Homework 4 Solutions

1 Factor analysis

A. (a) Based on the Factor Analysis (FA), we have the probability of x_i given the parameter θ as:

$$p(x_i|\theta) = \int \mathcal{N}(x_i|Wz_i + \mu, \Psi)\mathcal{N}(z_i|\mu_0, \Sigma_0)dz_i \tag{1}$$

$$= \mathcal{N}(x_i|W\mu_0 + \mu, \Psi + W\Sigma_0 W^T). \tag{2}$$

By taking $\mu_0 = 0$ and $\Sigma_0 = I$, we have:

$$p(x_i|\theta) = \mathcal{N}(x_i|\mu, \Psi + WW^T) \tag{3}$$

So now, we want to calculate the number of independent parameters in the FA model or the Degree of Freedom (DOF). In order to calculate it, we should sum all the DOFs related to the W (Weights matrix), Ψ (Diagonal covariance) and R (Orthonormal Rotational).

$$\mu \Longrightarrow D \quad (DOF) \tag{4}$$

$$W \Longrightarrow D \times L \quad (DOF) \tag{5}$$

$$\Psi \Longrightarrow D \quad (DOF) \tag{6}$$

$$R \Longrightarrow L(L-1)/2 \quad (DOF) \tag{7}$$

It we to use the full matrix of weights W, we would end up in a situation where we would have fail to have a non-unique solution. This is because the weights by default are invariant of rotation (see part b), thus we *subtract* the DOF of the rotation matrix from the weight to ensure we have a unique solution:

$$DOF = DL + 2D - L(L-1)/2. \tag{8}$$

(b) If we take $\tilde{W} = WR$ where R is the orthonormal rotation matrix, then we have:

$$p(x_i|\theta) = \mathcal{N}(x_i|\mu, \Psi + \tilde{W}\tilde{W}^T) \tag{9}$$

$$= \mathcal{N}(x_i|\mu, \Psi + WRR^T W^T) \tag{10}$$

$$= \mathcal{N}(x_i|\mu, \Psi + WW^T). \tag{11}$$

So, we can conclude that it is invariant to rotation as $RR^T = I$.

(c) Given an arbitrary unitary matrix $\mathbf{R}$ so that $\mathbf{R}\mathbf{R}^{\mathrm{T}} = \mathbf{R}^{\mathrm{T}}\mathbf{R} = \mathbf{I}$, interpretable as an orthogonal projection or rotation of the space, we see that the covariance in the model above is nonidentifiable since we may always define a new projection $\widetilde{\mathbf{W}} = \mathbf{W}\mathbf{R}$ with corresponding covariance

$$\mathbb{C}[\mathbf{x}, \mathbf{x}] = \widetilde{\mathbf{W}}\widetilde{\mathbf{W}}^{\mathrm{T}} + \mathbf{\Psi} = \mathbf{W}\mathbf{R}\mathbf{R}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}} + \mathbf{\Psi} = \mathbf{W}\mathbf{W}^{\mathrm{T}} + \mathbf{\Psi}$$

which is identical to that given by the original factor loadings. Thus the model covariance is invariant to any unitary rotations of the space.

B. Using the 'factornan' MATLAB implementation with no rotational constrains, we find the results of Figure 1. Note that this function automatically centers and scales the input appropriate to the method. We see that the x-axis in the 2-dimensional space is nearly parallel with the price of the vehicles; while it is difficult to describe what the y-axis is most like, it is useful to note that the mpg features and weight are almost perfectly parallel and and though they are less orthogonal to the price than features such as wheelbase and length are, they are at least much more directly interpretable in the latent space.

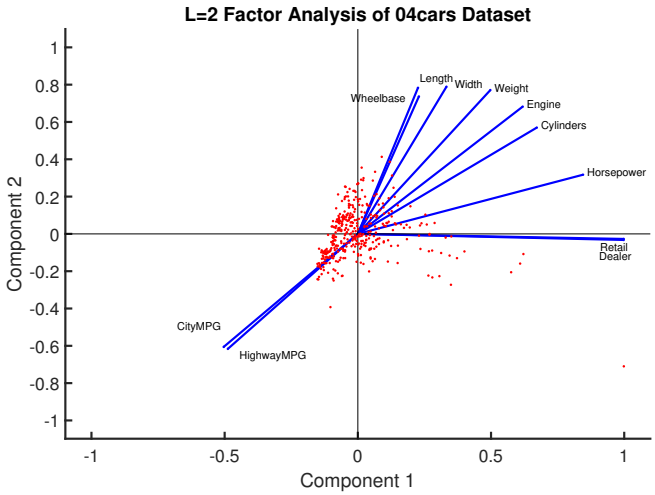


Figure 1: Plot of projected scores and unit bases into the 2-dimensional latent factor space.

2 Bayesian logistic regression

A. Based on the assumption, we have the likelihood and prior as:

$$p(y|x,w) = Ber(y|sigm(w^T x)) = \prod_{n=1}^{N} t_n^{y_n}(1-t_n)^{(1-y_n)} \tag{12}$$

where t_n is equal to $\sigma(w^T x_0)$.

$$p(w) = \mathcal{N}(w|0, V_0) \tag{13}$$

Through application of Bayes rule we obtain a posterior for the weight w:

$$p(w|y) \propto p(w)p(y|w) \tag{14}$$

$$lnp(w|y) \propto -\frac{1}{2}w^T V_0^{-1} w + \sum_{n=1}^{N}\{y_n ln(t_n) + (1-y_n)ln(1-t_n)\} + c \tag{15}$$

