CS 189/289A Introduction to Machine Learning
Fall 2022 Jennifer Listgarten, Jitendra Malik

DIS1

1 Maximum Likelihood Estimation

Maximum Likelihood Estimation (MLE) is a method of estimating the parameters of a statistical model given observations, by finding the parameters that maximize the likelihood of the observations. Concretely, given observations $y_1, y_2, \ldots, y_n$ distributed according to $p_\theta(y_1, y_2, \ldots, y_n)$ (here p_θ can be a probability mass function for discrete observations or a density for continuous observations), the likelihood function is defined as $L(\theta) = p_\theta(y_1, y_2, \ldots, y_n)$ and the MLE is

$$\hat{\theta}_{\text{MLE}} = \arg\max_\theta L(\theta).$$

We often make the assumption that the observations are *independent and identically distributed* or iid, in which case $p_\theta(y_1, y_2, \ldots, y_n) = p_\theta(y_1) \cdot p_\theta(y_2) \cdot \cdots \cdot p_\theta(y_n)$.

(a) Your friendly TA recommends maximizing the log-likelihood $\ell(\theta) = \log L(\theta)$ instead of $L(\theta)$. Why does this yield the same solution $\hat{\theta}_{\text{MLE}}$? Why is it easier to solve the optimization problem for $\ell(\theta)$ in the iid case? Given the observations $y_1, y_2, \ldots, y_n$, write down both $L(\theta)$ and $\ell(\theta)$ for the Gaussian $f_\theta(y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(y-\mu)^2}{2\sigma^2}}$ with $\theta = (\mu, \sigma)$.

Solution: As the log is strictly monotonically increasing, maximizing $\ell(\theta) = \log L(\theta)$ and $L(\theta)$ will yield the same solution. Concretely, if θ^* is a unique maximum of $L(\theta)$, we have $L(\theta) < L(\theta^*)$ for all $\theta \neq \theta^*$ in the parameter space and therefore due to strict monotonicity of the log, $\ell(\theta) = \log L(\theta) < \log L(\theta^*) = \ell(\theta^*)$, which means θ^* is also a unique maximum of $\ell(\theta)$.

In the iid case, the log-likelihood decomposes into a sum

$$\ell(\theta) = \sum_{i=1}^{n} \log f_\theta(y_i)$$

and it is often easier to optimize over these sums rather than products:

Numerically: There are special algorithms like stochastic gradient descent available for sums that you will learn about later in lecture. Another reason is that forming the product of many probabilities will yield a very small number and it is easy to generate a floating point underflow this way. On the other hand, adding the logs of probabilities is a more stable operation because the partial sums stay in a reasonable range.

Analytically: Usually it is easier to compute the gradient of $\ell(\theta)$ than for $L(\theta)$. As an example, consider the case of a Gaussian distribution:

The likelihood function is

$$L(\theta) = \left(\frac{1}{2\pi\sigma^2}\right)^{n/2} \cdot e^{-\frac{\sum_{i=1}^{n}(y_i-\mu)^2}{2\sigma^2}}.$$

Taking logs yields

$$\ell(\theta) = \sum_{i=1}^{n} \log f_\theta(y_i) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i-\mu)^2$$

which is much easier to minimize than $L(\theta)$.

(b) The Poisson distribution is $f_\lambda(y) = \frac{\lambda^y e^{-\lambda}}{y!}$. Let $Y_1, Y_2, \ldots, Y_n$ be a set of independent and identically distributed random variables with Poisson distribution with parameter λ. Find the joint distribution of $Y_1, Y_2, \ldots, Y_n$. Find the maximum likelihood estimator of λ as a function of observations $y_1, y_2, \ldots, y_n$.

Solution:

The joint probability mass function is the product of the probability mass functions of all n independent variables y_i,

$p_\theta(y_1, y_2, \ldots, y_n) = \prod_{i=1}^{n} \frac{\lambda^{y_i} e^{-\lambda}}{y_i!}$.

The log likelihood will thus be $\ell(\lambda) = \sum_{i=1}^{n}(y_i \log(\lambda) - \lambda - \log(y_i!))$

We find the maximum by finding the derivative and setting it to 0:

$\ell'(\lambda) = (\sum_{i=1}^{n} \frac{y_i}{\lambda}) - n = 0$. Hence, the estimate should be $\hat{\lambda} = \frac{\sum_{i=1}^{n} y_i}{n} = \bar{Y}$, which is the mean of the observations.

2 ℓ_1- and ℓ_2-Regularization

Consider sample points $X_1, X_2, \ldots, X_n \in \mathbb{R}^d$ and associated values $y_1, y_2, \ldots, y_n \in \mathbb{R}$, an $n \times d$ design matrix $X = [X_1 \quad \ldots \quad X_n]^\top$ and an n-vector $y = [y_1 \quad \ldots \quad y_n]^\top$.

For the sake of simplicity, assume (1) that the sample data have been centered (i.e each feature has mean 0) and (2) that the sample data have been whitened, meaning a linear transformation is applied to the original data matrix so that the resulting features have variance 1 and the features are uncorrelated; i.e., $X^\top X = nI$.

