PCA, Probabilistic PCA, Kernel PCA, ICA

Boyang Liu

Abstract

Summary of PCA, PPCA, KPCA, ICA and other related approaches. Mainly based on PRML chapter 12 and FOML. We also includes some disentangled representation objective in deep learning.

1 PCA

Denote our data matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$, we want to find the subspace $\mathbf{Z} \in \mathbb{R}^{n \times k}$, where $k << d$, to compress the data.

There are two approaches to understand PCA. The first one is maximum variance principle and the second one is minimum reconstruction loss principle.

1.1 Variance Maximization

We first consider the case when $k = 1$. Let $\mathbf{u}_1 \in \mathbb{R}^k$ be the projection vector, the variance in subspace can be written as:

$$\frac{1}{N}\sum_{n=1}^{N}\left\{\mathbf{u}_1^{\mathrm{T}}\mathbf{x}_n - \mathbf{u}_1^{\mathrm{T}}\overline{\mathbf{x}}\right\}^2 = \mathbf{u}_1^{\mathrm{T}}\mathbf{S}\mathbf{u}_1,$$

$$\text{where } \mathbf{S} = \frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{x}_n - \overline{\mathbf{x}}\right)\left(\mathbf{x}_n - \overline{\mathbf{x}}\right)^{\mathrm{T}},$$

$$\overline{\mathbf{x}} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n.$$

The problem becomes to:

$$\max_{\mathbf{u}_1} \mathbf{u}_1^{\mathrm{T}}\mathbf{S}\mathbf{u}_1$$

The above optimization problem is ill posed since we can easily set $\mathbf{u}_1$ to be infinitely large or small to make the objective not upper bounded. Notice we only care about the direction of $\mathbf{u}_1$, we can constrain $\mathbf{u}_1$ to be unit length. Then, we have:

$$\max_{\mathbf{u}_1^T\mathbf{u}_1=1} \mathbf{u}_1^{\mathrm{T}}\mathbf{S}\mathbf{u}_1$$

Introducing Lagrangian multiplier, and take the derivative w.r.t. $\mathbf{u}_1$, we have:

$$\max_{\mathbf{u},\lambda_1} \mathbf{u}_1^{\mathrm{T}}\mathbf{S}\mathbf{u}_1 + \lambda_1\left(1 - \mathbf{u}_1^{\mathrm{T}}\mathbf{u}_1\right)$$

$$\mathbf{S}\mathbf{u}_1 = \lambda_1\mathbf{u}_1$$

$$\mathbf{u}_1^{\mathrm{T}}\mathbf{S}\mathbf{u}_1 = \lambda_1$$

Noticing $\mathbf{S}\mathbf{u}_1 = \lambda_1\mathbf{u}_1$ indicates that the largest value of variance is taken when λ_1 is the eigenvalue of $\mathbf{S}$ while $\mathbf{u}_1$ is the corresponding eigenvector, and $\mathbf{u}_1^{\mathrm{T}}\mathbf{S}\mathbf{u}_1$ indicates that λ_1 should be the largest eigenvalue.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

For the case $k = 1$, note every time, we are finding the orthogonal subspace, which requires that $\mathbf{u}_1$ and $\mathbf{u}_2$ be orthogonal to each other, which has the form $\mathbf{u}_1^T \mathbf{u}_2 = 1$. Add this constrain and follow the similar rule, we can get that $\mathbf{u}_2$ is the the eigenvector of second largest eigenvalue of centered sample matrix. When $k > 2$, it can be easily proofed by induction.

1.2 Minimum Reconstruction Error

Let $\mathbf{X}$ be a mean-centered data matrix, $\mathcal{P}_k$as the set of N-dimensional rank-k orthogonal projection matrix. PCA tries to seek the projection matrix, such that after projection and inverse-projection, the reconstruction error is minimum. Consider the following optimization problem (here $\mathbf{X} \in \mathbb{R}^{d \times n}$):

$$\min_{\mathbf{P} \in \mathcal{P}_k} \|\mathbf{PX} - \mathbf{X}\|_F^2,$$

where $\mathcal{P}_k = \mathbf{U_k}{\mathbf{U_k}}^T$, and $\mathbf{U_k} \in \mathbb{R}^{d \times k}$. After we solve this optimization problem, the subspace representation is $\mathbf{Y} = \mathbf{U}_k^\top \mathbf{X}$.

