TNM Exercise 1: Bayesian Inference

Saurabh Vaishampayan (svaishampaya@student.ethz.ch)
Linus Rüttimann (rlinus@student.ethz.ch)
Elia Torre (etorre@student.ethz.ch) Group **H**
Timothy Kurer (tkurer@student.ethz.ch) 11.03.2023

1.1 Maximum Likelihood and Overfitting

Given the polynomial model of order P: $y = \theta_0 + \theta_1 x + \theta_2 x^2 + ... + \theta_P x^P + \varepsilon$ where ε denotes an iid. Gaussian noise term with zero mean and variance σ^2 and N independent observations

a Log Likelihood

We can rewrite the polynomial:

$$y = \theta_0 + \theta_1 x + \theta_2 x^2 + ... + \theta_P x^P + \varepsilon$$

in matrix form as:

$$y = \underbrace{\begin{bmatrix}1 & x & x^2 & ... & x^P\end{bmatrix}^\top}_{\Phi^\top(x)} \begin{bmatrix}\theta_0 \\ \theta_1 \\ \theta_2 \\ \vdots \\ \theta_P\end{bmatrix} + \varepsilon$$

For n datapoints we stack the vectors:

$$\underbrace{\begin{bmatrix}y_1 \\ y_2 \\ \vdots \\ y_n\end{bmatrix}}_{y} = \underbrace{\begin{bmatrix}1 & x_1 & x_1^2 & ... & x_1^P \\ 1 & x_1 & x_1^2 & ... & x_1^P \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_1 & x_1^2 & ... & x_1^P\end{bmatrix}}_{\Phi} \underbrace{\begin{bmatrix}\theta_0 \\ \theta_1 \\ \vdots \\ \theta_P\end{bmatrix}}_{\theta} + \underbrace{\begin{bmatrix}\varepsilon_0 \\ \varepsilon_1 \\ \vdots \\ \varepsilon_P\end{bmatrix}}_{\varepsilon}$$

Which can be rewritten as:

$$y = \Phi\theta + \varepsilon, \quad \varepsilon \sim \mathcal{N}(\varepsilon, 0, \sigma^2 I)$$

From which we obtain the following likelihood distribution:

$$p(Y|\theta) \sim \mathcal{N}(\varepsilon, 0, \sigma^2 I) = p(y_1, ... , y_n|\theta_0, \theta_1, ... , \theta_P)$$

Then, we derive the log-likelihood:

$$\text{Log Likelihood} = \log(p(Y|\theta)) = \log\left(\frac{1}{(2\pi)^{\frac{n}{2}}\sqrt{det(\sigma^2 I)}} \exp(-\frac{1}{2}(y-\Phi\theta)^\top(\sigma^2 I)^{-1}(y-\Phi\theta))\right)$$

$$= \underbrace{\frac{1}{2\sigma^2}\|y-\Phi\theta\|^2}_{function\ of\ \theta} - \underbrace{\frac{n}{2}\log(2\pi) - \frac{n}{2}log(\sigma^2)}_{constants}$$

b Maximum Log Likelihood

Here, we want to derive the MLE for θ:

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \frac{1}{2\sigma^2}\|y - \Phi\theta\|^2 = \underset{\theta}{\operatorname{argmin}} \|y - \Phi\theta\|^2$$

We can then rewrite as follows:

$$f(\theta) = (y - \Phi\theta)^\top (y - \Phi\theta) = y^\top y - y^\top \Phi\theta - \theta^\top \Phi^\top y + \theta^\top \Phi^\top \Phi\theta$$

$f(\theta)$ is a convex function. So, by setting the gradient of the function to zero, we obtain the global minima.

$$\nabla f(\theta) = 0 \Rightarrow 2\left(-y^\top \Phi + \theta^\top \Phi^\top \Phi\right)^\top = 0$$

Therefore:

$$\Phi^\top \Phi\theta = \Phi^\top y$$

$$\theta = \left(\Phi^\top \Phi\right)^{-1} \Phi^\top y$$

This holds for all $y = \Phi^\top(x)\theta + \varepsilon$ type models.

