therefore, locally optimal algorithms such as AM and EM are often used for estimation in mixed models. More recently, a variant of AMP has been developed for use with matrix-valued signals [10], which will be described in Section 2.6.

2.4 Expectation-Maximisation Algorithm

The Expectation-Maximisation (EM) algorithm is an algorithm for approximate maximum-likelihood fitting in latent variable models. Let c be the vector of latent variables in a matrix GLM, i.e. $y_i = f(\beta_j^T x_i, \epsilon_i)$ if $c_i = j$. Then define the following lower bound on the log-likelihood, known as the free energy

$$\mathcal{F}(h(c), \theta) = \sum_c h(c) \log \frac{p(y \mid c, X, \theta) p(c \mid \theta)}{h(c)} \leq \log p(y \mid X, \theta) \tag{12}$$

where $h(c)$ is an arbitrary probability distribution over the latent variables and θ are the parameters of interest. This includes the signal matrix B and any component-specific parameters, e.g. mixing proportions or component-specific noise variances. The EM algorithm is a two-step process:

1. E step: for fixed θ_{t-1}, maximize lower bound $\mathcal{F}\left(h(c), \theta_{t-1}\right)$ wrt $h(c)$. This is equivalent to setting $h(c)$ to be the the current posterior estimate given θ_{t-1}

$$h_t(c) = p\left(c \mid y, X, \theta_{t-1}\right)$$

2. M step: for fixed $h_t(c)$ maximize the lower bound $\mathcal{F}\left(h_t(c), \theta\right)$ wrt θ

$$\theta_t = \underset{\theta}{\text{argmax}} \sum_c h_t(c) \log \left(p(y \mid c, X, \theta) p(c \mid \theta)\right)$$

In many models, the M-step of the EM algorithm is equivalent to fitting a weighted regression model for each component. For example, the EM algorithm for the mixtures

of linear regression model in equation (10) is

$$h_t(c_i = j) = \frac{\hat{\alpha}_j^{(t)} \mathcal{N}\left(y_i; x_i^T \hat{\beta}_j^{(t)}, \left(\hat{\sigma}^{(t)}\right)^2\right)}{\sum_{k=1}^{L} \hat{\alpha}_k^{(t)} \mathcal{N}\left(y_i; x_i^T \hat{\beta}_k^{(t)}, \left(\hat{\sigma}^{(t)}\right)^2\right)} \qquad \text{(E-step)} \qquad (13)$$

$$\hat{\beta}_j^{(t+1)} = \left(X^{\mathrm{T}} W_j^{(t)} X\right)^{-1} X^T W_j^{(t)} y \qquad \text{(M-step)} \qquad (14)$$

$$\left(\hat{\sigma}^{(t+1)}\right)^2 = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{L} h_t(c_i = j) \left(y_i - x_i^T \hat{\beta}_j^{(t)}\right)^2 \qquad \text{(M-step)} \qquad (15)$$

$$\hat{\alpha}_j^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} h_t(c_i = j) \qquad \text{(M-step)} \qquad (16)$$

where $W_j^{(t)} \in \mathbb{R}^{n \times n}$ is a diagonal matrix with entries $h_t(c_i = j)$ and the M-step for each signal β_j is a weighted least squares estimate. The EM algorithm for the mixed logistic and rectified regression models follow a similar structure, which will be described in later sections.

2.5 Alternating-Minimisation Algorithm

Alternating Minimisation (AM) is a simple method for inference in mixed models. The algorithm iterates between the following two steps:

1. Given an estimate $\hat{B}^{(t)} = (\hat{\beta}_1, \ldots, \hat{\beta}_L)$ of B, split the samples x_i, y_i into partitions $P_j^{(t)}$ based on which mixture component they are closest to

$$P_j^{(t)} := \left\{ i \in \{1, \ldots, N\} : j = \operatorname*{argmin}_{j \in \{1, \ldots, L\}} \left(y_i - q^*(x_i^T \hat{B}^{(t)}, c_i = j)\right)^2 \right\} \qquad (17)$$

   where $q^*()$ is the GLM link function in the absence of noise.

2. For each component j, estimate the signal β_j using only the data in partition P_j and a simple non-mixed fitting algorithm (e.g. least squares)

$$\hat{\beta}_j^{(t+1)} = \operatorname*{argmin}_{\beta_j} \sum_{i \in P_j^{(t)}} (y_i - q^*(x_i^T B, c_i = j))^2 \qquad (18)$$

This method can be viewed as a variant of Expectation-Maximisation that performs a hard assignment of samples to mixture components followed by simple fitting for each component.

2.6 GAMP for Matrix GLMs

The GAMP algorithm for matrix-valued iterates proceeds in much the same way as the standard GAMP algorithm described in Section 2.2. The algorithm produces estimates of iterates $\hat{B}^k \in \mathbb{R}^{p \times L}, \Theta^k \in \mathbb{R}^{n \times L}$ of $B, \Theta = XB$ [10]:

$$\begin{array}{ll}
\Theta^k = X\widehat{B}^k - \widehat{R}^{k-1}\left(F^k\right)^\top & B^k = X^\top \widehat{R}^{k-1} - \widehat{B}^{k-1}\left(C^{k-1}\right)^\top \\
C^k = \frac{1}{n}\sum_{i=1}^{n} g_k'\left(\Theta_i^k, y_i\right) & \widehat{R}^k = g_k\left(\Theta^k, y\right) \\
\widehat{B}^k = f_k\left(B^k\right) & F^{k+1} = \frac{1}{n}\sum_{j=1}^{p} f_{k+1}'\left(B_j^{k+1}\right)
\end{array}$$

for $k \in \mathbb{N}_0$. The functions $g_k : \mathbb{R}^L \times \mathbb{R} \to \mathbb{R}^L$ and $f_{k+1} : \mathbb{R}^L \to \mathbb{R}^L$ act row-wise on their matrix arguments. g_k' and f_{k+1}' denote the Jacobians of g_k, f_{k+1} with respect to their first arguments.

As before, the iterates follow a state evolution so that the empirical distribution of the rows of Θ^k and B^{k+1} converge to $\mathrm{M}_\Theta^k Z + G_\Theta^k \in \mathbb{R}^L$ and $\mathrm{M}_B^{k+1}\bar{B} + G_B^{k+1} \in \mathbb{R}^L$, respectively. The matrices $G_\Theta^k \sim \mathcal{N}\left(0, T_\Theta^k\right)$ and $G_B^k \sim \mathcal{N}\left(0, T_B^k\right)$ are independent Gaussian noise. The random vector $\bar{B} \in \mathbb{R}^L$ is distributed according to the row-wise prior on B and $Z \sim \mathcal{N}\left(0, \frac{1}{\delta}\mathbb{E}\left[\bar{B}\bar{B}^T\right]\right)$.

The state evolution now consists of random vectors $(Z, Z^k) \sim \mathcal{N}(0, \Sigma^k)$, where $\Sigma^k \in \mathbb{R}^{2L\times 2L}$ is given by

$$\Sigma^{k+1} = \frac{1}{\delta}\begin{bmatrix} \mathbb{E}\left[\bar{B}\bar{B}^T\right] & \mathbb{E}\left[\bar{B} f_{k+1}(\bar{B}^{k+1})^T\right] \\ \mathbb{E}\left[f_{k+1}(\bar{B}^{k+1})\bar{B}^T\right] & \mathbb{E}\left[f_{k+1}(\bar{B}^{k+1}) f_{k+1}(\bar{B}^{k+1})^T\right] \end{bmatrix} \tag{19}$$

where $\bar{B}^{k+1} = \mathrm{M}_B^{k+1}\bar{B} + G_B^{k+1}$.

The Bayes-optimal form of the denoising functions g_k and f_k is the same as in equation (5) but with vector-valued arguments. It can be observed that the only part of the GAMP algorithm that depends on the form of the GLM is the function

$$g_k^*\left(Z^k, \bar{Y}\right) = \frac{\mathbb{E}\left[Z \mid Z^k, \bar{Y}\right] - \mathbb{E}\left[Z \mid Z^k\right]}{\mathrm{Var}\left[Z \mid Z^k\right]} \tag{20}$$

where the terms $\mathbb{E}\left[Z \mid Z^k\right]$ and $\mathrm{Var}\left[Z \mid Z^k\right]$ are standard parameters of a conditional Gaussian distribution. Therefore, when applying GAMP to a new GLM, the only part of the algorithm that changes is the expectation $\mathbb{E}\left[Z \mid Z^k, \bar{Y}\right]$.

For the remainder of this report, we consider the case of a zero mean Gaussian prior on B, that is the rows of B are generated according to

$$\bar{B} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \Sigma_B\right) \tag{21}$$

where the Σ_B specifies the dependencies between each of the L signals. Under these conditions, the optimal denoising function f_k^* is given by a conditional Gaussian mean

$$f_{k+1}\left(\mathrm{M}_B^{k+1}\bar{B} + G_B^{k+1} = S\right) = \mathbb{E}[\bar{B} \mid \mathrm{M}_B^{k+1}\bar{B} + G_B^{k+1} = S] \tag{22}$$

$$= \Sigma_B \left(\mathrm{M}_B^{k+1}\right)^\top \left(\mathrm{M}_B^{k+1}\Sigma_B \left(\mathrm{M}_B^{k+1}\right)^\top + \mathrm{T}_B^{k+1}\right)^{-1} S. \tag{23}$$

Note that unlike the EM algorithm, GAMP only provides an estimate of the signal matrix and not the mixing proportions or noise level. If these parameters are unknown, an extra EM-step can be added between each GAMP iteration to fit them.

2.6.1 GAMP for Mixtures of Linear Regressions

The application of GAMP to finite mixture models will now be illustrated using mixtures of linear regressions as an example. As mentioned above, the only non-standard step of AMP is the function g_k^*, for which we must have a computable characterisation of the expectation $\mathbb{E}\left[Z \mid Z^k, \bar{Y}\right]$. Previous work has presented this for $L = 2$ and $L = 3$ components [10], but here we present the general case for L mixtures.

The expectation required can be decomposed with respect to the latent class label c

$$\mathbb{E}[Z \mid Z^k, \bar{Y}] = \sum_{j=1}^{L} \mathbb{E}[Z \mid Z^k, \bar{Y}, \bar{c} = j]\mathbb{P}(\bar{c} = j | Z^k, \bar{Y}). \tag{24}$$

where the relevant random variables are given by

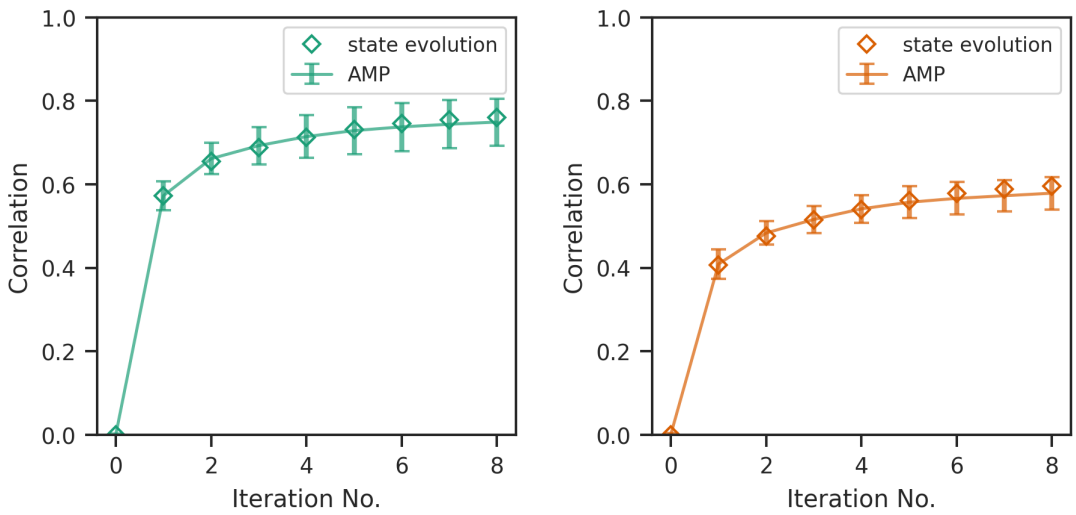
$$\bar{c} \sim \text{cat}(\alpha), \quad \bar{\epsilon} \sim \mathcal{N}(0, \sigma_\epsilon^2) \tag{25}$$

$$\bar{Y} = q(Z, [\bar{c}, \bar{\epsilon}]) = \bar{\epsilon} + \sum_{j=1}^{L} \mathbb{I}(\bar{c} = j) Z_j \tag{26}$$

The distribution $\mathbb{P}(Z \mid Z^k, \bar{Y}, \bar{c})$ is jointly Gaussian and it is straightforward to calculate the mean. The second distribution requires consideration of all L mixtures

$$\mathbb{P}(\bar{c} = j \mid Z^k, \bar{Y}) = \frac{\mathbb{P}(\bar{c} = j)\mathbb{P}(Z^k, \bar{Y} \mid \bar{c} = j)}{\sum_{l=1}^{L} \mathbb{P}(\bar{c} = l)\mathbb{P}(Z^k, \bar{Y} \mid \bar{c} = l)} = \frac{\alpha_j \mathbb{P}(Z^k, \bar{Y} \mid \bar{c} = j)}{\sum_{l=1}^{L} \alpha_l \mathbb{P}(Z^k, \bar{Y} \mid \bar{c} = l)} \tag{27}$$

where the distribution of $\mathbb{P}(Z^k, \bar{Y} \mid \bar{c})$ is also jointly Gaussian. These results can be plugged into (20) and used to compute the optimal form of the denoising function g_k^*.



(a) β_1, $\alpha_1 = 0.6$ (b) β_2, $\alpha_2 = 0.4$

Fig. 2: GAMP for mixed linear regression, signal correlation vs iteration number. $L = 2, \sigma_\epsilon^2 = 0.01, p = 500, n = 2500$

Figure 2 shows the correlation of the GAMP estimate vs true signal. A two component signal matrix was generated according to equation (21) with the following parameters

$$\alpha = [0.6, 0.4], \qquad \Sigma_B = \begin{bmatrix} 1 & 0.4 \\ 0.4 & 1 \end{bmatrix} \tag{28}$$

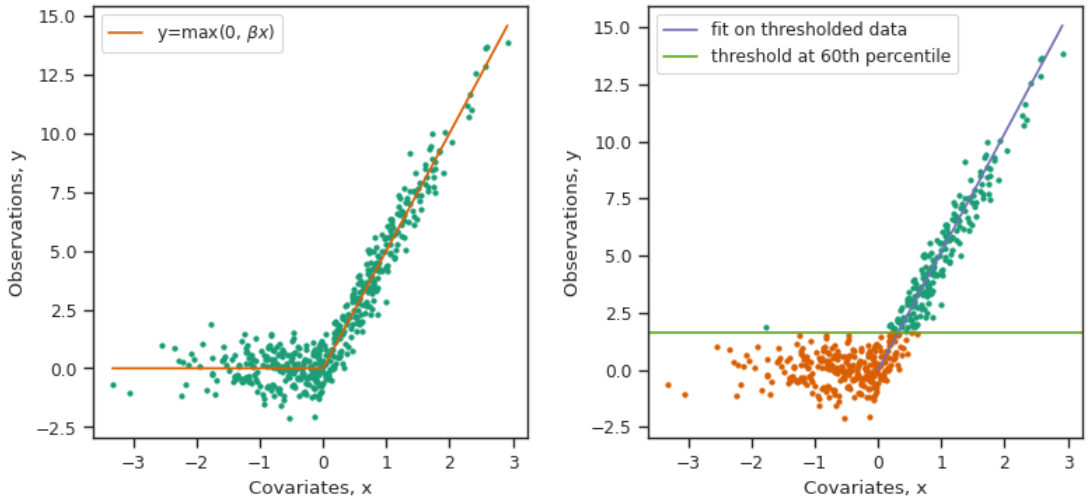
The state evolution prediction of the correlation is calculated according to equation (8). 25 runs of GAMP were performed and error bars report one standard deviation from the mean. We can see that the correlation of the GAMP signal follows the state evolution prediction almost exactly. Additionally, the first component achieves a stronger correlation than the second component, since 60% of the data comes from β_1.

3 Rectified Linear Regression

In this section, the rectified linear model and associated estimation algorithms are presented. In a rectified model, observations y_i are generated according to

$$y_i = \max(0, \beta^T x_i) + \epsilon_i, \qquad \epsilon_i \sim \mathcal{N}(0, \sigma_\epsilon^2) \tag{29}$$

where we have a Gaussian prior on the signal such that $\beta \sim \mathcal{N}(0, \sigma_\beta^2)$. Figure 3a shows an example of a dataset generated from the rectified model. It has been shown that in the general case, estimation in such a model is NP-hard if no assumptions are made on the distribution of the covariates x [6]. Under Gaussian assumptions, gradient-based optimisation methods can be used to fit the parameters of rectified models [5].



