CS 412 — Introduction to Machine Learning (UIC) February 06, 2025

Lecture 5

Instructor: Aadirupa Saha *Scribe(s): Amith Bhat Hosadurga Anand*

Overview

In the last lecture, we covered the following main topics:

1. Logistic Regression
2. Multiclass Logistic Regression (MLR)

This lecture focuses on:

1. Multiclass Logistic Regression (contd)
2. MLE (Bernoulli)
3. MAP (Bernoulli)

1 Multiclass Logistic Regression

1.1 A Brief Recap of Multiclass Logistic Regression

Consider our Multiclass classification problem, with the input (or instance) space X. Each instance $x \in X$ is a d-dimensional real valued vector(i.e. x $\in$ X $\subseteq \mathbb{R}^d$).
The output space (or label space) $y = \{1, 2, 3....C\}$ consists of class labels, where C represents the total number of classes available for classification
The class of all linear multi-class logistic functions is given by

$$\boxed{\mathcal{F}_N^{\text{lin. multiclass logistic}} = \left\{ f_W : X \to \Delta_C \mid f_W(x)[j] = \frac{\exp(w_j^T x)}{\sum_{k=1}^C \exp(w_k^T x)}, \quad f_W(x) \in \Delta_C, \quad \forall x \in \mathbb{R}^d \right\}}$$

where

1. $\Delta_C = \left\{ x \in [0,1]^C \mid \sum_{i=1}^C x_i = 1 \right\}$,
2. $f_W(x)$ is the probability distribution over C classes,
3. $f_W(x)[j]$ is the j^{th} component of $f_W(x) \in \Delta_C$, and
4. $softmax(w^T \cdot x)_j$ = $\frac{\exp(w_j^T x)}{\sum_{k=1}^C \exp(w_k^T x)}$

We define W as

$$W = \begin{bmatrix} \leftarrow w_1 \rightarrow \\ \leftarrow w_2 \rightarrow \\ \vdots \\ \leftarrow w_C \rightarrow \end{bmatrix}_{(C \times d)}$$

Definition 1 (Translation Invariance)**.** *A function $f : \mathcal{X} \mapsto \mathbb{R}$ is called translationally invariant if shifting its input by some fixed amount does not change its output. That is, for all $x \in \mathcal{X}$ and for any translation $a \in \mathcal{X}$, the function satisfies: $f(x + a) = f(x) \quad \forall x, a \in \mathcal{X}$*

Exercise 1.1: Softmax Function in translationally invariant

Prove that the softmax function defined above is translationally invariant in w meaning that shifting all elements of w by the same scalar α does not change the output

$$softmax(w + \alpha)_j = softmax(w)_j$$

for all j, where $\alpha \in \mathbb{R}$ is a scalar added to each component of w

Solution

$$softmax(w + \alpha)_j = \frac{\exp(w_j+\alpha)}{\sum_{k=1}^{C} \exp(w_k+\alpha)} = \frac{\exp(w_j)\exp(\alpha)}{\sum_{k=1}^{C} \exp(w_k)\exp(\alpha)} = \frac{\exp(w_j)}{\sum_{k=1}^{C} \exp(w_k)} = softmax(w)_j$$

Since the softmax function is translation invariant under W, we can define $\tilde{W} = W - W_C$, knowing the softmax function will give the same output taking either W or $\tilde{W}$ as the input.
Here W_C is a matrix comprising of dimensions ($C \times d$) where each row is w_C i.e

$$W_C = \begin{bmatrix} \leftarrow w_C \rightarrow \\ \leftarrow w_C \rightarrow \\ \vdots \\ \leftarrow w_C \rightarrow \end{bmatrix} \implies \tilde{W} = \begin{bmatrix} \leftarrow w_1 - w_C \rightarrow \\ \vdots \\ \leftarrow w_{C-1} - w_C \rightarrow \\ \leftarrow w_C - w_C \rightarrow \end{bmatrix} = \begin{bmatrix} \leftarrow w_1 - w_C \rightarrow \\ \vdots \\ \leftarrow w_{C-1} - w_C \rightarrow \\ \leftarrow 0 \rightarrow \end{bmatrix}$$

Since the last row is a d-dimensional vector of zeros, we can discard it leaving us with

$$\tilde{W} = \begin{bmatrix} \leftarrow w_1 - w_C \rightarrow \\ \vdots \\ \leftarrow w_{C-1} - w_C \rightarrow \end{bmatrix}_{(C-1 \times d)}$$

