Problem Sheet 3

```julia
begin
    using Pkg
    Pkg.activate(".")
    Pkg.add(["Distributions", "Plots", "PlutoUI"])
    using Distributions
    using LinearAlgebra
    using Plots
    using PlutoUI
    default(linewidth = 3.0, legendfontsize= 15.0)
end
```

Table of Contents

1. Bayes inference for the variance of a Gaussian

Use a Bayesian approach to estimate the inverse variance λ of a univariate Gaussian distribution

$$p(x|\lambda) = \sqrt{\frac{\lambda}{2\pi}}\ \exp\left[-\frac{\lambda x^2}{2}\right].$$

Here we have assumed for simplicity that the data has zero mean $\mu = 0$. To apply Bayesian inference we specify a **Gamma** prior distribution for λ,

$$p(\lambda) = \frac{\lambda^{\alpha-1}\exp[-\lambda/\beta]}{\Gamma(\alpha)\beta^{\alpha}}$$

where the positive numbers α and β, the \emph{hyperparameters} of the model are assumed to be known and $\Gamma(\alpha)$ is Euler's **gamma** function. We then observe a dataset $D = (x_1, x_2, \ldots, x_N)$ comprising N independent random samples from $p(x|\lambda)$.

(a) [MATH] Show that the posterior probability $p(\lambda|D)$ of the inverse variance is also a Gamma distribution with parameters

$$\alpha_p = \alpha + \frac{N}{2}, \qquad \frac{1}{\beta_p} = \frac{1}{\beta} + \frac{1}{2}\sum_{i=1}^{N} x_i^2.$$

Solution

- Likelihood of the data set

$$p(D|\lambda) = \prod_{i=1}^{N}\sqrt{\frac{\lambda}{2\pi}}\exp\left(-\frac{\lambda}{2}x_i^2\right)$$

- Joint distribution for D and λ

$$\begin{aligned}
p(D,\lambda) =& p(D|\lambda)p(\lambda) \\
=& \frac{\lambda^{\alpha-1}\exp(-\lambda\beta^{-1})}{\Gamma(\alpha)\beta^{\alpha}}\prod_{i=1}^{N}\sqrt{\frac{\lambda}{2\pi}}\exp\left(-\frac{\lambda}{2}x_i^2\right) \\
=& \frac{\lambda^{(\alpha+N/2)-1}}{(2\pi)^{N/2}\Gamma(\alpha)\beta^{\alpha}}\exp\left[-\lambda\left(\frac{1}{\beta}+\frac{1}{2}\sum_{i=1}^{N}x_i^2\right)\right] \\
=& \frac{\lambda^{\alpha_p-1}\exp(-\lambda\beta_p^{-1})}{(2\pi)^{N/2}\Gamma(\alpha)\beta^{\alpha}} \\
=& \frac{\Gamma(\alpha_p)\beta_p^{\alpha_p}}{(2\pi)^{N/2}\Gamma(\alpha)\beta^{\alpha}}\frac{\lambda^{\alpha_p-1}\exp(-\lambda\beta_p^{-1})}{\Gamma(\alpha_p)\beta_p^{\alpha_p}}
\end{aligned}$$

- Posterior for λ

$$p(\lambda|D) = \frac{\lambda^{\alpha_p-1}\exp(-\lambda\beta_p^{-1})}{\Gamma(\alpha_p)\beta_p^{\alpha_p}}$$

(b) [MATH] Compute the mean of the posterior distribution of λ. Compare the result with the result

from the maximum-likelihood estimation, $\lambda_{\mathrm{ML}} = 1/\sigma^2_{\mathrm{ML}}$ and explain what happens if $N \to \infty$.

Solution

- Mean of the posterior distribution:

$$\begin{aligned}
\langle\lambda_p\rangle &= \int_0^\infty \lambda\, p(\lambda|D)\, d\lambda \\
&= \int_0^\infty \lambda\, \frac{\lambda^{\alpha_p-1}\exp(-\lambda\beta_p^{-1})}{\Gamma(\alpha_p)\beta_p^{\alpha_p}}\, d\lambda \\
&= \frac{1}{\Gamma(\alpha_p)\beta_p^{\alpha_p}} \int_0^\infty \lambda^{\alpha_p}\, e^{-\lambda/\beta_p}\, d\lambda \\
&= \frac{\beta_p}{\Gamma(\alpha_p)} \int_0^\infty z^{\alpha_p}\, e^{-z}\, dz \\
&= \frac{\Gamma(\alpha_p+1)}{\Gamma(\alpha_p)}\, \beta_p \\
&= \alpha_p\beta_p \\
&= \left(\alpha + \frac{N}{2}\right)\left(\frac{1}{\beta} + \frac{1}{2}\sum_{i=1}^N x_i^2\right)^{-1}
\end{aligned}$$

