

# Bayesian Linear Regression

Guest lecture at KTH 2024

Mattias Villani

Department of Statistics  
Stockholm University



# Lecture overview

- Bayesian inference (see Timo's lecture)
- [Recap: the normal model with known variance]
- Linear regression
- Regularization priors
- Outlook: Bayes in complex problems

Slides on course page and at: <https://mattiasvillani.com/news>

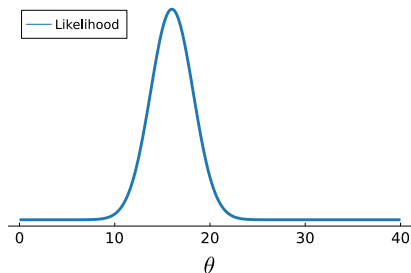
Rough draft book at: <https://github.com/mattiasvillani/BayesianLearningBook>

# Am I really getting my 20Mbit/sec?

- Internet connection should be at least 20Mbit/sec on average.
- **Data**:  $x = (15.77, 20.5, 8.26, 14.37, 21.09)$  Mbit/sec.
- **Model**: Normal data with known variance

$$X_1, \dots, X_n | \theta \stackrel{iid}{\sim} N(\theta, \sigma^2).$$

- **Measurement errors**:  $\sigma = 5$  ( $\pm 10$ Mbit with 95% probability)
- **Likelihood function** is proportional to  $N(\bar{x}, \sigma^2/n)$  density.



# Great theorems make great tattoos

## ■ Bayes theorem

$$p(\theta|\text{Data}) = \frac{p(\text{Data}|\theta)p(\theta)}{p(\text{Data})}$$

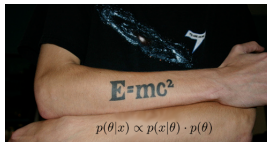
## ■ All you need to know:

$$p(\theta|\text{Data}) \propto p(\text{Data}|\theta)p(\theta)$$

$$\text{Posterior} \propto \text{Likelihood} \cdot \text{Prior}$$

## ■ A probability distribution for $\theta$ is extremely useful:

- ▶ **Predictions**
- ▶ **Decision making**
- ▶ **Regularization**



# Normal data, known variance - normal prior

## ■ Prior

$$\theta \sim N(\mu_0, \tau_0^2)$$

## ■ Posterior

$$\begin{aligned} p(\theta | x_1, \dots, x_n) &\propto p(x_1, \dots, x_n | \theta, \sigma^2) p(\theta) \\ &\propto N(\theta | \mu_n, \tau_n^2), \end{aligned}$$

where the **posterior mean** is

$$\mu_n = w\bar{x} + (1 - w)\mu_0$$

$$w = \frac{\frac{n}{\sigma^2}}{\frac{n}{\sigma^2} + \frac{1}{\tau_0^2}}.$$

■ Define: Precision  $\equiv 1/\text{Variance}$ .

■ Posterior precision = Data precision + Prior precision

$$\frac{1}{\tau_n^2} = \frac{n}{\sigma^2} + \frac{1}{\tau_0^2}$$

# Interactive - Bayes for Gaussian iid model

## Prior-Posterior - Gaussian data with known variance

Model:  $X_1, \dots, X_n \mid \theta, \sigma^2 \sim N(\theta, \sigma^2)$  with  $\sigma^2$  known.

Prior:  $\theta \sim N(\mu_0, \tau_0^2)$

Posterior:  $\theta \mid x \sim N(\mu_n, \tau_n^2)$

Posterior precision:  $\frac{1}{\tau_n^2} = \frac{1}{\tau_0^2} + \frac{1}{\sigma^2} = 0.240$

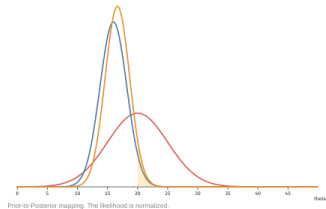
Posterior mean:  $\mu_n = w\bar{x} + (1-w)\mu_0 = 16.6$

Weight on data:  $w = \frac{\frac{1}{\sigma^2}}{\frac{1}{\sigma^2} + \frac{1}{\tau_0^2}} = 0.633$



Posterior quantile:  $P(\theta \leq 20 \mid x) = 0.0155$

Legend: █ likelihood █ posterior █ prior



Prior-to-Posterior mapping. The likelihood is normalized.

