Week 3, Lecture 5 - Bayesian vs. frequentist approaches

Aaron Meyer

Outline

- Administrative Issues
- Bayesian Statistics
- A Couple Examples

Based on slides from Joyce Ho.

Frequentist vs Bayesian

Frequentist

- Data are a repeatable random sample (there is a frequency)
- Underlying parameters remain constant during repeatable process
- Parameters are fixed
- Prediction via the estimated parameter value

Bayesian

- Data are observed from the realized sample
- Parameters are unknown and described probabilistically (random variables)
- Data are fixed
- Prediction is expectation over unknown parameters

Freq v. Bayes can hugely influence how we interpret the world

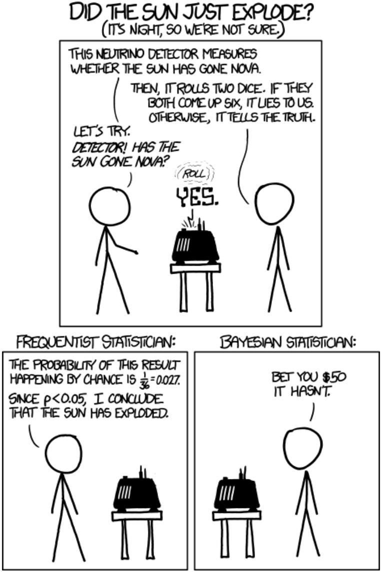


Figure: https://xkcd.com/1132/

Bayesian statistics

Bayesian - Derivation

Bayes' theorem may be derived from the definition of conditional probability:

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)}, \text{ if } P(B) \neq 0$$

$$P(B \mid A) = \frac{P(B \cap A)}{P(A)}, \text{ if } P(A) \neq 0$$

because

$$P(B \cap A) = P(A \cap B)$$

$$\Rightarrow P(A \cap B) = P(A \mid B)\,P(B) = P(B \mid A)\,P(A)$$

$$\Rightarrow P(A \mid B) = \frac{P(B \mid A)\,P(A)}{P(B)}, \text{ if } P(B) \neq 0$$

Classic Example: Binomial Experiment

- Given a sequence of coin tosses $x_1, x_2, \ldots, x_M$, we want to estimate the (unknown) probability of heads:

$$P(H) = \theta$$

- The instances are independent and identically distributed samples
- Note that x can take on many possible values potentially if we decide to use a multinomial distribution instead

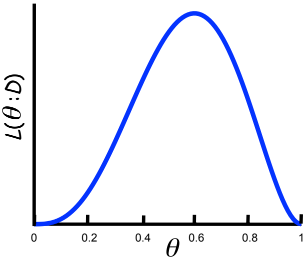
Likelihood Function

- How good is a particular parameter?
  - Answer: Depends on how likely it is to generate the data

$$L(\theta; D) = P(D \mid \theta) = \sum_m P(x_m \mid \theta)$$

- Example: Likelihood for the sequence: H, T, T, H, H

$$L(\theta; D) = \theta(1-\theta)(1-\theta)\theta\theta = \theta^3(1-\theta)^2$$



Maximum Likelihood Estimate (MLE)

- Choose parameters that maximize the likelihood function
  - Commonly used estimator in statistics
  - Intuitively appealing
- In the binomial experiment, MLE for probability of heads

$$\hat{\theta} = \frac{N_H}{N_H + N_T}$$

- Optimization problem approach

Is MLE the only option?

- Suppose that after 10 observations, MLE estimates the probability of a heads is 0.7, would you bet on heads for the next toss?
- How certain are you that the true parameter value is 0.7?
- Were there enough samples for you to be certain?

Bayesian Approach

- Formulate knowledge about situation probabilistically
  - Define a model that expresses qualitative aspects of our knowledge (e.g., forms of distributions, independence assumptions)
  - Specify a **prior** probability distribution for unknown parameters in the model that expresses our beliefs about which values are more or less likely
- Compute the **posterior** probability distribution for the parameters, given observed data
- Posterior distribution can be used for:
  - Reaching conclusions while accounting for uncertainty
  - Make predictions by averaging over posterior distribution

Posterior Distribution

- Posterior distribution for model parameters given the observed data combines the prior distribution with the likelihood function using Bayes' rule:

$$P(\theta \mid D) = \frac{P(\theta)P(D \mid \theta)}{P(D)}$$

- Denominator is just a normalizing constant so you can write it proportionally as:

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

- Predictions can be made by integrating with respect to posterior:

$$P(\text{newdata} \mid D) = \int_\theta P(\text{newdata} \mid \theta)P(\theta \mid D)$$

Revisiting Binomial Experiment

- Prior distribution: uniform for θ in [0, 1]
- Posterior distribution:

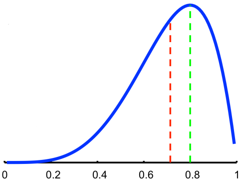
$$P(\theta \mid x_1, \ldots, x_M) \propto P(x_1, \ldots, x_M \mid \theta) \times 1$$

