Statistics and Data Science: Lecture 1

Roger Barlow
Huddersfield University

Cockcroft Lecture Series

23^{rd} May 2022

Statistics

Descriptive Statistics	Inferential Statistics
Summarise your dataset (which may be large and/or complicated) in a few numbers or plots for your audience (who may be readers, listeners, pupils, grant funding bodies...)	What does the data tell us about the process that produced it?
Something of an art	*Very much a science*
This lecture (mostly)	Second lecture

Programming languages

Compiled: for simulations and number-crunching

- Fortran
- C++ (C, C#)

Interpretive: for analysing the results and drawing plots

- Python
- R
- MATLAB (Octave)
- Visual Basic
- ROOT

Plus a whole lot of others: Java, Javascript, Haskell, Swift ...

I'll assume you are working with Python or R or MATLAB

Data

Sitting on a file somewhere, either as text (slower and larger) or as binary (harder to handle)

Statistics

- Quantitative (Numeric)
  - Continuous
  - Discrete
  - Ranked
- Qualitative (Categoric, also known as factors)

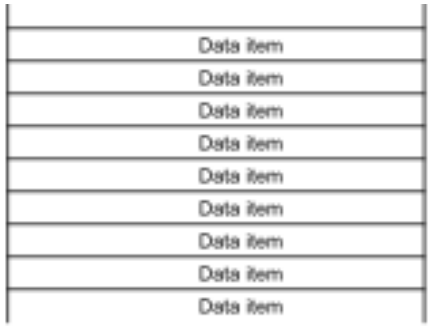
Computing

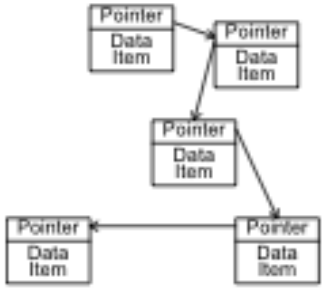
- Numeric
  - Integer
  - Real (floating point or 'double')
- Boolean (logical)
- Character String

Python distinguishes reals and integers but does so in sneaky silence.
Try `x=3; print(10/x)` and then `x=3.0; print(10/x)`
R and MATLAB default to double. Integers are left for experts.
Low level languages (Fortran and C/C++) distinguish single characters and character strings. Higher level languages just use strings

Basic Data Structures

Arrays and lists





Array

A block of memory allocated to data
All data objects of the same type and size
Fast
Hard to add or delete items
Indexed from 1 or 0 depending on language

List

A set of nodes, each containing a data object and a pointer to the next node (or null at the end)
Objects can be of different type
Easy to add and delete items
Slow (more operations and memory caching lost)
Very general and can get complicated (lists of lists...)

More Advanced Data Structures

Vectors, Matrices

Vectors

An array that knows its own length
Basic in R.
Special case of Matrix in Matlab.
Not in basic Python but import from `numpy` - called `array`.

Matrices

A vector addressable by 2 (or more, or less) indices
In R: a vector with a defined dimension `dim(whatever)`
Fundamental in Matlab ("MATrix LABoratory")
Also in `numpy` and also called `array`

Stacks

Used in program control but not for data

Even More Advanced Data Structures

Tuples, Data frames

Tuples

A specified set of data, possibly of different types, stored in a continuous block of memory

Python: like lists but use round brackets instead of square

```
nt=(1,2,3,'red',4)
```

Not easily available in R or MATLAB

Data Frames

A vector of tuples

Looks like an old-style table in a lab notebook

Columns may be numeric or factors (strings)

Rows are all alike

In Python, import from `pandas`

In Matlab, these are called tables

Measures of location

Summarising a set of numbers with just one value

```
665384.04
268596.46
163694.74
147039.71
121590.30
113594.08
110571.82
104093.67
98378.54
97813.01
96713.51
86935.83
84896.88
83568.95
82278.36
77347.81
77115.68
71297.95
70698.32
69898.87
69706.99
65757.70
65496.38
59425.15
57913.55
56272.81
54554.98
53819.16
53178.55
52420.07
52378.13
48431.78
46054.34
44825.58
42774.93
42144.25
40407.80
36419.55
35379.74
32020.49
24230.04
12405.93
10931.72
10554.39
9616.36
9349.16
8722.58
5543.41
2488.72
1544.89
```

