Bayesian Linear Regression

Richard Dirauf, M.Sc.
Machine Learning and Data Analytics (MaD) Lab
Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU)
MLTS Exercise, 07.11.2024

MLTS Exercise – Organization

~~Introduction (31.10.2024)~~

Bayesian Linear Regression (07.11.2024)

Bayesian Linear Regression (14.11.2024)

Kalman Filter (21.11.2024)

Kalman Filter (28.11.2024)

Dynamic Time Warping (05.12.2024)

Dynamic Time Warping (12.12.2024)

No exercise planned (19.12.2024)

Holiday

RNN + LSTM (09.01.2025)

RNN + LSTM (16.01.2025)

Transformers (23.01.2025)

Transformers (30.01.2025)

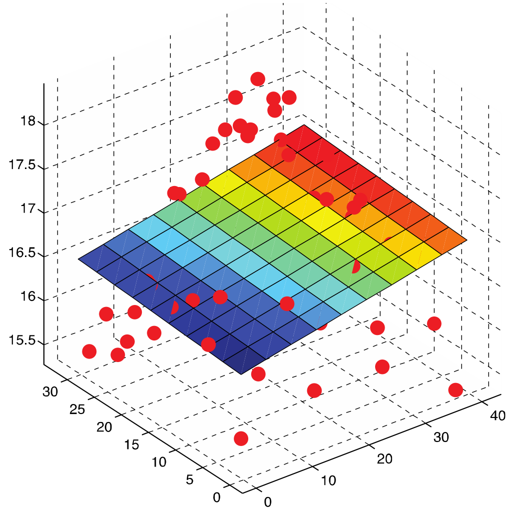
What is Linear Regression?

MLTS Exercise 02

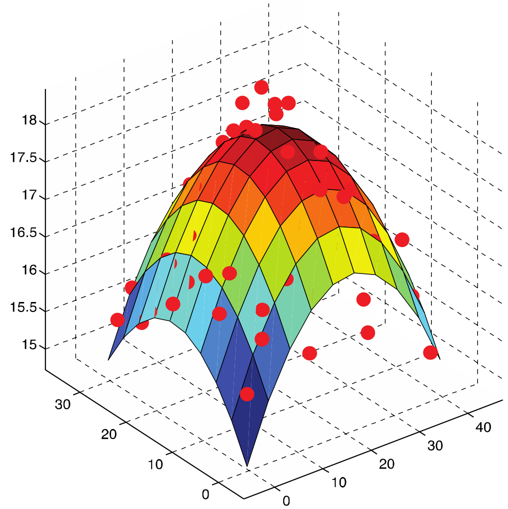
Given some tuples in a dataset:

$$\mathcal{D} = \{(X_A, y_A), (X_B, y_B), \dots, (X_N, y_N)\}$$

We want to predict a scalar $\boldsymbol{y}$ response with one or multiple explanatory variables $\boldsymbol{x}$



$$y = w_0 + w_1 x_1 + w_2 x_2$$



$$y = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2$$

Practice Questions

MLTS Exercise 02

Can the following function be considered in a linear regression:

$$\boldsymbol{y = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2}$$

?

What is Linear Regression?

MLTS Exercise 02

Given: $\mathcal{D} = \{(X_A, y_A), (X_B, y_B), \dots, (X_N, y_N)\}$

Where: $X \in \mathcal{R}^D,\ y \in \mathcal{R}$

Find: $f_w: \mathcal{R}^D \to \mathcal{R}$

Predict:

$$y = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_D x_D + \epsilon$$

Noise Error e.g.: $\epsilon = \mathcal{N}(\mathbf{0}, \mathbf{1})$

→ Random sampling noise or effect of variables not included in the model

Best Parameters in Frequentists View?

MLTS Exercise 02

Ordinary Least Squares (Smallest Residual Error)

$$RSS(W) = \sum_{i=1}^{N} (y_i - W^T X_i)^2$$

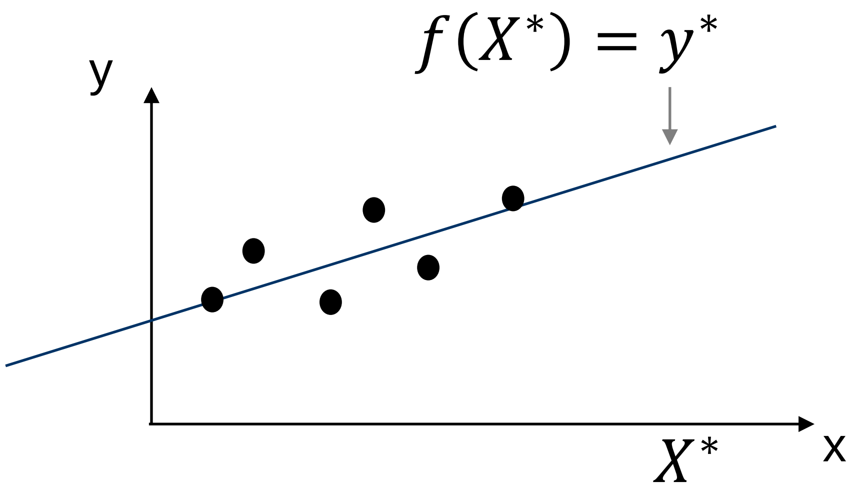
Find parameters:

