EMAlgorithm for Factor Analysis

BY AMEER QAQISH

April 1, 2025

1 Proposal

Our proposal is to program factor analysis (https://www.cs.columbia.edu/~blei/seminar/2020-representation/readings/TippingBishop1999.pdf) to handle estimation of the parameters for datasets where there is missingness in all of the covariates, meaning each subject is missing several covariates. Assume we have an $N\times d$ design matrix $T=\begin{pmatrix} t_1^T \\ \vdots \\ t_N^T \end{pmatrix}$ and that we pick a dimension $q<d$. Using notation consistent with the paper, the model for the $N\times d$ design matrix $T=\begin{pmatrix} t_1^T \\ \vdots \\ t_N^T \end{pmatrix}$ is that

$$t_j \;=\; \mu+Wx_j+\varepsilon_j, \quad j=1,\ldots,N,$$

where $x_1,\ldots,x_N$ are i.i.d. $N(0,I_q)$, W is a $d\times q$ matrix, $\varepsilon_1,\ldots,\varepsilon_N$ are i.i.d. $N(0,\Psi)$, $\Psi=\mathrm{diag}(\sigma_1^2,\ldots,\sigma_d^2)$, independent of $x_1,\ldots,x_N$, and $\mu\in\mathbb{R}^d$. Thus (t_j,x_j,ε_j), $j=1,\ldots,N$ are i.i.d.. Estimating the parameters $\mu,W,\sigma_1,\ldots,\sigma_d$ in the prescence of missing data requires a nontrivial EM algorithm implementation detailed in Section 2.

The factor analysis model can be used for dimension reduction by using posterior means $z_j=E(x_j|T_{\mathrm{obs}},\hat{\mu},\hat{W},\hat{\sigma}_1^2,\ldots,\hat{\sigma}_d^2)\in\mathbb{R}^q$ as the projections of the t_js. The formula for $E(x|t_{\mathrm{obs}},\hat{\mu},\hat{W},\hat{\sigma}_1,\ldots,\hat{\sigma}_d)$ derived in 2.2 shows that z_j is an affine transformation of t_j.

We will test the dimension reduction capability of factor analysis by the following:

- Take a dataset $(t_i,y_i)_{i=1}^N$ where t_i is the covariate vector and y_i is a categorical outcome associated to the ith subject.
- For each covariate (column of the design matrix T), uniformly randomly make a percentage of its present values missing so that the missingness percentage is $p\%$, say $p=0,10,20,\ldots,100$. The EM algorithm maximizes the likelihood of $\mu,W,\sigma_1,\ldots,\sigma_d$ ignoring the missing mechanism. This coincides with the full likelihood if the missing data are missing at random and the missingness probabilities do not depend on $\mu,W,\sigma_1,\ldots,\sigma_d$.
- Train factor analysis and use the low dimensional projections of the covariate vectors to train a logistic regression classifier for y.
- Compare the performance of this factor analysis classifier across various values of p and also compare it to other classification approaches that handle missing covariates.

2 EM Algorithm

The observed data are the vectors $t_{p,j}=Q_j^T t_j$, $j=1,\ldots,N$ where Q_j has columns e_k for each k such that t_{jk} is observed.

2.1 M Step

Theorem 1. *The M step updates are*

$$\begin{aligned}\begin{pmatrix} \mu & W \end{pmatrix} &= \begin{pmatrix} P_N\langle t\rangle & P_N\langle tx^T\rangle \end{pmatrix}\begin{pmatrix} 1 & P_N\langle x\rangle^T \\ P_N\langle x\rangle & P_N\langle xx^T\rangle \end{pmatrix}^{-1}\\ (\sigma_j^2)_{j=1}^d &= P_N\mathrm{diag}(\langle tt^T\rangle+\mu\mu^T+W\langle xx^T\rangle W^T-2\langle t\rangle\mu^T-2\langle tx^T\rangle W^T+2\mu\langle x\rangle^T W^T).\end{aligned}$$

Here $\langle\rangle$ denotes conditional expectation with respect to the current values of the parameters conditional on the observed values $t_{p,j}=Q_j^T t_j$, $j=1,\ldots,N$, and

$$P_N f(t,x) \;:=\; \frac{1}{N}\sum_{j=1}^N f(t_j,x_j)$$

is the sample mean operator.

