# CS229 Lecture notes

原作者：Andrew Ng (吴恩达)，翻译：CylcerUser

## 1 因子分析（Factor analysis）

如果有一个高斯模型混合（a mixture of several Gaussians）而来的数据集 $x^{(i)} \in R^n$，那么就可以用期望最大化算法（EM algorithm）来对这个混合模型（mixture model）进行拟合。这种情况下，对于有充足数据（sufficient data）的问题，我们通常假设可以从数据中识别出多个高斯模型结构（multiple-Gaussian structure）。例如，如果我们的训练样本集合规模（training set size）m 远远大于（significantly larger than）数据的维度（dimension）n，就符合这种情况。然后来考虑一下反过来的情况，也就是 $n$ 远远大于 $m$，即 $n > m$。在这样的问题中，就可能单独一个高斯模型来对数据建模很艰难，更不用说了高斯模型的混合模型了。由于 $m$ 个数据点所张开（span）的只是一个 $n$ 维空间 $R^n$ 的低维度子空间（low-dimensional subspace）, 如果用高斯模型（Gaussian）对数据进行建模，然后还是用常规的最大似然估计 (usual maximum likelihood estimators) 来估计（estimate）平均值（mean）和方差（covariance)，得到的则是：

$$\mu = \frac{1}{m}\sum_{i=1}^{m} x^{(i)}$$

$$\Sigma = \frac{1}{m}\sum_{i=1}^{m}(x^{(i)}-\mu)(x^{(i)}-\mu)^T$$

we would find that the matrix $\Sigma$ is singular.This means that $\Sigma^{-1}$ does not exist,and $1||\Sigma|^{\frac{1}{2}} = 1/0$.But both of these terms are needed in computing the usual density of a multivariate Gaussian distribution.Another way of the stating this difficulty is that maximum likelihood estimate of the parmeters result in a Gaussian that places all of its probability in the affine space spanned by the data [1]。,and the corresponds to a singular convariance matrix.

我们会发现这里的 $\Sigma$ 是一个奇异（singular）矩阵。这就意味着其逆矩阵 $\Sigma$ 不存在，而 $1/|\Sigma|^{\frac{1}{2}} = 1/0$。但这几个变量都是必须的，要用来计算一个多元高斯函数分布（multivariate Gaussian distribution）的常规密度函数（usual density）。还可以用另外一种方法来讲述清楚这个难题，也就是对参数（parameters）的最大似然估计（maximum likelihood:estimates）会产生一个高斯分布（Gausssian）, 其概率分分布在有样本数据所张成的放射空间（affine space）中，对应着一个奇异的协方差矩阵（singular convariance matrix)。

通常情况下，除非 m 比 n 大出相当多（some reasonable amount）, 否则最大似然估计（maximum likelihood estimates）得到的均值（mean）和方差（convariance）都会很差（quite poor)。尽管如此，我们还是希望能用自己已有的数据，拟合出一条合理（reasonable）的高斯模型（Gaussian model）, 而且希望能够识别出数据中的某些有意义的协方差结构（convariance structure)。那这可怎们办？

在接下来的这一部分内容里，我们首先回顾一下对 $\Sigma$ 的两个可能约束（possible restrictions)，这来那个约束条件能让我们使用小规模数据来拟合 $\Sigma$，但是都不能就我们的问题给出让人满意的解 (staisfactory solution)。然后接下来我们要讨论一下高斯模型的边界和条件分布。最后，我们会讲下因子分析模型 (factor analysis model)，以及对应的期望最大化算法（EM algorithm)。

[1] This is the set of points x satisfying $x = \sum_{i=1}^{m} \alpha_i x^{(i)}$,for some $\alpha_i$'s so that $\sum_{i=1}^{m} \alpha_i = 1$



零基础自学人工智能，过来人帮你少走弯路，微信公众号：learningthem

## 1.1 $\Sigma$ 的约束条件

如果我们没有充足的数据来拟合一个完整的协方差矩阵 (covariance matrix)，就可以对矩阵空间 $\Sigma$ 给出某些约束条件（restrictions）。例如，我们可以选择拟合一个对角（diagonal）的协方差矩阵 $\Sigma$。这样，读者很容易就能验证这样的一个协方差矩阵的最大似然估计（maximum lilklihoof estimate），可以有对角矩阵（diagonal matrix）$\Sigma$ 满足：

$$\Sigma_{jj}=\frac{1}{m}\Sigma_{i=1}^{m}(x_j^i-\mu_j)^2$$

因此，$\Sigma$ 就是对数据中的 $j$ 个坐标位置的方差的经验估计（empeirical estimate）。

Recall that the contours of a Gaussian density are ellipses. A diagonal $\Sigma$ corresponds to a Gaussian where the major axes of these ellipses are axis-aligned.

