STATS 306B: Unsupervised Learning **Spring 2014**

Lecture 12 — May 7

Lecturer: Lester Mackey *Scribe: Lan Huong, Snigdha Panigrahi*

12.1 Beyond Linear State Space Modeling

Last lecture we completed our discussion of linear Gaussian state space models. While these techniques are broadly applicable, they are not appropriate in all settings. Indeed, in many settings, non-linear / non-Gaussian transition and emission models are much more appropriate. For example, in **weather forecasting** we are interested in modeling specific weather outcomes at future timepoints, but our model of dynamics is decidedly nonlinear, being determined by complex geophysical simulations. In **pose estimation** we are interested in tracking the precise pose of different body parts over time, but our observations (images or frames of a video) are complex function of the 3D poses, nearby objects, lightning conditions, and camera calibration.

Unfortunately, in non-linear setting our filtering, prediction, and other inferential conditional distributions are typically quite complex and cannot be computed in closed form. To tackle this issue a various approximate nonlinear filters for approximate inference in these models have been developed, including **histogram filters**, **extended and unscented Kalman filters**, and **particle filters**. We will not have the time to discuss any of these approximate filters in detail, but a brief summary of their properties is found in the slides.

12.2 Independent Component Analysis

We now turn our attention to a new latent feature modeling approach, **Independent Component Analysis**, that, as usual, seeks to find the latent components and loadings that underlie our data but that employs a very different objective in selecting the loadings and components. In the terminology of the source community, ICA describes a class of related methods designed to separate a linearly mixed multivariate signal into additive non-Gaussian source signals (i.e., components), where the source signals are encouraged to be independent of one other.

12.2.1 Example ICA applications

Please see the accompanying slides.

The cocktail party problem In the cocktail party problem, p microphones are positioned in different locations in a room; each microphone records a different mixture of the audio signals emanating from the mouths of the party guests (i.e., the different conversations). Our goal is to recover the individual conversations from those mixtures (which often sound like incomprehensible hums). ICA allows you to retrieve p independent sources of sound (p voices) from recordings from p microphones placed around the room.

Natural images representation In this example, each square patch of an image is considered to be a datapoint (with pixels as the vector coordinates). The collection of these vectors for an image forms an image database. Our goal is to find a more natural basis for these images than the input pixel basis. ICA provides such a basis in the columns of the mixing matrix A. The slide shows the learned basis vectors; they capture visual structures like edges and contrast that are known stimuli of the primary visual cortex in humans.

12.2.2 Model formulation

In ICA the datapoints $x_1, x_2, \ldots, x_n \in \mathbb{R}^p$ (e.g., x_i may be a the audio signal recorded by each of p microphones at time i) are viewed as independent realizations of the simple probabilistic model

$$x = As$$

which satisfies the following assumptions:

1. s is a latent random vector in $\mathbb{R}^p$ which has zero mean, $\mathbb{E}[s] = 0$, independent components, and identity covariance, $\mathrm{Cov}(s) = I$.

2. The parameter A is an unknown **mixing matrix** $\in \mathbb{R}^{p\times p}$.

Hence, x is a random vector in $\mathbb{R}^p$ with $\mathrm{Cov}(x) = AA^\top$.

In our usual generative model notation, we could equivalently write

$$\begin{aligned} &s_i \overset{\text{iid}}{\sim} \mathbb{P} \quad \text{(for } \mathbb{P} \text{ an unknown distribution on } \mathbb{R}^p \text{ with independent components)} \\ &x_i = As_i \quad \text{(for unknown } A\text{)}. \end{aligned}$$

Note that s_i is analogous to the latent features z_i discussed in prior models, while A is analagous to the loading matrix U discussed in prior models.

See the slides for a discussion of the differences between modeling assumptions in ICA and Gaussian models like FA or PPCA.

12.2.3 Preprocessing

Before applying ICA, one typically mean-centers and **whitens** the data matrix. Whitening is a transformation that decorrelates a set of random variables. Suppose we have a set of data y with a known covariance matrix, $\mathrm{Cov}[y] = \Sigma$, and mean $\mathbb{E}[y] = 0$. Then the whitened form of y is $\tilde{y} = \Sigma^{-1/2}y$ with covariance $\mathrm{Cov}[\tilde{y}] = I$.

Since we do not have access to the true distribution of X, only to the sample $x_1, \ldots, x_n$ of this distribution, we replace all expectations over X in ICA with empirical expectations over the dataset. In this context, this means that we mean-center and pre-whiten each datapoint using the empirical mean, $\bar{x}_n = \frac{1}{n}\sum_{i=1}^n x_i$, and empirical covariance, $\mathbf{Q} = \frac{1}{n-1}\sum_{i=1}^n (x_i -$

$\bar{x}_n)(x_i - \bar{x}_n)^T$. Once we do this scaling, we can restrict our search of the mixing matrix A to the set of orthogonal matrices, since for pre-whitened x, we have

$$I = \mathrm{Cov}[x] = AA^T.$$

Hence our final **ICA goal** is to find $W = A^{-1}$ such that for $s = Wx$, the coordinates of s are minimally dependent.