Proof. Set

$$\begin{aligned}\Psi(\sigma_1,\ldots,\sigma_d) &= \mathrm{diag}(\sigma_1^2,\ldots,\sigma_d^2),\\ \Omega(\sigma_1,\ldots,\sigma_d) &= \Psi(\sigma_1,\ldots,\sigma_d)^{-1}\\ &= \mathrm{diag}(\omega_1,\ldots,\omega_d).\end{aligned}$$

Up to additive terms independent of the parameters $\mu,W,\sigma_1,\ldots,\sigma_d$, the complete data log-likelihood is

$$\begin{aligned}\ell(\mu,W,\sigma_1,\ldots,\sigma_d|T,X) &= \log f(T,X|\mu,W,\sigma_1,\ldots,\sigma_d)\\ &= \sum_{j=1}^N \log f(t_j,x_j|\mu,W,\sigma_1,\ldots,\sigma_d)\\ &= \sum_{j=1}^N (\log f(x_j|\mu,W,\sigma_1,\ldots,\sigma_d)+\log f(t_j|x_j,\mu,W,\sigma_1,\ldots,\sigma_d))\\ &= \sum_{j=1}^N \left(-\frac{1}{2}|x_j|^2+\frac{1}{2}\log\det(\Omega)-\frac{1}{2}(t_j-\mu-Wx_j)^T\Omega(t_j-\mu-Wx_j)\right)\\ &= \sum_{j=1}^N \left(\frac{1}{2}\log\det(\Omega)-\frac{1}{2}(t_j-\mu-Wx_j)^T\Omega(t_j-\mu-Wx_j)\right).\end{aligned}$$

We have

$$\begin{aligned}\nabla_\mu\ell(\mu,W,\sigma_1,\ldots,\sigma_d|T,X) &= \sum_{j=1}^N \Omega(t_j-\mu-Wx_j)\\ \langle\nabla_\mu\ell(\mu,W,\sigma_1,\ldots,\sigma_d|T,X)\rangle &= \sum_{j=1}^N \Omega(\langle t_j\rangle-\mu-W\langle x_j\rangle)\\ &= \Omega\left(\sum_{j=1}^N \langle t_j\rangle-N\mu-W\sum_{j=1}^N \langle x_j\rangle\right).\end{aligned}$$

The differential with respect to W is

$$\begin{aligned}d_W\ell(\mu,W,\sigma_1,\ldots,\sigma_d|T,X) &= \sum_{j=1}^N -(-dWx_j)^T\Omega(t_j-\mu-Wx_j)\\ &= \sum_{j=1}^N x_j^T dW^T\Omega(t_j-\mu-Wx_j)\\ &= \sum_{j=1}^N \mathrm{Tr}(dW^T\Omega(t_j-\mu-Wx_j)x_j^T).\end{aligned}$$

Hence

$$\begin{aligned}\nabla_W\ell(\mu,W,\sigma_1,\ldots,\sigma_d|T,X) &= \sum_{j=1}^N \Omega(t_j-\mu-Wx_j)x_j^T\\ \langle\nabla_W\ell(\mu,W,\sigma_1,\ldots,\sigma_d|T,X)\rangle &= \Omega\left(\sum_{j=1}^N (\langle t_jx_j^T\rangle-\mu\langle x_j\rangle^T-W\langle x_jx_j^T\rangle)\right)\\ &= \Omega\left(\sum_{j=1}^N \langle t_jx_j^T\rangle-\mu\sum_{j=1}^N \langle x_j\rangle^T-W\sum_{j=1}^N \langle x_jx_j^T\rangle\right).\end{aligned}$$

Setting $\langle\nabla_\mu\ell(\mu,W,\sigma_1,\ldots,\sigma_d|T,X)\rangle=0$ and $\langle\nabla_W\ell(\mu,W,\sigma_1,\ldots,\sigma_d|T,X)\rangle=0$ yields

$$\begin{aligned} P_N\langle t\rangle-\mu-WP_N\langle x\rangle &= 0\\ P_N\langle tx^T\rangle-\mu P_N\langle x\rangle^T-WP_N\langle xx^T\rangle &= 0. \end{aligned}$$

In matrix form,

$$\begin{pmatrix}\mu & W\end{pmatrix}\begin{pmatrix}1 & P_N\langle x\rangle^T\\ P_N\langle x\rangle & P_N\langle xx^T\rangle\end{pmatrix} = \begin{pmatrix}P_N\langle t\rangle & P_N\langle tx^T\rangle\end{pmatrix}$$