c Generate Data

The first dataset has been generated using the following code:

```
sigma = sqrt(0.001);
theta_gt = [0.3 -0.1 0.5]; %defining theta parameters

xdata1 = transpose(-0.5:0.1:0.2); %defining x for tasks 1.1 (c)
[ydata1] = generate_data(xdata1, sigma, theta_gt); %generating y observations for tasks 1.1 (c)
```

```
function [ydata] = generate_data(xdata, sigma_noise, theta_gt)
    P = length(theta_gt)-1; % Polynomial degree definition
    ydata = zeros(size(xdata));
    for i=0:P
        ydata = ydata+theta_gt(i+1)*(xdata.^(i)); % y observations generation
    end

    ydata = ydata+sigma_noise*randn(size(xdata)); % Adding noise to y observations
end
```

d Estimate ML

- What do you notice when comparing the ML parameter estimates to their true values?
  - The absolute value of difference between ground truth and ML estimate $\|\theta - \theta^*\|$ is smallest for $P = 2$ (when it matches the model from which the data was generated). It is greater for $P = 1$ because of bias and also greater for $P = 7$ because of variance.
- What do you notice about the value of the log-likelihood function at the ML solution for different values of P?
  - The value of the log-likelihood increases for increasing P, even if the estimate becomes worse for $P > 2$ because of over-fitting.

e Log Likelihood with new data

We generated this new dataset of observations with the same code as before, only changing the x data as follows:

```
sigma = sqrt(0.001);
theta_gt = [0.3 -0.1 0.5]; %defining theta parameters

xdata2 = transpose(-0.5:0.01:0.2); %defining x for tasks 1.1 (e)
[ydata2] = generate_data(xdata2, sigma, theta_gt); %generating y observations for tasks 1.1 (e)
```

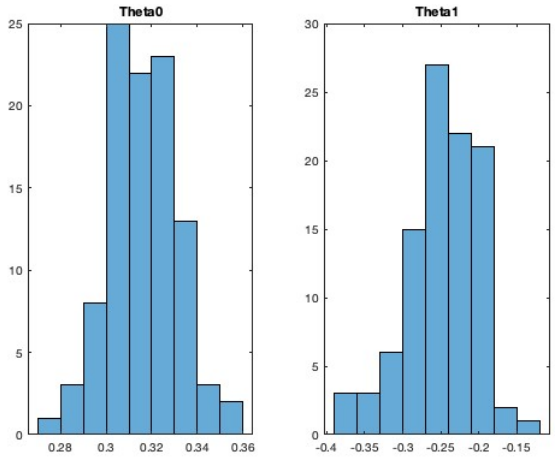
```
function [ydata] = generate_data(xdata, sigma_noise, theta_gt)
    P = length(theta_gt)-1; % Polynomial degree definition
    ydata = zeros(size(xdata));
    for i=0:P
        ydata = ydata+theta_gt(i+1)*(xdata.^(i)); % y observations generation
    end

    ydata = ydata+sigma_noise*randn(size(xdata)); % Adding noise to y observations
end
```

- What do you observe now?
  - The model is still over-fitting for P=7, in that it estimates values $\|\theta_i^*\| \gg 0$ for $i > 2$, but a bit less than before because of more data samples.

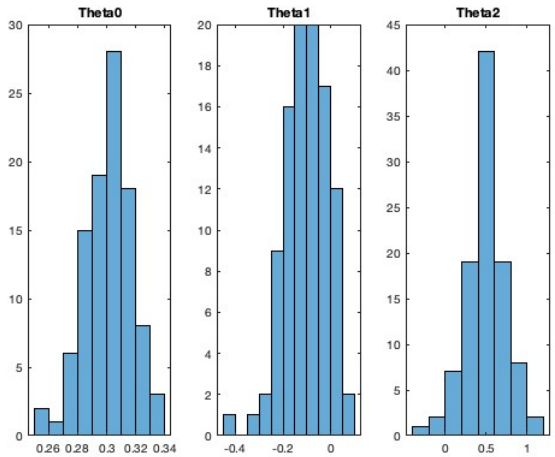
f ML Histograms

Here, we performed 100 iterations of the data generation and MLE Estimation and we obtain the following histograms:

