CS 231A Section 2
Math background for PS1

Jiahui Shi

Announcement

- Updated deadline of PS1:

  10/14, 12 noon

- Extra office hour for PS1:

  10/13, 7-9pm, Gates 104

Overview

- Maximizing and Minimizing quadratic forms
- Least squares
- Linear filtering
- Maximum likelihood estimation

Quadratic Forms

- Quadratic forms look like (A is symmetric),

$$x^T A x$$

or if we have a 2x2 matrix

$$\begin{bmatrix} x_1 x_2 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$x_1^2 + 4x_1x_2 + x_2^2$$

Quadratic Forms in Optimization

- We will look at the optimization problem for a symmetric matrix, this will be useful for PS1

$$\begin{aligned} &\text{min. } x^T A x \\ &\text{s.t. } \|x\|_2^2 = 1 \end{aligned}$$

- This can be solved using the eigenvector corresponding to the smallest eigenvalue.
- Similarly, the eigenvector of the largest eigenvalue will give the max value.

Optimizing Quadratic Forms

- Formulate the Lagrangian

$$\mathcal{L}(x,\lambda) = x^T A x + \lambda(1 - x^T x)$$

- Now we take the partial with respect to ***x***

$$\begin{aligned}\nabla_x \mathcal{L}(x,\lambda) &= \nabla_x (x^T A x + \lambda(1 - x^T x)) \\ &= 2Ax - 2\lambda x = 0\end{aligned}$$

Optimizing Quadratic Forms

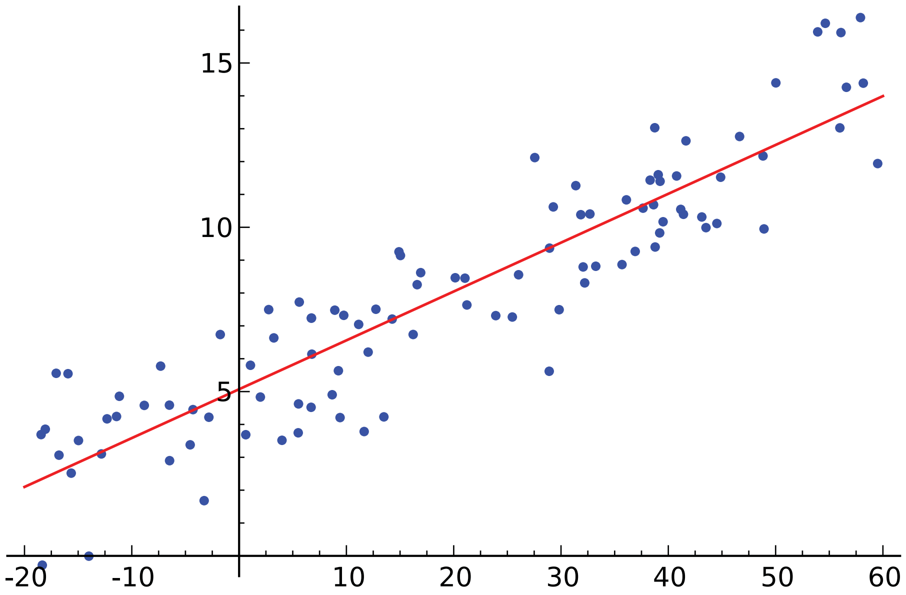
- Using this condition we know that our optimal ***x***'s must satisfy

$$Ax = \lambda x$$

- Our possible optimizing vectors are just the eigenvectors of A
- Plugging in: $x^T A x = \lambda x^T x = \lambda$
- The constraint on the matrix is pretty stringent. The matrix must be **square**, and **symmetric**.

Linear Regression

- Fit a linear model to data
- PS1 - use least squares approach to fit a model to our data with minimum total error.



Least Squares

- Given $\boldsymbol{A}$ and $\boldsymbol{y}$, find optimal $\boldsymbol{x}$, s.t. $\boldsymbol{Ax} = \boldsymbol{y}$.
- Residual is defined as

$$r = Ax - y$$

Assume $\boldsymbol{A}$ is skinny (#rows> #columns, or more equations than unknowns) and full rank.

