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

Lecture 17 — June 2

Lecturer: Lester Mackey *Scribe: Chris Davis*

17.1 PCA with Missing Data

In the last lecture, we posed the question of how to carry out unsupervised learning in the presence of missing data, and we began to explore a potential solution, adapting unsupervised procedures to directly cope with data missingness. As a first example, we considered carrying out principal component analysis in situations in which some feature values are missing. We defined an observation set Ω satisfying $(i, j) \in \Omega$ if and only if component x_{ij} is observed, a missing data objective

$$\min_{M\in\mathbb{R}^{n\times p}} \sum_{(i,j)\in\Omega} (x_{ij} - m_{ij})^2 \quad \text{s.t. } \operatorname{rank}(M) \leq k, \tag{17.1}$$

and a scheme to approximately solve this problem, **iterated SVD**. Unfortunately, this method is unsatisfying for large datasets with large amounts of missingness (such as the Netflix Prize dataset, where fewer than 1% of features are observed), as simply imputing and storing all np entries may be prohibitively expensive. In this lecture, we will consider more scalable approaches to (approximately) solve the missing data PCA problem (17.1).

17.1.1 Factorized Approaches

Factorized approaches to solving (17.1) take advantage of the fact that $\operatorname{rank}(M) \leq k$ if and only if $M = AB^T$ for $A \in \mathbb{R}^{n\times k}, B \in \mathbb{R}^{p\times k}$. Using this fact, we can transform (17.1) into a biconvex problem in A and B:

$$\min_{A\in\mathbb{R}^{n\times k}, B\in\mathbb{R}^{p\times k}} \sum_{(i,j)\in\Omega} (x_{ij} - a_i^T b_j)^2 + \sum_{i=1}^{n} \lambda_i \parallel a_i \parallel_2^2 + \sum_{j=1}^{p} \tilde{\lambda}_j \parallel b_j \parallel_2^2 \tag{17.2}$$

The terms involving tuning parameters $\lambda_i, \tilde{\lambda}_j \geq 0$ are commonly added to regularize the problem. It is common to set $\lambda_i = \lambda$ and $\tilde{\lambda}_j = \lambda$ or $\lambda_i = \lambda n_i$ and $\tilde{\lambda}_j = \lambda p_j$, for n_i the number of times i appears in Ω, p_j the number of times j appears in Ω, and $\lambda \geq 0$ a common tuning parameter. The latter setting is particularly effective when when missingness is non-uniform across rows or columns.

We still have no general closed form solution for this problem, but because this minimization is biconvex in A and B, a natural strategy is to perform block coordinate descent on A and B. This is called **alternating least squares**. Conveniently, when B is fixed, the problem decouples into n independent ridge regression problems depending on parameters

a_i. This yields closed form updates that are not only easier to calculate, but also very parallelizable. The situation is similar when A is fixed (the problem decouples into p independent ridge regression problems depending on parameters b_j). Surprisingly, recent work has shown that, with appropriate initialization and under appropriate assumptions on X, this technique admits reconstruction accuracy guarantees for the missing entries (see the online readings).

A second popular and scalable approach to optimizing (17.2) is **stochastic gradient descent**. Here we update parameters using unbiased estimate of the objective gradient by sampling terms uniformly from Ω. Let . We can pull the regularization parameters into the sum over observed entries as follows:

$$\sum_{(i,j)\in\Omega}\left[(x_{ij}-a_i^T b_j)^2+\frac{\lambda_i}{n_i}\parallel a_i\parallel_2^2+\frac{\tilde{\lambda}_j}{p_j}\parallel b_j\parallel_2^2\right]. \tag{17.3}$$

A typical stochastic gradient algorithm then repeats the following until convergence:

- Sample (i, j) uniformly from Ω. (Alternatively and more commonly, iterate over the entries (i, j) in Ω.)
- Update:

$$a_i \leftarrow a_i-\gamma\left(b_j(b_j^T a_i-x_{ij})+\frac{\lambda_i}{n_i}a_i\right) \tag{17.4}$$

$$b_j \leftarrow b_j-\gamma\left(a_i(a_i^T b_j-x_{ij})+\frac{\tilde{\lambda}_j}{p_j}b_j\right) \tag{17.5}$$

  where γ is the (tunable) learning rate. Note that we *only* update parameters for the entry (i, j) selected in the previous step.

This method uses simple and cheap updates, which scale well to larger problems. Recent work has shown that this algorithm can also be parallelized with little loss. While this method has been shown to yield accurate reconstructions in practice, there is as of yet little theory supporting its success.

17.1.2 Convex Approaches

Instead of attempting to optimize the original non-convex problem directly, we will next attempt to solve a related convex optimization problem. More precisely, we will replace the rank constraint $\text{rank}(M) \leq k$ with a convex constraint or penalty. Two surrogates well-suited for this purpose are the **nuclear norm** and the **max norm**. The nuclear norm (also known as the trace norm) is the sum of all the singular values of M

$$\parallel M\parallel_*=\sum_i\sigma_i(M)=\parallel\sigma(M)\parallel_1 . \tag{17.6}$$