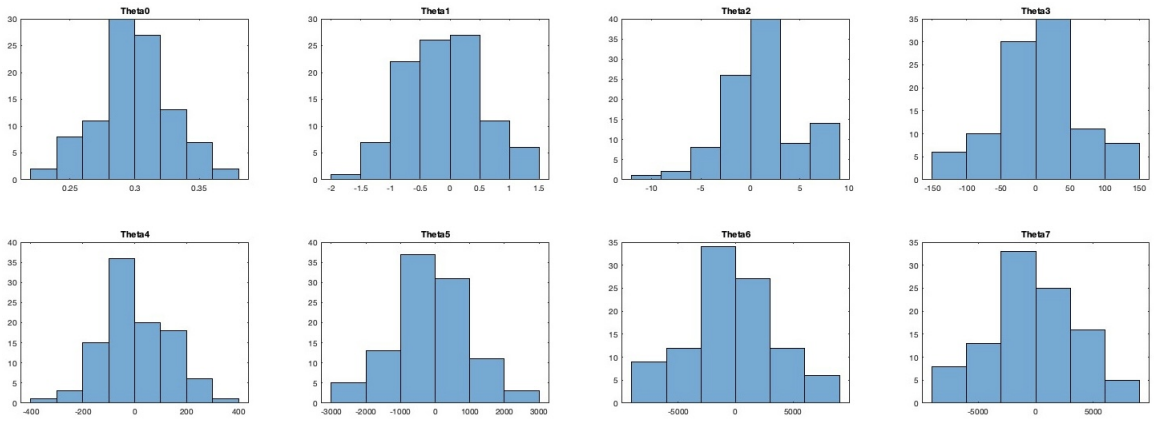
For P=1:



For P=2:



For P=7:



- What do you notice about the consistency of the ML parameter estimates across repetitions?
  - The consistency is small and and the variance of the parameter estimates increases for increasing P.

1.2 Maximum-A-Posteriori Estimation

a Log Posterior Distribution

Here, we want to derive the log-posterior distribution $\log_p(\theta \mid y)$:

$$\log_p(\theta \mid y) = \frac{p(\theta)p(y \mid \theta)}{p(y)}$$

Where $P(y)$ is the evidence, a multiplicative constant independent of θ.

$$\log p(\theta \mid y) = \log p(y \mid \theta) + \log p(\theta) + \text{ const}$$

Where

$$p(\theta) \sim \mathcal{N}(\theta; 0, I) \text{ as given.}$$

Therefore:

$$\log p(\theta) = \log \frac{1}{(\sqrt{2\pi})^n} e^{-\frac{\|-\theta\|^2}{2}} = -\frac{1}{2}\theta^\top\theta - \frac{n}{2}\log 2\pi.$$

Hence, we have that:

$$\log p(\theta \mid y) = -\frac{1}{2}\theta^\top\theta - \frac{n}{2}\log 2\pi - \frac{1}{2\sigma^2}\|y - \Phi\theta\|^2 - \frac{n}{2}\log 2\pi - \frac{n}{2}\log\sigma^2 - \log p(y)$$

b MAP θ^* Estimate

Here, we are going to find the MAP θ^* estimate:

$$\theta^* = \underset{\theta}{\text{argmin}}\ \text{`} - \log p(\theta \mid y) = \underset{\theta}{\text{argmin}}\ \frac{1}{2}\theta^\top\theta + \frac{1}{20^2}(y - \Phi\theta)^\top(y - \Phi\theta)$$

This is again a convex function.
Thus, setting the gradient of the function to zero gives us:

$$\left(\theta^{*\top} + \frac{1}{\sigma^2}\left(\theta^{*\top}\Phi^\top\Phi - y^\top\Phi\right)\right)^\top = 0$$

From which:

$$\left(\frac{1}{\sigma^2}\Phi^\top\Phi + I\right)\theta^* = \frac{1}{\sigma^2}\Phi^\top y$$

Therefore:

$$\theta^* = \left(\Phi^\top \Phi + \sigma^2 I\right)^{-1} \Phi^\top y$$

c Applied MAP

Here we used the same generating functions shown in 1.1 (c).