# Linear regression

- The linear regression model in **matrix form**

$$\underset{(n \times 1)}{y} = \underset{(n \times k)(k \times 1)}{X\beta} + \underset{(n \times 1)}{\varepsilon}$$

- First column of  $X$  is the unit vector and  $\beta_1$  is the intercept.
- Normal errors:  $\varepsilon_i \stackrel{iid}{\sim} N(0, \sigma^2)$ , so  $\varepsilon \sim N(0, \sigma^2 I_n)$ .

- **Likelihood**

$$y|\beta, \sigma^2, X \sim N(X\beta, \sigma^2 I_n)$$

# Linear regression - uniform prior

- Standard **non-informative prior**: uniform on  $(\beta, \log \sigma^2)$

$$p(\beta, \sigma^2) \propto \sigma^{-2}$$

- **Joint posterior** of  $\beta$  and  $\sigma^2$ :

$$\begin{aligned}\beta | \sigma^2, y &\sim N \left[ \hat{\beta}, \sigma^2 (X^\top X)^{-1} \right] \\ \sigma^2 | y &\sim \text{Inv-}\chi^2(n - k, s^2)\end{aligned}$$

where  $\hat{\beta} = (X^\top X)^{-1} X^\top y$  and  $s^2 = \frac{1}{n-k} (y - X\hat{\beta})^\top (y - X\hat{\beta})$ .

- **Simulate** from the joint posterior by simulating from

- ▶  $p(\sigma^2 | y)$
- ▶  $p(\beta | \sigma^2, y)$

- **Marginal posterior** of  $\beta$ :

$$\beta | y \sim t_{n-k} \left[ \hat{\beta}, s^2 (X^\top X)^{-1} \right]$$



# Interactive - Scaled Inv- $\chi^2$

$\nu$

$\tau^2$

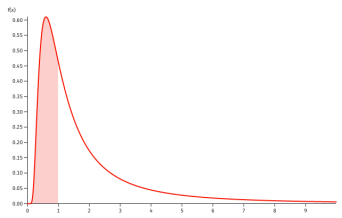
Quantile:

If  $X \sim \text{Inv-}\chi^2(3, 1)$  then

$$E(X) = \frac{\tau^2 \nu}{\nu - 2} = 0.75 \text{ for } \nu > 2$$

$$\text{Var}(X) = \frac{2\tau^4}{(\nu - 2)^2(\nu - 4)} = 0.21 \text{ for } \nu > 4$$

$$P(X \leq 1) = 0.3916$$



 Mattias Villani Scaled inverse  $\chi^2$  distribution

 Observable

# Linear regression - conjugate prior

## ■ Joint prior for $\beta$ and $\sigma^2$

$$\begin{aligned}\beta|\sigma^2 &\sim N(\mu_0, \sigma^2 \Omega_0^{-1}) \\ \sigma^2 &\sim \text{Inv-}\chi^2(\nu_0, \sigma_0^2)\end{aligned}$$

## ■ Posterior

$$\begin{aligned}\beta|\sigma^2, y &\sim N[\mu_n, \sigma^2 \Omega_n^{-1}] \\ \sigma^2|y &\sim \text{Inv-}\chi^2(\nu_n, \sigma_n^2)\end{aligned}$$

$$\begin{aligned}\mu_n &= W\hat{\beta} + (I - W)\mu_0 \\ W &= \left(X^\top X + \Omega_0\right)^{-1} X^\top X \\ \Omega_n &= X^\top X + \Omega_0\end{aligned}$$