To approximate a Gaussian, we will use the Laplace approximation to find the approximate variance and the map estimate as the mean approximation. Thus to find we need to find the second derivative of the posterior:

$$S_N^{-1} = -\nabla\nabla lnp(w|y) \propto {V_0}^{-1} + \frac{\partial^2 F}{\partial w^2} \tag{16}$$

where the F is equal to:

$$F = \sum_{n=1}^{N} \{y_n ln(t_n) + (1 - y_n) ln(1 - t_n)\} \tag{17}$$

First let us find the first derivative, $\frac{\partial F}{\partial w}$:

$$\frac{\partial F}{\partial w} = \sum_{n=1}^{N} y_n \frac{1}{\sigma(w^t x)} \sigma'(w^t x)x - (1 - y_n) \frac{1}{(1 - \sigma(w^t x))} \sigma'(w^t x)x. \tag{18}$$

we need to make use of the following identities related to the sigmoid function:

$$\sigma(a) = \frac{1}{1 + e^{-a}} \tag{19}$$

$$\sigma'(a) = \frac{e^{-a}}{(1 + e^{-a})^2} = e^{-\alpha} \sigma(a)^2. \tag{20}$$

Resulting in:

$$\frac{\sigma'(a)}{\sigma(a)} = e^{-\alpha} \sigma(a) \tag{21}$$

$$\frac{\sigma'(a)}{1 - \sigma(a)} = \sigma(a). \tag{22}$$

Now, we express the expressions for the first and second derivation of F wrt w:

$$\frac{\partial F}{\partial w} = \sum_{n=1}^{N} x y_n e^{-(w^T x)} \sigma(w^T x) - (1 - y_i) x \sigma(w^T x) \tag{23}$$

Now taking the second derivative:

$$\frac{\partial^2 F}{\partial w^2} = \sum_{n=1}^{N} x y_n \{-e^{-(w^T x)} \sigma(w^T x)x + e^{-(w^T x)} \sigma'(w^T x)x\} - (1 - y_i) x \{\sigma'(w^T x)x\} \tag{24}$$

$$= \sum_{n=1}^{N} x y_n \{x e^{-(w^T x)} (-\sigma(w^T x) + \sigma'(w^T x))\} - (1 - y_n) x^2 \sigma'(w^T x). \tag{25}$$

For the $\sigma'(a) - \sigma(a)$ term, we can express this as:

$$\sigma'(a) - \sigma(a) = -\sigma^2(\alpha). \tag{26}$$

This allows us to write the second derivative as the following:

$$\frac{\partial^2 F}{\partial w^2} = \sum_{n=1}^{N} x^2 y_n e^{-(w^T x)}(-\sigma^2(w^T x)) - (1-y_n)x^2\sigma'(w^T x) \tag{27}$$

$$= \sum_{n=1}^{N} x^2\{-y_n e^{-w^T x}\sigma^2(w^T x) - \sigma'(w^T x) + y_n\sigma'(w^T x)\} \tag{28}$$

$$= \sum_{n=1}^{N} x^2 \left\{ y_n \left[-\frac{e^{-(w^T x)}}{(1+e^{-(w^T x)})^2} + \frac{e^{-(w^T x)}}{(1+e^{-(w^T x)})^2} \right] - \sigma'(w^T x) \right\} \tag{29}$$

$$= \sum_{n=1}^{N} x^2\sigma'(w^T x) \tag{30}$$

$$= \sum_{n=1}^{N} x^2 \frac{e^{-(w^T x)}}{[1+e^{-(w^T x)}]^2} \tag{31}$$

We can note the following property:

$$t_n(1-t_n)x_n(x_n)^T = \sigma(w^T x)(1-\sigma(w^T x))x^2 = \frac{1}{1+e^{-(w^T x)}}\frac{e^{-(w^T x)}}{1+e^{-(w^T x)}}x^2 \tag{32}$$

and substitute it into the second derivative to yield:

$$\frac{\partial^2 F}{\partial w^2} = \sum_{n=1}^{N} t_n(1-t_n)x_n(x_n)^T \tag{33}$$

And finally we fine:

$$S_N{}^{-1} = V_0{}^{-1} + \sum_{n=1}^{N} t_n(1-t_n)x_n x_n^T \tag{34}$$

And thus we approximate our posterior as:

$$p(\boldsymbol{w}|\mathcal{D}) \approx \mathcal{N}(w_{map}, S_N^{-1}). \tag{35}$$

B. The posterior predictive distribution has the following form:

$$p(y|x,\mathcal{D}) = \int p(y|x,\boldsymbol{w})p(\boldsymbol{w}|\mathcal{D})d\boldsymbol{w} \tag{36}$$

This integral is not tractable thus we will need to approximate it.

Monte Carlo approximation:

$$p(y|x,\mathcal{D}) \approx \frac{1}{S}\sum_{s=1}^{S} \text{sigm}((\boldsymbol{w}^*)^T x) \tag{37}$$

where, $w^* \sim p(\boldsymbol{w}|\mathcal{D})$ are weight samples from the posterior.

Probit regression:

If we have a Gaussian approximation to the posterior as we did in the previous section, we can compute a deterministic approximation to the posterior predictive distribution:

$$p(y=1|\boldsymbol{x},\boldsymbol{\mathcal{D}}) = \int p(y=1|\boldsymbol{x},\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\mathcal{D}})d\boldsymbol{w} \tag{38}$$

$$\approx \int \text{sigm}(\boldsymbol{w}^T\boldsymbol{x})p(\boldsymbol{w}|\boldsymbol{\mathcal{D}})d\boldsymbol{w} \tag{39}$$

$$\approx \int \text{sigm}(a)p(a|\mu_a,\sigma_a^2)d\boldsymbol{a} \tag{40}$$

where we have defined

$$a \triangleq \boldsymbol{w}^T\boldsymbol{x} \tag{41}$$

$$\mu_a \triangleq \mathbb{E}[a] = \boldsymbol{m}_{\boldsymbol{N}}^T\boldsymbol{x} \tag{42}$$

$$\sigma_a^2 \triangleq var[a] = \int p(a|D)[a^2 - \mathbb{E}[a^2]]da \tag{43}$$

$$= \int p(\boldsymbol{w}|\boldsymbol{\mathcal{D}})[(\boldsymbol{w}^T\boldsymbol{x})^2 - (\boldsymbol{m}_{\boldsymbol{N}}^T\boldsymbol{x})^2]d\boldsymbol{w} = \boldsymbol{x}^T\boldsymbol{V}_N\boldsymbol{x} \tag{44}$$

