ECE 5984: Introduction to Machine Learning

Topics:
   – Regression

Readings: Barber 17.1, 17.2

Dhruv Batra
Virginia Tech
Administrativia

• HW1
  – Due on Sun 02/15, 11:55pm

• Project Proposal
  – Due: Tue 02/24, 11:55 pm
  – <=2pages, NIPS format

• HW2
  – Out today
  – Due on Friday 03/06, 11:55pm
  – Please please please please please please start early
  – Implement linear regression, Naïve Bayes, Logistic Regression
Recap of last time
Learning a Gaussian

• Collect a bunch of data
  – Hopefully, i.i.d. samples
  – e.g., exam scores

• Learn parameters
  – Mean
  – Variance

\[ P(x \mid \mu, \sigma) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \]
MLE for Gaussian

- Prob. of i.i.d. samples $D=\{x_1,\ldots,x_N\}$:

$$P(D \mid \mu, \sigma) = \left(\frac{1}{\sigma \sqrt{2\pi}}\right)^N \prod_{i=1}^{N} e^{\frac{-(x_i-\mu)^2}{2\sigma^2}}$$

- Log-likelihood of data:

$$\ln P(D \mid \mu, \sigma) = \ln \left[ \left(\frac{1}{\sigma \sqrt{2\pi}}\right)^N \prod_{i=1}^{N} e^{\frac{-(x_i-\mu)^2}{2\sigma^2}} \right]$$

$$= -N \ln \sigma \sqrt{2\pi} - \sum_{i=1}^{N} \frac{(x_i - \mu)^2}{2\sigma^2}$$
Your second learning algorithm:

**MLE for mean of a Gaussian**

- What’s MLE for mean?

\[
\frac{d}{d\mu} \ln P(D | \mu, \sigma) = \frac{d}{d\mu} \left[ -N \ln \sigma \sqrt{2\pi} - \sum_{i=1}^{N} \frac{(x_i - \mu)^2}{2\sigma^2} \right]
\]
Learning Gaussian parameters

- MLE:

\[ \hat{\mu}_{MLE} = \frac{1}{N} \sum_{i=1}^{N} x_i \]

\[ \hat{\sigma}^2_{MLE} = \frac{1}{N} \sum_{i=1}^{N} (x_i - \hat{\mu})^2 \]
Bayesian learning of Gaussian parameters

• Conjugate priors
  – Mean: Gaussian prior
  – Variance: Inverse Gamma or Wishart Distribution

• Prior for mean:

\[ P(\mu \mid \eta, \lambda) = \frac{1}{\lambda \sqrt{2\pi}} e^{-\frac{(\mu - \eta)^2}{2\lambda^2}} \]
MAP for mean of Gaussian

\[
P(\mu \mid \eta, \lambda) = \frac{1}{\lambda \sqrt{2\pi}} e^{-\frac{(\mu - \eta)^2}{2\lambda^2}}
\]

\[
P(\mathcal{D} \mid \mu, \sigma) = \left( \frac{1}{\sigma \sqrt{2\pi}} \right)^N \prod_{i=1}^{N} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}}
\]

\[
\frac{d}{d\mu} \left[ \ln P(\mathcal{D} \mid \mu) P(\mu) \right] = \frac{d}{d\mu} \left[ \ln P(\mathcal{D} \mid \mu) + \ln P(\mu) \right]
\]
Plan for Today

• Regression
  – Linear Regression
  – Connections with Gaussians
New Topic: Regression
1-NN for Regression

- Often bumpy (overfits)
Linear fitting to data

- We want to fit a linear function to an observed set of points $X = [x_1, \ldots, x_N]$ with associated labels $Y = [y_1, \ldots, y_N]$.
  
- Once we fit the function, we want to use it to predict the $y$ for new $x$.  

![Graph showing linear fitting to data]
Linear fitting to data

- We want to fit a linear function to an observed set of points $X = [x_1, \ldots, x_N]$ with associated labels $Y = [y_1, \ldots, y_N]$.

- Once we fit the function, we want to use it to predict the $y$ for new $x$.

- Least squares (LSQ) fitting criterion: find the function that minimizes sum (or average) of square distances between actual $y$s in the training set and predicted ones.
Linear fitting to data

- We want to fit a linear function to an observed set of points \( X = [x_1, \ldots, x_N] \) with associated labels \( Y = [y_1, \ldots, y_N] \).

- Once we fit the function, we want to use it to \textit{predict} the \( y \) for new \( x \).

- Least squares (LSQ) fitting criterion: find the function that minimizes sum (or average) of square distances between actual \( y \)s in the training set and predicted ones.

