Data analysis

Miaoyan Wang, 12/26/2019

1 Agenda for discussion

- Computational efficiency.

  I modified the algorithm using Minimization Maximization (MM). The basic idea is to approximate the objective by quadratic function and to solve the minimization by least-squares. The algorithm takes ~ 40 mins to decompose the brain data into a rank-$(24, 24, 7)$ tensor. (20 mins if reducing the determination rule `Relative decrement` $\leq 0.02\%$.)

Updating scheme	Memory	Total Runtime	Per-iter	# Iter (depending on termination rules)
Alternating Minimization (AM)				
Minimization Maximization (MM)	okay	40 mins	16 sec	146

  My result is saved in `brain_result.Rdata`.

  1. There is no asymmetric issue in the output $\Rightarrow$ A_1 and A_2 are perfectly same, and the core tensor $\mathcal{C}$ is symmetric w.r.t. first two modes. $\Rightarrow$ similar clustering based on either weighting scheme.
  2. Final cost function for MM: $224,451$. $\hat{\omega} = (-2.483677, 4.151929)$
  3. Impossible to reproduce Figures 1 and 2. Randomness involved in the K-means. Should use multiple initialization e.g. `nstart = 20`, in the K-means command.

- Visualization of brain analysis. Plot estimated or raw connection within each group.
- Missing value prediction: Output a probability vector, rather than a point prediction. Choice of ω.

2 Description of MM algorithm

For simplicity, I describe the algorithm for order-3 tensors below. The extension for higher-orders is similar. Recall that our cost function is the negative log-likelihood:

$$\mathcal{L}(\Theta) = -\sum_{\ell \in [K]} \sum_{y_{ijk}=\ell} \log f_\ell(\theta_{ijk}),$$

where $f_\ell(\theta) = \phi(w_\ell + \theta) - \phi(w_{\ell-1} + \theta)$ and $\phi(x) = \frac{1}{1+e^{-x}}$ is the logistic function. The idea of MM algorithm is to construct an auxiliary function $Q(\Theta_1, \Theta_2)$ that satisfies the two key properties:

1. $\mathcal{L}(\Theta_1) \leq Q(\Theta_1, \Theta_2)$

2. $\mathcal{L}(\Theta_1) = Q(\Theta_1, \Theta_2)$ if $\Theta_1 = \Theta_2$.

We can guarantee that the update $\Theta^{(t+1)} = \arg\min_{\Theta} Q(\Theta, \Theta^{(t)})$ leads to monotonic decrement in the cost function. To see this, let $\Theta^{(t)}$ be the current iterate, and $\Theta^{(t+1)} = \arg\min_{\Theta_1} Q(\Theta_1, \Theta^{(t)})$ the next iterate. Then the two key properties ensure

$$\mathcal{L}(\Theta^{(t+1)}) \leq Q(\Theta^{(t+1)}, \Theta^{(t)}) \leq Q(\Theta^{(t)}, \Theta^{(t)}) = \mathcal{L}(\Theta^{(t)}).$$

Therefore, the goal is to find the auxiliary function $Q(\Theta_1, \Theta_2)$.

We utilize quadratic approximation based on Taylor expansion to construct $Q(\Theta, \Theta)$. Specifically, write the cost function for each tensor entry $\theta = \theta_{i,j,k}$,

$$\begin{aligned}\mathcal{L}(\theta) &= -\sum_{\ell\in[K]} \mathbb{1}_{\{y=\ell\}} \log f_\ell(\theta)\\ &= \mathcal{L}(\theta_0) + \underbrace{\frac{\partial L(\theta)}{\partial \theta}\Big|_{\theta=\theta_0}}_{=:\text{Gradient}(\theta_0)} (\theta-\theta_0) + \frac{1}{2}\frac{\partial L^2(\theta)}{\partial \theta^2}\Big|_{\theta=\check{\theta}}(\theta-\theta_0)^2. \end{aligned} \tag{1}$$

Define $p_\ell = \phi(\theta + w_\ell)$ for $\ell = 1, \ldots, K$. The gradient and Hession of $\mathcal{L}(\theta)$ can be calculated as

$$\frac{\partial L(\theta)}{\partial \theta} = \begin{cases} p_1 - 1 & \ell = 1,\\ p_\ell + p_{\ell-1} - 1 & \ell = 2, \cdots, (K-1),\\ p_K & \ell = K, \end{cases}$$

and

$$\frac{\partial L^2(\theta)}{\partial \theta^2} = \begin{cases} p_1(1-p_1) & \ell = 1,\\ p_\ell(1-p_\ell) + p_{\ell-1}(1-p_{\ell-1}) & \ell = 2, \cdots, (K-1),\\ p_K(1-p_K) & \ell = K. \end{cases}$$

Note that $p_\ell \in [0,1]$. Therefore we have bounded Hession $\frac{\partial L^2(\theta)}{\partial \theta^2} \leq \frac{1}{2}$. Plugging the Hession bound into (1) gives

$$\begin{aligned}\mathcal{L}(\theta) &\leq \mathcal{L}(\theta_0) + \text{Gradient}(\theta_0)(\theta-\theta_0) + \frac{1}{4}(\theta-\theta_0)^2\\ &\leq \frac{1}{4}\left[\theta - \theta_0 + 2\text{Gradient}(\theta_0)\right]^2 - \text{Gradient}^2(\theta_0) + \mathcal{L}(\theta_0).\end{aligned} \tag{2}$$

Substituting θ_{ijk} for θ in (2) yields

$$\begin{aligned}\mathcal{L}(\Theta) &= \sum_{(i,j,k)} \mathcal{L}(\theta_{ijk})\\ &\leq \frac{1}{4}\left\|\Theta - (\Theta_0 - 2\text{Gradient}(\Theta_0))\right\|_F^2 - C(\Theta_0)\end{aligned}$$

$$\stackrel{\text{def}}{=} Q(\Theta, \Theta_0).$$

It is easy to verify that $Q(\Theta, \Theta_0)$ satisfies the aforementioned two key properties. In particular,

$$\hat{\Theta} = \mathop{\arg\min}_{\Theta:\text{rank}(\Theta)=r} Q(\Theta, \Theta_0) = \mathop{\arg\min}_{\Theta:\text{rank}(\Theta)=r} \left\| \Theta - \underbrace{(\Theta_0 - 2\text{Gradient}(\Theta_0))}_{:=\mathcal{M}} \right\|_F^2$$

is simply the rank-r Tucker decomposition of $\mathcal{M} \stackrel{\text{def}}{=} \Theta_0 - 2\text{Gradient}(\Theta_0)$.

To summarize, we have shown that given the current iterate $\Theta^{(t)}$, the update at t-th step can be solved via

$$\Theta^{(t+1)} = \text{Rank-}r \text{ Tucker decomposition of } \left(\Theta^{(t)} - 2\text{Gradient}(\Theta_0)\right).$$