浙江大学爱丁堡大学联合学院

ZJU-UoE Institute

ADS2 Lecture 2 Intuition and Probability

Dr Duncan MacGregor duncan.macgregor@ed.ac.uk

Semester 1, Week 2

2023-24

Intuition and Probability

Objectives for this week

- Start thinking about numbers and probabilities
- Look at probability distributions and how to plot these using histograms
- Use random number generation to simulate data

The world in numbers

There are many types of data, but most of what we work with is numbers

How do we translate the world into numbers?

- mostly by measuring and counting
- but often too many to count, or things are difficult to measure

Important to maintain sense of what the numbers mean

Use your intuition!

The world in numbers

Exploring the world with numbers

- What is my question?
- What can I count or measure?
- Or what data is already available?
- How do those numbers relate to the question?

Questions are often complicated (often more so than you first realise)

How can you break the question down into things you can count or measure?

The world in numbers

How do you judge an answer?

- How good are the measurements?
- Have we done enough counting?
- Is X different from Y?

Often we use probability

Important in science for judging the value of numbers

Thinking about numbers and probabilities

Background

- Thinking about the world in terms of numbers and probabilities is an important skill.
- This skill can be practised by estimation exercises: You are asked to give a rough estimate of a large(-ish) number.
- Solve these without using the internet, just using your awesome brain (and that of the person next to you).
- Using a calculator is allowed.

Exercise 1: Family

You share a common ancestor with the person sitting next to you. How long ago did that common ancestor live?

Thinking about numbers and probabilities

Exercise 2: Walking

How long would it take to walk from Haining to Edinburgh?

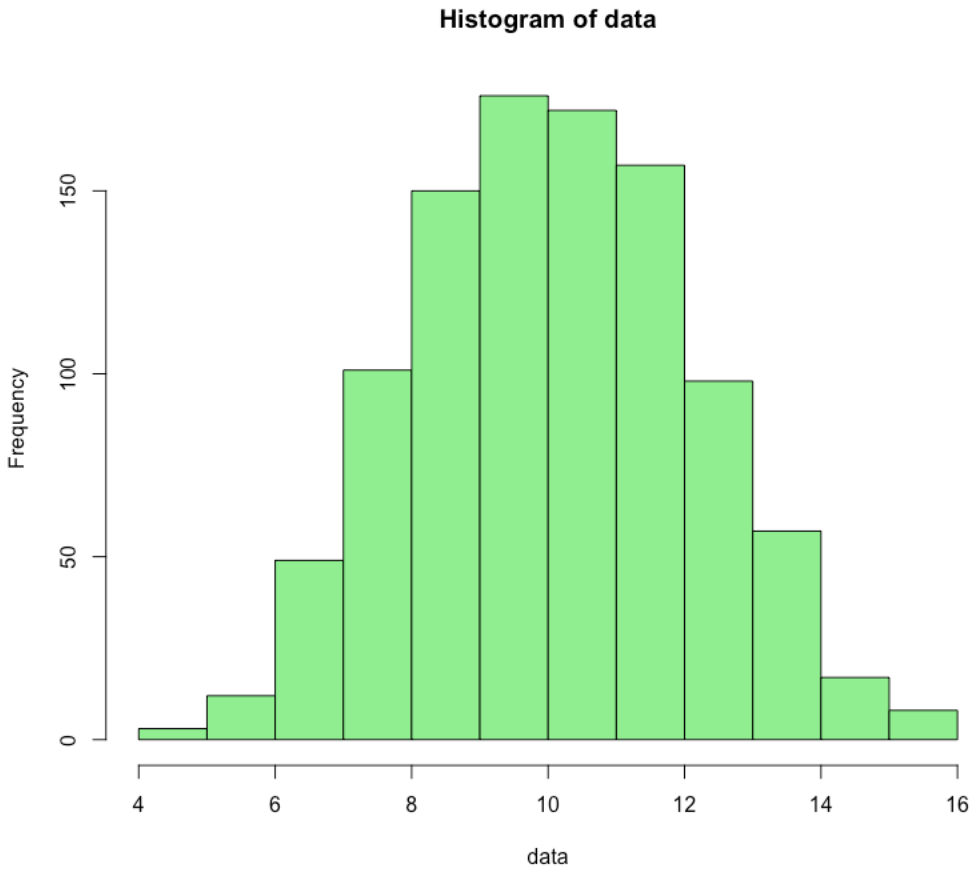
Intuition and Probability

Objectives for this week

- Start thinking about numbers and probabilities
- Look at probability distributions and how to plot these using histograms
- Use random number generation to simulate data