回忆一下，高斯模型的密度的形状是椭圆形的。对角矩阵 $\Sigma$ 对应的就是椭圆的长轴（major axes）对齐（axis-aligned）的高斯模型。

有时候，我们还要对这个协方差矩阵（convariance matrix）给出进一步的约束，不仅设为对角的（major axes），还要求所有的对角元素（diagonal entries）都相等。这时候，就有 $\Sigma=\sigma^2 I$, 其中 $\sigma^2$ 是我们控制的参数。对于这个 $\sigma^2$ 的最大似然估计则为：

$$\sigma^2=\frac{1}{mn}\Sigma_{j=1}^{n}\Sigma_{i=1}^{m}(x_j^i-\mu_j)^2$$

这种模型对应的是密度函数为圆形轮廓的高斯模型（在二维空间也就是平面中是圆形，在更高维度当中就是球（spheres）或者超球体（hyperspheres））。如果我么对数据要拟合一个完整的，不受约束的（unconstrained ）协方差矩阵 $\Sigma$, 就必须满足 $m\ge n+1$，这样才使得对 $\Sigma$ 的最大似然估计不是奇异矩阵（singular matrix）。在上面提到的两个约束条件之下，只要 $m\ge 2$，我们就能获得非奇异的（non-singular）$\Sigma$。

然而，讲 $\Sigma$ 限定为对角矩阵，也就意味着对数据中不同坐标（coordinates）的 $x_i, x_j$ 建模都不相关（uncorrelated），且互相独立（independent）。通常，还是从样本数据里获得某些有趣的相关信息结构比较好。如果使用上面对 $\Sigma$ 的某一种约束，就可能没有办法获取这些信息了。在本章讲义里面，我们会提到因子分析模型（factor analysis model），这个模型使用的参数比对角矩阵 $\Sigma$ 更多，而且能从数据中获取某些相关性的信息（captures some correlations），但也不能对完整的协方差矩阵（full covariance matrix）进行拟合。

## 1.2 多重高斯模型（Gaussians）的边界（Marginal）和条件（Conditional）

在讲解因子分析之前（factor analysis ）之前，我们要说一下一个联合多元高斯分析（joint multivariate Gaussian distribution）下的随机变量（random variables）的条件（conditional）和边界（marginal 分布（distributions）。

假如我们有一个值为向量的随机变量（vector-valued random variable）:

$$x=\begin{bmatrix}x_1\\x_2\end{bmatrix}$$

其中 $x_1\in R^r, x_2\in R^s$，因此 $x\in R^{r+s}$。设 $x\sim N(\mu,\Sigma)$, 即以 $\mu$ 和 $\Sigma$ 为参数的正太分布，则这两个参数为：

$$\mu=\begin{bmatrix}\mu_1\\\mu_2\end{bmatrix},\Sigma=\begin{bmatrix}\Sigma_{11}&\Sigma_{12}\\\Sigma_{21}&\Sigma_{22}\end{bmatrix}$$



零基础自学人工智能，过来人帮你少走弯路，微信公众号：learningthem

其中，$\mu \in R^r, \mu_2 \in R^s, \Sigma_{11} \in R^{r\times r}, \Sigma_{12}^{r\times s}$ ，依次类推。由于协方差矩阵 (convariance matrix) 是对称的 (symmetric)，所以有 $\Sigma_{12} = \Sigma_{21}^T$。