The major advantage of using the probit is that one can convolve it with a Gaussian analytically:

$$\Phi(\lambda a)\mathcal{N}(a|\mu,\sigma^2)da = \Phi\left(\frac{a}{(\lambda^{-2}+\sigma^2)^{(1/2)}}\right) \tag{45}$$

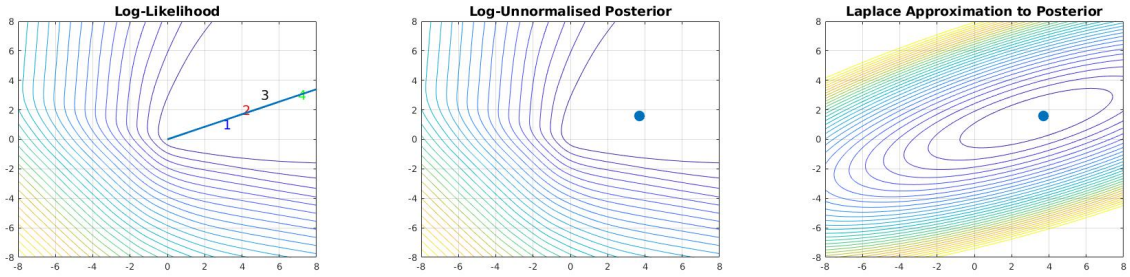
We now plug in the approximation sigm(a) $\approx \Phi(\lambda a)$ to both sides of this equation to get

$$\int sigm(a)\mathcal{N}(a|\mu,\sigma^2)da \approx sigm(\kappa(\sigma^2)\mu) \tag{46}$$

$$\kappa(\sigma^2) \triangleq (1+\pi\sigma^2/8)^{-1/2}. \tag{47}$$

Finally for our logisitc regression model we get the following:

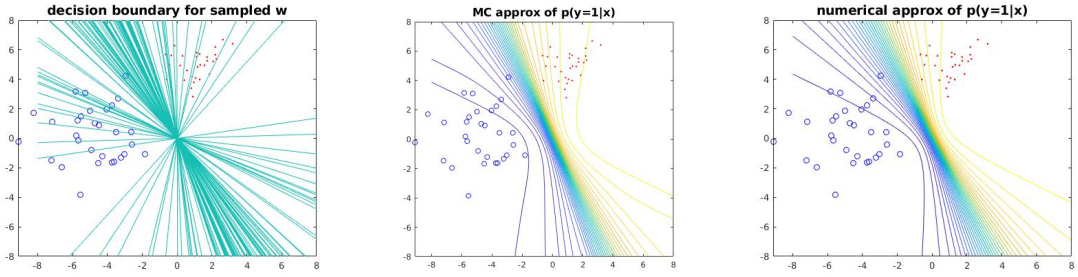
$$p(y=1|\boldsymbol{x},\boldsymbol{\mathcal{D}}) \approx sigm(\kappa(\sigma_a^2)\mu_a). \tag{48}$$



(a) (b) (c)

Figure 2: (Left to Right) The log-likelihood, log-unormalized posterior and the laplace approximation. The blue dot is the MAP estiamte.

C.



(a) (b) (c)

Figure 3: (Left to Right) Samples of weights drawn from the weights, Monte Carlos approximation of the posterior and the approximated posterior using the probit approximation.

D.

3 Matrix inversion lemma

Since Cis not a full rank matrix we can use the matrix inversion lemma:
$|\boldsymbol{C}| = |\boldsymbol{W}\boldsymbol{W}^T| + \sigma^2\boldsymbol{I}_{D\times D} = |\boldsymbol{U}_M(\boldsymbol{L} - \sigma^2\boldsymbol{I}_{M\times M})\boldsymbol{U}^T + \sigma^2\boldsymbol{I}_{D\times D}$
$|\boldsymbol{C} = |(\boldsymbol{L}_M - \sigma^2\boldsymbol{I})^{-1}| + \boldsymbol{U}^T\sigma^{-2}\boldsymbol{I}_{M\times M}\boldsymbol{U}||L_M - \sigma^2\boldsymbol{I}_{M\times M})||\sigma^2\boldsymbol{I}_{D\times D})|$
Here we used the MLE value for $\boldsymbol{W} = \boldsymbol{U}_M(\boldsymbol{L}_M - \sigma^2\boldsymbol{I})$
and the matrix inversion lemma:$det(\boldsymbol{A} + \boldsymbol{U}\boldsymbol{W}\boldsymbol{V}_T) = det(\boldsymbol{W}^{-1} + \boldsymbol{V}^T\boldsymbol{A}^{-1}\boldsymbol{U})det\boldsymbol{W}det\boldsymbol{A}$

Taking logs and evaluating the above determinants:
$\log|\boldsymbol{C}| = \left(\sum_{i=1}^{M}\log\lambda_i - M\log\sigma^2 - \sum_{i=1}^{M}\log\left(\lambda_i - \sigma^2\right)\right) + D\log\sigma^2$

$log|\boldsymbol{W}\boldsymbol{W}^T| + \sigma^2\boldsymbol{I}| = (D-M)\log\sigma^2\sum_{i=1}^{M}\log\lambda_i$

