

Computer Science

IB Data Science

MRes AI4ER

Core Data Science

Lecturer

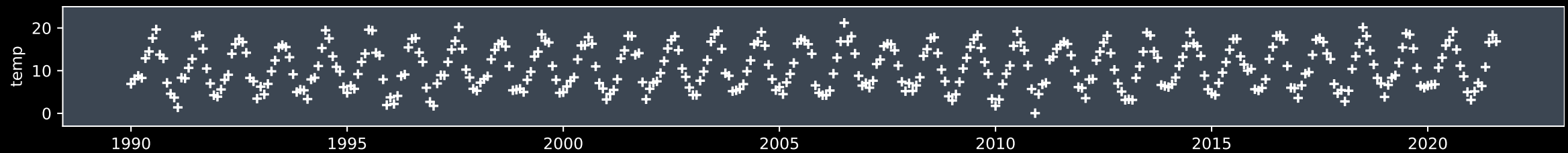
Dr Damon Wischik

Met Office climate dataset

<https://www.metoffice.gov.uk/research/climate/maps-and-data/historic-station-data>

Monthly readings from 37 weather stations around the country. Let's look at Cambridge, from 1990.

station	yyyy	mm	t	af	rain	sun	tmin	tmax	temp
Cambridge	1990	1	1990.00	0	43.8	64.7	4.0	9.8	6.90
Cambridge	1990	2	1990.08	1	71.1	102.0	4.7	11.4	8.05
Cambridge	1990	3	1990.16	3	23.2	153.2	4.7	12.9	8.80
⋮									

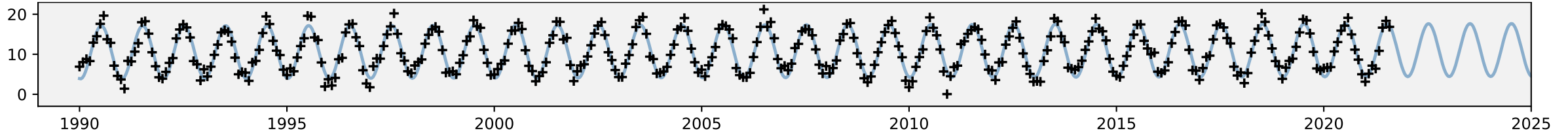


What model / formula would you suggest to fit this dataset?

```
def temp_model(t, ...):  
    return ...
```

A SCIENTIST'S DETERMINISTIC MODEL

```
def temp_model(t,  $\alpha$ ,  $\phi$ , c,  $\gamma$ ):  
    return c +  $\alpha$  * np.sin(2* $\pi$ *(t+ $\phi$ )) +  $\gamma$ *t
```



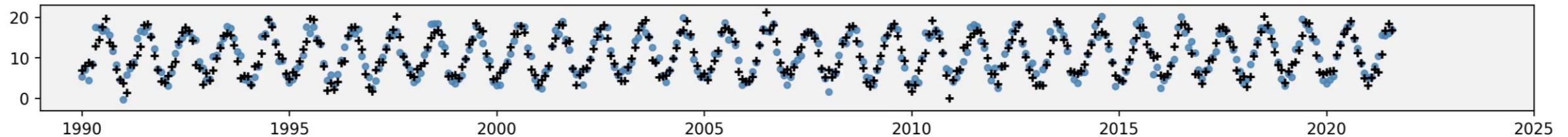
why? To describe the data in front of me!

To tell my fitting procedure how much attention to pay to outliers

To be able to say "This model can't really fit the data"

A DATA SCIENTIST'S PROBABILITY MODEL

```
def rtemp(t,  $\alpha$ ,  $\phi$ , c,  $\gamma$ ,  $\sigma$ ):  
    pred = c +  $\alpha$  * np.sin(2* $\pi$ *(t+ $\phi$ )) +  $\gamma$ *t  
    return np.random.normal(loc=pred, scale= $\sigma$ )
```



All of machine learning is based on a single idea:

1. Write out a probability model
2. Fit the model from data

This is behind

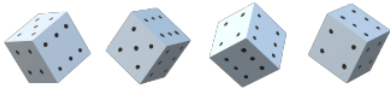
- A-level statistics formulae
- our climate model
- ChatGPT training

