Introduction to Machine Learning.
CSCI-UA 9473, Lecture 8.

Augustin Cosse

Ecole Normale Supérieure, DMA & NYU
Fondation Sciences Mathématiques de Paris.

2018

Principal components (General Intro I)

- The principal components of a set of data provides a sequence of best linear approximations to the data.

- Consider a sequence of prototypes $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$. Instead of learning a simple linear regression for the data, we might want to learn a rank-q linear model

$$f(\lambda) = \mu + \boldsymbol{W}\lambda$$

- $\boldsymbol{W}$ is a $p \times q$ matrix with orthogonal unit vectors as columns

- We then might want to fit this model to the data in order to learn the first q directions which best describe this data

Principal components (General Intro II)

- Fitting the rank-q model to the data is done by minimizing the reconstruction error

$$\min_{\mu,\{\boldsymbol{\lambda}_i\},\boldsymbol{W}} \sum_{i=1}^{N} \|\boldsymbol{x}_i - \boldsymbol{\mu} - \boldsymbol{W}\boldsymbol{\lambda}_i\|^2$$

- Optimizing with respect to $\boldsymbol{\mu}$ and $\boldsymbol{\lambda}_i$ first, we get

$$\mu^* = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$\boldsymbol{\lambda}_i^* = \boldsymbol{W}^T \left(\boldsymbol{x}_i - \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{x}_i\right)$$

Principal components (General Intro III)

- Substituting the expressions for the mean μ and the coefficients λ_i in the expression of the reconstruction error, we get

$$\min_{\boldsymbol{W}} \sum_{i=1}^{N} \left\| (\boldsymbol{x}_i - \bar{\boldsymbol{x}}) - \boldsymbol{W}\boldsymbol{W}^T(\boldsymbol{x}_i - \bar{\boldsymbol{x}}) \right\|^2$$

  where we let $\bar{\boldsymbol{x}} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i$

- $\mathcal{P} = \boldsymbol{W}\boldsymbol{W}^T$ is a projection matrix that maps each point $\boldsymbol{x}_i - \bar{\boldsymbol{x}}$ onto the subspace spanned by the q columns of $\boldsymbol{W}$

Principal components (General Intro IV)

$$\min_{\boldsymbol{W}} \sum_{i=1}^{N} \left\| (\boldsymbol{x}_i - \bar{\boldsymbol{x}}) - \boldsymbol{W}\boldsymbol{W}^T(\boldsymbol{x}_i - \bar{\boldsymbol{x}}) \right\|^2$$

- Let us use $\boldsymbol{X}$ to denote the $N \times p$ matrix whose rows corresponds to the N prototypes $\boldsymbol{x}_i$.

- The solution that minimizes the reconstruction error can be computed directly through the Singular Value Decomposition (SVD) of $\boldsymbol{X}$.

- If we use $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T$ to denote the SVD of $\boldsymbol{X}$, the solution for $\boldsymbol{W}$ consists of the first q columns of $\boldsymbol{V}$ (principal vectors or principal directions). Given those columns, the principal components scores are encoded in the first q columns $\boldsymbol{U}_q\boldsymbol{D}_q$.

Conventions

- There seems to be two conventions regarding the designation of the vectors, (i.e. the columns of $\boldsymbol{W}$ that span the linear subspace) and the scores, $\boldsymbol{W}^T\boldsymbol{X}$ (i.e the projection of the original points onto those vectors defining the coefficients of the linear combination, $\boldsymbol{x}_i = \sum_{i=1}^{q} \lambda_i \boldsymbol{w}_i$)
- The first convention calls principal components the vectors encoded in the columns of $\boldsymbol{W}$ and principal component scores the weights $\boldsymbol{W}^T\boldsymbol{X}$
- The second convention calls principal axes, principal direction or even principal component vectors the columns of $\boldsymbol{V}$ and designate as principal components the weights $\boldsymbol{W}^T\boldsymbol{X}$

PCA: Main Theorem

Classical Principal Component Analysis (Murphy 2012)