If we consider no dimensionality reduction ($M = D$), then
$\boldsymbol{U}_M = \boldsymbol{U}, \boldsymbol{L}_M = \boldsymbol{L}$,
and using $\boldsymbol{U}\boldsymbol{U}^T = \boldsymbol{I}, \boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}$, we see the covariance $\boldsymbol{C}$ of the marginal distribution for x becomes
$\boldsymbol{C} = \boldsymbol{W}\boldsymbol{W}^T + \sigma^2\boldsymbol{I} = \boldsymbol{U}\big(L - \sigma^2\boldsymbol{I}\big)^{\frac{1}{2}}\boldsymbol{R}\boldsymbol{R}^T\big(L - \sigma^2\boldsymbol{I}\big)^{\frac{1}{2}}\boldsymbol{U}^T + \sigma^2\boldsymbol{I} = \boldsymbol{U}\boldsymbol{L}\boldsymbol{U}^T = \boldsymbol{S}$

4 K-means algorithm

The built-in MATLAB procedure is used to cluster the multivariate time series data shown in Fig. 4, which models yeast gene expression via red/green fluorescence trace ratio. Two distance metrics are used to determine the clusters: standard Euclidean (Fig. 5) and 'correlation' distance (Fig. 6). The differing results illustrate how sensitive the results of k-means can be to the way distance is measured; with Euclidean distance more the more similarly scaled series are coupled to give a greater variability in activation amounts, typically ranging upwards of ± 3, whereas the correlation distance groups things more qualitatively regardless of the original scale of the series, which drives average centroid values to zero and limits their variability by combining multiple

series of similar shape but opposite scale.

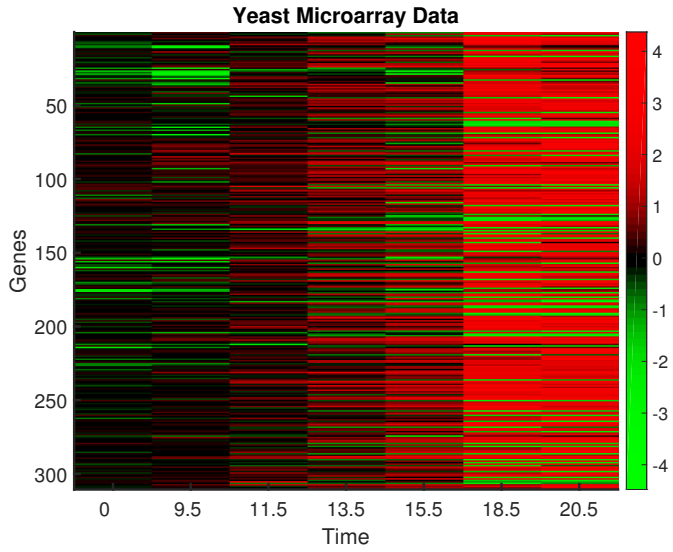


Figure 4: Colorplot illustrating the red/green activation ratio of each gene sequence over time.

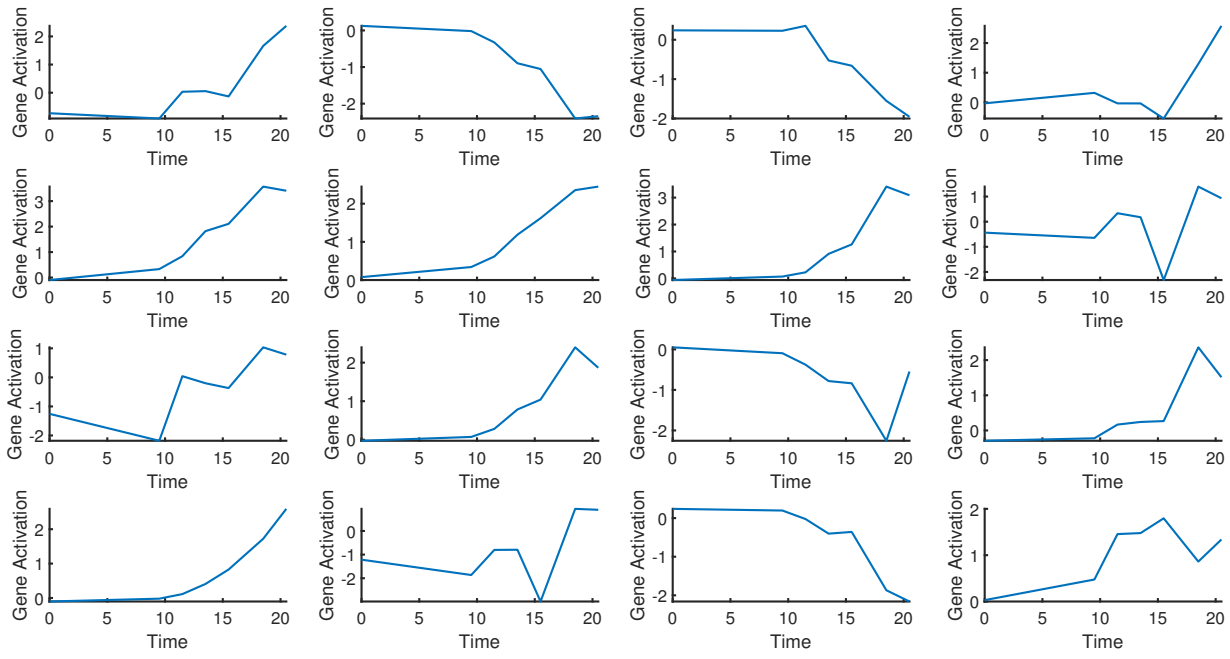


Figure 5: Centroids of k-means clustering with $k = 16$ and the Euclidean distance metric.

