# Exercise session 2 / 14

March 4, 2019

The objective of this exercise session are the following:

- Apply Bayesian statistics to a simple example: linear regression with Gaussian noise.

## 1 Linear regression

We now consider a complex dataset composed of $n$ pairs:

$$(\boldsymbol{x}_i, y_i)$$

where each $\boldsymbol{x}_i$ is a vector in $\mathbb{R}^d$ and $y_i$ is a scalar.

We will analyze these pairs with the classical Gaussian linear regression model. The principle of this model consists in modeling only the $Y$ variable, conditional on $X$ as:

$$Y_i = \langle \boldsymbol{x}_i, \boldsymbol{\theta} \rangle + \sigma\eta$$
$$\eta \overset{IID}{\sim} \mathcal{N}(0, 1)$$

which is more compactly written in matrix form:

$$\boldsymbol{Y} = \mathcal{X}\boldsymbol{\theta} + \sigma\boldsymbol{\eta}$$
$$\boldsymbol{\eta} \sim \mathcal{N}(0, I_n)$$

where $\mathcal{X}$ is the $(n, d)$ matrix which contains all $\boldsymbol{x}_i$ vectors stacked on top of one another and $\boldsymbol{Y}$ is a $n$ length vector.

We recall that the classical frequentist estimators are:

$$\hat{\boldsymbol{\theta}} = \left(\mathcal{X}^T\mathcal{X}\right)^{-1} \mathcal{X}^T \boldsymbol{y}$$
$$\hat{\boldsymbol{y}} = \mathcal{X}\hat{\boldsymbol{\theta}}$$
$$\sigma^2 \approx S^2 = \frac{1}{n-d} \left\| \boldsymbol{y} - \hat{\boldsymbol{y}} \right\|_2^2$$

We will now derive the properties of the Bayesian Gaussian linear regression model.

### 1.1 Mathematical assignments

**This section is optional** due to how mathematically intensive it is. We give the answers at the end.

We will first reparameterize the model with $\beta = \sigma^{-2}$.

Our prior will be a member of the Gamma-Multivariate Normal family:

$$f(\boldsymbol{\theta}, \beta) = \frac{b^a}{\Gamma(a)} \frac{|\boldsymbol{S}|^{1/2}}{(2\pi)^{d/2}} \beta^{a-1/2} \exp\left(-b\beta - \frac{1}{2}\beta (\boldsymbol{\theta} - \boldsymbol{\mu}_0)^T \boldsymbol{S} (\boldsymbol{\theta} - \boldsymbol{\mu}_0)\right)$$

which has four parameters $\boldsymbol{\mu}_0, \boldsymbol{S}, a, b$ where $\boldsymbol{\mu}_0$ is a length-$d$ vector and $\boldsymbol{S}$ is a $(d, d)$ matrix.

1. Prove that the posterior is also a member of the Gamma-Multivariate Normal family. Compute the updated values of the four parameters.
2. Derive the value of the normalization constant of the posterior: $f((\boldsymbol{x}_1, y_1) \ldots (\boldsymbol{x}_n, y_n))$.

The answers are:

$$\begin{aligned} \boldsymbol{\mu}_0 &\to \boldsymbol{\mu}_{post} \\ \boldsymbol{S} &\to \boldsymbol{S}_{post} = \boldsymbol{S} + \mathcal{X}^T \mathcal{X} \\ a &\to a_{post} = a + n/2 \\ b &\to b_{post} \end{aligned}$$

where:

$$\begin{aligned} \boldsymbol{\mu}_{post} &= \left[\boldsymbol{S} + \mathcal{X}^T \mathcal{X}\right]^{-1} \left[\boldsymbol{S}\boldsymbol{\mu}_0 + \sum_{i=1}^{n} y_i \boldsymbol{x}_i\right] \\ b_{post} &= b + \frac{1}{2} (\boldsymbol{\mu}_{post} - \boldsymbol{\mu}_0)^T \boldsymbol{S} (\boldsymbol{\mu}_{post} - \boldsymbol{\mu}_0) + \frac{1}{2} \sum_{i=1}^{n} (y_i - \boldsymbol{\mu}_{post} . \boldsymbol{x}_i)^2 \end{aligned}$$

and:

$$f(d) = \frac{1}{(2\pi)^{n/2}} \frac{b^a}{b_{post}^{a_{post}}} \frac{\Gamma(a_{post})}{\Gamma(a)} \frac{|\boldsymbol{S}|^{1/2}}{|\boldsymbol{S}_{post}|^{1/2}}$$

where $|\boldsymbol{S}|$ is the determinant of the matrix.

