Module introduction

Computational Statistics
— random simulation and statistical optimisation

What is computational statistics?

Definition

Computational statistics develops and studies algorithms for carrying out statistical calculations.

Example: How much stock should a shop order?

A shop orders sandwiches before it knows tomorrow's demand.

- Too many sandwiches: unsold food costs money.
- Too few sandwiches: customers leave empty-handed.

Suppose demand averages **40 sandwiches a day**.

Discuss

Would you order 40? What else would you want to know?

What does a wrong decision cost?

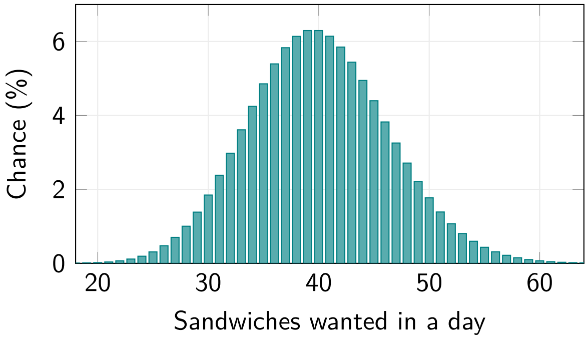
Teaching assumptions

- Each leftover sandwich costs £ 2.
- Each sandwich of unmet demand carries a £ 8 penalty.

Order	Actual demand	Cost
45	40	5 leftovers: £ 10
45	50	5 unmet orders: £ 40

Shortages are more expensive than leftovers in this example.

A model describes possible demand



A distribution describes which outcomes are more or less likely.

Days near 40 are common.

Very quiet or busy days are less common.

Hypothetical demand model. These are model probabilities, not observed sales.

Five possible days

Suppose we try both order quantities on the **same five days**.

Demand	Cost if we order 40	Cost if we order 45
32	£ 16	£ 26
38	£ 4	£ 14
40	£ 0	£ 10
45	£ 40	£ 0
50	£ 80	£ 40

An average lets us compare the decisions

	Order 40	Order 45
Total cost over the five days	£ 140	£ 90
Average cost per day	£ 28	£ 18

$$\text{Average cost} = \frac{\text{Total cost across the days}}{\text{Number of days}}$$

Discuss

Would five selected days be enough to trust this decision?

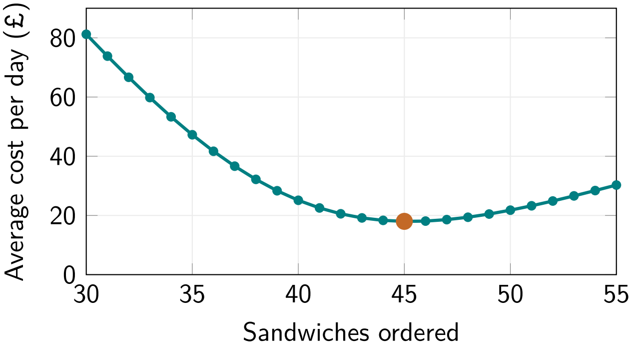
A computer can generate many possible days

1. Generate a day's demand using the model.
2. Calculate the cost of each order quantity.
3. Repeat for many simulated days.
4. Compare the average costs.

Random does not mean arbitrary.

More likely demands should appear more often.

Comparing many order quantities



100,000 simulated days
Order 40: about £ 25.12 a day.
Order 45: about £ 18.00 a day.
45 has the lowest simulated average among these choices.

Results from executing the simulation under our hypothetical model.

What does the answer mean?

Under these demand and cost assumptions, ordering 45 looks useful.

- It lowers average cost compared with ordering 40.
- It can still leave leftovers or miss some customers.
- A different demand pattern or shortage penalty may change the choice.

Key idea

A good average decision can still have a bad day.

We have just used Monte Carlo simulation

Monte Carlo simulation uses random draws from a model to estimate a quantity of interest.

For the shop, we generated demand and averaged costs.

Question	Calculation with simulated days
What is the average cost?	Add the costs and divide by the number of days.
How often do we run out?	Count the days with a shortage and divide by the number of days.