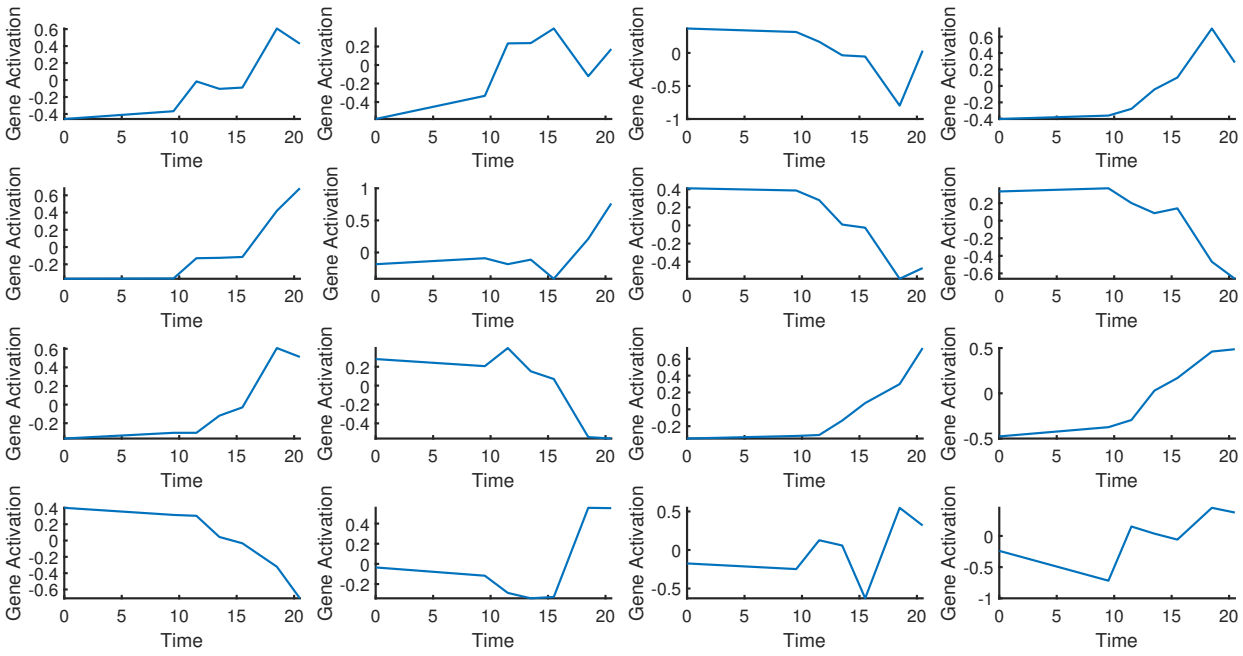


Figure 6: Centroids of k-means clustering with $k = 16$ and the 'correlation' distance metric.

5 Gaussian mixture, expectation maximization and mixture of experts

A. For a single univariate data point y, the classic (central) Student-t distribution may be expressed as the solution to the following integral equation

$$\mathcal{T}(y \mid \tau^2, \nu) = \int_0^\infty \mathcal{N}(y \mid 0, \frac{\tau^2}{\sigma^2}) \Gamma(\sigma^2 \mid \frac{\nu}{2}, \frac{\nu}{2}) \, d\sigma^2$$

whose Reimann sum over the partitions $\Delta\sigma_k^2 \in \mathbb{R}^+$ is expressible as

$$= \lim_{\Delta\sigma_k^2 \to 0} \sum_{k=1}^{\infty} \mathcal{N}(y \mid 0, \frac{\tau^2}{\sigma_k^2}) \Gamma(\sigma_k^2 \mid \frac{\nu}{2}, \frac{\nu}{2}) \, \Delta\sigma_k^2$$

and is therefore easily seen to be a mixture of an infinite number of Gaussians, each weighted by a gamma distribution over the denominator of the variance.

B. For the respective general RUM model, we have that the kernel of the complete data likelihood necessary for the E-step is

$$L_c(\mathbf{z}, \mathbf{w} \mid \mathbf{X}, \mathbf{y}) \propto \mathbb{P}(\mathbf{y} \mid \mathbf{z}) \mathbb{P}(\mathbf{z} \mid \mathbf{X}, \mathbf{w}) \mathbb{P}(\mathbf{w})$$

with respect to a potential prior distribution on the weights $\mathbb{P}(\mathbf{w})$. Note that since $\mathbb{P}(\mathbf{y}_i = 1 \mid \mathbf{z}_i) = \mathbb{I}(\mathbf{z}_i > 0)$ for the indicator function $\mathbb{I}(\cdot)$, we therefore have $\mathbb{P}(\mathbf{y}_i = 0 \mid \mathbf{z}_i) = \mathbb{I}(\mathbf{z}_i \leq 0)$. Now, specifying probit

regression in this context is equivalent to specifying uncorrelated normal errors with unit variance in the RUM model and thus we can further express the complete data likelihood as

$$L_c(\mathbf{z}, \mathbf{w} \mid \mathbf{X}, \mathbf{y}) \propto \prod_{i=1}^{N} \mathbb{I}(\mathbf{z}_i > 0)^{\mathbf{y}_i} \mathbb{I}(\mathbf{z}_i \leq 0)^{1-\mathbf{y}_i} \mathcal{N}(\mathbf{z}_i \mid \mathbf{X}_{i,:}\mathbf{w}, 1)\mathbb{P}(\mathbf{w})$$

in terms of the N uncorrelated data samples. In the log domain, this becomes

$$\ell_c(\mathbf{z}, \mathbf{w} \mid \mathbf{X}, \mathbf{y}) = -\frac{1}{2}\left\{\sum_{i\in\mathcal{C}_1} \mathbb{I}(\mathbf{z}_i > 0)(\mathbf{z}_i - \mathbf{X}_{i,:}\mathbf{w})^2 + \sum_{i\in\mathcal{C}_0} \mathbb{I}(\mathbf{z}_i \leq 0)(\mathbf{z}_i - \mathbf{X}_{i,:}\mathbf{w})^2\right\} + \ln[\mathbb{P}(\mathbf{w})] + A$$

