CS 189 Introduction to Machine Learning
Fall 2017 DIS5

1 Derivation of PCA

In this question we will derive PCA. PCA aims to find the direction of maximum variance among a dataset. You want the line such that projecting your data onto this line will retain the maximum amount of information. Thus, the optimization problem is

$$\max_{u:\|u\|_2=1} \frac{1}{n}\sum_{i=1}^{n}\left(u^T x_i - u^T \hat{x}\right)^2$$

where n is the number of data points and $\hat{x}$ is the sample average of the data points.

(a) Show that this optimization problem can be written in this format:

$$\max_{u:\|u\|_2=1} u^T \Sigma u$$

where $\Sigma = \frac{1}{n}\sum_{i=1}^{n}(x_i-\hat{x})(x_i-\hat{x})^T$.

Solution:

We can modify the objective function (call it $f_0(u)$) in this way:

$$\begin{aligned}
f_0(u) &= \frac{1}{n}\sum_{i=1}^{n}\left(u^T x_i - u^T \hat{x}\right)^2 && (1)\\
&= \frac{1}{n}\sum_{i=1}^{n}\left((x_i-\hat{x})^T u\right)^2 && (2)\\
&= \frac{1}{n}\sum_{i=1}^{n}(u^T(x_i-\hat{x}))((x_i-\hat{x})^T u) && (3)\\
&= u^T\left(\frac{1}{n}\sum_{i=1}^{n}(x_i-\hat{x})(x_i-\hat{x})^T\right)u && (4)\\
&= u^T\Sigma u && (5)
\end{aligned}$$

(b) Show that the maximizer for this problem is equal to v_1, where v_1 is the eigenvector corresponding to the largest eigenvalue λ_1. Also show that optimal value of this problem is equal to λ_1.

Solution:

We start by invoking the spectral decomposition of $\Sigma = V\Lambda V^T$, which is a symmetric positive semi-definite matrix.

$$\begin{aligned}\max_{u:\|u\|_2=1} u^T\Sigma u &= \max_{u:\|u\|_2=1} u^T V\Lambda V^T u && (6)\\ &= \max_{u:\|u\|_2=1} (V^T u)^T \Lambda V^T u && (7)\end{aligned}$$

Here is an aside: note through this one line proof that left-multiplying a vector by an orthogonal (or rotation) matrix preserves the length of the vector:

$$\|V^T u\|_2 = \sqrt{(V^T u)^T (V^T u)} = \sqrt{u^T V V^T u} = \sqrt{u^T u} = \|u\|_2$$

I define a new variable $z = V^T u$, and maximize over this variable. Note that because V is invertible, there is a one to one mapping between u and z. Also note that the constraint is the same because the length of the vector u does not change when multiplied by an orthogonal matrix.

$$\max_{z:\|z\|_2=1} z^T \Lambda z = \max_z \sum_{i=1}^d \lambda_i z_i^2 \; : \; \sum_{i=1}^d z_i^2 = 1$$

From this new formulation, it is obvious to see that we can maximize this by throwing all of our eggs into one basket and setting $z_i^* = 1$ if i is the index of the largest eigenvalue, and $z_i^* = 0$ otherwise. Thus,

$$z^* = V^T u^* \implies u^* = V z^* = v_1$$

where v_1 is the ”principle” eigenvector, and corresponds to λ_1. Plugging this into the objective function, we see that the optimal value is λ_1.

2 Canonical Correlation Analysis

Assume that you have a database of images of the words typed in two different fonts. $X, Y \in \mathbb{R}^{n\times d}$ corresponds to the dataset of font 1 and font 2 respectively. Think of the database X as being composed on n independent draws (samples) from a random variable $\mathbf{X} \in \mathbb{R}^d$, and similarly Y as n draws from a random variable $\mathbf{Y}$. Your goal is to use machine learning to build a text recognition of word images.

(a) Explain why you would want to consider using CCA in this problem.

Solution: These two feature matrices share similar information. Using CCA helps removing redundant information between the two views. CCA is also useful in eliminating noise specific to only one view.