Note: While we subtracted w_C we could achieve the same result by subtracting any w_j

Consider the case of Binary classification ($C = 2$), the dimensions of $\tilde{W}$ i.e $(1, d)$ are consistent with the dimensions of W defined there.
Henceforth whenever we use W in the course we are actually referring to $\tilde{W}$ of dimensions $(C - 1) \times d$

1.2 Linear Multiclass - Logistic Regression Formulation

Next, we try to find the optimal weights W such our loss for Multiclass-Logistic Regression is minimized i.e we want find

$$\arg\min_{W \in \mathbb{R}^{C \times d}} \sum_{i=1}^{n} \ell(y_i, f_W(x_i))$$

where

$$\ell(y_i, f_W(x_i)) = \sum_{j=i}^{C} 1(y_i = j)\,(-\log f_W(x_i)[j])$$

This is not an easy function to optimize due to the non-linearity of the softmax function and in fact does not have a closed form solution in W. We will return to this later.

---

2 Maximum Likelihood Estimator - Bernoulli

2.1 MLE formulation for a Bernoulli Random Variable

Definition 2 (PMF of a Bernoulli Random Variable)**.** *Consider a random variable X that follows the Bernoulli distribution i.e $X \sim Ber(p)$. Its probability mass function is given by*

$$P(X = x) = \begin{cases} p, & \textit{if } x = 1 \\ 1 - p, & \textit{if } x = 0 \end{cases}$$

p is the parameter of the given Bernoulli Distribution

Instead of using cases we can express the pmf as single equation via exponents i.e $P(X = x) = p^x \cdot (1-p)^{1-x}$
if $x = 0$ then $P(X = 0) = p^0 \cdot (1-p)^{(1-0)} = 1 - p$
if $x = 1$ then $P(X = 1) = p^1 \cdot (1-p)^{(1-1)} = p$

Now we are given a dataset $D = \{(x_i, y_i)\}_{i=1}^{n}$, with our task being Binary classification.
Our inputs $x \in X \subseteq \mathbb{R}^d$ and our outputs or labels $y = \{0, 1\}$ or $\{-1, 1\}$ depending on our choice of notation.
Let us assume that our true labels $y_i \sim Ber(\mu^*)$ where μ^* is the true parameter (that is unknown to us) which we seek to estimate or learn.

We begin by defining our likelihood function, $L_D(\mu)$.
Likelihood is the probability of observing the dataset D given the parameter (here) μ. This is equivalent to $P(D|\mu)$.
We wish to find the μ that maximizes our likelihood, so what we have to find is

$$\arg\max_{\mu \in [0,1]} L_D(\mu) = \arg\max_{\mu \in [0,1]} P(D|\mu)$$

Now substituting our earlier defined Dataset values we get

$$P(D|\mu) = P(\{(x_1, y_1), (x_2, y_2), \cdots (x_n, y_n)\}|\mu)$$

We observe that each of the data points (x_i, y_i) are independent of each other i.e (x_1, y_1) is independent of $(x_2, y_2), (x_3, y_3) \cdots (x_n, y_n)$ and vice versa.

Theorem 2.1: Independent Events

Given two independent events A and B:

$$P(A, B) = P(A) \cdot P(B)$$

Applying this to our dataset we get

$$L_D(\mu) = P(\{(x_1, y_1), (x_2, y_2), \cdots (x_n, y_n)\}|\mu) = \prod_{i=1}^{n} P(x_i, y_i|\mu)$$

Theorem 2.2: Conditional Probability

Given three events A, B and C:

$$P(A, B|C) = P(A|B, C) \cdot P(B|C)$$

Applying this to our equation we get

$$L_D(\mu) = \prod_{i=1}^{n} P(x_i, y_i|\mu) = \prod_{i=1}^{n} P(y_i|x_i, \mu) \cdot P(x_i|\mu) = \prod_{i=1}^{n} P(y_i|x_i, \mu) \cdot P(x_i)$$

Since x_i is independent of μ, $P(x_i|\mu)$ simplifies to $P(x_i)$ in our last step.
Since $P(x_i)$ is independent of μ we can ignore this term hence forth in our derivation.

Note: After taking the log likelihood we obtain our MLE by differentiating w.r.t μ. Even if we did not ignore $P(x_i)$ here, since $P(x_i)$ independent of μ its derivative w.r.t μ would be zero, more specifically $\frac{d}{d\mu}\log(P(x_i)) = 0$. Thus we can conclude $P(x_i)$, and more generally terms independent of the parameter we are optimizing for (in this case μ), do not contribute to the optimization process.

