CSE517A Machine Learning **Spring 2020**

Lecture 4: MLE and MAP for Discriminative Supervised Learning

Instructor: Marion Neumann *Scribe: Jingyu Xin*

Reading: FCML 2.8 (MLE), 3.8 (MAP), 4.2-4.3 (MAP), 5.2 (Bayes Classifier and Logistic Regression)

Application

Let's consider our yield prediction problem from last lecture. This can be cast as a classical *discriminative supervised learning problem*: predict the **production of bushels of corn** per acre on a farm as a function of the proportion of that farm's planting area that was treated with a new pesticide by modeling $p(y \mid \mathbf{x})$ which incorporates a reasonable way to model the noise in the observed data (https://www.developer.com/mgmt/real-world-machine-learning-model-evaluation-and-optimization.html).

In addition to the point estimate of the yield for a given amount of treated area, it will be very informative for the farmer to know what the expected deviation from this point estimate is. In other words, we would like to provide the standard deviation as an estimator of uncertainty.

http://www.corncapitalinnovations.com/production/300-bushel-corn/

1 Introduction

1.1 Predictive Distribution

In **discriminative supervised machine learning** our goal is to model the *posterior predictive distribution*

$$\begin{aligned} p(y \mid D, \mathbf{x}) &= \int_\theta p(y, \theta \mid D, \mathbf{x})\, \mathrm{d}\theta \\ &= \int_\theta p(y \mid D, \mathbf{x}, \theta)\, p(\theta \mid D)\, \mathrm{d}\theta \end{aligned} \tag{1}$$

This makes sense, since we really want to incorporate **<u>all</u> possible models** parameterized by their respective model parameters θ weighted by the parameter's probability (i.e. the *posterior probability over parameters*); cf. FCML 3.8.6.

Unfortunately, the above integral is generally intractable in closed form and sampling techniques, such as *Monte Carlo approximations*, are used to approximate the distribution. So, oftentimes we will actually not use this distribution for predictions but estimate the model parameters via MLE or MAP and then plug those into our model $p(y \mid \mathbf{x}, \hat{\theta})$ for predictions. We will meet the posterior predictive distribution again when discussing Gaussian processes later in the course.

1.2 Parameter Estimation

Discriminative Supervised Learning Assumptions

Usually, there are two assumptions in *discriminative supervised learning*:

(1) $\mathbf{x}_i$ are known $\Rightarrow$ $\mathbf{x}_i$ independent of the model parameters $\mathbf{w}$ $\Rightarrow p(X \mid \mathbf{w}) = p(X)$, also $p(\mathbf{w} \mid X) = p(\mathbf{w})$

(2) $y_i's$ are independent given the input features $\mathbf{x}_i$ and $\mathbf{w}$

Notation: $X = [\mathbf{x}_1, ..., \mathbf{x}_n] \in \mathbb{R}^{d \times n}$ where $\mathbf{x}_i \in \mathbb{R}^d$; $\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \in \mathbb{R}^n$

Our goal is to estimate $\mathbf{w}$ directly from $D = \{(\mathbf{x}, y_i)\}_{i=1}^n$ using the *joint conditional likelihood* $p(\mathbf{y} \mid X, \mathbf{w})$.

Exercise 1.1. Prove that maximizing the likelihood $p(D \mid \mathbf{w}) = p(\mathbf{y}, X \mid \mathbf{w})$ is equivalent to maximizing the conditional likelihood $p(\mathbf{y} \mid X, \mathbf{w})$. HINT: use assumption **(1)**.

Maximum Likelihood Estimation

Choose $\mathbf{w}$ to maximize the *conditional likelihood.*

$$\begin{aligned} \hat{\mathbf{w}}_{MLE} &= \arg\max_{\mathbf{w}} p(\mathbf{y} \mid X, \mathbf{w}) \\ &\overset{\mathbf{(2)}}{=} \arg\max_{\mathbf{w}} \prod_{i=1}^{n} p(y_i \mid \mathbf{x}_i, \mathbf{w}) \\ &= \arg\max_{\mathbf{w}} \underbrace{\sum_{i=1}^{n} \log p(y_i \mid \mathbf{x}_i, \mathbf{w})}_{log-likelihood} \end{aligned} \tag{2}$$

Maximum-a-posterior Estimation

Bayesian Way: Model $\mathbf{w}$ as a *random variable* from $p(\mathbf{w})$ and use $p(\mathbf{w} \mid D)$.
Choose $\mathbf{w}$ to maximize the posterior $p(\mathbf{w} \mid X, \mathbf{y})$ over $\mathbf{w}$.