(b) Assume that X and Y include zero-mean features of the word images. Given two vectors $u, v \in \mathbb{R}^d$, what is the correlation coefficient of the projected variables? Correlation coefficient between two scalar random variables P and Q is computed by:

$$\rho(P,Q) = \frac{cov(P,Q)}{\sigma_P \sigma_Q}$$

Solution: The question is asking for $\rho(u^\top \mathbf{X}, v^\top \mathbf{Y})$ for random variables $\mathbf{X}$ and $\mathbf{Y}$. Writing out the algebra (notice that $u^\top \mathbf{X}$ and $v^\top \mathbf{Y}$ are zero mean), we have

$$\begin{aligned}
\rho(u^\top \mathbf{X}, v^\top \mathbf{Y}) &= \frac{\mathbb{E}[u^\top \mathbf{X} v^\top \mathbf{Y}]}{\sqrt{\mathbb{E}[u^\top \mathbf{X} u^\top \mathbf{X}]\mathbb{E}[v^\top \mathbf{Y} v^\top \mathbf{Y}]}} \\
&= \frac{\mathbb{E}[u^\top \mathbf{X}\mathbf{Y}^\top v]}{\sqrt{\mathbb{E}[u^\top \mathbf{X}\mathbf{X}^\top u]\mathbb{E}[v^\top \mathbf{Y}\mathbf{Y}^\top v]}} \\
&= \frac{u^\top \mathbb{E}[\mathbf{X}\mathbf{Y}^\top] v}{\sqrt{u^\top \mathbb{E}[\mathbf{X}\mathbf{X}^\top] u \mathbb{E}[v^\top \mathbb{E}[\mathbf{Y}\mathbf{Y}^\top] v}}.
\end{aligned}$$

Let us also outline a way to estimate the variance and covariance from data. Assume we have i.i.d. samples of the random variable $\mathbf{X}$, given by $x_1, x_2, \ldots, x_n \in \mathbb{R}^d$, and similarly for $\mathbf{Y}$. Also assume that we stack up these samples as rows to form the matrix $X \in \mathbb{R}^{n\times d}$, and similarly the matrix $Y \in \mathbb{R}^{n\times d}$. We now have the estimates:

$$\begin{aligned}
\Sigma_{XY} &= \mathbb{E}[\mathbf{X}\mathbf{Y}^\top] \approx \frac{1}{n}\Sigma_{i=1}^n x_i y_i^\top = \frac{1}{n}X^\top Y, \\
\Sigma_{XX} &= \mathbb{E}[\mathbf{X}\mathbf{X}^\top] \approx \frac{1}{n}\Sigma_{i=1}^n x_i x_i^\top = \frac{1}{n}X^\top X, \\
\Sigma_{YY} &= \mathbb{E}[\mathbf{Y}\mathbf{Y}^\top] \approx \frac{1}{n}\Sigma_{i=1}^n y_i y_i^\top = \frac{1}{n}Y^\top Y.
\end{aligned}$$

Plugging these into the definition of ρ yields our estimate of the correlation.

(c) Assume that the features of matrix X are rescaled to have values between -1 and 1. How does this change the correlation coefficient?

Solution: From the expression above, it can be seen that correlation coefficient is invariant to scalings of either datasets. For example, if we scale X by a constant, then the denominator and numerator of the correlation coefficient will scale equally so the correlation coefficient will not change.

(d) CCA aims to find the projection vectors u and v that maximizes the correlation coefficient. Show that CCA optimization problem can be written as the following assuming that $\|Xu\| = \|Yv\|$:

$$\max_{u,v} \begin{pmatrix} u^T & v^T \end{pmatrix} \begin{pmatrix} 0 & X^T Y \\ Y^T X & 0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}$$

subject to

$$\left\| \begin{pmatrix} X & 0 \\ 0 & Y \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} \right\| = \sqrt{2}$$