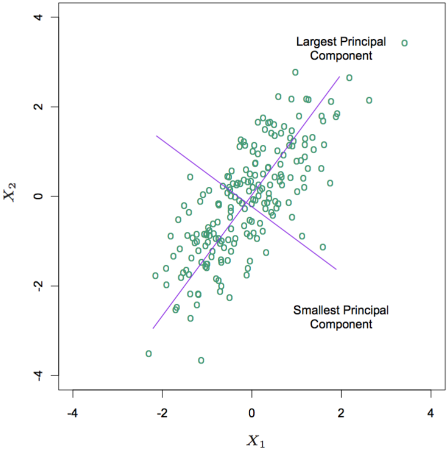
Suppose we want to find an orthogonal set of L linear basis vectors $\boldsymbol{w}_j \in \mathbb{R}^D$ and the scores $\boldsymbol{z}_i \in \mathbb{R}^L$ such that we minimize

$$J(\boldsymbol{W}, \boldsymbol{Z}) = \frac{1}{N} \sum_{i=1}^{N} \|\boldsymbol{x}_i - \hat{\boldsymbol{x}}_i\|^2$$

where $\hat{\boldsymbol{x}}_i = \boldsymbol{W}\boldsymbol{z}_i$, subject to the constraint that $\boldsymbol{W}$ is orthonormal. Equivalently, we can write this objective as $J(\boldsymbol{W}, \boldsymbol{Z}) = \|\boldsymbol{X} - \boldsymbol{W}\boldsymbol{Z}^T\|_F^2$ where $\boldsymbol{Z}$ is an $N \times L$ matrix with the $\boldsymbol{z}_i$ as rows and $\|\boldsymbol{A}\|_F$ is the Frobenius norm of the matrix $\boldsymbol{A}$ which is defined as $\|\boldsymbol{A}\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij}^2}$

The solution is then given by setting $\hat{\boldsymbol{W}}$ to $\boldsymbol{V}_L$ where $\boldsymbol{V}_L$ contains the L largest eigenvectors of the empirical covariance matrix $\hat{\boldsymbol{\Sigma}} = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{x}_i \boldsymbol{x}_i^T$

Principal components: illustration I



H,T,F, Elements of Statistical Learning

Principal components: illustration II

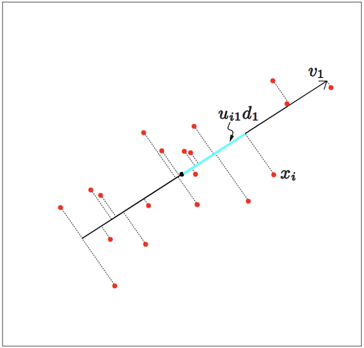


FIGURE 14.20. *The first linear principal component of a set of data. The line minimizes the total squared distance from each point to its orthogonal projection onto the line.*

H,T,F, Elements of Statistical Learning

Principal components: illustration III

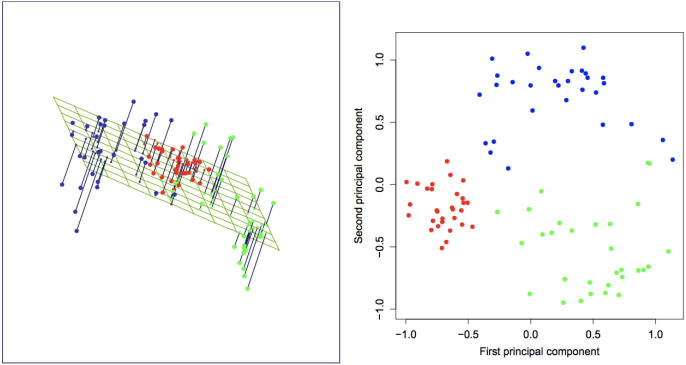


FIGURE 14.21. *The best rank-two linear approximation to the half-sphere data. The right panel shows the projected points with coordinates given by* $\mathbf{U}_2\mathbf{D}_2$*, the first two principal components of the data.*

H,T,F, Elements of Statistical Learning

Sparse Principal Component Analysis (I)

- In PCA, we are often interested in deriving an interpretation of the principal directions v_j. In particular, we want to understand which component plays a more important role. This is typically made easier when the vectors are sparse.