基于我们的假设，$x_1$ 和 $x_2$ 是联合多元高斯分布 (jointly multivariate Gaussian)。那么 $x_1$ 的边界分布是什么？不难看出 $x_1$ 的期望 $E[x_1] = \mu_1$，而协方差 $Cov(x_1) = E[(x_1 - \mu_1)(x_1 - \mu_1)] = \Sigma_{11}$，接下来为了验证后面一项成立，要用 $x_1$ 和 $x_2$ 的联合方差的概念。

$$
\begin{aligned}
Cov(x) &= \Sigma \\
&= \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \\
&= E[(x-\mu)(x-u)^T] \\
&= E\left[ \begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix} \begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix}^T \right] \\
&= E\left[ \begin{pmatrix} (x_1 - \mu_1)(x_1 - \mu_1)^T \\ (x_2 - \mu_2)(x_1 - \mu_1)^T \end{pmatrix} \begin{pmatrix} (x_1 - \mu_1)(x_2 - \mu_2)^T \\ (x_2 - \mu_2)(x_2 - \mu_2)^T \end{pmatrix}^T \right]
\end{aligned}
$$

Matching the upper-left sub blocks in the matrices in the second and the last lines above gives the result.

高斯分布的边界条件（marginal distributions）本身也是高斯分布。所以我么就可以给出一个正太分布 $x_1\ N(\mu_1, \Sigma_{11})$ 来作为 $x_1$ 的边界分布（marginal distributions）。

此外，我们还可以提出领一个问题，给定 $x_2$ 的情况下 $x_1$ 的条件分布是什么呢？通过参考多元高斯分布的定义，就能得到条件分布 $x_1|x_2\ N(\mu_{1|2}, \Sigma_{1|2})$ 为：

$$
\mu_{1|2} = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2) \quad (1) \quad \Sigma_{1|2} = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} \quad (2)
$$

在下一节对因子分析模型 (factor analysis model) 的讲解中，上面这些公式就很有用了，可以帮助高斯分布的条件和边界分布（conditional and marginal distributions）

## 1.3 因子分析模型（Factor analysis model）

在因子分析模型（Factor analysis model）中，我们定制在 $(x, z)$ 上的一个联合分布，如下所示，其中 $z \in R^k$ 是一个潜在随机变量（latent randon variable）：

$$
\begin{aligned}
z\ N(0, I) \\
x|z\ N(\mu + \Lambda z, \Psi)
\end{aligned}
$$

上面的式子中，我们这个模型中的参数是向量 $\mu \in R^n$，矩阵 $\Lambda \in R^{n\times k}$, 以及一个对角矩阵 $\Psi \in R^{n\times n}$。$k$ 的值通常都选择比 $n$ 小一点的。

这样，我们就设想每个数据点 $x^i$ 都是通过一个 $k$ 维度的多元高斯分布 $z^i$ 中取样获得的。然后，通过计算 $\mu + \Lambda z^i$, 就可以映射到实数域 $R^n$ 中的一个 $k$ 维仿射空间 $(k - dimensionalaffinespace)$，在 $\mu + \Lambda z^{(i)}$ 上加上协方差 $\Psi$ 噪音，就得到了 $x^{(i)}$。



零基础自学人工智能，过来人帮你少走弯路，微信公众号：learningthem

反过来，咱们也就可以定义因子分析模型（factor anaylsis model），使用下面的设定：

$$
\begin{aligned}
& z\ N(0, I) \\
& \xi\ N(0, \psi) \\
& x = \mu + \Lambda z + \xi
\end{aligned}
$$

其中的 $\xi$ 和 $z$ 是相互独立的。然后咱们来确切地看看这个模型定义的分布（distribution our）。其中，随机变量 $z$ 和 $x$ 有一个联合高斯分布（joint Gaussian distribution）：

$$
\begin{bmatrix} z \\ x \end{bmatrix}\ N(\mu_{zx}, \Sigma)
$$

然后咱们要找到 $\mu_{zx}$ 和 $\Sigma$。

我们知道 $z$ 的期望 $E(z) = 0$，这是因为 $z$ 服从的是均值为 0 的正太分布 $z\ N(0, I)$。此外我们还知道：

$$
\begin{aligned}
E[x] &= E[\mu + \Lambda z + \xi] \\
&= \mu + \Lambda E[z] + E[\xi] \\
&= \mu
\end{aligned}
$$

