CS281A/Stat241A: Statistical Learning Theory

Factor Analysis and Kalman Filtering (11/2/04)

Lecturer: Michael I. Jordan *Scribes: Byung-Gon Chun and Sunghoon Kim*

1 Factor Analysis

Factor analysis is used for dimensionality reduction. It finds a subspace where the data lie. Figure 1 shows the geometry of the factor analysis model. In the figure, μ is the mean or the centroid of manifold, Λ is the coordinate system, and Ψ is the noise variance (represented by the sphere in the figure). The latent Gaussian variable $X_n \sim \mathcal{N}(0, I)$, and the observed variable $Y_n \sim \mathcal{N}(\mu + \Lambda x_n, \Psi)$ where Ψ is a diagonal covariance matrix. When λ_1 and λ_2 are basis vectors, $\Lambda = (\lambda_1\ \lambda_2)$. $\hat{\mu} + \hat{\Lambda}\hat{x_n}$ is the expectation of Y_n.

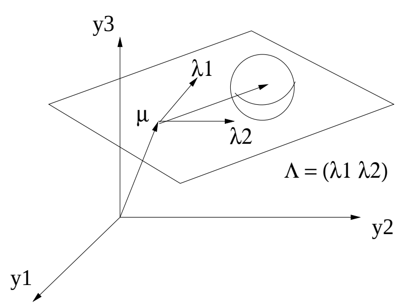

Figure 1: The geometry of the factor analysis model.

Why do we need dimensionality reduction? One application of factor analysis is compression. In the k-means algorithm, we can represent data with centroid coordinates and residuals. Likewise, with factor analysis, we can compress data using the coordinates of the subspace and residuals. Also, it can be used to predict regression. In high dimensions, there can be a large amount of noise. With dimensionality reduction, we can perform prediction with a small number of states, in effect reducing the impact of the noise.

2 Principal Component Analysis

Principal component analysis (PCA) finds the direction of maximal variance, and is a more simplistic model than factor analysis. We have data $\mathcal{D} = \{y_1, ..., y_N\}$. First, we center the data, that is, subtract the sample mean from data. Let the modified data be $\tilde{y_n}$. $\tilde{y_n} = y_n - \frac{1}{N}\sum_{n=1}^{N} y_n$. Then, we find θ to maximize $J = \sum_{n=1}^{N} \left(\theta^T \tilde{y_n}\right)^2$ such that $\|\theta\| = 1$. This is a simple optimization problem. Let's manipulate J to achieve θ to maximize J.

$$J = \sum_{n=1}^{N} \left(\theta^T \tilde{y_n}\right)^2 \tag{1}$$

$$
\begin{aligned}
&= \sum_{n=1}^{N}(\theta^T \tilde{y_n}\tilde{y_n}^T\theta) && (2)\\
&= \theta^T(\sum_{n=1}^{N}\tilde{y_n}\tilde{y_n}^T)\theta && (3)\\
&= \theta^T Y\theta && (4)
\end{aligned}
$$

$\hat{\theta}$ that maximizes J meets $Y\hat{\theta} = \lambda\hat{\theta}$. As you can see, when λ is the largest eigenvalue, $\hat{\theta}$ is the first eigenvector.

We compare factor analysis (FA) with principal component analysis (PCA). FA is insensitive to scale factors. Since FA captures the underlying coordinate systems, FA does not change the direction of projections when Y^i has different variance. The uncertainty is given by covariance matrix in FA. When Y_n is far off from the subspace, the covariance increases and the sphere in Figure 1 grows. On the contrary, PCA is sensitive to scale factors, since the eigenvectors are affected by scale factors. Whether to use FA or PCA depends on goals. If the goal is compression, the scale factor does not matter much, and PCA can be effective.

3 Maximum Likelihood Estimation of the Factor Analysis Model

In this section, we derive the maximum likelihood estimation of the factor analysis model. $\widehat{\mu}_{ML}$ is the sample mean and we subtract $\widehat{\mu}_{ML}$ from data. The graphical model is presented in Figure 2. In the model, unconditional mean and unconditional variance are as follows.

$$
\text{unconditional mean} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{5}
$$

$$
\text{unconditional covariance} = \begin{pmatrix} I & \Lambda^T \\ \Lambda & \Lambda\Lambda^T + \Psi \end{pmatrix} \tag{6}
$$

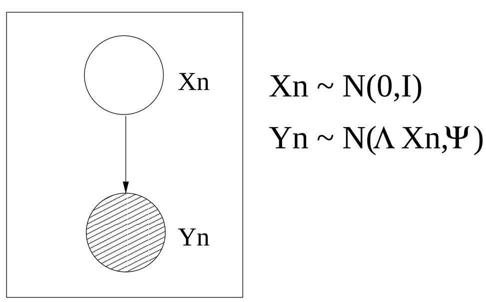

Figure 2: The factor analysis model as a graphical model.

