Bayesian Linear Regression

Guest lecture at KTH 2023

Mattias Villani

Department of Statistics
Stockholm University

Department of Computer and Information Science
Linköping 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**

Likelihood function - normal data

- **Normal data** with **known variance**:

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

- **Likelihood** from independent observations: $x_1, ..., x_n$

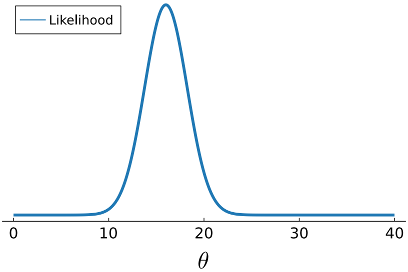
$$p(x_1, ..., x_n|\theta) = \prod_{i=1}^{n} p(x_i|\theta) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i - \theta)^2\right)$$
$$\propto \exp\left(-\frac{1}{2(\sigma^2/n)}(\theta - \bar{x})^2\right)$$

- **Maximum likelihood**: $\hat{\theta} = \bar{x}$ maximizes $p(x_1, ..., x_n|\theta)$.

- Given the data $x_1, ..., x_n$, plot $p(x_1, ..., x_n|\theta)$ as a function of θ.

Am I really getting my 20Mbit/sec?

- I have a 50Mbit/sec internet connection.
- ISP promises at least 20Mbit/sec on average.
- **Data**: $x = (15.77, 20.5, 8.26, 14.37, 21.09)$ Mbit/sec.
- **Measurement errors**: $\sigma = 5$ ($\pm$10Mbit with 95% probability)
- The likelihood function is proportional to $\mathrm{N}(\bar{x}, \sigma^2/n)$ density.



Great theorems make great tattoos

- Bayes theorem

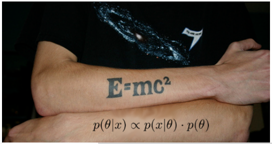
$$p(\theta|\text{Data}) = \frac{p(\text{Data}|\theta)p(\theta)}{p(\text{Data})}$$

- 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 θ is extremely useful:
  - **Predictions**
  - **Decision making**
  - **Regularization**



Normal data, known variance - normal prior

- **Prior**

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

- **Posterior**

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

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: $\text{Precision} \equiv 1/\text{Variance}$.
- Posterior precision = Data precision + Prior precision

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

Download speed

- **Data**: $x = (15.77, 20.5, 8.26, 14.37, 21.09)$ Mbit/sec.
- **Model**: $X_1, ..., X_5 \sim N(\theta, \sigma^2)$.
- Assume $\sigma = 5$ (measurements can vary $\pm$10MBit with 95% probability)
- My **prior**: $\theta \sim N(20, 5^2)$.

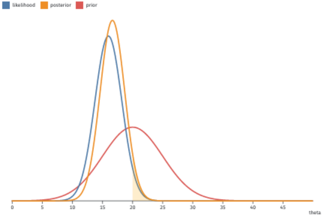
Interactive - Bayes for Gaussian iid model

Prior-Posterior - Gaussian data with known variance

Model:	$X_1, \ldots, X_n \mid \theta, \sigma^2 \sim \mathrm{N}(\theta, \sigma^2)$ with σ^2 known.
Prior:	$\theta \sim N(\mu_0, \tau_0^2)$
Posterior:	$\theta \vert \boldsymbol{x} \sim N(\mu_n, \tau_n^2)$
Posterior precision:	$\frac{1}{\tau_n^2} = \frac{n}{\sigma^2} + \frac{1}{\tau_0^2}$ = 0.240
Posterior mean:	$\mu_n = w\bar{x} + (1-w)\mu_0$ = 16.6
Weight on data:	$w = \frac{\frac{n}{\sigma^2}}{\frac{n}{\sigma^2} + \frac{1}{\tau_0^2}}$ = 0.833

