📈

MLE, MAP, etc.

This page gives a summary of various model parameterisation methods, beginning with maximum likelihood estimation (MLE), and extending to maximum a posteriori (MAP) and others.

Maximum likelihood estimation (MLE)

Most scientists and engineers are familiar with the idea of ‘least squares’ for curve fitting, but where does this actually come from?

This problem is very well known, and there are a variety of techniques for fitting parameters to data. MLE is a very common method of doing this, and has been in use for over a century.

Consider a simple static algebraic ‘model’ (really just a function), with input(s) x, output(s) y and parameters θ. We might write such a model as $y = f(x, \theta)$. The question then arises, given some measurements of x and y, and some assumption about the nature of the function $f()$, which parameters of the function ‘best’ fit the available data?

We can make this a bit more concrete by proposing some structure for the example model. Consider a simple linear-in-parameters model with $\theta = [c_0 \;\; c_1]^T$:

$$y = c_0 + c_1 x$$

This could be generalised to a very useful family of models (still linear-in-parameters), as

$$y = c^T \phi(x),$$

where c is a vector (or matrix for a multiple input system) of parameters and $\phi()$ is a vector of basis functions. We will, however, continue with the simple example for now. Our aim is to learn c_0 and c_1 from data. Consider making a single data observation of measured data (x_i, y_i) and we can write that

$$y_i = c_0 + c_1 x_i + \epsilon_i,$$

where ϵ_i is a term accounting for observation noise. It is very common to assume that this noise is normally distributed with zero mean, i.e.,

$$\epsilon \sim \mathcal{N}(0, \sigma^2)$$

where σ^2 is the variance of the observation noise. Combining the above equations we see that

$$y_i \sim \mathcal{N}(c^T x_i, \sigma^2).$$

The *likelihood* of each observation is the probability of the output given the input and some assumed parameters, and assuming a Gaussian likelihood (as above),

$$p(y_i|x_i, c, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{\left(y_i - c^T x_i\right)^2}{2\sigma^2}\right),$$

recalling that a univariate Gaussian is $f(x) = (1/(\sigma\sqrt{2\pi}))\exp\left(-\left(x-\mu\right)^2/2\sigma^2\right)$. We see that the mean model prediction is $c^T x_i$.

For all observations, we wish to find the parameter set c and the noise variance σ^2 by *maximising the total likelihood of the observations.* Assuming individual observations are independent (this is a big if, but could be modified later), the total likelihood is given by

$$\mathcal{L} = \prod p\left(y_i|x_i, c\right),$$

since independent probabilities multiply. Dealing with this product directly is tricky because we quickly run into machine precision issues. However, we can transform this with a monotonic function and still preserve the *position* of the maximum. Hence it is usual instead to *minimise the negative log likelihood*,

$$\min\ N\log\sigma\sqrt{(2\pi)} + \sum_{i=1}^{N} \frac{(y_i - c^T x_i)^2}{2\sigma^2}.$$

We now see where ‘least squares’ comes from, it is the numerator of the fraction in the sum. Note as an aside that if we had assumed a different probability distribution we would of course get a different answer. Assuming a Laplace distribution, rather than Gaussian, would lead to the equivalent of minimising the L_1 norm rather than the L_2 norm.

To solve the problem above, there are ‘direct solutions’ (by differentiation) but these are not numerically scalable, so it is usual to use an optimisation algorithm to first find c^T and then σ.

Maximum a posteriori (MAP) estimation

The ideas above can readily be extended into the Bayesian world. Note first a philosophical point: frequentist statistics (sadly this is what is mostly taught at school) considers *data to be uncertain and the parameters to the fixed*, and therefore does not ‘allow’ the idea of parameters having probability distributions - instead, ‘estimators’ are constructed for these. Bayesian statistics is the other way around (and lines up better with the way people think of probabilities, i.e., like betting odds or degree of belief, rather than number of outcomes in a repeated trial), in other words, *data are fixed and parameters are uncertain*. Because of this, it is natural in the Bayesian world to think of everything, including parameters, as having an uncertainty associated with it (we could even think of the choice of model itself as a probability distribution). We can use Bayes rule to link the parameter uncertainties to the likelihood,

$$p(\theta|x_i, y_i) = \frac{p(y_i|x_i, \theta)p(\theta)}{p(x_i, y_i)}.$$

The left hand side of this equation is called the *posterior*, the numerator on the right hand side is the product of the *likelihood* and the *prior* and the denominator is called the *marginal likelihood* which may be found by integrating (think of this like averaging) - which in the Bayesian world is called ‘marginalising’ - over θ. For now, we will ignore the marginal likelihood and note that

$$p(\theta|x_i, y_i) \,\alpha\, p(y_i|x_i, \theta)p(\theta).$$

We can improve the numerical conditioning of the estimation problem by finding the maximum of the posterior distribution instead of the maximum likelihood estimate. This requires us to define a *prior* distribution on the parameters, which is simply our best assumption of what they might be before we observe any data. It's usual to make this a wide distribution, such as a Gaussian with large variance, or a Gamma distribution if the parameter is always positive. (Some frequentist statisticians get uptight about priors, thinking they are subjective, but just think of the Bayesian approach as forcing you to be clear about your assumptions, and as we shall see, this is almost identical to *regularisation* - but more flexible; we could for example capture correlations between prior parameters.)

To find the MAP estimate of the parameters given some data, we now construct a cost function from the right hand side of the above equation. Assume a Gaussian prior with mean zero and variance τ^2. The cost function to minimise, after some rearranging, now looks like this:

$$\min \; N \log \sigma \sqrt{(2\pi)} + \sum_{i=1}^{N} \frac{(y_i - c^T x_i)^2}{2\sigma^2} + \frac{\sigma^2}{\tau^2} c^T c.$$

The new term on the right hand side improves the numerical conditioning, and is identical to ridge regression with regularisation controlled by $\lambda = \sigma^2/\tau^2$.

Further reading/sources

None of the above is original, but was instead inspired by:

- An excellent blog post by Katherine Bailey
- Bishop's book "Pattern recognition and machine learning" (see, e.g., chapter 3) which can also be downloaded for free as a pdf

In terms of battery applications, we have written a few papers thinking about MLE and Bayesian estimates of model parameters, for example:

- "Nonlinear Electrochemical Impedance Spectroscopy for Lithium-Ion Battery Model Parameterization" - has an extensive section on parameter estimation including supplementary information showing how to combine two different sets of observations in a single MLE cost function
- "Bayesian Model Selection of Lithium-Ion Battery Models via Bayesian Quadrature" - uses advanced/new methods to estimate parameters and compare models
- "Predicting battery end of life from solar off-grid system field data using machine learning" - uses a fully Bayesian parameterisation effort to parameterise very simple circuit models and therefore make predictions about battery life.
- "Bayesian Parameter Estimation Applied to the Li-ion Battery Single Particle Model with Electrolyte Dynamics" - estimates electrochemical model (SPMe) parameters from data using Monte Carlo methods to find the posterior parameter probabilities, and shows that this can be used to investigate identifiability.
- "Identifiability and parameter estimation of the single particle lithium-ion battery model" - not a Bayesian paper, but shows the issue with lack of identifiability of battery model parameters caused by lack of a reference electrode and flatness of OCV curves

There is also some nice work by Jamie Foster and colleagues on Bayesian parameterisation of battery models, and one might also think about the idea of sloppy models when thinking about parameters.