Hence

$$\begin{pmatrix}\mu & W\end{pmatrix} = \begin{pmatrix}P_N\langle t\rangle & P_N\langle tx^T\rangle\end{pmatrix}\begin{pmatrix}1 & P_N\langle x\rangle^T\\ P_N\langle x\rangle & P_N\langle xx^T\rangle\end{pmatrix}^{-1}.$$

The transpose of this equation is a $(q+1)\times(q+1)$ positive definite matrix inverse times a $(q+1)\times d$, so the equation is very efficiently solved in $O(q^2d)$ time using the scipy.linalg.solve function from Python with the assume_a = "pos" option.

For Ω, we expand the log-likelihood as

$$\begin{aligned} \ell(\mu,W,\sigma_1,\ldots,\sigma_d|T,X) &= \sum_{j=1}^N\left(\frac{1}{2}\log\det(\Omega)-\frac{1}{2}(t_j-\mu-Wx_j)^T\Omega(t_j-\mu-Wx_j)\right)\\ &= \sum_{j=1}^N\left(\frac{1}{2}\sum_{k=1}^d\log\omega_k-\frac{1}{2}\sum_{k=1}^d(t_j-\mu-Wx_j)_k^2\omega_k\right) \end{aligned}$$

to get

$$\begin{aligned} \partial_{\omega_k}\ell(\mu,W,\sigma_1,\ldots,\sigma_d|T,X) &= \frac{1}{2}\sum_{j=1}^N(\omega_k^{-1}-(t_j-\mu-Wx_j)_k^2)\\ \langle\partial_{\omega_k}\ell(\mu,W,\sigma_1,\ldots,\sigma_d|T,X)\rangle &= \frac{1}{2}\left(N\omega_k^{-1}-\sum_{j=1}^N\langle(t_j-\mu-Wx_j)_k^2\rangle\right). \end{aligned}$$

Hence

$$\omega_k^{-1} = P_N\langle(t-\mu-Wx)_k^2\rangle,\quad k=1,\ldots,d.$$

We have

$$\begin{aligned} \langle(t-\mu-Wx)_k^2\rangle &= \langle e_k^T(t-\mu-Wx)(t-\mu-Wx)^Te_k\rangle\\ &= e_k^T\langle(t-\mu-Wx)(t-\mu-Wx)^T\rangle e_k\\ &= e_k^T(\langle tt^T\rangle+\mu\mu^T+W\langle xx^T\rangle W^T-2\langle t\rangle\mu^T-2\langle tx^T\rangle W^T+2\mu\langle x\rangle^TW^T)e_k. \end{aligned}$$

Hence

$$(\sigma_j^2)_{j=1}^d = P_N\mathrm{diag}(\langle tt^T\rangle+\mu\mu^T+W\langle xx^T\rangle W^T-2\langle t\rangle\mu^T-2\langle tx^T\rangle W^T+2\mu\langle x\rangle^TW^T).$$

□

2.2 E Step

This is the harder part due to the missing data. Abusing notation somewhat, let $\mu,W,\sigma_1,\ldots,\sigma_d$ denote the current value of the parameters. First, notice that the updates for the parameters only depend on the sample averages of $\langle x\rangle$, $\langle t\rangle$, $\langle xx^T\rangle$, $\langle tx^T\rangle$, and diag$(\langle tt^T\rangle)$. Recall that the observed data are the vectors $t_{p,j}=Q_j^Tt_j$, $j=1,\ldots,N$ where Q_j has columns e_k for each k for which t_{jk} is observed. The missing data are $t_{m,j}=R_j^Tt_j$, where R_j has columns e_k for each k for which t_{jk} is missing.

Since we are calculating $\langle x\rangle$, $\langle t\rangle$, $\langle xx^T\rangle$, $\langle tx^T\rangle$, and $\mathrm{diag}(\langle tt^T\rangle)$, we focus on a single sample (t,x,ε). Define permutation τ by

$$\begin{aligned}\tilde{t} &= \begin{pmatrix} t_p \\ t_m \end{pmatrix}\\ \tilde{t}_j &= t_{\tau(j)}\\ t_j &= \tilde{t}_{\tau^{-1}(j)}.\end{aligned}$$