Finding the value of the parameter that maximizes likelihood is equivalent to finding the value maximizes the $\log$ of the likelihood. Hence taking $\log$ on both sides of our above equation

$$\log(L_D(\mu)) = \log\left(\prod_{i=1}^{n} P(y_i|x_i, \mu)\cdot\right) = \log\left(\prod_{i=1}^{n} P(y_i|\mu)\cdot\right)$$

The last step follows since we know that from the way y_i's have been generated from the Bernoulli Distribution, they do not depend on any x_i's. Now using our earlier defined pmf pf Bernoulli distribution we can say that

$$P(y_i|\mu) = \mu^{y_i}(1-\mu)^{1-y_i}$$

Substituting this in our equation, and also converting the product to sum by the properties of logarithms we get

$$\log(L_D(\mu)) = \log\left(\prod_{i=1}^{n} P(y_i|\mu)\cdot\right) = \sum_{i=1}^{n} \log\left(\mu^{y_i}(1-\mu)^{1-y_i}\right)$$

$$\log(L_D(\mu)) = \sum_{i=1}^{n} y_i \cdot \log\mu + \sum_{i=1}^{n}(1-y_i)\cdot\log(1-\mu) = \log\mu \cdot \sum_{i=1}^{n} y_i + \log(1-\mu)\cdot\sum_{i=1}^{n}(1-y_i)$$

Before proceeding we define two new terms

Let N_1 be the total number of time label 1 occurred in our dataset D, it can be defined as

$$N_1 = \sum_{i=1}^{N} 1(y_i = 1) = \sum_{i=1}^{N} y_i$$

Similarly let N_0 be the total number of time label 0 occurred in our dataset D, it can be defined as

$$N_0 = \sum_{i=1}^{N} 1(y_i = 0) = \sum_{i=1}^{N} (1 - y_i)$$

$$\text{where} \quad N_1 + N_0 = n$$

Using this we can rewrite our original equation as

$$\log(L_D(\mu)) = N_1 \log(\mu) \; + N_0 \log(1 - \mu)$$

We need to find the μ that maximizes our likelihood or our log likelihood

$$\arg \max_{\mu \in [0,1]} log(L_D(\mu)) = \arg \max_{\mu \in [0,1]} (N_1 \log(\mu) + N_0 \log(1 - \mu))$$

To obtain the argmax of the RHS:

1. Take the derivative w.r.t μ and equate it to zero.
2. Solve for μ
3. The μ so obtained is the MLE for μ

$$\frac{d}{d\mu}(N_1 \log(\mu) + N_0 \log(1 - \mu)) = 0 \implies \frac{N_1}{\mu} - \frac{N_0}{1 - \mu} = 0$$

Solving the above equation for μ we obtain $\hat{\mu}_{MLE} = \dfrac{N_1}{N_1 + N_0}$

2.2 Examples

Exercise 2.1: Calculate $\hat{\mu}_{MLE}$

Consider the following dataset $D = \{(x, 1), (x, 0), (x, 1), (x, 1)\}$
Q1) What is $\hat{\mu}_{MLE}$?
Q2) Recalculate $\hat{\mu}_{MLE}$ for $D = \{(x, 1), (x, 1), (x, 1), (x, 1)\}$?

Solution:

1A) $N_1 = 3, N_0 = 1 \implies \hat{\mu}_{MLE} = \dfrac{3}{3+1} = \dfrac{3}{4}$
2A) $\hat{\mu}_{MLE} = 1$
We see in both cases that $\hat{\mu}_{MLE}$ is over fitting the data
Further exercises on deriving MLE's can be found in questions 2 and 3 of the bonus quiz and in reference 2

3 Maximum Aposteriori Estimate - Bernoulli

3.1 MAP formulation for a Bernoulli Random Variable with a Beta Prior

For MLE we are choosing a parameter value that maximizes the probability of observed .Till now we have only considered μ as a parameter but what if we had some prior information about μ.
Consider the following Prior: $\mu \sim Beta(\alpha, \beta)$.
Then we must find a way to incorporate this prior into our estimation i.e we must choose a parameter value that is most probable given observed data and prior belief. We achieve this by maximizing the posterior probability $P(\mu|D)$.
The estimator used here for this maximization is $\hat{\mu}_{MAP} = \arg\max_{\mu \in [0,1]} P(\mu|D)$

Before continuing with this derivation we will first revisit the properties of the Beta distribution