Another simulation gives another answer

Suppose we keep the model and the order quantity fixed.

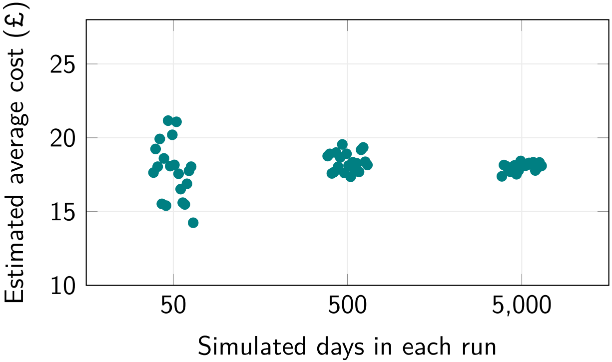
- One run may contain more unusually busy days.
- Another may contain more quiet days.
- Their average costs will usually differ.

Discuss

Which would you expect to be more stable: an average of 50 days or an average of 5,000 days?

Simulation error: variation caused by using a finite random sample.

Longer runs usually give more stable averages



Each dot is a separate simulation run.
20 runs at each sample size.
The larger samples produce a tighter cluster of answers.

Executed simulations. Order fixed at 45. The demand model stays the same.

More simulated days cannot fix a poor model

Imagine the shop is next to a football stadium.

- The model describes ordinary weekdays.
- Tomorrow is a match day.
- Running the weekday model a million times still describes weekdays.

Discuss

What should change before we trust the recommendation?

More computation improves precision within a model.

Relevant evidence helps us assess the model itself.

Hospital capacity: planning for busy days

Practical question

How often might demand exceed the available beds?

- Patient arrivals and lengths of stay vary.
- A model can simulate many possible weeks.
- We can compare capacity plans by how often beds run out.

Discuss

Why might an average bed occupancy below capacity still be a concern?

Medical imaging: making sense of indirect measurements

A scanner records measurements that contain noise and incomplete information.

1. Propose an image.
2. Check how well it explains the measurements.
3. Improve the image using a clearly stated criterion.

Optimisation means searching for the best value of a chosen criterion.

Key idea

We also want to understand which features the measurements support confidently.

Evaluating an AI model

A fitted classifier gets **90 out of 100** test cases correct.

Discuss

Does that guarantee 90% accuracy on the next 100 cases?

- Another set of cases may give a different result.
- Resampling the test cases can help explore this variability.
- The test cases must represent the intended users and conditions.

Hypothetical example. The model is fixed and the test cases are separate from training.

What is computational statistics?

Computational statistics studies how to carry out statistical calculations reliably and efficiently.

A recurring task	Our examples
Simulate outcomes	Possible days of demand or patient arrivals.
Estimate a quantity	Average cost or chance of running out.
Optimise a choice	An order quantity or an image reconstruction.

The course questions

Chapters	What we will learn to ask
1–2	How can simulation estimate an average, and how accurate is it?
3–5	How can we make better use of a limited computing budget?
6–7	How can we generate samples from more complicated models?
8	How can we fit a model by improving a chosen criterion?

Each method will come with an example, an algorithm, and an explanation of when it works.

A manageable starting point

Useful ideas to refresh

- Averages and percentages.
- Reading a graph and describing variation.
- Basic probability: possible outcomes and their chances.

Later, we will use calculus, linear algebra, and programming.

For each method, ask: **what does it calculate, and what assumptions does it need?**

What is computational statistics?

Working definition

Computational statistics develops and studies algorithms for carrying out statistical calculations.

- A model may require an integral that we cannot evaluate analytically.
- An estimate may require solving an optimisation problem.
- Simulation may require samples from a complicated distribution.
- Even an available formula may need careful numerical implementation.

Note

Difficult computations can arise with small datasets as well as large ones.

Two recurring mathematical tasks

Evaluate an expectation

$$I = \mathbb{E}_p[f(X)].$$

Average a quantity over a probability distribution. Examples: a probability, a mean, a predictive loss.

Optimise an objective

$$\widehat{\theta} \in \arg\min_{\theta} F(\theta).$$

Find parameters that give the best value of a stated criterion. Examples: least squares, negative log-likelihood.

