Erasmus
School of
Economics

Statistics for Data Science
Lecture 7

Dennis Fok (Econometric Institute)

September – October, 2025

Erasmus University Rotterdam

Assignment 6.3/Take home assignment

- Use the website data
- Continue from In-class Assignment 6.3 and consider the logit model
- Predict the active probability for
  - `exog={'age': 40, 'income': 2000, 'region' : 1}`
  - `exog={'age': 40, 'income': 3000, 'region' : 1}`
- Calculate the difference in predicted probabilities
- Convert the difference into a single number by selecting the [0] element
- Construct the 95% confidence interval for this difference using bootstrap (at least 1000 times)

→ See also the example bootstrap code on Canvas

Before next time

- Nothing to read
- Reconsider/finish the in-class assignments of this week
- Look at (the code of) an additional example/exercise using binary data (next slide)
- Prepare questions for next time (final lecture!)
  - Theory
  - Applications
  - Exercises
  - Final assignment
  - Statistical challenges...
- You can already work on part 3 of the assignment

Plan for today

- Catch up with last week's material (GLM + Bootstrap)
- Bayesian statistics
- Wrap-up

Bayesian statistics

Background

Up to now we have studied Frequentist Statistics
$\rightarrow$ There is more!

The other approach to statistics is called Bayesian Statistics
Named after reverend Thomas Bayes (1702-1761)

Frequentist vs. Bayesian statistics

Concept of probability:

- Frequentist: probability is a "frequency in the long run"
- Bayesian: probability is a "degree of belief"

What are parameters?

- Frequentists: A parameter corresponds to a fixed (non random) population quantity
- Bayesians: Parameters are also random variables that have associated beliefs

Source of (parameter) uncertainty

- Frequentists: what would another sample have given us?
  $\rightarrow$ We need to consider hypothetical repetitions (=difficult?)
- Bayesians: how much information does the current sample bring us?
  $\rightarrow$ Beliefs can be updated

Parameter estimation/learning

Frequentist statistics

1. Get a point estimate
   - Minimize sum squared error, or
   - Maximize likelihood (or minimize deviance), or
   - Optimize . . .
2. Work out the (asymptotic) distribution (or use bootstrap) to get to know the uncertainty

Bayesian statistics

1. Start with a prior distribution for the parameter
   - Before looking at data what are your own subjective beliefs?
   - Code this as a distribution
2. Consider the information that the data brings (in the form of the likelihood)
3. Combine both sources of information (prior+likelihood) to update beliefs
   → Results in the posterior distribution
4. Posterior gives point estimate and full uncertainty

Advantages and disadvantages

Advantages Bayes

- Is always exact (does not require large samples/asymptotics)
  $\rightarrow$ Works well in small samples
- Is more intuitive
  - Bayesians **can** calculate the probability that a (null) hypothesis is true!
  - Updating information (learning) as data is collected is (conceptually) easy
- Allows for the inclusion of prior (eg. expert) information

Disadvantages Bayes

- Takes the distribution of the data more seriously in general (can be a strong assumption)
- Requires more computational effort (most of the time)
- Priors are subjective $\rightarrow$ others may not agree
- Formulating a good prior may be difficult

The mechanics

Combination of the two sources of information uses a theorem of Thomas Bayes
$\rightarrow$ Conditional probabilities/conditional densities

Rule of conditional probability

$$\text{Probability of event A given that event B happened} = \Pr[A|B] = \frac{\Pr[A \;\&\; B]}{\Pr[B]}$$
$$= \frac{\text{Probability of event A \textbf{and} B happening}}{\text{Probability of event B happening}}$$

Similar rule applies to densities

$$\text{conditional density} = f(y|x) = \frac{\text{joint density}}{\text{marginal density}} = \frac{f(y,x)}{f(x)}$$

Example of conditional probability

Probability of throwing a 4 with a fair dice given that the throw is even

$$\Pr[X = 4|X = \text{even}] = \frac{\Pr[X = 4 \;\&\; X = \text{even}]}{\Pr[X = \text{even}]} = \frac{\Pr[X = 4]}{\Pr[X = \text{even}]} = \frac{\frac{1}{6}}{\frac{1}{2}} = \frac{1}{3}$$

More difficult example:

Solution for the 3 door problem

Before choosing we know: Pr[Price in 1] = Pr[Price in 2] = Pr[Price in 3] = $\frac{1}{3}$ (*prior*)