$$\begin{aligned} \hat{\mathbf{w}}_{MAP} &= \arg\max_{\mathbf{w}} p(\mathbf{w} \mid X, \mathbf{y}) \\ &= \arg\max_{\mathbf{w}} \underbrace{p(\mathbf{y} \mid X, \mathbf{w})}_{likelihood} \underbrace{p(\mathbf{w})}_{prior} \\ &= \arg\max_{\mathbf{w}} \underbrace{\sum_{i=1}^{n} \log p(y_i \mid \mathbf{x}_i, \mathbf{w})}_{\text{same as MLE}} + \log p(\mathbf{w}) \end{aligned} \tag{3}$$

2 Example: Linear Regression

Model Assumption: $y_i = \mathbf{w}^\top \mathbf{x}_i + \epsilon_i \in \mathbb{R}$, where we use the Gaussian distribution (cf. FCML 2.5.3) to model the noise $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$, which is *independent identically distributed* (iid).

$$\Rightarrow y_i \mid \mathbf{x}_i, \mathbf{w} \sim \mathcal{N}(\mathbf{w}^\top \mathbf{x}_i, \sigma^2) \quad \Rightarrow \; p(y_i \mid \mathbf{x}_i, \mathbf{w}) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(\mathbf{w}^\top \mathbf{x}_i - y_i)^2}{2\sigma^2}} \tag{4}$$

where σ^2 is a hyperparameter.

2.1 Learning Phase

To train our model we estimate $\mathbf{w}$ from D.

MLE

Use Eq.(2):

$$\begin{aligned}
\hat{\mathbf{w}}_{MLE} &= \arg\max_{\mathbf{w}} \sum_{i=1}^{n} \log p(y_i \mid \mathbf{x}_i, \mathbf{w}) \\
&= \arg\max_{\mathbf{w}} \sum_{i=1}^{n} \log(\frac{1}{\sqrt{2\pi\sigma^2}}) + \log(e^{\frac{-(\mathbf{w}^\top \mathbf{x}_i - y_i)^2}{2\sigma^2}}) \\
&= \arg\max_{\mathbf{w}} \sum_{i=1}^{n} -(\mathbf{w}^\top \mathbf{x}_i - y_i)^2 \\
&= \arg\min_{\mathbf{w}} \underbrace{\frac{1}{n} \sum_{i=1}^{n} (\mathbf{w}^\top \mathbf{x}_i - y_i)^2}_{\text{OLS/squared loss}}
\end{aligned} \tag{5}$$

The loss thus $l(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^{n} (\mathbf{w}^\top \mathbf{x}_i - y_i)^2$ aka square loss or Ordinary Least Squares (OLS). OLS can be optimized with gradient descent, Newton's method, or in closed form.

Closed Form Solution: $\mathbf{w} = (XX^\top)^{-1} X\mathbf{y}$.
Note: We need to take the inverse; for low dimensional data this is fine since $XX^\top$ is $d \times d$, for high-dimensional data we will have to get an approximate solution.

MAP

Additional Model Assumption: prior distribution (ensure for yourself that the following is a *conjugate prior* to our likelihood)

$$p(\mathbf{w}) = \frac{1}{\sqrt{2\pi\sigma_p^2}} e^{\frac{-\mathbf{w}^\top \mathbf{w}}{2\sigma_p^2}}$$

Use Eq.(3):

$$\begin{aligned}
\hat{\mathbf{w}}_{MAP} &= \arg\max_{\mathbf{w}} \sum_{i=1}^{n} \log p(y_i \mid \mathbf{x}_i, \mathbf{w}) + \log p(\mathbf{w}) \\
&= \arg\min_{\mathbf{w}} \frac{1}{2\sigma^2} \sum_{i=1}^{n} (\mathbf{w}^\top \mathbf{x}_i - y_i)^2 + \frac{1}{2\sigma_p^2} \mathbf{w}^\top \mathbf{w} \\
&= \arg\min_{\mathbf{w}} \underbrace{\frac{1}{n} \sum_{i=1}^{n} (\mathbf{w}^\top \mathbf{x}_i - y_i)^2}_{\text{squared loss}} + \underbrace{\lambda ||\mathbf{w}||_2^2}_{l_2-regularization}
\end{aligned} \tag{6}$$

This formulation is known as *ridge regression* and we have derived it before in a frequentist setting using structural risk minimization (SRM).