n	5
σ	5
μ_0	20
τ_0	5
quantile posterior	20

Posterior quantile: $P(\theta \leq 20 \vert \boldsymbol{x}) = 0.0155$



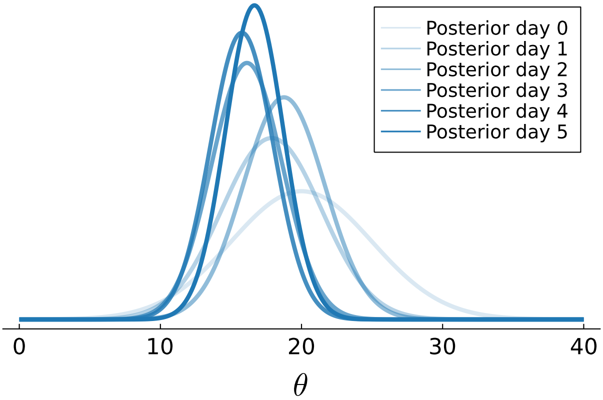
Prior-to-Posterior mapping. The likelihood is normalized.

Mattias Villani Gaussian iid data with known variance

Observable

Bayesian Online learning

- Yesterday's posterior is today's prior.



Bayesian Prediction

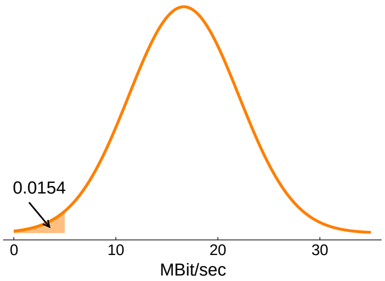
- **Predictive distribution** averages over the unknown parameter

$$\underbrace{p(x_{n+1}|x_{1:n})}_{\text{predictive dist}} = \int \underbrace{p(x_{n+1}|\theta)}_{\text{model}}\underbrace{p(\theta|x_{1:n})}_{\text{posterior}}d\theta$$

- Normal data, normal prior:

$$x_{n+1}|x_{1:n} \sim N(\mu_n, \sigma^2 + \tau_n^2)$$

- My streaming buffers whenever $x < 5$ MBit/Sec. 🤬



Linear regression

- The linear regression model in matrix form

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

- First column of X is the unit vector and β_1 is the intercept.
- Normal errors: $\varepsilon_i \overset{iid}{\sim} N(0, \sigma^2)$, so $\varepsilon \sim N(0, \sigma^2 I_n)$.
- Likelihood

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

Linear regression - uniform prior

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

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

- Joint posterior of β and σ^2:

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

  where $\hat{\beta} = (\mathrm{X}^\top \mathrm{X})^{-1}\mathrm{X}^\top \mathrm{y}$ and $s^2 = \frac{1}{n-k}(\mathrm{y} - \mathrm{X}\hat{\beta})^\top(\mathrm{y} - \mathrm{X}\hat{\beta})$.
- Simulate from the joint posterior by simulating from
  - $p(\sigma^2|\mathrm{y})$
  - $p(\beta|\sigma^2, \mathrm{y})$
- Marginal posterior of β :

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

Interactive - Scaled Inv-χ^2

ν 3

τ^2 1

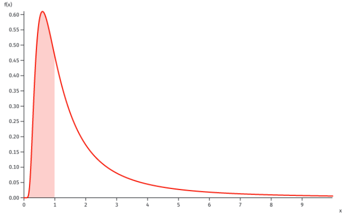
Quantile: 1

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

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

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

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



Mattias Villani Scaled inverse χ2\chi^2χ2 distribution

Observable

Linear regression - conjugate prior

- **Joint prior** for β and σ^2

$$\beta|\sigma^2 \sim N\left(\mu_0, \sigma^2\Omega_0^{-1}\right)$$
$$\sigma^2 \sim \mathrm{Inv-}\chi^2\left(\nu_0, \sigma_0^2\right)$$

- **Posterior**

$$\beta|\sigma^2, \mathrm{y} \sim N\left[\mu_n, \sigma^2\Omega_n^{-1}\right]$$
$$\sigma^2|\mathrm{y} \sim \mathrm{Inv-}\chi^2\left(\nu_n, \sigma_n^2\right)$$

$$\mu_n = \left(\mathrm{X}^\top\mathrm{X} + \Omega_0\right)^{-1}\left(\mathrm{X}^\top\mathrm{X}\hat{\beta} + \Omega_0\mu_0\right)$$
$$\Omega_n = \mathrm{X}^\top\mathrm{X} + \Omega_0$$
$$\nu_n = \nu_0 + n$$
$$\sigma_n^2 = \left(\nu_0\sigma_0^2 + \mathrm{y}^\top\mathrm{y} + \mu_0^\top\Omega_0\mu_0 - \mu_n^\top\Omega_n\mu_n\right)/\nu_n$$