For this question, we will not use a fictitious dimension nor a bias term; our linear regression function will output zero for $x = 0$.

Consider linear least-squares regression with regularization in the ℓ_1-norm, also known as Lasso. The Lasso cost function is

$$J(w) = |Xw - y|^2 + \lambda \|w\|_1$$

where $w \in \mathbb{R}^d$ and $\lambda > 0$ is the regularization parameter. Let $w^* = \arg\min_{w \in \mathbb{R}^d} J(w)$ denote the weights that minimize the cost function.

In the following steps, we will explore the sparsity-promoting property of the ℓ_1-norm and compare this with the ℓ_2-norm.

1. We use the notation X_{*i} to denote column i of the design matrix X, which represents the i^{th} feature. Write $J(w)$ in the following form for appropriate functions g and f.

$$J(w) = g(y) + \sum_{i=1}^{d} f(X_{*i}, w_i, y, \lambda)$$

Solution: We expand the objective, and simplify it using the fact that $X^\top X = nI$ since the data is whitened.

$$\begin{aligned} J(w) &= w^\top X^\top X w - 2y^\top X w + \lambda \|w\|_1 + \|y\|^2 \\ &= n|w|^2 - 2y^\top X w + \lambda \|w\|_1 + |y|^2 \end{aligned}$$

We can write each term in sum form: $n|w|^2 = \sum_{i=1}^{d} nw_i^2$. $\lambda\|w\|_1 = \sum_{i=1}^{d} \lambda|w_i|$. And $-2y^\top Xw = \sum_{i=1}^{d} -2y^\top X_{*i}w_i$. So the appropriate functions are $g(y) = |y|^2$, and

$$f(X_{*i}, w_i, y, \lambda) = nw_i^2 - 2y^\top X_{*i}w_i + \lambda|w_i|$$

2. If $w_i^* > 0$, solve for the optimal value w_i^*. *Hint: use your answer in the previous part.*

Solution: We want to minimize

$$-2y^\top X_{*i}w_i + nw_i^2 + \lambda w_i.$$

Setting the derivative to zero yields

$$w_i^* = \frac{1}{n}(y^\top X_{*i} - \lambda/2).$$

3. If $w_i^* < 0$, solve for the optimal value w_i^*.

Solution: We want to minimize

$$-2y^\top X_{*i}w_i + nw_i^2 - \lambda w_i.$$

Setting the derivative to zero yields

$$w_i^* = \frac{1}{n}(y^\top X_{*i} + \lambda/2).$$

4. Considering parts 2 and 3, what is the condition for w_i^* to be zero?

Solution: w_i^* cannot be positive if $y^\top X_{*i} - \lambda/2 \leq 0$, and w_i^* cannot be negative if $y^\top X_{*i} + \lambda/2 \geq 0$. So w_i^* is zero if both are true, i.e., $-\lambda \leq 2y^\top X_{*i} \leq \lambda$.

5. Now consider ridge regression, which uses the ℓ_2 regularization term $\lambda |w|^2$. How does this change the function $f(\cdot)$ from part 1? What is the new condition in which $w_i^* = 0$? How does it differ from the condition you obtained in part 4?

Solution: The portion $f(\cdot)$ of the cost function involving w_i is

$$-2y^\top X_{*i} w_i + n w_i^2 + \lambda w_i^2.$$

Setting the derivative to zero yields

$$w_i^* = \frac{y^\top X_{*i}}{n + \lambda}.$$

Hence w_i^* is zero if $y^\top X_{*i} = 0$. In contrast, $w_i^* = 0$ when $|2y^\top X_{*i}| < \lambda$ in Lasso regression. This is why ℓ_1-norm regularization encourages sparsity.

Also note that we have shown that whitened training data decouples the features, so that w_i^* is determined by the i^{th} feature alone (i.e., column i of the design matrix X), regardless of the other features. This is true for both Lasso and ridge regression.

3 Probabilistic Interpretation of Lasso

Let's start with the probabilistic interpretation of least squares. Start with labels $y \in \mathbb{R}$, data $\mathbf{x} \in \mathbb{R}^d$, and noise $z \sim \mathcal{N}(0, \sigma^2)$, where $y = \mathbf{w}^T\mathbf{x} + z$. Recall from lecture that we then have

$$P(y|\mathbf{x}, \sigma^2) \sim \mathcal{N}(\mathbf{w^T x}, \sigma^2)$$

However, maximum likelihood estimates (MLE) can overfit by picking parameters that mirror the training data. To ameliorate this issue, we can assume a Laplace prior on $w_j \sim \text{Laplace}(0, t)$, i.e.

$$P(w_j) = \frac{1}{2t} e^{-|w_j|/t}$$

$$P(\mathbf{w}) = \prod_{j=1}^{D} P(w_j) = (\frac{1}{2t})^D \cdot e^{-\frac{\sum |w_j|}{t}}$$

Here, we will see that this modification results in the Lasso objective function.

