Linear Models for Regression

CSci 5525: Machine Learning

Instructor: Nicholas Johnson

September 10, 2020

Announcements

- HW0 posted last Tue (due Tue Sept. 15)
- Office hours updated

Problem

Suppose you work at a restaurant and want to predict how much the customers tip. You are given the following data consisting of the total bill amount and the tip added.

Total Bill	Tip
16.99	1.01
10.34	1.66
21.01	3.50
23.68	3.31
24.59	3.61
25.29	4.71
8.77	2.00

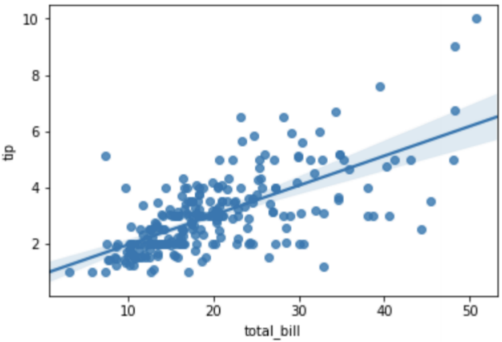
One heuristic is to predict the average tip: $2.83.
Can we do better?

Regression

- Dataset: $\mathcal{D} = \{(\text{total bill}_i, \text{tip}_i)\}_{i=1}^n = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$
- Features $\mathbf{x}_i$ and targets y_i
- Supervised learning problem
    - $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}$, $y \in \mathcal{Y} \subset \mathbb{R}$ (regression)
- Goal: find prediction function $f : \mathcal{X} \to \mathcal{Y}$

Linear Functions

- Choose hypothesis (prediction function) class $\mathcal{C}$ to be linear functions
- $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$ (we often write $\mathbf{x} = [\mathbf{x}; 1]$)
- Many functions to choose from:



Which to choose?

Empirical Risk Minimization (ERM)

- Most supervised learning follows ERM
- ERM recipe:
  - Pick class of predictors $\mathcal{C}$ (linear in this lecture)
  - Pick loss function $\ell(\cdot)$
  - Minimize empirical risk over model/parameters

Loss Functions

- Learning is often based on *minimizing expected loss*
- 0/1 Loss: $L(f, \mathbf{x}, y) = \mathbb{1}_{[f(\mathbf{x}) \neq y]}$, expected loss

$$\mathbb{E}[L(f, \mathbf{x}, y)] = \mathbb{E}[\mathbb{1}_{[f(\mathbf{x}) \neq y]}] = P(f(\mathbf{x}) \neq y)$$

- Hinge Loss:

$$L(f, \mathbf{x}, y) = \max(0, 1 - yf(\mathbf{x})) = \begin{cases} 1 - yf(\mathbf{x}) & \text{if } yf(\mathbf{x}) < 1, \\ 0 & \text{otherwise.} \end{cases}$$

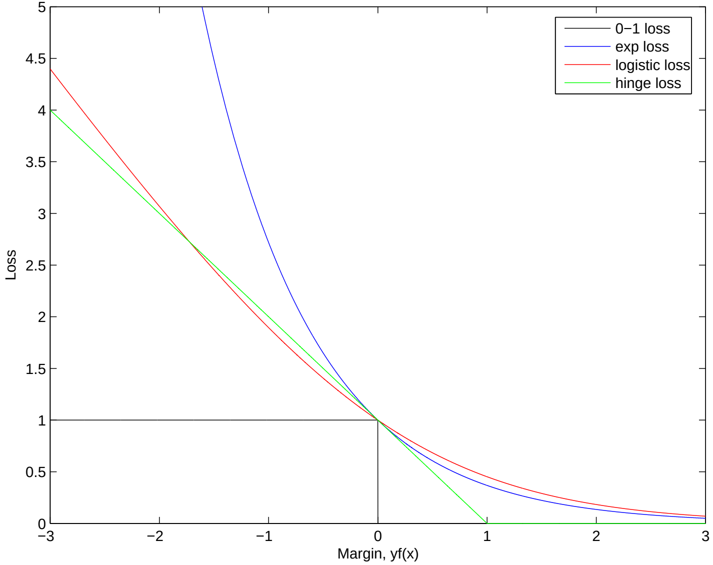
- Exponential Loss:

