In this post, I will briefly review the basic theory about ordinary linear regression using frequentist and Bayesian estimation methods. This will lay the terrain for a later post about Gaussian Processes. This post closely follows the presentation in Murphy (2012).

# Basics of Linear Regression

Linear regression used to fit a model that is linear in the parameters to a data set , where represents the independent variables and the dependent variable, and has the form

where is a vector of model parameters and is the model error variance. The unusual notation for the normal distribution should be read as “a normal distribution of with (or given) mean and variance .” The traditional method of estimating the parameters of a linear regression model is by using the frequentist method of maximum likelihood estimation (MLE). MLE provides point estimates for each of the regression model parameters by maximizing the likelihood function with respect to and by solving the maximization problem

where the likelihood function is defined as

where is the mean of the model error ( hereafter), is the number of points in the data set, is an identity matrix, and is a vector of values of the independent variables for an individual data point of matrix . If a non-linear function shape is sought, linear regression can be used to fit a model linear in the parameters over a function of the data. This procedure is called basis function expansion and has a likelihood function of the form

where can have, for example, the form

for fitting a parabola to a one-dimensional over . When minimizing the squared residuals we get to the famous Ordinary Least Squares regression. Linear regression can be further developed, for example, into Ridge Regression to better handle multicollinearity by introducing bias to the parameter estimates, and into Kernel Ridge regression to implicitly add non-linear terms to the model. These formulations are beyond the scope of this post.

What is important to notice is that the standard approaches for linear regression described here, although able to fit linear and non-linear functions, do not provide much insight into the model errors. That is when Bayesian methods come to play.

# Bayesian Linear Regression

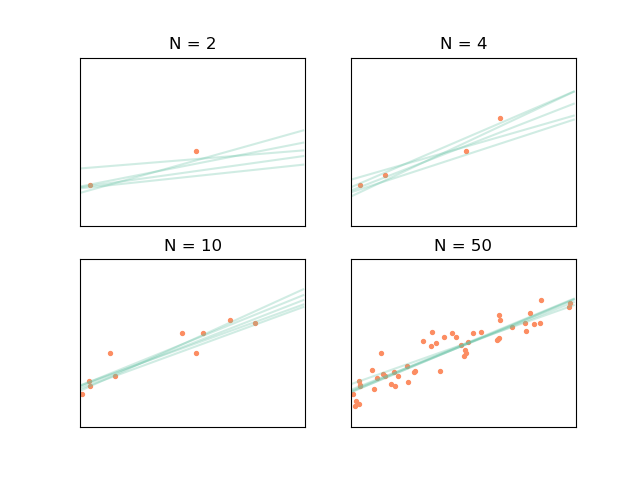
In Bayesian Linear regression (and in any Bayesian approach), the parameters are treated themselves as random variables. This allows for the consideration of model uncertainty, given that now instead of having the best point estimates for we have their full distributions from which to sample models – a sample of corresponds to a model. The distribution of parameters for , , called the parameter posterior distribution, is calculated by multiplying the likelihood function used in MLE by a prior distribution for the parameters. A prior distribution is assumed from knowledge prior to analyzing new data. In Bayesian Linear Regression, a Gaussian distribution is commonly assumed for the parameters. For example, the prior on can be for algebraic simplicity. The parameter posterior distribution assuming a known then has the form

, given is a constant depending only on the data.

We now have an expression from which to derive our parameter posterior for our linear model from which to sample . If the likelihood and the prior are Gaussian, the parameter posterior will also be a Gaussian, given by

where

If we calculate the parameter posterior (distribution over parameters ) for a simple linear model for a data set in which and are approximately linearly related given some noise , we can use it to sample values for . This is equivalent to sampling linear models for data set . As the number of data points increase, the variability of the sampled models should decrease, as in the figure below

