Eli Weissler

Math189R SU17
Homework 5
Monday, June 5, 2017

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 $\mathbf{v}_j^\top \boldsymbol{\Sigma} \mathbf{v}_j = \lambda_j \mathbf{v}_j^\top \mathbf{v}_j = \lambda_j$.

(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) First note that

$v_j \in \mathbb{R}^D$ = jth principal direction
$x_i \in \mathbb{R}^D$ = ith high-d representation
$z_i \in \mathbb{R}^K$ = ith low-d representation

Therefore $z_{ij} = x_i^T v_j$ i.e. the projection of x_i onto v_j

$= v_j^T x_i$ ∎

Expand it out

$$\|x_i - \sum z_{ij} v_j\|^2 = x_i^T x_i - x_i^T \sum z_{ij} v_j - \left(\sum z_{ij} v_j^T\right) x_i + \sum z_{ij} v_j^T \sum z_{ij} v_j$$

$$= x_i^T x_i - 2 \sum z_{ij} v_j^T x_i + \sum_j z_{ij} v_j^T \sum_k z_{ik} v_k$$

b/c $v_k^T v_j = \delta_{jk}$

$$\rightarrow \sum z_{ij} v_j^T v_j z_{ij} = \sum z_{ij} z_{ij}$$

$$= x_i^T x_i - 2 \sum v_j^T x_i x_i^T v_j + \sum v_j^T x_i x_i^T v_j$$

$$= \boxed{x_i^T x_i - \sum_j v_j^T x_i x_i^T v_j}$$

norm of high d rep

norm of lower-d representation in high d

b) Recall $\Sigma = \frac{1}{N} \sum_{i=1}^{N} x_i x_i^T$ is the empirical covariance matrix

$$J_k = \frac{1}{n} \sum_{i=1}^{n} \left(x_i^T x_i - \sum_j v_j^T x_i x_i^T v_j\right) = \frac{1}{n} \sum_i \left(x_i^T x_i\right) - \sum_j v_j^T \Sigma v_j$$

v_j is eigenvec of Σ w/ $\lambda = \lambda_i$

$\rightarrow v_j^T \Sigma v_j = \lambda_j v_j^T v_j = \lambda_j$

$$\rightarrow J_k = \boxed{\frac{1}{n} \sum_{i=1}^{n} x_i^T x_i - \sum_{j=1}^{k} \lambda_j}$$

c) $J_{k=d} = \frac{1}{n} \sum x_i^T x_i - \sum_{j=1}^{d} \lambda_j = 0$ because we include all dimensions

J_k can thus be written as

$$J_k - J_d = \sum_{j=1}^{k} \lambda_j - \sum_{j=1}^{d} \lambda_j = \boxed{\sum_{j=k+1}^{d} \lambda_j}$$

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

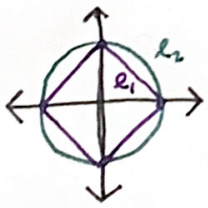
minimize: $f(\mathbf{x})$
subj. to: $\|\mathbf{x}\|_p \leq k$

is equivalent to

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 norm ball is clearest in 2D



We can represent the minimization problem using Lagrange multipliers. We must find the critical points of

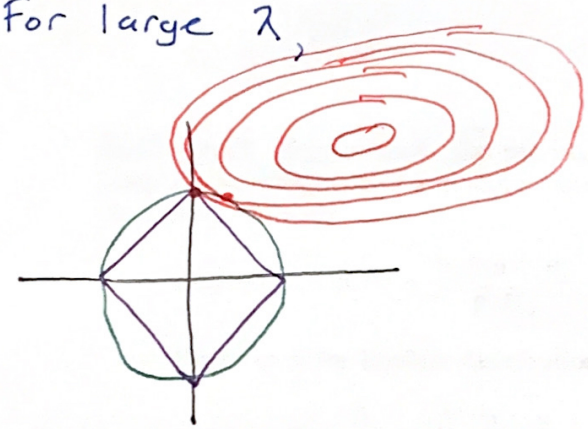
$$L(\vec{x}, \lambda) = f(\vec{x}) + \lambda(\|\vec{x}\|_p - k)$$

which is equivalent to the critical points/minimizing

$$f(\vec{x}) + \lambda\|\vec{x}\|_p$$

as $-\nabla_x \lambda k = 0$ ∎

For large λ,



← level sets of convex cost function

We are looking for the intersection of convex cost function and the norm ball. A level set will be far more likely to intersect the ℓ_1 ball near an axis because that is where it is uniquely furthest from the origin, unlike the rotationally symmetric ℓ_2 norm ball.

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

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

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).

This is equivalent to maximizing $\log P(\theta|D)$ w.r.t. θ

$$\log P(D|\theta) + \log P(\theta) - \log P(D)$$

which is equivalent to minimizing

$$-\log P(D|\theta) - \log P(\theta)$$

as $\log P(D)$ does not depend on θ. Now insert our prior

$$-\log P(D|\theta) - \log b \exp(-|\theta|)$$

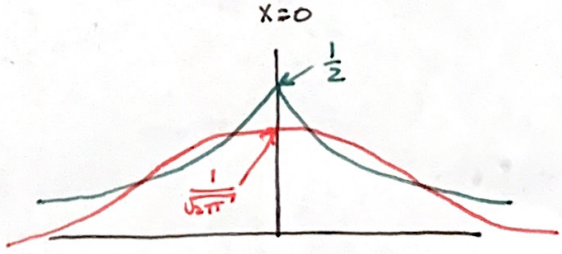
$$= -\log P(D|\theta) - \log b + b|\theta|$$

which is equivalent to minimizing

$$-\log P(D|\theta) + b|\theta|$$

↑ prob of data given θ

which is ℓ_1 normalization!



Laplace has more mass near 0, so we'd expect more sparse solutions, i.e. more θ near 0