## 1.2 Programming assignments

1. Implement a function to generate a linear regression dataset.
   Your function should take an **optional argument** which enables the user to change the noise model.

2. Implement a function to compute the posterior distribution conditional on a given dataset.
   It should take as input the prior parameters and the dataset. By default, use the following values:

$$
\begin{aligned}
\boldsymbol{\mu}_0 &= 0 \\
\boldsymbol{S} &= 0.1 \ I_d \\
a &= 1 \\
b &= 1
\end{aligned}
$$

   which correspond to a "vague" prior.
   It should return the posterior parameters and the normalizing constant $f(d)$.

3. Implement a function to generate a non-linear regression dataset.
   It should take the following input:

   (a) Either a probability distribution or an array of samples for a predictor variable $t$.

   (b) A function $y(t)$.
   NB: python accepts function handles as inputs to another function.

   (c) A noise model.

   (d) A noise variance $\sigma^2$.

   And return samples from the model:

$$Y_i = y(t_i) + \sigma \eta_i$$

   where the $\eta_i$ are IID variables which obey the specified noise model.

4. Implement a function to perform polynomial regression.
   IE, it takes as input:

   (a) Examples of pairs $(t_i, y_i)$ that might represent a non-linear relationship.

   (b) A degree $d$.

   It then analyzes the data using a $d+1$ dimensional linear regression model:

$$Y \sim \sum_{j=0}^{d} \theta_j (t)^j + \sigma \eta$$

5. Implement a function which performs model comparison to select the appropriate degree $d$ for polynomial regression.
   It should take as input:

   (a) A prior distribution on the degrees $d$.
   One possibility consists in using a geometric prior:

   $$f(d) \propto \alpha^d$$

   with $\alpha \in ]0,1[$.

   (b) Examples of pairs $(t_i, y_i)$ that might represent a non-linear relationship.

   And return the posterior distribution over the degrees:

   $$f(d | (t_1, y_i) \dots (t_n, y_n))$$

   NB: do not evaluate the posterior over all values $d \in \mathbb{N}$ but instead stop once a cut-off is reached.
   Initially, set the cut-off at $d = 10$ but try to find a way to automatically select the cut-off point.

## 1.3 Experimental assignments

1. Test your linear regression function on various generated datasets

   (a) As you increase $n$, does the posterior mean (parameter $\boldsymbol{\mu}_{post}$) converge to the correct value?

2. Test your polynomial regression function on various generated datasets.

   (a) As you vary $n$, does the posterior over the degree recover the true degree of a truely polynomial relationship?

   (b) As you vary $\alpha$, how does the posterior change when computed for the same dataset?

   (c) Generate an example where the true relationship isn't polynomial (e.g. $t \mapsto |t|$). How does the posterior over $d$ behave as you vary $n$?

## 1.4 Advanced assignments

There is a very important practical step of analyzing a linear regression dataset. This consists in "standardizing" each predictor variable $X_j$ $j \in [1, d]$ and the target variable $Y$ in the dataset.

In practice, for every variable $X_j$ in the dataset:

1. We compute an estimator of central tendency, most often the empirical mean $\bar{x}_j$.
2. We compute an estimator of scale, most often the empirical standard deviation (the square-root of the empirical variance $s^2$).
3. We transform the original variable by centering and scaling it:

$$\tilde{X}_j = \frac{X_j - Center}{Scale}$$

There are multiple ways to justify this. I believe the most elegant to be that we want our statistical analysis to be invariant under parameterizations of the problem. Namely, if we measure a variable $X_j$ in millimeters or meters or feet, the statistical inference should be constant. Standardizing the variables as discussed ensures this.

1. Implement a function to standardize a dataset using the empirical mean and variance of each variable.
   Your function should return the transformed dataset as well as two functions:
   (a) One to translate from the original space to standardized space.
   (b) One to translate from standardized space to the original space.
2. Implement a function to standardize a dataset using two other estimators of central tendency and scale.
3. Compare the posterior distribution computed on the original dataset to the posterior distribution computed on the standardized dataset in various examples.