- Example: 5 coin tosses with 4 heads, 1 tail
  - MLE estimate:

$$P(\theta) = \frac{4}{5} = 0.8$$

  - Bayesian prediction:

$$P(x_{M+1} = H \mid D) = \int \theta P(\theta \mid D) d\theta = \frac{5}{7}$$



Bayesian Inference and MLE

- MLE and Bayesian prediction differ
- However...
  - IF prior is well-behaved (i.e., does not assign 0 density to any "feasible" parameter value)
  - THEN both MLE and Bayesian prediction converge to the same value as the training data becomes infinitely large

Features of the Bayesian Approach

- Probability is used to describe "physical" randomness and uncertainty regarding the true values of the parameters
  - Prior and posterior probabilities represent degrees of belief, before and after seeing the data
- Model and prior are chosen based on the knowledge of the problem and not, in theory, by the amount of data collected or the question we are interested in answering

Priors

- Objective priors: noninformative priors that attempt to capture ignorance and have good frequentist properties
- Subjective priors: priors should capture our beliefs as well as possible. They are subjective but not arbitrary.
- Hierarchical priors: multiple levels of priors
- Empirical priors: learn some of the parameters of the prior from the data ("Empirical Bayes")
    - Robust, able to overcome limitations of mis-specification of prior
    - Double counting of evidence / overfitting

Conjugate Prior

- If the posterior distribution are in the same family as prior probability distribution, the prior and posterior are called conjugate distributions
- All members of the exponential family of distributions have conjugate priors

Likelihood	**Conjugate prior distribution**	**Prior hyperparameter**	**Posterior hyperparameters**
Bernoulli	Beta	α, β	$\alpha + \sum x_i, \beta + n - \sum x_i$
Multinomial	Dirichlet	α	$\alpha + \sum x_i$
Poisson	Gamma	α, β	$\alpha + \sum x_i, \beta + n$

Linear Regression (Classic Approach)

$$y = w^\top x + \epsilon, \;\; \epsilon \sim N(0, \sigma^2)$$

$$P(y_i | w, x_i, \sigma^2) = N(w^\top x_i, \sigma^2)$$

$$P(y | w, X, \sigma^2) = \prod_i P(y_i | w, x_i, \sigma^2)$$

$$\Bigg\Downarrow \text{ maximize log likelihood}$$

$$\max \;\; \ln(P(y | w, x, \sigma^2) = \max \sum_i \ln(N(y_i | w, x_i, \sigma^2))$$

$$w_{\text{MLE}} = \text{argmin}_w \frac{1}{2} \sum_i (y_i - x_i^\top w)^2$$

$$w = (X^\top X)^{-1} X^\top y$$

Bayesian Linear Regression

- Prior is placed on either the weight, w, or the variance, σ
- Conjugate prior for w is normal distribution

$$P(w) \sim N(\mu_0, S_0)$$

$$P(w \mid y) \sim N(\mu, S)$$

$$S^{-1} = S_0^{-1} + \frac{1}{\sigma^2} X^T X$$

$$\mu = S \left(S_0^{-1} \mu_0 + \frac{1}{\sigma^2} X^T y \right)$$

- Mean is weighted average of OLS estimate and prior mean, where weights reflect relative strengths of prior and data information

Computing the Posterior Distribution

Analytical integration Works when "conjugate" prior distributions can be used, which combine nicely with the likelihood—usually not the case

Gaussian approximation Works well when there is sufficient data compared to model complexity—posterior distribution is close to Gaussian (Central Limit Theorem) and can be handled by finding its mode

Markov Chain Monte Carlo Simulate a Markov chain that eventually converges to the posterior distribution—currently the dominant approach

Variational approximation Cleverer way to approximate the posterior and maybe faster than MCMC but not as general and exact

Limitations and Criticisms of Bayesian Methods

- It is hard to come up with a prior (subjective) and the assumptions may be wrong
- Closed world assumption: need to consider all possible hypotheses for the data before observing the data
- Computationally demanding (compared to frequentist approach)
- Use of approximations weakens coherence argument

Bayesian statistics

Example problem - HIV test

- Rapid home tests will pick up an infection 97.7% of the time at 28 days after exposure (sensitivity).
- These same tests have a specificity of ~95%.
- 0.34% of the US population is estimated to be infected.

Given a positive test, what is the chance of the average person in the US being infected?
How would this change if ~10% of the population were infected?

Implementation - PyMC3

```
import pymc3 as pm

n = 100
heads = 61

with pm.Model() as coin_context:
    p = pm.Beta('p', alpha=2, beta=2)
    y = pm.Binomial('y', n=n, p=p, observed=heads)
    trace = pm.sample()

pm.summary(t, varnames=['p'])
```

Implementation - PyMC3

Output:

```
p:

  Mean             SD               MC Error         95% HP
  -----------------------------------------------------------

  0.615            0.050            0.002            [0.517

  Posterior quantiles:
  2.5            25             50             75
  |--------------|==============|==============|----------

  0.517          0.581          0.616          0.654
```

Implementation - emcee

```
import emcee

def lnprob(p):
    return lnprior(p) + lnobs(p, heads, n)

sampler = emcee.EnsembleSampler(nwalkers=3, ndim=1, lnprob=lnpro

sampler.run_mcmc(pos, 500)

samples = sampler.chain
```

Further Reading

- PyMC3 (python)
- emcee (python)
- Stan (C++, python, R)
- Bayesian Data Analysis
- Probabilistic Programming & Bayesian Methods for Hackers