- In genomics, one is interested in datasets where each variable correspond to a specific gene. Enforcing spare principal components will enable the researchers to focus exlusively on a subset of the genes which might be more closely related to each other

- In financial data analysis and portfolio hedging, one is interested in hedging the risk by writing the value of the portfolio as a combination of few factors (e.g. level, spread and convexity). Simple approach would assign non zero weights to all assets which implies high fixed transaction costs

Sparse Principal Component Analysis (II)

- The approaches in sparse PCA focus either on the maximum variance property of the principal components or the minimum reconstruction error.

- Maximal Variance (e.g. SCoTLASS). For a $\boldsymbol{X} \in \mathbb{R}^{N \times p}$

$$\max \boldsymbol{v}^T \boldsymbol{X}^T \boldsymbol{X} \boldsymbol{v}, \quad \text{subject to} \sum_{j=1}^{p} |v_j| \leq t, \; \boldsymbol{v}^T \boldsymbol{v} = 1$$

- Minimum reconstruction error. Here, if we let x_i to denote the i^{th} component of $\boldsymbol{X}$, we solve

$$\min_{\theta, \boldsymbol{v}} \sum_{i=1}^{N} \|\boldsymbol{x}_i - \boldsymbol{\theta} \boldsymbol{v}^T \boldsymbol{x}_i\|_2^2 + \lambda \|\boldsymbol{v}\|_2^2 + \lambda_1 \|\boldsymbol{v}\|_1 \quad s.t. \quad \|\boldsymbol{\theta}\|_2 = 1$$

Sparse Principal Component Analysis (III)

- When considering multiple components, the minimum reconstruction error approach naturally extends as

$$\min_{\boldsymbol{\Theta},\boldsymbol{V}} \sum_{i=1}^{N} \|\boldsymbol{x}_i - \boldsymbol{\Theta}\boldsymbol{V}^T\boldsymbol{x}_i\|_2^2 + \lambda \sum_{k=1}^{K} \|\boldsymbol{v}_k\|_2^2 + \lambda_1 \sum_{k=1}^{K} \|\boldsymbol{v}_k\|_1$$

$$\text{subject to } \boldsymbol{\Theta}^T\boldsymbol{\Theta} = \boldsymbol{I}_K$$

- The criterion is not jointly convex with respect to $\boldsymbol{V}$ and $\boldsymbol{\Theta}$ but it is convex in each parameter with the other fixed, and it can thus be minimized iteratively.

Non negative matrix factorization

- Non Negative matrix factorization provides an alternative to PCA in which the data matrix as well as its factorization are assumed to be non negative. For a data matrix $\boldsymbol{X}$ we look for a factorization $\boldsymbol{X} \approx \boldsymbol{W}\boldsymbol{H}$ with $x_{ij}, h_{ij}, w_{ij} \geq 0$

- As the quantitities are non negative, one approach is to minimize an extension of the Kullback Leibler divergence to matrices (Lee and Seung 2001)

$$D(\boldsymbol{W}\boldsymbol{H}||\boldsymbol{X}) = \sum_{i=1}^{N}\sum_{j=1}^{p}\left(\boldsymbol{X}_{ij}\log(\frac{\boldsymbol{X}_{ij}}{(\boldsymbol{W}\boldsymbol{H})_{ij}}) - \boldsymbol{X}_{ij} + (\boldsymbol{W}\boldsymbol{H})_{ij}\right)$$