The above optimization problem can be formalized as following

$$\begin{aligned} \|\mathbf{PX} - \mathbf{X}\|_F^2 &= \operatorname{Tr}\left[(\mathbf{PX} - \mathbf{X})^\top (\mathbf{PX} - \mathbf{X})\right] = \operatorname{Tr}\left[\mathbf{X}^\top \mathbf{P}^2 \mathbf{X} - 2\mathbf{X}^\top \mathbf{PX} + \mathbf{X}^\top \mathbf{X}\right] \\ &= -\operatorname{Tr}\left[\mathbf{X}^\top \mathbf{PX}\right] + \operatorname{Tr}\left[\mathbf{X}^\top \mathbf{X}\right] \end{aligned}$$

, The above derivation use the fact that $\mathbf{P}$ is orthogonal matrix and projection matrix. ($\mathbf{P}^T\mathbf{P} = \mathbf{I}$, $\mathbf{PP} = \mathbf{P}$).

Remove the constant part, we have:

$$\operatorname*{argmin}_{\mathbf{P} \in \mathcal{P}_k} \|\mathbf{PX} - \mathbf{X}\|_F^2 = \operatorname*{argmax}_{\mathbf{P} \in \mathcal{P}_k} \operatorname{Tr}\left[\mathbf{X}^\top \mathbf{PX}\right],$$

By the constrain $\mathcal{P}_k = \mathbf{U_k}{\mathbf{U_k}}^T$, we get:

$$\operatorname{Tr}\left[\mathbf{X}^\top \mathbf{PX}\right] = \operatorname{Tr}\left[\mathbf{U}^\top \mathbf{X}\mathbf{X}^\top \mathbf{U}\right] = \sum_{i=1}^{k} \mathbf{u}_i^\top \mathbf{X}\mathbf{X}^\top \mathbf{u}_i.$$

This form is the same as maximize the variance and thus we can get the same conclusion.

2 Probabilistic PCA

Probabilistic PCA (PPCA) provides the probabilistic view of PCA. It is more fast the PCA, and can be used when data contains missing value (EM algorithm).

Assumption of PPCA:

$$\begin{aligned} p(\mathbf{z}) &= \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I}) \\ \mathbf{x} &= \mathbf{Wz} + \boldsymbol{\mu} + \boldsymbol{\epsilon} \\ p(\boldsymbol{\epsilon}) &= \mathcal{N}\left(\mathbf{0}, \sigma^2 \mathbf{I}\right) \end{aligned}$$

2.1 MLE for PPCA

Derive the distribution of $\mathbf{x}$ by the above assumption, we get

$$\begin{aligned} p(\mathbf{x}) &= \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \mathbf{C}) \\ \mathbf{C} &= \mathbf{W}\mathbf{W}^{\mathrm{T}} + \sigma^2 \mathbf{I} \end{aligned}$$

Then, we want to maximize the likelihood function of $\mathbf{x}$ w.r.t $\mu, \mathbf{W}, \sigma$.

The log-likelihood of $\mathbf{X}$ is

$$\begin{aligned} &\ln p\left(\mathbf{X}|\boldsymbol{\mu}, \mathbf{W}, \sigma^2\right) = \textstyle\sum_{n=1}^{N} \ln p\left(\mathbf{x}_n|\mathbf{W}, \boldsymbol{\mu}, \sigma^2\right) \\ &= -\tfrac{ND}{2} \ln(2\pi) - \tfrac{N}{2} \ln|\mathbf{C}| - \tfrac{1}{2} \textstyle\sum_{n=1}^{N} \left(\mathbf{x}_n - \boldsymbol{\mu}\right)^{\mathrm{T}} \mathbf{C}^{-1} \left(\mathbf{x}_n - \boldsymbol{\mu}\right) \end{aligned}$$