Definition 3 (PDF of a Beta Random Variable)**.** *Consider a random variable X that follows the Bernoulli distribution i.e $X \sim Beta(\alpha, \beta)$. It's probability density function is given by*

$$P(X = x) = \frac{x^{\alpha-1} \cdot (1-x)^{\beta-1}}{B(\alpha, \beta)}$$

where
$$B(\alpha, \beta) = \frac{\Gamma(\alpha) \cdot \Gamma(\beta)}{\Gamma(\alpha + \beta)} \quad \textit{where} \quad \Gamma(z) = \int_0^\infty t^{z-1} \cdot e^{-t} dt$$

Theorem 3.1: Bayes Theorem

Given two events A and B with $P(B) \neq 0$:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

Coming back to our derivation we can apply Bayes theorem to the posterior probability

$$\arg\max_{\mu \in [0,1]} P(\mu|D) = \arg\max_{\mu \in [0,1]} \frac{P(D|\mu).P(\mu)}{P(D)}$$

where $P(\mu)$ is our prior belief of μ

Similar to the MLE derivation we will apply $\log$ function to make optimization easier

$$\arg\max_{\mu \in [0,1]} \log\left(\frac{P(D|\mu).P(\mu)}{P(D)}\right) = \arg\max_{\mu \in [0,1]} (\log(P(D|\mu) + \log(P(\mu)) - \log(P(D))$$

Again similar to the MLE derivation we can ignore terms that are independent of μ, which is this case is $P(D)$
If we were to write D as $\{(x_1, y_1), (x_2, y_2), \cdots (x_n, y_n)\}$ we observe that $P(D|\mu) = P(\{(x_1, y_1), (x_2, y_2), \cdots (x_n, y_n)\}|\mu)$
The last equation is exactly the same as our likelihood function in our MLE derivation meaning we can directly use the results of that derivation, more specifically

$$\arg\max_{\mu \in [0,1]} log(L_D(\mu)) = \arg\max_{\mu \in [0,1]} P(\{(x_1, y_1), (x_2, y_2), \cdots (x_n, y_n)\}|\mu) = \arg\max_{\mu \in [0,1]} (N_1 \log(\mu) + N_0 \log(1-\mu))$$

Next we can find $P(\mu)$ as we know $\mu \sim Beta(\alpha, \beta) \implies P(\mu) = \dfrac{\mu^{\alpha-1} \cdot (1-\mu)^{\beta-1}}{B(\alpha, \beta)}$

Substituting $P(D|\mu)$ and$P(\mu)$ in our original equation we obtain

$$\arg\max_{\mu\in[0,1]} \left(N_1 \log(\mu) + N_0 \log(1-\mu) + \log\left(\frac{\mu^{\alpha-1}\cdot(1-\mu)^{\beta-1}}{B(\alpha,\beta)}\right)\right)$$

$$\implies \arg\max_{\mu\in[0,1]} \left(N_1 \log(\mu) + N_0 \log(1-\mu) + (\alpha-1)\cdot\log\mu + (\beta-1)\log(1-\mu) - \log(B(\alpha,\beta)\right)$$

We can drop $\log(B(\alpha,\beta)$ as it is independent of μ so our function to optimize to becomes
$$\arg\max_{\mu\in[0,1]} \left(N_1 \log(\mu) + N_0 \log(1-\mu) + (\alpha-1)\cdot\log\mu + (\beta-1)\log(1-\mu)\right)$$

Similar to how we found $\hat{\mu}_{MLE}$ to obtain $\hat{\mu}_{MAP}$ we differentiate our function w.r.t μ and equate it to 0 and solve for μ
Post differentiating we obtain

$$\frac{N_1}{\mu} - \frac{N_0}{1-\mu} + \frac{\alpha-1}{\mu} - \frac{(\beta-1)}{1-\mu} = 0$$

$$\implies \frac{(N_1+\alpha-1)}{\mu} = \frac{N_0+\beta-1}{1-\mu}$$

$$\implies (N_1+\alpha-1) - (N_1+\alpha-1)\cdot\mu = (N_0+\beta-1)\cdot\mu$$

$$\implies (N_1+\alpha-1) = (N_0+\beta-1)\cdot\mu + (N_1+\alpha-1)\cdot\mu$$

$$\implies (N_1+\alpha-1) = (N_0+N_1+\alpha+\beta-2)\cdot\mu$$

Solving the above for μ we obtain our $\hat{\mu}_{MAP}$ which is

$$\boxed{\hat{\mu}_{\text{MAP}} = \frac{N_1+\alpha-1}{N_0+N_1+\alpha+\beta-2}}$$

The presence of α and β in our final estimate of $\hat{\mu}_{MAP}$ indicates the influence of the prior belief on our MAP estimate.

