In this lecture

- Inference
- Maximum likelihood
- Linear regression
- Compression
Inference and regression
An unfair coin toss

We toss a coin $N$ times, obtaining $n$ heads.

What is the probability distribution this events are drawn from?

What is the probability that toss $N+1$ is head?

What assumptions do we have to make?

$$\int_0^1 dp \, p^\alpha (1 - p)^\beta = \frac{\alpha! \beta!}{(\alpha + \beta + 1)!}$$
Normal distribution

\[ P(x | \mu, \sigma) = \frac{1}{\sqrt{2\pi \sigma^2}} \exp \left( -\frac{(x - \mu)^2}{2\sigma^2} \right) \]
Estimators

A function of observations predicting some quantities of interest:

$$\theta_n = f(x_1, \ldots, x_n)$$

Bias: \( \text{bias}(\theta_n) = \mathbb{E}[\theta_n] - \theta \)

Consider a Gaussian and

$$\mu_n = \frac{1}{n} \sum_{i=1}^{n} x_i, \quad \sigma_n^2 = \frac{1}{n} \sum_{i=1}^{n} (x_i - \mu_n)^2$$
Maximum likelihood

Choosing the parameters that maximize the likelihood $P(D|\theta, \mathcal{H})$

(often easier to work with logarithms)

$$\theta_{ML} = \arg \max_\theta \ \log P(D|\theta, \mathcal{H}) = \arg \max_\theta \ \sum_i \log P(x_i|\theta, \mathcal{H})$$

[cf. minimizing the cross entropy]

Calculate for a Gaussian the MLE of mean and variance.

[for proper marginalization see McKay chap. 24]
Linear regression

Given \((\vec{x}_i, y_i)\), find the best linear function describing their relation.

\[ \hat{y}_i = \vec{w} \cdot \vec{x}_i \]

Least squares: minimize the estimator \( C(\vec{w}) = \sum_i (y_i - \vec{w} \cdot \vec{x}_i)^2 \)

In matrix form: \( C(\vec{w}) = (X\vec{w} - y)^\top (X\vec{w} - y) \)

One could do numerically or take derivative: \( w_{\text{min}} = (X^\top X)^{-1} X^\top y \)

MLE: Assume a Gaussian per point, with means from linear equation.
Data compression
Why compression?

- Entropy and information content
- Redundancy
- Typicality and dimensionality
- Representation
Intuitive examples

How would you compress the following strings?

- 000000000000000000100000000010000000100000000000100000000010000000001
- 11110000111000011110000
- 1111000011100001100110011110000111100001

What is the best way of guessing a number between 0 and 63?
Typicality

What is the probability that a binary string of N=10 bits with $P(1)=0.1$ contains two “1”s?

What if N=100 and 20 “1”s?

What is the distributions of “1”s in general?

What does a typical string look like? What is its information content?
Shannon’s source coding theorem

N i.i.d. random variables each with entropy \( H(X) \) can be compressed into more than \( N H(X) \) bits with negligible risk of information loss, as \( N \to \infty \); but conversely, if they are compressed into fewer than \( N H(X) \) bits it is virtually certain that information will be lost.
Symbol codes

We encode each element of a set with a (variable length) symbol.

If the 4 elements of X have probability \{0.5, 0.25, 0.125, 0.125\}, what is the optimal length of symbols for each element?

What is the expected length \( L(C, X) = \sum_x P(x)l(x) \)?

Can we shorten all messages?

What are symbol codes missing? → Arithmetic codes
Bonus topic: hashes

How to efficiently store a mostly empty database?

How to check the integrity of a file?

How to safely check a password?
That’s all for now...