## ■ Posterior Precision $\Omega_n = \text{Data Precision } X^\top X + \text{Prior Precision } \Omega_0$

# Bayesian Linear regression in Julia/Turing.jl 🥰

```
# Define the scaled-inverse-chi-squared distribution.
ScaledInverseChiSq(v,  $\tau^2$ ) = InverseGamma(v/2, v* $\tau^2$ /2)

@model function linear_regression(X, y,  $\mu_o$ ,  $\Omega_o$ ,  $v_o$ ,  $\sigma_o^2$ )

    # Priors
     $\sigma^2 \sim$  ScaledInverseChiSq( $v_o$ ,  $\sigma_o^2$ )
     $\beta \sim$  MvNormal( $\mu_o$ ,  $\sigma^2 * \text{inv}(\Omega_o)$ )

    return y  $\sim$  MvNormal(X* $\beta$ ,  $\sigma^2 * I$ )
end

# Simulate from posterior using HMC
n, p = size(X)
 $\mu_o$  = zeros(p)
 $\Omega_o$  = 0.1*I
 $v_o$  = p+1
 $\sigma_o^2$  = 1
model = linear_regression(X, y,  $\mu_o$ ,  $\Omega_o$ ,  $v_o$ ,  $\sigma_o^2$ )
chain = sample(model, NUTS(0.65), 3000)
```

# Bike share data

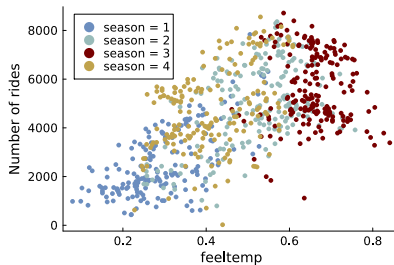
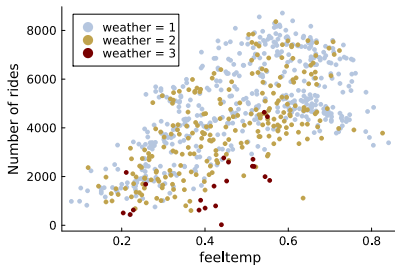
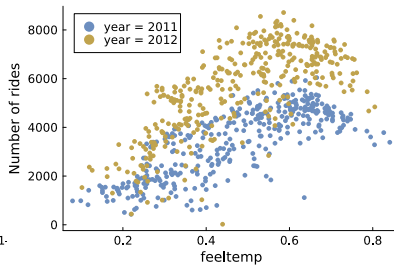
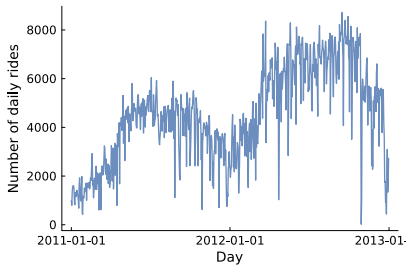
- **Bike share data.** Predict the number of bike rides.
- Response variable: number of rides on 731 days.

variable	description	data type	values	comment
nrides	number of rides	counts	$\{0, 1, \dots\}$	min= 22, max= 8714
feeltemp	perceived temp	continuous	$[0, 1]$	min= 0.07, max= 0.85
hum	humidity	continuous	$[0, 1]$	min= 0.00, max= 0.98
wind	wind speed	continuous	$[0, 1]$	min= 0.02, max= 0.51
year	year	binary	$\{0, 1\}$	year 2011 = 0
season	season	categorical	$\{1, 2, 3, 4\}$	winter $\rightarrow$ fall
weather	weather	ordinal	$\{1, 2, 3\}$	clear $\rightarrow$ rain/snow
weekday	day of week	categorical	$\{0, 1, \dots, 6\}$	sunday $\rightarrow$ saturday
holiday	holiday	binary	$\{0, 1\}$	holiday = 1