3.2 Examples

1. If $\alpha=\beta=1$, note that $\hat{\mu}_{MAP} = \hat{\mu}_{MLE}$
   By substituting $\alpha = 1$ and $\beta = 1$ in our equation for $\hat{\mu}_{MAP}$ we obtain
   $\hat{\mu}_{MAP} = \dfrac{N_1+1-1}{N_0+N_1+1+1-2} = \dfrac{N_1}{N_1+N_0} = \hat{\mu}_{MLE}$

   This holds as $Beta(1,1)$ is a $Unif(0,1)$ distribution.

   Definition 4 (PDF of a Uniform Random Variable)**.** *Consider a random variable X that follows the Bernoulli distribution i.e $X \sim Unif(a,b)$. It's probability density function is given by*

$$P(X=x) = \begin{cases} \dfrac{1}{b-a} & \textit{if } x \in [a,b] \\ 0 & \textit{everywhere else} \end{cases}$$

In our derivation for $\hat{\mu}_{MAP}$ we substituted the value $P(\mu)$ based on the prior.
If our prior was $Beta(1,1)$ i.e $Unif(0,1)$ then $P(\mu)$ works out to be $\dfrac{1}{1-0} = 1 \implies \log(P(\mu)) = 0$.
From here, on continuing the derivation we would obtain the above result.
$\implies$ In general MAP incorporates prior belief into the estimation process, while MLE assumes a uniform prior, treating all values of μ as equally likely. (The constant value from the uniform prior would get canceled/filtered out in the optimization process as seen above for our $Unif(0,1)$ case)

2. In $Beta(\alpha, \beta)$ distribution, a higher value of $\alpha \implies$ a "RIGHT" heavy pdf, which signals a large prior on μ. This follows as $\hat{\mu}_{MAP}$ is higher for higher α (as α is in the numerator and the denominator)

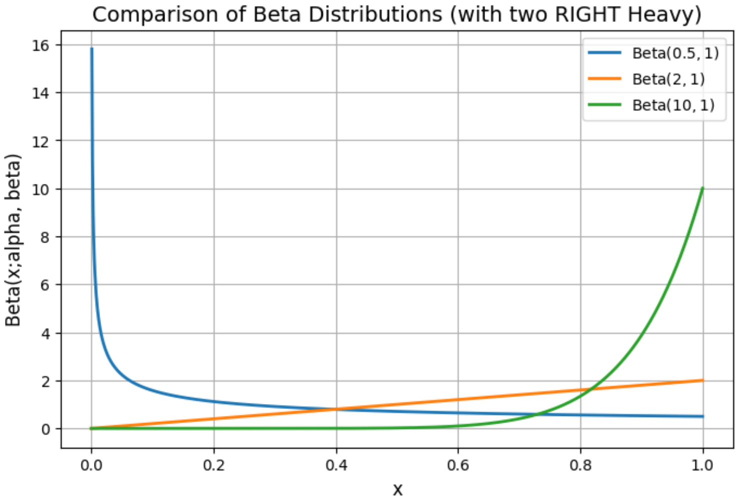

Figure 1: Two "Right Heavy" Beta distributions

3. In contrast to 2. a higher $Beta(\alpha, \beta) \implies$ a "LEFT" heavy pdf which signals a small prior on μ. This follows as $\hat{\mu}_{MAP}$ is lower for higher β (as β is only in the denominator) More such plots can be created

