6.7900 Fall 2024: Lecture Notes 18

1 Missing data

Recap: In the previous lecture, we introduced the different types of missing data, categorized as follows:

1. **Missing Completely at Random (MCAR)**. Missingness is independent of both observed and latent variables X, Y.
2. **Missing at Random (MAR)**. Missingness depends only on observed data (not latent variables)
3. **Missing Not at Random (MNAR or NMAR)**. Missingness can depend on something besides the observed data.

1.1 What to do when data is missing?

Why do anything? Many standard machine learning algorithms need complete data to run. Additionally, in order to ensure quality of results we might need to effectively address missing values. A few options for addressing missing data:

1. **Visualization**. Start by visualizing data to identify patterns of missingness.
2. **Discard data**. Remove data points or features with missing values.
    - **Pros:** Simple to implement and often the default in software.
    - **Cons:** Leads to loss of data and potential bias if data is not MCAR.
3. **Information in the missingness**. We might have some useful information in whether or not a value is missing. Treat missingness as a feature or category, especially when working with categorical data.
4. **Single imputation**. Replace missing values with a single estimate. E.g. use the mean of the observed value or a random sample from the observed values (need MCAR assumption to reduce bias). Alternatively use regression to estimate the missing features from the other observed features (MAR assumption,

see readings to better understand the connection between this method and the MAR assumption).

5. **Multiple imputation**. Similarly to single imputation, we aim to estimate the missing value (e.g. using regression), but we want to add some noise/uncertainty about the missing data values. Generate multiple estimates of missing values to account for uncertainty.

6. **Full Bayesian model**. Model is also over the x values (as opposed to only modeling $p(y|x)$ as we have done in the past). This approach requires careful modeling.

2 Dimensionality reduction

2.1 Principle Component Analysis (PCA)

Motivating example for Principle Component Analysis. We often make lots of noisy and possibly redundant measurements to try to understand some underlying phenomenon.

Consider a ball attached to a spring that moves only along a single axis (w_1). Although there are three spatial dimensions (w_1, w_2, w_3), the motion of the ball is confined to one axis. Imagine using three cameras to record this movement, resulting in six redundant and noisy features ($x_1, x_2, \ldots, x_6$). PCA helps reduce this redundancy by identifying the primary axis of variation.

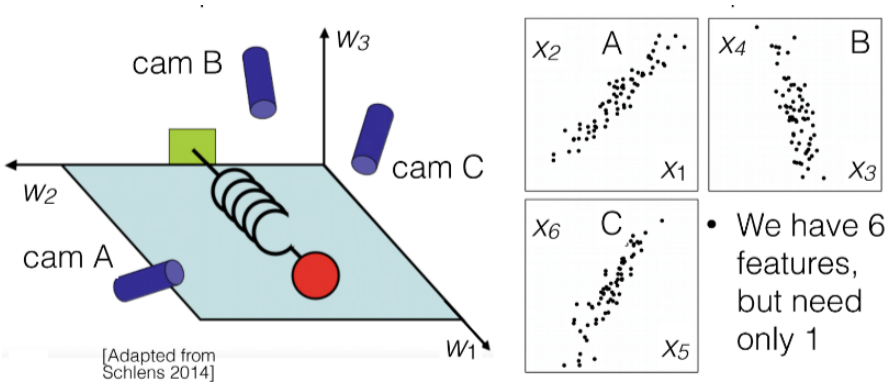

Figure 1: Motivating example for PCA. The ball is only moving in the w_1 axis, but we get 6 features ($x_1, \ldots x_6$).

Problem setup for PCA

- $x^{(n)}$ is a $D \times 1$ vector
- Pre-process by zero-centering so $\frac{1}{N}\sum_{n=1}^{N} x^{(n)} = 0_D$
- Assume: we want to approximate the data with its projection onto a low-dimensional subspace with orthonormal basis $w_1, ..., w_L$, which are the principle components. $x^{(n)} \approx \sum_{\ell=1}^{L} z_\ell^{(n)} w_\ell = W z^{(n)} =: \hat{x}^{(n)}$

> **Note:** $\hat{x}^{(n)}$ can be defined as a linear combination of the W basis since we assume that they are orthonormal basis (linear sub-space assumption).