for some constant A and class sets defines as $\mathcal{C}_j = \{i \mid \mathbf{y}_i = j\}$. With this established we evaluate the E-step as

$$Q(\mathbf{w}, \widehat{\mathbf{w}}^{(t-1)}) = \mathbb{E}_{\mathbf{z}|\mathbf{w}^{(t-1)}}\left[\ell_c(\mathbf{z}, \mathbf{w} \mid \mathbf{X}, \mathbf{y})\right]$$

$$= \mathbb{E}_{\mathbf{z}|\mathbf{w}^{(t-1)}}\left[-\frac{1}{2}\left\{\sum_{i\in\mathcal{C}_1} \mathbb{I}(\mathbf{z}_i > 0)(\mathbf{z}_i - \mathbf{X}_{i,:}\mathbf{w})^2 + \sum_{i\in\mathcal{C}_0} \mathbb{I}(\mathbf{z}_i \leq 0)(\mathbf{z}_i - \mathbf{X}_{i,:}\mathbf{w})^2\right\} + \ln[\mathbb{P}(\mathbf{w})] + A\right]$$

$$= -\frac{1}{2}\left\{\sum_{i\in\mathcal{C}_1} \mathbb{E}_{\mathbf{z}|\mathbf{w}^{(t-1)}}\left[\mathbb{I}(\mathbf{z}_i > 0)(\mathbf{z}_i^2 - 2\mathbf{z}_i\mathbf{X}_{i,:}\mathbf{w} + (\mathbf{X}_{i,:}\mathbf{w})^2)\right]\right.$$

$$\left. + \sum_{i\in\mathcal{C}_0} \mathbb{E}_{\mathbf{z}|\mathbf{w}^{(t-1)}}\left[\mathbb{I}(\mathbf{z}_i \leq 0)(\mathbf{z}_i^2 - 2\mathbf{z}_i\mathbf{X}_{i,:}\mathbf{w} + (\mathbf{X}_{i,:}\mathbf{w})^2)\right]\right\} + \ln[\mathbb{P}(\mathbf{w})] + A$$

$$= -\frac{1}{2}\left\{\sum_{i\in\mathcal{C}_1} \mathbb{E}_{\mathbf{z}|\mathbf{w}^{(t-1)}}\left[\mathbf{z}_i^2 \mid \mathbf{z}_i > 0\right] - 2\mathbf{X}_{i,:}\mathbf{w}\mathbb{E}_{\mathbf{z}|\mathbf{w}^{(t-1)}}\left[\mathbf{z}_i \mid \mathbf{z}_i > 0\right] + (\mathbf{X}_{i,:}\mathbf{w})^2\right.$$

$$\left. + \sum_{i\in\mathcal{C}_0} \mathbb{E}_{\mathbf{z}|\mathbf{w}^{(t-1)}}\left[\mathbf{z}_i^2 \mid \mathbf{z}_i \leq 0\right] - 2\mathbf{X}_{i,:}\mathbf{w}\mathbb{E}_{\mathbf{z}|\mathbf{w}^{(t-1)}}\left[\mathbf{z}_i \mid \mathbf{z}_i \leq 0\right] + (\mathbf{X}_{i,:}\mathbf{w})^2\right\} + \ln[\mathbb{P}(\mathbf{w})] + A$$

and I refrain from immediately evaluating the conditional expectations and proceed to the M-step, where we have

$$\mathbf{w}^{(t)} = \underset{\mathbf{w}}{\operatorname{argmax}}\, Q(\mathbf{w}, \mathbf{w}^{(t-1)})$$

$$\frac{d}{d\mathbf{w}}Q(\mathbf{w}, \mathbf{w}^{(t-1)}) = -\frac{1}{2}\left\{\sum_{i\in\mathcal{C}_1} -2\mathbf{X}_{i,:}^{\mathrm{T}}\mathbb{E}_{\mathbf{z}|\mathbf{w}^{(t-1)}}\left[\mathbf{z}_i \mid \mathbf{z}_i > 0\right] + 2\mathbf{X}_{i,:}^{\mathrm{T}}\mathbf{X}_{i,:}\mathbf{w}\right.$$

$$\left. + \sum_{i\in\mathcal{C}_0} -2\mathbf{X}_{i,:}^{\mathrm{T}}\mathbb{E}_{\mathbf{z}|\mathbf{w}^{(t-1)}}\left[\mathbf{z}_i \mid \mathbf{z}_i \leq 0\right] + 2\mathbf{X}_{i,:}^{\mathrm{T}}\mathbf{X}_{i,:}\mathbf{w}\right\} + \frac{d}{d\mathbf{w}}\ln[\mathbb{P}(\mathbf{w})]$$

$$= \sum_{i\in\mathcal{C}_1}\left\{\mathbf{X}_{i,:}^{\mathrm{T}}\mathbb{E}_{\mathbf{z}|\mathbf{w}^{(t-1)}}\left[\mathbf{z}_i \mid \mathbf{z}_i > 0\right]\right\} + \sum_{i\in\mathcal{C}_0}\left\{\mathbf{X}_{i,:}^{\mathrm{T}}\mathbb{E}_{\mathbf{z}|\mathbf{w}^{(t-1)}}\left[\mathbf{z}_i \mid \mathbf{z}_i \leq 0\right]\right\} - \mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{w} + \frac{d}{d\mathbf{w}}\ln[\mathbb{P}(\mathbf{w})]$$