Your friend the histogram

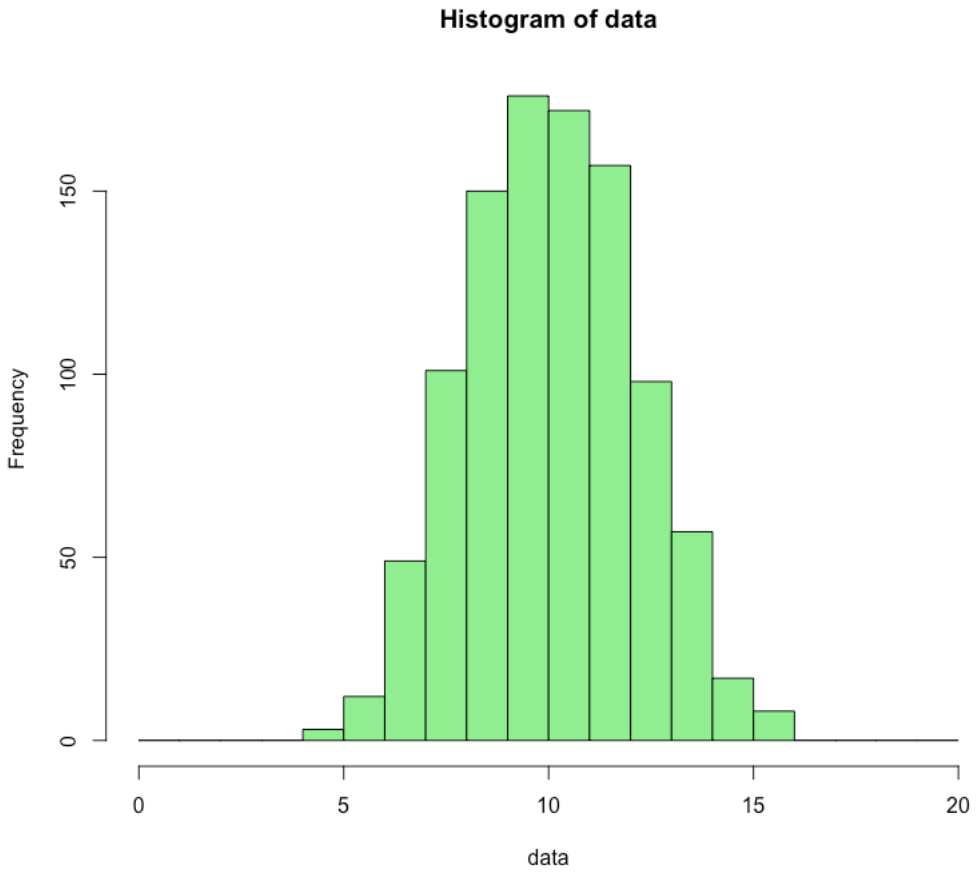
```
12.271446  9.633233  8.194364  6.685447 12.175688 12.012277
10.639707 10.396930  9.087270 11.721358  6.374044 12.495361
11.880212  8.400324  7.955587 11.205404 11.800200 13.951175
 7.620520  9.909170 10.925550 10.856680  8.418916  7.777094
 6.902583 10.752104  8.502821  7.485077  9.189830  6.768223
12.690519 11.388399  7.097011 12.070794 12.085875  8.035717
11.345543  9.054798  8.136323 11.220868 10.995553  9.886602
 9.215344  9.841829  7.930220 10.766767  9.975863  7.847753
10.695474 10.363516  5.167790  9.557041 11.104517 12.517539
11.711477  9.304271  7.039410  8.588905 11.471497 11.064157
 9.631632  7.827141  8.225722 10.971833 10.717128 12.484750
11.142114  8.043979  7.579542 10.257687  7.469888  9.188303
10.421474  8.839554  9.712336 11.575511 11.918475  9.440598
 9.699228 12.681182  7.060329 11.076885  7.597787 12.383183
 8.909318  6.823100 11.096187  8.961050  8.613483 11.269421
 9.083466  8.310002 12.022800  5.596728  7.104118 11.255457
 9.167674  6.779713 11.074817  7.260063  9.933101  6.223882
 7.858558 10.185972 13.423760  8.570353 10.776498 12.374886
12.431057 11.569958  8.668113  7.729392  9.678689 11.565184
 7.958211  7.159852 10.651330 13.915170 11.552451  9.151422
 7.732577 10.637062 10.737179  7.504574  9.937415  8.031942
```




```
data = rnorm(1000, 10, 2)
hist(data, 10, col = 'lightgreen')
```