- Negative logarithm of the likelihood

$$\mathcal{L} = -\log p(D|\lambda) = \frac{\lambda}{2}\sum_{i=1}^N x_i^2 - \frac{N}{2}\log\lambda + \frac{N}{2}\log(2\pi)$$

- Maximum likelihood estimate

$$\frac{d\mathcal{L}}{d\lambda} = 0 \quad\Longleftrightarrow\quad \frac{1}{2}\sum_{i=1}^N x_i^2 - \frac{N}{2\lambda} = 0 \quad\Longleftrightarrow\quad \lambda_{\mathrm{ML}} = \left(\frac{1}{N}\sum_{i=1}^N x_i^2\right)^{-1}$$

- For $N \to \infty$ the posterior mean $\langle\lambda_p\rangle$ approaches the maximum likelihood estimate λ_{ML} asymptotically:

$$\langle\lambda_p\rangle = \frac{\alpha + N/2}{\beta^{-1} + N/2\,\lambda_{\mathrm{ML}}^{-1}} \quad\Longrightarrow\quad \lim_{N\to\infty}\langle\lambda_p\rangle = \lambda_{\mathrm{ML}}$$

(c) [MATH] Show that the variance of the posterior distribution

$$V[\lambda_{\mathrm{post}}] = \langle\lambda^2\rangle - \langle\lambda\rangle^2$$

shrinks to zero as $N \to \infty$. Here we have used the notation $\langle\cdot\rangle$ for posterior expectations.

Solution

- Variance of the posterior distribution

$$
\begin{aligned}
\langle\lambda_p^2\rangle - \langle\lambda_p\rangle^2 &= \int_0^\infty \lambda^2\, p(\lambda|D)\, d\lambda - \alpha_p^2\,\beta_p^2 \\
&= \int_0^\infty \lambda^2\, \frac{\lambda^{\alpha_p-1}\exp(-\lambda\beta_p^{-1})}{\Gamma(\alpha_p)\beta_p^{\alpha_p}}\, d\lambda - \alpha_p^2\,\beta_p^2 \\
&= \frac{1}{\Gamma(\alpha_p)\beta_p^{\alpha_p}} \int_0^\infty \lambda^{\alpha_p+1}\, e^{-\lambda/\beta_p}\, d\lambda - \alpha_p^2\,\beta_p^2 \\
&= \frac{\beta_p^2}{\Gamma(\alpha_p)} \int_0^\infty z^{\alpha_p+1}\, e^{-z}\, dz - \alpha_p^2\,\beta_p^2 \\
&= \frac{\Gamma(\alpha_p+2)}{\Gamma(\alpha_p)}\, \beta_p^2 - \alpha_p^2\,\beta_p^2 \\
&= \alpha_p\,(\alpha_p+1)\,\beta_p^2 - \alpha_p^2\,\beta_p^2 \\
&= \alpha_p\,\beta_p^2 \\
&= \left(\alpha + \frac{N}{2}\right)\left(\frac{1}{\beta} + \frac{1}{2}\sum_{i=1}^N x_i^2\right)^{-2}
\end{aligned}
$$

- Asymptotic behaviour for $N \to \infty$

$$
\lim_{N\to\infty} \mathrm{Var}(\lambda_p) \;=\; \lim_{N\to\infty} \frac{1}{N}\,\frac{\alpha/N + 1/2}{(\beta^{-1}/N + \lambda_{\mathrm{ML}}^{-1}/2)^2} \;=\; 2\lambda_{\mathrm{ML}}^2 \lim_{N\to\infty}\frac{1}{N} \;=\; 0
$$

(d) [MATH] Show that the predictive distribution is

$$
p(x|D) = \frac{1}{\sqrt{2\pi}}\,\frac{\Gamma(\alpha_p+1/2)}{\Gamma(\alpha_p)}\sqrt{\beta_p}\left(1 + \frac{x^2\beta_p}{2}\right)^{-\alpha_p-1/2}
$$

where α_p and β_p were defined above. Note, this is not a Gaussian!

Solution

$$
\begin{aligned}
p(x|D) &= \int_0^\infty p(x|\lambda)p(\lambda|D)d\lambda \\
&= \int_0^\infty \sqrt{\frac{\lambda}{2\pi}} \exp\left(-\frac{\lambda}{2}x^2\right) \frac{\lambda^{\alpha_p-1}\exp(-\lambda\beta_p^{-1})}{\Gamma(\alpha_p)\beta_p^{\alpha_p}} d\lambda \\
&= \frac{1}{\sqrt{2\pi}\,\Gamma(\alpha_p)\beta_p^{\alpha_p}} \int_0^\infty \lambda^{\alpha_p-1/2} \exp\left[-\lambda\left(\frac{1}{\beta_p}+\frac{1}{2}x^2\right)\right] d\lambda \\
&= \frac{1}{\sqrt{2\pi}} \frac{\Gamma(\alpha_p+1/2)}{\Gamma(\alpha_p)} \frac{1}{\beta_p^{\alpha_p}} \left(\frac{1}{\beta_p}+\frac{1}{2}x^2\right)^{-\alpha_p-1/2} \\
&= \frac{1}{\sqrt{2\pi}} \frac{\Gamma(\alpha_p+1/2)}{\Gamma(\alpha_p)} \sqrt{\beta_p}\left(1+\frac{x^2\,\beta_p}{2}\right)^{-\alpha_p-1/2}
\end{aligned}
$$