$$W^* = (X^T X)^{-1} X^T Y$$

Predict:

$$y^* = f(X^*) = W^T X^* = \sum_{i=1}^{D} w_i x_i^*$$

We only get a point estimate!



What to do instead?

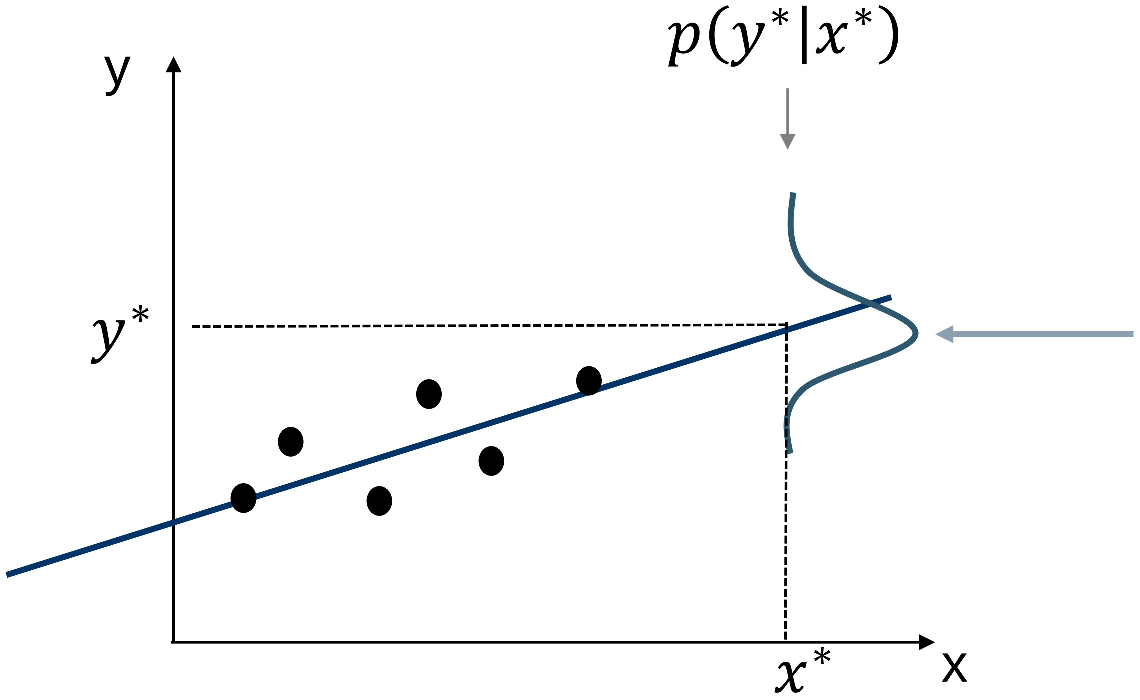
Get **distribution** of possible **y** values given **X**

$$p(y|X)$$

Formulate LR using probability distributions instead of point estimates:

$$p(y|X) = \mathcal{N}(y|\mu(X), \sigma^2(X));\ \theta = (\mu, \sigma^2)$$

Get **distribution** of possible **y** values given **x**



Some distribution (e.g., Gaussian)

Why might we want to employ a Bayesian instead of a Frequentists model in a safety-critical environment?

