Notes: Lab 7

March 4, 2024

1 Overview

- In the lab, we discussed a broad approach that we could use (mainly in the context of linear models), in order to train a machine learning algorithm from scratch
  - Pick a model which gives us a representation of the relationship between the features and the label
  - Choose an appropriate loss function
  - Optional: add regularization
  - Fit the model (minimize the loss function, using your favorite optimization algorithm)
- Here, we review these steps, in the context of parametric linear models.

2 Notation

- x $\in \mathcal{X}$ is a vector of inputs (or covariates, or features).
- y $\in \mathcal{Y}$ is the label (also target, response, etc).
- Given data $\mathcal{D} = \{(x^{(i)}, y^{(i)}) | i = 1..N\}$, we want to find some function $f : \mathcal{X} \to \mathcal{Y}$ which maps our inputs to the label.

3 Linear Regression

- We assume that f is linear; i.e, we use a linear function of inputs $(x_1, x_2, ..., x_K) \in \mathbb{R}^K$ to make predictions $\hat{y}$ as follows:

$$\hat{y} = f(x) = \sum_k \theta_k x_k$$

- Our goal is to find the "best" θ which minimizes the error in prediction.

4 Loss / Cost

- The loss function $\mathcal{L}(\hat{y}, y)$ is the error in prediction for a single training example (or the residual).
- The cost function $J(\theta)$ averages loss over all training examples (also commonly referred to as the empirical loss).
- We examined the squared error loss:

$$\mathcal{L}(\hat{y}, y) = \frac{1}{2}(\hat{y} - y)^2$$

- And the corresponding cost function:

$$J(\theta) = \frac{1}{2N}\sum_i (\hat{y}^{(i)} - y^{(i)})^2$$

5 Derivatives of the loss function

- Thus far, we've been primarily reviewing gradient descent, as our primary optimization tool. In order to implement gradient descent, we need to obtain the vector of partial derivatives (i.e the gradient), for a given cost function. We reviewed how to do this in the preceding labs.
- Consider the squared error loss function:

$$\begin{aligned}\mathcal{L}(\hat{y}, y) &= \frac{1}{2}(\hat{y} - y)^2 \\ \frac{\partial \mathcal{L}}{\partial \theta_k} &= \frac{\partial \mathcal{L}}{\partial \hat{y}}\frac{\partial \hat{y}}{\partial \theta_k} \\ &= (\hat{y} - y)x_k\end{aligned}$$

- Here, we simply applied the chain rule to find the partial derivative. Note that in the case of the intercept (θ_0) $x_0 = 1$
- We average this over data points (linearity of differentiation) to obtain:

$$\frac{\partial J}{\partial \theta_k} = \frac{1}{N}\sum_i (\hat{y} - y)x_k$$

- The minimum of the function must occur at the point where the partial derivatives are 0.
- Note that we added in $\frac{1}{2}$ to the loss function – this is to simplify calculations

6 Loss Functions Summary

	Loss Function	Partial Derivative
Squared Error	$\mathcal{L}(\hat{y}, y) = \frac{1}{2}(\hat{y} - y)^2$	$\frac{\partial \mathcal{L}}{\partial \theta_k} = (\hat{y} - y)x_k$
Squared Error + Ridge	$\mathcal{L}(\hat{y}, y) = \frac{1}{2}[(\hat{y} - y)^2 + \lambda \sum_k \theta_k^2]$	$\frac{\partial \mathcal{L}}{\partial \theta_k} = (\hat{y} - y)x_k + \lambda\theta_k$
Log Loss	$\mathcal{L}(\hat{y}, h) = -[ylog(h) + (1-y)log(1-h)]$	$\frac{\partial \mathcal{L}}{\partial \theta_k} = (y - h)x_k$ where $h = \frac{1}{1+e^{-(\sum_k \theta_k x_k)}}$

References

CSC 311 - Fall 2023