- We want to minimize the squared 2-norm of residual:

$$\|r\|_2^2 = x^T A^T A x - 2y^T A x + y^T y$$

Least Squares

- To minimize

$$\|r\|_2^2 = x^T A^T A x - 2y^T A x + y^T y$$

take the gradient with respect to ***x*** and set equal to zero

$$2A^T A x - 2A^T y = 0$$

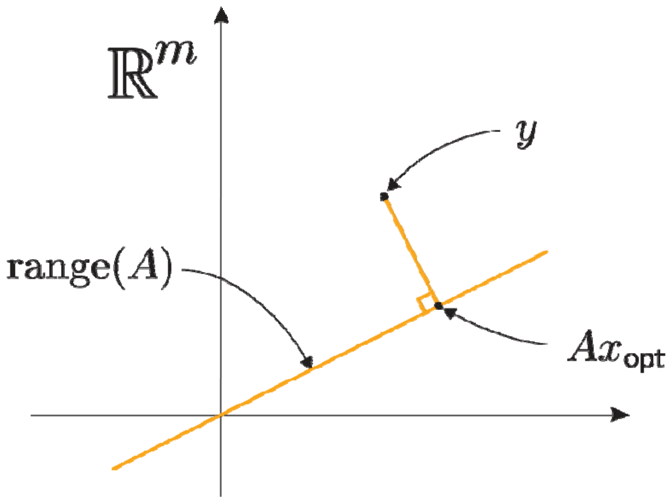
solving

$$x_{\text{opt}} = (A^T A)^{-1} A^T y$$

Geometric Approach

- Pick x_{opt} by

$$A x_{\text{opt}} - y \perp \text{range}(A)$$
$$\Leftrightarrow A x_{\text{opt}} - y \in \text{null}(A^T)$$
$$\Leftrightarrow A^T(A x_{\text{opt}} - y) = 0$$



thus we have

$$A^T A x_{\text{opt}} = A^T y$$

since A is skinny (#rows > #columns) and full rank we can write

$$x_{\text{opt}} = (A^T A)^{-1} A^T y$$

Least Squares Solution

- When there is an optimization problem of the form
$$\min.\ \|Ax - y\|_2^2$$
we can immediately write down the solution,
$$x_{\text{opt}} = (A^T A)^{-1} A^T y$$
- Trick is to get it in that form, to better illustrate this lets see an example

Example Problem

$$\min. \ \|Ax - y\|_2^2 + \|Fx - g\|_2^2$$

Example Problem

$$\min. \ \|Ax - y\|_2^2 + \|Fx - g\|_2^2$$

- Solution:

$$x_{opt} = (B^T B)^{-1} B^T h$$

$$B = \begin{bmatrix} A \\ F \end{bmatrix}, h = \begin{bmatrix} y \\ g \end{bmatrix}$$

Linear Filtering

- Remember that convolution is linear
- Convolution properties: commutative, associative, distributive, shift-invariance
- As a result:

$$F * (k_1 I_1 + k_2 I_2) = k_1 F * I_1 + k_2 F * I_2$$

Linear Filtering

- Trig identities are very useful for the problem set and analyzing linear filters in general
  - Useful for the problem set:

$$\cos(\theta \pm \phi) = \cos\theta\cos\phi \mp \sin\theta\sin\phi$$

$$\cos\phi = \cos\left(\arctan\left(\frac{y}{x}\right)\right) = x$$

$$\sin\phi = \sin\left(\arctan\left(\frac{y}{x}\right)\right) = y$$

Likelihood function

- You have a model which is parameterized by θ which characterizes the probability of each of the n outputs $y^{(i)}$
- This model is also function of the n observed input data $x^{(i)}$
- The model is given by $p(y^{(i)} \mid x^{(i)}; \theta)$, which reads: the probability of $y^{(i)}$ given $x^{(i)}$ and parameterized by θ
- Assuming all our data is i.i.d. we can write the probability of all our data as $\prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)}; \theta)$

Likelihood function

- This distribution describes the probability of the output data given our input data and the model θ
- We want to find the model θ, to make predictions in the future
- To find our model we use a likelihood function
- The likelihood is defined as

$$L(\theta) = \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)}; \theta)$$

which is a function of θ

Maximum likelihood estimation

- Once we have $L(\theta)$ we would like to maximize it over θ
- This is known as maximum likelihood (ML) estimation
- To make maximization easier and still solving an equivalent optimization problem we will maximize $\ln L(\theta)$, instead of trying to directly maximize $L(\theta)$

Maximum a posteriori (MAP)

- Very similar to maximum likelihood, but now we view the our model θ as a random variable
- This allows us to put some prior distribution over what instances θ can take on
- With this prior we can write our the probability of our data as $$\prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)}; \theta) p(\theta)$$
- We can maximize the same way as maximum likelihood, in fact maximum likelihood is a MAP problem with a uniform prior

Solving MAP and ML estimate

- Write down

$$L(\theta) = \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)}; \theta)$$

- Now take the log

$$\begin{aligned} \ln L(\theta) = \ell(\theta) &= \ln \prod_{i=1}^{n} p(y^{(i)} \mid x^{(i)}; \theta) \\ &= \sum_{i=1}^{n} \ln \left(p(y^{(i)} \mid x^{(i)}; \theta) \right) \end{aligned}$$

- Now maximize by taking the gradient and setting equal to zero

$$\nabla_\theta \sum_{i=1}^{n} \ln \left(p(y^{(i)} \mid x^{(i)}; \theta) \right) = 0$$