(a) Data generated from rectified linear regression model

(b) Best fit using the threshold method on the 60th percentile

Fig. 3: Example dataset generated from a rectified linear regression with best fit using threshold method.

Given an additional zero-mean assumption, a simple thresholding method can also be used to fit the parameters of a rectified linear model. Since the covariates x are zero mean and Gaussian, it is expected that approximately half of the data will be clipped at 0. Therefore, if we discard all samples x_i, y_i for which $y_i < 3\sigma_\epsilon$, the remaining samples will lie within the linear region with high probability. If the noise level is not known, the threshold can be chosen as a percentile of the observations y_i. For example, Figure 3b shows the best fit of a rectified linear regression model using a threshold at the 60th percentile of the data. Signal recovery can be further improved by using regularised linear regression to incorporate the prior distribution of β.

One disadvantage of the thresholding method is the fact that over half of the data is discarded before fitting. This can be significant in the high dimensional regime where $n \approx p$.

3.1 GAMP for Rectified Linear Regression

The GAMP algorithm described in Section 2.2 can be applied to the rectified linear model. This requires the computation of the denoising function g_k^* and the expectation

$\mathbb{E}\left[Z \mid Z^k, \bar{Y}\right]$. As before, the random variables Z, Z^k are multivariate Gaussian distributed with $\mathbb{P}\left(Z \mid Z^k\right) = \mathcal{N}\left(Z; \mu_1, \sigma_1^2\right)$. In this model, the random variable $\bar{Y}$ is given by

$$\bar{Y} = \max(0, Z) + \bar{\epsilon}, \quad \bar{\epsilon} \sim \mathcal{N}\left(0, \sigma_\epsilon^2\right) \tag{30}$$

The required expectation can be decomposed using the law of total expectation

$$\mathbb{E}\left[Z \mid Z^k, \bar{Y}\right] = \underbrace{\mathbb{E}\left[Z \mid Z^k, \bar{Y}, Z > 0\right] \mathbb{P}\left(Z > 0 \mid Z^k, \bar{Y}\right)}_{\text{Linear region}} \tag{31}$$

$$+ \underbrace{\mathbb{E}\left[Z \mid Z^k, \bar{Y}, Z \leq 0\right] \mathbb{P}\left(Z \leq 0 \mid Z^k, \bar{Y}\right)}_{\text{Rectified region}} \tag{32}$$

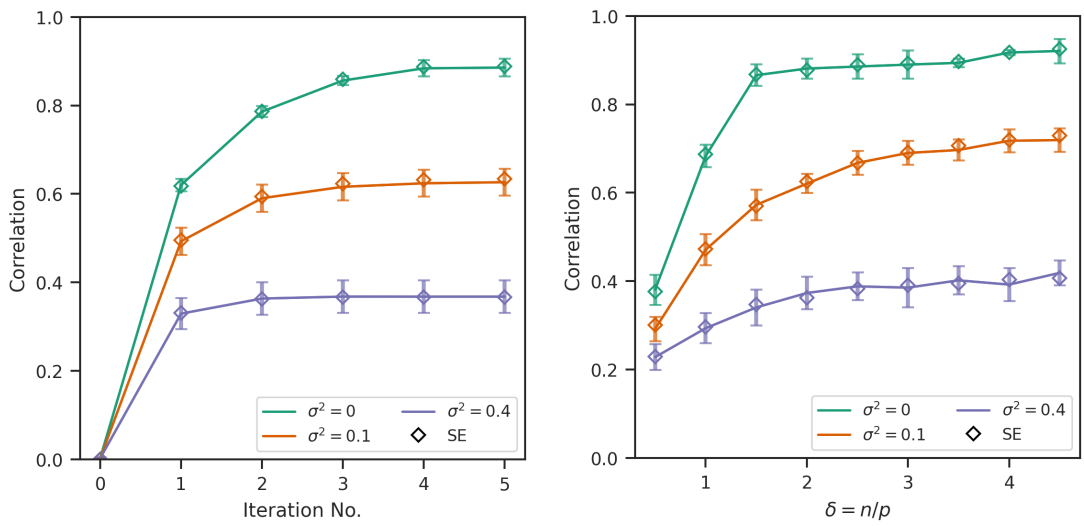
A full derivation of this expectation is provided in Appendix C, with the main result presented here. Let μ_1, σ_1^2 be the parameters of the conditional Gaussian distribution $\mathbb{P}(Z \mid Z^k)$. Then the expectations for each of the regions are the means of truncated Gaussian distributions, given by

$$\mathbb{E}\left[Z \mid Z^k, \bar{Y}, Z \leq 0\right] = \mu_1 - \sigma_1^2 \frac{\mathcal{N}(0; \mu_1, \sigma_1^2)}{\Phi(0; \mu_1, \sigma_1^2)} \tag{33}$$

$$\mathbb{E}\left[Z \mid Z^k, \bar{Y}, Z > 0\right] = \mu_2 + \sigma_2^2 \frac{\mathcal{N}(0; \mu_2, \sigma_2^2)}{1 - \Phi(0; \mu_2, \sigma_2^2)} \tag{34}$$

$$\mu_2 = \frac{\bar{Y}\sigma_1^2 + \mu_1 \sigma_\epsilon^2}{\sigma_1^2 + \sigma_\epsilon^2}, \qquad \sigma_2^2 = \frac{\sigma_1^2 \sigma_\epsilon^2}{\sigma_1^2 + \sigma_\epsilon^2} \tag{35}$$

Plugging these results into (32) and then (5) allows us to compute the Bayes-optimal GAMP recursion.



(a) Correlation vs iteration number, $\delta = 2$ (b) Correlation vs δ

Fig. 4: GAMP for rectified linear regression with varying noise levels $\sigma_\epsilon^2 = 0,\ 0.1,\ 0.4$. Model parameters are $p = 500$, $\sigma_\beta^2 = 1$.

Figure 4 shows numerical results from running GAMP on the rectified linear regression model. The $p = 500$ dimensional signal β is generated i.i.d. from a standard normal

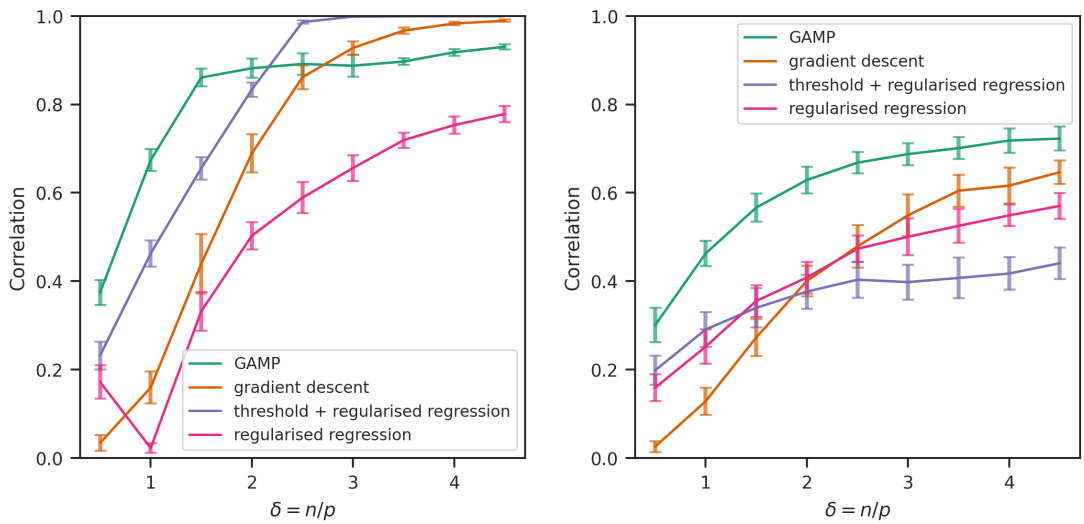
distribution with zero mean and unit variance. Each point is obtained from 25 independent runs with error bars at one standard deviation from the mean. The signal correlation and state evolution predictions are obtained from equation (8). In Figure 4a, we can see that GAMP exhibits rapid convergence in only a few iterations. Figure 4b shows that the performance of the algorithm improves as number of datapoints n increases. In the noiseless case, GAMP approaches full recovery of the signal for $\delta > 2$. However, the introduction of noise makes the estimation problem significantly more difficult. In all cases, the performance of the GAMP algorithm matches the state evolution almost exactly.

3.2 Comparison of Algorithms for Rectified Linear Regression

We will now compare GAMP to other existing estimators for the rectified linear regression model. In particular, the following algorithms are considered:

- Gradient descent for the rectified regression problem, as presented in [5].
- Thresholding followed by regularised linear regression, described above on page 11.
- Regularised linear regression.

The regularised linear regression was implemented using `sklearn.linear_model.Ridge` from scikit-learn [9].



(a) Noiseless case, $\sigma_\epsilon^2 = 0$ (b) Noisy case, $\sigma_\epsilon^2 = 0.1$

Fig. 5: Comparison of rectified regression algorithms with $p = 500$, $\sigma_\beta^2 = 1$

Figure 5 illustrates the performance of each estimation algorithm for a range of δ, where each point is generated from 25 independent runs. The signal β is generated i.i.d. according to $\mathcal{N}(0, 1)$ with $p = 500$. In the noiseless case, the thresholding method rapidly approaches complete recovery. This is unsurprising as thresholding can precisely capture the linear region when no noise is present. However, it is important to remember that the thresholding approach discards over half of the data. Consequently, for values of $\delta < 2$, the performance of GAMP remains superior. For large values of δ, gradient descent also performs well in the absence of noise. Figure 5b shows that GAMP is a robust estimator that significantly outperforms the other algorithms in the presence of noise.

4 Mixed Rectified Linear Regression

In the mixed rectified linear model, a mixture of L rectified linear regressions is considered. The GLM is given by

$$y_i = \max(0, \Gamma_i) + \epsilon_i, \quad \Gamma_i = \sum_{j=1}^{L} \mathbb{I}(c_i = j) \left[B^T x_i\right]_j, \quad c_i \sim \text{cat}(\alpha), \quad \epsilon_i \sim \mathcal{N}(0, \sigma_\epsilon^2) \tag{36}$$

where $B \in \mathbb{R}^{p \times L}$ is a matrix of L signals and the latent variable c_i indicates which of the signals the i-th observation is drawn from. The latent variables c_i are distributed according to a categorical prior such that $\mathbb{P}(c_i = j) = \alpha_j$. The signal B is distributed row-wise according to $\mathcal{N}(0, \Sigma_B)$, meaning that the signals for each component may be correlated.

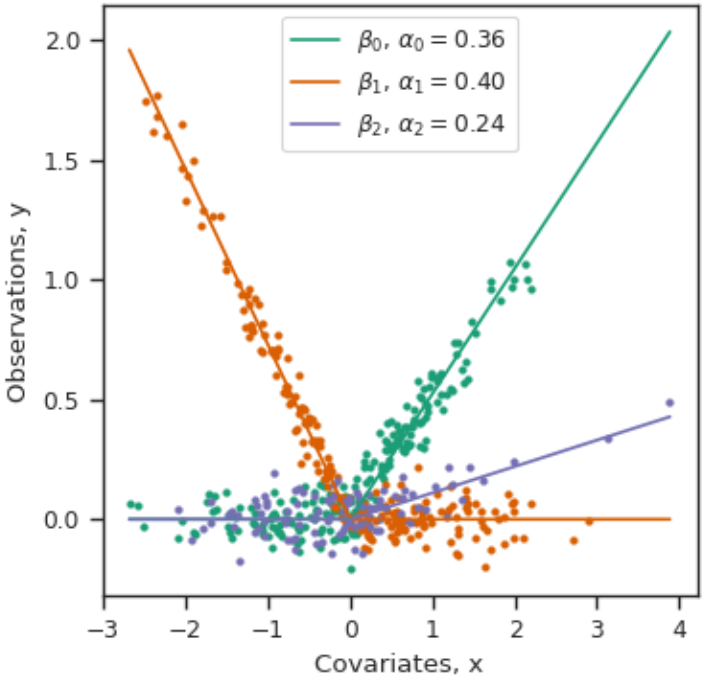


Fig. 6: Example mixed rectified linear regression, with $L = 3$

Figure 6 shows an example dataset generated from a mixed rectified regression model with $L = 3$ and $p = 1$. Fitting the parameters of the mixed rectified model can be achieved using EM, AM, gradient descent or GAMP, which we present in the following sections.

4.1 Expectation-Maximisation for Mixed Rectified Linear Regression

The EM algorithm presented in Section 2.4 can be used to estimate the parameters in a mixed rectified regression model. We assume that the parameters $\alpha, \Sigma_B, \sigma_\epsilon^2$ are known, so only the signal matrix B needs to be estimated. The E-step is given by

$$h_t(c_i = j) = \frac{\alpha_j \mathcal{N}\left(y_i; \max\left(x_i^T \hat{\beta}_j^{(t)}, 0\right), \sigma_\epsilon^2\right)}{\sum_{k=1}^{L} \alpha_k \mathcal{N}\left(y_i; \max\left(x_i^T \hat{\beta}_k^{(t)}, 0\right), \sigma_\epsilon^2\right)}, \tag{37}$$

and the M-step takes the form of any weighted rectified linear regression fitting algorithm. The algorithms from the previous section were all considered, but numerical results showed that standard regularised regression weighted by $h_t(c_i)$ gave the best performance.

4.2 Alternating-Minimisation for Mixed Rectified Linear Regression

The AM algorithm involves partitioning the data into groups based on which component it is most likely to have come from, followed by standard rectified linear regression fitting. In this case, we assign a datapoint (x_i, y_i) to component j if

$$j = \underset{j}{\operatorname{argmax}}\ h_t(c_i = j) \tag{38}$$

where $h_t(c_i)$ is given by the E-step in (37). Again any rectified linear regression algorithm can be used for the fitting step, but simple regularised regression was found to perform better than other methods.

4.3 Gradient Descent for Mixed Rectified Linear Regression

Gradient descent can be used to fit the parameters for each component individually by minimising the loss for each component individually. Defining a loss function with component-wise squared loss,

$$L(B, X, y) = \sum_{i=1}^{n} \sum_{j=1}^{L} \left(\max(\beta_j^T x_i, 0) - y_i\right)^2 \tag{39}$$

we can use a result from [5] to obtain the following gradient descent algorithm

$$\hat{\beta}_j^{(t+1)} = \hat{\beta}_j^{(t)} - \frac{\eta}{n} \sum_{i=1}^{n} 2 \left(\max(x_i^T \hat{\beta}_j^{(t)}, 0) - y_i\right) \left(1 + \operatorname{sign}\left(x_i^T \hat{\beta}_j^{(t)}\right)\right) x_i \tag{40}$$

where $\eta > 0$ is the learning rate. This simple algorithm proved to be effective in fitting the parameters of the mixed rectified model.

4.4 GAMP for Mixed Rectified Linear Regression

Computing the Bayes-optimal GAMP recursion for the mixed rectified model requires a computable expression for the denoising function g_k^* in equation (20). In the case of the mixed rectified model, the random variables of interest are given by

$$\bar{Y} = \max\left(0, \sum_{j=1}^{L} \mathbb{I}(\bar{c} = j) Z_j\right) + \bar{\epsilon}, \quad \bar{c} \sim \operatorname{cat}(\alpha), \quad \bar{\epsilon} \sim \mathcal{N}(0, \sigma_\epsilon^2), \quad \omega_j = \mathbb{I}(Z_j > 0) \tag{41}$$

and we require the expectation

$$\mathbb{E}\left[Z \mid Z^k, \bar{Y}\right] = \sum_{j=1}^{L} \mathbb{P}\left(\bar{c} = j \mid Z^k, \bar{Y}\right) \begin{bmatrix} \mathbb{E}\left[Z_1 \mid Z^k, \bar{Y}, \bar{c} = j\right] \\ \vdots \\ \mathbb{E}\left[Z_L \mid Z^k, \bar{Y}, \bar{c} = j\right] \end{bmatrix}. \tag{42}$$

Like the non-mixed case, we can break this down into the rectified and linear regions

$$\begin{aligned}\mathbb{E}\left[Z_i \mid Z^k, \bar{Y}, \bar{c}=j\right] = &\overbrace{\mathbb{P}\left[\omega_j = 0 \mid Z^k, \bar{Y}, \bar{c}=j\right] \mathbb{E}\left[Z_i \mid Z^k, \bar{Y}, \bar{c}=j, \omega_j = 0\right]}^{\text{Rectified region}} \\ &+ \underbrace{\mathbb{P}\left[\omega_j = 1 \mid Z^k, \bar{Y}, \bar{c}=j\right] \mathbb{E}\left[Z_i \mid Z^k, \bar{Y}, \bar{c}=j, \omega_j = 1\right]}_{\text{Linear region}} \end{aligned} \tag{43}$$

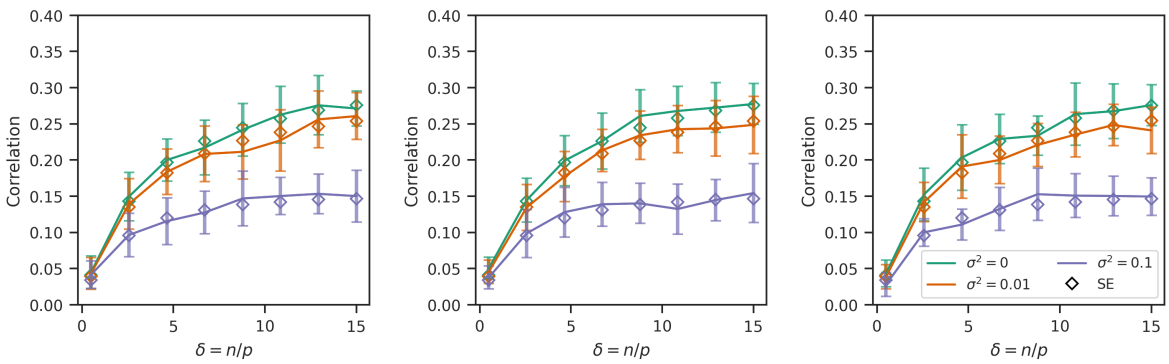
Exact expressions for all the quantities above can be found in Appendix D and can be used to implement a Bayes-optimal GAMP recursion.