HANDOUT

Likelihood:

modelling and machine learning with probability

Damon Wischik, Computer Laboratory, Cambridge University



Contents

Introduction	v
I Learning with probability models	1
1 Specifying and fitting models	3
1.1 Specifying a probability model	3
1.2 Standard random variables	8
1.3 Maximum likelihood estimation	11
1.4 Numerical optimization with scipy	17
1.5 Likelihood notation	19
1.6 Generative models / unsupervised learning	22
1.7 Supervised learning	25
2 Feature spaces / linear regression	29
2.1 Fitting a linear model	30
2.2 Feature design	33
2.2.1 One-hot coding	33
2.2.2 Non-linear response	34
2.2.3 Comparing groups	34
2.2.4 Periodic patterns	35
2.2.5 Secular trend	36
2.3 Diagnosing a linear model	38
2.4 Linear regression and least squares	40
2.5 The geometry of linear models	41
2.6 Interpreting parameters	44
2.7 Gauss's invention of least squares	49
3 Neural networks	51
3.1 Prediction accuracy	53
3.2 Probabilistic deep learning	56

- ABRIDGED NOTES
(contain all examinable material)
- EXTENDED NOTES
(contain all examinable material + extras)

- For more printouts, ask student admin

- The handout has more wordy explanations and more examples than lectures

- Use the handout like a textbook and take your own notes during lectures

§x

- Slides for each lecture are on the website and most slides say which section they're for

- What's examinable?
Everything in the lecture schedule, except for sections marked *

Department of Computer Science x +

cl.cam.ac.uk/teaching/2324/DataSci/materials.html

UNIVERSITY OF CAMBRIDGE

Course pages 2023–24

Data Science

Syllabus Course materials Recordings Information for supervisors

Lecture notes

- [Abridged notes](#) as printed — examinable material only
- [Extended notes](#) with extra material on non-examinable material such as neural networks

If you spot a mistake in the printed notes, let me know.

Announcements and Q&A — [Moodle](#)

Lecture schedule

This is the planned lecture schedule. It will be updated as and when actual lectures deviate from schedule. Material marked * is non-examinable. **Slides** are uploaded the night before a lecture, and re-uploaded after the lecture with annotations made during the lecture.

Example sheet 0 (prerequisites) and solutions

Lecture 1 [1. Learning with probability models](#) (4:08) code — fitting, climate [slides]

- [1.1 Specifying probability models](#) (15:20) code — fitting
- [1.2 Standard random variables](#) (3:21)

Lecture 2 [1.3 Maximum likelihood estimation](#) (17:35) code — fitting

- [1.4 Numerical optimization](#) (8:01)

Lecture 3 [1.5 Likelihood notation](#) (10:00)

- [1.6 Generative models](#) (8:14) code — fitting
- [1.7 Supervised learning](#) (14:18) code — fitting

Lecture 4 **Mock exam question 1** and [walkthrough](#) (23:36)

Lecture 5 [2.1 Linear modelling](#) (13:27) code — lm

- [2.2 Feature design](#) (19:39)
- [2.3 Diagnosing a linear model](#) (5:29)

The screenshot shows a web browser window with a YouTube playlist. The address bar shows the URL: `youtube.com/playlist?list=PLknxd7zG11MLG2w-I0...`. The YouTube logo and search bar are visible at the top. The playlist title is "IB Data Science (2021-22)" with a pencil icon for editing. Below the title, it says "5 videos • No views • Updated today" and "Public" with a dropdown arrow. There are icons for share, link, and more options. The video list includes:

- A SCIENTIST'S MODEL (with a graph showing a sine wave and the text "increasing at 3.15 °C/century")
- A DATA SCIENTIST'S MODEL (with a similar graph)
- 1. Learning with probability models (4:08)
- 1.1 Specifying probability models (15:20)
- 1.2 Standard random variables (3:21)
- 1.3 Maximum likelihood estimation (17:35)

Each video entry has a thumbnail, a title, a duration, and the channel name "Foundations of Data Science". A "PLAY ALL" button is located below the first two video thumbnails.

- Pre-recorded videos from 2021-22 are on YouTube
- All examinable material is in these videos
- For recordings of lectures ...