- Goal: the projection should be "close" to the original data (square loss):

$$\min \sum_{n=1}^{N} \|x^{(n)} - \hat{x}^{(n)}\|^2 = \min_{W,Z} \|X^T - WZ^T\|_F^2 = \min_{W,Z} \|X - ZW^T\|_F^2$$

  - Optimization: the optimization is over $W(D \times L)$ and $Z(N \times L)$, where W is the collection of principle components (orthonormal basis for the subspace). Z is telling us where in the subspace W we are projecting our x, so W is the direction and Z are the weights.
  - Constraint: W represents an orthonormal basis.

> **Note:** if we find the best basis W, we could instead use -1 times any basis vector, -1 times the corresponding z values, and get the same result

Solving when $L = 1$:

Assume that the data is zero-centered: $\sum_{n=1}^{N} x^{(n)} = 0_D$. Let w_1 represent an orthonormal basis (a unit vector).

For $L = 1$, we aim to minimize the projection error:

$$\min_{w_1,z} \sum_{n=1}^{N} \|x^{(n)} - z_1^{(n)} w_1\|^2.$$

Expanding the squared error:

$$\sum_{n=1}^{N} \left((x^{(n)})^\top x^{(n)} - 2 z_1^{(n)} w_1^\top x^{(n)} + (z_1^{(n)})^2 w_1^\top w_1 \right).$$

Simplify by dropping terms constant in $z_1^{(n)}$ and w_1, and noting that $w_1^\top w_1 = 1$:

$$\min_{w_1,z} \sum_{n=1}^{N} -2 z_1^{(n)} w_1^\top x^{(n)} + (z_1^{(n)})^2.$$

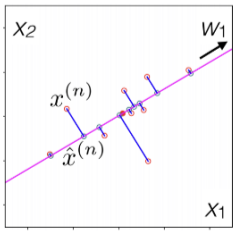

Figure 2: Visualization of the projection $x^{(n)}$

Approach for solving this optimization: alternate between optimizing z and w.

- We take the derivative with respect to $z_1^{(n)}$ (for each n) and set it to 0 (we also check the second order condition that the objective is convex). The solution is $z_1^{(n)} = w_1^\top x^{(n)}$. Note that $w_1^\top x^{(n)}$ is the scalar projection of the data in the w_1 direction.
- Next, in order to find w_1, we plug in $z_1^{(n)} = w_1^\top x^{(n)}$ in the objective. So we want to minimize

$$-\sum_{n=1}^{N}(w_1^\top x^{(n)})^2 = -w_1^\top \left[\sum_{n=1}^{N} x^{(n)}(x^{(n)})^\top\right] w_1$$

  Since we assume that $\sum_{n=1}^{N} x^{(n)} = 0_D$, then the term $\sum_{n=1}^{N} x^{(n)}(x^{(n)})^\top$ is the empirical covariance $\hat{\Sigma}$. We also need to include the constraint on W that is an orthonormal basis (since $L = 1$, the constraint reduces to $w_1^\top w_1 = 1$), so we use Lagrangian and optimize:

$$w_1^\top \hat{\Sigma} w_1 - \lambda_1(w_1^\top w_1 - 1)$$

  We take the derivative with respect to w_1 and set it to 0: $\hat{\Sigma} w_1 = \lambda_1 w_1$, so the best w_1 is an eigenvector of the empirical covariance. In order to find which eigenvector it is, we plug it back into the objective:

$$-w_1^\top \hat{\Sigma} w_1 = -\lambda_1 w_1^\top w_1 = -\lambda_1$$

  Note that in order to minimize the objective, we want the maximum value λ_1. So the best w_1 is the eigenvector of covariance with the largest eigenvalue.

Notes on the solution

For $L > 1$, we can solve inductively (more details in the book by Murphy). When we get to the Lagrangian step, we will use the constraints: $w_\ell^\top w_\ell = 1$ and $\forall k \in \{1, ..., \ell - 1\}, w_\ell^\top w_k = 0$:

- Next basis vector is a unit vector
- Next basis vector is orthogonal to all previous vectors