We now present numerical results for the mixed rectified linear model with $L = 3$ and $p = 500$. All results are generated from 25 independent runs with error bars at one standard deviation.

Case 1: Equal proportions, independent signals

Figure 7 shows the performance of GAMP where the component proportions $\alpha = \left[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right]$ and the prior covariance $\Sigma_B = I_3$ is the identity matrix. This is a challenging case, as any solution is invariant to the permutation of the signal indices. For example, the GAMP algorithm could estimate a signal matrix $\hat{B} \approx (\beta_2, \beta_1, \beta_0)$, which has low correlation with the true signal matrix due to the permutation of the signals. Therefore we observe that GAMP cannot distinguish the signals reliably and the correlation is below 0.3 even in the noiseless case.

In all cases, the correlation of the estimates produced by GAMP follows the state evolution prediction closely.



(a) β_1, $\alpha_1 = 0.33$ (b) β_2, $\alpha_2 = 0.33$ (c) β_3, $\alpha_3 = 0.33$

Fig. 7: GAMP for the mixed rectified regression model: Equal proportions, independent signals

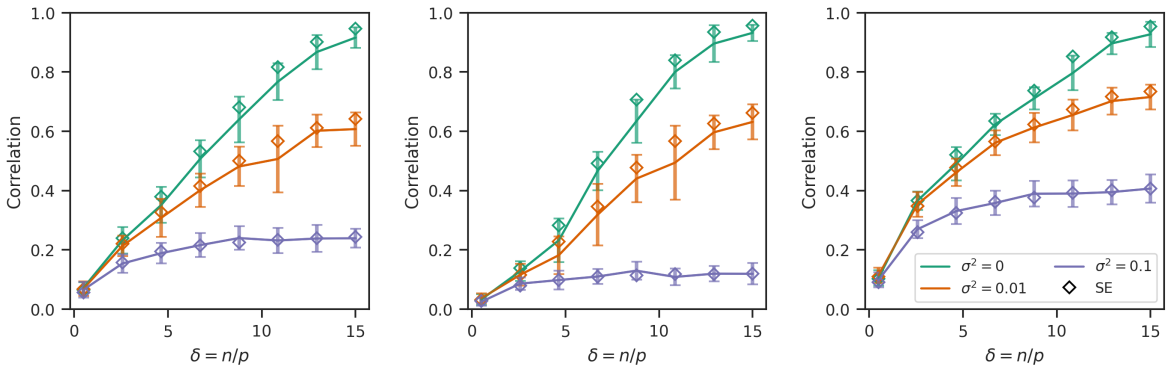
Case 2: Equal proportions, dependent signals

We now consider the case where the prior contains dependencies between the signals of each component. The prior covariance was chosen to be

$$\Sigma_B = \begin{bmatrix} 1 & -0.2 & 0.6 \\ -0.2 & 1.1 & 0.1 \\ 0.6 & 0.1 & 1.5 \end{bmatrix} \tag{44}$$

where signals 1 and 3 are strongly correlated and signal 2 is weakly correlated with the others. The component proportions are set to $\alpha = \left[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right]$.

Figure 8 shows the performance of GAMP under these conditions. The performance is greatly improved over the previous case, as the symmetry between the signals is broken. In the noiseless case, the algorithm achieves almost full recovery at $\delta = 15$. The amount of data required for strong recovery is far more than in the non-mixed case. This is unsurprising as inference in a mixed model is more challenging than in the non-mixed setting. We also observe that β_1 and β_3 exhibit stronger recovery than β_2 due to the strong dependencies between them.

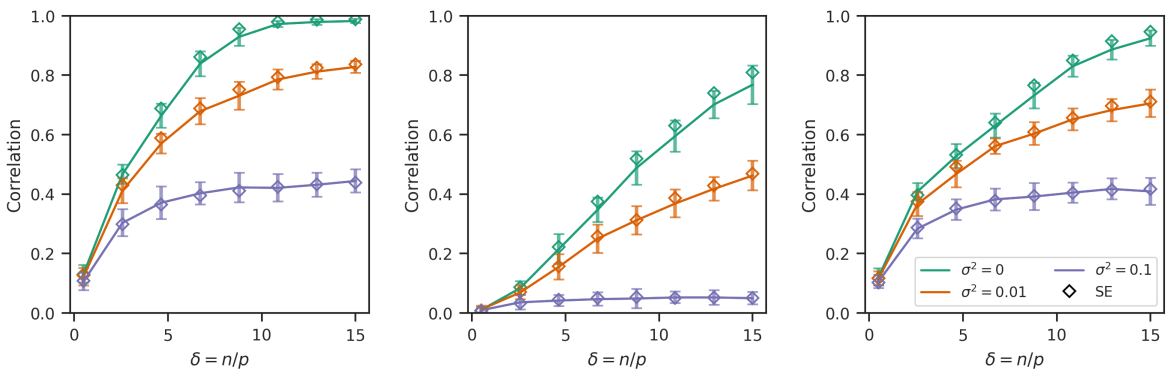


(a) β_1, $\alpha_1 = 0.33$ (b) β_2, $\alpha_2 = 0.33$ (c) β_3, $\alpha_3 = 0.33$

Fig. 8: GAMP for the mixed rectified regression model: Equal proportions, dependent signals

Case 3: Unequal proportions, dependent signals

In this case, the component proportions are $\alpha = [0.5, 0.2, 0.3]$. The signals are generated according to same Σ_B in equation (44). The performance of GAMP in this setting can be seen in Figure 9. Here 50% of the observations come from a single signal and the GAMP estimates of β_1 show strong correlation. Component 2 has the least data and GAMP almost completely fails to estimate it when $\sigma_\epsilon^2 = 0.1$. Component 3 also has fewer observations, but estimation is helped by the dependence with β_1. This dependency provides additional information that aids in the estimation of β_3.



(a) β_1, $\alpha_1 = 0.5$ (b) β_2, $\alpha_2 = 0.2$ (c) β_3, $\alpha_3 = 0.3$

Fig. 9: GAMP for the mixed rectified regression model: Unequal proportions, dependent signals

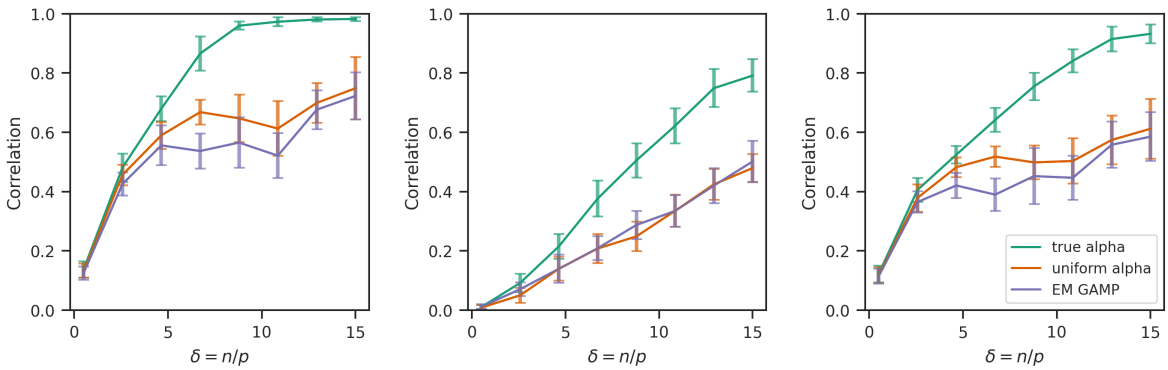
Case 4: Effect of unknown proportions

In most real-world applications, the true component proportions are unknown. Running the GAMP algorithm with incorrect parameters results in sub-optimal denoising functions f_k and g_k being used. A variant of GAMP known as EM-GAMP was proposed to estimate both the signal B and parameters α. EM-GAMP uses an EM-step to estimate the component proportions using equation (16) with h_t from equation (37).

Figure 10 shows the performance of GAMP when the true component proportions are unknown. The data was generated according to $\alpha = [0.5, 0.2, 0.3]$, $\sigma_\epsilon^2 = 0$ and Σ_B from equation (44). We compare the following three cases:

1. GAMP with true component proportions
2. GAMP with uniform $\alpha = \left[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right]$
3. EM-GAMP with EM-step to update component proportions

It was observed that when an exact estimate of the parameter B was available, the EM-step successfully estimated the mixing proportions. However, in cases where the estimate of B was poor, the EM-step's estimates of the mixing proportions deviated significantly from the true values. In Figure 10, the quality of the estimates of B from the EM-GAMP algorithm was insufficient for accurate estimation of the mixing proportions. Therefore, the introduction of the EM-step did not enhance the overall performance vs simply running GAMP with uniform proportions. It is possible that a better initialisation method could be used to improve the performance of EM-GAMP.

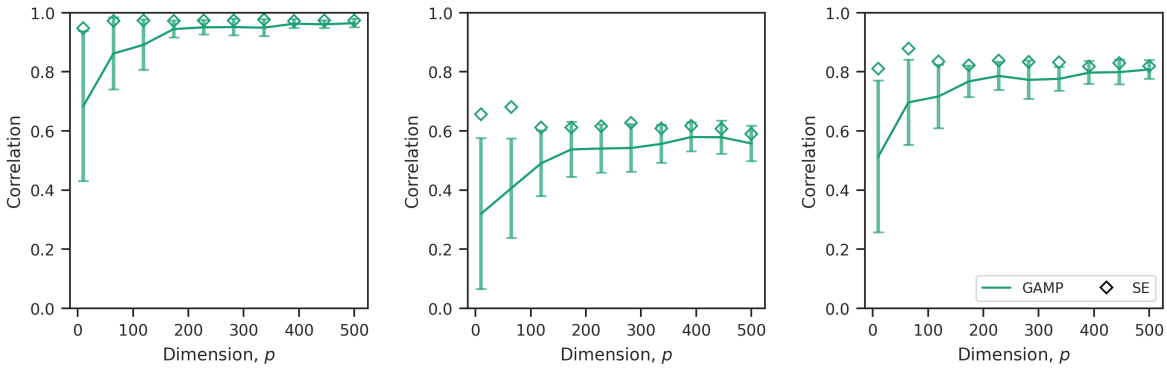


(a) β_1, $\alpha_1 = 0.5$ (b) β_2, $\alpha_2 = 0.2$ (c) β_3, $\alpha_3 = 0.3$

Fig. 10: GAMP vs EM-GAMP for the mixed rectified regression model

Case 5: Effect of dimensionality

Figure 11 shows the effect that changing the dimension p has on the performance of GAMP. The signal is again generated with $\alpha = [0.5, 0.2, 0.3]$ and Σ_B from equation (44). We observe that for low dimensional problems, $p < 200$, the GAMP estimates no longer match the state evolution predictions. This is because several values in the GAMP iteration, including C^k, F^k and Σ^k, take the form of expectations that we approximate directly from the samples [10]. Therefore, the variance of the numerical approximation grows as the number of samples n, p decreases and the performance of GAMP degrades.



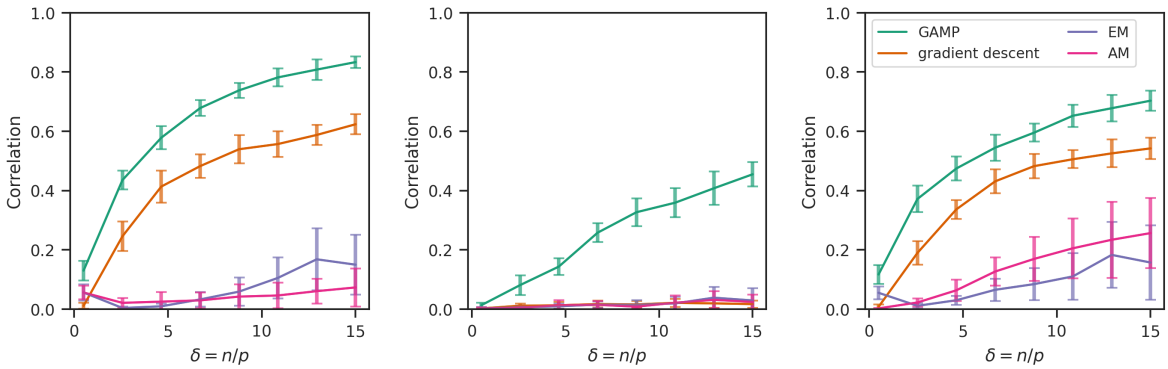
(a) β_1, $\alpha_1 = 0.5$ (b) β_2, $\alpha_2 = 0.2$ (c) β_3, $\alpha_3 = 0.3$

Fig. 11: Effect of dimensionality on GAMP for the mixed rectified regression model, $\delta = 10$

4.5 Comparison of Algorithms for Mixed Rectified Linear Regression

A comparison of the algorithms described above will now be presented. A moderate noise level of $\sigma_\epsilon^2 = 0.01$ was used and the signals were generated as in case 3 above.

Figure 12 displays the performance of the algorithms on the mixed rectified regression model. It is clear that GAMP significantly outperforms the other estimators. Although gradient descent exhibits comparable performance for β_1 and β_3, it fails to accurately estimate β_2. Similarly, both EM and AM fail to achieve highly correlated estimates for any of the signals. This could be attributed to the random initialization, which potentially causes EM and AM to converge to poor local optima. Despite attempting a spectral initialization method [11], its impact on performance was found to be minimal. Future work is required to develop more appropriate initialization methods tailored for the mixed rectified linear model.



(a) β_1, $\alpha_1 = 0.5$ (b) β_2, $\alpha_2 = 0.2$ (c) β_3, $\alpha_3 = 0.3$

Fig. 12: Comparison of estimators for the mixed rectified linear model, unequal prior and dependent signals

5 Logistic Regression

Logistic regression is a widely employed model for tackling simple classification problems. In this model, the binary-valued random variables y_i serve as observations and are generated by

$$y_i = \begin{cases} 1 & \text{with probability } \sigma\left(\beta^T X_i\right) \\ 0 & \text{with probability } 1 - \sigma\left(\beta^T X_i\right) \end{cases} \tag{45}$$

where $\sigma(u) = (1 + \exp(-u))^{-1}$ is the sigmoid function. We consider the Bayesian case where we have a Gaussian prior on the signal such that $\beta \sim \mathcal{N}(0, \sigma_\beta^2)$. Figure 13 shows an example dataset generated from a logistic GLM.