The fitted line is used as a predictor
Linear Regression

- Demo
  - http://hspm.sph.sc.edu/courses/J716/demos/LeastSquares/LeastSquaresDemo.html
Linear functions

- General form: $f(x; w) = w_0 + w_1x_1 + \ldots + w_dx_d$
- 1D case ($\mathcal{X} = \mathbb{R}$): a line
- $\mathcal{X} = \mathbb{R}^2$: a plane
- *Hyperplane* in general, $d$-D case.
Least squares: estimation

- We need to minimize w.r.t. \( w \)

\[
L(w, X) = L(w) = \frac{1}{N} \sum_{i=1}^{N} (y_i - w^T x_i)^2
\]

\[
= \frac{1}{N} \sum_{i=1}^{N} (y_i - w_0 - w_1 x_{i1} - \ldots - w_d x_{id})^2
\]

- Necessary condition to minimize \( L \): derivatives w.r.t. \( w_0, w_1, \ldots, w_d \) must be zero.
Least squares in matrix form

\[
X = \begin{bmatrix}
1 & x_{11} & \cdots & x_{1d} \\
\vdots & \vdots & \ddots & \vdots \\
1 & x_{N1} & \cdots & x_{Nd}
\end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}, \quad w = \begin{bmatrix} w_0 \\ \vdots \\ w_d \end{bmatrix}.
\]

- Predictions: \( \hat{y} = Xw \), errors: \( y - Xw \), empirical loss:

\[
L(w, X) = \frac{1}{N} (y - Xw)^T (y - Xw)
\]
Derivative of loss

\[ L(w) = \frac{1}{N} \left( y^T - w^T X^T \right) (y - Xw). \]

\[ \frac{\partial a^T b}{\partial a} = \frac{\partial b^T a}{\partial a} = b, \quad \frac{\partial a^T b a}{\partial a} = 2b a \]
Derivative of loss

\[ L(w) = \frac{1}{N} \left( y^T - w^T X^T \right) (y - Xw). \]

\[ \frac{\partial a^T b}{\partial a} = \frac{\partial b^T a}{\partial a} = b, \quad \frac{\partial a^T B a}{\partial a} = 2B a \]

\[ \frac{\partial L(w)}{\partial w} = \frac{1}{N} \frac{\partial}{\partial w} \left[ y^T y - w^T X^T y - y^T X w + w^T X^T X w \right] \]

\[ = \frac{1}{N} \left[ 0 - X^T y - (y^T X)^T + 2X^T X w \right] \]

\[ = -\frac{2}{N} \left( X^T y - X^T X w \right) \]
Least squares solution

\[
\frac{\partial}{\partial w} L(w) = -\frac{2}{N} (X^T y - X^T X w) = 0
\]
Least squares solution

\[
\frac{\partial}{\partial \mathbf{w}} L(\mathbf{w}) = -\frac{2}{N} (\mathbf{X}^T \mathbf{y} - \mathbf{X}^T \mathbf{X} \mathbf{w}) = 0
\]

\[
\mathbf{X}^T \mathbf{y} = \mathbf{X}^T \mathbf{X} \mathbf{w} \Rightarrow \mathbf{w}^* = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}
\]

- \( \mathbf{X}^\dagger \triangleq (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \) is called the Moore-Penrose pseudoinverse of \( \mathbf{X} \).

- Linear regression in Matlab:
  
  ```matlab
  \% \( \mathbf{X}(i,:) \) is \( i \)-th example, \( \mathbf{y}(i) \) is \( i \)-th label
  \% \( \mathbf{w}_{LSQ} = \text{pinv}([\text{ones(size}(\mathbf{X},1),1) \mathbf{X}])\mathbf{y}; \)
  
  Prediction:
  
  \[
  \hat{\mathbf{y}} = \mathbf{w}^*^T \begin{bmatrix} 1 \\ \mathbf{x}_0 \end{bmatrix} = \mathbf{y}^T \mathbf{X}^\dagger^T \begin{bmatrix} 1 \\ \mathbf{x}_0 \end{bmatrix}
  \]
  ```
But, why?

• Why sum squared error???
• Gaussians, Watson, Gaussians…
Gaussian noise model

\[ y = f(x; w) + \nu, \quad \nu \sim \mathcal{N}(\nu; 0, \sigma^2) \]

- Given the input \( x \), the label \( y \) is a random variable

\[ p(y|x; w, \sigma) = \mathcal{N}(y; f(x; w), \sigma^2) \]

that is,

\[ p(y|x; w, \sigma) = \frac{1}{\sigma \sqrt{2\pi}} \exp \left( -\frac{(y - f(x; w))^2}{2\sigma^2} \right) \]

- This is an explicit model of \( y \) that allows us, for instance, to sample \( y \) for a given \( x \).
MLE Under Gaussian Model

- On board
Is OLS Robust?

• Demo
  – http://www.calpoly.edu/~srein/StatDemo/All.html

• Bad things happen when the data does not come from your model!

• How do we fix this?
Robust Linear Regression

- $y \sim \text{Lap}(w'x, b)$

- On board