Sidhant Rastogi
Math189R SP19
Homework 5
Monday, Mar 11, 2019

Feel free to work with other students, but make sure you write up the homework and code on your own (no copying homework *or* code; no pair programming). Feel free to ask students or instructors for help debugging code or whatever else, though.

1 (Murphy 12.5 - Deriving the Residual Error for PCA) It may be helpful to reference section 12.2.2 of Murphy.

(a) Prove that

$$\left\| \mathbf{x}_i - \sum_{j=1}^{k} z_{ij} \mathbf{v}_j \right\|^2 = \mathbf{x}_i^\top \mathbf{x}_i - \sum_{j=1}^{k} \mathbf{v}_j^\top \mathbf{x}_i \mathbf{x}_i^\top \mathbf{v}_j.$$

Hint: first consider the case when $k = 2$. Use the fact that $\mathbf{v}_i^\top \mathbf{v}_j$ is 1 if $i = j$ and 0 otherwise. Recall that $z_{ij} = \mathbf{x}_i^\top \mathbf{v}_j$.

(b) Now show that

$$J_k = \frac{1}{n} \sum_{i=1}^{n} \left(\mathbf{x}_i^\top \mathbf{x}_i - \sum_{j=1}^{k} \mathbf{v}_j^\top \mathbf{x}_i \mathbf{x}_i^\top \mathbf{v}_j \right) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i^\top \mathbf{x}_i - \sum_{j=1}^{k} \lambda_j.$$

Hint: recall that $\mathbf{v}_j^\top \mathbf{\Sigma} \mathbf{v}_j = \lambda_j \mathbf{v}_j^\top \mathbf{v}_j = \lambda_j$.

(c) If $k = d$ there is no truncation, so $J_d = 0$. Use this to show that the error from only using $k < d$ terms is given by

$$J_k = \sum_{j=k+1}^{d} \lambda_j.$$

Hint: partition the sum $\sum_{j=1}^{d} \lambda_j$ into $\sum_{j=1}^{k} \lambda_j$ and $\sum_{j=k+1}^{d} \lambda_j$.

(a) We can rewrite the LHS as

$$\begin{aligned}
\left\| \mathbf{x}_i - \sum_{j=1}^{k} z_{ij}\mathbf{v}_j \right\|^2 &= (\mathbf{x}_i - \sum_{j=1}^{k} z_{ij}\mathbf{v}_j)^T(\mathbf{x}_i - \sum_{j=1}^{k} z_{ij}\mathbf{v}_j) \\
&= (\mathbf{x}_i^T - \sum_{j=1}^{k} z_{ij}\mathbf{v}_j^T)(\mathbf{x}_i - \sum_{j=1}^{k} z_{ij}\mathbf{v}_j) \\
&= \mathbf{x}_i^T\mathbf{x}_i - \sum_{j=1}^{k} z_{ij}\mathbf{x}_i^T\mathbf{v}_j - \sum_{j=1}^{k} \mathbf{v}_j^T\mathbf{x}_i z_{ij} + (\sum_{j=1}^{k} z_{ij}\mathbf{v}_j^T)(\sum_{j=1}^{k} z_{ij}\mathbf{v}_j) \\
&= \mathbf{x}_i^T\mathbf{x}_i - \sum_{j=1}^{k} \mathbf{x}_i^T\mathbf{v}_j\mathbf{x}_i^T\mathbf{v}_j - \sum_{j=1}^{k} \mathbf{v}_j^T\mathbf{x}_i\mathbf{x}_i^T\mathbf{v}_j + (\sum_{j=1}^{k} \mathbf{x}_i^T\mathbf{v}_j\mathbf{v}_j^T)(\sum_{j=1}^{k} \mathbf{x}_i^T\mathbf{v}_j\mathbf{v}_j) \\
&= \mathbf{x}_i^T\mathbf{x}_i - \sum_{j=1}^{k} \mathbf{x}_i^T\mathbf{v}_j\mathbf{x}_i^T\mathbf{v}_j - \sum_{j=1}^{k} \mathbf{v}_j^T\mathbf{x}_i\mathbf{x}_i^T\mathbf{v}_j + \sum_{j=1}^{k} \mathbf{x}_i^T\mathbf{v}_j\mathbf{x}_i^T\mathbf{v}_j \\
&= \mathbf{x}_i^T\mathbf{x}_i - \sum_{j=1}^{k} \mathbf{v}_j^T\mathbf{x}_i\mathbf{x}_i^T\mathbf{v}_j
\end{aligned}$$

which is precisely what we wished to show.