Histogram bins

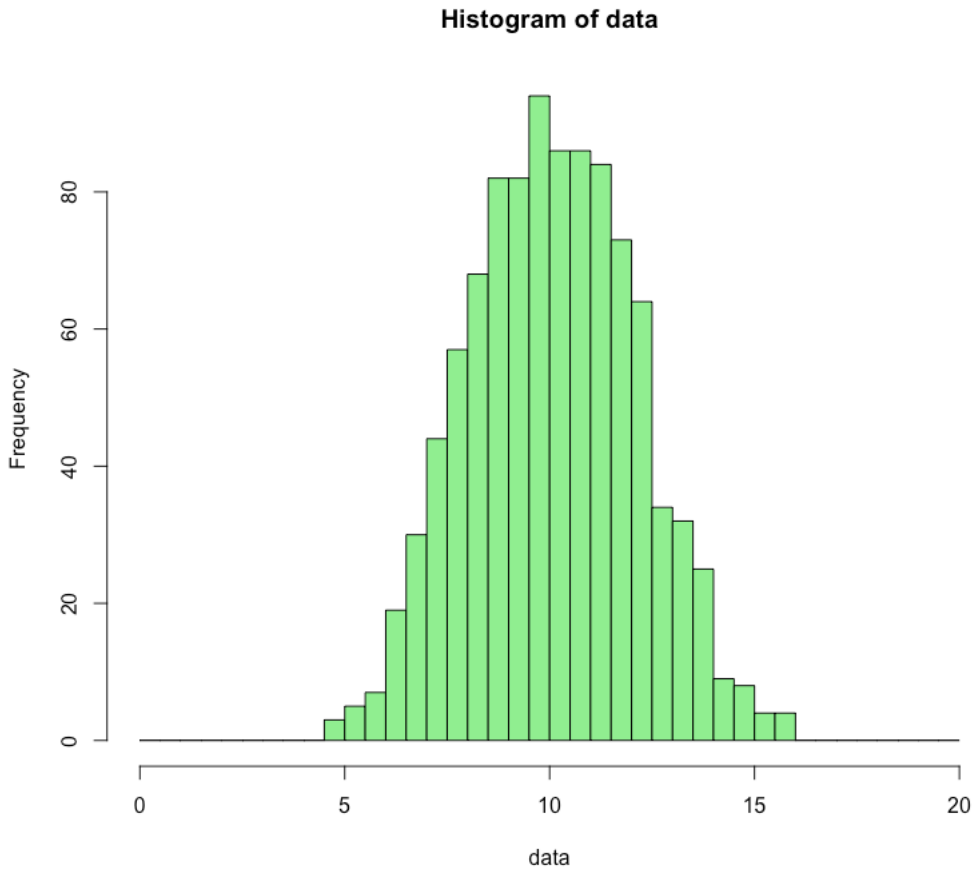
```
12.271446  9.633233  8.194364  6.685447 12.175688 12.012277
10.639707 10.396930  9.087270 11.721358  6.374044 12.495361
11.880212  8.400324  7.955587 11.205404 11.800200 13.951175
 7.620520  9.909170 10.925550 10.856680  8.418916  7.777094
 6.902583 10.752104  8.502821  7.485077  9.189830  6.768223
12.690519 11.388399  7.097011 12.070794 12.085875  8.035717
11.345543  9.054798  8.136323 11.220868 10.995553  9.886602
 9.215344  9.841829  7.930220 10.766767  9.975863  7.847753
10.695474 10.363516  5.167790  9.557041 11.104517 12.517539
11.711477  9.304271  7.039410  8.588905 11.471497 11.064157
 9.631632  7.827141  8.225722 10.971833 10.717128 12.484750
11.142114  8.043979  7.579542 10.257687  7.469888  9.188303
10.421474  8.839554  9.712336 11.575511 11.918475  9.440598
 9.699228 12.681182  7.060329 11.076885  7.597787 12.383183
 8.909318  6.823100 11.096187  8.961050  8.613483 11.269421
 9.083466  8.310002 12.022800  5.596728  7.104118 11.255457
 9.167674  6.779713 11.074817  7.260063  9.933101  6.223882
 7.858558 10.185972 13.423760  8.570353 10.776498 12.374886
12.431057 11.569958  8.668113  7.729392  9.678689 11.565184
 7.958211  7.159852 10.651330 13.915170 11.552451  9.151422
 7.732577 10.637062 10.737179  7.504574  9.937415  8.031942
```




```
hist(data, c(0:20), col = 'lightgreen')
```