Consent to recordings of live lectures

<https://www.educationalpolicy.admin.cam.ac.uk/policy-index/recording>

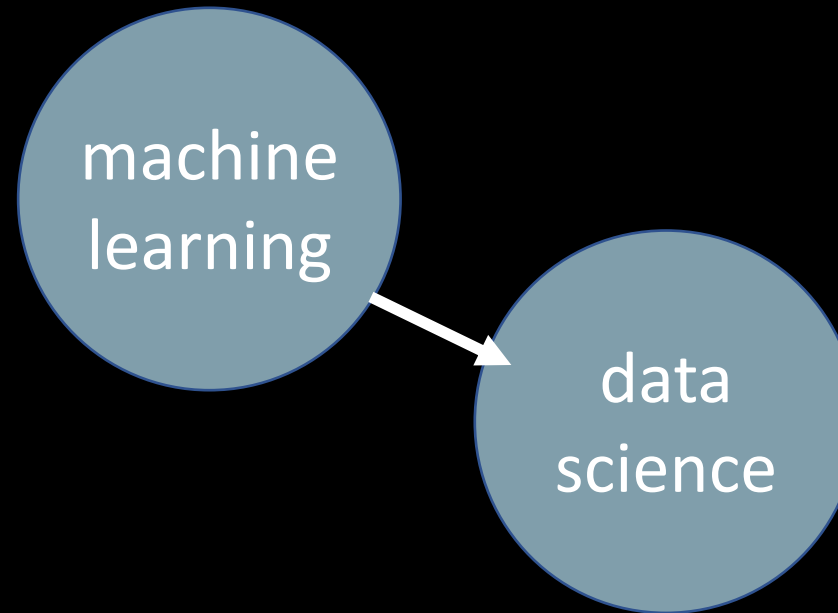
For any teaching session where your contribution is mandatory or expected, we must seek your consent to be recorded.

You are not obliged to give this consent, and you have the right to withdraw your consent after it has been given.

Do you give your consent to recordings?

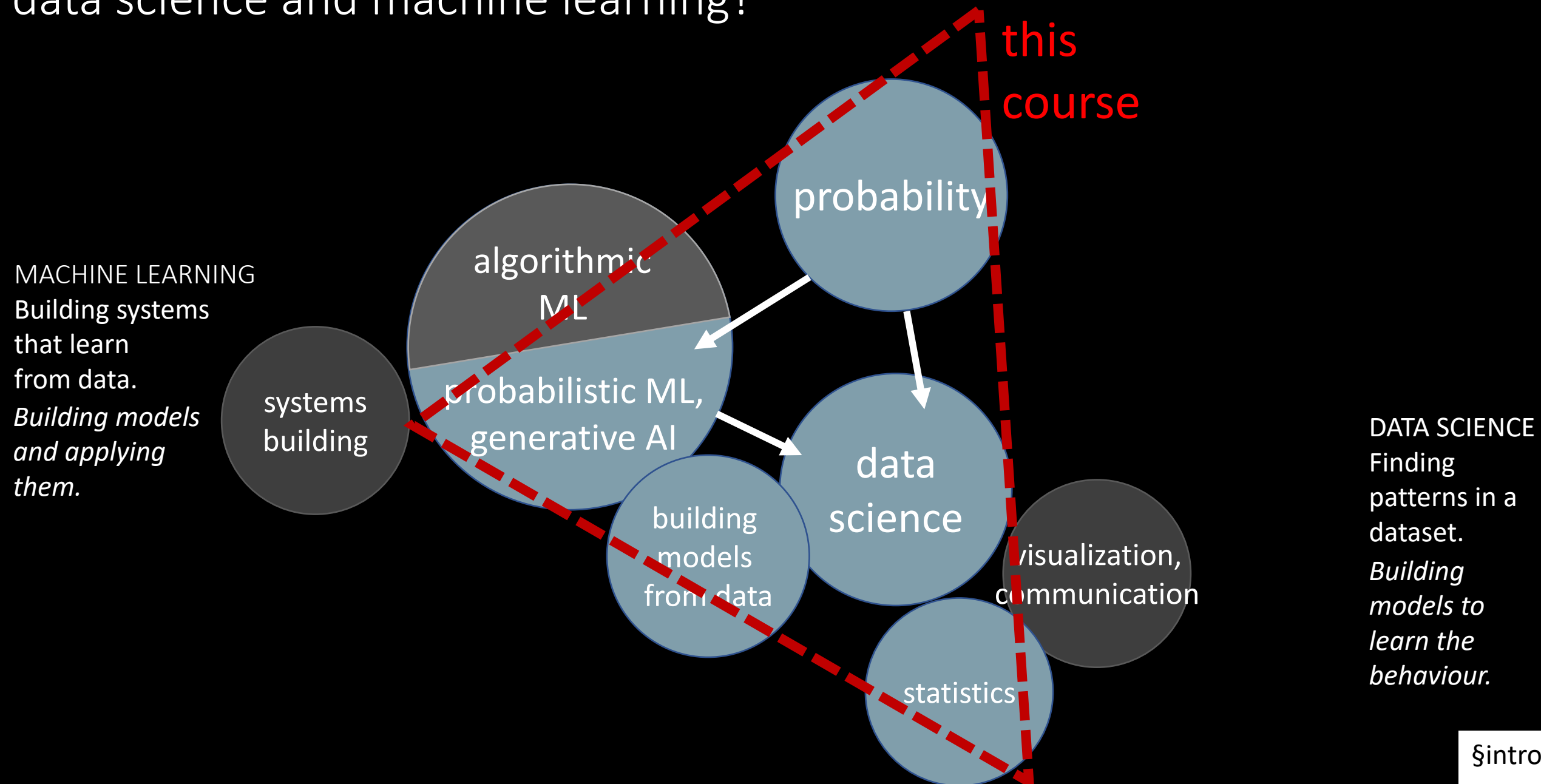
What is data science? What's the difference between data science and machine learning?