Recall that the MLE objective finds the parameters that maximize the likelihood of the data,

$$\begin{aligned}
\mathbf{w}^* &= \arg\max_{\mathbf{w}} \ L(\mathbf{w}) \\
&= \arg\max_{\mathbf{w}} \ P(Y_1, \ldots, Y_n, |\mathbf{w}, \mathbf{X_1}, \ldots, \mathbf{X_n}, \sigma^2) \\
&= \arg\max_{\mathbf{w}} \ \prod_{i=n}^{n} P(Y_i|\mathbf{X_i}, \mathbf{w}, \sigma^2).
\end{aligned}$$

When working in a Bayesian framework, we instead focus on the posterior distribution of the parameters conditioned on the data, $P(\mathbf{w}|Y_1, \ldots, Y_n, \mathbf{X_1}, \ldots, \mathbf{X_n}, \sigma^2)$. To pick a single model, we can choose the $\mathbf{w}$ that is most likely according to the posterior,

$$\begin{aligned}
\mathbf{w}^* &= \arg\max_{\mathbf{w}} \ P(\mathbf{w}|Y_1, \ldots, Y_n, \mathbf{X_1}, \ldots, \mathbf{X_n}, \sigma^2) \\
&= \arg\max_{\mathbf{w}} \ \frac{P(\mathbf{w}, Y_1, \ldots, Y_n|\mathbf{X_1}, \ldots, \mathbf{X_n}, \sigma^2)}{P(Y_1, \ldots, Y_n|\mathbf{X_1}, \ldots, \mathbf{X_n}, \sigma^2)} \\
&= \arg\max_{\mathbf{w}} \ \frac{P(Y_1, \ldots, Y_n, |\mathbf{w}, \mathbf{X_1}, \ldots, \mathbf{X_n}, \sigma^2)P(\mathbf{w})}{P(Y_1, \ldots, Y_n|\mathbf{X_1}, \ldots, \mathbf{X_n}, \sigma^2)} \\
&= \arg\max_{\mathbf{w}} \ \frac{L(\mathbf{w})P(\mathbf{w})}{P(Y_1, \ldots, Y_n)} \\
&= \arg\max_{\mathbf{w}} \ L(\mathbf{w})P(\mathbf{w}) \qquad \text{since } P(Y_1, \ldots, Y_n|\mathbf{X_1}, \ldots, \mathbf{X_n}, \sigma^2) \text{ does not depend on } \mathbf{w}.
\end{aligned}$$

We call $\mathbf{w}^*$ the Maximum a posteriori (MAP) estimate.

(a) Write the log-likelihood for this MAP estimate.

Solution:

We start with the likelihood.

$$P(w|\mathbf{X_i}, Y_i) \propto (\prod_{i=1}^{n} \mathcal{N}(Y_i|\mathbf{w^T X_i}, \sigma^2)) \cdot P(\mathbf{w}) = (\prod_{i=1}^{n} \mathcal{N}(Y_i|\mathbf{w^T X_i}, \sigma^2)) \cdot \prod_{j=1}^{D} P(w_j)$$

Taking the log of the above expression, we now have:

$$\begin{aligned} l(\mathbf{w}) &= \sum_{i=1}^{n} log\mathcal{N}(Y_i|\mathbf{w^T X_i}, \sigma^2) + \sum_{j=1}^{D} \log P(w_j) \\ &= \sum_{i=1}^{n} \log\left(\frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(Y_i - \mathbf{w^T X_i})^2}{2\sigma^2})\right) + \sum_{j=1}^{D} \log\left(\frac{1}{2t} exp(\frac{-|w_j|}{t})\right) \\ &= -\sum_{i=1}^{n} \frac{(Y_i - \mathbf{w^T X_i})^2}{2\sigma^2} + \frac{-\sum_{j=1}^{D} |w_j|}{t} + nlog(\frac{1}{\sqrt{2\pi}\sigma}) + D\log\left(\frac{1}{2t}\right) \end{aligned}$$

(b) We already have the log-likelihood for MAP. Show that the MAP you derived in the previous part (in this case, Gaussian noise with a Laplace prior) is equivalent to minimizing the following. Additionally, identify the constant λ. Note that $\|\mathbf{w}\|_1 = \sum_{j=1}^{D} |w_j|$.

$$J(\mathbf{w}) = \sum_{i=1}^{n} (Y_i - \mathbf{w^T X_i})^2 + \lambda \|\mathbf{w}\|_1$$

Solution: We drop constants from the above expression, which do not affect the argmax of our expression. We multiply both sides by $\frac{2\sigma^2}{t}$ to further simplify the expression - note that multiplying $J(\mathbf{w})$ by a constant does not change the value at which the expression will be minimized.

$$J(\mathbf{w}) = \sum_{i=1}^{n}(Y_i - \mathbf{w^T X_i})^2 + \frac{2\sigma^2}{t}\sum_{j=1}^{D}|w_j| = \sum_{i=1}^{n}(Y_i - \mathbf{w^T X_i})^2 + \lambda||\mathbf{w}||_1$$

where $\lambda = \frac{2\sigma^2}{t}$.