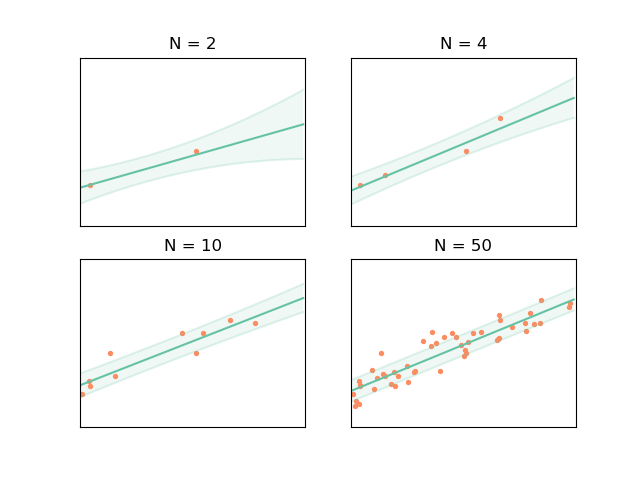
 [fig:linearmodel]

This is all interesting but the parameter posterior per se is not of much use. We can, however, use the parameter posterior to find the posterior predictive distribution, which can be used to both get point estimates of and the associated error. This is done by multiplying the likelihood by the parameter posterior and marginalizing the result over . This is equivalent to performing infinite sampling of blue lines in the example before to form density functions around a point estimate of , with denoting a new point that is not in . If the likelihood and the parameter posterior are Gaussian, the posterior predictive then takes the form below and will also be Gaussian (in Bayesian parlance, this means that the Gaussian distribution is conjugate to itself)!

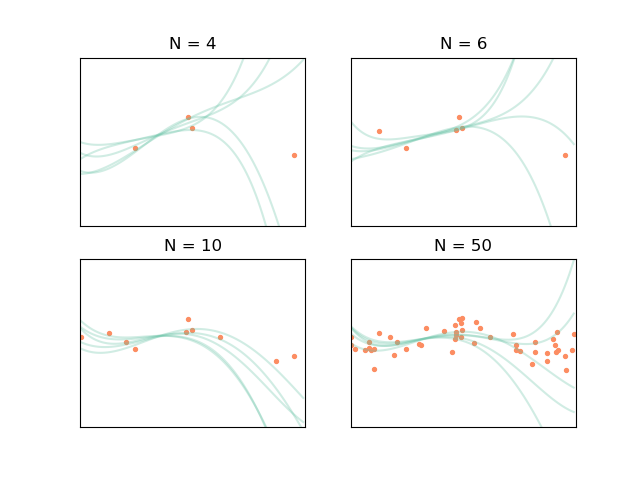
where

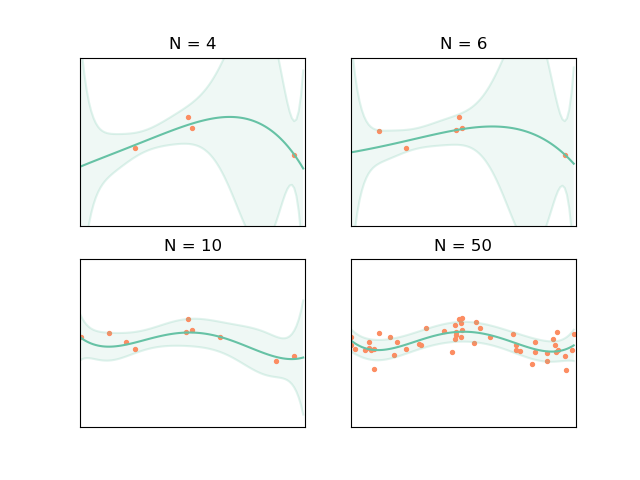
or, to put it simply,

The posterior predictive, meaning final linear model and associated errors (the parameter uncertainty equivalent of frequentist statistics confidence intervals) are shown below

 [fig:linearregression]

If, instead of having a data set in which and are related approximately according to a 4th order polynomial, we use a function to artificially create more random variables (or features, in machine learning parlance) corresponding to the non-linear terms of the polynomial function. Our function would be and the resulting would therefore have five columns instead of two (), so now the task is to find . Following the same logic as before for and , where prime denotes the new set random variable from function , we get the following plots





This looks great, but there is a problem: what if we do not know the functional form of the model we are supposed to fit (e.g. a simple linear function or a 4th order polynomial)? This is often the case, such as when modeling the reliability of a water reservoir system contingent on stored volumes, inflows and evaporation rates, or when modeling topography based on samples surveyed points (we do not have detailed elevation information about the terrain, e.g. a DEM file). Gaussian Processes (GPs) provide a way of going around this difficulty.

# References

Murphy, Kevin P., 2012. Machine Learning: A Probabilistic Perspective. The MIT Press.