综合以上这些条件，就得到了：

$$
\mu_{zx} = \begin{bmatrix} \vec{0} \\ \mu \end{bmatrix}
$$

下一步就是要找出 $\Sigma$，我们需要计算出 $\Sigma = E[x - E[Z](z - E(x))^T]$（矩阵 $\Sigma$ 左上部分（upper-left block）)，$\Sigma_{xx} = E[(z - E(z))(x - E(x))^T]$（右上部分（upper-right block）），以及 $E[(x - E(x))(x - E(x))^t]$（右下部分（lowe-right block））。

由于 $z$ 是一个正太分布 $z\ N(0, 1)$，很容易就能知道 $\Sigma_{zz} = Cov(z) = I$，另外：

$$
\begin{aligned}
E[(z - E(z))(x - E[x])^T] &= E[z(\mu + \Lambda z + \xi - \mu)^T] \\
&= E[zz^T] + E[z\xi^T] \\
&= \Lambda^T
\end{aligned}
$$

在上面的最后一步中，使用到了结论 $E[zz^T] = Cov(z)$（因为 z 的均值为 0，而且 $E[z\xi] = E[z]E[\xi^T]$)（因为 $z$ 和 $\xi$ 相互独立，因此乘积（product）的期望（expectation）等于期望的乘积）。

同样的方法，我们可以用下面的方法论来找到 $\Sigma_{xx}$:

$$
\begin{aligned}
E[(x - E(x))(x - E[x])^T] &= E[(\mu + \Lambda z + \xi - \mu)(\mu + \Lambda z + \xi - \mu)^T] \\
&= E[\Lambda z z^T \Lambda^T] + E[\xi z^T \Lambda^T + \Lambda z \xi^T + \xi \xi^T] \\
&= \Lambda E[zz^T]\Lambda^T + E[\xi\xi^T] \\
&= \Lambda\Lambda^T + \Psi
\end{aligned}
$$



零基础自学人工智能，过来人帮你少走弯路，微信公众号：learningthem

把上面综合到一起，就得到了:

$$\mu_{zx} = \begin{bmatrix} z \\ hx \end{bmatrix} \ N\left(\begin{bmatrix} \vec{0} \\ \mu \end{bmatrix} \ \begin{bmatrix} I & \Lambda^T \\ \Lambda & \Lambda\Lambda^T + \Psi \end{bmatrix}\right)$$

因此，我们还能发现 $x$ 边界分布（marginal distribution）为：$x\ N(\mu\ \Lambda\Lambda^T + \Psi)$。所以，给定一个训练样本集合 $\{x^{(i);i=1\cdots,m}\}$，参数（parameters）的最大似然估计函数的对数函数（log likelihood），就可以写为：

$$l(\mu, \Lambda, \Psi) = log\prod_{i=1}^{m} \frac{1}{(2\pi)^{\frac{n}{2}}|\Lambda\Lambda^T + \Psi|^{\frac{1}{2}}} exp(-\frac{1}{2}(x^{(i)-\mu})^T(\Lambda\Lambda^T + \Psi)^{-1}(x^i - \mu))$$

为了进最大似然估计，我们就要最大化上面这个参数的函数。但确切地对上面这个方程进行最大化，是很艰难的，不信你自己试试哈，而且我们都知道没有算法能够以封闭形式（closed-form）来实现对大化。所以，我们就该用期望最大化算法 (EM algorithm)。下一节里面，咱们就来推导一下针对因子分析模型（factor analysis）的期望最大化算法（EM）

### 1.4 针对因子分析模型（factor analysis）的期望最大化算法（EM））

EM 算法步骤的推导很简单。只需要计算出来 $Q_i(z^i) = p(z^{(i)}|x^{(i)}; \mu, \Lambda, \Psi)$。把等式 (3) 当中给出的分布带入到方程 $(1-2)$，来找出一个高斯分布的条件分布，我们就能发现 $z^{(i)|x^{(i)}}; \mu, \Lambda, \Psi \sim N(\mu_{z^i}, \Sigma_{z^{(i)}|x^i})$，其中：

$$\begin{aligned} \mu_{z^{(i)}|x^{(i)}} &= \Lambda^T(\Lambda\Lambda^T + \Psi)^{-1}(x^{(i)-\mu}) \\ \Sigma_{z^{(i)}|x^{(i)}} &= I - \Lambda^T(\Lambda\Lambda^T + \Psi)^{-1}\Lambda \end{aligned}$$