Solution: We compute the maximum of correlation coefficient:

$$\max_{u,v} \frac{u^T X^T Y v}{\sqrt{u^T X^T X u v^T Y^T Y v}}$$

In the optimization problem above, we are computing max of inner product of two normalized vectors Xu and Yv. So we can write it as a max of the inner product of these vectors subject to $\|Xu\| = 1$ and $\|Yv\| = 1$:

$$= \max_{u,v} u^T X^T Y v$$
$$s.t. \|Xu\| = 1, \|Yv\| = 1$$

This can be written as:

$$\max_{u,v} \begin{pmatrix} u^T & v^T \end{pmatrix} \begin{pmatrix} 0 & X^T Y \\ Y^T X & 0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}$$
$$s.t. \left\| \begin{pmatrix} X & 0 \\ 0 & Y \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} \right\| = \sqrt{2}$$

Because we are assuming that $\|Xu\| = \|Yv\|$ the condition of this optimization problem is the same as above.

3 Convex Functions for Gradient Descent

Convex functions have properties that make them very useful in machine learning and optimization. In this exercise, we will take a look at these properties.

Recall the definition of convexity: a function f is convex if and only if for any x, y and scalar $\alpha \in [0,1]$, the following inequality holds:

$$f(\alpha x + (1-\alpha)y) \le \alpha f(x) + (1-\alpha) f(y)$$

(a) Show that if two functions f and g are convex, then their sum $f + g$ is convex.

Solution: For any x, y and scalar $\alpha \in [0,1]$, the convexity of f and g implies

$$f(\alpha x + (1-\alpha)y) \le \alpha f(x) + (1-\alpha) f(y)$$
$$g(\alpha x + (1-\alpha)y) \le \alpha g(x) + (1-\alpha) g(y)$$

Let $h = f + g$, and taking sum of the two inequalities, we have

$$\begin{aligned} h(\alpha x + (1-\alpha)y) &= f(\alpha x + (1-\alpha)y) + g(\alpha x + (1-\alpha)y) \\ &\le \alpha f(x) + (1-\alpha) f(y) + \alpha g(x) + (1-\alpha) g(y) \\ &= \alpha h(x) + (1-\alpha) h(y). \end{aligned}$$

Thus, h is a convex function.

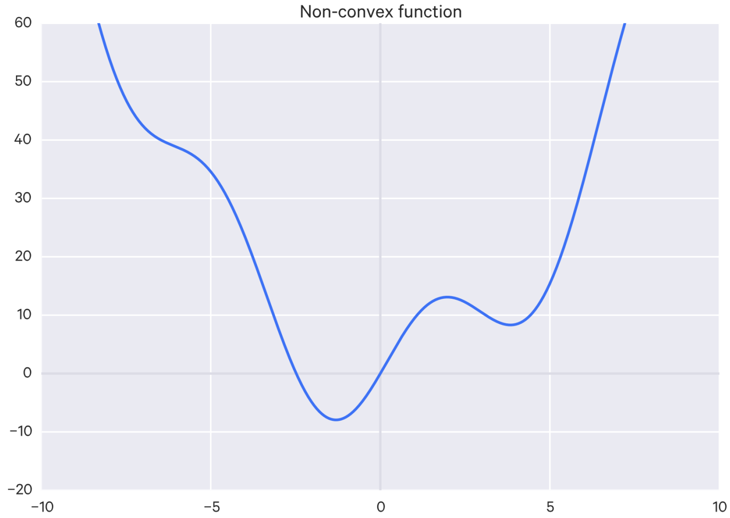

Figure 1: A nonconvex function

(b) Show that l_2 regularized least squares is an optimization problem over a convex function.

Solution: In l_2 regularized least squares we have a sum of two convex functions (why?), so the total error is a convex function.

(c) Consider the nonconvex function in the figure. How does the choice of initial value affect the result of the gradient descent? What do you suggest to improve that?

Solution: Based on the initial value, gradient descent may get stuck in a local minimum. One way to decrease the chance getting stuck in a local minimum is to choose initial values randomly and restart gradient descent multiple times in order to increase our chances of hitting the global minimum.

4 Extra: PCA Review

In this problem, we fix an $n \times n$ positive semi-definite matrix A. We will derive, from first principles, the best rank-1 approximation to A. Recall the following metric of approximation: for any integer $1 \leq r \leq n$, we define the best rank-r approximation as any minimizer A_r of

$$\min_{M \in R^{n \times n}} \|A - M\|_F : \mathsf{rank}(M) \leq r \tag{8}$$

Note that such a minimizer is not necessarily unique (why?). Since A is positive semi-definite, A_r must be as well: you make take this fact for granted. In this problem, we focus on the special case of $r = 1$. Also assume $A \neq 0$ to avoid any uninteresting, degenerate cases.