Then

$$\begin{aligned}\langle\tilde{t}\rangle &= \begin{pmatrix} t_p \\ \langle t_m\rangle \end{pmatrix}\\ \langle\tilde{t}x^T\rangle &= \begin{pmatrix} t_p\langle x\rangle^T \\ \langle t_m x^T\rangle \end{pmatrix}\\ \langle\tilde{t}\tilde{t}^T\rangle &= \begin{pmatrix} t_p t_p^T & t_p\langle t_m\rangle^T \\ \langle t_m\rangle t_p^T & \langle t_m t_m^T\rangle \end{pmatrix}\\ \langle t\rangle_j &= \tilde{t}_{\tau^{-1}(j)}\\ \langle tx^T\rangle_{j,k} &= \langle\tilde{t}x^T\rangle_{\tau^{-1}(j),k}\\ \mathrm{diag}(\langle tt^T\rangle)_j &= \mathrm{diag}\begin{pmatrix} t_p t_p^T & t_p\langle t_m\rangle^T \\ \langle t_m\rangle t_p^T & \langle t_m t_m^T\rangle \end{pmatrix}_{\tau^{-1}(j)}\\ &= \begin{pmatrix} t_p\odot t_p \\ \mathrm{diag}(\langle t_m t_m^T\rangle) \end{pmatrix}_{\tau^{-1}(j)}.\end{aligned}$$

Hence we just need $\langle x\rangle$, $\langle t_m\rangle$, $\langle xx^T\rangle$, $\langle t_m x^T\rangle$, and $\mathrm{diag}(\langle t_m t_m^T\rangle)$. Define the submatrices

$$\begin{aligned}\mu_p &= Q^T\mu\\ \mu_m &= R^T\mu\\ W_p &= Q^T W Q\\ W_m &= R^T W R\\ \Psi_p &= Q^T\Psi Q\\ \Psi_m &= R^T\Psi R.\end{aligned}$$

Theorem 2. *Let*

$$\Sigma = (I+W_p^T\Psi_p^{-1}W_p)^{-1}.$$

Then conditional on t_p,

$$x \sim N(\Sigma W_p^T\Psi_p^{-1}(t_p-\mu_p),\Sigma).$$

Proof. Note that

$$\begin{aligned}t_p &= Q^T(\mu+Wx+\varepsilon)\\ &= \mu_p+W_p x+\varepsilon_p.\end{aligned}$$

Hence, up to additive terms that don't depend on x,

$$\begin{aligned}\log f(x|t_p) &= \log f(x)+\log f(t_p|x)\\ &= -\frac{1}{2}x^T x-\frac{1}{2}(t_p-\mu_p-W_p x)^T\Psi_p^{-1}(t_p-\mu_p-W_p x)\\ &= -\frac{1}{2}x^T x-\frac{1}{2}x^T W_p^T\Psi_p^{-1}W_p x+x^T W_p^T\Psi_p^{-1}(t_p-\mu_p)\\ &= -\frac{1}{2}x^T(I+W_p^T\Psi_p^{-1}W_p)x+x^T W_p^T\Psi_p^{-1}(t_p-\mu_p).\end{aligned}$$

Comparing this to the $N(m,\Sigma)$ log-likelihood $-\frac{1}{2}(x-m)^T\Sigma^{-1}(x-m) = -\frac{1}{2}x^T\Sigma^{-1}x + x^T\Sigma^{-1}m$ yields

$$\begin{aligned} x &\sim N(m,\Sigma) \\ \Sigma &= (I+W_p^T\Psi_p^{-1}W_p)^{-1} \\ m &= \Sigma W_p^T\Psi_p^{-1}(t_p-\mu_p). \end{aligned}$$

□

Theorem 3. *We have*

$$\begin{aligned} \langle x\rangle &= \Sigma W_p^T\Psi_p^{-1}(t_p-\mu_p) \\ \langle t_m\rangle &= \mu_m+W_m\langle x\rangle \\ \langle xx^T\rangle &= \Sigma+\langle x\rangle\langle x\rangle^T \\ \langle t_m x^T\rangle &= W_m\Sigma+\langle t_m\rangle\langle x\rangle^T \\ \langle t_m t_m^T\rangle &= W_m\Sigma W_m^T+\Psi_m+\langle t_m\rangle\langle t_m\rangle^T. \end{aligned}$$