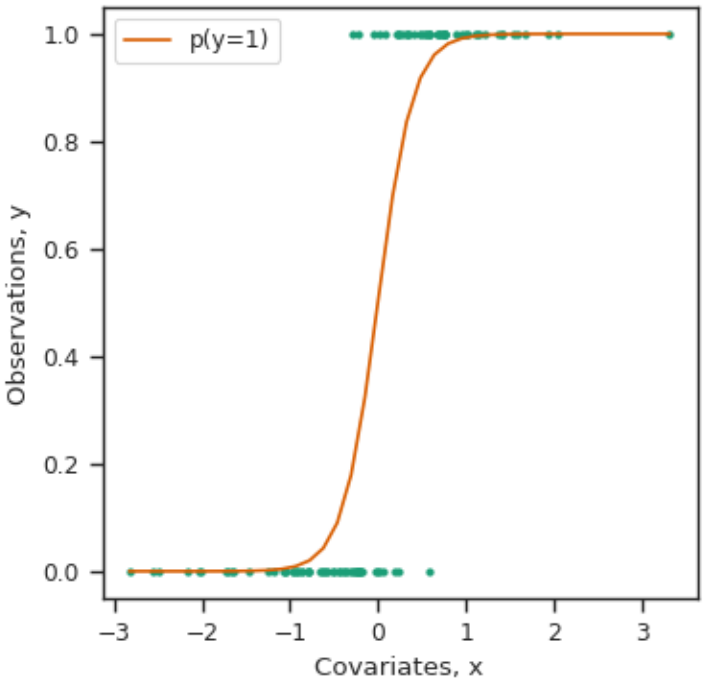


Fig. 13: Example data generated from a logistic GLM

A maximum-likelihood fit of the parameters of a logistic regression can be achieved using a variety of numerical optimisation algorithms. This is because the logistic loss function is convex leading to a straightforward optimisation problem. There is no analytical solution for full Bayesian inference in a logistic regression model, but a variety of approximate methods exist [2].

Noise in the logistic regression model is introduced implicitly through the probabilistic generation of observations. Therefore, an effective noise level can be controlled through the signal strength σ_β^2. For example, if σ_β^2 is large, the decision boundary will be sharp. This makes the signal estimation problem significantly easier as the observations will match $\sigma(\beta^T X)$ very closely.

5.1 GAMP for Logistic Regression

In this section, we present the Bayes-optimal GAMP recursion for logistic regression. Recall that in the non-mixed case, the optimal function g_k^* in equation (5) requires the computation of the expectation $\mathbb{E}\left[Z \mid Z^k, \bar{Y}\right]$, where random variables Z, Z^k and $\bar{Y}$ are distributed as

$$\begin{bmatrix} Z \\ Z^k \end{bmatrix} \sim \mathcal{N}\left(0, \Sigma^k\right) = \mathcal{N}\left(0, \begin{bmatrix} \sigma_Z^2 & \rho \\ \rho & \sigma_{Z^k}^2 \end{bmatrix}\right) \tag{46}$$

$$\mathbb{P}\left(Z \mid Z^k\right) = \mathcal{N}\left(Z; \frac{\rho}{\sigma_{Z^k}^2} Z^k, \sigma_Z^2 - \frac{\rho^2}{\sigma_{Z^k}^2}\right) = \mathcal{N}\left(Z; \mu_1, \sigma_1^2\right) \tag{47}$$

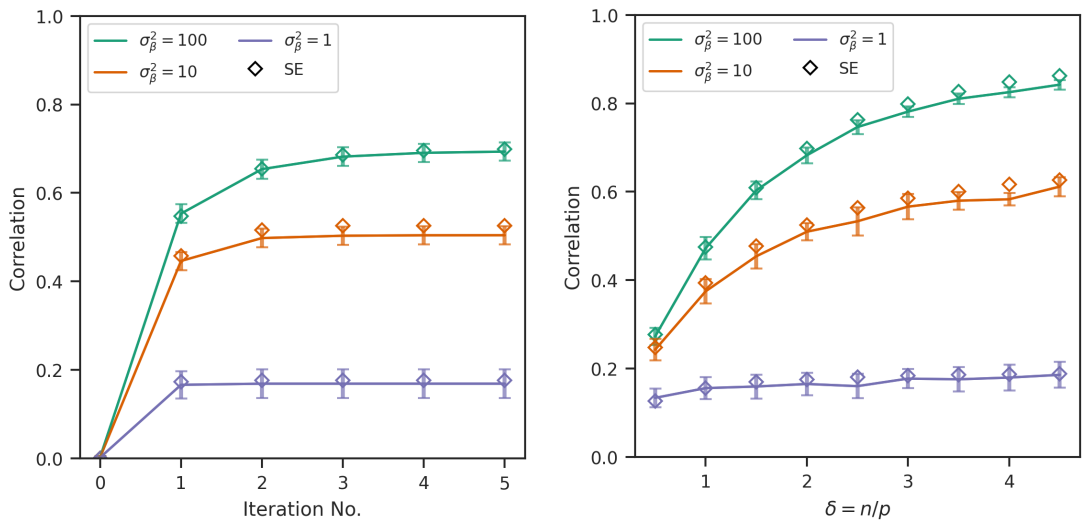
$$\bar{Y} = \begin{cases} 1 & \text{w.p. } \sigma(Z) \\ 0 & \text{w.p. } 1 - \sigma(Z) \end{cases} \tag{48}$$

Unfortunately, computing the distribution of $\mathbb{P}(Z|Z^k, \bar{Y})$ requires an integral over the product of a normal distribution and a sigmoid function, which is intractable. However, we can obtain an approximation of the integral via the probit approximation [2]. Defining $\lambda = \sqrt{\pi/8}$ and $\gamma = \frac{\lambda}{\sqrt{1+\sigma_1^2\lambda^2}}$, the required expectation is given by

$$\mathbb{E}\left[Z \mid Z^k, \bar{Y} = 1\right] \approx \frac{\gamma\sigma_1^2\ \phi\left(\gamma\mu_1\right) + \mu_1\Phi\left(\gamma\mu_1\right)}{\Phi\left(\gamma\mu_1\right)} \tag{49}$$

$$\mathbb{E}\left[Z \mid Z^k, \bar{Y} = 0\right] \approx \frac{\mu_1 - \gamma\sigma_1^2\ \phi\left(\gamma\mu_1\right) - \mu_1\Phi\left(\gamma\mu_1\right)}{1 - \Phi\left(\gamma\mu_1\right)} \tag{50}$$

where ϕ and Φ denote the standard normal pdf and cdf, respectively. Details of this result can be found in Appendix A.



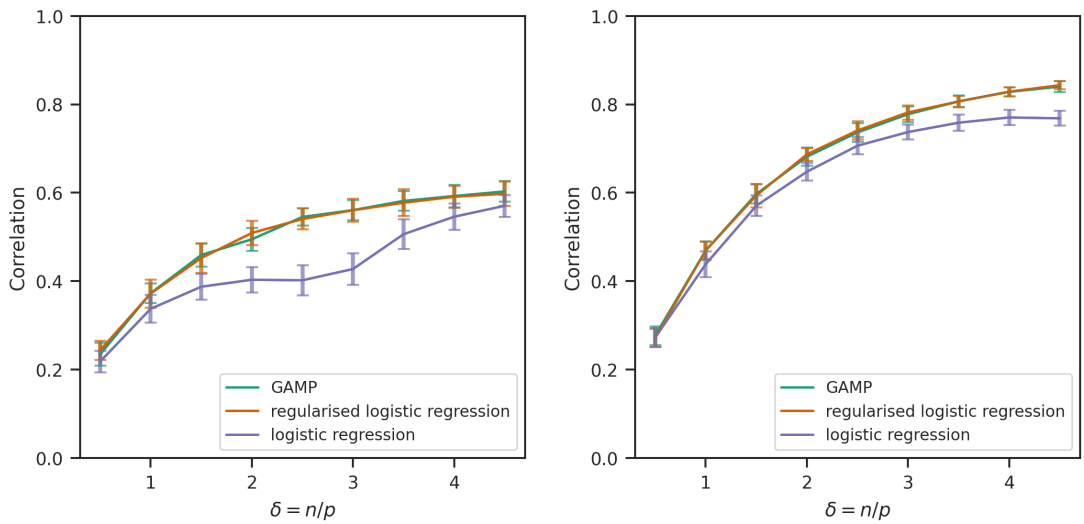
(a) Correlation vs iteration number, $\delta = 2$ (b) Correlation vs δ

Fig. 14: GAMP for logistic regression, $p = 500$

Figure 14 shows trial runs of GAMP for the logistic regression model. Despite the use of the probit approximation in the computation of g_k^*, the correlation of the GAMP estimates still closely matches the state evolution. As described above, increasing the signal strength σ_β^2 significantly improves the recovery of β.

5.2 Comparison of Algorithms for Logistic Regression

Figure 15 displays a comparison of GAMP vs standard logistic regression solvers. The signal is generated i.i.d. from a Gaussian distribution such that $\beta \sim \mathcal{N}(0, \sigma_\beta^2)$. Each datapoint was generated from 25 independent runs with error bars at one standard deviation. We use `sklearn.linear_model.LogisticRegression` from scikit-learn [9] as a comparison. This model allows for both standard and regularised logistic regression fitting. The performance of GAMP and the regularised logistic regression solver is found to be identical. For regularised logistic regression, the loss function is convex and there is a unique solution. Although GAMP does not solve this optimisation problem directly, it appears that it still reaches this optimal solution. Despite the identical performance on the standard logistic regression case, an advantage of GAMP is its ability to generalise to mixed logistic models, which we present in the following section.



(a) $\sigma_\beta^2 = 10$ (b) $\sigma_\beta^2 = 100$

Fig. 15: Comparison of algorithms for logistic regression, $p = 500$

6 Mixed Logistic Regression

Now we introduce the mixed logistic regression model. Under this model, each observation y_i generated from one of L logistic regressions according to a latent variable c_i

$$y_i = \begin{cases} 1 & \text{w.p. } \sigma(\Gamma_i) \\ 0 & \text{w.p. } 1 - \sigma(\Gamma_i) \end{cases}, \qquad \Gamma_i = \sum_{j=1}^{L} \mathbb{I}(c_i = j) \left[B^T x_i\right]_j, \qquad c_i \sim \text{cat}(\alpha) \tag{51}$$

Figure 16 shows an example dataset with $L = 2$. There are two main inference aims for this type of data. Firstly, we may wish to infer the signal matrix B. Alternatively, we could be interested in the performance of a classifier on unseen data. Inference of the signal matrix in this setting is a hard problem, as the observations are binary and there are several degenerate settings. For example, in the case of $L = 2$, $\alpha_1 = \alpha_2$ and $\beta_1 = -\beta_2$, the distribution of $\mathbb{P}(y = 1 \mid x) = 0.5$ regardless of the input x.

In the following sections, the EM, AM and GAMP algorithms for the mixed logistic regression model are presented. The performance of the algorithms is then compared both in terms of signal correlation and as a classifier.

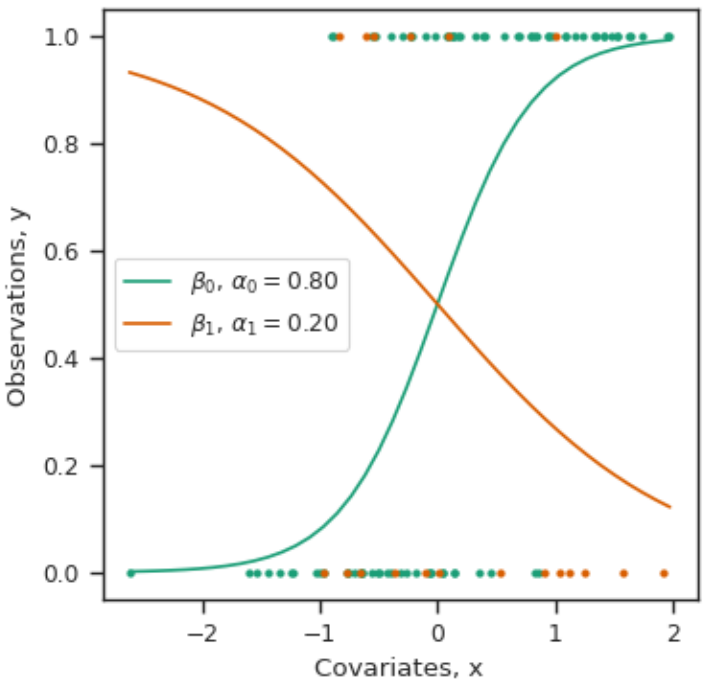


Fig. 16: Example mixed logistic regression, with $L = 2$

6.1 Expectation-Maximisation for Mixed Logistic Regression

The EM algorithm for mixed logistic regression takes a similar form to the one presented in Section 2.4. The E-step corresponds to computing an estimate of the posterior distribution $h_t(c_i = j) = \mathbb{P}(c_i = j \mid y_i, x_i, \beta_j)$:

$$h_t(c_i = j) = \frac{\hat{\alpha}_j^{(t)} \left[y_i \sigma \left(x_i^T \beta_j^{(t)} \right) + (1 - y_i) \sigma \left(-x_i^T \beta_j^{(t)} \right) \right]}{\sum_{k=1}^{L} \hat{\alpha}_k^{(t)} \left[y_i \sigma \left(x_i^T \beta_k^{(t)} \right) + (1 - y_i) \sigma \left(-x_i^T \beta_k^{(t)} \right) \right]} \tag{52}$$

The M-step then takes the form of fitting the parameters of a weighted logistic regression:

$$\hat{\beta}_j^{(t+1)} = \underset{\beta_j}{\operatorname{argmin}} \sum_{i=1}^{n} h_t(c_i = j) \log \left(\hat{\alpha}_j^{(t)} \left[y_i \sigma \left(x_i^T \beta_j \right) + (1 - y_i) \sigma \left(-x_i^T \beta_j \right) \right] \right) \tag{53}$$

There is no analytical solution to this problem, but it is convex and a wide variety of optimisation methods can be used. Numerical tests showed that using a regularised weighted logistic regression solver for the M-step provided the best performance.

6.2 Alternating-Minimisation for Mixed Logistic Regression

The AM algorithm for the mixed logistic regression model consists of two steps. Firstly, we assign each sample x_i, y_i to the class with the highest posterior probability:

$$P_j^{(t)} := \left\{ i \in \{1, \ldots, N\} : j = \underset{j \in \{1, \ldots, L\}}{\operatorname{argmax}} \left(y_i \sigma \left(x_i^T \hat{\beta}_j^{(t)} \right) + (1 - y_i) \sigma \left(-x_i^T \hat{\beta}_j^{(t)} \right) \right) \right\} \tag{54}$$

Then the parameters of a simple logistic regression are fit to each class,

$$\hat{\beta}_j^{(t+1)} = \underset{\beta_j}{\operatorname{argmin}} \sum_{i \in P_j^{(t)}} \log\left(\hat{\alpha}_j^{(t)}\left[y_i \sigma\left(x_i^T \beta_j\right) + (1-y_i)\sigma\left(-x_i^T \beta_j\right)\right]\right) \tag{55}$$

which can be done using numerical optimisation methods.

6.3 GAMP for Mixed Logistic Regression

To compute the GAMP recursion for mixed logistic regression, the form of the denoising function in equation (20) must be found. The iterates $Z, Z^k \in \mathbb{R}^L$ are now jointly Gaussian random *vectors* and the random variable $\bar{Y}$ is defined via the link function

$$\bar{Y} = \begin{cases} 1 & \text{w.p. } \sigma(\bar{\Gamma}) \\ 0 & \text{w.p. } 1-\sigma(\bar{\Gamma}) \end{cases}, \quad \bar{\Gamma} = \sum_{j=1}^{L} \mathbb{I}(\bar{c}=j) Z_j, \quad \bar{c} \sim \operatorname{cat}(\alpha) \tag{56}$$

$$\begin{bmatrix} Z \\ Z^k \end{bmatrix} \sim \mathcal{N}(0, \Sigma^k) = \mathcal{N}\left(0, \begin{bmatrix} \Sigma^{(11)} & \Sigma^{(12)} \\ \Sigma^{(21)} & \Sigma^{(22)} \end{bmatrix}\right) \tag{57}$$

The expectation $\mathbb{E}\left[Z \mid Z^k, \bar{Y}\right]$ is now a vector and can be decomposed with respect to $\bar{c}$

$$\mathbb{E}\left[Z \mid Z^k, \bar{Y}\right] = \sum_{j=1}^{L} \mathbb{P}\left(\bar{c}=j \mid Z^k, \bar{Y}\right) \begin{bmatrix} \mathbb{E}\left[Z_1 \mid Z^k, \bar{Y}, \bar{c}=j\right] \\ \vdots \\ \mathbb{E}\left[Z_L \mid Z^k, \bar{Y}, \bar{c}=j\right] \end{bmatrix} \tag{58}$$

The resulting expectations take a similar form to the non-mixed case

$$\mathbb{E}[Z_i \mid Z^k, \bar{Y}=1, \bar{c}=j] \approx \frac{\frac{\alpha_{ji}\sigma_i^2}{\sqrt{1+\alpha_{ji}^2\sigma_i^2}}\phi\left(\frac{\alpha_{ji}\mu_i+\gamma_{ji}}{\sqrt{1+\alpha_{ji}^2\sigma_i^2}}\right) + \mu_i \Phi\left(\frac{\alpha_{ji}\mu_i+\gamma_{ji}}{\sqrt{1+\alpha_{ji}^2\sigma_i^2}}\right)}{\Phi\left(\frac{\mu_j}{\sqrt{\lambda^{-2}+\sigma_j^2}}\right)} \tag{59}$$

$$\mathbb{E}[Z_i \mid Z^k, \bar{Y}=0, \bar{c}=j] \approx \frac{\mu_i - \frac{\alpha_{ji}\sigma_i^2}{\sqrt{1+\alpha_{ji}^2\sigma_i^2}}\phi\left(\frac{\alpha_{ji}\mu_i+\gamma_{ji}}{\sqrt{1+\alpha_{ji}^2\sigma_i^2}}\right) - \mu_i \Phi\left(\frac{\alpha_{ji}\mu_i+\gamma_{ji}}{\sqrt{1+\alpha_{ji}^2\sigma_i^2}}\right)}{1-\Phi\left(\frac{\mu_j}{\sqrt{\lambda^{-2}+\sigma_j^2}}\right)} \tag{60}$$

where we have defined the parameters of the conditional Gaussian distributions as

$$\mathbb{P}(Z_i \mid Z^k) = \mathcal{N}(Z_i; \mu_i, \sigma_i^2) \tag{61}$$

$$\mathbb{P}(Z_j \mid Z_i, Z^k) = \mathcal{N}(Z_j; \mu_{ji}, \sigma_{ji}^2) \tag{62}$$

$$\mu_{ji}\left(\sqrt{\lambda^{-2}+\sigma_{ji}^2}\right)^{-1} = \alpha_{ji} Z_i + \gamma_{ji} \tag{63}$$

The probability $\mathbb{P}\left(\bar{c}=j \mid Z^k, \bar{Y}\right)$ can be found straightforwardly using Bayes' rule. A computable approximation of the Bayes-optimal denoising function g_k^* has been obtained, allowing the GAMP recursion to be applied to the mixed logistic model. Appendix B contains the full derivation of the results presented above.

