Probability

Random variables

- You can think of each feature of a dataset as a random variable
- Each datapoint (row) is an observation, or realisation
- Why is this useful? Understanding the probability distribution:
    - can give greater understanding of the data
    - can allow us to generate data
    - can give us measures of uncertainty

An example

I'm flipping a coin. Random variable X is in {H, T}.

Here are the results of 8 coin flips:

HHHH HHTT

What is the probability the next observation is a head (H)?

Your answer depends on the probability distribution you believe governs the data generation.

Let’s try it!

$$\binom{n}{k} = \frac{n!}{(n-k)!k!} \qquad \binom{8}{2} = \frac{8!}{6!2!} = \frac{8\cdot 7}{2} = 28$$

Tip: create a new scratch notebook for this session

```
$  import numpy as np
$  np.random.choice(['H', 'T'], 8)
$  for ii in range(5):
       print(''.join(np.random.choice(['H', 'T'], 8)))
$  np.random.seed(1337)
$  for ii in range(5):
       print(''.join(np.random.choice(['H', 'T'], 8)))
$  # Probability of this specific sequence: HHHH HHTT?
$  .5**8  # <0.4%
$  # Probability of two tails?
$  .5**8 * 28  # ~11%
```

CAMBRIDGE SPARK

Bent coin

$$\binom{n}{k} = \frac{n!}{(n-k)!k!}$$

```
$  np.random.seed(None)
$  ''.join(np.random.choice(['H', 'T'], size=8, p=[0.6, 0.4]))
$  # Probability of this specific sequence: HHHH HHTT?
$  .6**6 * .4**2  # >0.7%
$  # Probability of two tails?
$  .6**6 * .4**2 * 28  # ~21%
$  # Open question: is the original coin fair i.e. p(H) = .5?
```

CAMBRIDGE SPARK

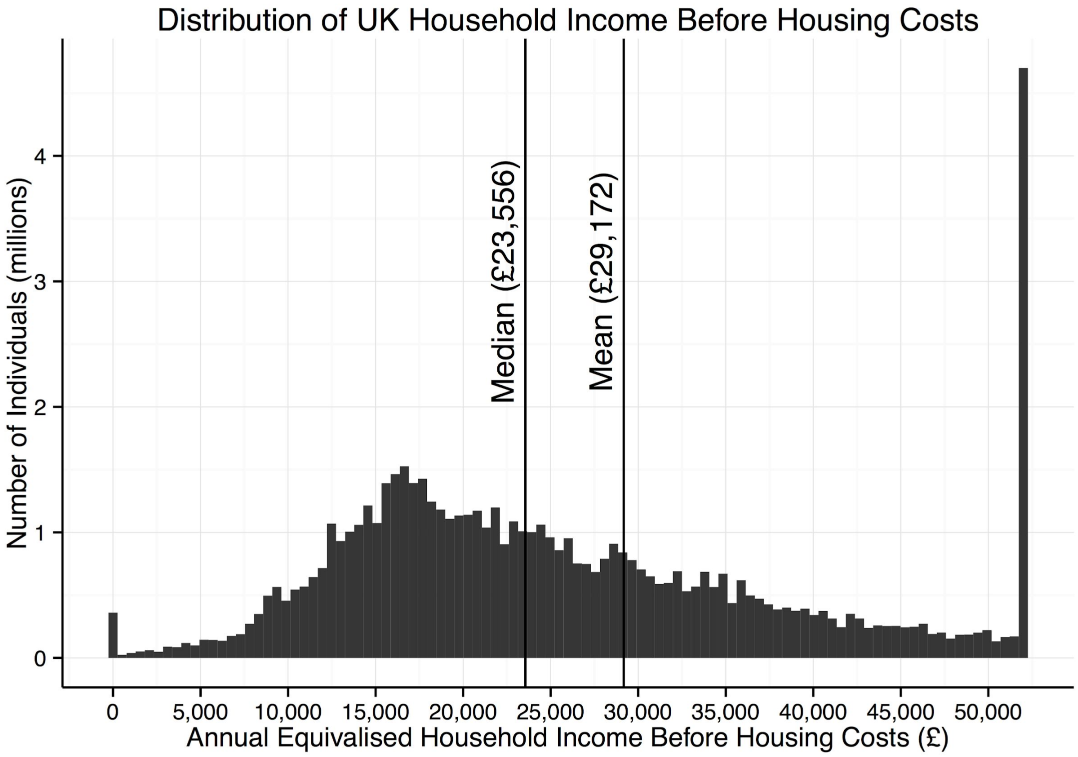
Fitting a distribution: maximum likelihood