Given complete data, the complete log likelihood is a product of Gaussian distributions. The complete log likelihood is:

$$
\ell_c(\Lambda, \Psi) = -\frac{N}{2}\log|\Psi| - \frac{N}{2}tr(S\Psi^{-1}) \tag{7}
$$

$$
\text{where } S = \frac{1}{N}\sum_{n=1}^{N}(Y_n - \Lambda X_n)(Y_n - \Lambda X_n)^T \tag{8}
$$

Now, we take the conditional expectation of the complete log likelihood, conditioning on the observed data and the current parameter vector. $<\dot{}>$ denotes the conditional expectation. The expected complete log likelihood is:

$$< \ell_c(\Lambda, \Psi) >= -\frac{N}{2}\log|\Psi| - \frac{N}{2}tr(< S > \Psi^{-1}) \tag{9}$$

$$< S >= \frac{1}{N}\sum_{n-1}^{N}\{Y_nY_n^T - Y_n < X_n^T > \Lambda^T - \Lambda < X_n > Y_n^T + \Lambda < X_nX_n^T > \Lambda^T\} \tag{10}$$

In the E-step of the algorithm, sufficient statistics we need are $< X_n >$ and $< X_nX_n^T >$.

$$< X_n >= E[X_n|Y_n] = \Lambda^T(\Lambda\Lambda^T + \Psi)^{-1}Y_n \tag{11}$$

$$< X_nX_n^T >= Var(X_n|Y_n) + E[X_n|Y_n]E[X_n|Y_n]^T = I - \Lambda^T(\Lambda\Lambda^T + \Psi)^{-1}\Lambda + E[X_n|Y_n]E[X_n|Y_n]^T \tag{12}$$

In the M-step of the algorithm, we update Λ and Ψ using the updated sufficient statistics.

$$\Psi^{(t+1)} = (\sum_{n=1}^{N} Y_n < X_n^T >)(\sum_{n=1}^{N} < X_nX_n^T >)^{-1} \tag{13}$$

Recall $\hat{\theta} = (X^TX)^{-1}X^TY$.

$$\Psi^{(t+1)} = \frac{1}{N}diag\{\sum_{n=1}^{N}(Y_n - \Lambda^{(t+1)} < X_n >)Y_n^T\} \tag{14}$$

Here, $\Lambda^{(t+1)} < X_n >$ is the best guess and $Y_n - \Lambda^{(t+1)} < X_n >$ is the residual. In the above, we derive the E-step and the M-step for estimation.

4 State Space Models

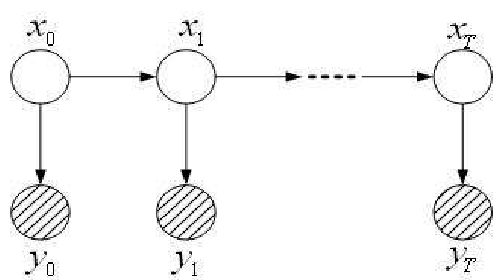

Figure 3: The SSM as a graphical model

The state nodes in the factor analysis model are continuous, vector-valued nodes endowed with a Gaussian probability distribution. To develop a dynamical generalization of the factor analysis model we must represent the transition between the nodes at successive moments in time. Perhaps the simplest choice that we can make is to allow the mean of the state at time $t + 1$ to be a linear function of the state at time t.

$$x_{t+1} = Ax_t + Gw_t$$

where w_t is a "noise" term — a Gaussian random variable that is independent of w_s for $s < t$, and thus independent of x_t. We assume that w_t has zero mean and covariance matrix Q and endow the initial state,

x_0, with a Gaussian distribution having mean 0 and covariance P_0 The assumption of zero mean is without loss of generality (a non-zero mean gives rise to a deterministic component that can added to the probabilistic solution)

$$w_t \sim \mathcal{N}(0, Q) \qquad x_0 \sim \mathcal{N}(0, P_0)$$

Given that the sum of Gaussian variables is Gaussian, we have that $x_t + 1$ is indeed Gaussian.