μ has the closed-form solution as sample mean. Substitute $\mu = \frac{1}{n}\sum_i \mathbf{x}_i$ back to the above likelihood function, we have:

$$\ln p\left(\mathbf{X}|\mathbf{W},\boldsymbol{\mu},\sigma^2\right) = -\frac{N}{2}\left\{D\ln(2\pi) + \ln|\mathbf{C}| + \mathrm{Tr}\left(\mathbf{C}^{-1}\mathbf{S}\right)\right\}$$

Then, we take derivative w.r.t $\mathbf{W}$ and σ, we can get the closed-form solution as follows:

$$\mathbf{W}_{\mathrm{ML}} = \mathbf{U}_M\left(\mathbf{L}_M - \sigma^2\mathbf{I}\right)^{1/2}\mathbf{R}$$
$$\sigma^2_{\mathrm{ML}} = \frac{1}{D-M}\sum_{i=M+1}^{D}\lambda_i$$

where $\mathbf{U}_M$ is a $D \times M$ matrix whose columns are given by any subset (of size M) of the eigenvectors of the data covariance matrix $\mathbf{S}$, the $M \times M$ diagonal matrix $\mathbf{L}_M$ has elements given by the corresponding eigenvalues λ_i, and $\mathbf{R}$ is an arbitrary M x M orthogonal matrix.
Furthermore, Tipping and Bishop (1999b) showed that the maximum of the likelihood function is obtained when the M eigenvectors are chosen to be those whose eigenvalues are the M largest (all other solutions being saddle points).

2.2 EM for PPCA

Note $\mathbf{Z}$ is latent variable, we can directly borrow EM to optimize it. The complete data log likelihood function is:

$$\ln p\left(\mathbf{X},\mathbf{Z}|\boldsymbol{\mu},\mathbf{W},\sigma^2\right) = \sum_{n=1}^{N}\left\{\ln p\left(\mathbf{x}_n|\mathbf{z}_n\right) + \ln p\left(\mathbf{z}_n\right)\right\}$$

Initialize the $\mathbf{W}$, σ E step: (take expectation over the latent variable):

$$\begin{aligned}&\mathbb{E}\left[\ln p\left(\mathbf{X},\mathbf{Z}|\boldsymbol{\mu},\mathbf{W},\sigma^2\right)\right] = -\textstyle\sum_{n=1}^{N}\left\{\frac{D}{2}\ln\left(2\pi\sigma^2\right) + \frac{1}{2}\mathrm{Tr}\left(\mathbb{E}\left[\mathbf{z}_n\mathbf{z}_n^{\mathrm{T}}\right]\right)\right.\\ &+\tfrac{1}{2\sigma^2}\left\|\mathbf{x}_n - \boldsymbol{\mu}\right\|^2 - \tfrac{1}{\sigma^2}\mathbb{E}\left[\mathbf{z}_n\right]^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\left(\mathbf{x}_n - \boldsymbol{\mu}\right)\\ &\left.+\tfrac{1}{2\sigma^2}\mathrm{Tr}\left(\mathbb{E}\left[\mathbf{z}_n\mathbf{z}_n^{\mathrm{T}}\right]\mathbf{W}^{\mathrm{T}}\mathbf{W}\right)\right\}\end{aligned}$$

The expectation of $\mathbf{z}$ given $\mathbf{W}$, σ are

$$\begin{aligned}\mathbb{E}\left[\mathbf{z}_n\right] &= \mathbf{M}^{-1}\mathbf{W}^{\mathrm{T}}\left(\mathbf{x}_n - \overline{\mathbf{x}}\right)\\ \mathbb{E}\left[\mathbf{z}_n\mathbf{z}_n^{\mathrm{T}}\right] &= \sigma^2\mathbf{M}^{-1} + \mathbb{E}\left[\mathbf{z}_n\right]\mathbb{E}\left[\mathbf{z}_n\right]^{\mathrm{T}}\end{aligned} \tag{1}$$