Histogram bins

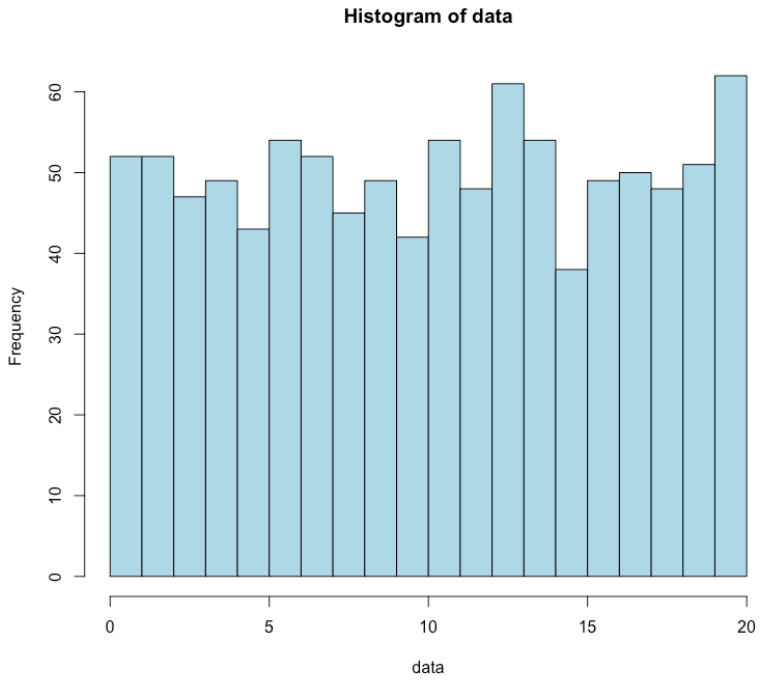
```
12.271446  9.633233  8.194364  6.685447 12.175688 12.012277
10.639707 10.396930  9.087270 11.721358  6.374044 12.495361
11.880212  8.400324  7.955587 11.205404 11.800200 13.951175
 7.620520  9.909170 10.925550 10.856680  8.418916  7.777094
 6.902583 10.752104  8.502821  7.485077  9.189830  6.768223
12.690519 11.388399  7.097011 12.070794 12.085875  8.035717
11.345543  9.054798  8.136323 11.220868 10.995553  9.886602
 9.215344  9.841829  7.930220 10.766767  9.975863  7.847753
10.695474 10.363516  5.167790  9.557041 11.104517 12.517539
11.711477  9.304271  7.039410  8.588905 11.471497 11.064157
 9.631632  7.827141  8.225722 10.971833 10.717128 12.484750
11.142114  8.043979  7.579542 10.257687  7.469888  9.188303
10.421474  8.839554  9.712336 11.575511 11.918475  9.440598
 9.699228 12.681182  7.060329 11.076885  7.597787 12.383183
 8.909318  6.823100 11.096187  8.961050  8.613483 11.269421
 9.083466  8.310002 12.022800  5.596728  7.104118 11.255457
 9.167674  6.779713 11.074817  7.260063  9.933101  6.223882
 7.858558 10.185972 13.423760  8.570353 10.776498 12.374886
12.431057 11.569958  8.668113  7.729392  9.678689 11.565184
 7.958211  7.159852 10.651330 13.915170 11.552451  9.151422
 7.732577 10.637062 10.737179  7.504574  9.937415  8.031942
```




```
hist(data, c(0:40)*0.5, col = 'lightgreen')
```