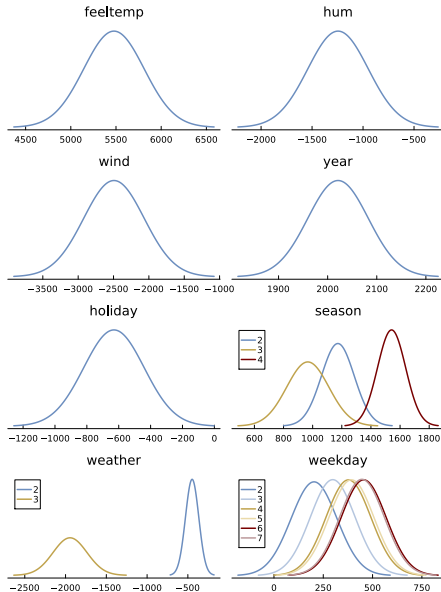
- Prior:

- ▶  $\mu_0 = (1000, 0, \dots, 0)^\top$
- ▶  $\Omega_0 = \frac{\kappa_0}{n} \mathbf{X}^\top \mathbf{X}$  with  $\kappa_0 = 1$  (unit information prior)
- ▶  $\sigma_0^2 = 1000^2$  and  $\nu_0 = 5$ .

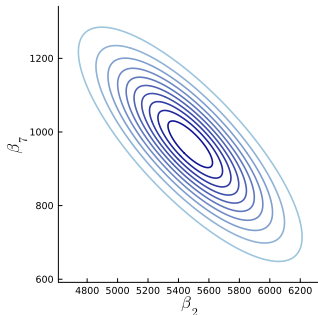
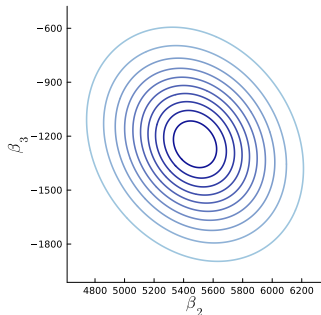
# Bike share data



# Bike share data - marginal posteriors of $\beta$



# Bike share data - joint posteriors of $\beta$



# Interactive - Bayesian regression





# Ridge regression = iid normal prior

- Smoothness/shrinkage/regularization prior [ $\Omega_0 = \lambda I$ ]

$$\beta_i | \lambda, \sigma^2 \stackrel{\text{iid}}{\sim} \mathcal{N}\left(0, \frac{\sigma^2}{\lambda}\right)$$

- Posterior mean is the ridge regression estimator

$$\mu_n = \left(X^\top X + \lambda I\right)^{-1} X^\top y$$

- Shrinkage toward zero

$$\text{As } \lambda \rightarrow \infty, \mu_n \rightarrow 0$$

- When  $X^\top X = I$

$$\mu_n = (1 - \phi)\hat{\beta}, \quad \text{for } \phi = \frac{\lambda}{1 + \lambda}$$

- Shrinkage factor  $\phi \in [0, 1]$ .

# Learning the optimal shrinkage

- Cross-validation is often used to determine  $\lambda$ .
- Bayesian:  $\lambda$  is **unknown**  $\Rightarrow$  **use a prior** for  $\lambda$ .
- $\lambda^{-1} \sim \text{Inv-}\chi^2(\omega_0, \psi_0^2)$ . The user specifies  $\omega_0$  and  $\psi_0^2$ .
- Joint posterior
$$p(\beta, \sigma^2, \lambda | \mathbf{y}, \mathbf{X})$$
- Marginal posterior  $\lambda$ .
- Gibbs sampling

# Learning the optimal shrinkage

## Gibbs sampling linear regression - L2 regularization prior

The posterior for the linear regression model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \boldsymbol{\varepsilon} \sim N(\mathbf{0}, \sigma^2 I_n), \quad (11.16)$$

with hierarchical L2 regularization prior

$$\begin{aligned}\boldsymbol{\beta} | \sigma^2, \lambda &\sim N(\mathbf{0}, (\sigma^2 / \lambda) I_p) \\ \sigma^2 &\sim \text{Inv-}\chi^2(\tau_0^2, \nu_0) \\ \lambda^{-1} &\sim \text{Inv-}\chi^2(\omega_0, \psi_0^2).\end{aligned}$$

can be sampled by a two-block Gibbs sampler:

