Machine Learning, Spring 2019
EN 2105
MWF 9-9:50am

Chao Lan
EN 4084A
clan@uwyo.edu

Dimensionality Reduction

Dimensionality reduction is the task of reducing features for e.g., data visualization, computation speedup or saving memory. It can be done by feature selection or feature transformation. The former selects a subset of observed features to represent subjects, while the latter generates a (smaller) set of latent features to represent subjects.

We will briefly introduce feature selection and then focus on feature transformation.

Feature selection ranks features based on certain criterion and selects higher-ranked ones. An example criterion is Fisher score. Given a sample of size n, the Fisher score of feature $x_{\cdot j}$ is

$$\text{FisherScore}(x_{\cdot j}) = \frac{\sum_{k=1}^{c} n_k \cdot (\mu_{kj} - \mu_j)^2}{\sum_{k=1}^{c} n_k \cdot \sigma_{kj}^2}, \tag{1}$$

where n_k is the number of instances in class k, μ_{kj} is the mean of $x_{\cdot j}$ in class k, μ_j is the mean of $x_{\cdot j}$ in the entire sample, and σ_{kj}^2 is the variance of $x_{\cdot j}$ in class k.

Fisher score ignores the impact of selected features on prediction performance, and thus belongs to the filter method for feature selection. Comparatively, the wrapper method selects features that maximize prediction performance. An example is stepwise forward selection (Alg 1).

Algorithm 1 Stepwise Forward Feature Selection

Input: a sample S, a prediction model f (with unfixed feature size)
Initialization: an empty set of selected features F
while stopping criterion is not met **do**
  1: for each $x_{\cdot k}$, train f on feature set $F \cup \{x_{\cdot k}\}$ and get prediction performance s_k
  2: add $x_{\cdot j}$ to F if it has the highest s_j
end while
Output: a selected feature set F

Wrapper method is computationally inefficient. The embedded method jointly selects features and trains prediction model. An example is Lasso.

Principle Component Analysis

PCA is a feature transformation method. Let $x \in \mathcal{R}^p$ be an instance and $w \in \mathcal{R}^p$ be a projection vector. PCA generates a latent feature $\tilde{x}$ by projecting x onto w, i.e.,

$$\tilde{x} = w^T x. \tag{2}$$

[*Discussion*] What is the geometric interpretation of the projection?

PCA learns a w that can maximally preserve the original data structure in the projected space. This criterion has two equivalent implementations: (1) maximize data variance and (2) minimize data reconstruction error. We will introduce both implementations.

Implementation 1: Maximize Data Variance

Based on (2), data variance in the projected space is

$$\Sigma_{\tilde{x}} = E\left[\,(\tilde{x} - E[\tilde{x}])^2\,\right] = E\left[\,(w^T x - E[w^T x])^2\,\right] = E\left[\,(w^T x - w^T E[x])^2\,\right]. \tag{3}$$

It can be estimated from a sample of size n (with mean μ and covariance Σ_x) as

$$\begin{aligned}\hat{\Sigma}_{\tilde{x}} &= \frac{1}{n}\sum\nolimits_{i=1}^{n}(w^T x_i - w^T\mu)^2 \\ &= \frac{1}{n}\sum\nolimits_{i=1}^{n}(w^T x_i - w^T\mu)\cdot(w^T x_i - w^T\mu)^T \\ &= w^T\Sigma_x w,\end{aligned} \tag{4}$$

[*Exercise*] Derive (4).

Maximizing data variance in the projected space can reduce data compression loss. PCA thus learns a w from S that can maximize $\hat{\Sigma}_{\tilde{x}}$ while restricting $||w||^2 = w^T w = 1$, i.e.,

$$\begin{aligned}&\max\nolimits_w\, w^T\Sigma_x w, \\ &s.t.\ w^T w = 1.\end{aligned} \tag{5}$$

[*Discussion*] Why does variance maximization reduce compression loss?

[*Discussion*] What if there is no restriction on $||w||$?