The uniform distribution

```
11.186817 12.450132 12.782673  6.561106  1.853700  4.316020
13.721204  5.231991 17.666000 17.874138  0.592516  1.313807
 4.383799  5.642425 11.344731 18.595386 12.589535 17.249680
11.957210  4.295247 12.974101  3.827229 13.586250  2.072798
 6.908788  3.604994 10.635333 16.587864  8.705957 18.489899
 5.136630  7.644433 15.493930  6.322784  2.557375  1.064126
16.964228  7.215027 12.082244 13.975465  2.729795 16.180269
 1.903075 10.749250  4.252019 13.322500 18.465167  2.172368
13.760972  2.821952  9.381043 18.538593 12.049212 11.245596
16.063295 19.490154  0.922816 16.840722 19.586792 12.680507
17.661684 17.621061  1.846665  8.698954  8.688303  1.302692
 0.404462  6.657587 12.850178 14.699415  8.718683  1.205261
13.964659 16.467781  0.809125  0.771790  5.790395  8.482069
 8.021103 19.703101 19.111370  7.718273 18.689045 18.960603
14.353359 17.628809 19.764675  3.292990 18.802589  6.380616
 1.745618 13.805511 19.886739 12.337743  7.062120 16.303174
16.861953  8.036585 15.146375  8.787403 12.563657  5.966036
19.370305 15.166851  3.560012  8.946136 18.630866 17.478975
 7.614810 15.703358  3.671887 18.438645  6.394097  4.993337
13.376519  1.337625  8.378179 15.671987  7.128447  6.175446
 6.700135 15.981472 13.197896 15.746671  5.141954 15.068856
 0.256164 10.295154 12.615397  1.228090  8.933571 17.863016
```



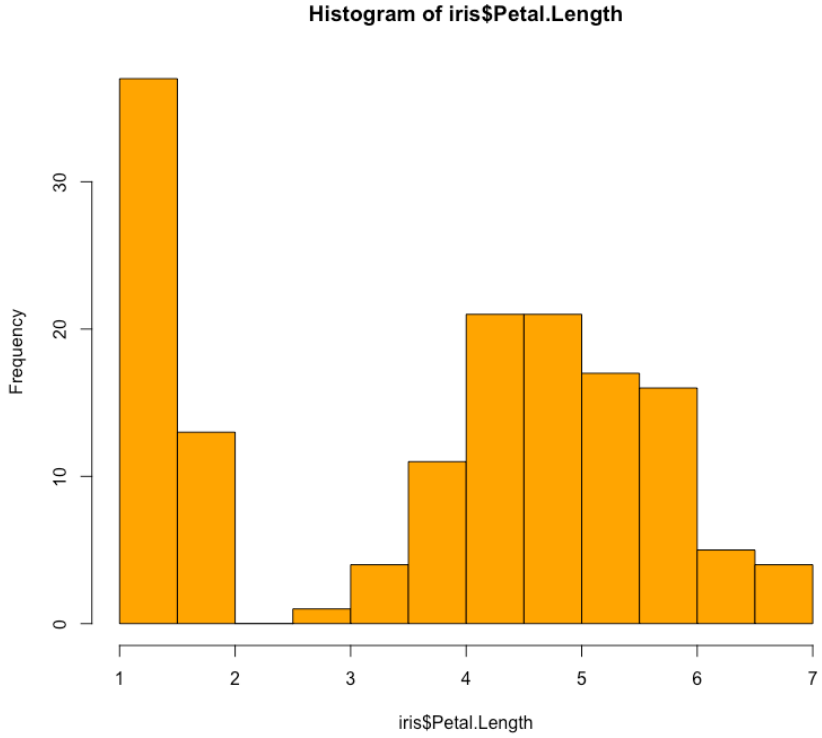

```
data = runif(1000, 0, 20)

hist(data, c(0:20), col = 'lightblue')
```

The iris dataset

```
> head(iris)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
> tail(iris)
    Sepal.Length Sepal.Width Petal.Length Petal.Width   Species
145          6.7         3.3          5.7         2.5 virginica
146          6.7         3.0          5.2         2.3 virginica
147          6.3         2.5          5.0         1.9 virginica
148          6.5         3.0          5.2         2.0 virginica
149          6.2         3.4          5.4         2.3 virginica
150          5.9         3.0          5.1         1.8 virginica
```

The iris dataset

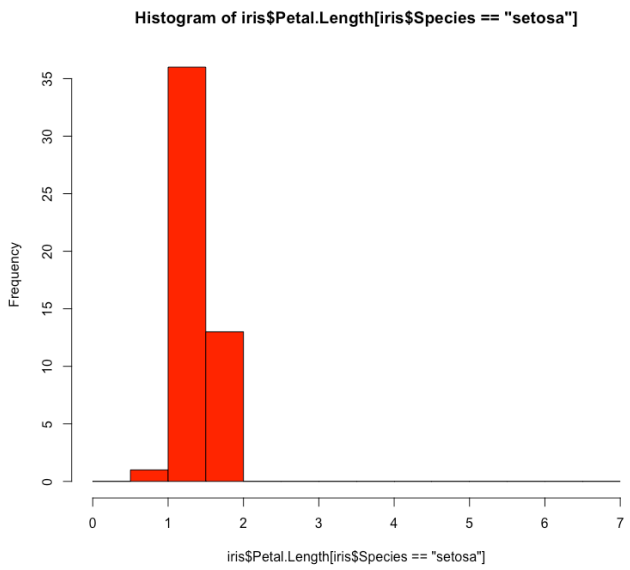

```
hist(iris$Sepal.Length, col = 'pink')
```