Non negative matrix factorization

- One can show (see Theorem 2 in Lee and Seung 2001) that the divergence D is non increasing under the updates

$$\boldsymbol{W}_{ia} \leftarrow \boldsymbol{W}_{ia} \frac{\sum_\mu \boldsymbol{H}_{a\mu}\boldsymbol{X}_{i\mu}/(\boldsymbol{W}\boldsymbol{H})_{i\mu}}{\sum_\nu \boldsymbol{H}_{a\nu}}$$

$$\boldsymbol{H}_{a\mu} \leftarrow \boldsymbol{H}_{a\mu} \frac{\sum_i \boldsymbol{W}_{ia}\boldsymbol{X}_{i\mu}/(\boldsymbol{W}\boldsymbol{H})_{i\mu}}{\sum_k \boldsymbol{W}_{ka}}$$

- Just as PCA and sparse PCA, Non Negative Matrix Factorization finds applications in bioinformatics and genomics where it is used to identify patterms of mutations

- It is also used in text mining (see for example Arora et al., 2013) where it is applied to document/term matrices which encode the number of occurence of specific terms in a sequence of documents.

Factor Analysis (General Introduction I)

- For a data matrix $\boldsymbol{X}$, the singular value decomposition, $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T$ can be interpreted as a latent variable representation.

- Writing $\boldsymbol{S} = \sqrt{N}\boldsymbol{U}$ and $\boldsymbol{A}^T = \boldsymbol{D}\boldsymbol{V}^T/\sqrt{N}$, we have $\boldsymbol{X} = \boldsymbol{S}\boldsymbol{A}^T$

- Each of the columns of $\boldsymbol{X}$ (encoding the prototypes) can be viewed as a linear combination of the columns of $\boldsymbol{S}$, i.e.

$$\begin{aligned}
X_1 &= a_{11}S_1 + a_{12}S_2 + \ldots + a_{1p}S_p \\
X_2 &= a_{21}S_1 + a_{22}S_2 + \ldots + a_{2p}S_p \\
\vdots &\qquad \vdots \\
X_p &= a_{p1}S_1 + a_{p2}S_2 + \ldots + a_{pp}S_p
\end{aligned}$$

Factor Analysis (General Introduction II)

- The particular choice $\boldsymbol{S} = \sqrt{N}\boldsymbol{U}$ is important because it implies $\mathrm{Cov}(\boldsymbol{S}) = N\boldsymbol{U}\boldsymbol{U}^* = N\boldsymbol{I}$ (I.e the correlated vectors $\boldsymbol{X}_i$ are expressed as a linear combination of the uncorrelated vectors $\boldsymbol{S}_j$)

- The issue with such a latent representation is that it is not unique. For a given matrix $\boldsymbol{X}$, we can always write $\boldsymbol{X} = \boldsymbol{A}\boldsymbol{S} = \boldsymbol{A}\boldsymbol{R}\boldsymbol{R}^T\boldsymbol{S} = \boldsymbol{A}^*\boldsymbol{S}^*$ for any matrix orthogonal $p \times p$ $\boldsymbol{Q}$.

- Factor analysis alleviates this by reducing the number of degrees of freedom and considering the decomposition

$$
\begin{aligned}
X_1 &= a_{11}S_1 + a_{12}S_2 + \ldots + a_{1q}S_q + \varepsilon_1 \\
X_2 &= a_{21}S_1 + a_{22}S_2 + \ldots + a_{2q}S_q + \varepsilon_2 \\
\vdots \quad &\quad \vdots \\
X_p &= a_{p1}S_1 + a_{p2}S_2 + \ldots + a_{pq}S_q + \varepsilon_p
\end{aligned}
$$

From Factor Analysis to Independent Component Analysis

- In Factor Analysis, the latent factors $\boldsymbol{S}$ are encoding the common source of variation among the prototypes (and account for the correlation) while the ε_i are particular to each X_i and encode the remaining variation

- Independent Component Analaysis (ICA) has the same form as Factor Analysis except that it relies on independence among the signals S_ℓ.

- Recall that two variables X_1 and X_2 are independent if an only if their joint pdf is factorizable as

$$p(Y_1, Y_2) = p_1(Y_1)p_2(Y_2)$$

where $p_1(Y_1)$ and $p_2(Y_2)$ denote the marginals $p_1(Y_1) = \int p(Y_1, Y_2)\, dY_2$ and equivalently for $p_2(Y_2)$.