(a) Show that $A_1 = mm^T$, where m is any minimizer of the following *unconstrained* optimization problem

$$\min_{m \in R^n} \left\|A - mm^T\right\|_F . \tag{9}$$

Solution: Since A is positive semi-definite,

$$\min_{M\in R^{n\times n}}\{\|A-M\|_F : \text{rank}(M)\le 1\} = \min_{M\in R^{n\times n}}\{\|A-M\|_F : \text{rank}(M)\le 1, M\succeq 0\} \tag{10}$$

$$= \min_{m\in R^n}\left\{\|A-M\|_F : M = mm^T\right\} \tag{11}$$

$$= \min_{m\in R^n}\left\|A-mm^T\right\|_F. \tag{12}$$

(b) Define the function $f : R^n \longrightarrow R$ as $f(m) = \left\|A-mm^T\right\|_F^2$. Compute $\nabla f(m)$.

Solution: For two $n\times n$ matrices A,B, define $\langle A,B\rangle = \mathbf{Tr}(A^TB)$. Now,

$$f(m) = \left\|A-mm^T\right\|_F^2 = \|A\|_F^2 + \left\|mm^T\right\|_F^2 - 2\langle A, mm^T\rangle \tag{13}$$

$$= \|A\|_F^2 + \|m\|_2^4 - 2m^TAm. \tag{14}$$

Recall that $\nabla_m m^TAm = 2Am$. Furthermore, $\nabla_m \|m\|_2^4 = \nabla_m(\|m\|_2^2)^2 = 4mm^Tm$. Therefore,

$$\nabla_m f(m) = 4mm^Tm - 4Am = 4(mm^T - A)m. \tag{15}$$

(c) Set $\nabla f(m) = 0$ and conclude that all minimizers m of f must satisfy $m = \sqrt{\lambda}v$ where $\lambda \ge 0$ is an eigenvalue of A and v is the corresponding (unit normalized) eigenvector of A.

Solution: Setting $\nabla f(m) = 0$, all minimizers must satisfy the non-linear equation

$$Am = \|m\|_2^2 m. \tag{16}$$

First, we rule out the case that $m = 0$, which satisfies (16). Since $A \neq 0$ and PSD, it must have a non-zero diagonal entry, say A_{kk}. Hence, setting $m = \sqrt{A_{kk}}e_k$, we clearly have $m \neq 0$ and $f(m) < f(0)$.

Therefore, all optimizers of f are non-zero solutions to (16). This means that $m = \alpha v$ for some scalar $\alpha \neq 0$ and eigenvector v with eigenvalue λ of unit norm. Plugging in this parameterization of m into (16) we conclude that $\alpha\lambda v = \alpha^3 v$. Since $\alpha \neq 0$, we can solve for $\alpha = \sqrt{\lambda}$.

(d) Argue from part (c) that $A_1 = \lambda_1 v_1 v_1^T$, where λ_1 is the maximum eigenvalue of A and v_1 is a corresponding (unit normalized) eigenvector.

Solution: By part (b), we have at most n candidate solutions for the optimizer, so we simply check them all. Each candidate solution is $M_i = \lambda_i v_i v_i^T$, with $1 \le i \le \text{rank}(A)$. However

$$\|A-M_i\|_F^2 = \left\|\sum_{j=1}^r \lambda_j v_j v_j^T - \lambda_i v_i v_i^T\right\|_F^2 = \left\|\sum_{j\neq i}^r \lambda_j v_j v_j^T\right\|_F^2 = \sum_{j\neq i}^r \lambda_j^2 = \|A\|_F^2 - \lambda_i^2. \tag{17}$$

Hence, we have shown that

$$\|A-M_1\|_F^2 \le \|A-M_i\|_F^2, i = 1,2,\ldots,\text{rank}(A). \tag{18}$$