(b) First, note that the covariance matrix $\mathbf{\Sigma}$ can be written as

$$\mathbf{\Sigma} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i\mathbf{x}_i^T$$

Now, we can rewrite the LHS as

$$\begin{aligned}
J_k &= \frac{1}{n}\sum_{i=1}^{n}\left(\mathbf{x}_i^\top\mathbf{x}_i - \sum_{j=1}^{k}\mathbf{v}_j^\top\mathbf{x}_i\mathbf{x}_i^\top\mathbf{v}_j\right) \\
&= \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i^\top\mathbf{x}_i - \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{k}\mathbf{v}_j^\top\mathbf{x}_i\mathbf{x}_i^\top\mathbf{v}_j \\
&= \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i^\top\mathbf{x}_i - \sum_{i=1}^{k}\mathbf{v}_j^\top(\frac{1}{n}\sum_{j=1}^{n}\mathbf{x}_i\mathbf{x}_i^\top)\mathbf{v}_j \\
&= \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i^\top\mathbf{x}_i - \sum_{i=1}^{k}\mathbf{v}_j^\top\mathbf{\Sigma}\mathbf{v}_j \\
&= \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i^\top\mathbf{x}_i - \sum_{i=1}^{k}\lambda_j
\end{aligned}$$

(c) If $J_d = 0$, then, we must have that

$$\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i^\top\mathbf{x}_i = \sum_{i=1}^{d}\lambda_j = \sum_{i=1}^{k}\lambda_j + \sum_{i=k+1}^{d}\lambda_j \implies \sum_{i=1}^{k}\lambda_j = \sum_{i=1}^{d}\lambda_j - \sum_{i=k+1}^{d}\lambda_j$$

Then we get that

$$J_k = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i^\top\mathbf{x}_i - \sum_{i=1}^{k}\lambda_j = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i^\top\mathbf{x}_i - (\sum_{i=1}^{d}\lambda_j - \sum_{i=k+1}^{d}\lambda_j) = \sum_{i=k+1}^{d}\lambda_j$$

■

2 (ℓ_1-Regularization) Consider the ℓ_1 norm of a vector $\mathbf{x} \in \mathbb{R}^n$:

$$\|\mathbf{x}\|_1 = \sum_i |\mathbf{x}_i|.$$

Draw the norm-ball $B_k = \{\mathbf{x} : \|\mathbf{x}\|_1 \leq k\}$ for $k = 1$. On the same graph, draw the Euclidean norm-ball $A_k = \{\mathbf{x} : \|\mathbf{x}\|_2 \leq k\}$ for $k = 1$ behind the first plot. (Do not need to write any code, draw the graph by hand).