M step: maximize the likelihood respect to $\mathbf{W}$ and σ

$$\begin{aligned}\mathbf{W}_{\text{ new}} &= \left[\sum_{n=1}^{N}\left(\mathbf{x}_n - \overline{\mathbf{x}}\right)\mathbb{E}\left[\mathbf{z}_n\right]^{\mathrm{T}}\right]\left[\sum_{n=1}^{N}\mathbb{E}\left[\mathbf{z}_n\mathbf{z}_n^{\mathrm{T}}\right]\right]^{-1}\\ \sigma^2_{\mathrm{new}} &= \frac{1}{ND}\sum_{n=1}^{N}\left\{\left\|\mathbf{x}_n - \overline{\mathbf{x}}\right\|^2 - 2\mathbb{E}\left[\mathbf{z}_n\right]^{\mathrm{T}}\mathbf{W}_{\mathrm{new}}^{\mathrm{T}}\left(\mathbf{x}_n - \overline{\mathbf{x}}\right)\right.\\ &\quad\left.+\mathrm{Tr}\left(\mathbb{E}\left[\mathbf{z}_n\mathbf{z}_n^{\mathrm{T}}\right]\mathbf{W}_{\mathrm{new}}^{\mathrm{T}}\mathbf{W}_{\mathrm{new}}\right)\right\}\end{aligned} \tag{2}$$

3 Kernel PCA

PCA is for linear dimension reduction. Now, we use kernel to mapping data $\mathbf{X}$ to $\phi(\mathbf{X})$

By the same derivative of maximum variance, we have:

$$\mathbf{C} = \frac{1}{N}\sum_{n=1}^{N}\phi\left(\mathbf{x}_n\right)\phi\left(\mathbf{x}_n\right)^{\mathrm{T}}$$
$$\mathbf{C}\mathbf{v}_i = \lambda_i\mathbf{v}_i$$
$$\frac{1}{N}\sum_{n=1}^{N}\phi\left(\mathbf{x}_n\right)\left\{\phi\left(\mathbf{x}_n\right)^{\mathrm{T}}\mathbf{v}_i\right\} = \lambda_i\mathbf{v}_i$$

The above can be rewritten as:

$$\mathbf{v}_i = \sum_{n=1}^{N} a_{in}\phi\left(\mathbf{x}_n\right)$$

We know the projection vector is a linear combination of $\phi(\mathbf{x}_n)$. Left multiply $\phi(\mathbf{X})^T$ on both hand side of equation, we have

$$\frac{1}{N}\sum_{n=1}^{N} k\left(\mathbf{x}_l, \mathbf{x}_n\right)\sum_{m=1}^{m} a_{im}k\left(\mathbf{x}_n, \mathbf{x}_m\right) = \lambda_i \sum_{n=1}^{N} a_{in}k\left(\mathbf{x}_l, \mathbf{x}_n\right)$$

In matrix form, the above equation is:

$$\mathbf{K}^2\mathbf{a}_i = \lambda_i N\mathbf{K}\mathbf{a}_i \tag{3}$$

Since K is PSD kernel, then we can get the optimum $\mathbf{a}$ by solving the eigenvalue problem:

$$\mathbf{K}\mathbf{a}_i = \lambda_i N\mathbf{a}_i \tag{4}$$

After we solve the eigenvalue problem, we substitute the linear span expression of projection vector, we have

$$y_i(\mathbf{x}) = \boldsymbol{\phi}(\mathbf{x})^{\mathrm{T}}\mathbf{v}_i = \sum_{n=1}^{N} a_{in}\boldsymbol{\phi}(\mathbf{x})^{\mathrm{T}}\boldsymbol{\phi}\left(\mathbf{x}_n\right) = \sum_{n=1}^{N} a_{in}k\left(\mathbf{x}, \mathbf{x}_n\right) \tag{5}$$

However, the above derivation assumes the high dimensional projected data has zero mean. We need to calibrate this. Suppose the mean centered high dimensional data is

$$\widetilde{\phi}\left(\mathbf{x}_n\right) = \phi\left(\mathbf{x}_n\right) - \frac{1}{N}\sum_{l=1}^{N}\phi\left(\mathbf{x}_l\right) \tag{6}$$

