Deep Learning - Foundations and Concepts

Chapter 16. Continuous Latent Variables

nonlineark@github

April 7, 2025

Outline

Maximum variance formulation

Problem

Consider a data set of observations $\{x_n\}$ where $n = 1, \ldots, N$, and x_n is a Euclidean variable with dimensionality D. Our goal is to project the data onto a space having dimensionality $M < D$ while maximizing the variance of the projected data.

Maximum variance formulation

Let's calculate the variance of the projected data on a unit direction v:

$$y_n = x_n \cdot v$$

$$E(y_n) = \frac{1}{N}\sum_{n=1}^{N} y_n = E(x_n) \cdot v$$

$$\begin{aligned}\mathrm{var}(y_n) &= \frac{1}{N}\sum_{n=1}^{N}(y_n - E(y_n))^2 = \frac{1}{N}\sum_{n=1}^{N}((x_n - E(x_n)) \cdot v)^2 \\ &= \frac{1}{N}\sum_{n=1}^{N} v^T(x_n - E(x_n))(x_n - E(x_n))^T v = v^T S v\end{aligned}$$

where S is the data covariance matrix defined by:

$$S = \frac{1}{N}\sum_{n=1}^{N}(x_n - E(x_n))(x_n - E(x_n))^T$$

Maximum variance formulation

Let's find the unit direction v_1 for the largest variance. Suppose that $\lambda_1 \geq \cdots \geq \lambda_D$ are the D eigenvalues of S, and their corresponding orthonormal eigenvectors are $u_1, \ldots, u_D$ respectively. Let $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_D)$, $U = \begin{pmatrix} u_1 & \cdots & u_D \end{pmatrix}$. We have:

$$v_1 = U\alpha_1$$
$$v_1^T S v_1 = \alpha_1^T U^T S U \alpha_1 = \alpha_1^T \Lambda \alpha_1 \leq \lambda_1 ||\alpha_1||^2 = \lambda_1$$

The equality holds if and only if v_1 is an eigenvector corresponds to the largest eigenvalue λ_1. Without loss of generality, we could set $v_1 = u_1$.

Maximum variance formulation

Let's find the unit direction v_2 for the second largest variance. Because v_2 is orthogonal to v_1 thus u_1, in the coordinate system formed by the orthonormal basis $u_1, \ldots, u_D$, its first coordinate is 0:

$$v_2 = U\alpha_2$$
$$v_2^T S v_2 = \alpha_2^T \Lambda \alpha_2 \leq \lambda_2 ||\alpha_2||^2 = \lambda_2$$

Again, the equality holds if and only if v_2 is an eigenvector corresponds to the second largest eigenvalue λ_2. Without loss of generality, we could set $v_2 = u_2$.

Maximum variance formulation

If we consider the general case of an M-dimensional projection space, the optimal linear projection for which the variance of the projected data is maximized is now defined by the M eigenvectors $u_1, \ldots, u_M$ of the data covariance matrix S corresponding to the M largest eigenvalues $\lambda_1, \ldots, \lambda_M$.

Minimum-error formulation

We now discuss an alternative formulation of PCA based on projection error minimization:

- We want to find an orthonormal basis $u_1, \ldots, u_D$, where the M-dimensional linear subspace can be presented by the first M of the basis vectors.
- Each data point x_n is approximated by $\tilde{x}_n = \sum_{i=1}^{M} z_{ni} u_i + \sum_{i=M+1}^{D} b_i u_i$.

such that the squared distance between the original data point x_n and its approximation $\tilde{x}_n$, averaged over the data set:

$$J = \frac{1}{N} \sum_{n=1}^{N} ||x_n - \tilde{x}_n||^2$$

is minimized.