We now present numerical results of running GAMP for mixed logistic regression with $L = 3$ and $p = 500$. As the results are similar to those presented in Section 4, we only briefly detail them below.

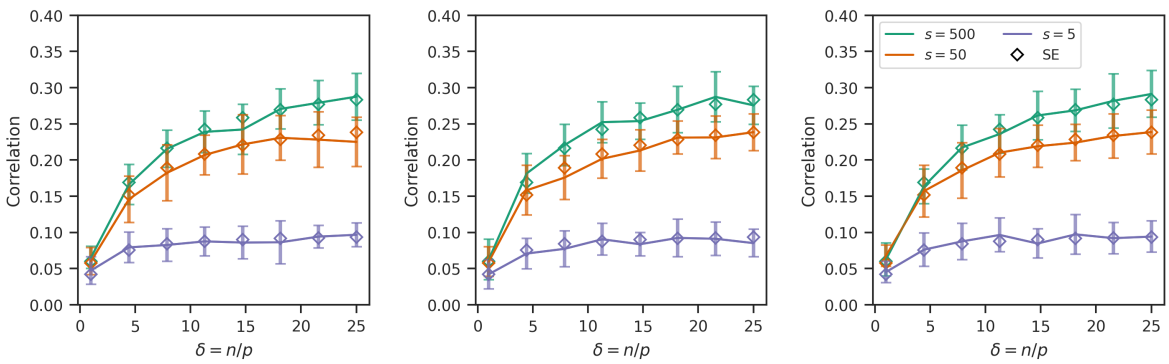
1. **Equal proportions, independent signals.** Figure 17 shows the results of running GAMP with $\alpha = \left[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right]$ and the prior covariance $\Sigma_B = sI_3$. The scale factor s controls the effective noise level. As in the mixed rectified linear regression case, the symmetrical prior makes estimation of the signals difficult. Therefore, GAMP is unable to obtain estimates with correlation greater than 0.3. All estimates follow the state evolution predictions closely.

2. **Equal proportions, dependent signals.** We now set a dependent prior on the signals, with the signal covariance equal to

$$\Sigma_B = s \begin{bmatrix} 1 & -0.2 & 0.6 \\ -0.2 & 1.1 & 0.1 \\ 0.6 & 0.1 & 1.5 \end{bmatrix}. \tag{64}$$

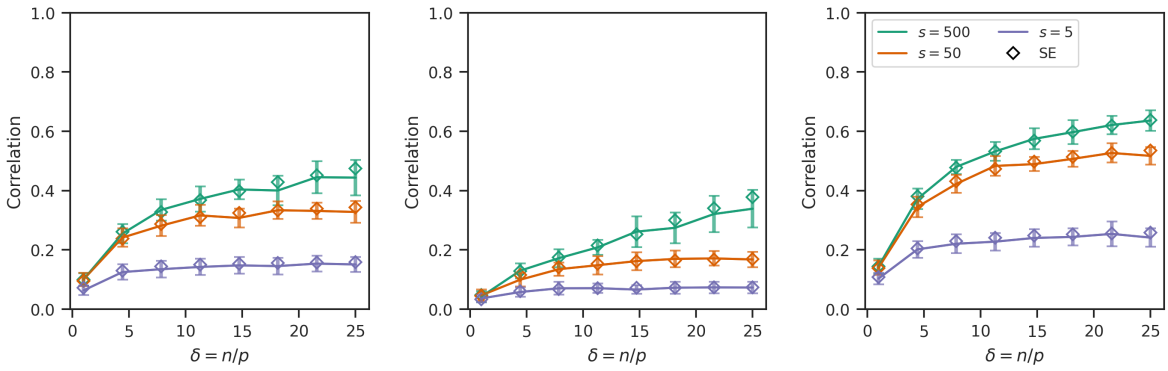
   We can see in Figure 18 that the added dependencies between the signals improves the performance of GAMP. The correlation of β_1 and β_3 improves their estimates, while the weakly correlated β_2 has the worst performance. We note that in all cases a large amount of data is required for recovery, $\delta > 20$. This is due to the inherent difficulty of signal estimation given only binary valued observations.

3. **Unequal proportions, dependent signals.** Now we set $\alpha = [0.5, 0.2, 0.3]$ and keep Σ_B as in (64). Figure 19 shows the results of running GAMP under these conditions. It is clear that the extra data generated from β_1 greatly improves its recovery. On the other hand, the weak correlation and small proportion α_2 limit the performance of β_2.



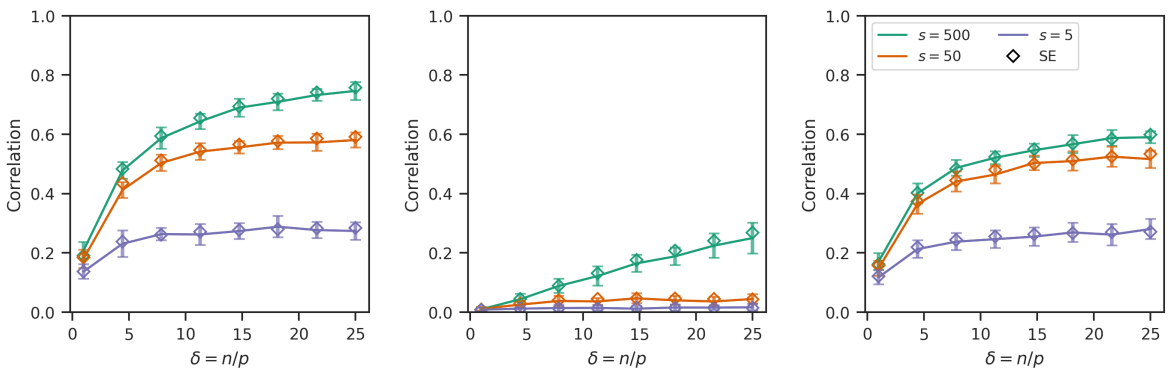
(a) β_1, $\alpha_1 = 0.33$ (b) β_2, $\alpha_2 = 0.33$ (c) β_3, $\alpha_3 = 0.33$

Fig. 17: GAMP for the mixed logistic regression model: Equal proportions, independent signals



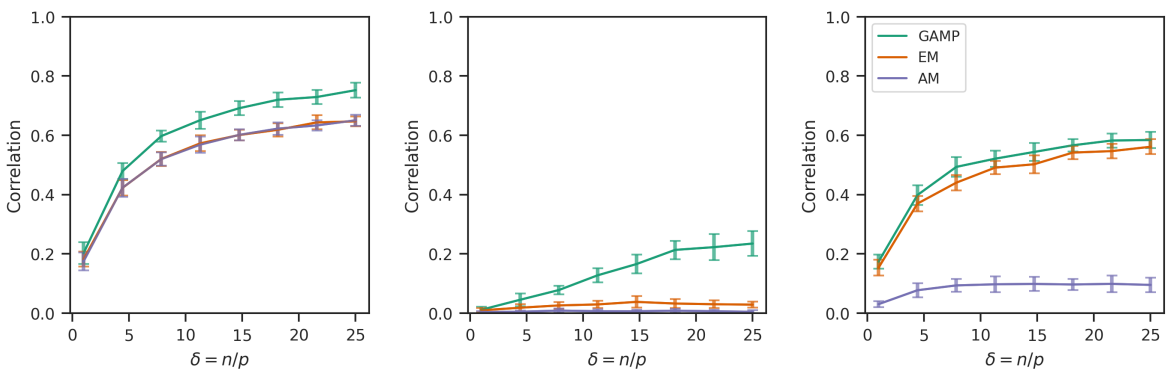
(a) β_1, $\alpha_1 = 0.33$ (b) β_2, $\alpha_2 = 0.33$ (c) β_3, $\alpha_3 = 0.33$

Fig. 18: GAMP for the mixed logistic regression model: Equal proportions, dependent signals



(a) β_1, $\alpha_1 = 0.5$ (b) β_2, $\alpha_2 = 0.2$ (c) β_3, $\alpha_3 = 0.3$

Fig. 19: GAMP for the mixed logistic regression model: Unequal proportions, dependent signals



(a) β_1, $\alpha_1 = 0.5$ (b) β_2, $\alpha_2 = 0.2$ (c) β_3, $\alpha_3 = 0.3$

Fig. 20: Comparison of estimators for the mixed logistic model: Unequal prior and dependent signals, $s = 500$, $p = 500$.

6.4 Comparison of Algorithms for Mixed Logistic Regression

The performance of EM, AM and GAMP is compared in Figure 20. GAMP significantly outperforms both AM and EM in all cases. We can see that AM only successfully estimates β_1, while EM is comparable to GAMP in both β_1 and β_3 but fails to estimate β_2. The superior performance of GAMP is due to the fact that it takes into account the full Gaussian prior over the signals. This allows it to effectively leverage information from all the signals during the estimation process.

6.5 Comparison of Classification Performance

The main application of logistic regression models is classification problems. In this section, we compare the performance of GAMP and EM against a non-linear Support Vector Machine (SVM) classifier. To calculate the prediction error, an additional testing dataset of size n is generated using the original parameters.

Figure 21 shows the classification performance of EM, GAMP and SVM. We can see that even given the optimal parameters, the prediction error is around 25%. For low amounts of data, the SVM classifier is more effective than the signal estimation methods. This is because the algorithm has the relatively easier task of simply obtaining an optimal decision boundary between the classes. However, as δ increases, GAMP and EM achieve strong recovery of the signal matrix B. This allows them to approach the optimal prediction error while the performance of the SVM remains roughly constant.

Therefore, GAMP is viable as a classifier in the mixed logistic model if enough data is available. GAMP also has the advantage of rapid convergence and low algorithmic complexity, meaning that it scales well to large datasets.

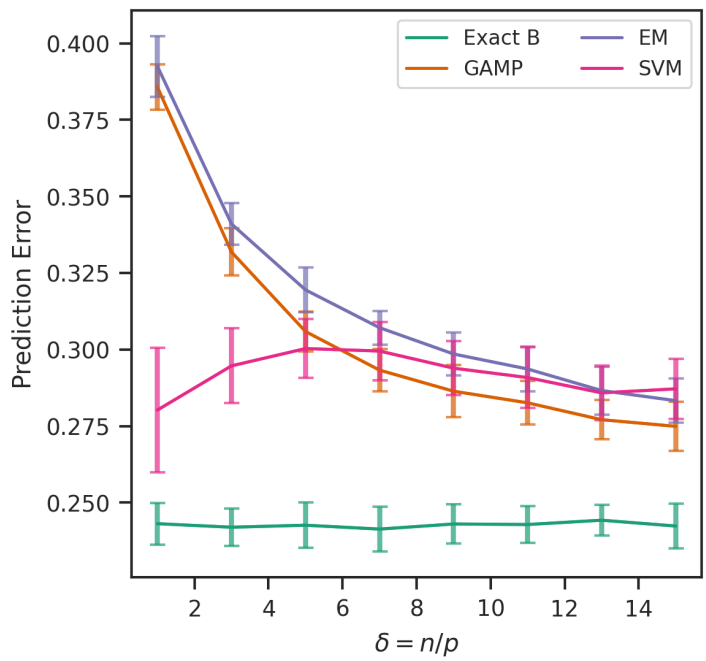


Fig. 21: Comparison of classifiers for the mixed logistic regression model with unequal prior and dependent signals, $s = 500$, $p = 500$

7 Conclusions and Future Work

This project has successfully derived computable expressions for the Bayes-optimal Generalised Approximate Message Passing algorithms in the rectified linear and logistic regression models. The GAMP denoising function was exactly characterized for rectified linear models, while suitable approximations were employed for logistic regression models.

Numerical results have demonstrated the effectiveness of GAMP algorithms as estimators in these models. In mixed regression cases, GAMP outperformed existing estimators like EM and AM by providing strongly correlated estimates for all signals. This is due to GAMP's unique ability to incorporate full prior information and exploit signal dependencies, leading to improved estimation. Additionally, GAMP exhibits robustness to noise, fast convergence, and good computational efficiency, making it an attractive option for signal estimation in heterogeneous populations.

Future research can explore the combination of other priors with rectified linear and logistic regression models, such as using a sparse prior to achieve compressed data representations. Additionally, more sophisticated initialisation methods could be employed to improve the performance of AM and EM for mixed models. Finally, the application of GAMP could be extended to a wide range of other generalised linear models, including Poisson or binomial regression models.

References

[1] Mohsen Bayati and Andrea Montanari. The dynamics of message passing on dense graphs, with applications to compressed sensing. *IEEE Transactions on Information Theory*, 57(2):764–785, 2011.

[2] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

[3] Susana Faria and Gilda Soromenho. Fitting mixtures of linear regressions. *Journal of Statistical Computation and Simulation*, 80(2):201–225, 2010.

[4] Oliver Y Feng, Ramji Venkataramanan, Cynthia Rush, Richard J Samworth, et al. A unifying tutorial on approximate message passing. *Foundations and Trends in Machine Learning*, 15(4):335–536, 2022.

[5] Seyed Mohammadreza Mousavi Kalan, Mahdi Soltanolkotabi, and Amir Salman Avestimehr. Fitting relus via SGD and quantized SGD. *CoRR*, abs/1901.06587, 2019.

[6] Pasin Manurangsi and Daniel Reichman. The computational complexity of training relu(s). 2018.

[7] Andrea Montanari. Graphical models concepts in compressed sensing. *CoRR*, abs/1011.4328, 2010.

[8] D. B. Owen. A table of normal integrals. *Communications in Statistics - Simulation and Computation*, 9(4):389–419, 1980.

[9] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[10] Nelvin Tan and Ramji Venkataramanan. Mixed regression via approximate message passing. 2023.

[11] Xinyang Yi, Constantine Caramanis, and Sujay Sanghavi. Alternating minimization for mixed linear regression. In *International Conference on Machine Learning*, pages 613–621. PMLR, 2014.