From Factor Analysis to Independent Component Analysis

- Independence in particular implies that for any two functions $h_1(Y_1)$ and $h_2(Y_2)$, we have

$$\mathbb{E}\left\{h_1(Y_1)h_2(Y_2)\right\} = \mathbb{E}\left\{h_1(Y_1)\right\}\mathbb{E}\left\{h_2(Y_2)\right\}$$

- The key idea in ICA is to assume that the observations $x_1, \ldots, x_n$ can be represented as mixtures of n independent components $S_1, \ldots, S_n$.

- Note that the decomposition $\boldsymbol{X} = \boldsymbol{AS}$ implies an ambiguity regarding the magnitude of the independent components s_i as well as their ordering.

- Any scaling αs in s can be compensated by a scaling $(1/\alpha)\boldsymbol{A}$ in $\boldsymbol{A}$ and we can always replace the weight matrix $\boldsymbol{A}$ and the components matrix $\boldsymbol{S}$ by a permutation $\boldsymbol{P}$ of those matrices

$$\boldsymbol{X} = \boldsymbol{AP}^{-1}\boldsymbol{PS}$$

From Factor Analysis to Independent Component Analysis

- The most famous example of application of Independent Component Analysis is the coktail party problem.

- In this problem which is also known as blind source separation or blind signal separation consists in unmixing signals from their linear combination (as one would for example try to recover disctinct speeches from the recording of their linear combination taped by multiple microphones)

- Besides the applications to sound signals, ICA has also been applied successfully to EEG and ECG, financial data processing and any other problem in which latent sources are mixed linearly and carry meaningful information.

- ICA has been important in the study of brain dynamics where the assumption is that signals recorded at each electrode are mixtures of independent potentials arising from different cortical activities.

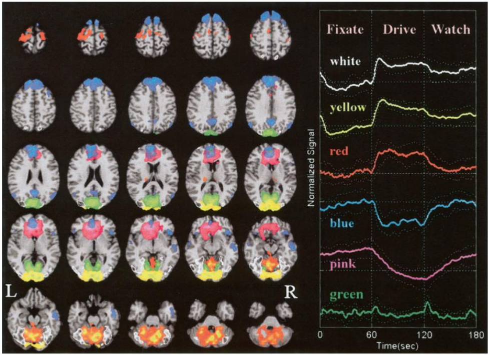
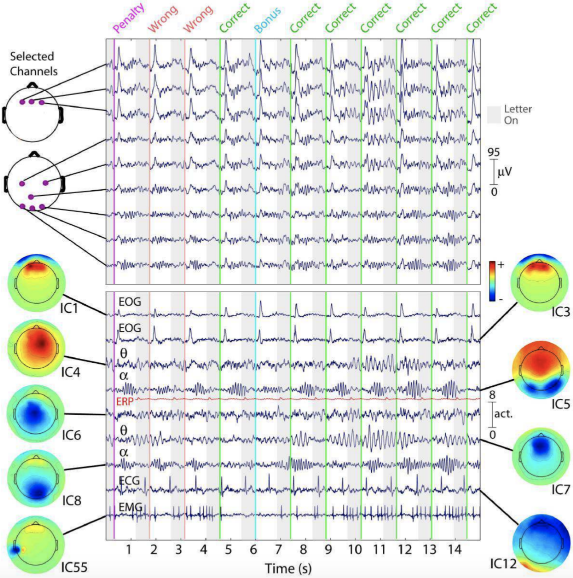


Figure 2.