$$x_{t+1} \mid x_t \sim \mathcal{N}(Ax_t, GQG^T)$$

In the factor analysis model, the output is endowed with a Gaussian distribution having a mean that is a linear function of the state. We continue to use this model for the output of the SSM:

$$y_t = Cx_t + v_t$$

where v_t is a Gaussian random variable with zero mean and covariance matrix R.

$$v_t \sim \mathcal{N}(0, R)$$

Conditional on x_t, Y_t is a Gaussian random variable with mean Cx_t and covariance R.

$$y_t \mid x_t \sim \mathcal{N}(Cx_t, R)$$

The inference problem for the SSM is the same as it was for the HMM — that of calculating the posterior probability of the states given an output sequence. Based on our experience with the HMM, we hope to be able to calculate such posterior probabilities recursively.

$$\begin{array}{ll} \text{Goal:} & \text{Compute } p(x_t \mid y_1, \ldots, y_t) \\ & \text{(sum-product, a.k.a., Kalman filters)} \\ & \text{Compute } p(x_t \mid y_1, \ldots, y_T) \\ & \text{(sum-product, a.k.a. Rauch-Tung-Striebel, Smoothing)} \\ & \text{Estimate } P_0, A, C, G, Q, R \\ & \text{(EM)} \end{array} \tag{15}$$

5 Filtering

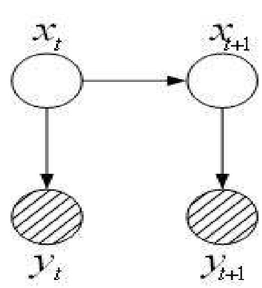

Figure 4: A fragment of a SSM

The problem is to calculate an estimate of the state x_t based on a partial output sequence $y_0, \ldots, y_t$.

$$\text{Compute } p(x_{t+1} \mid y_0, \ldots, y_t, y_{t+1})$$

Sums of Gaussian variables are Gaussian, and thus, considering all of the variables in the SSM jointly, we have a (large) multivariate Gaussian distribution. Conditionals of Gaussians are Gaussian and thus the probability distribution $p(x_t \mid y_0, \ldots, y_t)$ must be Gaussian.

We write $\hat{x}_{t|t}$ to denote the mean of x_t conditioned on the partial sequence $y_0, \ldots, y_t$. The covariance matrix of x_t conditioned on $y_0, \ldots, y_t$. is denoted $P_{t|t}$; thus:

$$\begin{aligned} \hat{x}_{t|t} &\triangleq \quad E[x_t \mid y_0, \ldots, y_t] \\ P_{t|t} &\triangleq \quad E[(x_t - \hat{x}_{t|t})(x_t - \hat{x}_{t|t})^T \mid y_0, \ldots, y_t] \end{aligned}$$

We assume that we have already calculated $p(x_t \mid y_0, \ldots, y_t)$ that is, we have calculated $\hat{x}_{t|t}$ and $p(x_t \mid y_0, \ldots, y_t)$. We wish to carry this distribution forward into the fragment on the right, where we condition on $y_0, \ldots, y_{t+1}$. We decompose the transformation into two steps:

$$\begin{array}{ll} \text{time update:} & p(x_t \mid y_0, \ldots, y_t) \rightarrow p(x_{t+1} \mid y_0, \ldots, y_t) \\ \text{measurement update:} & p(x_{t+1} \mid y_0, \ldots, y_t) \rightarrow p(x_{t+1} \mid y_0, \ldots, y_{t+1}) \end{array}$$

Thus, in the time update step, we simply propagate the distribution forward one step in time, calculating the new mean and covariance based on the old mean and covariance, but conditioning on no new measurements (i.e., no new output). In the measurement update step, we incorporate the new measurement y_{t+1} and update the probability distribution for x_{t+1}. The overall result is a transformation from $\hat{x}_{t|t}$ and $P_{t|t}$ to $\hat{x}_{t+1|t+1}$ and $P_{t+1|t+1}$.
Given $\hat{x}_{t|t}$, compute $\hat{x}_{t+1|t}$

$$\begin{aligned} \hat{x}_{t+1|t} = \quad & E[x_{t+1} \mid y_0, \ldots, y_t] \\ & = E[Ax_t + Gw_t \mid y_0, \ldots, y_t] \\ & = AE[x_t \mid y_0, \ldots, y_t] \\ & = A\hat{x}_{t|t} \end{aligned}$$