Both tasks require us to understand what an algorithm computes and how accurately it computes it.

Expectations turn models into numerical questions

If X has density p on $\mathbb{R}^d$ and $\mathbb{E}_p[|f(X)|] < \infty$, then

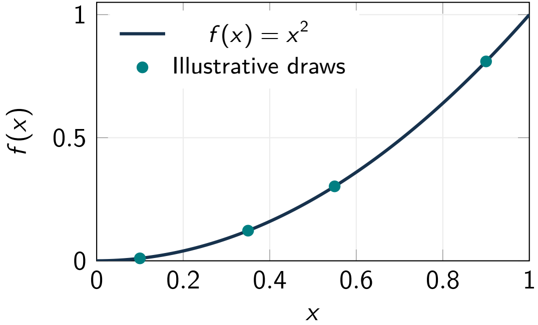
$$\boxed{I = \mathbb{E}_p[f(X)] = \int_{\mathbb{R}^d} f(x)p(x)\,\mathrm{d}x.}$$

Choice of $f(x)$	Target	Interpretation
$x \quad$ (scalar X)	$\mathbb{E}[X]$	Population mean
$\mathbf{1}_A(x)$	$\mathbb{P}(X \in A)$	Event probability
$(x-\mu)^2$	$\mathrm{Var}(X)$	Spread about $\mu = \mathbb{E}[X]$

Note

The model supplies p. The scientific question determines f.

A first Monte Carlo picture



$$I = \int_0^1 x^2 \, \mathrm{d}x = \frac{1}{3}.$$

For $U_i \overset{\text{iid}}{\sim} U(0,1)$,

$$\widehat{I}_N = \frac{1}{N} \sum_{i=1}^{N} U_i^2.$$

Four illustrative values give

$$\widehat{I}_4 = 0.31125.$$

Another sample gives another answer.

Probabilities can also be estimated by averaging

Let A be an event of interest, such as a simulated system failure.

$$\mathbf{1}_A(x) = \begin{cases} 1, & x \in A, \\ 0, & x \notin A. \end{cases} \qquad \mathbb{P}(X \in A) = \mathbb{E}[\mathbf{1}_A(X)].$$

For independent draws from the model,

$$\widehat{P}_N = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}_A(X_i) = \frac{\text{number of simulated occurrences}}{N}.$$

Pause and discuss

If no failures occur in a finite simulation, what can we conclude?

Sampling provides the inputs to Monte Carlo

Sampling question

Given a distribution p, how can we generate draws with that distribution?

$$X_1, \ldots, X_N \sim p$$

Examples include uniform and normal distributions.

Monte Carlo question

How can we use simulated draws to estimate a numerical quantity?

$$\widehat{I}_N = \frac{1}{N} \sum_i f(X_i)$$

We must assess the resulting simulation error.

Note

The sampling method determines which assumptions we can use. Independence must be established.

Optimisation begins with an objective

Suppose observations satisfy

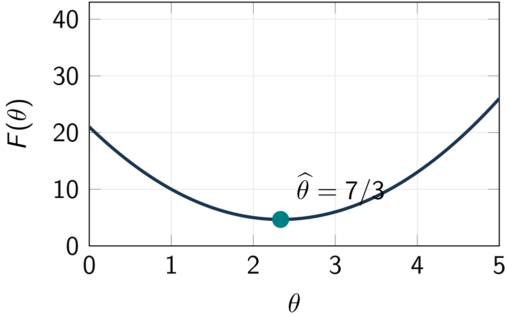
$$y_i = m(x_i; \theta) + \varepsilon_i, \qquad i = 1, \ldots, n.$$

Least squares chooses parameters to minimise

$$\boxed{F(\theta) = \sum_{i=1}^{n} \big[y_i - m(x_i; \theta)\big]^2.}$$

- The data (x_i, y_i) stay fixed during this optimisation.
- The algorithm changes θ to reduce the objective.
- Nonlinear models often require iterative numerical methods.