Suppose I choose door 3 and Monty opens doors 1 (=*data*), we now want to know Pr[Price in 3|Monty opens 1]

Need to consider

- Pr[Monty opens 1|Price in 1] = 0 (he will not reveal the car)
- Pr[Monty opens 1|Price in 2] = 1 (he has no other choice)
- Pr[Monty opens 1|Price in 3] = $\frac{1}{2}$ (he can choose door 1 or 2)

Rules of conditional probability gives *posterior*

$$\begin{aligned}
\Pr[\mathrm{P}{=}3|\mathrm{M}{=}1] &= \frac{\Pr[\mathrm{P}{=}3 \text{ and } \mathrm{M}{=}1]}{\Pr[\mathrm{M}{=}1]} = \frac{\Pr[\mathrm{M}{=}1|\mathrm{P}{=}3]\Pr[\mathrm{P}{=}3]}{\Pr[\mathrm{M}{=}1]} \\
&= \frac{\Pr[\mathrm{M}{=}1|\mathrm{P}{=}3]\Pr[\mathrm{P}{=}3]}{\sum_{p=1}^{3}\Pr[\mathrm{M}{=}1 \textbf{ and } \mathrm{P}{=}p]} = \frac{\Pr[\mathrm{M}{=}1|\mathrm{P}{=}3]\Pr[\mathrm{P}{=}3]}{\sum_{p=1}^{3}\Pr[\mathrm{M}{=}1|\mathrm{P}{=}p]\Pr[\mathrm{P}=p]} \\
&= \frac{\frac{1}{2}\cdot\frac{1}{3}}{0\cdot\frac{1}{3}+1\cdot\frac{1}{3}+\frac{1}{2}\cdot\frac{1}{3}} = \frac{1}{3} \rightarrow \text{it is best to switch! Door 2 has probability } \frac{2}{3}.
\end{aligned}$$

Applied to learning a parameter β

Ingredients

- Prior: $f(\beta)$
  (eg. density of $\pi = \Pr[\text{head}]$)
- Likelihood $f(\text{data}|\beta)$
  (eg. prob. of observing $2\times$ head in two tosses given $\pi \to \pi^2$)
- Want to know *posterior* $f(\beta|\text{data})$
  (eg. density of π given that we observe 2 heads, 0 tails)

From Bayes Rule (twice)

$$f(\beta|\text{data}) = \frac{f(\beta, \text{data})}{f(\text{data})} = \frac{f(\text{data}|\beta)f(\beta)}{f(\text{data})} = c \times f(\text{data}|\beta)f(\beta),$$

where c can be seen as a constant

$\rightarrow$ *Posterior is proportional to prior* $\times$ *likelihood*

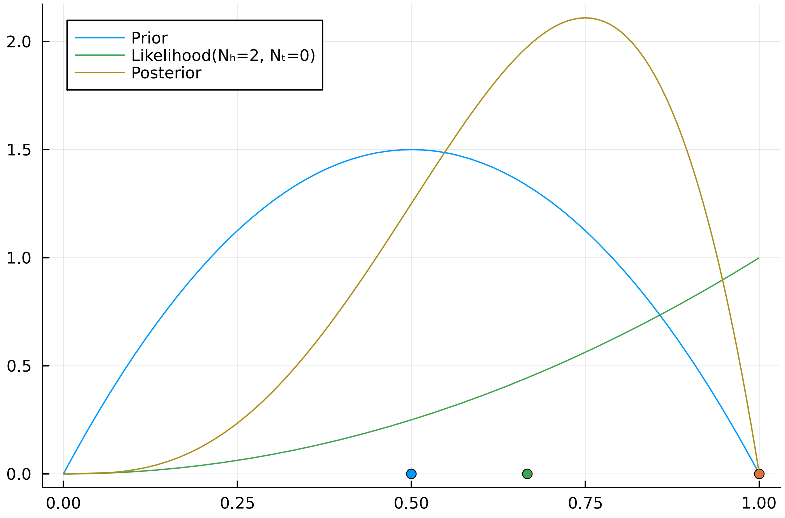
Posterior

The posterior codes everything that we know about β given the data
$\rightarrow$ we have the complete distribution!

We can obtain

- Posterior mean/median/mode
- Posterior variance ("estimation uncertainty")
- 95% credible interval (parameter will be in this interval with 95% probability)
- Probability that parameter exceeds x
- Probability that one parameter is larger than another
- ...

