S1: Principles of Data Science

Problem Sheet 4

MPhil in Data Intensive Science

Matt Kenzie
mk652@cam.ac.uk

Michaelmas Term 2023

Problem Sheet 4

Lectures 19 – 24
Topics covered: advanced methods, forward modelling, Gaussian mixutre models, sWeighting

1. Familiarise yourself with bootstrapping and jackknife resampling. By generating a small data sample from a normal distribution, demonstrate that the bias in the sample variance (without the Bessel correction) can be obtained by a resampling.

2. Assume the standard example of fitting a small peaking signal on top of a large smoothly falling background. Show that if the true background distribution is of the form $y = x^{-a}$ then the signal yield estimate using a MLE approach will be biased (and have the incorrect coverage) if one assumes the background is distributed as $y = e^{-x}$.

3. The Brookhaven AGS experiment measured the relative decay angle between opposite sign pions produced from a beam of neutral long lived kaons, K^0_{L}. Typically K^0_{L} mesons will decay into three pions (π^+, π^-, π^0), thus the distribution of the decay angle between the charged pair will be flat. The experiment was searching for two pion decays of the K^0_{L} meson, in which case the decay angle between the charged pair is π.

   (a) If the total proportion of 2 pion decays is given by the parameter f, the angular resolution of the detector (in terms of the cosine of the decay angle) was approximately $\sigma(\cos\theta) \approx 0.05$, and the expected total number of events observed was 5000, what value of f can be excluded at 90%C.L. using the CL_{sb} method?

   (b) What is the value for the CL_s method?

   (c) Take a look at the sample in `datasets/kaon.npy` which is an array of measurements of $\cos(\theta)$ mimicking the data taken at the Brookhaven AGS experiment. What value of f do you determine from this sample? What scientific conclusion does this lead you to?

4. Kernel Density Estimation. There is an array of datapoints saved in `datasets/peaking.npy`. They have a distribution which looks like it peaks.

   (a) Try fitting these points with a normal distribution. What do you think of the outcome?

   (b) Try fitting these points with a sum of two normal distributions. Is it looking any better?

   (c) What about a sum of three? Can you find any empirical distribution which fits ok?

   (d) Now try using a kernel density estimate with bandwidths of 0.1, 0.2, 0.5, 1 and chosen using the Scott algorithm. Which do you think is best?

5. Expectation maximisation on a biased coin flip. Imagine that you have two biased coins, called A and B. The probability that coin A gives a head is θ_A and the probability that coin B gives a head is θ_B (where θ_A and θ_B may not necessarily be 0.5). Imagine you have a collection of 5 datasets in which first one of the coins is selected (with probability of 0.5), you can call this random variable Z, and then the chosen coin is flipped 10 times, this random variable can be X. You have the obtained data in terms of X (which I put below) but you do not know for a given dataset whether coin A or coin B was used, thus Z is a hidden variable. Use an expectation maximisation procedure to iteratively converge on estimates of $\hat{\theta}_A$ and $\hat{\theta}_B$ given the data below. If it helps this data is also stored in `datasets/biased_coin_flip.npy`.

Set	Outcomes
1	T H H H H H H H T H
2	T T T T T T H T H T
3	H T T H H H H H H H
4	T T H T T H H H T H
5	H T H H T T H H H H

6. A Gaussian Mixture Model question. Take a look at the data in `datasets/gmm.npy`. This is an array of two dimensional data which is generated from some number of multivariate normal distributions (with different means and covariances). Use a Gaussian Mixture Model approach to determine how many separate underlying multivariate normal distributions you think there are, and estimate the means and covariances of those underlying distributions.

7. An *sWeights* question. Take a look at the data in `dataset/sweights.npy`. This is an array of two dimensional data containing two components (a signal and a background). The first dimension will be considered our "discriminant" dimension (we'll fit this one). The second dimension is the "control" dimension which we want to extract the signal properties of.

   (a) First you will need to use some estimation procedure to estimate the parameters of the signal and background models when fitted to dimension X. The background is a log normal distribution with free parameters, s, μ_b and σ_b. The signal is a log gamma distribution with free parmaeters, c, μ_s, σ_s. The last free parameter is the signal fraction f. You will also need to incorporate an overall normalisation parameter N, which can be done by parameterising $Ns = fN$ and $Nb = (1-f)N$.

(b) Once this fit has been established you should extract *sWeights* using the estimated yields and shapes in the discriminant dimension X. Apply these weights to the data in the control dimension Y. The signal distribution in Y is an exponential decay. Use the *sWeights* to determine the slope parameter. Notice how you never need an explicit parameterisation of the background distribution in Y in order to do this. For reference I find an estimate of the slope parameter to be $\hat{\lambda} = 2.38 \pm 0.03$.