For all conjugate priors used in Bayesian analysis of the Gaussian distribution (including Normal, Gamma-Normal, Wishart etc...), see this review from Kevin Murphy : **Conjugate Bayesian analysis of the Gaussian distribution**

(e) [CODE] Program a function generating samples from a known value λ and compute both the posterior distribution and ML estimator by adding progressively new samples.

```
gamma_params = (α=2.0, β=0.5);
```

```
true_λ = 0.5163889780582498
```

```
true_λ = rand(Gamma(gamma_params...)) # We sample a random value λ
```

```
generate_x(λ) = rand(Normal(0, 1 / sqrt(λ))); # Generate a random value x from a
give λ
```

```
αp(x, α) = α + length(x) / 2;
```

```
βp(x, β) = inv(inv(β) + 0.5 * sum(abs2, x));
```

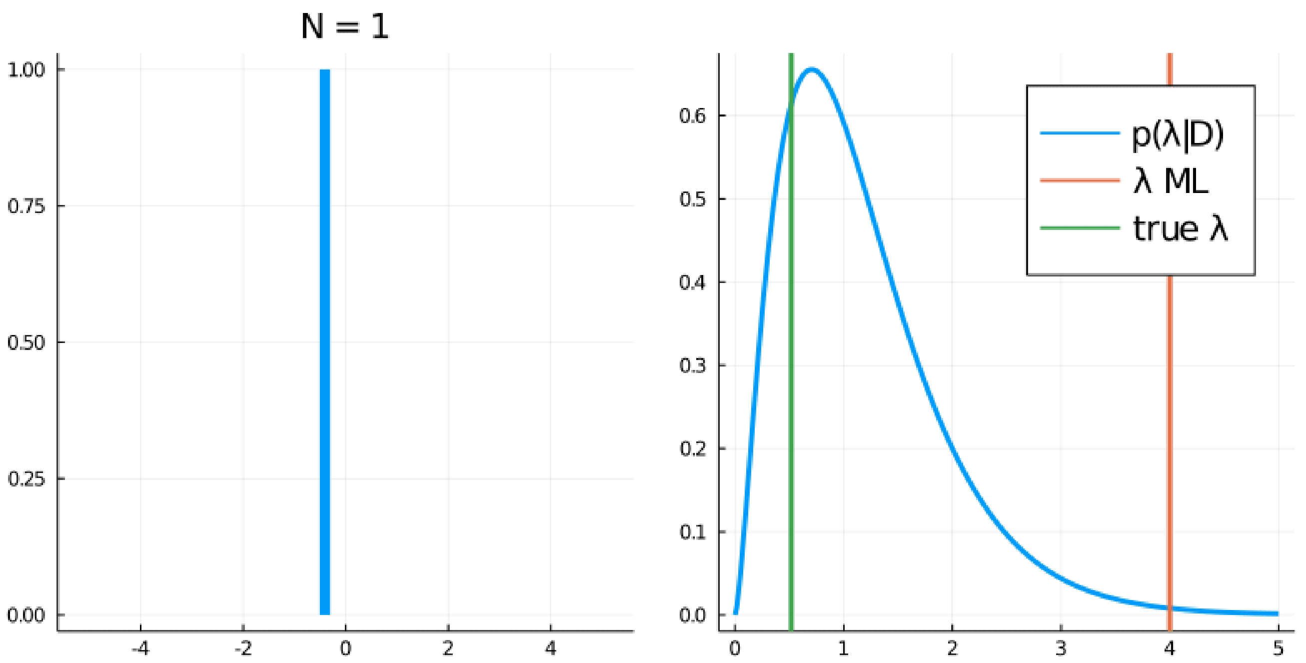
```
posterior_λ(x, α, β) = Gamma(αp(x, α), βp(x, β)); # Posterior distribution
```

```
λ_ML(x) = inv(sum(abs2, x) / length(x)); # ML estimator
```

```
begin  # Plotting values
    N = 10000
    bins = range(-5, 5, length = 50)
    range_λ = range(0, 5.0, length = 300)
end;
```

```
begin
    xs = Float64[]
    anim = Animation()
    for i in 1:N
        push!(xs, generate_x(true_λ))
```

```julia
        if i % 10^floor(Int64, log10(i)) == 0
            p_x = histogram(xs, bins = bins, lw = 0.0, label = "", title = "N = $i")
            p_λ = plot(range_λ, x -> pdf(posterior_λ(xs, gamma_params...), x), label
= "p(λ|D)")
            vline!([λ_ML(xs)], label = "λ ML")
            vline!([true_λ], label = "true λ")
            plot(p_x, p_λ, size = (800,400))
            frame(anim)
        end
    end
end
```




```
Distributions.Gamma{Float64}(α=5002.0, θ=0.00010392488297250539)
```

```julia
posterior_λ(xs, gamma_params...)
```