Then, we can compute the kernel of centered kernel feature:

$$\begin{aligned}
\widetilde{K}_{nm} =& \widetilde{\phi}\left(\mathbf{x}_n\right)^{\mathrm{T}}\widetilde{\phi}\left(\mathbf{x}_m\right) \\
=& \phi\left(\mathbf{x}_n\right)^{\mathrm{T}}\boldsymbol{\phi}\left(\mathbf{x}_m\right) - \frac{1}{N}\sum_{l=1}^{N}\boldsymbol{\phi}\left(\mathbf{x}_n\right)^{\mathrm{T}}\boldsymbol{\phi}\left(\mathbf{x}_l\right) \\
& - \frac{1}{N}\sum_{l=1}^{N}\boldsymbol{\phi}\left(\mathbf{x}_l\right)^{\mathrm{T}}\boldsymbol{\phi}\left(\mathbf{x}_m\right) + \frac{1}{N^2}\sum_{j=1}^{N}\sum_{l=1}^{N}\boldsymbol{\phi}\left(\mathbf{x}_j\right)^{\mathrm{T}}\boldsymbol{\phi}\left(\mathbf{x}_l\right) \\
=& k\left(\mathbf{x}_n, \mathbf{x}_m\right) - \frac{1}{N}\sum_{l=1}^{N}k\left(\mathbf{x}_l, \mathbf{x}_m\right) - \frac{1}{N}\sum_{l=1}^{N}k\left(\mathbf{x}_n, \mathbf{x}_l\right) + \frac{1}{N^2}\sum_{j=1}^{N}\sum_{l=1}^{N}k\left(\mathbf{x}_j, \mathbf{x}_l\right)
\end{aligned} \tag{7}$$

Matrix form of mean centered kernel is:

$$\tilde{\mathbf{K}} = \mathbf{K} - \mathbf{1}_N\mathbf{K} - \mathbf{K}\mathbf{1}_N + \mathbf{1}_N\mathbf{K}\mathbf{1}_N \tag{8}$$

4 Independent Component Analysis

- ICA is a linear method.
- ICA tries to recover the independent components rather than orthogonal components (minimize the mutual information).
- ICA will fail when more than one subspace is Gaussian

4.1 Minimize the Mutual Information

$$\begin{aligned}
& l\left(\{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_d\}\right) \\
& = \textstyle\sum_{i=1}^{d} H\left(\mathbf{y}_i\right) - H\left(\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_d\right) \\
& H\left(\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_d\right) = H(\mathbf{X}\mathbf{W}) = H\left(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_d\right) + \log\det(\mathbf{W})
\end{aligned} \tag{9}$$

The MI can be rewritten as:

$$\begin{array}{l} l\left(\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_d\right) \\ =\sum_{i=1}^n H\left(y_i\right)-H\left(\mathbf{x}_1, x_2, \ldots, \mathbf{x}_d\right)-\log \det (W) \end{array} \tag{10}$$

Assume y to be unit variance

$$\begin{array}{l} E\left(\mathbf{y}^T \mathbf{y}\right)=\mathbf{W} E\left\{\mathbf{x} \mathbf{x}^{\top}\right\} \mathbf{W}^T==\mathbf{I} \\ \det (\mathbf{I})=1=\det (\mathbf{W})\left(\det E\left\{\mathbf{x} \mathbf{x}^{\mathbf{T}}\right\}\right) \det \left(\mathbf{W}^T\right) \end{array} \tag{11}$$

This indicates $det(W)$ is a constant. Now the MI can be written as

$$\begin{array}{l} l\left(\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_d\right) \\ =\sum_{i=1}^n H\left(y_i\right)-H\left(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_d\right) \end{array} \tag{12}$$

Maximize the non-gaussianity can achieve ICA.

4.2 Non-Gaussianity Estimation

We will introduce a few metric for evaluating the non-gaussianity.