Appendix A: Derivation of GAMP for Logistic Regression

In this section, we detail the derivation of the Bayes-optimal GAMP algorithm for the (non-mixed) logistic regression model presented in Section 5. We start by stating the following Gaussian integrals (obtained from [8], equations 10011.3 and 10010.8):

$$\int_{-\infty}^{\infty} \Phi(a+bx)\phi(x)dx = \Phi\left(\frac{a}{\sqrt{1+b^2}}\right) \tag{65}$$

$$\int_{-\infty}^{\infty} x\Phi(a+bx)\phi(x)dx = \frac{b}{\sqrt{1+b^2}}\phi\left(\frac{a}{\sqrt{1+b^2}}\right) \tag{66}$$

where ϕ and Φ are the standard normal pdf and cdf. Additionally, we will require the probit approximation as presented in [2]:

$$\sigma(u) = \frac{1}{1+\exp(u)} \approx \Phi(\lambda u), \quad \lambda = \sqrt{\frac{\pi}{8}}. \tag{67}$$

The distribution $\mathbb{P}\left(Z \mid Z^k, \bar{Y}=1\right)$ can be found using Bayes' rule, giving

$$\mathbb{P}\left(Z \mid Z^k, \bar{Y}=1\right) = \frac{\mathbb{P}\left(\bar{Y}=1 \mid Z, Z^k\right)\mathbb{P}\left(Z \mid Z^k\right)}{\mathbb{P}\left(\bar{Y}=1 \mid Z^k\right)} = \frac{\sigma(Z)\mathcal{N}\left(Z;\mu_1,\sigma_1^2\right)}{\mathbb{P}\left(\bar{Y}=1 \mid Z^k\right)}. \tag{68}$$

The denominator can be obtained from the marginal of $\mathbb{P}\left(\bar{Y}=1, Z \mid Z^k\right)$ using the probit approximation

$$\begin{aligned}
\mathbb{P}\left(\bar{Y}=1 \mid Z^k\right) &= \int_{-\infty}^{\infty} \mathbb{P}(\bar{Y}=1 \mid Z)\,\mathbb{P}\left(Z \mid Z^k\right) dZ && (69)\\
&= \int_{-\infty}^{\infty} \sigma(Z)\,\mathcal{N}\left(Z;\mu_1,\sigma_1^2\right) dZ && (70)\\
&\approx \int_{-\infty}^{\infty} \Phi(\lambda Z)\,\mathcal{N}\left(Z;\mu_1,\sigma_1^2\right) dZ && (71)\\
&= \Phi\left(\frac{\mu_1}{\sqrt{\lambda^{-2}+\sigma_1^2}}\right) && (72)
\end{aligned}$$

where we have applied another standard result from [2] to obtain the result in (72). The required expectation can be obtained by integrating (68):

$$\begin{aligned}
\mathbb{E}\left[Z \mid Z^k, \bar{Y}=1\right] &= \int_{-\infty}^{\infty} Z\frac{\mathbb{P}\left(\bar{Y}=1 \mid Z, Z^k\right)\mathbb{P}\left(Z \mid Z^k\right)}{\mathbb{P}\left(\bar{Y}=1 \mid Z^k\right)} dZ && (73)\\
&= \frac{1}{\mathbb{P}\left(\bar{Y}=1 \mid Z^k\right)} \int_{-\infty}^{\infty} Z\sigma(Z)\mathcal{N}\left(Z;\mu_1,\sigma_1^2\right) dZ && (74)\\
&\approx \frac{1}{\mathbb{P}\left(\bar{Y}=1 \mid Z^k\right)} \int_{-\infty}^{\infty} Z\Phi(\lambda Z)\mathcal{N}\left(Z;\mu_1,\sigma_1^2\right) dZ && (75)\\
&= \frac{1}{\sigma_1\mathbb{P}\left(\bar{Y}=1 \mid Z^k\right)} \int_{-\infty}^{\infty} Z\Phi(\lambda Z)\phi\left(\frac{Z-\mu_1}{\sigma_1}\right) dZ && (76)
\end{aligned}$$

Now we perform a change of variables to get the integral in a standard form. Letting $W = (Z - \mu_1)/\sigma_1$, we have

$$\mathbb{E}\left[Z \mid Z^k, \bar{Y} = 1\right] = \frac{1}{\mathbb{P}\left(\bar{Y} = 1 \mid Z^k\right)} \int_{-\infty}^{\infty} (\sigma_1 W + \mu_1)\, \Phi\left(\lambda\left(\sigma_1 W + \mu_1\right)\right) \phi(W) dW \tag{77}$$

$$= \frac{\frac{\sigma_1^2 \lambda}{\sqrt{1+\sigma_1^2\lambda^2}} \phi\left(\frac{\mu_1 \lambda}{\sqrt{1+\sigma_1^2\lambda^2}}\right) + \mu_1 \Phi\left(\frac{\mu_1\lambda}{\sqrt{1+\sigma_1^2\lambda^2}}\right)}{\phi\left(\frac{\mu_1\lambda}{\sqrt{1+\sigma_1^2\lambda^2}}\right)} \tag{78}$$

where we have used the standard Gaussian integrals in (65) and (66).

The derivation of the case for $\bar{Y} = 0$ can reuse many of the results derived above. In particular, we have

$$\mathbb{E}\left[Z \mid Z^k, \bar{Y} = 0\right] = \int_{-\infty}^{\infty} Z \frac{\mathbb{P}\left(\bar{Y} = 0 \mid Z, Z^k\right) \mathbb{P}\left(Z \mid Z^k\right)}{\mathbb{P}\left(\bar{Y} = 0 \mid Z^k\right)} dZ \tag{79}$$

$$= \frac{1}{\mathbb{P}\left(\bar{Y} = 0 \mid Z^k\right)} \int_{-\infty}^{\infty} Z(1 - \sigma(Z)) \mathcal{N}\left(Z; \mu_1, \sigma_1^2\right) dZ \tag{80}$$

$$= \frac{\mu_1 - \frac{\sigma_1^2\lambda}{\sqrt{1+\sigma_1^2\lambda^2}} \phi\left(\frac{\mu_1\lambda}{\sqrt{1+\sigma_1^2\lambda^2}}\right) - \mu_1 \Phi\left(\frac{\mu_1\lambda}{\sqrt{1+\sigma_1^2\lambda^2}}\right)}{1 - \phi\left(\frac{\mu_1\lambda}{\sqrt{1+\sigma_1^2\lambda^2}}\right)} \tag{81}$$

We now have all the required components to compute the optimal g_k^* needed for the GAMP iteration.

Appendix B: Derivation of GAMP for Mixed Logistic Regression

The derivation of g_k^* for the mixed logistic model follows from the non-mixed case presented in the previous section. However, in this case there is an additional layer of marginalisation required to achieve the result.

We start by defining the multivariate Gaussian distributions of $Z = (Z_1, \ldots, Z_L)^T \in \mathbb{R}^L$ and $Z^k \in \mathbb{R}^L$ in more detail.

$$\begin{bmatrix} Z \\ Z^k \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} \Sigma^{(11)} & \Sigma^{(12)} \\ \Sigma^{(21)} & \Sigma^{(22)} \end{bmatrix}\right) \tag{82}$$

$$\mathbb{P}\left(Z_i \mid Z^k\right) = \mathcal{N}\left(Z_i; \mu_i, \sigma_i^2\right) \tag{83}$$

$$\mathbb{P}\left(Z_j \mid Z_i, Z^k\right) = \mathcal{N}\left(Z_j; \mu_{ji}, \sigma_{ji}^2\right) \tag{84}$$

$$\mu_i = \left[\Sigma^{(12)} \left(\Sigma^{(22)}\right)^{-1} Z^k\right]_i \tag{85}$$

$$\sigma_i^2 = \left[\Sigma^{(11)} - \Sigma^{(12)} \left(\Sigma^{(22)}\right)^{-1} \Sigma^{(21)}\right]_{ii} \tag{86}$$

$$\mu_{ji} = \left[\Sigma_{ji}^{(11)} \Sigma_{j*}^{(12)}\right] \begin{bmatrix} \Sigma_{ii}^{(11)} & \Sigma_{i*}^{(12)} \\ \Sigma_{*i}^{(21)} & \Sigma^{(22)} \end{bmatrix}^{-1} \begin{bmatrix} Z_i \\ Z^k \end{bmatrix} \tag{87}$$

$$\sigma_{ji}^2 = \Sigma_{jj}^{(11)} - \left[\Sigma_{ji}^{(11)} \Sigma_{j*}^{(12)}\right] \begin{bmatrix} \Sigma_{ii}^{(11)} & \Sigma_{i*}^{(12)} \\ \Sigma_{*i}^{(21)} & \Sigma^{(22)} \end{bmatrix}^{-1} \begin{bmatrix} \Sigma_{ij}^{(11)} \\ \Sigma_{*j}^{(21)} \end{bmatrix} \tag{88}$$

Note that $\mu_{jj} = Z_j$ and $\sigma_{jj}^2 = 0$, so $\mathbb{P}(Z_j = z \mid Z_j, Z^k) = \delta_{Z_j}(z)$. The expectation $\mathbb{E}\left[Z \mid Z^k, \bar{Y}\right]$ can be decomposed with respect to $\bar{c}$

$$\mathbb{E}\left[Z \mid Z^k, \bar{Y}\right] = \sum_{j=1}^{L} \mathbb{P}\left(\bar{c} = j \mid Z^k, \bar{Y}\right) \begin{bmatrix} \mathbb{E}\left[Z_1 \mid Z^k, \bar{Y}, \bar{c} = j\right] \\ \vdots \\ \mathbb{E}\left[Z_L \mid Z^k, \bar{Y}, \bar{c} = j\right] \end{bmatrix} \tag{89}$$

The distribution of $\mathbb{P}(Z_i \mid \bar{Y} = 1, Z^k, \bar{c})$ can be decomposed using Bayes' rule

$$\mathbb{P}\left(Z_i \mid \bar{Y} = 1, Z^k, \bar{c} = j\right) = \frac{\mathbb{P}\left(\bar{Y} = 1 \mid Z_i, Z^k, \bar{c} = j\right) \mathbb{P}\left(Z_i \mid Z^k\right)}{\mathbb{P}\left(\bar{Y} = 1 \mid Z^k, \bar{c} = j\right)} \tag{90}$$

where the normalisation constant follows similarly to (72) using the probit approximation.

$$\mathbb{P}\left(\bar{Y} = 1 \mid Z^k, \bar{c} = j\right) = \int_{-\infty}^{\infty} \mathbb{P}\left(\bar{Y} = 1 \mid Z_j, \bar{c} = j\right) \mathbb{P}\left(Z_j \mid Z^k\right) dZ_j \tag{91}$$

$$= \int_{-\infty}^{\infty} \sigma\left(Z_j\right) \mathcal{N}\left(Z_j; \mu_j, \sigma_j^2\right) dZ_j \tag{92}$$

$$\approx \Phi\left(\frac{\mu_j}{\sqrt{\lambda^{-2} + \sigma_j^2}}\right) \tag{93}$$

In order to obtain $\mathbb{P}\left(\bar{Y} = 1 \mid Z_i, Z^k, \bar{c} = j\right)$, we marginalise the joint distribution

$$\mathbb{P}\left(\bar{Y} = 1 \mid Z_i, Z^k, \bar{c} = j\right) = \int_{-\infty}^{\infty} \mathbb{P}\left(\bar{Y} = 1 \mid Z_j\right) \mathbb{P}\left(Z_j \mid Z_i, Z^k\right) dZ_j \tag{94}$$

$$= \int_{-\infty}^{\infty} \sigma\left(Z_j\right) \mathcal{N}\left(Z_j; \mu_{ji}, \sigma_{ji}^2\right) dZ_j \tag{95}$$

$$\approx \Phi\left(\frac{\mu_{ji}}{\sqrt{\lambda^{-2} + \sigma_{ji}^2}}\right). \tag{96}$$

In the case of $i = j$, this becomes $\mathbb{P}\left(\bar{Y} = 1 \mid Z_i, Z^k, \bar{c} = i\right) \approx \Phi(\lambda Z_i) \approx \sigma(Z_i)$ as expected. Now we integrate (90) to obtain the expectation

$$\mathbb{E}\left[Z_i \mid \bar{Y} = 1, Z^k, \bar{c} = j\right] = \int_{-\infty}^{\infty} Z_i \mathbb{P}\left(Z_i \mid \bar{Y} = 1, Z^k, \bar{c} = j\right) dZ_i \tag{97}$$

$$= \frac{1}{\mathbb{P}\left(\bar{Y} = 1 \mid Z^k, \bar{c} = j\right)} \int_{-\infty}^{\infty} Z_i \mathbb{P}\left(\bar{Y} = 1 \mid Z_i, Z^k, \bar{c} = j\right) \mathbb{P}\left(Z_i \mid Z^k\right) dZ_i \tag{98}$$

$$\approx \frac{1}{\mathbb{P}\left(\bar{Y} = 1 \mid Z^k, \bar{c} = j\right)} \int_{-\infty}^{\infty} Z_i \Phi\left(\frac{\mu_{ji}}{\sqrt{\lambda^{-2} + \sigma_{ji}^2}}\right) \mathcal{N}\left(Z_i; \mu_i, \sigma_i^2\right) dZ_i \tag{99}$$

As in the non-mixed case, we can perform this integral using (65) and (66). Letting

$\mu_{ji}\left(\sqrt{\lambda^{-2}+\sigma_{ji}^2}\right)^{-1} = \alpha_{ji}Z_i + \gamma_{ji}$ and substituting $W = \frac{Z_i-\mu_i}{\sigma_i}$, we have

$$\mathbb{E}\left[\cdot\right] \approx \frac{1}{\mathbb{P}\left(\bar{Y}=1 \mid Z^k, \bar{c}=j\right)} \int_{-\infty}^{\infty} Z_i \Phi\left(\alpha_{ji}Z_i + \gamma_{ji}\right) \mathcal{N}\left(Z_i; \mu_i, \sigma_i^2\right) dZ_i \tag{100}$$

$$= \frac{1}{\mathbb{P}\left(\bar{Y}=1 \mid Z^k, \bar{c}=j\right)} \int_{-\infty}^{\infty} \left(\sigma_i W + \mu_i\right) \Phi\left(\alpha_{ji}\sigma_i W + \alpha_{ji}\mu_i + \gamma_{ji}\right) \phi(W) dW \tag{101}$$

$$= \frac{\frac{\alpha_{ji}\sigma_i^2}{\sqrt{1+\alpha_{ji}^2\sigma_i^2}}\phi\left(\frac{\alpha_{ji}\mu_i+\gamma_{ji}}{\sqrt{1+\alpha_{ji}^2\sigma_i^2}}\right) + \mu_i\Phi\left(\frac{\alpha_{ji}\mu_i+\gamma_{ji}}{\sqrt{1+\alpha_{ji}^2\sigma_i^2}}\right)}{\Phi\left(\frac{\mu_j}{\sqrt{\lambda^{-2}+\sigma_j^2}}\right)}. \tag{102}$$

The case of $\mathbb{E}\left[Z_i \mid \bar{Y}=0, Z^k, \bar{c}=j\right]$ can be found using the same steps and takes the form shown in (60). The only remaining part of (89) is $\mathbb{P}\left(\bar{c}=j \mid Z^k, \bar{Y}\right)$ which can be found using Bayes' rule

$$\mathbb{P}\left(\bar{c}=j \mid Z^k, \bar{Y}\right) = \frac{\mathbb{P}(\bar{c}=j)\mathbb{P}\left(\bar{Y} \mid Z^k, \bar{c}=j\right)\mathbb{P}\left(Z^k\right)}{\sum_{l=1}^{L}\mathbb{P}(\bar{c}=l)\mathbb{P}\left(\bar{Y} \mid Z^k, \bar{c}=j\right)\mathbb{P}\left(Z^k\right)} \tag{103}$$

$$= \frac{\alpha_j\mathbb{P}\left(\bar{Y} \mid Z^k, \bar{c}=j\right)}{\sum_{l=1}^{L}\alpha_l\mathbb{P}\left(\bar{Y} \mid Z^k, \bar{c}=l\right)} \tag{104}$$

where $\mathbb{P}\left(\bar{Y} \mid Z^k, \bar{c}=j\right)$ is given in (93).

Now we have all the expressions required to substitute back into (89). This should allow us to compute g_k^* and therefore perform the GAMP recursion to estimate the signal matrix for the mixed logistic model.

