Factor Analysis (FA): quick recap

To recap, the FA model is defined by a low-dimensional Gaussian latent variable, a linear mapping to higher dimensional observation space, and independent Gaussian noise with different variances along each dimension:

The model:

\[
\begin{align*}
    z & \sim \mathcal{N}(0, I_m) \\
    x & = Az + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \Psi) \\
    \text{or} & \quad x|z \sim \mathcal{N}(Az, \Psi)
\end{align*}
\]

where \( I_m \) denotes an \( m \times m \) identity matrix, \( A \) is a \( d \times m \) matrix that maps the latent space to the observation space, and the noise covariance is the diagonal matrix \( \Psi = \text{diag}(\sigma_1^2, \ldots, \sigma_d^2) \). The model parameters are \( \theta = \{A, \Psi\} \).

The marginal likelihood is also Gaussian:

\[
    x \sim \mathcal{N}(0, AA^\top + \Psi).
\]

Thus FA is basically a model of the covariance of the data, and seeks to represent it as the sum of a low-rank component and a diagonal component.

Recognition distribution

The recognition distribution follows from Bayes’ rule:

\[
p(z|x) \propto p(x|z)p(z) = \mathcal{N}(x|Az, \Psi) \cdot \mathcal{N}(z|0, I)
\]

we can "complete the square" to find the Gaussian distribution in \( z \) that results from normalizing the right-hand-side:

\[
p(z|x) = \mathcal{N}\left((A\Psi^{-1}A^\top + I)^{-1}A^\top\Psi^{-1}x, (A^\top\Psi^{-1}A + I)^{-1}\right)
\]

\[
= \mathcal{N}(\Lambda A^\top\Psi^{-1}x, \Lambda),
\]
where posterior covariance is denoted by \( \Lambda = (A^\top \Psi^{-1} A + I)^{-1} \), and the posterior mean is \( \mu = \Lambda A^\top \Psi^{-1} x \).

Note that the mean of the recognition distribution involves first projecting \( x \) onto the \( \Psi^{-1} \), a diagonal matrix containing the inverse of the variance along each component. This means that, when inferring the latent from \( x \), the components of \( x \) are downweighted in proportion to the amount of independent noise they contain.

**Derivation:**

We can derive the recognition distribution by completing the square in (eq. 15):

\[
N(x|Az, \Psi) \cdot N(z|0, I) \propto \exp \left( -\frac{1}{2} (Az - x)^\top \Psi^{-1} (Az - x) - \frac{1}{2} z^\top z \right)
\]

(8)

\[
\propto \exp \left( -\frac{1}{2} (z^\top (A^\top \Psi^{-1} A)z - 2z^\top A^\top \Psi^{-1} x + z^\top z) \right)
\]

(9)

\[
= \exp \left( -\frac{1}{2} (z^\top (A^\top \Psi^{-1} A + I_m)z - 2z^\top A^\top \Psi^{-1} x) \right)
\]

(10)

then substituting \( \Lambda^{-1} = (A^\top \Psi^{-1} A + I) \),

\[
= \exp \left( -\frac{1}{2} (z^\top \Lambda^{-1} z - 2z^\top A^\top \Psi^{-1} x) \right)
\]

(11)

\[
\propto \exp \left( -\frac{1}{2} (z - \Lambda A^\top \Psi^{-1} x)^\top \Lambda^{-1} (z - \Lambda A^\top \Psi^{-1} x) \right)
\]

(12)

\[
\propto N(z | \Lambda A^\top \Psi^{-1} x, \Lambda)
\]

(13)

### 3 EM for Factor Analysis

Suppose we have a dataset consisting of \( N \) samples \( X = \{x_i\}_{i=1}^N \). The FA model (like the MoG model we considered in the last section) considers these samples to be independent \textit{a priori}, so

\[
\log p(X) = \log \left( \prod_{i=1}^N p(x_i) \right) = \sum_{i=1}^N \log p(x_i).
\]

(14)

The negative free energy \( F \) is defined using a variational distribution \( q(z_i|\phi_i) \) that describes the conditional distribution over each latent given the corresponding \( x_i \). It can also be written as a sum of terms:

\[
F(\phi, \theta) = \sum_{i=1}^N \left[ \int q(z_i|\phi_i) \log p(x_i, z_i|\theta) dz_i - \int \log q(z_i|\phi_i) dz_i \right]
\]

(15)

Here we will use a Gaussian variational distribution \( q(z_i|\phi_i) = \mathcal{N}(\mu_i, \Lambda_i) \), meaning that the variational parameters are \( \phi_i = \{\mu_i, \Lambda_i\} \) for each sample. As we will see in a moment, the covariance \( \Lambda \) does not depend on \( x_i \), and therefore we need not index it by \( i \) for each sample.
3.1 E-step

The E step involves setting \( q(z_i|\phi_i) \) equal to the conditional distribution of \( z_i \) given the data and current parameters \( \theta = \{A, \Psi\} \). That is, the recognition probabilities given above in (eq. 7).