$$L(f, \mathbf{x}, y) = \exp(-yf(\mathbf{x}))$$

- Logistic Loss:

$$L(f, \mathbf{x}, y) = \log(1 + \exp(-yf(\mathbf{x})))$$

Loss Functions



Estimation and Approximation Error

- In practice, one chooses f_n^* from $\mathcal{C}$ given n training samples
- Clearly, $L(f_n^*) > L(f^*)$
- An important decomposition

$$L(f_n^*) - L(f^*) = \left(L(f_n^*) - \inf_{f \in \mathcal{C}} L(f) \right) + \left(\inf_{f \in \mathcal{C}} L(f) - L(f^*) \right) .$$

- First term is the *estimation error* (ee)
- Second term is the *approximation error* (ae)
- Choice of "bias" trades-off the two terms:
  - High "bias" $\Rightarrow$ low ee, high ae
  - Low "bias" $\Rightarrow$ high ee, low ae

Linear Regression

- Loss function: least square loss for prediction $\hat{y} = \hat{f}(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$

$$\ell(y, \hat{y}) = (y - \hat{y})^2$$

- Goal: minimize least squares empirical risk:

$$\mathcal{R}(\hat{f}) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, \hat{f}(\mathbf{x}_i)) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(\mathbf{x}_i))^2$$

- For linear functions, find $\mathbf{w} \in \mathbb{R}^d$ (our example $d = 2$) such that

$$\text{argmin}_{\mathbf{w}\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^{n}(y_i - \mathbf{w}^\top \mathbf{x}_i)^2$$

Least Squares Solution

- Design matrix:

$$\mathbf{X} = \begin{bmatrix} \leftarrow & \mathbf{x}_1^\top & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{x}_n^\top & \rightarrow \end{bmatrix}$$

- Response vector:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

- Empirical risk can be written as

$$\mathcal{R}(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \mathbf{w}^\top \mathbf{x}_i)^2 = \frac{1}{n}\|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2$$

Least Squares Solution

- Rescaling does not change solution, so least squares solution is given by:
$$\mathbf{w}^* \in \text{argmin}_{\mathbf{w}\in\mathbb{R}^d} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2$$
- Necessary condition for $\mathbf{w}$ to be minimizer of $\hat{\mathcal{R}}$ is that it needs to be a stationary point: $\nabla\hat{\mathcal{R}}(\mathbf{w}) = 0$
- This gives the condition: $(\mathbf{X}^\top\mathbf{X})\mathbf{w} = \mathbf{X}^\top\mathbf{y}$
- If $\mathbf{X}$ is full-rank then we can invert so: $\mathbf{w}^* = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y}$
- Otherwise, use pseudoinverse

A Statistical View

- We often study linear regression under the following model:

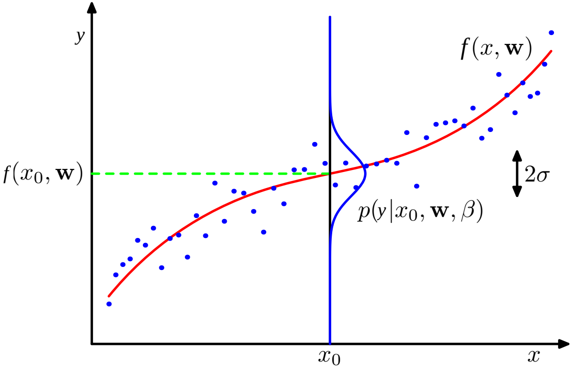
$$y_i = \mathbf{w}^\top \mathbf{x}_i + \epsilon_i \qquad \text{where } \epsilon \sim N(0, \sigma^2)$$

- In other words, the distribution of y_i given $\mathbf{x}_i$ is:

$$y_i | \mathbf{x}_i \sim N(\mathbf{w}^\top \mathbf{x}_i, \sigma^2)$$

$$\Rightarrow P(y_i | \mathbf{x}_i, \mathbf{w}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ -\frac{(\mathbf{w}^\top \mathbf{x}_i - y_i)^2}{2\sigma^2} \right\}$$