12.2.4 Entropy and Mutual Information

Our notion of minimal dependence will rely on the concepts of **entropy** and **mutual information**. Below we suppose that $y \in \mathbb{R}^p$ is a random vector probability density function f.

The **entropy** $H(y)$ of y is given by

$$H(y) = -\int f(v) \log f(v) dv.$$

This can be viewed as a measure of the randomness in a distribution or as a measure of closeness of the distribution to uniformity. An important fact is that a $\mathcal{N}(\mu, \Sigma)$ distribution has the maximum entropy amongst all distributions with the mean μ and covariance Σ.

The **mutual information** $I(y)$ of y is a measure of departure from independence that can defined in terms of entropies

$$I(y) = \sum_{j=1}^{p} H(y_j) - H(y),$$

where $H(y_j)$ is a marginal entropy of component y_j with density f_j. $I(y)$ can be also expressed as the **Kullback-Leibler divergence** between $f(v)$ and the independent version of the density, $\prod_{j=1}^{p} f_j(v_j)$.[1]

12.2.5 An Objective for ICA

We are now prepared to define a target objective for ICA on mean-centered, prewhitened x. A useful fact is that whenever W is orthogonal, we have

$$I(s) = I(Wx) = \sum_{j=1}^{p} H(w_j^T x) - H(Wx) = \sum_{j=1}^{p} H(w_j^T x) - H(x) - \log|W| = \sum_{j=1}^{p} H(w_j^T x) - H(x)$$

where

$$W = \begin{bmatrix} \text{---} & w_1^T & \text{---} \\ \text{---} & w_2^T & \text{---} \\ & \vdots & \\ \text{---} & w_p^T & \text{---} \end{bmatrix}$$

[1] This is the distribution with independent coordinates closest to the distribution of Y in KL divergence.

is orthogonal.

Hence, our goal of minimizing dependence amongst coordinates can be phrased as

$$\boxed{\min_{\text{orthogonal } W} I(Wx)}.$$

This problem is equivalent to

$$\boxed{\text{minimizing dependence amongst components } w_j^T x = s_j}$$

which is again equivalent to

$$\boxed{\text{minimizing sums of of the entropies of } w_j^T x = s_j}$$

which finally reduces to

$$\boxed{\text{maximizing summed departures of } w_j^T x = s_j \text{ distributions from Gaussianity.}}$$

Unfortunately, it is difficult to optimize this objective directly, but many methods are available to either approximately optimize this objective or to optimize approximate objectives.

12.2.6 Negentropy and FastICA

FastICA is one popular approach that optimizes an approximation to our target mutual information objective. More precisely, the FastICA paper defines an equivalent target objective called **negentropy**. For a single random variable Y_j, negentropy is an explicit measure of departure from Gaussianity defined as

$$J(Y_j) = H(Y_j) - H(Z_j)$$

where $Z_j \sim N(0, \text{Var}(Y_j))$. Note that minimizing $I(Wx)$ over orthogonal W is equivalent to minimizing $\sum_{j=1}^{p} J(w_j^\top x)$ over orthogonal W. Moreover, since, $\text{Cov}(x) = I$ and W is orthogonal, so $\text{Var}(s_j) = \text{Var}(w_j^\top x) = 1$ for all j.

FastICA extracts an initial loadings vector w_1 by approximately minimizing an approximation to $J(w_1^T x)$

$$J(w_1^T x) \approx (\mathbb{E}[G(w_1^T x)] - \mathbb{E}[G(Z_1)])^2$$

where $G(y) = \frac{1}{a}\log\cosh(ay)$ for some $a \in [1, 2]$, and where the population expectation over x is replaced by an empirical expectation over our data:

$$\textbf{Empirical objective: } (\frac{1}{n}\sum_{i=1}^{n} G(w_1^T x_i) - \mathbb{E}[G(Z_1)])^2.$$

The details of the Newton-type algorithm proposed to optimize this objective are given in today's ICA reading. This technique yields the first loadings vector w_1. Additional loadings vectors w_j can be obtained in succession by minimizing the equivalent approximation to $J(w_j^T x)$ under the constraint that $w_j^\top w_l$ for all $l < j$.

12.3 Canonical Correlation Analysis (CCA)

We will now consider a slightly different setting for linear dimensionality reduction in which we are given two paired mean-centered datasets, X and Y, where

$$X = \begin{bmatrix} \text{——} & x_1 & \text{——} \\ \text{——} & x_2 & \text{——} \\ & \vdots & \\ \text{——} & x_n & \text{——} \end{bmatrix} \quad \text{and} \quad Y = \begin{bmatrix} \text{——} & y_1 & \text{——} \\ \text{——} & y_2 & \text{——} \\ & \vdots & \\ \text{——} & y_n & \text{——} \end{bmatrix},$$

for $x_i \in \mathbb{R}^p, \;\; y_i \in \mathbb{R}^q$. Each pair (x_i, y_i) represents two views of the same entity. For example, in image retrieval x_i might be a pixel-based or visual representation of an image, and y_i may be a keyword or textual description of the same image. Another relevant setting is that of time series analysis where x_i might be the signal at time i and y_i might be the often highly related signal at time $i + 1$.