$$p(x; \mu) = \mu^x (1 - \mu)^{(1-x)}$$

```
$  data = 'HHHHHHTT'  # ''.join(['H']*6 + ['T']*2)
$  data = [1 if flip=='H' else 0 for flip in data]
$  mu = np.sum(data) / len(data)  # np.mean(data)
$  mu
$  # fair?
$  data = np.random.choice(2, 1000, p=[.5, .5])
$  np.mean(data)
$  data = np.random.choice(2, 1000, p=[.4, .6])
$  np.mean(data)
```

CAMBRIDGE SPARK

Real data: features as random variables



CAMBRIDGE SPARK

Visualising how data is distributed

```
$   import seaborn as sns
$   url = 'https://goo.gl/XE5CrW'
$   df = pd.read_csv(url, header=None)
$   df.describe(include='all')
$   sns.distplot(df[0])
$   sns.countplot(df[1])
$   df[0].plot(kind='hist')
$   df[12].plot(kind='density')
```

CAMBRIDGE SPARK

Useful distributions

$$
\begin{aligned}
x^{-1} &= \frac{1}{x} \\
x^{\frac{1}{2}} &= \sqrt{x} \\
\exp(x) &= e^{x} \\
\boldsymbol{X} \text{ is a matrix :} & \\
|\boldsymbol{X}| &= \text{determinant}(\boldsymbol{X}) \\
\boldsymbol{X}^{-1} &= \text{inverse}(\boldsymbol{X}) \\
\text{i.e. } \boldsymbol{X}\boldsymbol{X}^{-1} &= \mathbb{I} \\
\mathbb{R} &:= \text{Real numbers e.g. 1.3, -2.77 ('float')} \\
\mathbb{N} &:= \text{Natural numbers e.g. -1, 10 (int)} \\
\in &:= \text{is a member of}
\end{aligned}
$$

$$
\begin{aligned}
\text{Distribution} &: p(\text{data}\,;\text{parameters}) = \text{pmf or pdf}\,, \quad \text{data} \in \text{domain} \\
\text{Bernoulli} &: p(x;\theta) = \theta^{x}(1-\theta)^{(1-x)}, \quad x \in \{0,1\} \\
\text{Categorical} &: p(x=c;\boldsymbol{\theta}) = \theta_c, \quad x \in \{0,\ldots,c\} \\
\text{Poisson} &: p(x;\theta) = \frac{\theta^{x} e^{-\theta}}{x!}, \quad x \in \mathbb{N} \cup \{0\} \\
\text{Gaussian} &: p(x;\mu,\sigma^2) = \left(2\pi\sigma^2\right)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right), \quad x \in \mathbb{R} \\
\text{MV Gaussian} &: p(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma}) = |2\pi\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{T}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right), \quad x \in \mathbb{R}^{D} \\
\text{Uniform} &: p(x;a,b) = \frac{1}{b-a}, \quad x \in [a,b]
\end{aligned}
$$

Distributions with scipy

```
$   from scipy import stats
$   fig, ax = plt.subplots(1, 1)
$   df[0].plot(kind='hist', density=True, ax=ax)
$   rv = stats.norm(loc=df[0].mean(), scale=df[0].std())
$   xx = np.linspace(0, 90, 100)
$   prob = rv.pdf(xx)
$   plt.plot(xx, prob)
$   rv.rvs(10)
```

CAMBRIDGE SPARK

Recommended further reading

- *Barber*, **Chapter 8.3 and 1** of **Bayesian Reasoning and Machine Learning**
    - Free online: http://web4.cs.ucl.ac.uk/staff/D.Barber/pmwiki/pmwiki.php?n=Brml.Online
- *Murphy*, **Chapter 2** of **Machine Learning: A Probabilistic Perspective**
- *Bishop*, **Chapters 1 and 2** of **Pattern Recognition and Machine Learning**
- https://docs.scipy.org/doc/scipy/reference/tutorial/stats.html
- https://docs.scipy.org/doc/numpy/reference/routines.math.html

Hands-on session

01-probability-skeleton.ipynb
30 mins