4.2.1 Central Limit Theorem

- Distribution of a sum of independent random variables tends toward a Gaussian distribution
- Sum of several independent random variables usually has a distribution that is closer to gaussian than any of the original random variables

4.2.2 Kurtosis

Kurtosis is defined by:

$$\text{kurt}\left(\mathrm{y}\right)=\mathrm{E}\left\{\mathrm{y}^4\right\}-3\left(\mathrm{E}\left\{\mathrm{y}^2\right\}\right)^2 \tag{13}$$

When we constrain the y has zero mean and unit variance, the kurtosis has the form$E\left\{y^4\right\}-3$. Kurtosis measuares the degree of peakdness of a distribution and it is zero only for Gaussian distribution. However, it is sensitive to outliers due to the fourth moment.

4.2.3 Negentropy

The entropy of a random variable is defined as

$$H(y)=-\int_{-\infty}^{\infty} p(y) \log p(y) dy=-E\left\{\log p_i(y)\right\} \tag{14}$$

Denote y_G as the entropy of gaussian variable, the negentropy is defined as

$$J(y)=H\left(y_G\right)-H(y) \geq 0$$

It is hard to get $J(y)$ since we do not know the density distribution $p(y)$.

4.2.4 Approximation to Negentropy

Some approximation:

$$\begin{array}{ll} J(y) \approx \frac{1}{12} E\left\{y^3\right\}^2+\frac{1}{48} kurt(y)^2 & \text{not robust} \\ J(y) \approx \sum_{i=1}^p k_i\left[E\left\{G_i(y)\right\}-E\left\{G_i(g)\right\}\right]^2 & \text{better approximation,} \end{array}$$

where k_i are some positive constants, y is assumed to have zero mean and unit variance, and g is a Gaussian variable with zero mean and unit variance. G_i are some non-quadratic functions such as

$$G_1(y)=\frac{1}{a} \log \cosh (a y), \quad G_2(y)=-\exp\left(-y^2 / 2\right), \tag{15}$$

where $1 \leq a \leq 2$ is some suitable constant. Although this approximation may no be accurate, it is always greater than zero except when x is Gaussian.

4.2.5 ICA as Projection Pursuit (FastICA)

We want to minimize:

$$\sum_i E\left\{G\left(y_i\right)\right\} = \sum_i E\left\{G\left(\mathbf{w}_i^T\mathbf{x}\right)\right\}, \tag{16}$$

where w is a rotation matrix. Using Lagrangian Multiplier, we have:

$$O(\mathbf{w}) = E\left\{G\left(\mathbf{w}^T\mathbf{x}\right)\right\} - \beta\left(\mathbf{w}^T\mathbf{w} - 1\right)/2 \tag{17}$$

Take derivative respect to w:

$$F(\mathbf{w}) = \frac{\partial O(\mathbf{w})}{\partial \mathbf{w}} = E\left\{\mathbf{x}g\left(\mathbf{w}^T\mathbf{x}\right)\right\} - \beta\mathbf{w} = 0 \tag{18}$$

We can use newton method to optimize it. The Hessian is

$$J_F(\mathbf{w}) = \frac{\partial F}{\partial \mathbf{w}} = E\left\{\mathbf{x}\mathbf{x}^T g'\left(\mathbf{w}^T\mathbf{x}\right)\right\} - \beta\mathbf{I} \tag{19}$$

The first term can be approximated as

$$E\left\{\mathbf{x}\mathbf{x}^T g'\left(\mathbf{w}^T\mathbf{x}\right)\right\} \approx E\left\{\mathbf{x}\mathbf{x}^T\right\} E\left\{g'\left(\mathbf{w}^T\mathbf{x}\right)\right\} = E\left\{g'\left(\mathbf{w}^T\mathbf{x}\right)\right\}\mathbf{I} \tag{20}$$

Then, the Hessian becomes diagonal, and the inverse is easy to calculate.

$$J_F(\mathbf{w}) = \left[E\left\{g'\left(\mathbf{w}^T\mathbf{x}\right)\right\} - \beta\right]\mathbf{I} \tag{21}$$