We can solve (5) using the Lagrange multiplier. The Lagrange function with multiplier λ is

$$J(w) = w^T\Sigma_x w + \lambda(w^T w - 1) = w^T(\Sigma_x + \lambda I)w - \lambda. \tag{6}$$

Since $J(w)$ is a quadratic function of w, we can optimize it by the critical point method. Solving

$$J'(w) = 2(\Sigma_x - \lambda I)w = 0 \tag{7}$$

gives

$$\Sigma_x w = \lambda w. \tag{8}$$

By definition w is an eigenvector of Σ_x and λ is the associated eigenvalue. Further, since

$$w^T\Sigma_x w = \lambda \tag{9}$$

is what PCA aims to maximize in (5), w should be the leading eigenvector[1].

The above analysis gives one optimal projection vector w_1. The next optimal project vector w_2 is obtained similarly, with an additional constraint that the generated features $w_2^T x$ and $w_1^T x$ are statistical correlated (to reduce feature redundancy). Thus given w_1, PCA finds w_2 by

$$\begin{aligned}&\max\nolimits_{w_2}\, w_2^T\Sigma_x w_2, \\ &s.t.\ w_2^T w_2 = 1, \quad cov(w_2^T x, w_1^T x) = 0.\end{aligned} \tag{10}$$

[1]Leading eigenvector is the one having the largest eigenvalue.

The covariance constraint can be simplified (for easier optimization) as follows.

$$\begin{aligned}
cov(w_2^T x, w_1^T x) &= \frac{1}{n}\sum\nolimits_{i=1}^{n}(w_2^T x_i - w_2^T \mu)\cdot(w_1^T x_i - w_1^T \mu)^T \\
&= \frac{1}{n}\sum\nolimits_{i=1}^{n} w_2^T (x_i-\mu)(x_i-\mu)^T w_1 \\
&= w_2^T\left(\frac{1}{n}\sum\nolimits_{i=1}^{n}(x_i-\mu)(x_i-\mu)^T\right)w_1 \\
&= w_2^T \Sigma_x w_1 \\
&= w_2^T(\lambda w_1) \\
&= \lambda w_2^T w_1.
\end{aligned} \tag{11}$$

[*Discussion*] In (11), why is the fifth equality true?

So the new optimization problem for w_2 is

$$\begin{aligned}
&\max\nolimits_{w_2} w_2^T \Sigma_x w_2, \\
&s.t.\ w_2^T w_2 = 1, \quad w_2^T w_1 = 0.
\end{aligned} \tag{12}$$

Applying the Lagrange multiplier method, we have the Lagrange function

$$J(w_2) = w_2^T \Sigma_x w_2 - \lambda_1(w_2^T w_2 - 1) - \lambda_2(w_2^T w_1). \tag{13}$$

Applying the critical point method, we are supposed to solve

$$J'(w_2) = 2\Sigma_x w_2 - 2\lambda_1 w_2 - \lambda_2 w_1 = 0. \tag{14}$$

We can show $\lambda_2 = 0$ by left-multiplying both sides of (14), i.e.,

$$w_1^T \Sigma_x w_2 - \lambda_1 w_1^T w_2 - \lambda_2 w_1^T w_1 = 0 \ \Rightarrow\ 0 - 0 - \lambda_2 = 0 \ \Rightarrow\ \lambda_2 = 0 \tag{15}$$

Thus we can obtain w_2 by just solving

$$J'(w_2) = 2\Sigma_x w_2 - 2\lambda_1 w_2 = 0, \tag{16}$$

which implies w_2 is an eigenvector of Σ_x associated with the second largest eigenvalue.

By similar arguments, the rest PCA projection vectors are eigenvectors of Σ_x associated with the remaining largest eigenvalues.

Implementation 2: Minimize Data Reconstruction Error

Let $W = \{w_1, \ldots, w_p\}$ be a basis of $\mathcal{R}^p$. By definition x can be linearly expressed by W, i.e.,

$$x_i = \alpha_{i1}w_1 + \alpha_{i2}w_2 + \ldots + \alpha_{ip}w_p = \sum\nolimits_{j=1}^{p}\alpha_{ij}w_j. \tag{17}$$

Since base vectors are orthogonal, we have the following important result

$$\alpha_{ij} = w_j^T x_i. \tag{18}$$

[*Exercise*] Prove (18).