2. Hyperparameter estimation for a generalised linear model

Consider a model for a set of data $D = (y_1, \ldots, y_n)$ defined by

$$p(D|\mathbf{w}, \beta) = \left(\frac{\beta}{2\pi}\right)^{N/2} \exp\left[-\sum_{i=1}^{N} \frac{\beta}{2}\left(y_i - \sum_{j=1}^{K} w_j \Phi_j(x_i)\right)^2\right]$$

with a fixed set $\{\Phi_1(x), \ldots, \Phi_k(x)\}$ of K basis functions. The prior distribution on the weights is given by

$$p(\mathbf{w}|\alpha) = \left(\frac{\alpha}{2\pi}\right)^{K/2} \exp\left[-\frac{\alpha}{2}\sum_{j=1}^{K} w_j^2\right].$$

This **generalised linear model** assumes that the observations are generated from a weighted linear combination of the basis functions with additive Gaussian noise.

- **(a) [MATH] The posterior distribution $p(\mathbf{w}|D)$ of the vector of weights is a Gaussian. Compute the posterior mean vector $E[\mathbf{w}]$ and the posterior covariance in terms of the matrix $\mathbf{X}$ where $X_{lk} = \Phi_k(x_l)$.**

Solution

- Joint distribution in matrix notation

$$\begin{aligned}&p(D,\mathbf{w}|\alpha,\beta)\\=&\left(\frac{\alpha}{2\pi}\right)^{\frac{K}{2}}\left(\frac{\beta}{2\pi}\right)^{\frac{N}{2}}\exp\left(-\frac{\alpha}{2}\mathbf{w}^\top\mathbf{w}-\frac{\beta}{2}(\mathbf{y}-\mathbf{X}\mathbf{w})^\top(\mathbf{y}-\mathbf{X}\mathbf{w})\right)\end{aligned}$$

- Mean value of the posterior (see **Fisher information**)

$$\begin{aligned}\frac{\partial\log p(D,\mathbf{w}|\alpha,\beta)}{\partial\mathbf{w}}\bigg|_{\mathbf{w}=\langle\mathbf{w}\rangle}&=0\\ \Longleftrightarrow-\alpha\langle\mathbf{w}\rangle+\beta\mathbf{X}^\top(\mathbf{y}-\mathbf{X}\langle\mathbf{w}\rangle)&=0\\ \Longleftrightarrow(\alpha\mathbf{I}+\beta\mathbf{X}^\top\mathbf{X})\langle\mathbf{w}\rangle&=\beta\mathbf{X}^\top\mathbf{y}\\ \Longleftrightarrow\langle\mathbf{w}\rangle&=\left(\frac{\alpha}{\beta}\mathbf{I}+\mathbf{X}^\top\mathbf{X}\right)^{-1}\mathbf{X}^\top\mathbf{y}\end{aligned}$$

- Covariance of the posterior (see **Fisher information**)

$$\begin{aligned}\frac{\partial^2\log p(D,\mathbf{w}|\alpha,\beta)}{\partial\mathbf{w}^2}\bigg|_{\mathbf{w}=\langle\mathbf{w}\rangle}&=-\mathrm{Cov}(\mathbf{w})^{-1}\\ \Longleftrightarrow\mathrm{Cov}(\mathbf{w})^{-1}&=\alpha\mathbf{I}+\beta\mathbf{X}^\top\mathbf{X}\\ \Longleftrightarrow\mathrm{Cov}(\mathbf{w})&=\frac{1}{\beta}\left(\frac{\alpha}{\beta}\mathbf{I}+\mathbf{X}^\top\mathbf{X}\right)^{-1}\end{aligned}$$

- **(b) [MATH] Derive an EM algorithm for optimising the hyperparameter β by maximising the log-evidence**

$$p(D|\alpha,\beta)=\int p(D|\mathbf{w},\beta)p(\mathbf{w}|\alpha)\,d\mathbf{w}$$

Tip

Treat the weights $\mathbf{w}$ as a set of latent variables similar to the procedure for α given in the lecture. Express your result in terms of the posterior mean and variance.