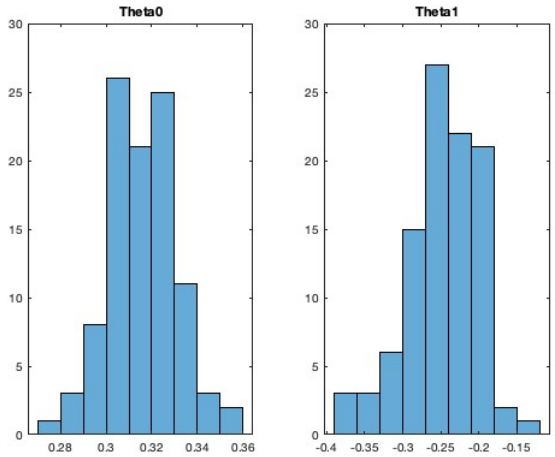
- What do you notice when comparing the MAP estimates to the ML estimates from exercise 1.1 (d) and the true values of the parameters in exercise 1.1 (c)?

  – For $P = 1$ and $P = 2$ the ML and MAP estimator perform similar, but for $P = 7$ the shrinkage prior keeps the estimates $\|\theta_i^*\|$ for $i > 2$ small, which results in less over-fitting.

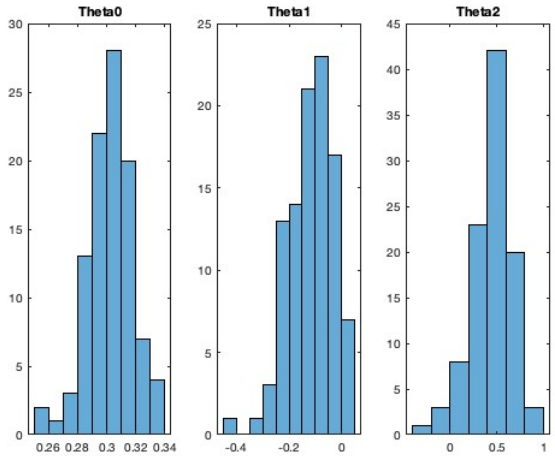
d MAP Histogram

Here, we performed 100 iterations of the data generation and MAP Estimation and we obtain the following histograms:

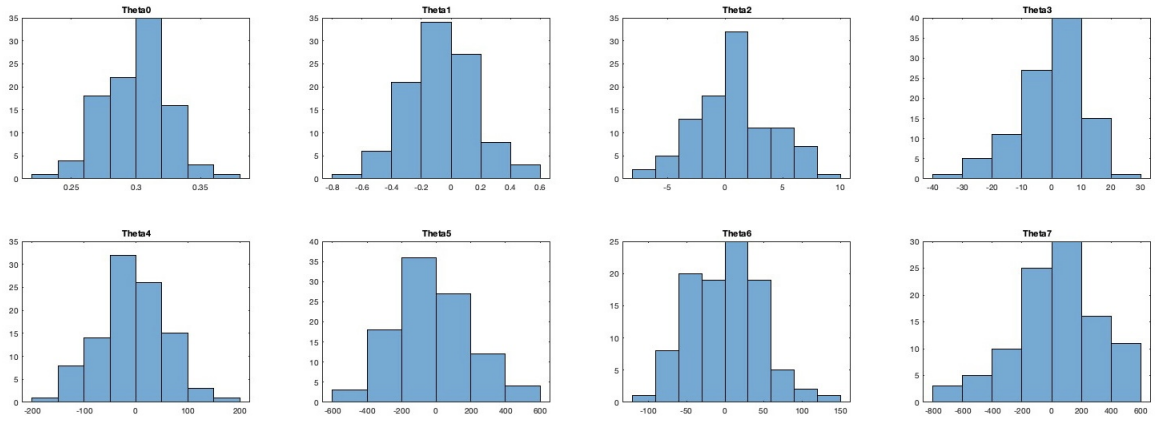
For P=1:



For P=2:



For P=7:



- Do the histograms look different than those from exercise 1.1 (f)?
  - The consistency is better than in the histograms of exercise 1.1 (f), in that the variances of the histograms are smaller.