The (arithmetic) mean $\frac{1}{N}\sum_i x_i$
The median: Half above and half below
The mode: The most popular

Python (after `import numpy as np` and `from statistics import *`)

```
with open("data1.txt") as f:
    a=f.readlines()
ar=np.array(a)
data=ar.astype('float')

print("mean ",mean(data)," median ",median(data))
```

R

```
df=read.table("data1.txt")

print(paste(" mean ",mean(df$V1)," median",median(df$V1)))
```

MATLAB

```
fID=fopen("data1.txt");
[data,count]=fscanf(fID,'%f',[1 Inf]);
fprintf(' mean %f median %f \n',mean(data),median(data))
```

Also - geometric mean, harmonic mean. Which to use? It depends...

Measures of dispersion

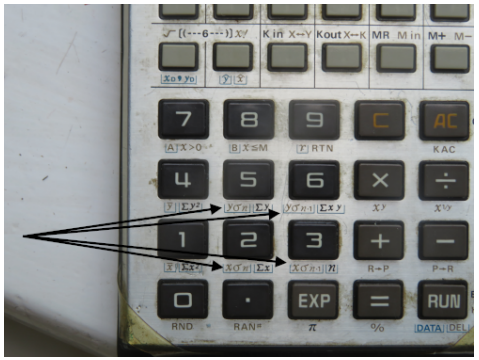
How to express the spread of your data

- The standard deviation σ: the root mean square deviation
  $V = \sigma^2 = \overline{(x - \overline{x})^2} = \overline{x^2} - \overline{x}^2$
  (V is the variance, just σ^2. Statisticians tend to use V, everyone else uses σ)
- The range. $x_{max} - x_{min}$
  Meaningful but susceptible to outliers and inevitably increases as you take more data
- The inter-quartile range: analogue of the median
- The Full Width at Half Max FWHM (or FWHH)

Bessel's correction

$\sqrt{N}$ or $\sqrt{N-1}$?

Should you use $\sigma = \sqrt{\sum \frac{(x_i - \overline{x})^2}{N}}$ or $\sqrt{\sum \frac{(x_i - \overline{x})^2}{N-1}}$?



Argument: if you take a sample then the variance of the sample tends to be smaller than the variance of the parent (because $\overline{x}$ is used rather than the true mean.)

This bias can be corrected by a factor $\frac{N}{N-1}$

Blows up for $N = 1$, which makes sense.

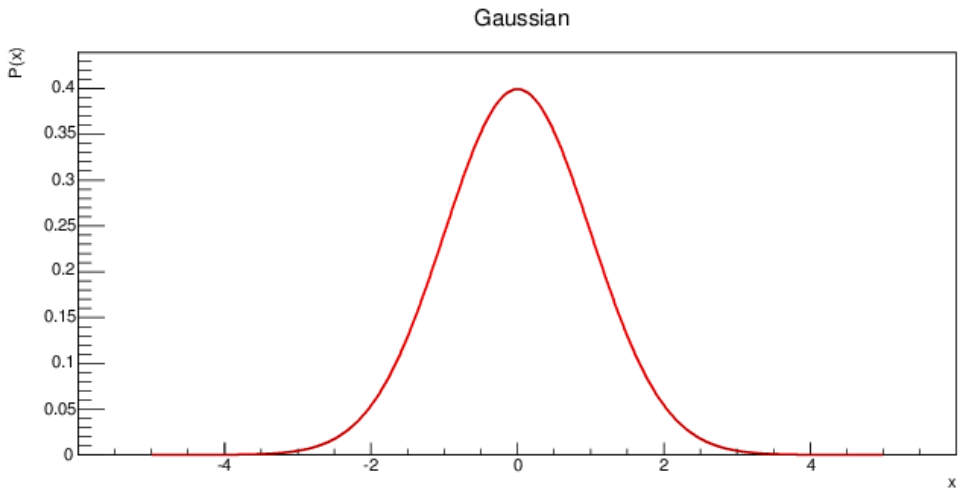
For *Descriptive Statistics* use $\sqrt{N}$. It is what it is.

For *Inferential Statistics* when you are using the standard deviation of a sample to make statements about the standard deviation of the parent, use $\sqrt{N-1}$

This removes the bias on $V = \sigma^2$. Not on σ. But σ^2 is more relevant.