Thus in the E-step we compute, first, for all samples (using the current \( A \) and \( \Psi \)):

\[
\Lambda = (A^\top \Psi^{-1} A + I)^{-1}
\]

(16)

Then we compute the conditional mean for each latent

\[
u_i = \Lambda A^\top \Psi^{-1} x_i
\]

(17)

At the end of this procedure we have a collection of \( q \) distributions, one for each sample:

\[
q(z_i|\phi_i) = \mathcal{N}(z_i|\mu_i, \Lambda)
\]

(18)

3.2 M-step

The M-step involves updating the parameters \( \theta = \{A, \Psi\} \) using the current variational distributions \( \{q(z_i|\phi_i)\} \). To do this, we compute the integral over \( z_i \) to evaluate the negative free energy (technically we might consider this part of the “E” step, since this is computing the expectation of the total-data log-likelihood). We will then differentiate with respect to the model parameters and solve for the maxima.

Plugging in \( q \) to the negative free energy gives:

\[
F = \sum_{i=1}^{N} \left( \int q(z_i|\phi_i) \log p(x_i|z_i, \theta) dz_i \right) + \text{const}
\]

(19)

\[
= \sum_{i=1}^{N} \left( \int \mathcal{N}(z_i|\mu_i, \Lambda) \log \mathcal{N}(x_i|A z_i, \Psi) dz_i \right) + \text{const}
\]

(20)

\[
= \sum_{i=1}^{N} \left( \int \mathcal{N}(z_i|\mu_i, \Lambda) \left[ -\frac{1}{2} \log |\Psi| - \frac{1}{2} (x_i - A z_i) \Psi^{-1} (x_i - A z_i) \right] dz_i \right) + \text{const}
\]

(21)

\[
= -\frac{N}{2} \log |\Psi| - \frac{1}{2} (x_i - A \mu_i) \Psi^{-1} (x_i - A \mu_i) - \frac{1}{2} \text{Tr}[A^\top \Psi^{-1} A \Lambda],
\]

(22)

where in the last line we have used the Gaussian identity for taking expectations of a quadratic form.

Differentiating with respect to \( A \) and solving gives:

\[
\hat{A} = \left( \sum_{i=1}^{N} x_i \mu_i^\top \right) \left( \sum_{i=1}^{N} \mu_i \mu_i^\top \right)^{-1} + N \Lambda
\]

(23)

We can obtain a slightly nicer way to write this if to define the matrices:

\[
X = \begin{bmatrix}
-x_1- \\
\vdots \\
-x_N-
\end{bmatrix}, \quad U = \begin{bmatrix}
-\mu_1- \\
\vdots \\
-\mu_N-
\end{bmatrix}
\]

(24)
Then we have
\[ \hat{A}^\top = (U^\top U + N\Lambda)^{-1}U^\top X, \] (25)
which recalls the form of a MAP estimate in linear regression.

Differentiating with respect to \( \Psi^{-1} \) and solving gives update
\[
\hat{\Psi} = \text{diag} \left( \frac{1}{N} \sum_{i=1}^{N} (x_i - A\mu_i)(x - A\mu_i)^\top + AA^\top \right)
\] (26)
\[
= \text{diag} \left( \frac{1}{N} (X - UA^\top)(X - UA^\top) + AA^\top \right),
\] (27)
where \( \text{diag}(M) \) denotes taking only the diagonal elements of the argument \( M \). An equivalent formula (from [1] sec. 12.2.4) is
\[
\hat{\Psi} = \text{diag} \left( \frac{1}{N} \sum_{i=1}^{N} x_i x_i^\top - \frac{1}{N} \sum_{i=1}^{N} x_i \mu_i^\top A^\top \right)
\] (28)
\[
= \text{diag} \left( \frac{1}{N} X^\top X - \frac{1}{N} X^\top U A^\top \right). \] (29)

4 Probabilistic PCA

The probabilistic principal components model (introduced in 1999 by [2]) provides a connection between PCA and FA. In particular, it provides an explicit probabilistic model of the data (like FA and unlike PCA), and can be estimated in closed form from the eigenvectors and eigenvalues of the sample covariance matrix (like PCA and unlike FA).

The model can be defined by constraining the matrix of noise variances to be a multiple of the identity,
\[ \Psi = \sigma^2 I, \] (31)
so that all neurons (outputs) have the same amount of noise. This makes the model less flexible than the FA model, but the advantage is that we don’t have to run EM to estimate it.

To derive the closed-form estimates for the PPCA model, we will parametrize the loadings matrix \( A \) by its left singular vectors and singular values:
\[ A = US = \begin{bmatrix} | & | & \cdots & | \\ \bar{u}_1 & \cdots & \bar{u}_m \\ | & | & \cdots & | \end{bmatrix} \begin{bmatrix} s_1 \\ \cdots \\ s_m \end{bmatrix}, \] (32)
where \( \{\bar{u}_i\} \) denote singular vectors and \( \{s_i\} \) denote singular values, for \( i \in \{1, \ldots m\} \).

Let \( \text{cov}(X) = \Sigma \) be the sample covariance of the data, and let \( \{\bar{b}_i\} \) and \( \{\lambda_i\} \) denote its eigenvectors and eigenvalues, respectively, for \( i \in \{1, \ldots, d\} \).

Then the PPCA model parameters can be estimated as follows:
1. Singular vectors of $A$: for $i \in \{1, \ldots, m\}$:

$$\hat{u}_i = \vec{b}_i, \quad (33)$$

2. Noise variance: given by average of lowest $m - d$ eigenvalues of sample covariance:

$$\hat{\sigma}^2 = \frac{1}{d - m} \sum_{i=m+1}^{d} \lambda_i. \quad (34)$$

3. Singular values of $A$: for $i \in \{1, \ldots, m\}$,

$$\hat{s}_i = \sqrt{\lambda_i - \hat{\sigma}^2} \quad (35)$$

References