Conditional Distribution



A Statistical View

- Consider maximum likelihood estimation (MLE) that aims to maximize:

$$P(\text{observed data}|\text{model parameters})$$

MLE

$$
\begin{aligned}
\mathbf{w} &= \operatorname{argmax}_{\mathbf{w}} P(y_1, \mathbf{x}_1, \ldots, y_n, \mathbf{x}_n | \mathbf{w}) \\
&= \operatorname{argmax}_{\mathbf{w}} \prod_{i=1}^{n} P(y_i, \mathbf{x}_i | \mathbf{w}) && \text{(Independence)} \\
&= \operatorname{argmax}_{\mathbf{w}} \prod_{i=1}^{n} P(y_i | \mathbf{x}_i, \mathbf{w}) P(\mathbf{x}_i | \mathbf{w}) && \text{(Chain rule)} \\
&= \operatorname{argmax}_{\mathbf{w}} \prod_{i=1}^{n} P(y_i | \mathbf{x}_i, \mathbf{w}) P(\mathbf{x}_i) && (\mathbf{x}_i \text{ independent of } \mathbf{w}) \\
&= \operatorname{argmax}_{\mathbf{w}} \prod_{i=1}^{n} P(y_i | \mathbf{x}_i, \mathbf{w}) && (P(\mathbf{x}_i) \text{ does not depend on } \mathbf{w}) \\
&= \operatorname{argmax}_{\mathbf{w}} \sum_{i=1}^{n} \log P(y_i | \mathbf{x}_i, \mathbf{w}) && \text{(log is a monotonic function)}
\end{aligned}
$$

MLE (cont.)

$$= \operatorname{argmax}_{\mathbf{w}} \sum_{i=1}^{n} \log P(y_i|\mathbf{x}_i, \mathbf{w}) \qquad \text{(log is a monotonic function)}$$

$$= \operatorname{argmax}_{\mathbf{w}} \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi\sigma^2}} + \log \exp\left\{-\frac{(\mathbf{w}^\top \mathbf{x}_i - y_i)^2}{2\sigma^2}\right\}$$

(Plugging in Gaussian distribution)

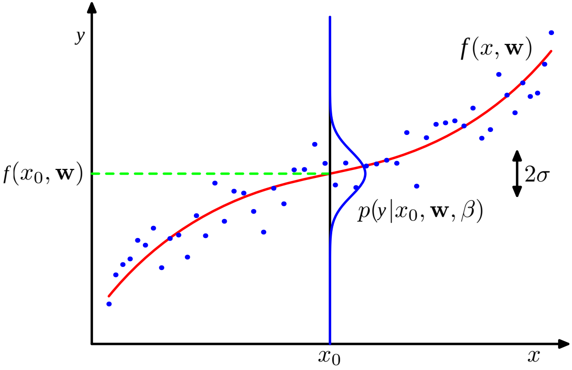
$$= \operatorname{argmax}_{\mathbf{w}} -\frac{1}{2\sigma^2} \sum_{i=1}^{n} (\mathbf{w}^\top \mathbf{x}_i - y_i)^2$$

(First term is a constant and $\log(\exp(z)) = z$)

$$= \operatorname{argmin}_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} (\mathbf{w}^\top \mathbf{x}_i - y_i)^2$$

(Equivalent to minimizing least squares risk)

Conditional Distribution

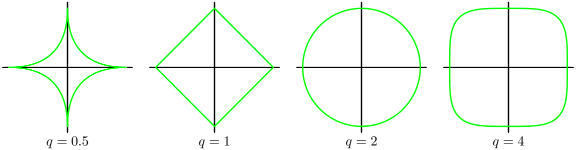


Regularized least squares

- Regularization to control over-fitting

$$E_D(\mathbf{w}) + \lambda E_W(\mathbf{w})$$

$$\frac{1}{2}\sum_{n=1}^{N}(y_n - \mathbf{w}^T\phi(\mathbf{x}_n))^2 + \frac{\lambda}{2}\sum_{j=1}^{M}|w_j|^q$$



General classes of regularizers:

$$\frac{1}{2}\sum_{n=1}^{N}(y_n - \mathbf{w}^T\phi(\mathbf{x}_n))^2 + \lambda\|\mathbf{w}\|_{\mathcal{H}}$$