Closed Form Solution: $\mathbf{w} = (XX^\top + \lambda I)^{-1} X\mathbf{y}$.
Note: The solution is numerically more stable as the term λI makes it numerically invertible.

2.2 Phase Prediction

Use the estimated model parameters $\hat{\mathbf{w}}$ in predictive distribution $p(y^* \mid \mathbf{x}^*, \hat{\mathbf{w}})$. For linear regression we have

$$p(y^* \mid \mathbf{x}^*, \hat{\mathbf{w}}) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(\hat{\mathbf{w}}^\top \mathbf{x}^* - y^*)^2}{2\sigma^2}}.$$

The point estimate would be given by the mean of this distribution: $\hat{y}^* = \hat{\mathbf{w}}^\top \mathbf{x}^*$.

2.3 Summary

- MLE solution is equivalent to ordinary least squares regression.
- MAP solution is equivalent to regularized OLS using an l_2 regularizer.
- We could use a different noise model such as the full Gaussian $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$, multiplicative noise, or non-stationary noise (e.g. *heteroscedastic noise*) to make this model more expressive.

Exercise 2.1. True or false? Justify your answer.

(a) If $n \to \infty$, MAP can recover from a wrong prior distribution over parameters, where we assume that our prior distribution is strictly larger than zero on [0,1].

(b) The MAP solution to linear regression is numerically less efficient to compute than the MLE solution.

3 Example: Logistic Regression

Model Assumption: We need to squash $\mathbf{w}^\top \mathbf{x}_i$ to get a value in [0,1]. In logistic regression we model $p(y \mid \mathbf{x}, \mathbf{w})$ and assume that it takes on the form:

$$p(y \mid \mathbf{x}, \mathbf{w}) = Ber(y \mid \frac{1}{1 + e^{-\mathbf{w}^\top \mathbf{x}}}), \tag{7}$$

where we use the Bernoulli distribution (cf. FCML 2.3.1):

$$Ber(a \mid \theta) = \begin{cases} \theta & \text{if } a = 1 \\ 1 - \theta & \text{if } a = -1. \end{cases}$$

For binary classification our observations are $y \in \{-1, +1\}$ and we can write Eq.(7) as $p(y \mid \mathbf{x}, \mathbf{w}) = \frac{1}{1+e^{-y(\mathbf{w}^\top \mathbf{x})}}$.

Exercise 3.1. Verify that $p(y \mid \mathbf{x}, \mathbf{w}) = \frac{1}{1+e^{-y(\mathbf{w}^\top \mathbf{x})}}$ is equivalent to Eq.(7).

3.1 Learning Phase

MLE

Now, plug this into Eq.(2) to get:

$$\begin{aligned} \hat{\mathbf{w}}_{MLE} &= \arg\max_{\mathbf{w}} \sum_{i=1}^{n} \log p(y_i \mid \mathbf{x}_i, \mathbf{w}) \\ &= \arg\min_{\mathbf{w}} \underbrace{\sum_{i=1}^{n} \log(1 + e^{-y_i(\mathbf{w}^\top \mathbf{x}_i)})}_{\text{negative log likelihood (nll)}} \end{aligned} \tag{8}$$

We need to estimate the parameter $\mathbf{w}$. To find the values of the parameter at minimum, we can try to find solutions for $\nabla_{\mathbf{w}} \sum_{i=1}^{n} \log(1 + e^{-y_i(\mathbf{w}^\top \mathbf{x}_i)}) = 0$. This equation has no closed form solution, so we will use Gradient Descent on the negative log likelihood $nll(\mathbf{w}) = \sum_{i=1}^{n} \log(1 + e^{-y_i(\mathbf{w}^\top \mathbf{x}_i)})$.

MAP

In the MAP estimate we treat $\mathbf{w}$ as a random variable and can specify a prior belief distribution over it. **Additional Model Assumption**:

$$\mathbf{w} \sim N(\mathbf{0}, \sigma^2 I)$$
$$p(\mathbf{w}) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-\mathbf{w}^\top \mathbf{w}}{2\sigma^2}}$$