Independent components and associated time courses from functional MRI scans. Random effects group fMRI maps are thresholded at $P < 0.00025$ ($t = 4.5$, $df = 14$). A total of six components are presented. A green component extends on both sides of the parieto-occipital sulcus including portions of cuneus, precuneus, motor areas is depicted in red. Orbitofrontal and anterior cingulate areas identified are depicted in pink. Finally, a component including medial frontal, parietal, and posterior cingulate regions is depicted in blue. Group averaged time courses (right) for the fixate-drive-watch order are also depicted with similar colors.

source: Calhoun et al, 2002



source: HTF, The Elements of Statistical Learning

From Factor Analysis to Independent Component Analysis

- Unlike in PCA, ICA requires the distributions of the sources to be non Gaussian (Gaussian independent components can only be defined up to a rotation and the use of Gaussian priors thus prevents identifiability of the factors)

- To illustrate the need for non Gaussian priors, think of PCA for which we have seen that the decomposition is invariant to any orthogonal transformation of the sources and mixing matrix. PCA can thus recover the best linear subspaces in which the signals live but it cannot recover those signals uniquely

- This phenomenon can be observed by taking two independent sources with uniform distributions and mix them by multiplying them with any mixing matrix $\boldsymbol{M}$

- We then obtain a set of observations on which we can apply both PCA and ICA. The result of PCA (which is known as whitening) recovers the data up to a rotation.

From Factor Analysis to Independent Component Analysis

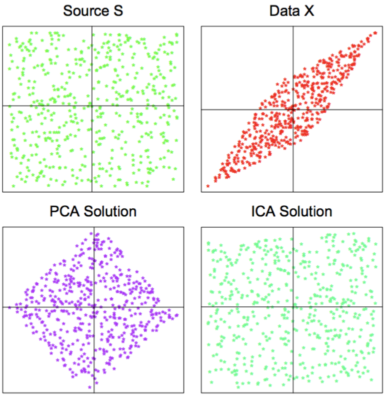


FIGURE 14.38. *Mixtures of independent uniform random variables. The upper left panel shows 500 realizations from the two independent uniform sources, the upper right panel their mixed versions. The lower two panels show the PCA and ICA solutions, respectively.*

source: H,T,F, *The Elements of Statistical Learning*

From Factor Analysis to Independent Component Analysis

- There exists several algorithm to solve the ICA problem. One of them (which we will cover later) compute the MLE by minimizing the negative log-likelihood.

- Another popular approach is to rely on the notion of entropy. For a random variable Y with density $g(Y)$, the differential entropy is defined as

$$H(Y) = -\int g(Y)\log(g(Y))\,dy$$

- Given the entropy, a natural measure of independence between the components of the random vector $\boldsymbol{Y}$ is the mutual information $I(\boldsymbol{Y})$,

$$I(\boldsymbol{Y}) = \sum_{j=1}^{p} H(Y_j) - H(\boldsymbol{Y})$$

From Factor Analysis to Independent Component Analysis

- $I(\boldsymbol{Y})$ can also be interpreted as the Kullback-Leibler divergence between the joint density $g(\boldsymbol{Y})$ and the independent version of this density $\prod_{j=1}^{p} g(Y_j)$ where $g_j(Y_j)$ here denotes the marginal density of the j^{th} component Y_j.

ICA in practice (I)

- Recall that ICA is interested in extracting an independent representation of the data.

$$f_S(s) = \prod_{j=1}^{p} f_j(s_j)$$

- One approach, on top of requiring $\boldsymbol{X} = \boldsymbol{AS}$, with $\boldsymbol{A}$ unitary, is to require those components to have independent tilted Gaussian distributions. (We take tilted Gaussians to avoid the uncertainty associated to Gaussians)

$$f_j(s_j) = \phi(s_j)e^{g_j(s_j)}$$

- The log-likelihood then reads

$$\ell(\boldsymbol{A}, \{g_j\}_{j=1}^{p}\,; \boldsymbol{X}) = \log(p(\boldsymbol{X})|\boldsymbol{A}, \{g_j\}_{j=1}^{p})$$

$$= \boxed{\sum_{i=1}^{N}\sum_{j=1}^{p}\left[\log(\phi_j(a_j^T x_i)) + g_j(a_j^T x_i)\right]}$$

ICA in practice (II)