Canonical correlation analysis (CCA) is a linear dimensionality reduction technique dating back to Hotelling in 1936 that attempts to reduce dimensionality of the two data views jointly. This is fruitful when the relationship between the two views reflects useful latent structure underlying each views.

Let us introduce a bit of new notation to make our future calculations more apparent. Hereafter we will let x represent a random vector distributed uniformly over the datapoints $\{x_1, \ldots, x_n\}$, so that it takes on value x_i with probability $1/n$. The same holds for y.

The CCA objective is to find projection directions called **canonical directions** u and v such that the correlation between $u^T x$ and $v^T y$ is maximized. That is, in CCA, we focus on how the **canonical variables** $u^T x$ and $v^T y$ are related and not on how much they vary individually (which is the primary concern of PCA). Note that

$$\mathrm{Var}(u^T x) = \frac{1}{n} \sum_{i=1}^n u^T x_i x_i^T u = \frac{1}{n} u^T X^T X u$$

$$\mathrm{Cov}(u^T x, v^T y) = \frac{1}{n} \sum_{i=1}^n u^T x_i y_i^T v = \frac{1}{n} u^T X^T Y v$$

$$Corr(u^T x, v^T y) = \frac{\mathrm{Cov}(u^T x, v^T y)}{\sqrt{Var(u^T x) Var(v^T y)}} = \frac{u^T X^T Y v}{\sqrt{u^T X^T X u \; v^T Y^T Y v}}.$$

This expression makes clear that the canonical variables are invariant to rotations or scalings of either data set.

12.3.1 Solving CCA

The CCA optimization problem is the following maximization

$$
\begin{aligned}
&\max_{u,v} \frac{u^T X^T Y v}{\sqrt{u^T X^T X u v^T Y^T Y v}} && (12.1)\\
\Leftrightarrow\ &\max_{u,v\ \ s.t.\ \ \|Xu\|_2=1, \|Yv\|_2=1} u^T X^T Y v && (12.2)\\
\Leftrightarrow\ &\max_{u,v\ \ s.t.\ \ \left\|\begin{pmatrix} X & 0 \\ 0 & Y \end{pmatrix}\begin{pmatrix} u \\ v \end{pmatrix}\right\|_2 = \sqrt{2}} \begin{pmatrix} u^T & v^T \end{pmatrix} \begin{pmatrix} 0 & X^T Y \\ Y^T X & 0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}, && (12.3)
\end{aligned}
$$

where the second formulation takes the form of a standard **generalized singular value problem**, and the third takes the form of a **generalized eigenvalue problem**.

The equivalence of (12.2) and (12.3) may not be apparent, as we constrained both $\|Xu\|_2$ and $\|Yv\|_2$ in (12.2) and constrained only the square root of their sum in (12.3). However, we will see that the solution to (12.3) must satisfy $\|Xu\|_2 = \|Yv\|_2$, which implies that the two formulations are in fact equivalent. Indeed, the solution to the generalized eigenvalue problem (12.3) is given by the generalized eigenvector $\begin{pmatrix} u^* \\ v^* \end{pmatrix}$ which satisfies

$$
\begin{pmatrix} 0 & X^T Y \\ Y^T X & 0 \end{pmatrix} \begin{pmatrix} u^* \\ v^* \end{pmatrix} = \lambda^* \begin{pmatrix} X^T X & 0 \\ 0 & Y^T Y \end{pmatrix} \begin{pmatrix} u^* \\ v^* \end{pmatrix}
$$

for the largest value of λ^* (the associated generalized eigenvalue). If this $\lambda^* = 0$, this expression implies that $u^{*T} X^T X u^* = 0 = v^{*T} Y^T Y v^*$. Otherwise, $u^{*T} X^T X u^* = u^{*T} X^T Y v^* \frac{1}{\lambda^*} = v^{*T} Y^T Y v^*$. In either case, we have the advertised equality $\|Xu^*\|_2 = \|Yv^*\|_2$.

Note moreover that if $X^T X$ and $Y^T Y$ are invertible, the above problem reduces to a standard eigenvalue problem finding $\begin{pmatrix} u^* \\ v^* \end{pmatrix}$ satisfying

$$
\begin{pmatrix} (X^T X)^{-1} & 0 \\ 0 & (Y^T Y)^{-1} \end{pmatrix} \begin{pmatrix} 0 & X^T Y \\ Y^T X & 0 \end{pmatrix} \begin{pmatrix} u^* \\ v^* \end{pmatrix} = \lambda^* \begin{pmatrix} u^* \\ v^* \end{pmatrix}
$$

for the largest λ^*.