Solution

- Expectation step

$$\begin{aligned}\mathcal{L} =&\langle \log p(D, \mathbf{w}|\alpha, \beta)\rangle \\ =&\left\langle \frac{K}{2}\log\frac{\alpha}{2\pi} + \frac{N}{2}\log\frac{\beta}{2\pi} - \frac{\alpha}{2}\mathbf{w}^\top\mathbf{w} - \frac{\beta}{2}(\mathbf{y}-\mathbf{X}\mathbf{w})^\top(\mathbf{y}-\mathbf{X}\mathbf{w})\right\rangle \\ =&\frac{K}{2}\log\frac{\alpha}{2\pi} + \frac{N}{2}\log\frac{\beta}{2\pi} - \frac{\alpha}{2}\langle\mathbf{w}^\top\mathbf{w}\rangle - \frac{\beta}{2}\langle(\mathbf{y}-\mathbf{X}\mathbf{w})^\top(\mathbf{y}-\mathbf{X}\mathbf{w})\rangle\end{aligned}$$

- Expected length of the weight vector

$$\begin{aligned}\langle\mathbf{w}^\top\mathbf{w}\rangle =&\mathrm{Tr}[\langle\mathbf{w}\mathbf{w}^\top\rangle] \\ =&\mathrm{Tr}[\mathrm{Cov}(\mathbf{w}) + \langle\mathbf{w}\rangle\langle\mathbf{w}^\top\rangle] \\ =&\mathrm{Tr}[\mathrm{Cov}(\mathbf{w})] + \langle\mathbf{w}^\top\rangle\langle\mathbf{w}\rangle\end{aligned}$$

- Expected distance between model and observations

$$\begin{aligned}&\langle(\mathbf{y}-\mathbf{X}\mathbf{w})^\top(\mathbf{y}-\mathbf{X}\mathbf{w})\rangle \\ =&\mathbf{y}^\top\mathbf{y} - \mathbf{y}^\top\mathbf{X}\langle\mathbf{w}\rangle - \langle\mathbf{w}^\top\rangle\mathbf{X}^\top\mathbf{y} + \mathrm{Tr}[\mathbf{X}\langle\mathbf{w}\mathbf{w}^\top\rangle\mathbf{X}^\top] \\ =&\mathbf{y}^\top\mathbf{y} - \mathbf{y}^\top\mathbf{X}\langle\mathbf{w}\rangle - \langle\mathbf{w}^\top\rangle\mathbf{X}^\top\mathbf{y} + \mathrm{Tr}[\mathbf{X}\mathrm{Cov}(\mathbf{w})\mathbf{X}^\top] \\ &+\mathrm{Tr}[\mathbf{X}\langle\mathbf{w}\rangle\langle\mathbf{w}^\top\rangle\mathbf{X}^\top] \\ =&\mathrm{Tr}[\mathbf{X}\mathrm{Cov}(\mathbf{w})\mathbf{X}^\top] + (\mathbf{y}-\mathbf{X}\langle\mathbf{w}\rangle)^\top(\mathbf{y}-\mathbf{X}\langle\mathbf{w}\rangle)\end{aligned}$$

- Maximization step

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\alpha} = 0 &\Longleftrightarrow \frac{K}{2\alpha} - \frac{1}{2}\langle\mathbf{w}^\top\mathbf{w}\rangle = 0 \\ &\Longleftrightarrow \alpha = \frac{K}{\langle\mathbf{w}^\top\mathbf{w}\rangle} \\ \frac{\partial\mathcal{L}}{\partial\beta} = 0 &\Longleftrightarrow \frac{N}{2\beta} - \frac{1}{2}\langle(\mathbf{y}-\mathbf{X}\mathbf{w})^\top(\mathbf{y}-\mathbf{X}\mathbf{w})\rangle = 0 \\ &\Longleftrightarrow \beta = \frac{N}{\langle(\mathbf{y}-\mathbf{X}\mathbf{w})^\top(\mathbf{y}-\mathbf{X}\mathbf{w})\rangle}\end{aligned}$$

(c) [CODE] Implement the posterior solution of a generalised linear model given an arbitrary base function $\Phi(X) = \{\Phi_1(X), \ldots, \Phi_K(X)\}$

```
Φ (generic function with 2 methods)
```

```
begin
```

```
    # Φ(x::Real) = [x] # Linear Case
    # Φ(x::Real) = [1, x, sin(x), cos(x)]
    Φ(x::Real) = [one(x), x, x^2, x^3]#evalpoly(x, ones(4))
    Φ(x::Vector) = mapreduce(Φ, hcat, x) # Create a matrix out of a vector
end
```

posterior_params (generic function with 1 method)