Bayesian Linear regression in Julia 😍

```julia
function BayesLinReg(y::Vector, X, μ₀, Ω₀, ν₀, σ²₀, nSim)

    # Compute posterior hyperparameters
    n = length(y)
    p = size(X,2)
    XX = X'*X
    βhat = X \ y
    Ωₙ = Symmetric(XX + Ω₀)
    μₙ = Ωₙ\(XX*βhat + Ω₀*μ₀)
    νₙ = ν₀ + n
    σ²ₙ = (ν₀*σ²₀ + (y-X*βhat)'*(y-X*βhat) +
            (μₙ-βhat)'*XX*(μₙ-βhat) +
            (μₙ-μ₀)'*Ω₀*(μₙ-μ₀)
            )/νₙ

    # Sampling from posterior
    invΩₙ = inv(Ωₙ)
    σ²sim = zeros(nSim)
    βsim = zeros(nSim,p)
    for i ∈ 1:nSim
        # Simulate from p(σ²|y,X)
        σ² = rand(ScaledInverseChiSq(νₙ,σ²ₙ))
        σ²sim[i] = σ²

        # Simulate from p(β|σ²,y,X)
        β = rand(MvNormal(μₙ,σ²*invΩₙ))
        βsim[i,:] = β'

    end
    return μₙ, Ωₙ, νₙ, σ²ₙ, βsim, σ²sim
end
```

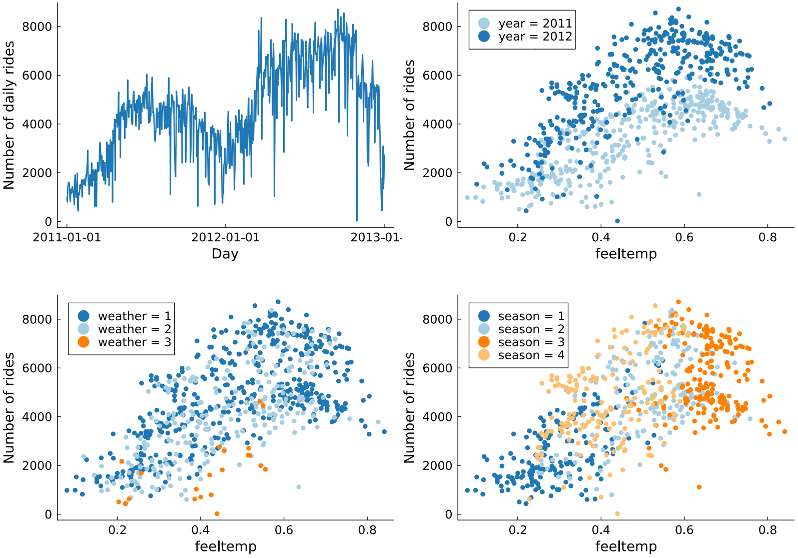
Bike share data

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

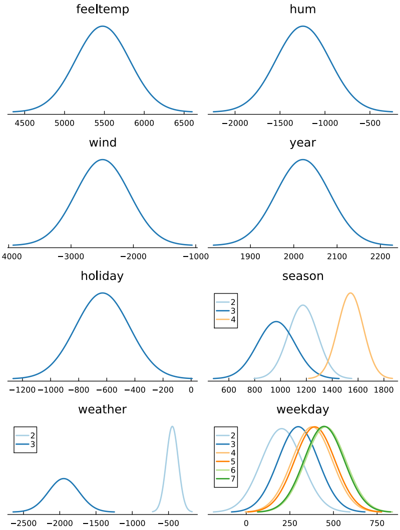
variable	description	data type	values	comment
`nrides`	number of rides	counts	$\{0, 1, ...\}$	min$= 22$, max$= 8714$
`feeltemp`	perceived temp	continuous	$[0, 1]$	min$= 0.07$, max$= 0.85$
`hum`	humidity	continuous	$[0, 1]$	min$= 0.00$, max$= 0.98$
`wind`	wind speed	continuous	$[0, 1]$	min$= 0.02$, max$= 0.51$
`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	$\{0, 1, ..., 6\}$	sunday $\rightarrow$ saturday
`holiday`	holiday	binary	$\{0, 1\}$	holiday $= 1$