which canceled many of the more complicated expectations, leaving only

$$\mathbb{E}_{\mathbf{z}|\mathbf{w}^{(t-1)}}\left[\mathbf{z}_i \mid \mathbf{z}_i > 0\right] = \left[\mathbf{X}_{i,:}\mathbf{w}^{(t-1)} + \frac{\phi(\mathbf{X}_{i,:}\mathbf{w}^{(t-1)})}{\Phi(\mathbf{X}_{i,:}\mathbf{w}^{(t-1)})}\right]$$

$$\mathbb{E}_{\mathbf{z}|\mathbf{w}^{(t-1)}}\left[\mathbf{z}_i \mid \mathbf{z}_i \leq 0\right] = \left[\mathbf{X}_{i,:}\mathbf{w}^{(t-1)} - \frac{\phi(\mathbf{X}_{i,:}\mathbf{w}^{(t-1)})}{1-\Phi(\mathbf{X}_{i,:}\mathbf{w}^{(t-1)})}\right]$$

which are given by the known properties of the truncated normal distribution. However, we still cannot generally solve yet without specifying a prior on $\mathbf{w}$, which I now specify to correspond to ridge regression by letting

$$\mathbb{P}(\mathbf{w} \mid \alpha) = \mathcal{N}(\mathbf{w} \mid \mathbf{0}, \frac{1}{\alpha}\mathbf{I}) \qquad \frac{d}{d\mathbf{w}}\ln[\mathbb{P}(\mathbf{w})] = -\frac{\alpha}{2}\frac{d}{d\mathbf{w}}\mathbf{w}^{\mathrm{T}}\mathbf{w} = -\alpha\mathbf{w}$$

which gives

$$Q(\mathbf{w}, \mathbf{w}^{(t-1)}) = -\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{w} - \alpha\mathbf{w} + \sum_{i\in\mathcal{C}_1}\mathbf{X}_{i,:}^{\mathrm{T}}\left[\mathbf{X}_{i,:}\mathbf{w}^{(t-1)} + \frac{\phi(\mathbf{X}_{i,:}\mathbf{w}^{(t-1)})}{\Phi(\mathbf{X}_{i,:}\mathbf{w}^{(t-1)})}\right] + \sum_{i\in\mathcal{C}_0}\mathbf{X}_{i,:}^{\mathrm{T}}\left[\mathbf{X}_{i,:}\mathbf{w}^{(t-1)} - \frac{\phi(\mathbf{X}_{i,:}\mathbf{w}^{(t-1)})}{1-\Phi(\mathbf{X}_{i,:}\mathbf{w}^{(t-1)})}\right]$$

$$= -\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{w} - \alpha\mathbf{w} + \mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{w}^{(t-1)} + \sum_{i\in\mathcal{C}_1}\frac{\mathbf{X}_{i,:}^{\mathrm{T}}\phi(\mathbf{X}_{i,:}\mathbf{w}^{(t-1)})}{\Phi(\mathbf{X}_{i,:}\mathbf{w}^{(t-1)})} - \sum_{i\in\mathcal{C}_0}\frac{\mathbf{X}_{i,:}^{\mathrm{T}}\phi(\mathbf{X}_{i,:}\mathbf{w}^{(t-1)})}{\Phi(-\mathbf{X}_{i,:}\mathbf{w}^{(t-1)})}$$

$$= -\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{w} - \alpha\mathbf{w} + \mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{w}^{(t-1)} + \mathbf{X}^{\mathrm{T}}\left[\phi(\mathbf{X}\mathbf{w}^{(t-1)}) \oslash \Phi(\mathbf{y}\circ\mathbf{X}\mathbf{w}^{(t-1)} + (\mathbf{y}-\mathbf{1})\circ\mathbf{X}\mathbf{w}^{(t-1)})\right]$$

for the Hadamard (elementwise) divisor $\oslash$ if we allow ϕ and Φ to act pointwise in matrix notation. Setting equal to zero and solving for the optimum $\mathbf{w}^{(t)}$ gives

$$-\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{w}^{(t)} - \alpha\mathbf{w}^{(t)} + \mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{w}^{(t-1)} + \mathbf{X}^{\mathrm{T}}\left[\phi(\mathbf{X}\mathbf{w}^{(t-1)}) \oslash \Phi(\mathbf{y}\circ\mathbf{X}\mathbf{w}^{(t-1)} + (\mathbf{y}-\mathbf{1})\circ\mathbf{X}\mathbf{w}^{(t-1)})\right] = 0$$

$$\mathbf{w}^{(t)}(\mathbf{X}^{\mathrm{T}}\mathbf{X} + \alpha\mathbf{I}) = \mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{w}^{(t-1)} + \mathbf{X}^{\mathrm{T}}\left[\phi(\mathbf{X}\mathbf{w}^{(t-1)}) \oslash \Phi(\mathbf{y}\circ\mathbf{X}\mathbf{w}^{(t-1)} + (\mathbf{y}-\mathbf{1})\circ\mathbf{X}\mathbf{w}^{(t-1)})\right]$$

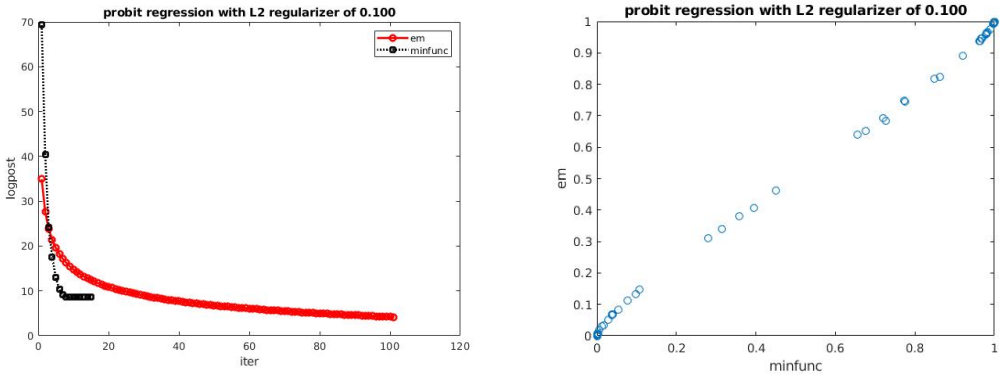
$$\mathbf{w}^{(t)} = (\mathbf{X}^{\mathrm{T}}\mathbf{X} + \alpha\mathbf{I})^{-1}\left\{\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{w}^{(t-1)} + \mathbf{X}^{\mathrm{T}}\left[\phi(\mathbf{X}\mathbf{w}^{(t-1)}) \oslash \Phi(\mathbf{y}\circ\mathbf{X}\mathbf{w}^{(t-1)} + (\mathbf{y}-\mathbf{1})\circ\mathbf{X}\mathbf{w}^{(t-1)})\right]\right\}$$