所以通过对 $\mu_{z^{(i)}|x^i}$ 和 $\Sigma_{z^{(i)}|x^i}$，进行这样的定义，就能得到：

$$Q_i(z^i) = \frac{1}{(2\pi)^{\frac{k}{2}}|\Sigma_{z^{(i)}|x^{(i)}}|^{\frac{1}{2}}} exp(\frac{-1}{2}(z^{(i)} - \mu_{z^{(i)}|x^{(i)}})^T\Sigma^{-1}_{z^{(i)}|x^{(i)}}(z^{(i)} - \mu_{z^{(i)}|x^{(i)}})$$

接下来就是 $M$ 步骤了。这里需要去最大化下面这个参数 $\mu, \Lambda, \Psi$ 的函数值：

$$\sum_{i=1}^{m}\int_{z^{(i)}} Q_i(z^{(i)})log\frac{p(x^{(i)}, z^{(i)}; \mu, \Lambda, \Psi)}{Q_i(z^{(i)})}dz^{(i)} \qquad (4)$$

我们在本文中仅仅对 $\Lambda$ 进行优化，关于 $\mu$ 和 $\Psi$ 的更新就作为练习留给自己进行推导了。

把等式 (4) 简化成下面的形式:

$$\begin{aligned} &\sum_{i=1}^{m}\int_{z^{(i)}} [logp(x^{(i)}|z^{(i)}; \mu, \Lambda, \Psi) + logp(z^{(i)}) - logQ_i(z^{(i)})]dz^{(i)} \qquad (5) \\ =&\sum_{i=1}^{m} E_{z^{(i)}\ Q_i}[logp(x^{(i)}|z^{(i)}; \mu, \Lambda, \Psi) + logp(z^{(i)}) - logQ_i(z^{(i)})] \qquad (6) \end{aligned}$$

上面的等式中，$z^{(i)}\ Q_i$ 这个下标（subscript），表示的意思是这个期望是从 $Q_i$ 中取得 $z^i$ 的。在后续的推导过程中，如果没有歧义的情况下，我们就会把这个省略掉。删除这些不依赖参数的项目后，我们就发现只需要最大化：



零基础自学人工智能，过来人帮你少走弯路，微信公众号：learningthem

我们对上面的函数进行关于 $\Lambda$ 的最大化。可见只有最后的一项依赖 $\Lambda$。求导数，同时利用下面几个结论：$Tra = a(for a \in R), TrAB = TrBA, \nabla_A TrABA^TC = CAB + C^TAB$, 就能得到：

$$\begin{aligned}
&\nabla_\lambda \sum_{i=1}^{m} -E[\frac{1}{2}(x^i-\mu-\Lambda z^i)^T\Psi^{-1}(x^i-\mu-\Lambda z^i)] \\
&= \sum_{i=1}^{m} \nabla_\Lambda E[-tr\frac{1}{2}{z^i}^T\Lambda^T\Psi^{-1}\Lambda z^i + tr{z^i}^T\Lambda^T\Psi^{-1}(x^i-\mu)] \\
&= \sum_{i=1}^{m} \nabla_\Lambda E[-tr\frac{1}{2}\Lambda^T\Psi^{-1}\Lambda z^i{z^i}^T + tr\Lambda^T\Psi^{-1}(x^i-\mu){z^i}^T] \\
&= \sum_{i=1}^{m} E[-\Psi^{-1}\Lambda z^i{z^i}^T + \Psi^{-1}(x^i-\mu){z^i}^T]
\end{aligned}$$

设置导数为 0, 然后简化，就能得到：

$$\sum_{i=1}^{m} \Lambda E_{z^i\ Q_i}[z^i{z^i}^T] = \sum_{i=1}^{m}(x^i-\mu)E_{z^i\ Q_i}[{z^i}^T]$$

接下来，求解 $\Lambda$, 就能得到：

$$\Lambda = (\sum_{i=1}^{m}(x^i-\mu)E_{z^i\ Q_i}[{z^i}^T])(\sum_{i=1}^{m}E_{z^i\ Q_i}[z^i{z^i}^T])^{-1} \quad (7)$$