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

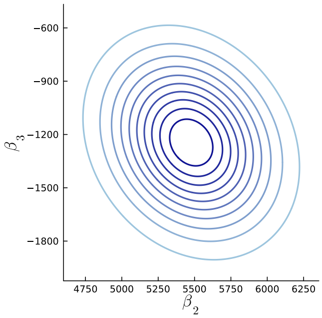
Bike share data

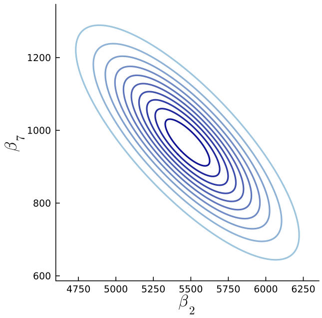


Bike share data - marginal posteriors of β



Bike share data - joint posteriors of β





Interactive - Bayesian regression

prior type: (•) Zellner g-prior ( ) Identity

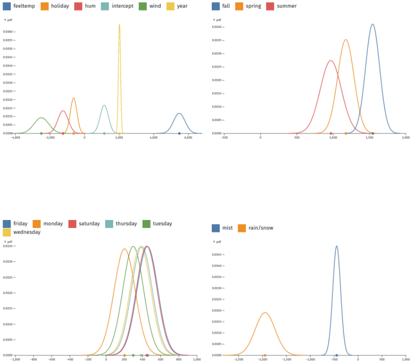
prior sample size, κ_0: 1

prior intercept, μ_0: 1000

ν_0: 5

σ_0: 1000

Marginal posteriors for selected variables. Least squared estimates are indicated by points.



Mattias Villani Bayesian linear regression- bike share data

Observable

Ridge regression = iid normal prior

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

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

- Posterior mean is the ridge regression estimator

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

- Shrinkage toward zero

$$\text{As } \lambda \to \infty, \;\; \mu_n \to 0$$

- When $\mathrm{X}^\top \mathrm{X} = I$

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

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

Learning the optimal shrinkage

- Cross-validation is often used to determine λ.
- Bayesian: λ is **unknown** $\Rightarrow$ **use a prior** for λ.
- $\lambda^{-1} \sim \text{Inv-}\chi^2(\omega_0, \psi_0^2)$. The user specifies ω_0 and ψ_0^2.
- Joint posterior

$$p(\beta, \sigma^2, \lambda | \boldsymbol{y}, \boldsymbol{X})$$

- Marginal posterior λ.
- Gibbs sampling

Learning the optimal shrinkage

Gibbs sampling linear regression - L2 regularization prior

The posterior for the linear regression model

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

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:

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

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

Lasso regression = Laplace prior

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

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

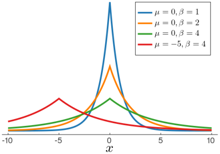
Laplace distribution

$X \sim \text{Laplace}(\mu, \beta)$ for $X \in \mathbb{R}$.

$$p(x) = \frac{1}{2\beta}\exp\left(-\frac{|x-\mu|}{\beta}\right)$$

$\mathbb{E}(X) = \mu$

$\mathbb{V}(X) = 2\beta^2$



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

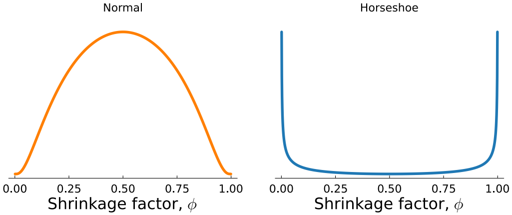
Horseshoe prior

- Normal and Laplace - one global shrinkage parameter λ.
- **Global-Local shrinkage**: global + local shrinkage for each β_j.
- **Horseshoe prior**:

$$\beta_j | \lambda_j^2, \tau^2 \sim N\left(0, \tau^2 \lambda_j^2\right)$$
$$\lambda_j \sim C^+(0,1)$$
$$\tau \sim C^+(0,1)$$