The Gaussian distribution

More about σ



$$P(x;\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{\sigma^2}}$$

Mean is μ

Standard deviation is σ. Also 68% probability that x is within one sigma of μ, 95% within two sigma, etc.

Why Gaussians are Normal

The Central Limit Theorem

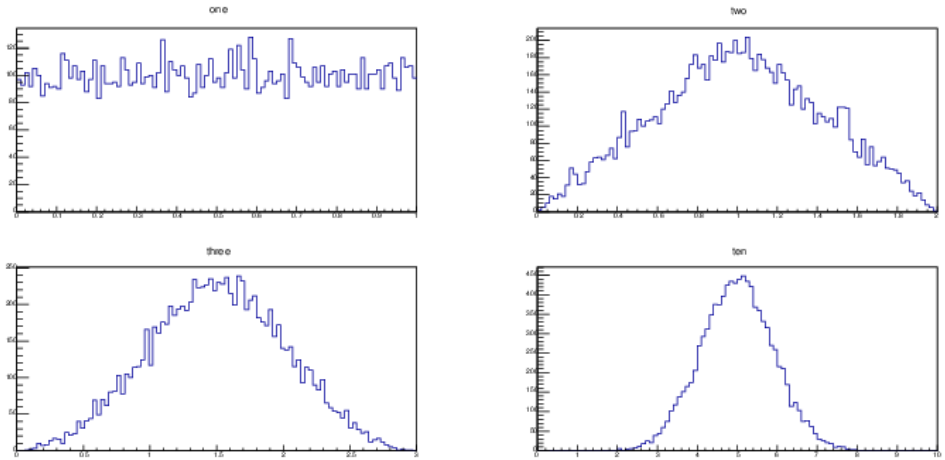
Theorem

If a random variable X is formed from the convolution of N i.i.d. variables $\{x_1 \ldots x_N\}$ then:

$\mu_X = \sum \mu_i$

$V_X = \sum V_i$

The distribution for X tends to a Gaussian as $N \to \infty$, whatever the distribution for the x_i



i.i.d. = independent and identically distributed

Higher powers

Skew

Fisher $\gamma = \frac{\overline{(x-\overline{x})^3}}{\sigma^3}$

Dimensionless number

Alternative (Pearson's skews): (mean-mode)/sigma or 3x(mean-median)/sigma

Positive skew has a tail to the right (e.g. Poisson)

Negative skew has a tail to the left (e.g. marks on easy exams)

Kurtosis

$\kappa = \frac{\overline{(x-\overline{x})^4}}{\sigma^4} - 3$

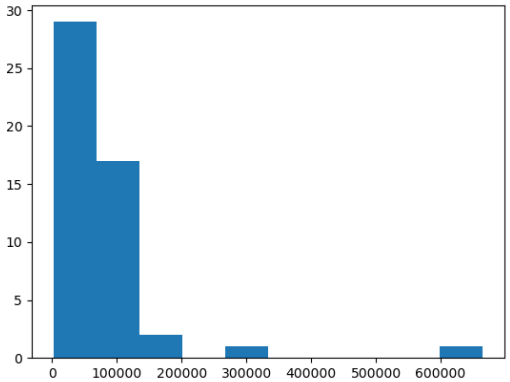
Zero for a Gaussian

Leptokurtic: $\kappa > 0$. Small but extreme tails. Typical of beams where a few particles have had very unusual experiences

Platykurtic: $\kappa < 0$. Central hump. Typical of beams after collimation.