1.3 Bayesian Inference in the Univariate Gaussian Case

a Likelihood

Starting with the following univariate Gaussian model:

$$y = x\theta + \varepsilon$$

And the following Gaussian noise term and prior:

$$p(\varepsilon) \sim \mathcal{N}\left(\varepsilon; 0, \sigma_\varepsilon^2\right)$$

$$p(\theta) \sim \mathcal{N}\left(\theta; \mu_p, \sigma_p^2\right)$$

We have:

$$\underbrace{p(y \mid \theta)}_{\text{Likelihood}} \sim \mathcal{N}\left(y; x\theta, \sigma_\varepsilon^2\right) = \frac{1}{\sqrt{2\pi}\sigma_\varepsilon} exp(-\frac{1}{2\sigma_\varepsilon^2}(y - x\theta)^2)$$

b Calculate Posterior

In order to calculate the posterior, we start from the Bayes Theorem:

$$p(\theta|y) = \frac{p(\theta)p(y \mid \theta)}{p(y)}$$

Since $p(\theta)$ is Gaussian, $p(y \mid \theta)$ is Gaussian therefore $p(\theta \mid y)$ is Gaussian.

Therefore:

$$p(\theta \mid y) = \frac{1}{\sqrt{2\pi}\sigma_{\theta|y}} e^{-\frac{1}{2}\frac{(\theta-\mu_{\theta|y})^2}{\sigma_{\theta|y}^2}} \propto e^{-\frac{1}{2}\frac{(y-x\theta)^2}{\sigma_\varepsilon^2}} e^{-\frac{1}{2}\frac{(\theta-\mu_p)^2}{\sigma_p^2}}$$

Since it is a Gaussian, we can recover $\mu_{\theta|y}$ and $\sigma_{\theta|y}^2$ by comparing terms in the exponent, which is a quadratic.

It is easier to compare after taking the negative log on both sides:

LHS:

$$\frac{1}{2}\frac{\left(\theta - \mu_{\theta|y}\right)^2}{\sigma_{\theta|y}^2} = \frac{1}{2}\left(\frac{\theta^2}{\sigma_{\theta|y}^2} - \frac{\theta\mu_{\theta|y}}{\sigma_{\theta|y}^2} + \frac{\mu_{\theta|y}^2}{\sigma_{\theta|y}^2}\right)$$

And RHS:

$$\frac{1}{2}\frac{(y-x\theta)^2}{\sigma_\varepsilon^2}+\frac{1}{2}\frac{(\theta-\mu_p)^2}{\sigma_p^2}=\frac{1}{2}\left[\frac{\theta^2}{\sigma_p^2}+\theta^2\frac{x^2}{\sigma_\varepsilon^2}-\theta\left(\frac{\mu_p}{\sigma_p^2}+\frac{yx}{\sigma_\varepsilon^2}\right)\right]+\frac{1}{2}\frac{y^2}{\sigma_\varepsilon^2}+\frac{1}{2}\frac{\mu_p^2}{\sigma_p^2}$$

The Coefficient of θ^2 gives us $\frac{1}{2}\frac{1}{\sigma^2_{\theta|y}}$

$$\sigma^2_{\theta|y}=\left(\frac{1}{\sigma_p^2}+\frac{x^2}{\sigma_\varepsilon^2}\right)^{-1}$$

The Coefficient of θ gives us $-\frac{1}{2}\frac{\mu_{\theta|y}}{\sigma^2_{\theta|y}}$

Therefore:

$$\frac{\mu_{\theta|y}}{\sigma^2_{\theta|y}}=\left(\frac{\mu_p}{\sigma_{p^2}}+\frac{yx}{\sigma_\varepsilon^2}\right)$$

Hence:

$$\mu_{\theta|y}=\sigma^2_{\theta|y}\left(\frac{\mu_p}{\sigma_p^2}+\frac{yx}{\sigma_\varepsilon^2}\right)$$

$$\mu_{\theta|y}=\frac{\left(\frac{\mu_p}{\sigma_p^2}+\frac{xy}{\sigma_\varepsilon^2}\right)}{\frac{1}{\sigma_p^2}+\frac{x^2}{\sigma_\varepsilon^2}}$$

Finally:

$$\boxed{P(\theta\mid y)\sim\mathcal{N}\left(\theta;\mu_{\theta|y},\sigma^2_{\theta|y}\right)}$$