The Newton iteration becomes:

$$\mathbf{w} \Leftarrow \mathbf{w} - \frac{1}{E\left\{g'\left(\mathbf{w}^T\mathbf{x}\right)\right\} - \beta}\left[E\left\{\mathbf{x}g\left(\mathbf{w}^T\mathbf{x}\right)\right\} - \beta\mathbf{w}\right] \tag{22}$$

Multiplying both sides by the scaler $\beta - E\left\{g'\left(\mathbf{w}^T\mathbf{x}\right)\right\}$, we have

$$\mathbf{w} \Leftarrow E\left\{\mathbf{x}g\left(\mathbf{w}^T\mathbf{x}\right)\right\} - E\left\{g'\left(\mathbf{w}^T\mathbf{x}\right)\right\}\mathbf{w} \tag{23}$$

We are just scale the parameters, we can renormalizes it. The projection pursuit for ICA can be summarized as follows:

- Choose initial random guess for w
- Iterate:
$$\mathbf{w} \Leftarrow E\left\{\mathbf{x}g\left(\mathbf{w}^T\mathbf{x}\right)\right\} - E\left\{g'\left(\mathbf{w}^T\mathbf{x}\right)\right\}\mathbf{w}$$
- Normalize
$$\mathbf{w} \Leftarrow \mathbf{w}/\|\mathbf{w}\|$$
- If not converged, go back to step 2
Estimate the 2^{nd} ICA component similarity, just add the orthogonal constraint.

4.2.6 MLE for ICA

Another view for deriving ICA is from MLE. The following equation always holds.

$$p_x(x) = p_s(Wx)\cdot|W| \tag{24}$$

Since we assume the source are independent:

$$p(s) = \prod_{i=1}^{n} p_s\left(s_i\right) \tag{25}$$

So the likelihood for our observation is

$$p(x) = \prod_{i=1}^{n} p_s\left(w_i^T x\right)\cdot|W| \tag{26}$$

We need to assume the probability distribution of the sources, and remember we cannot choose Gaussian. A reasonable CDF is the sigmoid function $g(s) = 1/\left(1+e^{-s}\right)$, and the pdf is $p_s(s) = g'(s)$ The log likelihood is:

$$\ell(W) = \sum_{i=1}^{m}\left(\sum_{j=1}^{n}\log g'\left(w_j^T x^{(i)}\right) + \log|W|\right) \tag{27}$$

, Applying gradient descent or SGD, we can get the update rule. ($\nabla_W|W| = |W|\left(W^{-1}\right)^T$)

5 Non-linear ICA

- It is an open problem
- It is largely related to Disentangling Representation
- Beta VAE, Factor VAE

5.1 VAE

$$
\begin{aligned}
&KL(q(z)\|p(z|x)) = \int q(z)\log\tfrac{q(z)}{p(z|x)}dz \\
&= \int q(z)[\log q(z) - \log p(z|X)]dz \\
&= \int q(z)[\log q(z) - \log p(X|z) - \log p(z)]dz + \log p(X) \\
&= \int q(z)\log p(X|z)dz - KL(q(z)\|p(z)) + \log p(X) \\
&<= \int q(z)\log p(X|z)dz - KL(q(z)\|p(z))
\end{aligned}
\tag{28}
$$

5.2 Beta-VAE

$$
\begin{aligned}
&\max_\theta \int p(x|z)dz \\
&\text{s.t.KL}\left(q_\phi(z|x)\|p(z)\right) < \epsilon
\end{aligned}
\tag{29}
$$

Objective of Beta-VAE

$$
\mathcal{F}(\theta,\phi,\beta;\mathbf{x},\mathbf{z}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - \beta\left(D_{KL}\left(q_\phi(\mathbf{z}|\mathbf{x})\|p(\mathbf{z})\right) - \epsilon\right) \tag{30}
$$

5.3 Factor-VAE

Objective:

$$
\mathbb{E}_{p(\mathbf{x})}\left[\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - D_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x})\|p(\mathbf{z})\right)\right] - \gamma D_{\mathrm{KL}}\left(q(\mathbf{z})\|\prod_{j=1}^{d} q\left(\mathbf{z}_j\right)\right) \tag{31}
$$