Show that the optimization problem

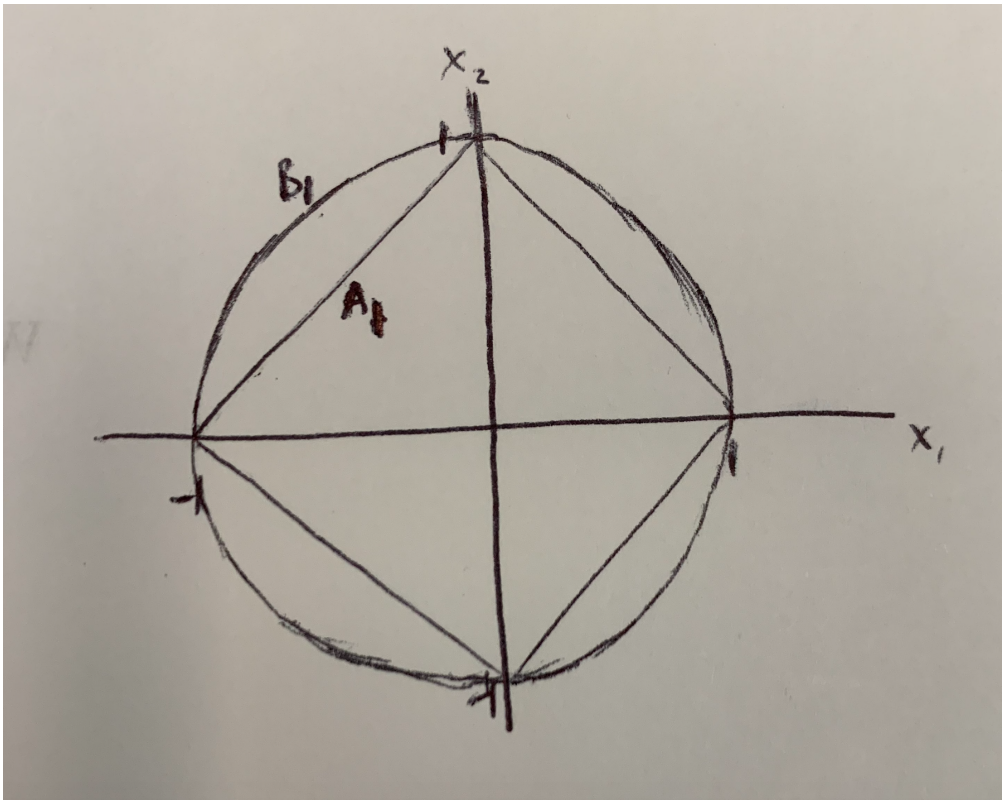
$$\begin{aligned} &\text{minimize: } f(\mathbf{x}) \\ &\quad \text{subj. to: } \|\mathbf{x}\|_p \leq k \end{aligned}$$

is equivalent to

$$\text{minimize: } f(\mathbf{x}) + \lambda\|\mathbf{x}\|_p$$

(hint: create the Lagrangian). With this knowledge, and the plots given above, argue why using ℓ_1 regularization (adding a $\lambda\|\mathbf{x}\|_1$ term to the objective) will give sparser solutions than using ℓ_2 regularization for suitably large λ.

The plot is shown below:



By the Lagrange multiplier method, we know that to minimize a function $f(\mathbf{x})$ subject to linear constraints $g_i(\mathbf{x}) \leq 0$, we can simply minimize $f(\mathbf{x}) + \sum_{i=1}^{n} \lambda_i g_i(\mathbf{x})$. Thus, the given optimization problem reduces to

$$\text{minimize: } f(\mathbf{x}) + \lambda(\|\mathbf{x}\|_p - k)$$

However, since $-k\lambda$ doesn't depend on $\mathbf{x}$, this is equivalent to

$$\text{minimize: } f(\mathbf{x}) + \lambda\|\mathbf{x}\|_p$$

From the plots above, we see that the ℓ_1 norm-ball has sharper corners than the ℓ_2 norm-ball. This means that there is a higher likelihood that a solution ends up at one of these corners (where one x-component is 0 and one is nonzero) than in the spherically symmetric ℓ_2 case. Thus, ℓ_1 regularization will lead to sparser solutions (more 0 components) than ℓ_2 regularization. ∎

Extra Credit (Lasso) Show that placing an equal zero-mean Laplace prior on each element of the weights $\boldsymbol{\theta}$ of a model is equivelent to ℓ_1 regularization in the Maximum-a-Posteriori estimate

$$\text{maximize: } \mathbb{P}(\boldsymbol{\theta}|\mathcal{D}) = \frac{\mathbb{P}(\mathcal{D}|\boldsymbol{\theta})\mathbb{P}(\boldsymbol{\theta})}{\mathbb{P}(\mathcal{D})}.$$

Note the form of the Laplace distribution is

$$\text{Lap}(x|\mu, b) = \frac{1}{2b}\exp\left(-\frac{|x-\mu|}{b}\right)$$

where μ is the location parameter and $b > 0$ controls the variance. Draw (by hand) and compare the density $\text{Lap}(x|0,1)$ and the standard normal $\mathcal{N}(x|0,1)$ and suggest why this would lead to sparser solutions than a Gaussian prior on each elements of the weights (which correspond to ℓ_2 regularization).

