MLE Review

 [(https://github.com/learn-co-curriculum/dsc-mle-review)](https://github.com/learn-co-curriculum/dsc-mle-review) [(https://github.com/learn-co-curriculum/dsc-mle-review/issues/new/choose)](https://github.com/learn-co-curriculum/dsc-mle-review/issues/new/choose)

Introduction

You've seen MLE (Maximum Likelihood Estimation) when discussing Bayesian statistics, but did you know logistic regression can also be seen from this statistical perspective? In this section, you'll gain a deeper understanding of logistic regression by coding it from scratch and analyzing the statistical motivations backing it. But first take some time to review maximum likelihood estimation.

Objectives

You will be able to:

- Describe how to take MLE of a binomial variable

MLE

Maximum likelihood estimation can often sound academic, confusing, and cryptic when first introduced. It is often presented and introduced with complex integrals of statistical distributions that scare away many readers. Hopefully, this hasn't been your experience. While the mathematics can quickly become complex, the underlying concepts are actually quite intuitive.

To demonstrate this, imagine a simple coin flipping example. Let's say that you flip a coin 100 times and get 55 heads. Maximum likelihood estimation attempts to uncover the underlying theoretical probability of this coin landing on heads given your observations. In other words, given the observations, what is the chance that the coin was fair and had a 0.5 chance of landing on heads each time? Or what is the chance that the coin actually had a 0.75 probability of lands of heads, given what we observed? It turns out that the answer to these questions is rather intuitive. If you observe 55 out of 100 coin flips, the underlying probability which maximizes the chance of us observing 55 out of 100 coin flips is 0.55. In this simple example, MLE simply returns the current sample mean as the underlying parameter that makes the observations most probable. Slight deviations to this would be almost as probable but slightly less so, and large deviations from our sample mean should be rare. This intuitively makes some sense; as your sample size increases, you expect the sample mean to converge to the true underlying parameter. MLE takes a flipped perspective, asking what underlying parameter is most probable given the observations.

Log-likelihood

When calculating maximum likelihood, it is common to use the log-likelihood, as taking the logarithm can simplify calculations. For example, taking the logarithm of a set of products allows you to decompose the problem from products into sums. (You may recall from high school mathematics that $x^{(a+b)} = x^a \bullet x^b$. Similarly, taking the logarithm of both sides of a function allows you to transform products into sums.

MLE for a binomial variable

Let's take a deeper mathematical investigation into the coin flipping example above.

In general, if you were to observe n flips, you would have observations $y_1, y_2, \ldots, y_n$.

In maximum likelihood estimation, you are looking to maximize the likelihood:

$L(p) = L(y_1, y_2, \ldots, y_n | p) = p^y (1-p)^{n-y}$ where $y = \sum_{i=1}^{n} y_i$

Taking the log of both sides:

$ln[L(p)] = ln[p^y (1-p)^{n-y}] = yln(p) + (n-y)ln(1-p)$

If $y = 1, 2, \ldots, n-1$ the derivative of $ln[L(p)]$ with respect to p is:

$\frac{d\, ln[L(p)]}{dp} = y(\frac{1}{p}) + (n-y)(\frac{-1}{1-p})$

As you've seen previously, the maximum will then occur when the derivative equals zero:

$0 = y(\frac{1}{p}) + (n-y)(\frac{-1}{1-p})$

Distributing, you have

$0 = \frac{y}{p} - \frac{n-y}{1-p}$

And solving for p:

$\frac{n-y}{1-p} = \frac{y}{p}$

$p(n-y) = \frac{y(1-p)}{p}$

$\frac{n-y}{y} = \frac{1-p}{p}$

$\frac{n}{y} - 1 = \frac{1}{p} - 1$

$\frac{n}{y} = \frac{1}{p}$

$p = \frac{y}{n}$

And voilà, you've verified the intuitive solution discussed above; the maximum likelihood for a binomial sample is the observed frequency!

Summary

In this lesson, you briefly reviewed maximum likelihood estimation. In the upcoming lesson, you'll see how logistic regression can also be interpreted from this framework, which will help set the stage for you to code a logistic regression function from scratch using NumPy. Continue on to the next lesson to take a look at how this works for logistic regression.

How do you feel about this lesson?

Have specific feedback?

Tell us here! (https://github.com/learn-co-curriculum/dsc-mle-review/issues/new/choose)