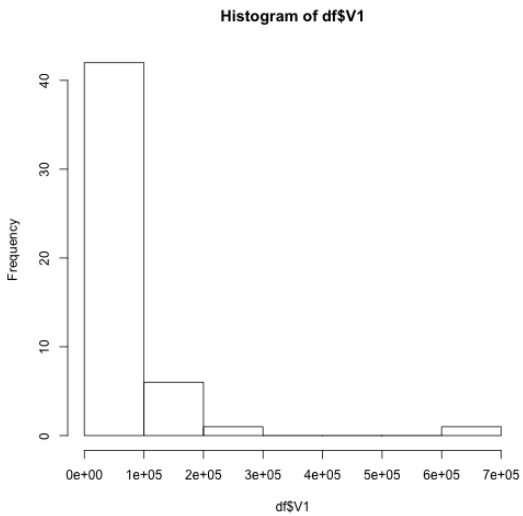
Histograms

Python



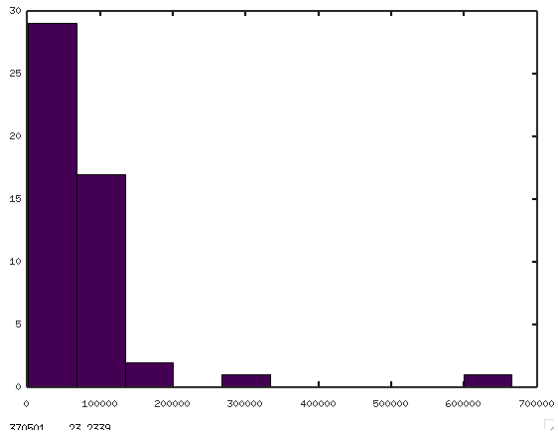

```
pl.hist(data)
pl.show()
```

R




```
hist(df$V1)
```

MATLAB




```
graphics_toolkit("gnuplot")
hist(data)
```

Gives a picture of a dataset - Need to choose binning

Histograms or bar-charts?

Histogram bars join, barcharts are separate

Histogram value $\propto$ area, barchart value $\propto$ height

Histogram horizontal axis quantitative, bar-chart qualitative

Errors on histograms

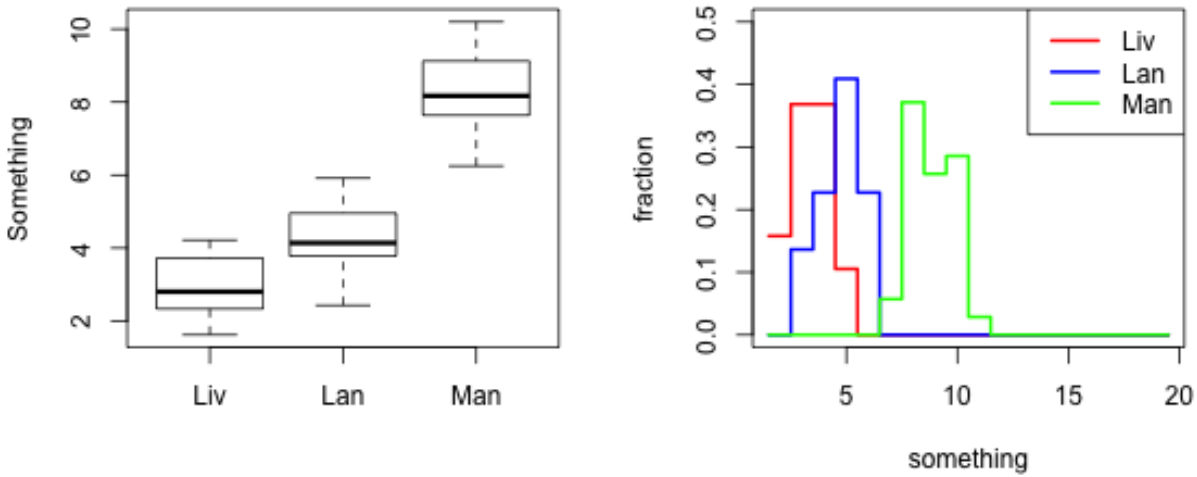
$\sqrt{N}$

Number of entries r in some bin is discrete
Given by Poisson statistics: $P(r;\mu) = e^{-\mu}\frac{\mu^r}{r!}$
r is an integer but μ, the mean, is a real
μ is also the variance. So $\sigma = \sqrt{\mu}$ and we usually get away with $\sigma = \sqrt{r}$

You may occasionally meet the binomial: $P(r;p,n) = \frac{n!}{r!(n-r)!}p^r(1-p)^{n-r}$
This has mean $\mu = np$ and variance $V = \sigma^2 = np(1-p)$
Example: suppose 95 out of 100 widgets pass their acceptance tests.
Efficiency 95% $\pm$?. $\sqrt{5}$? $\sqrt{95}$? No: $\sqrt{5 \times 95/100}$

Binomial tends to Gaussian for large n
Binomial tends to Poisson for large n, small p, fixed np
Poisson tends to Gaussian for large n
(Everything tends to Gaussian at large n, thanks to the CLT)

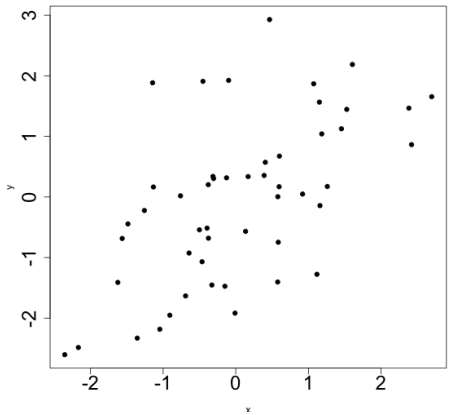
Comparing distributions



Can use several histograms or box-and-whisker plots.