The relationship between the nuclear norm and the rank of a matrix $\text{rank}(M)=\sum_i\mathbb{I}\left(\sigma_i(M)\neq 0\right)=\parallel\sigma(M)\parallel_0$ parallels the relationship between the ℓ_1 norm and the ℓ_0 cardinality of a vector. The max norm is less commonly used and can be defined as

$$\parallel M\parallel_{\max}=\min_{M=AB^T}\parallel A\parallel_{2,\infty}\parallel B\parallel_{2,\infty}, \tag{17.7}$$

where we define $\| \cdot \|_{2,\infty}$ as the max ℓ_2 row norm:

$$\| A \|_{2,\infty} = \max_{\|u\|_2=1} \| A_u \|_\infty \ . \tag{17.8}$$

A typical convex surrogate formulation for our missing data PCA problem (17.1) goes by the name **nuclear norm heuristic** and takes the form

$$\min_{M \in^{n\times p}} \sum_{(i,j)\in\Omega} (x_{ij} - m_{ij})^2 + \lambda \| M \|_* \ , \tag{17.9}$$

where λ is a tunable regularization parameter.

There are several benefits to the convex approach. The convexity implies that a *global* optimum can be found in polynomial time. Moreover, under various assumptions on X and Ω, reconstruction accuracy guarantees are available (see, for example, Candes & Recht '08). Further, many algorithms are available for solving (17.9). One particularly effective example is the **accelerated proximal gradient** algorithm (see the paper of Toh and Yun in the reading). The biggest drawback of this approach is that the methods developed for solving (17.9) are far less scalable than the factorized approaches we have discussed. For example, many of the leading convex optimization methods for (17.9) need to perform repeated truncated singular value decompositions of large matrices (say, once per iteration of gradient descent). Improving the scalability of convex optimization approaches to missing data modeling is an active area of research.

17.1.3 Relation between Nuclear Norm Heuristic and Factorized Approaches with Regularization

A useful fact about the nuclear norm is the following:[1]

$$\| M \|_* = \min_{M=AB^T} \frac{1}{2} \sum_i \| a_i \|_2^2 + \frac{1}{2} \sum_j \| b_j \|_2^2 \tag{17.10}$$

We can use (17.10) to show that (17.9) is actually equivalent to

$$\min_{A,B} \sum_{(i,j)\in\Omega} (x_{ij} - a_i^T b_j)^2 + \frac{\lambda}{2} \sum_i \| a_i \|_2^2 + \frac{\tilde{\lambda}}{2} \sum_j \| b_j \|_2^2 \tag{17.11}$$

which looks very similar to (17.2). The main difference is that there is no rank constraint on A and B. If we generalize (17.11) by introducing different weights λ_i for each row i and $\tilde{\lambda}_j$ for each column j, we obtain what is called a **weighted trace norm** (see the optional reading).

[1] This is left as an exercise to the reader.

17.2 K-means with Missing Data

The primary lesson from the example of PCA with missingness is that a viable strategy for dealing with missingness is to phrase an unsupervised learning task as data reconstruction and then only attempt to reconstruct the observed data entries. We now show that this approach works for k-means clustering as well. Recall that our k-means objective for complete data is given by

$$\min_{z_{1:n}, m_{1:k}} \sum_{i,j} (x_{ij} - m_{z_i j})^2, \tag{17.12}$$

where $z_{1:n}$ are the cluster indicators and $m_{1:k}$ the cluster centers. We can account for missingness by summing only over observed entries

$$\min_{z_{1:n}, m_{1:k}} \sum_{(i,j) \in \Omega} (x_{ij} - m_{z_i j})^2 \ . \tag{17.13}$$

We may now find an approximate solution by alternating between the following two updates until convergence (as in Lloyd's algorithm):

- Assign each x_i to the cluster with the closest center as measured by the observed features.
- From these assignments update m_{cj}:

$$m_{cj} = \frac{\sum_i \mathbb{I}\left(z_i = c\right) \mathbb{I}\left((i,j) \in \Omega\right) x_{ij}}{\sum_i \mathbb{I}\left(z_i = c\right) \mathbb{I}\left((i,j) \in \Omega\right)} \ . \tag{17.14}$$

This is the mean over the observed j features in cluster c.

17.3 End of Quarter Summary

We end the quarter with a reflection on what we have accomplished. Overall, we have surveyed a slew of popular and practical methods for unsupervised learning. These included latent class and clustering, like k-means and Gaussian mixture models, latent feature and dimensionality reduction models like principal component analysis and independent component analysis. We contrasted probabilistic and model free methods (GMM vs k-means; probablistic PCA vs factor analysis), explored the advantages of hierarchical versus flat clustering and the benefits of deep learning, and learned to introduce available dependence structures into our modeling, such as with hidden markov models.

We identified a number of the important challenges that arise in the unsupervised learning setting along with some partial solutions. We examined the issues of model selection, such as choosing a number of components or clusters, grappled with the difficulties of initialization and suboptimal solutions, and confronted the pervasive problem of evaluating our unsupervised models. We also discussed strategies for making our analyses more interpretable and for coping with missing features.

Despite our quarter of focus, quite a lot remains to be explored in the area of unsupervised learning. For instance, we made little mention of **Bayesian approaches** to unsupervised learning this quarter, nor have we explored **subspace clustering** methods for separating datapoints belonging to disparate lower-dimensional subspaces or **spectral method of moment techniques** which provide a modern alternative to EM for fitting discrete latent variable models. Please review the course project page for some references on these topics.

If you are interested in learning about other modern methods in applied statistics, you should consider the Modern Applied Statistics sequence Stats 315 A / B.