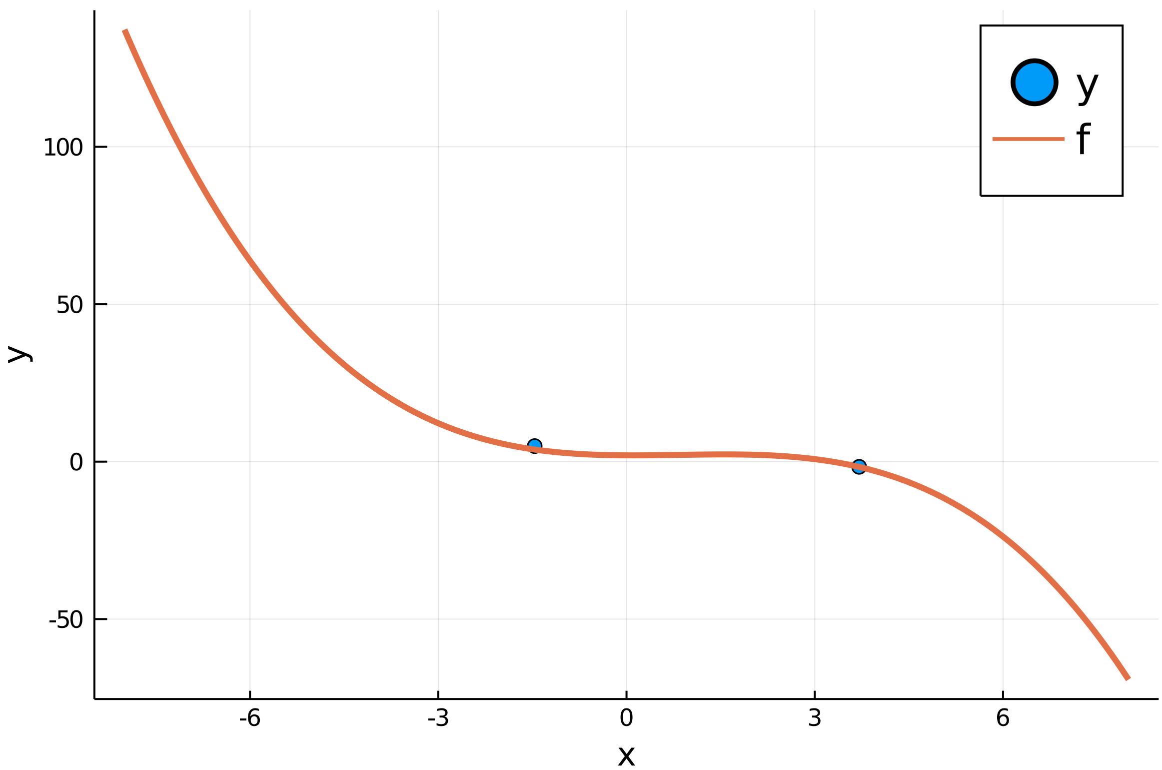
```
function posterior_params(val_Φ, y, α, β)
    Σ = inv(β) * inv(α / β * I + val_Φ*val_Φ')
    μ = β * Σ * val_Φ * y
    return μ, Σ
end
```

(d) [CODE] Load the data given and implement the EM algorithm to optimize α and β

data points : 10

β = 1.0

```
begin # Create some data
    X = rand(Uniform(-5, 5), Nx); # Sample X uniformly
    offset = 3
    X_test = collect(range(-5 - offset, 5 + offset, length = 500))
    w_true = [2.0, -0.1, 0.5, -0.2]
    f = evalpoly.(X, Ref(w_true)); # return w[0] + w[1] X + w[2] X^2 + ....
    y = f + randn(Nx) / sqrt(β_true); # Add some noise
end;
```



update_α (generic function with 1 method)

```
update_α(μ, Σ, K) = K / (tr(Σ) + norm(μ))
```

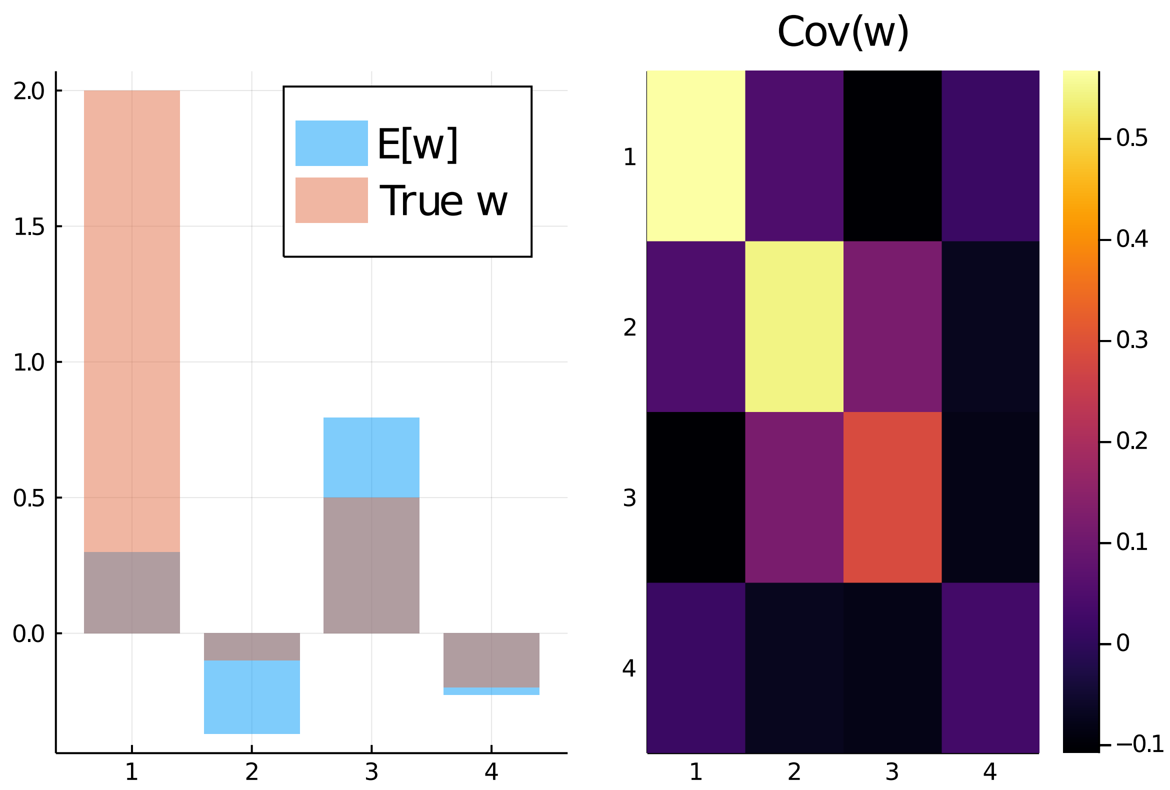
update_β (generic function with 1 method)

```
update_β(val_Φ, y, μ, Σ) = Nx / (tr(val_Φ' * Σ * val_Φ) + norm(y - val_Φ' * μ))
```