Minimum-error formulation

$$0 = \frac{\partial J}{\partial z_{ni}} = -\frac{2}{N}(x_n^T u_i - z_{ni}) \implies z_{ni} = x_n^T u_i$$

$$0 = \frac{\partial J}{\partial b_i} = -\frac{2}{N}\sum_{n=1}^{N}(x_n^T u_i - b_i) \implies b_i = (E(x_n))^T u_i$$

$$x_n - \tilde{x}_n = \sum_{i=M+1}^{D}((x_n - E(x_n)) \cdot u_i)u_i$$

$$J = \frac{1}{N}\sum_{n=1}^{N}||x_n - \tilde{x}_n||^2 = \sum_{i=M+1}^{D} u_i^T S u_i$$

Minimum-error formulation

We recognize that J is the total variance of the projected data on the unit directions $u_{M+1}, \ldots, u_D$. To minimize J, $u_{M+1}, \ldots, u_D$ should be the eigenvectors corresponding to the smallest $D - M$ eigenvalues of S, and hence the eigenvectors defining the principal subspace are those corresponding to the M largest eigenvalues.

Data compression

One application for PCA is data compression:

$$\tilde{x}_n = \sum_{i=1}^{M}(x_n \cdot u_i)u_i + \sum_{i=M+1}^{D}(E(x_n)\cdot u_i)u_i = E(x_n) + \sum_{i=1}^{M}((x_n - E(x_n))\cdot u_i)u_i$$

This represents a compression of the data set, because for each data point we have replaced the D-dimensional vector x_n with an M-dimensional vector.

Data whitening

Suppose we have a data set of observations $\{x^n\}$ where $n = 1, \ldots, N$, and x^n is a Euclidean variable with dimensionality D. We often want to transform the data set to standardize certain of its properties. For example, making a linear re-scaling of the individual variables such that each variable has zero mean and unit variance:

$$\bar{x}_d = \frac{1}{N} \sum_{n=1}^{N} x_d^n$$

$$\sigma_d^2 = \frac{1}{N} \sum_{n=1}^{N} (x_d^n - \bar{x}_d)^2$$

$$\tilde{x}_d^n = \frac{x_d^n - \bar{x}_d}{\sigma_d}$$

Data whitening

The covariance matrix for the standardized data has components:

$$\rho_{ij} = E(\tilde{x}_i^n \tilde{x}_j^n) - E(\tilde{x}_i^n)E(\tilde{x}_j^n) = \frac{1}{N}\sum_{n=1}^{N} \frac{x_i^n - \bar{x}_i}{\sigma_i} \frac{x_j^n - \bar{x}_j}{\sigma_j}$$

If two components x_i and x_j of the data are perfectly correlated, then $\rho_{ij} = 1$, and if they are uncorrelated, then $\rho_{ij} = 0$.

Data whitening

Using PCA we can make a more substantial normalization of the data to give it zero mean and unit covariance, so that different variables become decorrelated:

$$
\begin{aligned}
y^n &= \Lambda^{-\frac{1}{2}} U^T (x^n - \bar{x}) \\
E(y^n) &= 0 \\
E(y^n (y^n)^T) &= \frac{1}{N} \sum_{n=1}^{N} \Lambda^{-\frac{1}{2}} U^T (x^n - \bar{x})(x^n - \bar{x})^T U \Lambda^{-\frac{1}{2}} \\
&= \Lambda^{-\frac{1}{2}} U^T S U \Lambda^{-\frac{1}{2}} = \Lambda^{-\frac{1}{2}} \Lambda \Lambda^{-\frac{1}{2}} = I \\
\mathrm{cov}(y^n) &= E(y^n (y^n)^T) - E(y^n)(E(y^n))^T = I
\end{aligned}
$$

High-dimensional data

In some applications of PCA, the number of data points is smaller than the dimensionality of the data space. For such cases, we can calculate the eigenvalues and eigenvectors more efficiently this way:

- Let $X = \begin{pmatrix} x_1 - \bar{x} & \cdots & x_N - \bar{x} \end{pmatrix}^T$, then $S = \frac{1}{N} X^T X$.
- Calculate the eigenvalues and eigenvectors of $\frac{1}{N} X X^T$ instead, say $\frac{1}{N} X X^T v = \lambda v$.
- Then λ is an eigenvalue of S and $u = X^T v$ is an eigenvector of S.
  - $\frac{1}{\sqrt{N\lambda}} u$ is the corresponding unit eigenvector (suppose v is already a unit vector).