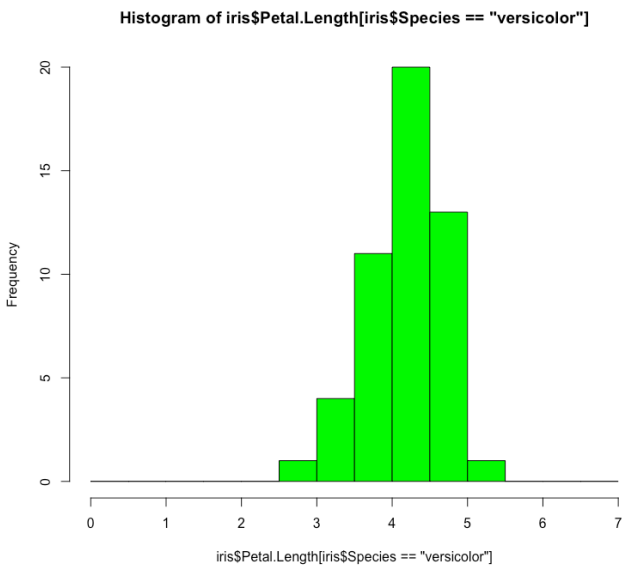


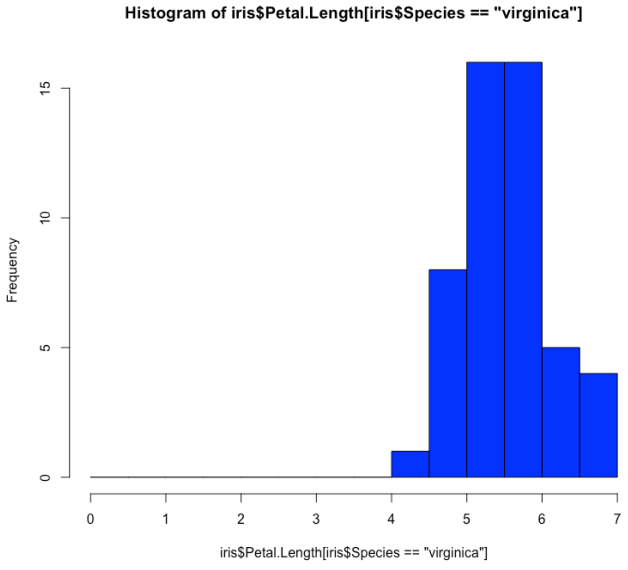

```
hist(iris$Petal.Length, col = 'orange')
```

The iris dataset

```
> unique(iris$Species)
[1] setosa     versicolor virginica
Levels: setosa versicolor virginica
> summary(iris$Petal.Length)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   1.600   4.350   3.758   5.100   6.900
```