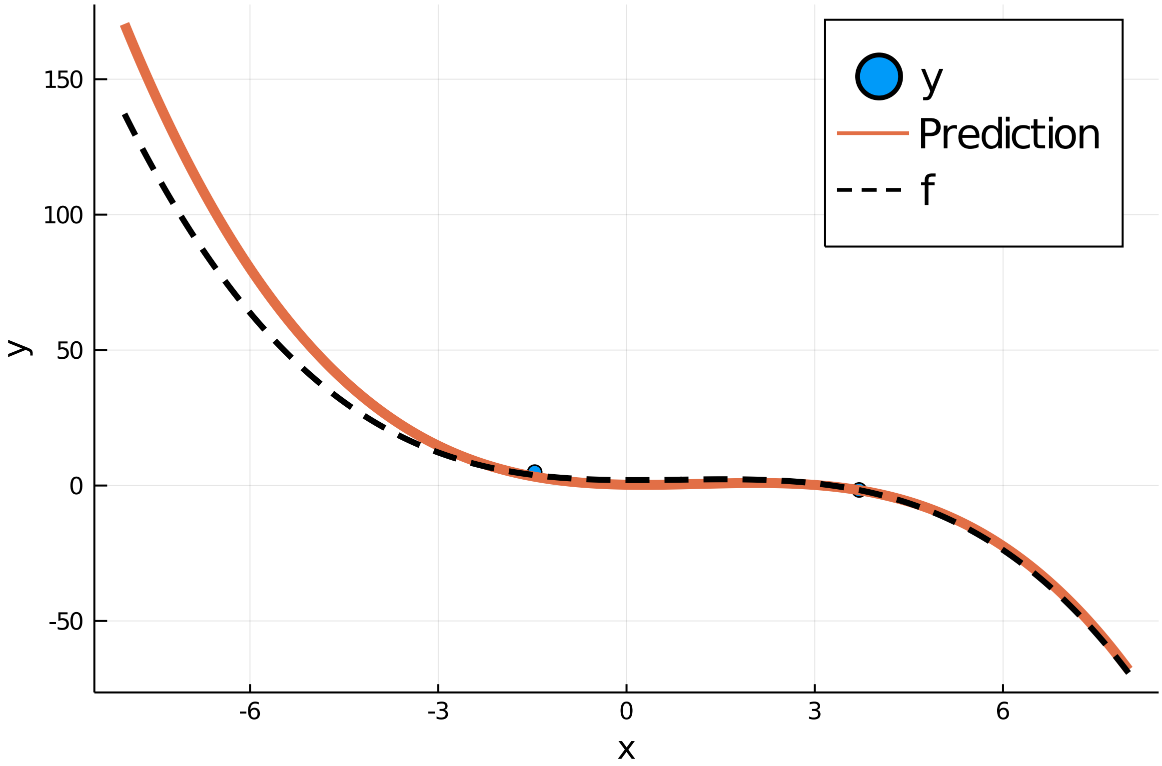
expectation_step (generic function with 1 method)

```
function expectation_step(val_Φ, y, μ, Σ, K, α, β)
    L = 0.5 * K * log(α / 2π)
    L += 0.5 * N * log(β / 2π)
    L += - 0.5 * α * (tr(Σ) + norm(μ))
    L += - 0.5 * β * (tr(val_Φ' * Σ * val_Φ) + norm(y - val_Φ' * μ))
    return L
end
```

```
begin
    α = 1.0 # Initial parameters
    β = 1.0
    val_Φ = Φ(X) # Feature map
    T = 10 # Number of steps
    K = size(val_Φ, 1) # Feature map dimension
    for i in 1:T
        μ, Σ = posterior_params(val_Φ, y, α, β)
        println("i = $i, pre L = $(expectation_step(val_Φ, y, μ, Σ, K, α, β)), α =
$α, β = $β")
        α = update_α(μ, Σ, K)
        β = update_β(val_Φ, y, μ, Σ)
        println("i = $i, post L = $(expectation_step(val_Φ, y, μ, Σ, K, α, β)), α =
$α, β = $β")
    end
end
```