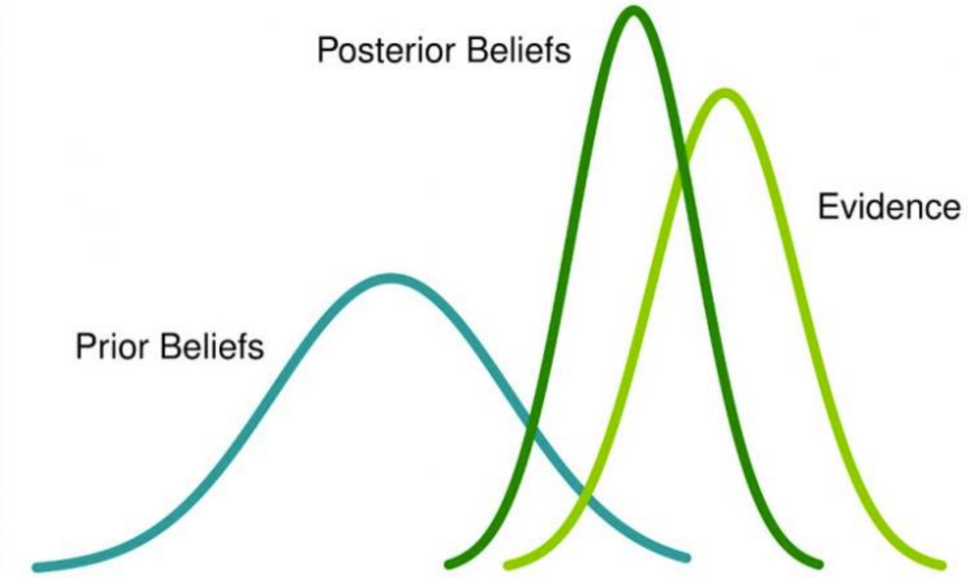
?

Bayesian Linear Regression

MLTS Exercise 02

Bayes Rule:

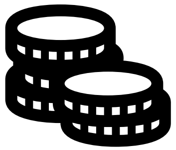
$$\underbrace{p(\theta|\mathcal{D})}_{\text{Posterior}} = \frac{\overbrace{p(\mathcal{D}|\theta)}^{\text{Likelihood}}\,\overbrace{p(\theta)}^{\text{Prior}}}{\underbrace{p(\mathcal{D})}_{\substack{\text{Marginal Likelihood}\\ \text{(Evidence)}}}}$$



Example

MLTS Exercise 02

What is the probability of the outcome of a coin flip game being fair?



$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{p(\mathcal{D})}$$

Observed data $\boldsymbol{\mathcal{D}}$

Model Parameters $\boldsymbol{\theta}$

Evidence $\boldsymbol{p(\mathcal{D})}$ → Probability of observing data across all possible θ

Prior $\boldsymbol{p(\theta)}$ → Believe of the fairness of the coin $\mathrm{p}(\theta) \in [0,1]$

Likelihood $\boldsymbol{p(\mathcal{D}|\theta)}$ → Likelihood of observing $\boldsymbol{\mathcal{D}}$ given $\boldsymbol{\theta}$

Posterior $\boldsymbol{p(\theta|\mathcal{D})}$ → Believe of parameters after observing data