- Without any additional constraints, the log-likelihood is over-parametrized

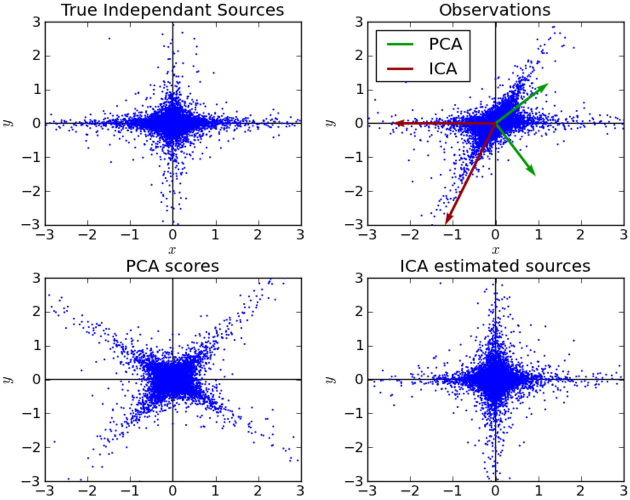
$$
\begin{aligned}
\ell(\boldsymbol{A}, \{g_j\}_{j=1}^{p}\,; \boldsymbol{X}) &= \log(p(\boldsymbol{X})|\boldsymbol{A}, \{g_j\}_{j=1}^{p}) \\
&= \sum_{i=1}^{N}\sum_{j=1}^{p}\left[\log(\phi_j(a_j^T x_i)) + g_j(a_j^T x_i)\right]
\end{aligned}
$$

- A popular approach is then to maximize a regularized version

$$
\sum_{j=1}^{p}\left[\frac{1}{N}\sum_{i=1}^{N}\Big(\log(\phi(a_j^T x_i) + g_j(a_j^T x_i))\Big) - \int \phi(t)e^{g_j(t)}\,dt\right]
- \sum_{j=1}^{p}\lambda_j \int \left\{g''(t)\right\}^2(t)\,dt
$$

- The first penalty enforces normalization $\int \phi(t)e^{\hat{g}_j(t)} = 1$. And the second enforces some regularization on the function g_j, $j = 1, \ldots, p$.

Principal components vs Independent components



Principal curves

- Just as we defined principal components, we can study representation of the data through principal curves. In this framework, we introduce the parametrized smooth curve $f(\lambda)$.

- $f(\lambda)$ is a smooth vector function with p components $f(\lambda) = (f_1(\lambda), f_2(\lambda), \dots, f_p(\lambda))$, each component $f_i(\lambda)$ being a smooth function of the parameter λ

- We say that $f(\lambda)$ is a principal curve for the data distribution of X if

$$f(\lambda) = \mathbb{E}\left\{X \mid \lambda_f(X) = \lambda\right\}$$

where $\lambda_f(X)$ is the mapping from X to the closest point on the curve. In other words f_λ should be the average of all the prototypes that project onto λ

Principal curves

Typical algorithms then iterate over the following two steps

- Average the points that relate to the same λ

$$\hat{f}_j(\lambda) \leftarrow \mathbb{E}\left\{\boldsymbol{X}_j | \lambda(X) = \lambda\right\}$$

- Update the parametrization λ so that the curve gets as close as possible to this average

$$\lambda = \operatorname*{argmin}_{\lambda'} \sum_j \|\hat{f}_j - f_j(\lambda')\|$$

- The method iterates between those two steps until convergence, starting, for example from the linear principal component

Principal Curves

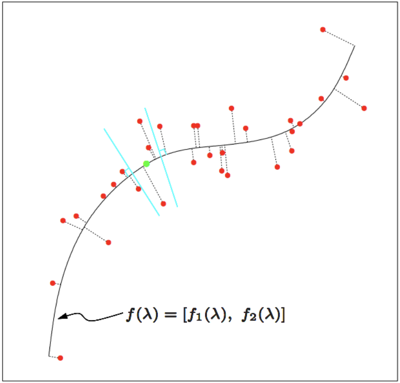


FIGURE 14.27. *The principal curve of a set of data. Each point on the curve is the average of all data points that project there.*

source: H,T,F, *The Elements of Statistical Learning*