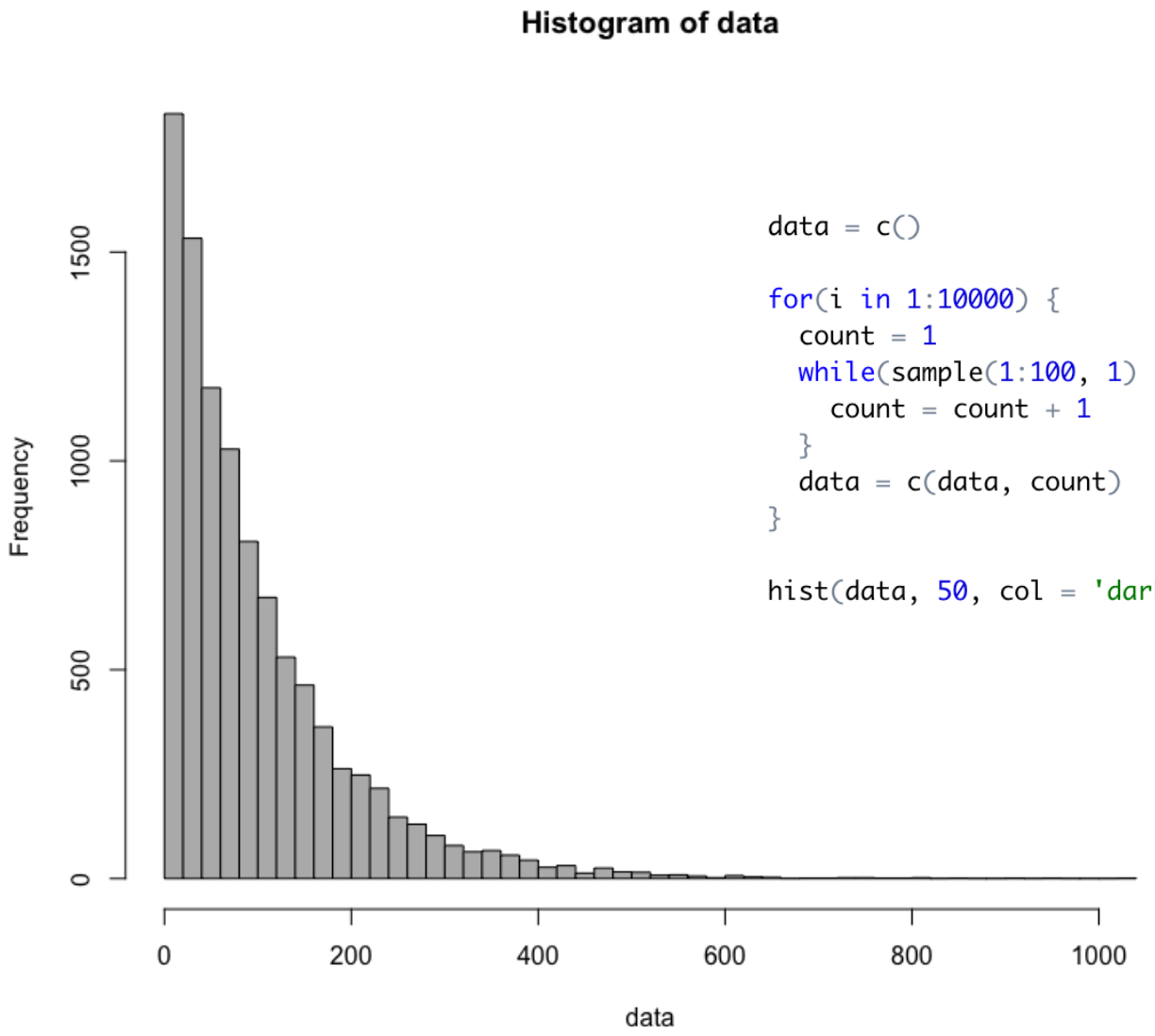







```
hist(iris$Petal.Length[iris$Species == 'setosa'], c(0:14)*0.5, col = 'red')

hist(iris$Petal.Length[iris$Species == 'versicolor'], c(0:14)*0.5, col = 'green')

hist(iris$Petal.Length[iris$Species == 'virginica'], c(0:14)*0.5, col = 'blue')
```