A least-squares problem we can see



Fit a constant to $y = (1, 2, 4)$:

$$F(\theta) = (1 - \theta)^2 + (2 - \theta)^2 + (4 - \theta)^2.$$

$$F'(\theta) = 6\theta - 14.$$

The minimum occurs at

$$\widehat{\theta} = \overline{y} = \frac{7}{3}.$$

More complex objectives may lack a simple solution.

Likelihood and EM: topics ahead

For independent observations with density $p(y;\theta)$,

$$\ell(\theta) = \sum_{i=1}^{n} \log p(y_i;\theta), \qquad \widehat{\theta} \in \arg\max_{\theta} \ell(\theta).$$

Maximum likelihood

Choose parameters that maximise the likelihood of the observed data.

Expectation–maximisation (EM)

An iterative approach often used when a model includes latent variables or incomplete data.

We will need to examine convergence and the possibility of local optima.

The Monte Carlo part of the course

1. **Probability and estimation review** Expectations, variance, estimators and mean squared error.
2. **Basic Monte Carlo** Sample averages and their computational accuracy.
3. **Variance reduction** Importance sampling and control variates.
4. **Sampling algorithms** How to generate draws from a specified distribution.

Guide to the supplied notes

Sections 1–2: foundations Sections 3–5: variance reduction
Sections 6–7: sampling, including Markov chain Monte Carlo.

The optimisation part of the course

- **Gradient descent:** use first derivatives to choose an update.
- **Newton's method:** also use curvature information.
- **Nonlinear least squares:** fit models through residual minimisation.
- **Maximum likelihood and EM**

Note

An algorithm needs assumptions, a stopping rule and a way to assess its output.

Reading: supplied notes, Section 8, pp. 39–43, for regression and optimisation.

Background knowledge to refresh

Probability and statistics

- Densities and probabilities.
- Expectations and variances.
- Independence and covariance.
- Sample means and parameter estimation.

Mathematics

- Basic integration.
- Derivatives and stationary points.
- Vectors and matrices.
- Gradients for functions of several variables.

The diagnostic activity will help distinguish topics that need revision from topics that are new.

A useful weekly study routine

1. Read the relevant definitions before the lecture.
2. During the lecture, record why each algorithm works and its assumptions.
3. Attempt exercises before consulting a solution.
4. Use guided study to discuss incomplete attempts and specific questions.
5. After feedback, repeat a similar calculation without looking at the notes.

Pause and discuss

Can you explain what an algorithm estimates before writing its formula?

Course materials and access

Canvas is the course information hub in the source lecture

Announcements, lecture notes, exercise problems, assignments and available recordings.

- Check that you can access the course page.
- Read announcements and check for revised notes regularly.
- Raise access or accessibility difficulties with the teaching team.

Questions and support

- Use the course contact page for the lecturer's email and office hours.
- Ask about unclear definitions or steps in a derivation.
- Bring attempted solutions to guided study.
- Discuss substantial gaps in background knowledge early.

Teaching format

Module size	20 credits
Teaching period	Approximately ten teaching weeks
Weekly format	Four lectures and one guided-study session
Guided study	Attempt problems, then discuss approaches

Historical arrangements

These details describe the original 2024 course. Confirm the teaching pattern for the intended offering before using this slide with students.

Assessment

Continuous assessment: 20%

Four homework problem sets.

Final examination: 80%

A final written examination.

The lecturer specified **pen-and-paper questions**, with **no programming questions** in the assessed homework or examination.

Historical arrangements

These are original course statements. Confirm the assessment rules for any new offering.

Original problem-set schedule

Problem set	Release week	Due week
1	3	4
2	5	6
3	9	10
4	10	11

Working time

Approximately one week for each problem set.

Exact dates and submission times require confirmation from the assignment notices.

The first mathematical lectures

1. Express a target as an expectation.
2. Distinguish a parameter, statistic, estimator and estimate.
3. Analyse bias, variance and mean squared error.
4. Introduce the basic Monte Carlo estimator and its accuracy.

Suggested reading for this adapted deck

Supplied notes: Section 1, pp. 1–3, and Section 2, pp. 3–4.

The following foundations lectures develop these ideas step by step.