02460 - Large scale modeling of functional MRI using Sparse Probabilistic Principal Component Analysis

Project notes and plans

Supervisor: Morten Mørup
Co-supervisor Jesper Løve Hinrich

Last updated April 7, 2016

1 General info

1. There is an error in the update formula for Σ_{W_i} in the SPPCA article by Guan and Dy [1]. Where the matrix $dg(z_i)$ should be multiplied from the left (as shown below), not from the right (as written in the article)

$$\Sigma_{W_{i,\cdot}} = dg(\langle z_{i,\cdot}\rangle)\left[\langle\sigma^{-2}\rangle\sum_{n=1}^{N}\langle\mathbf{x}_n\mathbf{x}_n^\top\rangle dg(\langle z_{i,\cdot}\rangle) + \mathbf{I}\right]^{-1}$$

2. The Matrix Cookbook contains a lot of useful matrix properties and can be found here
`http://www2.imm.dtu.dk/pubdb/views/edoc_download.php/3274/pdf/imm3274.pdf`

3. Especially the trace operator is important, as it is a very useful tool in the derivation of VB and will also help in making your models scalable. The trace of a matrix is the sum of its diagonal elements (see Section 2.1).

4. There is another error in Gaun and Dy [1]. In equation (31), it states that $\mu_{z_{ij}} = \mathbf{W}_{ij}^2$ but it should be $\mu_{z_{ij}} = \langle\,\mathbf{W}_{ij}^2\,\rangle = \mathbf{\Sigma}_{\mathbf{W}_i}^{(j,j)} + \langle\,\mathbf{W}_i j\,\rangle^2$, where (j,j) is an element of the covariance matrix.

1.1 Groups

I will refer to the groups by numbers 1 through 3 (For easy and because a.t.m. I'm unsure which people are in which groups)

- Group 1 (Victor Salling, Georgios Papoutsakis, Vincent Beliveau)
- Group 2 (Christian Eschen, Peter Bruun-Rasmussen)
- Group 3 (Marco Dal Farra, Nicolai Riis, Jacob Frøsig, Casper Eriksen)

The group numbers also indicate which dataset you should be analysing. (Just the single subject data for the moment.)

2 Tips and Tricks

2.1 The trace operator

Given a matrix $\mathbf{A}$ of size $d \times d$,then

$$\text{trace}(\mathbf{A}) = \sum_{i=1}^{d} \mathbf{A}_{ii}$$

Note that applying the trace always results in a scalar (by definition), so if we have a scalar b, then

$$\text{trace}(b) = b$$

Which is a trivial relation, but consider the matrix-vector product

$$\mathbf{x}^\top \mathbf{\Sigma} \mathbf{x}$$

where $\mathbf{x}$ is a column vector and $\mathbf{\Sigma}$ is a square matrix, then the product is a scalar, so

$$\mathbf{x}^\top \mathbf{\Sigma} \mathbf{x} = trace(\mathbf{x}^\top \mathbf{\Sigma} \mathbf{x})$$

Now using the cyclic property of the trace operator,

$$trace(\mathbf{x}^\top \mathbf{\Sigma} \mathbf{x}) = trace(\mathbf{\Sigma} \mathbf{x} \mathbf{x}^\top)$$

which comes in handy when we want to calculate the expected value in our

The cyclic property, says that for any matrix product, we can permute the order in the following way (here showed with three matrices)

$$\text{trace}(\mathbf{ABC}) = \text{trace}(\mathbf{BCA}) = \text{trace}(\mathbf{CAB}) = \text{trace}(\mathbf{ABC})$$

2.2 Short tips

- Remember that a covariance matrix is positive-semi-definite and symmetric.

2.3 Notes on numerical stability

2.3.1 Of the trace operator

Given two matrices $\mathbf{A}^{d \times N}, \mathbf{B}^{d \times N}$ then calculating the trace of their product can be done in a more numerically stable and computational efficient way, than simply $trace(\mathbf{AB})$ or $trace(\mathbf{BA})$ (cyclic property) which has asymptotic time complexity of $O(d^2 N)$ and $O(N^2 d)$, respectively. Which is as the sum over all element-wise multiplications

$$\text{trace}(\mathbf{AB}) = \sum_{i=1}^{d} \sum_{n=1}^{N} a_{in} b_{in} \tag{1}$$

which in Matlab would be `sum(sum(A.*B))`. Note the dimensions of $\mathbf{A}$ and $\mathbf{B}$ must be of equal size.

	Group 1	Group 2	Group 3
spData	7	6	4
Sdata	10	14	14
Fdata	2	2	2
multi			
spData	7	6	4
Sdata	13	11	15
Fdata	2	2	2

Table 1: True underlying number of components.

3 Nots on Synthetic Datasets

Note that the number of underlying components q will vary from group to group, while the observations N and dimensionality d are same across groups (but varies from spData, Fdata and Sdata).

3.1 spData

This data is generated from the SPPCA model [1]. The ground truth data contains $\mathbf{W}^{d\times q}, \mathbf{X}^{q\times N}, \boldsymbol{T}^{d\times N}, \boldsymbol{\mu}^{d\times 1}$ and a scalar value for the variance of the noise σ.

$$P(\boldsymbol{T}|\mathbf{W}, \mathbf{X}, \boldsymbol{\mu}, \sigma) = \prod_{n=1}^{N} \mathcal{N}\left(\mathbf{t}_n|\mathbf{W}\mathbf{x}_n + \boldsymbol{\mu}, \sigma\mathbf{I}_d\right) \tag{2}$$

3.2 Fdata

This data is generated similar to the synthetic data used by [1]. The ground truth data contains $\boldsymbol{V_1}^{1\times d}, \boldsymbol{V_2}^{1\times d}, \mathbf{X}^{d\times N}, \mathbf{X}_{real}^{d\times N}$, and a scalar value for the variance of the noise σ.

Here $\boldsymbol{V_1}, \boldsymbol{V_2}$ are the two underlying factors. $\mathbf{X}$ is the data with added noise $\epsilon_{ij} \sim \mathcal{N}(0, 10\sigma)$, and $\mathbf{X}_{real}$ is the data without noise.

3.3 Sdata

This data is generated from a sparse $\mathbf{W}$, you should ignore the variable `cvar`. The relation between the remaining matrices is

$$P(\boldsymbol{T}|\mathbf{W}, \mathbf{X}, \boldsymbol{\mu}, \sigma) = \prod_{n=1}^{N} \mathcal{N}\left(\mathbf{t}_n|\mathbf{W}\mathbf{x}_n + \boldsymbol{\mu}, \sigma\mathbf{I}_d\right) \tag{3}$$

Further, $\boldsymbol{\mu} \sim \mathcal{N}(\mathbf{0}, 100\mathbf{I}_d), \mathbf{x}_n \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_q)$.

References

[1] Yue Guan and Jennifer G Dy, "Sparse probabilistic principal component analysis," in *International Conference on Artificial Intelligence and Statistics*, 2009, pp. 185–192.