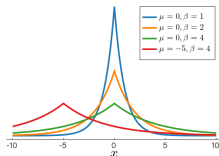
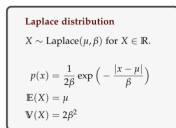
$$\begin{aligned}\text{Block1 : } \boldsymbol{\beta} | \sigma^2, \lambda, \mathbf{y} &\sim N(\hat{\boldsymbol{\beta}}_{L_2}, \sigma^2 (\mathbf{X}^\top \mathbf{X} + \lambda I_p)^{-1}) \\ \sigma^2 | \lambda, \mathbf{y} &\sim \text{Inv-}\chi^2(\tau_n^2, \nu_n)\end{aligned}$$

$$\text{Block2 : } \lambda^{-1} | \boldsymbol{\beta}, \sigma^2, \mathbf{y} \sim \text{Inv-}\chi^2(\omega_n, \psi_n^2),$$

# Lasso regression = Laplace prior

- **Lasso** is equivalent to posterior mode under Laplace prior

$$\beta_i | \lambda, \sigma^2 \stackrel{\text{iid}}{\sim} \text{Laplace} \left( 0, \frac{\sigma^2}{\lambda} \right)$$



- **Laplace prior:**
  - ▶ heavy tails
  - ▶ many  $\beta_i$  close to zero, but some  $\beta_i$  can be very large.
- **Normal prior:**
  - ▶ light tails
  - ▶ all  $\beta_i$ 's are similar in magnitude and no  $\beta_i$  very large.

# Horseshoe prior

- Normal and Laplace - one global shrinkage parameter  $\lambda$ .
- **Global-Local shrinkage**: global + local shrinkage for each  $\beta_j$ .
- **Horseshoe prior**:

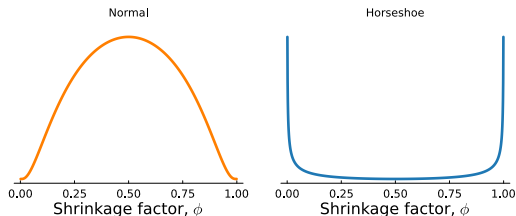
$$\beta_j | \lambda_j^2, \tau^2 \sim N(0, \tau^2 \lambda_j^2)$$

$$\lambda_j \sim C^+(0, 1)$$

$$\tau \sim C^+(0, 1)$$

- The posterior mean for  $\beta$  satisfies approximately

$$\mu_{nj} \approx (1 - \phi_j) \hat{\beta}_j, \text{ where } \frac{1}{1 + (n/\sigma^2) \tau^2 \lambda_j^2}$$



# Spike-and-slab prior

## ■ Spike-and-slab prior

$$\beta_j | \sigma^2, \lambda, l_j \sim \begin{cases} 0 & \text{if } l_j = 0 \\ N(0, \sigma^2 \omega) & \text{if } l_j = 1 \end{cases}$$

## ■ Prior for the variable selection indicators

$$l_j \stackrel{iid}{\sim} \text{Bernoulli}(\pi)$$

## ■ This is a mixture prior for the $\beta_j$

$$p(\beta_j) = (1 - \pi)\delta_0(\beta_j) + \pi N(\beta_j | \mu_j, \sigma^2 \omega^2)$$

## ■ Gibbs sampling gives Bayesian variable selection

$$\beta | \mathbf{y}, \mathbf{X}, \sigma^2, l_1, \dots, l_n \sim \text{Normal}$$

$$\sigma^2 | \mathbf{y}, \mathbf{X}, l_1, \dots, l_n \sim \text{Inv-}\chi^2$$

$$l_j | \mathbf{y}, \mathbf{X}, l_{-j}, \beta, \sigma^2 \sim \text{Bernoulli}(\bar{\pi}_j), \text{ for } j = 1, \dots, n$$

# Polynomial regression

- **Polynomial regression** is linear in  $\beta$ :

