

## Lecture 2, Friday w1, 2014-09-19

### Main lecture points:

- We cannot compress *all* files. So we choose to shrink the probable ones.
- Use a model to describe what's probable. *For now*, the model is fixed, assumed known.
- Simplest model: string of independent Bernoulli bits.
- Binomial: how many bits are on ( $k = \sum_n x_n$ ) in a length- $N$  bit-string?
- Take logs of large/tiny positive numbers if using floating point doubles.
- Block coding idea.

### Check your progress

Things you should be able to do and answer after reviewing the lecture:

- Sketch the binomial distribution for various  $N$  and  $p$ .
- Explain what the majority of binary strings of length  $N$  are like, using the sketch for  $p = 0.5$ .
- What is the most probable binary string of length  $N$  from an independent Bernoulli model with  $p < 0.5$ ?
- What is a *typical* string of length  $N$  like, assuming an independent Bernoulli model with  $p < 0.5$ ?
- How can it be that a 'typical' string is so much less probable than the most probable string?
- Compute binomial probabilities for large  $N$  and  $k$ .
- Understand idea of splitting list of all possible files, sorted by probability into two. Give most probable items a short code, and less probable items a longer one.

### Recommended reading

We have now covered most of the material in the 'week 1' slides (I didn't quite finish, but will next time). Ask on NB if anything is unclear, or too compressed.

Make sure you have done the maths background reading from the last lecture.

## **For keen people**

Being able to quickly compute and plot things is a useful research skill. Last time I suggested reproducing a plot of repetition code properties. This time you could try reproducing plots showing binomial distributions and the performance of block codes. It's possible you'd hit numerical or other problems, which I'd be happy to help with.