An important interpretation is that reducing the dimension of x_i to d means selecting d base vectors to approximately express x_i. Without loss of generality, assume the first d base vectors are selected. The instance reconstructed from the reduced feature space can be expressed as

$$\tilde{x}_i = \sum\nolimits_{j=1}^{d} \alpha_{ij} w_j + \sum\nolimits_{j=d+1}^{p} c_j w_j, \tag{19}$$

where c_j is constant (thus $c_j w_j$ is a constant and only serves as a bias term).

PCA aims to learn $\{w_j\}$, $\{\alpha_{ij}\}$ and $\{c_j\}$ that minimize the following reconstruction error

$$J(w, \alpha, c) = \frac{1}{n} \sum\nolimits_{i=1}^{n} ||x_i - \tilde{x}_i||^2. \tag{20}$$

To minimize J, first expand its error term as

$$J = \frac{1}{n} \sum_i x_i^T x_i - \frac{2}{n} \sum_{i,\, j=1}^{j=d} \alpha_{ij} w_j^T x_i - \frac{2}{n} \sum_{i,\, j=d+1}^{j=p} c_j w_j^T x_i + \frac{1}{n} \sum_{i,\, j=1}^{j=d} \alpha_{ij}^2 + \frac{1}{n} \sum_{i,\, j=d+1}^{j=p} c_j^2. \tag{21}$$

Clearly this is a quadratic function of parameters. We can apply the critical point method.

Solving $\frac{\partial J}{\partial \alpha_{ij}} = 0$ for α_{ij} gives

$$\alpha_{ij} = w_j^T x_i. \tag{22}$$

Let $\mu = \frac{1}{n} \sum_i x_i$ be the sample mean. Solving $\frac{\partial J}{\partial c_j} = 0$ for c_j gives

$$c_j = w_j^T \mu. \tag{23}$$

[*Exercise*] Derive (21), (22) and (23).

Plugging all above back to J gives

$$x_i - \tilde{x}_i = \sum\nolimits_{j=d+1}^{p} w_j^T (x_i - \mu) w_j, \tag{24}$$

and thus the <u>reconstruction error</u> on x_i is

$$\begin{aligned} ||x_i - \tilde{x}_i||^2 = (x_i - \tilde{x}_i)^T (x_i - \tilde{x}_i) &= \sum\nolimits_{j,j'} (w_j^T (x_i - \mu) w_j)^T (w_{j'}^T (x_i - \mu) w_{j'}) \\ &= \sum\nolimits_{j,j'} [w_{j'}^T (x_i - \mu)][(x_i - \mu)^T w_j] w_j^T w_{j'} \\ &= \sum\nolimits_{j} w_j^T (x_i - \mu)(x_i - \mu)^T w_j. \end{aligned} \tag{25}$$

[*Exercise*] Derive (24) and the last equality in (25).

Plugging (25) back to J, the reconstruction error becomes

$$J = \frac{1}{n} \sum\nolimits_{i=1}^{n} ||x_i - \tilde{x}_i||^2 = \sum\nolimits_{i,j} w_j^T (x_i - \mu)(x_i - \mu)^T w_j = \sum\nolimits_{j=d+1}^{p} w_j^T \Sigma_x w_j, \tag{26}$$

The rest analysis is similar to the first implementation of PCA. We can first show that w_p is an eigenvector of Σ_x associated with the *smallest* eigenvalue, and then the rest $w_{p-1}, \ldots, w_{d+1}$ are also eigenvectors associated with the smallest eigenvalues. Since Σ_x's eigenvectors $w_1, w_2, \ldots, w_p$ form a basis, and the above analysis suggests not using $w_{d+1}, \ldots, w_p$, PCA will selects $w_1, \ldots, w_d$ as the projection vectors – and they are the eigenvectors of Σ_x associated with the largest eigenvalues. In practice one can choose d so that 80%-95% of the total eigenvalues are preserved.