So, using Eq.(3) we have:

$$\begin{aligned}
\hat{\mathbf{w}}_{MAP} &= \operatorname*{arg\,max}_{\mathbf{w}} \sum_{i=1}^{n} \log p(y_i \mid \mathbf{x}_i, \mathbf{w}) + \log p(\mathbf{w}) \\
&= \operatorname*{arg\,min}_{\mathbf{w}} \sum_{i=1}^{n} \log(1 + e^{-y_i(\mathbf{w}^\top \mathbf{x}_i)}) + \frac{1}{2\sigma^2} ||\mathbf{w}||_2^2 \\
&= \operatorname*{arg\,min}_{\mathbf{w}} \underbrace{\sum_{i=1}^{n} \log(1 + e^{-y_i(\mathbf{w}^\top \mathbf{x}_i)}) + \lambda ||\mathbf{w}||_2^2}_{\text{negative log posterior (nlp)}}
\end{aligned} \tag{9}$$

where $\lambda = \frac{1}{2\sigma^2}$. Once again, this function has no closed form solution, but we can use Gradient Descent on the negative log posterior $l(\mathbf{w}) = \sum_{i=1}^{n} \log(1 + e^{-y_i(\mathbf{w}^\top \mathbf{x}_i)}) + \lambda ||\mathbf{w}||_2^2$ to find the optimal parameter. Note again that we derived this before via SRM using the log-loss and l_2-regularization (frequentist approach).

[optional] True Bayesian Logistic Regression

We have two options:

- Instead of approximating $p(\mathbf{y} \mid X, \mathbf{w})$ and $p(\mathbf{w})$, we approximate $p(\mathbf{w} \mid X, \mathbf{y})$ directly using a parametric distribution. The most common approach for this is *Laplace approximation*.
- Derive an algorithm for sampling from the posterior and use this as an approximation.

We will not cover this approach in this course. For further reference see FCML 4.4 and 4.5.

3.2 Prediction Phase

Use $\hat{\mathbf{w}}$ in Eq. (7):

$$p(y^* \mid \mathbf{x}^*, \hat{\mathbf{w}}) = Ber(y^* \mid \frac{1}{1 + e^{-\hat{\mathbf{w}}^\top \mathbf{x}^*}}).$$

To get a point estimate this means

$$\hat{y}^* = \begin{cases} 1 & \text{if } \hat{\mathbf{w}}^\top \mathbf{x}^* \geq 0 \\ -1 & \text{if } \hat{\mathbf{w}}^\top \mathbf{x}^* < 0. \end{cases}$$

which just simplifies to $\hat{y}^* = \text{sign}\left(\hat{\mathbf{w}}^\top \mathbf{x}^*\right)$.

3.3 Summary

Logistic regression is easy to

- fit (estimate $\mathbf{w}$ directly from D, linear in dn)
- interpret as log odds: $\log \frac{p(y=1|\mathbf{x})}{p(y=-1|\mathbf{x})} = \mathbf{w}^\top \mathbf{x}$
- easy to extend to multi-class classification: $p(y = c \mid \mathbf{x}, \mathbf{w}) = \frac{e^{\mathbf{w}_c^\top \mathbf{x}}}{\sum_c e^{\mathbf{w}_c^\top \mathbf{x}}}$

Exercise 3.2. One benefit of LR is that it is easy to interpret. This can be seen by looking at the log odds:

$$\log \frac{p(y=1 \mid \mathbf{x}, \mathbf{w})}{p(y=-1 \mid \mathbf{x}, \mathbf{w})}$$

Show that

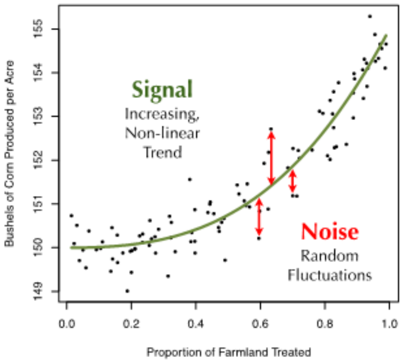
$$\log \frac{p(y=1 \mid \mathbf{x}, \mathbf{w})}{p(y=-1 \mid \mathbf{x}, \mathbf{w})} = \mathbf{w}^\top \mathbf{x}$$

Application

Back to our application of predicting the **production of bushels of corn** per acre on a farm as a function of the proportion of that farm's planting area that was treated with pesticides.

The data clearly shows a non-liner relationship between x and y. How could you use the MLE and MAP solutions developed in Section 2 to model this trend?

(Image source: https://www.developer.com/mgmt/real-world-machine-learning-model-evaluation-and-optimization.html)


Signal
Increasing, Non-linear Trend
Noise
Random Fluctuations
Bushels of Corn Produced per Acre
Proportion of Farmland Treated