Practice Questions

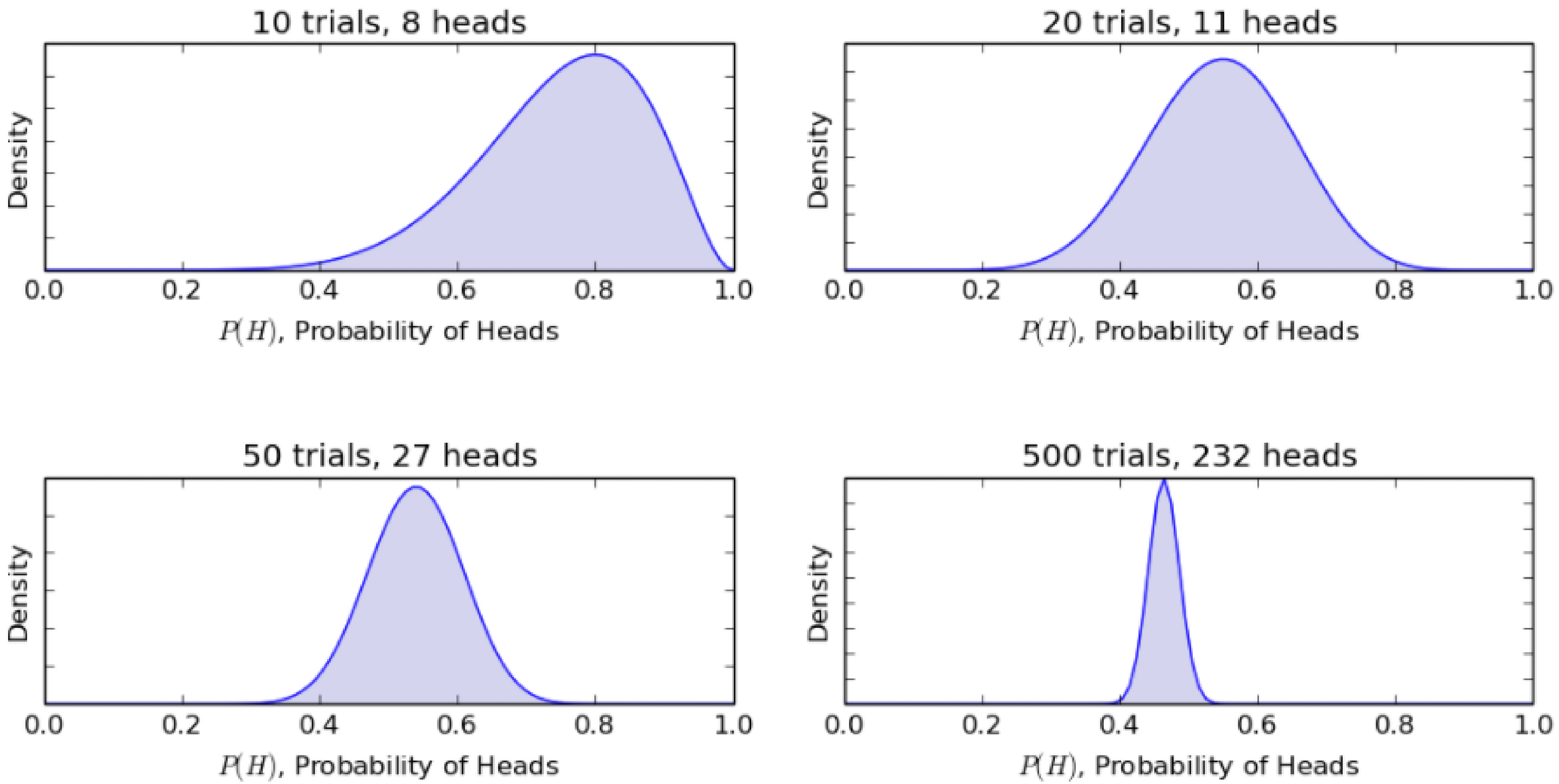
MLTS Exercise 02

What is a reasonable prior theta in the coin flip example?

?

Example

MLTS Exercise 02



Bayesian Linear Regression

Machine Learning Data Analytics FAU

Given the observed data $\mathcal{D} = \{x^{(n)}, y^{(n)}\}$, we assume to know the noise variance σ^2.

We would like to compute the posterior over the parameters, i.e,

$$p(w|\mathcal{D}, \sigma^2).$$

(We assume throughout a Gaussian likelihood model).

In linear regression **the likelihood is given by:**

$$p(y|X, w, \mu, \sigma^2) = \mathcal{N}(y|\mu + Xw,\ \sigma^2 I_N)$$

where μ is an offset term.

Bayesian Linear Regression

The conjugate prior of a Gaussian likelihood is also Gaussian*, which we will denote by

$$p(w) = \mathcal{N}(w|w_0, V_0).$$

Using the Bayes rule for Gaussian*, the posterior is given by

$$p(w|X, y, \sigma^2) \propto \mathcal{N}(w|w_0, V_0)\,\mathcal{N}(\mathrm{y}|\mathrm{Xw}, \sigma^2 I_N) = \mathcal{N}(w|w_N, V_N)$$

where

$$w_N = V_N V_0^{-1} w_0 + \frac{1}{\sigma^2} V_N X^T y$$

$$V_N = \sigma^2(\sigma^2 V_0^{-1} + X^T X)^{-1}$$

* See: Murphy K., „Machine Learning: A Probabilistic Perspective“ (2012)

Bayesian Linear Regression

The posterior predictive distribution at a test point x is given by

$$p(y|x, \boldsymbol{\mathcal{D}}, \sigma^2) = \int \mathcal{N}(y|x^T w, \sigma^2)\mathcal{N}(w|w_N, V_N)dw$$

$$= \mathcal{N}(y|w_N^T x, \sigma_N^2(x))$$

where $\sigma_N^2(x) = \sigma^2 + x^T V_N x$.

The variance in this prediction depends on the variance of the observation noise, σ^2, and the variance in the parameters, $\mathrm{V_N}$.

Bayesian Linear Regression

The marginal likelihood or evidence

$$p(y|X)$$

- is difficult to compute and
- a constant

Can be disregarded in the posterior computation.

But the marginal likelihood can be used to learn the parameters for the Bayesian Linear Regression model

→ See “*MLTS_Exercise_02_Maximize_Log_Marginal_Likelihood.pdf*” on StudOn

Bayesian Linear Regression

PyMC is a probabilistic programming library for Python that allows users to build Bayesian models with a simple Python API and fit them using Markov chain Monte Carlo (MCMC) methods.

https://www.pymc.io

Thank you for your attention!

Any questions?