Appendix C: Derivation of GAMP for Rectified Linear Regression

To perform the Bayes-optimal GAMP recursion, we require an expression for g_k^* in (5). The expectation needed for the definition of g_k^* can be decomposed with respect to the linear and rectified regions

$$\mathbb{E}\left[Z \mid Z^k, \bar{Y}\right] = \underbrace{\mathbb{E}\left[Z \mid Z^k, \bar{Y}, \omega=0\right]}_{\text{Part (i)}}\underbrace{\mathbb{P}\left(\omega=0 \mid Z^k, \bar{Y}\right)}_{\text{Part (iii)}} \tag{105}$$

$$+\underbrace{\mathbb{E}\left[Z \mid Z^k, \bar{Y}, \omega=1\right]}_{\text{Part (ii)}}\underbrace{\mathbb{P}\left(\omega=1 \mid Z^k, \bar{Y}\right)}_{\text{Part (iii)}} \tag{106}$$

where the relevant random variables are defined as

$$\bar{Y} = \max(0, Z) + \bar{\epsilon}, \quad \bar{\epsilon} \sim \mathcal{N}\left(0, \sigma_\epsilon^2\right) \tag{107}$$

$$\omega = \begin{cases} 1 & \text{if } Z > 0 \\ 0 & \text{if } Z \leq 0 \end{cases} \tag{108}$$

$$\begin{bmatrix} Z \\ Z^k \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} \sigma_Z^2 & \rho \\ \rho & \sigma_{Z^k}^2 \end{bmatrix}\right) \tag{109}$$

with the following conditional distributions

$$\mathbb{P}(\bar{Y} \mid Z) = \begin{cases} \mathcal{N}\left(\bar{Y}; Z, \sigma_\epsilon^2\right) & \text{if } Z > 0 \\ \mathcal{N}\left(\bar{Y}; 0, \sigma_\epsilon^2\right) & \text{if } Z < 0 \end{cases} \tag{110}$$

$$\mathbb{P}\left(Z \mid Z^k\right) = \mathcal{N}\left(Z; \frac{\rho}{\sigma_{Z^k}^2} Z^k, \sigma_Z^2 - \frac{\rho^2}{\sigma_{Z^k}^2}\right) = \mathcal{N}\left(Z; \mu_1, \sigma_1^2\right) \tag{111}$$

Truncated Gaussian Distributions

The derivation of the g_k^* for the rectified model requires the use of the truncated Gaussian distribution, defined as

$$X^+ \sim \mathcal{N}^+\left(\mu, \sigma^2\right), \qquad \mathbb{P}(X^+) = \mathbb{I}(X^+ > 0)\frac{\mathcal{N}\left(X^+; \mu, \sigma^2\right)}{1 - \Phi\left(0; \mu, \sigma^2\right)} \tag{112}$$

$$X^- \sim \mathcal{N}^-\left(\mu, \sigma^2\right), \qquad \mathbb{P}(X^-) = \mathbb{I}(X^- < 0)\frac{\mathcal{N}\left(X^-; \mu, \sigma^2\right)}{\Phi\left(0; \mu, \sigma^2\right)} \tag{113}$$

$$\mathbb{E}\left[X^+\right] = \mu + \sigma^2 \frac{\mathcal{N}\left(0; \mu, \sigma^2\right)}{1 - \Phi\left(0; \mu, \sigma^2\right)}, \qquad \mathbb{E}\left[X^-\right] = \mu - \sigma^2 \frac{\mathcal{N}\left(0; \mu, \sigma^2\right)}{\Phi\left(0; \mu, \sigma^2\right)} \tag{114}$$

where $\Phi(a; \mu, \sigma^2)$ denotes the Gaussian cdf with mean μ and variance σ^2 and $\mathbb{I}()$ is the indicator function.

The following identities for the product of Gaussian and truncated Gaussians will also be required

$$\mathcal{N}(X; W, a^2)\mathcal{N}^+(W; b, c^2) = \frac{\mathbb{I}(W > 0)\mathcal{N}\left(\begin{bmatrix} X \\ W \end{bmatrix}; 0, \begin{bmatrix} a^2 + c^2 & c^2 \\ c^2 & c^2 \end{bmatrix}\right) \exp\left(-\frac{b^2}{2c^2} + \frac{bW}{c^2}\right)}{1 - \Phi(0; b, c^2)} \tag{115}$$

$$\int_{-\infty}^{\infty} \mathcal{N}(X; W, a^2)\mathcal{N}^+(W; b, c^2) dW = \frac{\exp\left(-\frac{X^2}{2a^2} - \frac{b^2}{2c^2} + \frac{d^2}{2e^2}\right)\left(1 - \Phi\left(0; d, e^2\right)\right)}{\sqrt{2\pi(a^2 + c^2)}(1 - \Phi(0; b, c^2))} \tag{116}$$

where

$$d = \frac{Xc^2 + ba^2}{a^2 + c^2}, \qquad e^2 = \frac{a^2 c^2}{a^2 + c^2} \tag{117}$$

Part (i): $\mathbb{E}\left[Z \mid Z^k, \bar{Y}, \omega = 0\right]$

When $\omega = 0$, Z and $\bar{Y}$ are conditionally independent, so the distribution simply reduces to a truncated Gaussian distribution

$$\mathbb{P}\left(Z \mid Z^k, \bar{Y}, \omega = 0\right) = \mathbb{P}\left(Z \mid Z^k, \omega = 0\right) = \mathcal{N}^-\left(Z; \mu_1, \sigma_1^2\right) \tag{118}$$

$$\mathbb{E}\left[Z \mid Z^k, \bar{Y}, \omega = 0\right] = \mu_1 - \sigma_1^2 \frac{\mathcal{N}\left(0; \mu_1, \sigma_1^2\right)}{\Phi\left(0; \mu_1, \sigma_1^2\right)} \tag{119}$$

where we have used the mean of a truncated Gaussian distribution from (114).

Part (ii): $\mathbb{E}\left[Z \mid Z^k, \bar{Y}, \omega = 1\right]$

In order to calculate part (ii) of (106), we first require the joint distribution

$$\mathbb{P}\left(\bar{Y}, Z \mid Z^k, \omega = 1\right) = \mathbb{P}\left(\bar{Y} \mid Z, Z^k, \omega = 1\right) \mathbb{P}\left(Z \mid Z^k, \omega = 1\right) \tag{120}$$

$$= \mathcal{N}\left(\bar{Y}; Z, \sigma_\epsilon^2\right) \mathcal{N}^+\left(Z; \mu_1, \sigma_1^2\right). \tag{121}$$

Using the identity in (115), the distribution can be expressed as

$$\mathbb{P}\left(\bar{Y}, Z \mid Z^k, \omega = 1\right) = \frac{\mathbb{I}(Z > 0)}{C_1} \mathcal{N}\left([\bar{Y}, Z]^T; 0, \Sigma\right) \exp\left(\mu_1 \sigma_1^{-2} Z\right) \tag{122}$$

where

$$C_1 = \exp\left(\frac{\mu_1^2}{2\sigma_1^2}\right)\left(1 - \Phi\left(0; \mu_1, \sigma_1^2\right)\right), \qquad \Sigma = \begin{bmatrix} \sigma_\epsilon^2 + \sigma_1^2 & \sigma_1^2 \\ \sigma_1^2 & \sigma_1^2 \end{bmatrix} \tag{123}$$

We can see that $\bar{Y}$ and Z form a multivariate Gaussian, but with an extra term in Z that cannot be included.

Now we can obtain a simple truncated Gaussian expression for our desired distribution

$$\mathbb{P}\left(Z \mid Z^k, \bar{Y}, \omega = 1\right) = \frac{\mathbb{P}\left(Z, \bar{Y} \mid Z^k, \omega = 1\right)}{\int_{-\infty}^{\infty} \mathbb{P}\left(Z, \bar{Y} \mid Z^k, \omega = 1\right) dZ} \tag{124}$$

$$= \frac{C_1^{-1} \mathbb{I}(Z > 0) \mathcal{N}\left([\bar{Y}, Z]^T; 0, \Sigma\right) \exp\left(\mu_1 \sigma_1^{-2} Z\right)}{C_1^{-1} \int_0^{\infty} \mathcal{N}\left([\bar{Y}, Z]^T; 0, \Sigma\right) \exp\left(\mu_1 \sigma_1^{-2} Z\right) dZ} \tag{125}$$

$$= \mathcal{N}^+\left(Z; \mu_2, \sigma_2^2\right) \tag{126}$$

where

$$\mu_2 = \frac{\bar{Y}\sigma_1^2 + \mu_1 \sigma_\epsilon^2}{\sigma_\epsilon^2 + \sigma_1^2}, \qquad \sigma_2^2 = \frac{\sigma_\epsilon^2 \sigma_1^2}{\sigma_\epsilon^2 + \sigma_1^2} \tag{127}$$

Now taking the expectation using (114) gives

$$\mathbb{E}\left[Z \mid Z^k, \bar{Y}, \omega = 1\right] = \mu_2 + \sigma_2^2 \frac{\mathcal{N}\left(0; \mu_2, \sigma_2^2\right)}{1 - \Phi\left(0; \mu_2, \sigma_2^2\right)} \tag{128}$$

Part (iii): $\mathbb{P}\left[\omega \mid Z^k, \bar{Y},\right]$

Using Bayes' rule, the distribution of $\mathbb{P}\left(\omega = 0 \mid Z^k, \bar{Y}\right)$ is given by:

$$\mathbb{P}\left(\omega = 0 \mid Z^k, \bar{Y}\right) = \frac{\mathbb{P}\left(\bar{Y} \mid Z^k, \omega = 0\right) \mathbb{P}\left(\omega = 0 \mid Z^k\right)}{\mathbb{P}\left(\bar{Y} \mid Z^k, \omega = 0\right) \mathbb{P}\left(\omega = 0 \mid Z^k\right) + \mathbb{P}\left(\bar{Y} \mid Z^k, \omega = 1\right) \mathbb{P}\left(\omega = 1 \mid Z^k\right)} \tag{129}$$

$$= \frac{\mathcal{N}\left(\bar{Y}; 0, \sigma_\epsilon^2\right) \Phi\left(0; \mu_1, \sigma_1^2\right)}{\mathcal{N}\left(\bar{Y}; 0, \sigma_\epsilon^2\right) \Phi\left(0; \mu_1, \sigma_1^2\right) + \mathbb{P}\left(\bar{Y} \mid Z^k, \omega = 1\right)\left(1 - \Phi\left(0; \mu_1, \sigma_1^2\right)\right)} \tag{130}$$

where straightforward distributions have been substituted in.

The distribution $\mathbb{P}\left(\bar{Y} \mid Z^k, \omega = 1\right)$ can be obtained by integrating the joint distribution (122) using the identity in (116)

$$\mathbb{P}\left(\bar{Y} \mid Z^k, \omega = 1\right) = \int_{-\infty}^{\infty} \mathbb{P}\left(\bar{Y}, Z \mid Z^k, \omega = 1\right) dZ \tag{131}$$

$$= \frac{1}{C_1 \sqrt{2\pi(\sigma_\epsilon^2 + \sigma_1^2)}} \exp\left(-\frac{\bar{Y}^2}{2\sigma_\epsilon^2} + \frac{\mu_2^2}{2\sigma_2^2}\right)\left(1 - \Phi\left(0; \mu_2, \sigma_2^2\right)\right) \tag{132}$$

where μ_2, σ_2^2 and C_1 are defined as above. Note that this is not a simple Gaussian distribution since μ_2 has a linear dependence on $\bar{Y}$.

The final part of (106) can is simply $\mathbb{P}\left(\omega = 1 \mid Z^k, \bar{Y}\right) = 1 - \mathbb{P}\left(\omega = 0 \mid Z^k, \bar{Y}\right)$. Therefore, we have obtained a computable expression for all of the components required for g_k^*, which can be used to perform the optimal GAMP recursion.

Appendix D: Derivation of GAMP for Mixed Rectified Linear Regression

The derivation of GAMP for mixed rectified linear regression case combines techniques used in the previous two sections. First, we decompose the expectation $\mathbb{E}\left[Z \mid Z^k, \bar{Y}\right]$ as

$$\mathbb{E}\left[Z \mid Z^k, \bar{Y}\right] = \sum_{j=1}^{L} \overbrace{\mathbb{P}\left(\bar{c} = j \mid Z^k, \bar{Y}\right)}^{\text{Part (i)}} \begin{bmatrix} \mathbb{E}\left[Z_1 \mid Z^k, \bar{Y}, \bar{c} = j\right] \\ \vdots \\ \mathbb{E}\left[Z_L \mid Z^k, \bar{Y}, \bar{c} = j\right] \end{bmatrix} \tag{133}$$

$$\begin{aligned} \mathbb{E}\left[Z_i \mid Z^k, \bar{Y}, \bar{c} = j\right] = & \overbrace{\mathbb{P}\left[\omega_j = 0 \mid Z^k, \bar{Y}, \bar{c} = j\right]}^{\text{Part (ii)}} \overbrace{\mathbb{E}\left[Z_i \mid Z^k, \bar{Y}, \bar{c} = j, \omega_j = 0\right]}^{\text{Part (iii)}} \\ & + \underbrace{\mathbb{P}\left[\omega_j = 1 \mid Z^k, \bar{Y}, \bar{c} = j\right]}_{\text{Part (ii)}} \underbrace{\mathbb{E}\left[Z_i \mid Z^k, \bar{Y}, \bar{c} = j, \omega_j = 1\right]}_{\text{Part (iv)}} \end{aligned} \tag{134}$$

where

$$\bar{Y} = \max\left(0, \sum_{j=1}^{L} \mathbb{I}(\bar{c} = j) Z_j\right) + \bar{\epsilon} \tag{135}$$

$$\bar{c} \sim \text{cat}(\alpha), \quad \bar{\epsilon} \sim \mathcal{N}(0, \sigma_\epsilon^2), \quad \omega_j = \mathbb{I}(Z_j > 0) \tag{136}$$

$$\begin{bmatrix} Z \\ Z^k \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} \Sigma^{(11)} & \Sigma^{(12)} \\ \Sigma^{(21)} & \Sigma^{(22)} \end{bmatrix}\right) \tag{137}$$

with conditional distributions

$$\mathbb{P}(\bar{Y} \mid Z_j, \bar{c} = j) = \begin{cases} \mathcal{N}\left(\bar{Y}; Z_j, \sigma_\epsilon^2\right) & \text{if } Z_j > 0 \\ \mathcal{N}\left(\bar{Y}; 0, \sigma_\epsilon^2\right) & \text{if } Z_j < 0 \end{cases} \tag{138}$$

$$\mathbb{P}\left(Z_i \mid Z^k\right) = \mathcal{N}\left(Z_i; \mu_i, \sigma_i^2\right) \tag{139}$$

$$\mathbb{P}\left(Z_j \mid Z_i, Z^k\right) = \mathcal{N}\left(Z_j; \mu_{ji}, \sigma_{ji}^2\right) \tag{140}$$

$$\mu_{ji} = a_{ji} Z_i + b_{ji} \tag{141}$$

In the following sections, we derive expressions for all of the parts required in (134).

Part (i): $\mathbb{P}(\bar{c} = j \mid Z^k, \bar{Y})$

The distribution can be decomposed using Bayes' rule

$$\mathbb{P}(\bar{c} = j \mid Z^k, \bar{Y}) = \frac{\mathbb{P}(\bar{c} = j)\mathbb{P}\left(\bar{Y} \mid Z^k, \bar{c} = j\right)\mathbb{P}\left(Z^k\right)}{\sum_{l=1}^{L} \mathbb{P}(\bar{c} = l)\mathbb{P}\left(\bar{Y} \mid Z^k, \bar{c} = l\right)\mathbb{P}\left(Z^k\right)} \tag{142}$$

$$= \frac{\alpha_j \mathbb{P}\left(\bar{Y} \mid Z^k, \bar{c} = j\right)}{\sum_{l=1}^{L} \alpha_l \mathbb{P}\left(\bar{Y} \mid Z^k, \bar{c} = l\right)} \tag{143}$$

and $\mathbb{P}\left(\bar{Y} \mid Z^k, \bar{c} = j\right)$ can be further broken down as

$$\begin{aligned}\mathbb{P}\left(\bar{Y} \mid Z^k, \bar{c} = j\right) &= \mathbb{P}\left(\bar{Y} \mid \omega_j = 0, Z^k, \bar{c} = j\right) \mathbb{P}\left(\omega_j = 0 \mid Z^k, \bar{c} = j\right) \\ &+ \mathbb{P}\left(\bar{Y} \mid \omega_j = 1, Z^k, \bar{c} = j\right) \mathbb{P}\left(\omega_j = 1 \mid Z^k, \bar{c} = j\right) \\ &= \mathcal{N}(\bar{Y}; 0, \sigma_\epsilon^2)\Phi(0; \mu_j, \sigma_j^2) + \mathbb{P}\left(\bar{Y} \mid \omega_j = 1, Z^k, \bar{c} = j\right)\left(1 - \Phi(0; \mu_j, \sigma_j^2)\right)\end{aligned} \tag{144}$$

The calculation of $\mathbb{P}\left(\bar{Y} \mid \omega_j = 1, Z^k, \bar{c} = j\right)$ requires the marginalisation of the joint distribution $\mathbb{P}\left(\bar{Y}, Z_j \mid \omega_j = 1, Z^k, \bar{c} = j\right)$. For this, we can re-use the result of the integral of the product of Gaussian and truncated Gaussian densities from (116)

$$\mathbb{P}\left(\bar{Y} \mid Z^k, \omega_j = 1, \bar{c} = j\right) = \int \mathbb{P}\left(\bar{Y}, Z_j \mid Z^k, \omega_j = 1, \bar{c} = j\right) dZ_j \tag{145}$$

$$= \int \mathcal{N}(\bar{Y}; Z_j, \sigma_\epsilon^2)\mathcal{N}^+(Z_j; \mu_j, \sigma_j^2) dZ_j \tag{146}$$

$$= \frac{\exp\left(-\frac{\bar{Y}^2}{2\sigma_\epsilon^2} - \frac{\mu_j^2}{2\sigma_j^2} + \frac{m_j^2}{2s_j^2}\right)\left(1 - \Phi(0; m_j, s_j^2)\right)}{\sqrt{2\pi(\sigma_\epsilon^2 + \sigma_j^2)}\left(1 - \Phi(0; \mu_j, \sigma_j^2)\right)} \tag{147}$$

$$= \frac{1}{C_j} \exp\left(-\frac{\bar{Y}^2}{2\sigma_\epsilon^2} + \frac{m_j^2}{2s_j^2}\right)\left(1 - \Phi(0; m_j, s_j^2)\right) \tag{148}$$

where

$$m_j = \frac{\bar{Y}\sigma_j^2 + \mu_j \sigma_\epsilon^2}{\sigma_\epsilon^2 + \sigma_j^2}, \qquad s_j^2 = \frac{\sigma_j^2 \sigma_\epsilon^2}{\sigma_\epsilon^2 + \sigma_j^2}, \tag{149}$$

$$C_j = \exp\left(\frac{\mu_j^2}{2\sigma_j^2}\right)\sqrt{2\pi(\sigma_\epsilon^2 + \sigma_j^2)}\left(1 - \Phi(0; \mu_j, \sigma_j^2)\right) \tag{150}$$

Part (ii): $\mathbb{P}\left[\omega_j \mid Z^k, \bar{Y}, \bar{c} = j\right]$

Using Bayes' rule, we have

$$\mathbb{P}[\omega_j = 0 \mid Z^k, \bar{Y}, \bar{c} = j] = \frac{\mathbb{P}\left(\bar{Y}, \mid \omega_j = 0, Z^k, \bar{c} = j\right) \mathbb{P}\left(\omega_j = 0 \mid Z^k, \bar{c} = j\right)}{\mathbb{P}\left(\bar{Y} \mid Z^k, \bar{c} = j\right)} \tag{151}$$

$$= \frac{\mathcal{N}(\bar{Y}; 0, \sigma_\epsilon^2)\Phi(0; \mu_j, \sigma_j^2)}{\mathbb{P}\left(\bar{Y} \mid Z^k, \bar{c} = j\right)} \tag{152}$$

and $\mathbb{P}\left(\bar{Y} \mid Z^k, \bar{c} = j\right)$ can be substituted from (144).