MACHINE LEARNING
Building systems
that learn
from data.



DATA SCIENCE
Finding
patterns in a
dataset.

What is data science? What's the difference between data science and machine learning?



If you don't get this elementary, but mildly unnatural, mathematics of elementary probability into your repertoire, then you go through a long life like a one-legged man in an ass kicking contest.

Charles Munger, business partner of Warren Buffett

Example sheet 0

Prerequisites

IB Data Science—DJW—2023/2024

This course assumes that you know how to handle basic probability problems and that you know about random variables, as taught in IA *Introduction to Probability*. It also assumes that you know how to find the maximum or minimum of a function, using calculus, as taught in IA *Maths for NST*. The code snippets in the course are in Python and numpy, and you should be familiar with numpy's way of writing vectorized computations.

This example sheet reviews the material that you need to know. Please look through, and make sure you remember how to answer these questions! Solutions are provided on the course website. *For supervisors: this example sheet is not intended for supervision.*

Rules of probability (IA Probability lecture 1)

Understand what is meant by *sample space*, written Ω , and know that $\mathbb{P}(\Omega) = 1$. Be able to reason about probabilities of events with Venn diagrams. Know the core definitions and laws ...

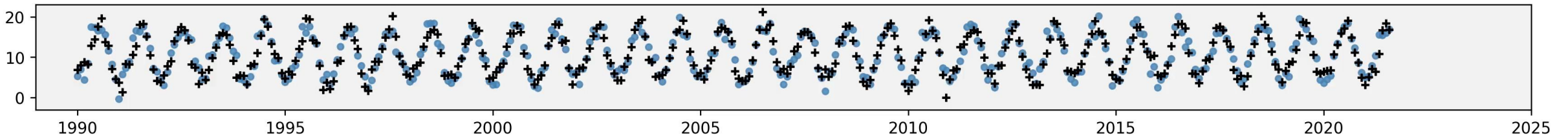
Conditional probability, or equivalently the chain rule:

$$\mathbb{P}(A | B) = \frac{\mathbb{P}(A, B)}{\mathbb{P}(B)} \quad \text{if } \mathbb{P}(B) > 0$$

$$\mathbb{P}(B, A) = \mathbb{P}(B) \mathbb{P}(A | B) \quad (\text{chain rule})$$

- Example sheet 0 is to remind you about IA Probability, Maths for NST, and Scientific Computing
- It's not for supervision; solutions are provided