`boxplot` in R. Takes list of vectors
`matplotlib.pyplot.boxplot` in Python
or `dataframe.boxplot()` using `pandas`
`boxplot` in MATLAB. Takes array.

Two (and more) dimensions



Covariance

$$Cov(xy) = \overline{(x-\overline{x})(y-\overline{y})} = \overline{xy} - \overline{x}\,\overline{y}$$

Correlation

$$\rho = \frac{Cov(x,y)}{\sigma_x \sigma_y}$$

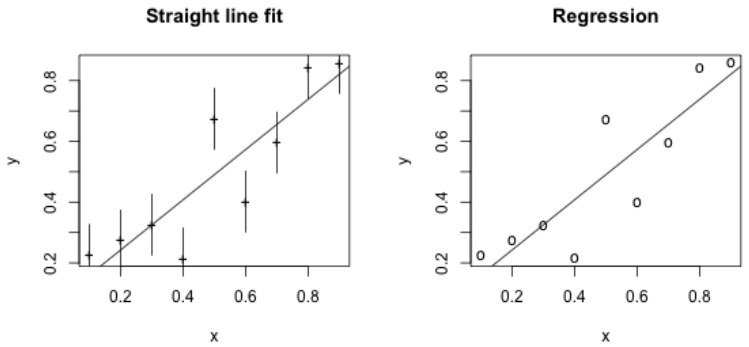
ρ is dimensionless, and lies between +1, exact correlation, and -1, exact anticorrelation. (Plot shown has $\rho \sim 0.6$)

$\rho = 0$ implies zero correlation but not necessarily independence

Generalises for many variables to $V_{ij} = \overline{x_i x_j} - \overline{x_i}\,\overline{x_j}$ and $\rho_{ij} = \frac{V_{ij}}{\sigma_i \sigma_j}$

Histograms possible in 2+ dimensions but suffer from the curse of dimensionality

Regression v. Straight line fitting



Least squares straight line fit

Slope $m = \frac{\overline{xy} - \bar{x}\,\bar{y}}{\overline{x^2} - \bar{x}^2}$ Constant $c = \bar{y} - m\bar{x}$

Error bars (weight by $\frac{N}{\sigma_i^2} / \sum \frac{1}{\sigma_j^2}$)

$\chi^2 = \sum \frac{(y_i - mx_i - c)^2}{\sigma_i^2}$ goodness of fit, should be approx N

Regression

Slope $m = \frac{\overline{xy} - \bar{x}\,\bar{y}}{\overline{x^2} - \bar{x}^2}$ Constant $c = \bar{y} - m\bar{x}$

Choice of $y = mx + c$ or $x = m'y + c'$

$R^2 = 1 - \frac{\sum (y_i - mx_i - c)^2}{\sum (y_i - \bar{y})^2}$ fraction of variation in y due to x (and $R^2 = \rho^2$)

Advice for presenting plots and numbers

- You need to help your audience to get the message
- Know how to control print format (decimal places & significant figures)
- Know how to use the non-default options in the plotting functions
- Learn about `legend` and how to place text on plots
- Label the axes. With units
- Make text large enough to be readable
- Beware of low-visibility colours

What happens next

1. Split into groups. Ideally of about 4 people in each. Everyone in a group using the same language (Python, Matlab, R, whatever) For those physically present this should be easy. Those on zoom can either (a) gather physically, if they're in one place (Lancaster? Strathclyde??) or (b) join one of the physical groups or (c) form pure zoom group(s).
2. Download `https://covid.ourworldindata.org/data/owid-covid-data.csv`
3. Look at the data and use it to say something
4. Prepare a short presentation of your results. Time about 10 minutes
5. After lunch (2:00) we re-convene. Groups make their presentations in turn, and the rest of us listen and learn and criticise and vote for the best talk.