where it is perhaps useful to note that if $\alpha = 0$ this simplifies to

$$\mathbf{w}^{(t)} = \mathbf{w}^{(t-1)} + \mathbf{X}^{-}\left[\phi(\mathbf{X}\mathbf{w}^{(t-1)}) \oslash \Phi(\mathbf{y}\circ\mathbf{X}\mathbf{w}^{(t-1)} + (\mathbf{y}-\mathbf{1})\circ\mathbf{X}\mathbf{w}^{(t-1)})\right]$$

for the left pseudoinverse $\mathbf{X}^{-}$, indicating a similarity to stochastic gradient updates with a dynamic learning rate.

C. For the probit regression model, we use two versions: one that uses the previously discussed expectation maximization algorithm and the other that uses a general functional optimizer. The results are shown in the Figure below. Over all the prediction results are similar, however EM appears to converge much slower compared to the general optimizer. That being said, expectation maximization does appear to be able to to obtain a better training error (log posterior)



(a) Convergence (b) Predictions

Figure 7: Comparison of probit regression predictions for two different optimization methods.

The implementation of the above EM algorithm is given in 'Part5c.m' and compared to performance against built-in MATLAB functions for fitting general linear models with a probit link. Both implementations reach an area under an ROC curve of 1, indicating perfect linear separation of the data. The final weights after $16,681$ iterations against a tolerance of 1×10^{-4} on the ℓ_2 norm difference between iteration values in the EM algorithm are similar to the ones found by the conventional method though are smaller in magnitude, likely due to the regularization (ridge with $\alpha = 0.0001$) used in the EM. This should hopefully prevent overfitting, which the built-in functionality warns is a distinct possibility if the data is linearly separable.

D. Following the derivation in Murphy **?** (Section 11.4.5.1), a MATLAB code was written to fit a Student's t distribution. The results are shown below:

$$[w_0, w_1] \approx [-0.35, 3.82]$$

It is observed that Student's t yield robust results as the effect of outliers is minimal.

E. The results of the mixture of experts is shown below for the provided data-set. We note that the model is able to have good coverage of the data, however I will mention that sometimes the model will not converge correctly. This is of course the difficultly of fitting complex data like this where once can get trapped in a local minimum. Additionally the solution is non-unique such that while the results show the middle expert being the red component, it could easily change places with the blue or green depending on the initial condition.

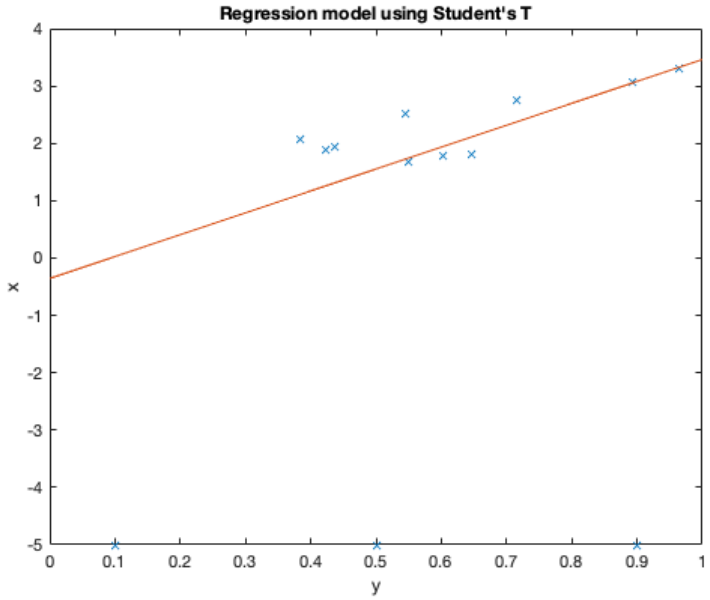
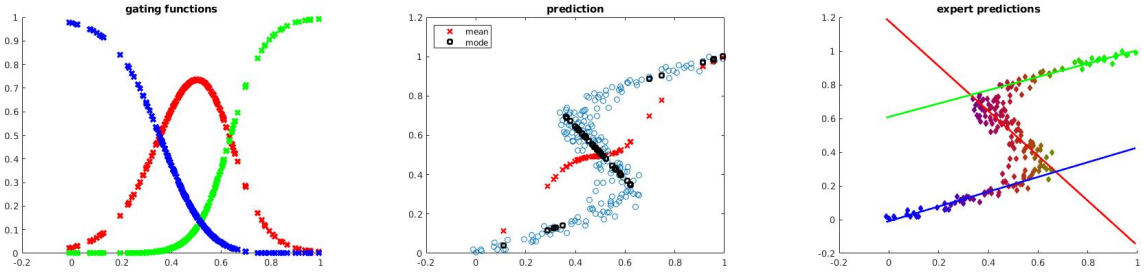


Figure 8: Linear regression using Student's t



(a) (b) (c)

Figure 9: Mixture of experts models fitted to the provided data-set.