Part (iii): $\mathbb{E}\left[Z_i \mid Z^k, \bar{Y}, \bar{c} = j, \omega_j = 0\right]$

Since $\bar{c} = j$ and $\omega_j = 0$, we can ignore the effect of $\bar{Y}$. This gives a simplified expression for the density $\mathbb{P}\left[Z_i \mid Z^k, \bar{Y}, \bar{c} = j, \omega_j = 0\right]$

$$\mathbb{P}\left[Z_i \mid Z^k, \omega_j = 0\right] = \int \mathbb{P}\left[Z_i, Z_j \mid Z^k, \omega_j = 0\right] dZ_j \tag{153}$$

$$= \int \frac{\mathbb{I}(Z_j < 0)\mathbb{P}(Z_i, Z_j \mid Z^k)}{\Phi(0; \mu_j, \sigma_j^2)} dZ_j \tag{154}$$

$$= \int_{-\infty}^{0} \frac{\mathcal{N}(Z_j; \mu_{ji}, \sigma_{ji}^2)\mathcal{N}(Z_i; \mu_i, \sigma_i^2)}{\Phi(0; \mu_j, \sigma_j^2)} dZ_j \tag{155}$$

$$= \frac{\Phi(0; \mu_{ji}, \sigma_{ji}^2)\mathcal{N}(Z_i; \mu_i, \sigma_i^2)}{\Phi(0; \mu_j, \sigma_j^2)} \tag{156}$$

To calculate the expectation, we can integrate this using the Gaussian integrals in (65) and (66)

$$\mathbb{E}\left[Z_i \mid Z^k, \omega_j = 0\right] = \int Z_i \mathbb{P}\left[Z_i \mid Z^k, \omega_j = 0\right] dZ_i \tag{157}$$

$$= \int Z_i \frac{\Phi(0; \mu_{ji}, \sigma_{ji}^2)\mathcal{N}(Z_i; \mu_i, \sigma_i^2)}{\Phi(0; \mu_j, \sigma_j^2)} dZ_i \tag{158}$$

$$= \int Z_i \frac{\Phi(0; a_{ji}Z_i + b_{ji}, \sigma_{ji}^2)\mathcal{N}(Z_i; \mu_i, \sigma_i^2)}{\Phi(0; \mu_j, \sigma_j^2)} dZ_i \tag{159}$$

$$= \int Z_i \frac{\Phi\left(\frac{-a_{ji}Z_i - b_{ji}}{\sigma_{ji}}\right)\phi\left(\frac{Z_i - \mu_i}{\sigma_i}\right)}{\sigma_i \Phi(0; \mu_j, \sigma_j^2)} dZ_i \tag{160}$$

$$= \int (\sigma_i W + \mu_i) \frac{\Phi\left(\frac{-a_{ji}\sigma_i W - a_{ji}\mu_i - b_{ji}}{\sigma_{ji}}\right)\phi(W)}{\Phi(0; \mu_j, \sigma_j^2)} dW \tag{161}$$

$$= \frac{1}{\Phi(0; \mu_j, \sigma_j^2)} \left[\frac{-\sigma_i^2 a_{ji}}{\sqrt{\sigma_{ji}^2 + a_{ji}^2\sigma_i^2}} \phi\left(\frac{-a_{ji}\mu_i - b_{ji}}{\sqrt{\sigma_{ji}^2 + a_{ji}^2\sigma_i^2}}\right) + \mu_i \Phi\left(\frac{-a_{ji}\mu_i - b_{ji}}{\sqrt{\sigma_{ji}^2 + a_{ji}^2\sigma_i^2}}\right) \right] \tag{162}$$

When $i = j$, we recover the non-mixed case in (119) since $a_{ji} = 1, b_{ji} = 0$ and $\sigma_{ji}^2 = 0$.

Part (iv): $\mathbb{E}\left[Z_i \mid Z^k, \bar{Y}, \bar{c} = j, \omega_j = 1\right]$

In this case, the dependencies are more complicated since Z_i depends on $\bar{Y}$ through Z_j. Therefore, we start by computing $\mathbb{P}\left[Z_i, Z_j \mid Z^k, \bar{Y}, \bar{c} = j, \omega_j = 1\right]$ and later marginalise out Z_j

$$\begin{aligned}
\mathbb{P}\left[Z_i, Z_j \mid Z^k, \bar{Y}, \bar{c} = j, \omega_j = 1\right] &= \frac{\mathbb{P}\left(\bar{Y} \mid Z_j, Z_i, Z^k, \omega_j = 1, \bar{c} = j\right)\mathbb{P}\left(Z_i, Z_j \mid Z^k, \omega_j = 1, \bar{c} = j\right)}{\mathbb{P}\left(\bar{Y} \mid Z^k, \omega_j = 1, \bar{c} = j\right)} \\
&= \frac{\mathcal{N}(\bar{Y}; Z_j, \sigma_\epsilon^2)\mathbb{I}(Z_j > 0)\mathcal{N}(Z_j; \mu_{ji}, \sigma_{ji}^2)\mathcal{N}(Z_i; \mu_i, \sigma_i^2)}{\mathbb{P}\left(\bar{Y} \mid Z^k, \omega_j = 1, \bar{c} = j\right)\left(1 - \Phi(0; \mu_j, \sigma_j^2)\right)} \\
&= \frac{\mathcal{N}(\bar{Y}; Z_j, \sigma_\epsilon^2)\mathcal{N}^+(Z_j; \mu_{ji}, \sigma_{ji}^2)\mathcal{N}(Z_i; \mu_i, \sigma_i^2)\left(1 - \Phi(0; \mu_{ji}, \sigma_{ji}^2)\right)}{\mathbb{P}\left(\bar{Y} \mid Z^k, \omega_j = 1, \bar{c} = j\right)\left(1 - \Phi(0; \mu_j, \sigma_j^2)\right)}
\end{aligned} \tag{163}$$

where we have used the fact that $\mathcal{N}(x; \mu, \sigma^2)\mathbb{I}(x > 0) = (1 - \Phi(x; \mu, \sigma^2))\mathcal{N}^+(x; \mu, \sigma^2)$. The marginalisation integral is in the form of (116), giving

$$\mathbb{P}\left[Z_i \mid Z^k, \bar{Y}, \bar{c} = j, \omega_j = 1\right] = \int \mathbb{P}\left[Z_i, Z_j \mid Z^k, \bar{Y}, \bar{c} = j, \omega_j = 1\right] dZ_j \tag{164}$$

$$= \int \frac{\mathcal{N}(\bar{Y}; Z_j, \sigma_\epsilon^2)\mathcal{N}^+(Z_j; \mu_{ji}, \sigma_{ji}^2)\mathcal{N}(Z_i; \mu_i, \sigma_i^2)\left(1 - \Phi(0; \mu_{ji}, \sigma_{ji}^2)\right)}{\mathbb{P}\left(\bar{Y} \mid Z^k, \omega_j = 1, \bar{c} = j\right)\left(1 - \Phi(0; \mu_j, \sigma_j^2)\right)} dZ_j \tag{165}$$

$$= \frac{\mathcal{N}(Z_i; \mu_i, \sigma_i^2)\left(1 - \Phi(0; \mu_{ji}, \sigma_{ji}^2)\right)}{\mathbb{P}\left(\bar{Y} \mid Z^k, \omega_j = 1, \bar{c} = j\right)\left(1 - \Phi(0; \mu_j, \sigma_j^2)\right)} \int \mathcal{N}(\bar{Y}; Z_j, \sigma_\epsilon^2)\mathcal{N}^+(Z_j; \mu_{ji}, \sigma_{ji}^2) dZ_j \tag{166}$$

$$= \frac{\mathcal{N}(Z_i; \mu_i, \sigma_i^2)\left(1 - \Phi(0; \mu_{ji}, \sigma_{ji}^2)\right)\exp\left(-\frac{\bar{Y}^2}{2\sigma_\epsilon^2} - \frac{\mu_{ji}^2}{2\sigma_{ji}^2} + \frac{m_{ji}^2}{2s_{ji}^2}\right)\left(1 - \Phi(0; m_{ji}, s_{ji}^2)\right)}{\mathbb{P}\left(\bar{Y} \mid Z^k, \omega_j = 1, \bar{c} = j\right)\left(1 - \Phi(0; \mu_j, \sigma_j^2)\right)\sqrt{2\pi\left(\sigma_\epsilon^2 + \sigma_{ji}^2\right)}\left(1 - \Phi(0; \mu_{ji}, \sigma_{ji}^2)\right)} \tag{167}$$

$$= \frac{1}{C_{ji}}\mathcal{N}(Z_i; \mu_i, \sigma_i^2)\exp\left(-\frac{\mu_{ji}^2}{2\sigma_{ji}^2} + \frac{m_{ji}^2}{2s_{ji}^2}\right)\left(1 - \Phi(0; m_{ji}, s_{ji}^2)\right) \tag{168}$$

where

$$m_{ji} = \frac{\bar{Y}\sigma_{ji}^2 + \mu_{ji}\sigma_\epsilon^2}{\sigma_{ji}^2 + \sigma_\epsilon^2}, \qquad s_{ji}^2 = \frac{\sigma_{ji}^2\sigma_\epsilon^2}{\sigma_{ji}^2 + \sigma_\epsilon^2} \tag{169}$$

$$C_{ji} = \exp\left(\frac{\bar{Y}^2}{2\sigma_\epsilon^2}\right)\sqrt{2\pi\left(\sigma_\epsilon^2 + \sigma_{ji}^2\right)}\mathbb{P}\left(\bar{Y} \mid Z^k, \omega_j = 1, \bar{c} = j\right)\left(1 - \Phi(0; \mu_j, \sigma_j^2)\right) \tag{170}$$

In order to integrate this expression, we must get it in the standard form of (65) and (66). Note that μ_{ji} and m_{ji} depend linearly on Z_i, which we denote using constants $\mu_{ji} = a_{ji}Z_i + b_{ji}$ and $m_{ji} = c_{ji}Z_i + d_{ji}$. Now we can combine the terms in Z_i into a single normal density:

$$\mathcal{N}(Z_i; \mu_i, \sigma_i^2)\exp\left(-\frac{\mu_{ji}^2}{2\sigma_{ji}^2} + \frac{m_{ji}^2}{2s_{ji}^2}\right) = \mathcal{N}(Z_i; \mu_i, \sigma_i^2)\exp\left(-\frac{(a_{ji}Z_i + b_{ji})^2}{2\sigma_{ji}^2} + \frac{(c_{ji}Z_i + d_{ji})^2}{2s_{ji}^2}\right)$$

$$= \frac{1}{\sqrt{2\pi\sigma_i^2}}\exp\left(-\frac{(Z_i - \mu_i)^2}{2\sigma_i^2} - \frac{(a_{ji}Z_i + b_{ji})^2}{2\sigma_{ji}^2} + \frac{(c_{ji}Z_i + d_{ji})^2}{2s_{ji}^2}\right) \tag{171}$$

$$= \frac{r_{ji}}{\sigma_i}\exp\left(-\frac{\mu_i^2}{2\sigma_i^2} - \frac{b_{ji}^2}{2\sigma_{ji}^2} + \frac{d_{ji}^2}{2s_{ji}^2} + \frac{w_{ji}^2}{2r_{ji}^2}\right)\mathcal{N}(Z_i; w_{ji}, r_{ji}^2) \tag{172}$$

where

$$r_{ji}^2 = \left(\frac{1}{\sigma_i^2} + \frac{a_{ji}^2}{\sigma_{ji}^2} - \frac{c_{ji}^2}{s_{ji}^2}\right)^{-1}, \qquad w_{ji} = r_{ji}^2\left(\frac{\mu_i}{\sigma_i^2} - \frac{a_{ji}b_{ji}}{\sigma_{ji}^2} + \frac{c_{ji}d_{ji}}{s_{ji}^2}\right) \tag{173}$$

The simplified density becomes

$$\mathbb{P}\left[Z_i \mid Z^k, \bar{Y}, \bar{c} = j, \omega_j = 1\right] = \frac{1}{D_{ji}}\mathcal{N}(Z_i; w_{ji}, r_{ji}^2)\left(1 - \Phi(0; m_{ji}, s_{ji}^2)\right) \tag{174}$$

$$D_{ji} = C_{ji}\frac{\sigma_i}{r_{ji}}\exp\left(\frac{\mu_i^2}{2\sigma_i^2} + \frac{b_{ji}^2}{2\sigma_{ji}^2} - \frac{d_{ji}^2}{2s_{ji}^2} - \frac{w_{ji}^2}{2r_{ji}^2}\right) \tag{175}$$

Integrating this, we get

$$\mathbb{E}\left[Z_i \mid Z^k, \bar{Y}, \bar{c} = j, \omega_j = 1\right] = \int Z_i\mathbb{P}\left[Z_i \mid Z^k, \bar{Y}, \bar{c} = j, \omega_j = 1\right] dZ_i \tag{176}$$

$$= \frac{1}{D_{ji}} \int Z_i \mathcal{N}(Z_i; w_{ji}, r_{ji}^2) \left(1 - \Phi(0; m_{ji}, s_{ji}^2)\right) dZ_i \tag{177}$$

$$= \frac{1}{D_{ji}} \left[w_{ji} - \int Z_i \mathcal{N}(Z_i; w_{ji}, r_{ji}^2) \Phi(0; m_{ji}, s_{ji}^2) dZ_i \right] \tag{178}$$

$$= \frac{1}{D_{ji}} \left[w_{ji} - \int \frac{Z_i}{r_{ji}} \phi\left(\frac{Z_i - w_{ji}}{r_{ji}}\right) \Phi\left(\frac{-c_{ji} Z_i - d_{ji}}{s_{ji}}\right) dZ_i \right] \tag{179}$$

$$= \frac{1}{D_{ji}} \left[w_{ji} - \int (r_{ji} W + w_{ji}) \phi\left(W\right) \Phi\left(\frac{-c_{ji} r_{ji} W - c_{ji} w_{ji} - d_{ji}}{s_{ji}}\right) dZ_i \right] \tag{180}$$

$$= \frac{1}{D_{ji}} \left[w_{ji} + \frac{c_{ji} r_{ji}^2}{\sqrt{s_{ji}^2 + c_{ji}^2 r_{ji}^2}} \phi\left(\frac{-c_{ji} w_{ji} - d_{ji}}{\sqrt{s_{ji}^2 + c_{ji}^2 r_{ji}^2}}\right) - w_{ji} \Phi\left(\frac{-c_{ji} w_{ji} - d_{ji}}{\sqrt{s_{ji}^2 + c_{ji}^2 r_{ji}^2}}\right) \right] \tag{181}$$