Proof. We have

$$t_m = \mu_m+W_m x+\varepsilon_m,$$

so

$$\begin{aligned} \langle t_m\rangle &= E(t_m|t_p) \\ &= E(E(t_m|x,t_p)|t_p) \\ &= E(E(\mu_m+W_m x+\varepsilon_m|x,t_p)|t_p) \\ &= E(\mu_m+W_m x|t_p) \\ &= \mu_m+W_m\langle x\rangle, \end{aligned}$$

since ε_m and (x,t_p) are independent. Similarly, since ε_m and (x,t_p) are independent,

$$\begin{aligned} \mathrm{Cov}(t_m,x|t_p) &= E(\mathrm{Cov}(t_m,x|x,t_p)|t_p)+\mathrm{Cov}(E(t_m|x,t_p),E(x|x,t_p)|t_p) \\ &= \mathrm{Cov}(\mu_m+W_m x,x|t_p) \\ &= W_m\Sigma \end{aligned}$$

Finally,

$$\begin{aligned} \mathrm{Var}(t_m|t_p) &= E(\mathrm{Var}(t_m|x,t_p)|t_p)+\mathrm{Var}(E(t_m|x,t_p)|t_p) \\ &= E(\mathrm{Var}(\mu_m+W_m x+\varepsilon_m|x,t_p)|t_p)+\mathrm{Var}(\mu_m+W_m x|t_p) \\ &= \Psi_m+W_m\Sigma W_m^T. \end{aligned}$$

□

2.3 Implementation

The E step is implemented in the following order. For each an observation index $j\in\{1,\dots,N\}$, do the following: calculate the following matrices; the time complexity is listed on the right.

$$\begin{aligned} \Sigma &= (I+W_p^T\Psi_p^{-1}W_p)^{-1} && q^2 d. \\ \langle x\rangle &= \Sigma W_p^T\Psi_p^{-1}(t_p-\mu_p) && qd \\ \langle xx^T\rangle &= \Sigma+\langle x\rangle\langle x\rangle^T && q^2 \\ \langle t_m\rangle &= \mu_m+W_m\langle x\rangle && qd \\ W_m\Sigma &= W_m\Sigma && q^2 d \\ \langle t_m x^T\rangle &= W_m\Sigma+\langle t_m\rangle\langle x\rangle^T && qd \\ \mathrm{diag}(\langle t_m t_m^T\rangle) &= \mathrm{rowSums}(W_m\odot(W_m\Sigma))+\mathrm{diag}(\Psi_m)+\langle t_m\rangle\odot\langle t_m\rangle && qd. \end{aligned}$$

Then use these to form

$$\begin{aligned}
\langle\tilde{t}\rangle &= \begin{pmatrix} t_p \\ \langle t_m\rangle \end{pmatrix} \\
\langle\tilde{t}x^T\rangle &= \begin{pmatrix} t_p\langle x\rangle^T \\ \langle t_m x^T\rangle \end{pmatrix} \\
\langle t\rangle_j &= \tilde{t}_{\tau^{-1}(j)} \\
\langle tx^T\rangle_{j,k} &= \langle\tilde{t}x^T\rangle_{\tau^{-1}(j),k} \\
\mathrm{diag}(\langle tt^T\rangle)_j &= \begin{pmatrix} t_p \odot t_p \\ \mathrm{diag}(\langle t_m t_m^T\rangle) \end{pmatrix}_{\tau^{-1}(j)}
\end{aligned}$$

in qd time. The M step updates are obtained by summing over N:

$$\begin{aligned}
\begin{pmatrix} \mu & W \end{pmatrix} &= \begin{pmatrix} P_N\langle t\rangle & P_N\langle tx^T\rangle \end{pmatrix} \begin{pmatrix} 1 & P_N\langle x\rangle^T \\ P_N\langle x\rangle & P_N\langle xx^T\rangle \end{pmatrix}^{-1} \\
(\sigma_j^2)_{j=1}^d &= P_N \mathrm{diag}(\langle tt^T\rangle + \mu\mu^T + W\langle xx^T\rangle W^T - 2\langle t\rangle\mu^T - 2\langle tx^T\rangle W^T + 2\mu\langle x\rangle^T W^T).
\end{aligned}$$

The total time complexity is $O(Nq^2d)$. Note that $\mathrm{diag}(AB)$ is implemented efficiently in code as $\mathrm{rowSums}(A \odot B^T)$. We note that if there is no missing data, the complexity can be improved to $O(Nqd)$, but since typically $q \ll d$, $O(Nq^2d)$ is good.