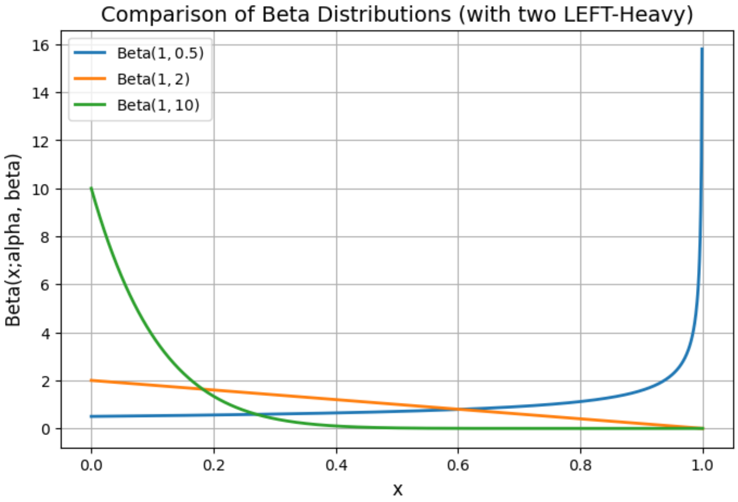

Figure 2: Two "Left Heavy" Beta distributions

by playing around with the parameters in the Beta Demo Notebook

4. Consider a dataset consisting of $N_1 = 6$ positive instances and $N_0 = 4$ negative instances, where each observation is modeled as an independent Bernoulli trial with unknown probability. Initially assume there in no prior or Uniform prior on μ

$$\hat{\mu}_{MLE} = \frac{N_1}{N_1 + N_0} = \frac{6}{10} = 0.6$$

Now assume that μ had a "LEFT" heavy prior of $Beta(1, 10)$, then our map estimate becomes

$$\hat{\mu}_{MAP} = \frac{N_1 + \alpha - 1}{N_0 + N_1 + \alpha + \beta - 2} = \frac{6 + 1 - 1}{6 + 4 + 1 + 10 - 2} = \frac{6}{19} \sim 0.3158$$

Instead of having a "LEFT" heavy prior instead assume a "RIGHT" heavy prior of $Beta(10, 1)$, our map estimate now becomes

$$\hat{\mu}_{MAP} = \frac{N_1 + \alpha - 1}{N_0 + N_1 + \alpha + \beta - 2} = \frac{6 + 10 - 1}{6 + 4 + 10 + 1 - 2} = \frac{15}{19} \sim 0.7895$$

We observe how having different types of priors affect our estimates for μ in the graph below

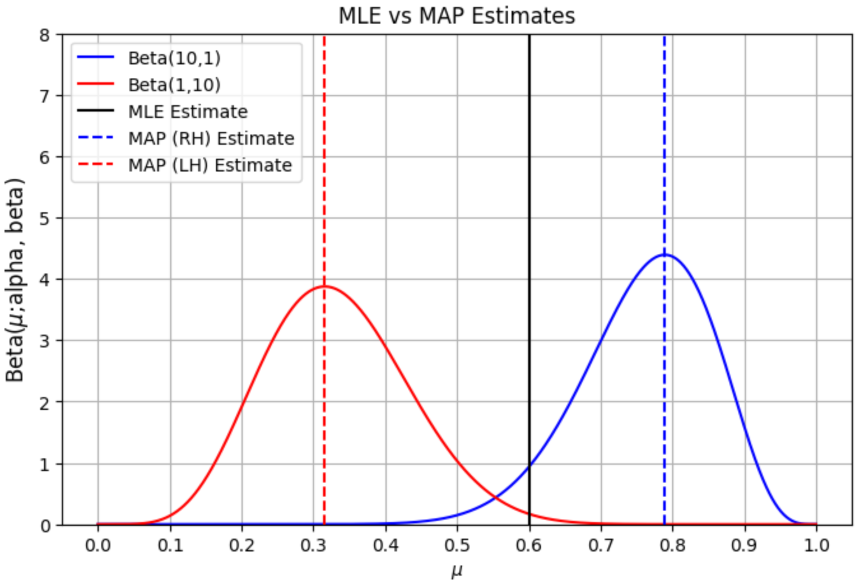

Figure 3: Effect of Priors on estimating μ

Conjugate Priors

Any prior $P(\theta)$ is called the conjugate prior for a likelihood function $L(\theta) = P(D|\theta)$ if the posterior $P(\theta|D)$ is of the same distributional family as $P(\theta)$.

For example the Dirichlet distribution is the conjugate prior for the Multinomial distribution, meaning that the posterior also follows a Dirichlet distribution. However, its parameters are updated based on observed data and need not remain the same as those of the prior

More examples of Conjugate Priors can be found in reference 2 and reference 3

Next Lecture

The next lecture will cover the following topics:

(i) MLE for Regression

(ii) MAP and Regularization

References:

1. Chapter 4.2, 4.5, Probabilistic Machine Learning - Kevin P. Murphy
2. Lecture 2, slides by Prof. Barnabás Póczos and Prof. Aarti Singh from course CMU-10701: Introduction to Machine Learning, 2014 Spring [Link]
3. Estimating Probabilities: MLE-MAP, Machine Learning - Tom Mitchell [Link]
4. Maximum Likelihood Explanation[Link]
5. Beta Demo Notebook [Link]