How to specify a probability model

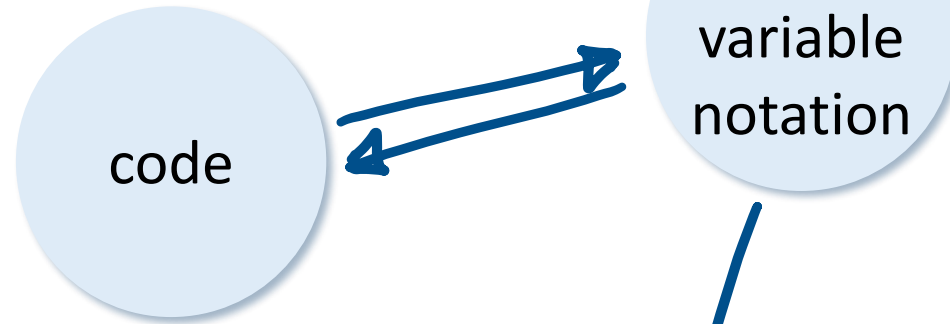


```
def rtemp(t,  $\alpha=10$ ,  $\phi=-0.25$ ,  $c=11$ ,  $\gamma=0.035$ ,  $\sigma=2$ ):  
    pred =  $c + \alpha * \text{np.sin}(2*\pi*(t+\phi)) + \gamma*t$   
    return np.random.normal(loc=pred, scale= $\sigma$ )
```

A probability model is a piece of code
where the output is random.