- The posterior mean for $\boldsymbol{\beta}$ satisfies approximately

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



Spike-and-slab prior

- **Spike-and-slab prior**

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

- Prior for the **variable selection indicators**

$$I_j \overset{iid}{\sim} \text{Bernoulli}(\pi)$$

- This is a **mixture prior** for the β_j

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

- **Gibbs sampling** gives **Bayesian variable selection**

$$\boldsymbol{\beta}|\boldsymbol{y}, \boldsymbol{X}, \sigma^2, I_1, \ldots, I_n \sim \text{Normal}$$

$$\sigma^2|\boldsymbol{y}, \boldsymbol{X}, I_1, \ldots, I_n \sim \text{Inv}-\chi^2$$

$$I_j|\boldsymbol{y}, \boldsymbol{X}, I_{-j}, \boldsymbol{\beta}, \sigma^2 \sim \text{Bernoulli}(\bar{\pi}_j), \text{ for } j = 1, \ldots, n$$

Learning the optimal shrinkage

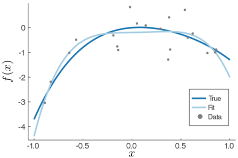
- Cross-validation is often used to determine λ.
- Bayesian: λ is **unknown** $\Rightarrow$ **use a prior** for λ.
- $\lambda \sim \text{Inv-}\chi^2(\eta_0, \lambda_0)$. The user specifies η_0 and λ_0.
- Joint posterior
$$p(\beta, \sigma^2, \lambda | \boldsymbol{y}, \boldsymbol{X})$$
- Marginal posterior λ.

Polynomial regression

- **Polynomial regression** is linear in β:

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

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



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

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

How long until maximal pain relief?

- Quadratic relationship between pain relief (y) and time (x)

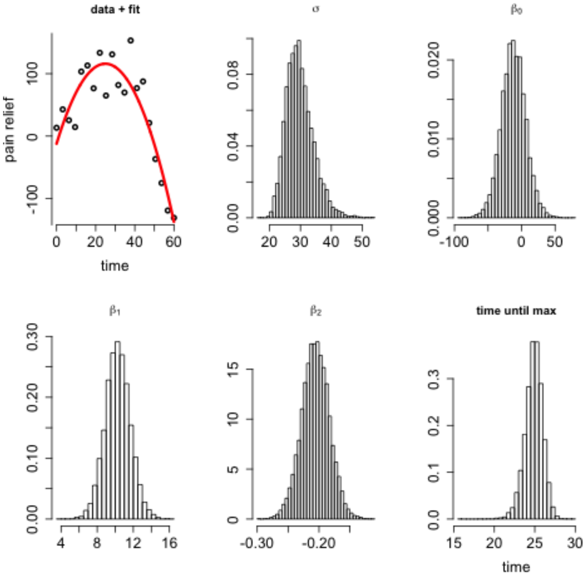
$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \varepsilon.$$

- At what time $x_{\max}$ is there **maximal pain relief**?

$$x_{\max} = -\beta_1 / 2\beta_2$$

- Easy to obtain marginal posterior $p(x_{max}|\mathrm{y}, \mathrm{X})$ by **simulation**:
  - Simulate N coefficient vectors from the posterior $\beta, \sigma^2|\mathrm{y}, \mathrm{X}$
  - For each simulated β, compute $x_{\max} = -\beta_1 / 2\beta_2$.
  - Plot a histogram. Converges to $p(x_{max}|\mathrm{y}, \mathrm{X})$ as $N \to \infty$.

How long until maximal pain relief?



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 ⇒ Probabilistic predictions ⇒ 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:

$$
\begin{aligned}
y_i|\theta_i \sim & \text{Pois}\left(\exp(\theta_i)\right), \quad \text{for } i = 1, \ldots, n \\
\theta_i &= \boldsymbol{x}_i^\top \boldsymbol{\beta} \\
\boldsymbol{\beta} &\sim N(0, \tau_0^2 I)
\end{aligned}
$$

```julia
# 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/.