We begin by noting that

$$\text{maximize: } \mathbb{P}(\boldsymbol{\theta}|\mathcal{D}) = \text{maximize: } \mathbb{P}(\mathcal{D}|\boldsymbol{\theta})\mathbb{P}(\boldsymbol{\theta})$$

since $\mathbb{P}(\mathcal{D})$ does not depend on $\boldsymbol{\theta}$. Now, just like with the Gaussian prior, we can reformulate the problem as minimizing the negative log-likelihood of the function:

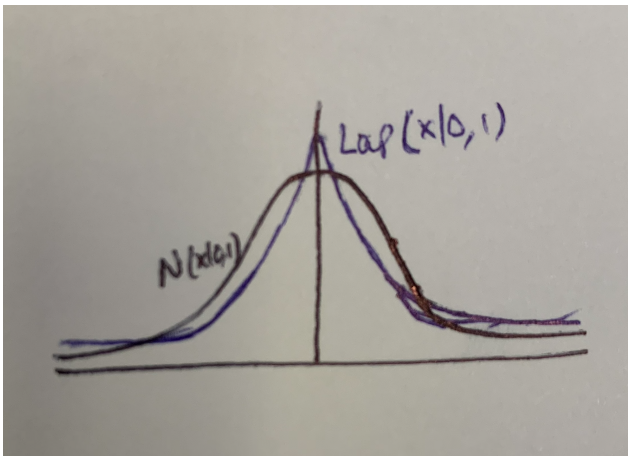
$$\min_{\boldsymbol{\theta}}[-\log\mathbb{P}(\mathcal{D}|\boldsymbol{\theta}) - \log\mathbb{P}(\boldsymbol{\theta})]$$

Now, we place a zero-mean Laplace prior on $\boldsymbol{\theta}$ to get that

$$\begin{aligned}
\min_{\boldsymbol{\theta}}[-\log\mathbb{P}(\mathcal{D}|\boldsymbol{\theta}) - \log\mathbb{P}(\boldsymbol{\theta})] &= \min_{\boldsymbol{\theta}}[-\log\mathbb{P}(\mathcal{D}|\boldsymbol{\theta}) - \log(\prod_{i=1}^{n}\frac{1}{2b}\exp\left(-\frac{|\theta_i|}{b}\right))] \\
&= \min_{\boldsymbol{\theta}}[-\log\mathbb{P}(\mathcal{D}|\boldsymbol{\theta}) - \log\frac{1}{(2b)^n} - (\sum_{i=1}^{n} -\frac{|\theta_i|}{b})] \\
&= \min_{\boldsymbol{\theta}}[-\log\mathbb{P}(\mathcal{D}|\boldsymbol{\theta}) + \frac{1}{b}\sum_{i=1}^{n}|\theta_i|] \\
&= \min_{\boldsymbol{\theta}}[-\log\mathbb{P}(\mathcal{D}|\boldsymbol{\theta}) + \lambda\|\mathbf{x}\|_1]
\end{aligned}$$

which is simply Lasso regularization with $\lambda = \frac{1}{b}$.

The densities of the two distributions are shown below:



We can see that the Laplacian prior would lead to sparser solutions than the Gaussian prior because the function $\text{Lap}(x|0,1) = \frac{1}{2}\exp(-|x|)$ has a sharp maximum at $x = 0$. This implies that there is a greater probability of getting solutions with components of $\boldsymbol{\theta}$ being 0 in the Laplacian case as opposed to the smooth Gaussian case. ■