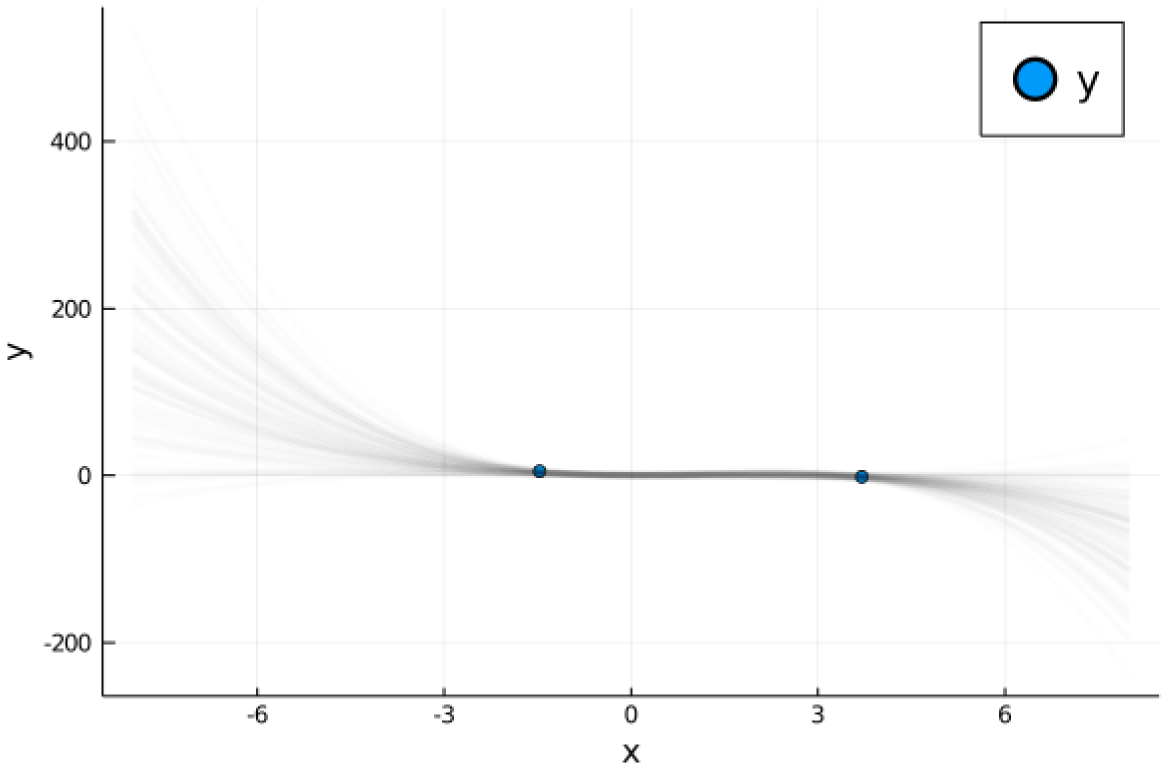

```
begin
    μ, Σ = posterior_params(val_Φ, y, α, β)
    plot(
        bar([μ, w_true], label = ["E[w]" "True w"], alpha = 0.5, lw = 0.0),
        heatmap(Σ, title = "Cov(w)", yflip =true),
    )
end
```




```
begin
    scatter(X, y, lab= "y", xlabel="x", ylabel="y")
    plot!(X_test, Φ(X_test)' * μ; lab="Prediction", linewidth=5.0)
    plot!(X_test, evalpoly.(X_test, Ref(w_true)); linestyle=:dash, color=:black,
label="f")
end
```

We can also sample from the posterior to visualize all the different possibilities

10




```
begin
    p = scatter(X, y, lab= "y", xlabel="x", ylabel="y")
```

```
    S = 100
    for i in 1:S
        w = rand(MvNormal(μ, Symmetric(Σ)))
        plot!(X_test, Φ(X_test)' * w; lab="", color=:black, alpha=0.01)
    end
    p
end
```

```
(α = 1.64887,  β = 0.280035)
```

```
(;α, β)
```

```
1.001
```

```
β_true
```