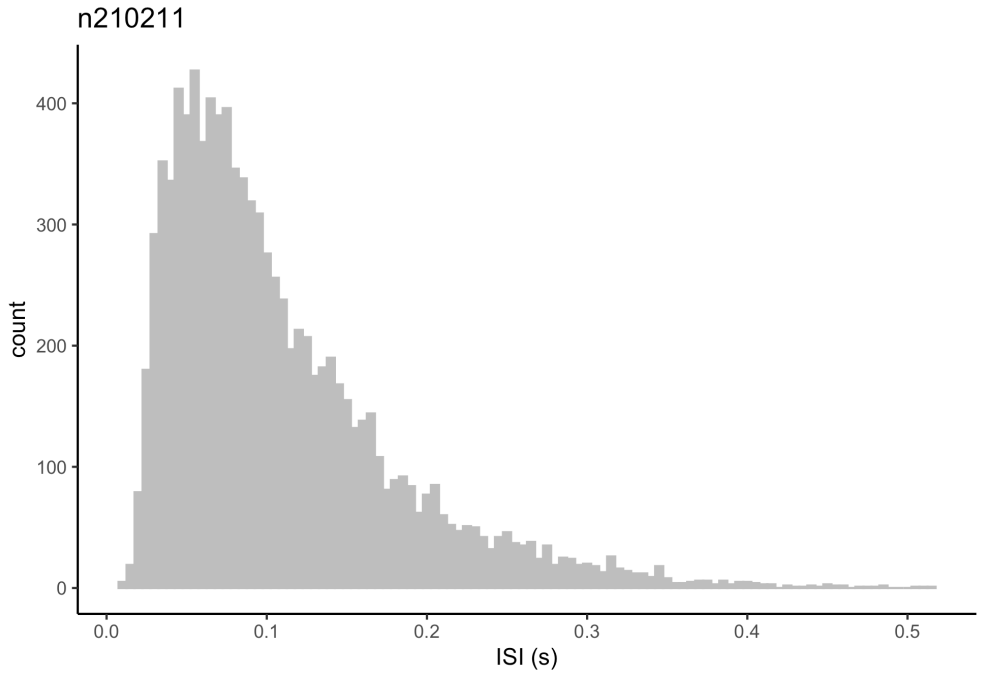
The Poisson distribution



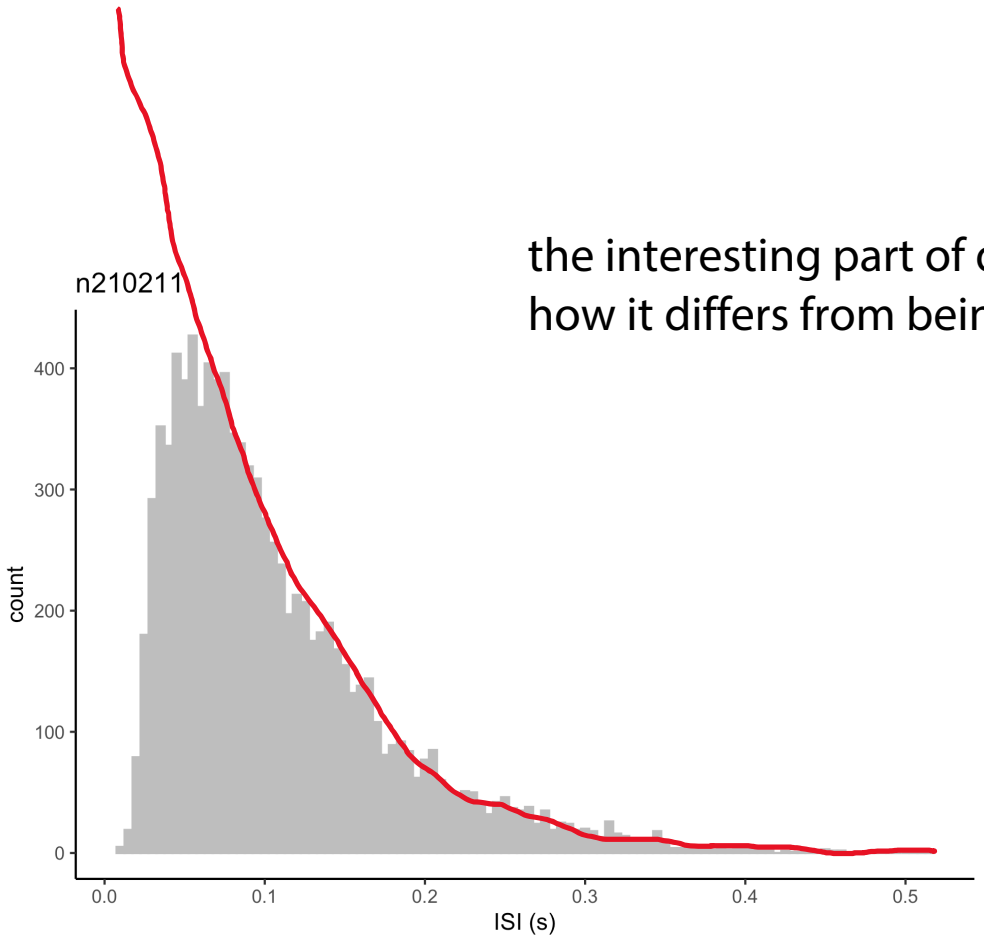

```r
data = c()

for(i in 1:10000) {
  count = 1
  while(sample(1:100, 1) != 100) {
    count = count + 1
  }
  data = c(data, count)
}

hist(data, 50, col = 'darkgrey')
```

Analysing a neuron

inter-spike interval histogram



Analysing a neuron



the interesting part of data is often how it differs from being random

Any questions?

- Talk to each other
- Use the discussion board!
- Email me duncan.macgregor@ed.ac.uk

Objectives for this week

- Start thinking about numbers and probabilities
- Look at probability distributions and how to plot these using histograms
- Use random number generation to simulate data