有一个很有意思的地方需要注意，上面这个等式和用最小二乘线性回归推出的正则方程有密切的关系：

$$\theta^T = (y^TX)(X^TX)^{-1}$$

与之类似，这里的 $x$ 是一个关于 $z$（以及噪声 noise）的线性方程。考虑在 $E$ 步骤中对 $z$ 已经给出猜测，接下来就可以来尝试与 $x$ 和 $z$ 相关的位置量 $\Lambda$ 进行估计。接下来不出意料，我们就会得到某种类似正则方程的结果。然而，这个还是和利用对 $z$ 的最佳猜测（best guesses）进行最下二乘算法有一个很大的区别的；这一点我们很快就会看到。

为了完成 $M$ 步骤的更新，接下来我们要解出等式 (7) 当中的期望值 (values of the expectations)。由于我们定义 $Q_i$ 是均值为 $\mu_{z^i|x^i}$, 协方差为
$Sigma_{z^i|x^i}$ 的一个高斯分布，所以很容易得到：

$$\begin{aligned}
E_{z^i\ Q_i}[{z^i}^T] &= \mu_{z^i|x^i}^T \\
E_{z^i\ Q_i}[z^i{z^i}^T] &= \mu_{z^i|x^i}\mu_{z^i|x^i}^T + \Sigma_{z^i|x^i}
\end{aligned}$$

上面第二个等式的推导依赖与下面这个事实：对于一个随机变量 $Y$, 协方差 $Cov(Y) = E[YY^T] - E[Y]E[Y]^T$, 所以 $E[YY^T] = E[Y]E[Y]^T + Cov(Y)$。把这个带入道等式 (7)，就得到 $M$ 步骤中 $\Lambda$ 的更行规则：

$$\Lambda = (\sum_{i=1}^{m}(x^i-\mu)\mu_{z^i|x^i}^T)(\sum_{i=1}^{m}\mu_{z^i|x^i}\mu_{z^i|x^i}^T + \Sigma_{z^i|x^i})^{-1} \quad (8)$$

上面这个等式中，要特别注意等号右边这一侧的 $\Sigma_{z^i|x^i}$。这是一个根据 $z^i$ 给出的 $x^i$ 后验发分布 $p(z^i|x^i)$ 的协方差，而在 M 步骤中必须考虑到在这个后验分布中 $z^i$ 的不确定性。推导 EM 算法的一个常见错误就是在 E 步骤进行假设，只需要算出在随机变量 $z$ 的期望 $E[z]$，然后把这个值放到 M 步骤当中 $z$ 出现的每个地方进行



零基础自学人工智能，过来人帮你少走弯路，微信公众号：learningthem

优化。当然，这能解决简单问题，例如高斯混合模型，在因子模型的推导过程中，就同时需要 $E[zz^T]$ 和 $E[z]$; 而我们已经知道,$E[zz^T]$ 和 $E[z]E[z]^T$ 随着 $\Sigma_{z|x}$ 而变化。因此，在 M 步骤就必须要考虑到后验分布 $p(z^i|x^i)$ 中 $z$ 的协方差。

最后，我们还可以发现，在 M 步骤对参数 $\mu$ 和 $\Psi$ 的优化。不难发现其中的 $\mu$ 为：

$$\mu = \frac{1}{m}\sum_{i=1}^{m} x^i$$

由于这个值随着参数的变换而该改变（也就是说，和 $\Lambda$ 的更新不同，这里等式右侧不依赖 $Q_i(z^i) = p(z^i|x^i;\mu,\Lambda,\Psi)$）, 这个 $Q_i(z^i)$ 是依赖参数的，这个只需要计算一次就可以，在算法运行过程中，也不需要进一步更新。类似地，对角矩阵 $\Psi$ 也可以通过计算下面这个式子来获得：

$$\phi = \frac{1}{m}\sum_{i=1}^{m} x^i x^{i^T} - x^i \mu_{z^i|x^i}^T \Lambda^T - \Lambda \mu_{z^i|x^i} x^{i^T} + \Lambda(\mu_{z^i|x^i}\mu_{z^i|x^i}^T + \Sigma_{z^i|x^i})\Lambda^T$$

然后只需要设 $\Psi_{ii} = \Phi_{ii}$(也就是说，设 $\Psi$ 为一个仅仅包含矩阵 $\Phi$ 中对角线元素的对角矩阵)



零基础自学人工智能，过来人帮你少走弯路，微信公众号：learningthem