Three views of a probability model

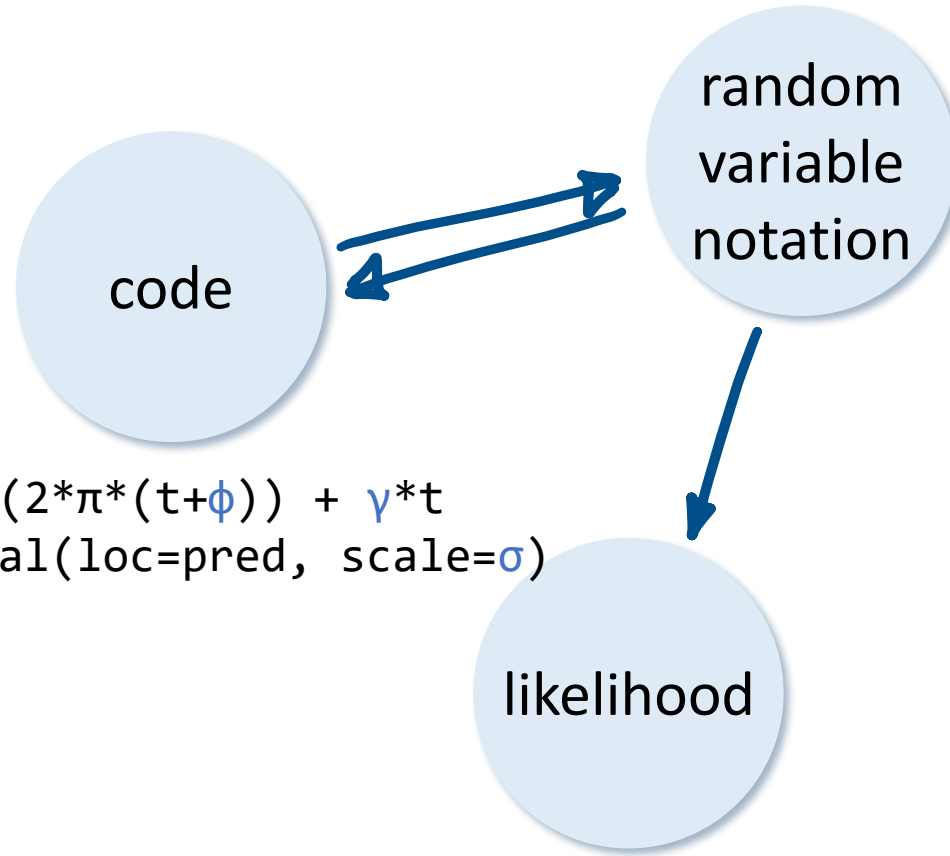
for intuition and
simulation



for learning
from data

Three views of a probability model

$$\text{Temp}_i \sim \alpha \sin(2\pi(t_i + \phi)) + c + \gamma t_i + \text{Normal}(0, \sigma^2), \\ i \in \{1, \dots, n\}$$



```
def rtemp(t,  $\alpha$ ,  $\phi$ ,  $c$ ,  $\gamma$ ,  $\sigma$ ):  
    pred =  $c$  +  $\alpha$  * np.sin(2* $\pi$ *(t+ $\phi$ )) +  $\gamma$ *t  
    return np.random.normal(loc=pred, scale= $\sigma$ )
```

```
def ry():  
    x = random.random()  
    y = x ** 2  
    return y
```

$$X \sim U[0,1]$$
$$Y = X^2$$

Generate X from the
Uniform distribution.

```
def ri(a,b):  
    x = random.random()  
    i = math.floor(a*x+b)  
    return i
```

$$X \sim U[0,1]$$
$$I = \lfloor aX + b \rfloor$$

Upper case: random variables
Lower case: parameters, constants.

```
x = random.random()  
y = x ** 2
```

$$X \sim U[0,1]$$
$$Y = X^2$$


```
def rz():
    x1 = random.random()
    x2 = random.random()
    return x1 * math.log(x2)
```

$$X_1, X_2 \sim U[0,1]$$
$$Z = X_1 \log X_2$$

' X_1 and X_2
are generated independently"
— knowing the value of one tells us
nothing about the value of the other.

In random variable notation,
assume independence

```
def rmyrandpair():
    x1 = random.random()
    x2 = random.random()
    y, z = (x1+x2, x1*x2)
    return (y, z)
```

$(Y, Z) \sim \text{Myrandpair}$ unless specified otherwise
(like this)

```
 $\lambda = 3$   
 $x_1 = \text{random.uniform}(0, \lambda)$   
 $x_2 = \text{random.uniform}(0, \lambda)$ 
```

$$X_1, X_2 \sim U[0, \lambda]$$

λ is lower-case, so it refers to a fixed value.
When we say " X_1 and X_2 are independent",
we mean " X_1 and X_2 are independent
given the parameters."

```
x = random.random()
y = 1 - x
```

$$X \sim U[0,1]$$
$$Y = 1 - X$$

\sim : "has the same distribution"
"has the same histogram"

$$X \sim U[0,1]$$
$$Y \sim U[0,1]$$
$$X \sim Y$$
$$X \sim 1 - Y$$

$=$: "always has the same value every time I run the code"

$$Y = 1 - X$$
$$X + Y = 1$$


```
x = random.random()  
y = np.random.normal(  
    loc=x, scale=0.1)
```

$X \sim U[0,1]$
 $Y \sim N(X, 0.1^2)$

"first generate X
then use it to generate Y "

```
def rtemp(t, α=10, φ=-0.25, c=11, γ=0.035, σ=2):
    pred = α*np.sin(2*π*(t+φ)) + c + γ*t
    return np.random.normal(loc=pred, scale=σ)
```

```
df = pandas.read_csv(...) # data frame, n=380 rows
Temp = rtemp(df.t)        df.t is a vector of length 380
```

$$\text{Temp}_i \sim \alpha \sin(2\pi(t_i + \varphi)) + c + \gamma t_i + \text{Normal}(0, \sigma^2), \quad i \in \{1, \dots, n\}$$

This expresses 380 separate equations.
 Each of these eqns has an independent $N(0, \sigma^2)$.
 (That's what the `np.random.normal` call generates)

Or, equivalently,

$$\text{Temp}_i = \alpha \sin(2\pi(t_i + \varphi)) + c + \gamma t_i + \varepsilon_i, \quad \varepsilon_i \sim \text{Normal}(0, \sigma^2), \quad i \in \{1, \dots, n\}$$

All of machine learning is based on a single idea:

1. Write out a probability model
2. Fit the model from data

This is behind

- A-level statistics formulae
- our climate model
- ChatGPT training

§1

A core skill is being able to design probability models. This course is for you to learn this skill, through examples.