Example: coin tosses with a Beta prior (unknown coin)



Prior:
prob. heads $\sim$ Beta(2,2)

Data: 2 heads in two tries

Frequentist estimate:
prob. heads $=1$
(a bit extreme, not?)

Posterior:
prob. heads $\sim$ Beta(4,2)
posterior mean: $\frac{2}{3}$

In-class assignment

In-class assignment 7.1 (see starter code on Canvas)

In this assignment we further investigate the previous example

Step 1: investigate properties of the Beta(α, β) distribution

- When do you get a symmetric distribution?
- How do you code a belief that the probability is above 0.8?
- How do you code a belief that the probability is extreme (close to 0 or close to 1)?

Step 2: investigate the posterior given 100 observations

- For what setting of α and β does the posterior mean equal the max. lik estimator?
- What happens when $\alpha = \beta =$ high?
- What happens when $\alpha =$ large and $\beta =$ small?

Applications

Frequentist models have Bayesian equivalents
$\rightarrow$ Just add a prior!

Can do

- Linear model with prior
- Generalized linear model with prior
- ...

Added value of a prior

Prior has practical added value especially when *information* is limited

- Few observations
- Individual-specific parameters and few observations per individual
- Many parameters in a model (relative to data size)

Often prior is $N(\mu, \sigma^2)$

- μ codes the value that we expect a priori
  - can be a specific value (also mean across individuals)
  - often 0 (variable has no impact)
- σ^2 codes how certain we are (strength of information)
  - Small variance: we are really sure
    $\rightarrow$ Posterior will be relatively close to prior
  - Large variance: actually we do not know
    $\rightarrow$ Uninformative prior

Use cases (with links)

- New product development
- Product ranking (e.g., Amazon, Wayfair)
- A/B testing for e-mail designs, website strategies
- Stock price prediction (dealing with novel phenomena like Covid-19)
- Determining disease risk and medical diagnosis

Obtaining the posterior

- Sometimes easy
  - Prior and likelihood nicely "match"
    $\rightarrow$ Called a *conjugate prior*
  - Analytical results can be used
  - Eg. the coin toss example (Binomial distribution + Beta prior)

- Sometimes hard
  - Analytical results do not exist for the posterior
  - Sometimes iterative optimization methods can be used
  - General purpose solution: Simulation method using Markov Chain Monte Carlo (MCMC)
    - Simulate each parameter conditional on data and other parameters
    - Simulate each parameter in turn
    - Repeat for many iterations
    - Distribution of draws will eventually converge to the posterior distribution
    - Use draws (at the end of the sequence) instead of actual distribution
  - This is advanced material!

Bayesian analysis in Python

Options

- Code up all simulations yourself (rather difficult)
- Use specific packages: $\rightarrow$ there are many
- We focus a relatively easy to use option: the `bambi` interface to `PyMC`
  $\rightarrow$ To install `pip install bambi` (in a terminal within the correct virtual environment)

Bayesian linear model in Pyton using bambi

1. `import arviz as az`
   `import bambi as bmb`
2. `model = bmb.Model("y ~ x1 + x2", data)` → sets priors automatically
3. Can change priors by setting for example
   `p = {'x1': bmb.Prior("Normal", mu=0, sigma=1), 'x2': bmb.Prior("Normal", mu=0, sigma=1)}`
   `model = bmb.Model("y ~ x1 + x2", data, priors=p)`
4. Plot priors `model.build()`
   `model.plot_priors(draws=10000)`
5. Fit using default settings: `fitted = model.fit(random_seed=1234)`
6. Show draws: `az.plot_trace(fitted)` (in case you see trends in the trace plot → increase no. tune draws!)
7. Summarize results: `az.summary(fitted)`
8. Can extract draws for a specific parameter:
   `az.extract(fitted)["x1"].values`

Nonlinear models

Can also do other models

- Logit: `bmb.Model("y ~ x1 + x2", data, family="bernoulli")`
- Count/Poisson regression with `family="poisson"`
- etc (see documentation)

In-class assignment

In-class assignment 7.2 (see starter code on Canvas)

We consider data on "self-reported illegal drug use" as a function of Big-5 personality items

- Consider the example code to load the data
- Specify the model using
  - O = Openness to experience
  - C = Conscientiousness
  - E = Extraversion
  - A = Agreeableness
  - N = Neuroticism
- Inspect the automatically suggested prior: why is prior used?
- Generate and inspect the results
- (Experiment with the prior settings if you have time)

Wrap-up

Questions?

- Previous material
- Today's material
- Assignment
- Applications of statistics