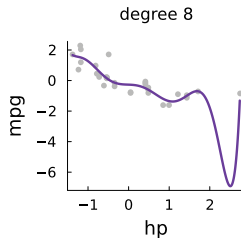
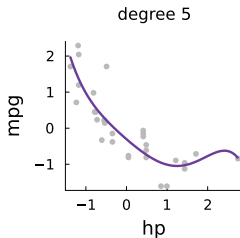
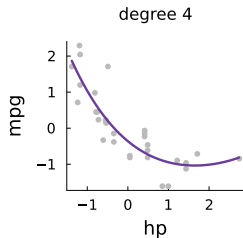
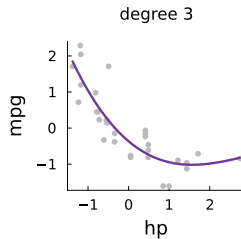
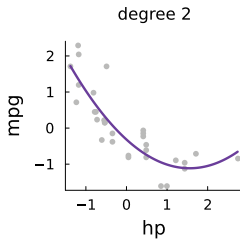
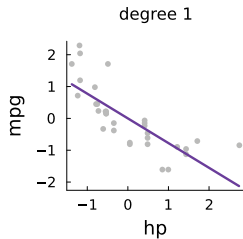
$$f(x_i) = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \dots + \beta_k x_i^k.$$

$$y = X\beta + \varepsilon, \text{ where } X = (1, x, x^2, \dots, x^k).$$

- Problem: higher order polynomials can **overfit** the data.
- Solution: **shrink** higher order coefficients harder:

$$\beta | \sigma^2 \sim N \left[ 0, \begin{pmatrix} 100 & 0 & 0 & \dots & 0 \\ 0 & \frac{1}{\lambda} & 0 & \dots & 0 \\ 0 & 0 & \frac{1}{2\lambda} & & \\ \vdots & \vdots & & \ddots & \\ 0 & 0 & 0 & \dots & \frac{1}{k\lambda} \end{pmatrix} \right]$$

# Polynomial regression mtcars data





# Bayes is easy to use

- Substantially more complex models can be analyzed by
  - ▶ **Markov Chain Monte Carlo** (MCMC) simulation
  - ▶ **Hamiltonian Monte Carlo** (HMC) simulation
  - ▶ **Variational inference** optimization
- **Deep Learning**. Bayes quantifies uncertainty  $\Rightarrow$  Probabilistic predictions  $\Rightarrow$  Decisions under uncertainty.
- Ongoing research on making Bayes more scalable to large data.  
My own contributions: <https://mattiasvillani.com/research>
- Probabilistic programming languages make Bayes easy:
  - ▶ **Stan** (R and more)
  - ▶ **Turing.jl** (Julia)
  - ▶ **Pyro** (Python)
- Bayesian Learning course at SU (March-April):  
<https://github.com/mattiasvillani/BayesLearnCourse>  
**Engineers welcome!**

# Poisson regression in Turing.jl (Julia)

## ■ Poisson regression:

$$y_i | \theta_i \sim \text{Pois}(\exp(\theta_i)), \quad \text{for } i = 1, \dots, n$$

$$\theta_i = \mathbf{x}_i^\top \boldsymbol{\beta}$$

$$\boldsymbol{\beta} \sim N(0, \tau_0^2 I)$$

```
# Bayesian poisson regression model in Turing.jl
@model poisson_reg(x, y, τ₀) = begin
    n = length(y)
    β₀ ~ Normal(0, τ₀^2)
    β₁ ~ Normal(0, τ₀^2)
    β₂ ~ Normal(0, τ₀^2)
    β₃ ~ Normal(0, τ₀^2)
    for i = 1:n
        θ = β₀ + β₁*x[i, 1] + β₂*x[i, 2] + β₃*x[i, 3]
        y[i] ~ Poisson(exp(θ))
    end
end

# Simulate from the posterior using HMC with NUTS tuning
sample(poisson_reg(X, y, 10), NUTS(200, 0.65), 2500)
```

## ■ Deep Neural Net in Turing.jl:

<https://turing.ml/dev/tutorials/03-bayesian-neural-network/>.