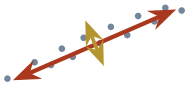
AI and Machine Learning
Unsupervised Learning and Responsible AI

Clustering, principal components, and model governance

groups nobody labelled



a few shared directions

Fatih Kansoy

Day 4 of 4

Worcester College, University of Oxford

Lecture overview

1. Supervised and unsupervised learning
2. K-means clustering of London listings
3. Principal component analysis of the yield curve
4. Risks in AI-assisted decisions
5. Course review and conclusions

Part I · Supervised and unsupervised learning

Days 1 to 3 all had an answer column

Every file carried a column of correct answers, and the job was to predict it. That is **supervised learning**.

Day	The answer we predicted	How we judged it
1	will this bank client subscribe?	accuracy, on held-out clients
2	what price for a London listing? and will a client respond?	MAE and log loss, held out
3	the same two answers, now with trees, forests and boosting	R^2 and log loss, tested once

The held-out answer was always the judge. Today that column is gone.

Most data arrives with no answer column

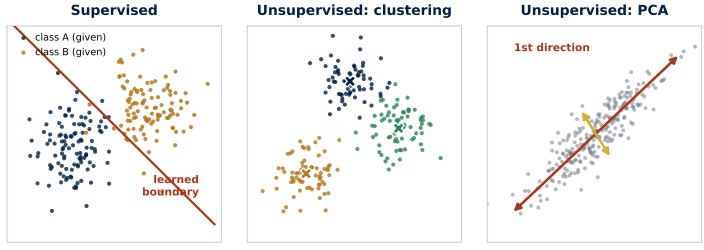
Labels are not found in data. Someone has to write them down, and that costs money.

- ImageNet's 14 million images were labelled by hand, by close to 50,000 people in 167 countries hired through Amazon Mechanical Turk. A machine cannot invent the labels it learns from.
- The same constraint shapes today's largest models. A language model is pretrained on ordinary text by predicting the next word, so the text is its own answer key. The human-labelled stage comes afterwards, and is far smaller.

Data an organisation already holds	The column nobody recorded
millions of card transactions	no "customer segment"
every support ticket ever filed	no "root cause"
years of machine sensor readings	no "about to fail" flag

Waiting for labels wastes the data already in hand.

Without an answer, look for groups or directions



	Supervised, Days 1–3	**Clustering**, Part II	**Components**, Part III
Returns	a predicted answer	one group label per row	a few scores per row
Run here on	bank clients; London listings, price given	the same London listings, price removed	36 years of Treasury yields
Judged by	held-out answers	nothing to check against, so judgement has to change	

A pattern is a proposal until it is tested

An unsupervised method cannot return "nothing found". Ask for four groups and you get four groups, even in data that has none.

Nothing in the output tells you which case you are in. So a pattern is a proposal, not a finding, until it is tested three ways:

Check	How to test it, and what it tells us
Held-out structure *(generalises?)*	Fit on one sample; ask whether the pattern also describes unseen cases. Similar train and test fit suggests it travels.
Stability *(repeats?)*	Refit after new starts, resamples, or time periods. Similar results mean reproducibility, not usefulness.
Downstream audit *(matters?)*	Compare with an important variable excluded from fitting. Association means external relevance, not causality.

find the structure ⟶ test that it travels ⟶ own the judgement.

Learning objectives

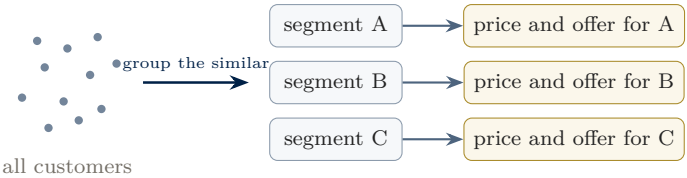
By the end of today you can:

- say what clustering and principal components are given, what each returns, and why neither can be scored on a held-out answer;
- read a set of groups, or a set of directions, and describe in plain words what each one represents;
- run the three checks on a pattern, and state what a passed check does and does not licence anyone to do;
- name a real failure of an AI system used on people, and the safeguard that would have caught it.

Part II · K-means clustering of London listings

Customer segmentation with clustering

Segmentation divides a customer base into groups, so that customers within a group are similar and the groups differ. Decisions are then made per group, not per average:



- The payoff is measurable: McKinsey (2021) finds personalisation most often lifts revenue by 10 to 15%.
- In practice the groups are built by **clustering** algorithms; the most widely used is **k-means**, the subject of this part.

Three real cases first; then we build a segmentation of the London short-let market.

K-means in UK neighbourhood classification

The UK's official geodemographic classification is built with k-means.

Problem	Government and researchers needed one consistent description of the social character of about 230,000 small neighbourhoods; the census offers dozens of variables per area, too many to read one by one.
Approach	The Office for National Statistics, with University College London, ran **k-means** on 60 census variables and sorted every neighbourhood into 8 supergroups, subdivided into 26 groups and 76 subgroups, each with a plain-language pen portrait.
Result	A free, official classification used in public health research, local-government service planning, and retail location analysis.

The 8 supergroups: Rural Residents · Cosmopolitans · Ethnicity Central · Multicultural Metropolitans · Urbanites · Suburbanites · Constrained City Dwellers · Hard-Pressed Living.

ONS, 2011 Output Area Classification (methodology and pen portraits); updated for 2021.

Clustering in Netflix recommendations

The best-known behavioural segmentation runs inside a recommender system.

Problem	Age, gender and geography turned out to be weak predictors of what a member will actually watch.
Approach	Netflix clusters members on viewing behaviour alone into **taste communities**: about 1,300 in 2017 (93 million members), roughly 2,000 by 2018 (137 million members). A member can belong to several communities at once. (2026: 325M!)
Result	The communities shape each member's homepage. Recommendations influence about 80% of the hours streamed, and Netflix estimates the system saves over $1 billion a year through reduced cancellations.

- The groups were not designed by anyone: they are found in the viewing data, exactly the task of a clustering algorithm.

Gomez-Uribe and Hunt (2015), ACM TMIS; community counts from Netflix press reporting, 2017 to 2018.

Customer segmentation at Tesco

The classic UK segmentation story began at the supermarket till.

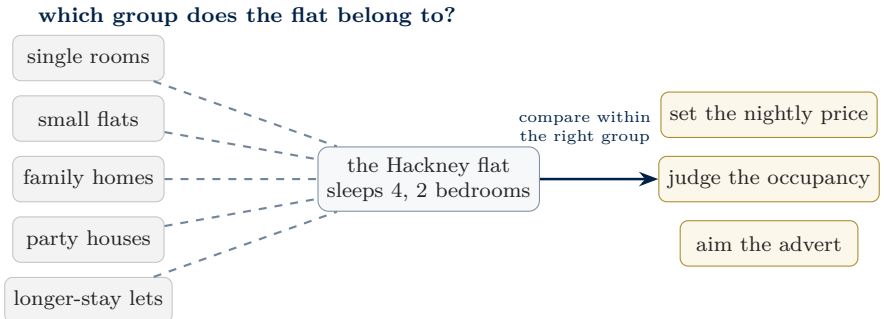
Problem	By the 1990s Tesco recorded millions of till transactions but could not connect them to customers: it did not know who bought what.
Approach	Clubcard (1995) linked purchases to households; the analytics firm dunnhumby then segmented shoppers by purchasing behaviour into named lifestyle groups, such as price-sensitive, mainstream and finer-foods shoppers, and targeted offers to each.
Result	dunnhumby reported targeted-coupon redemption above 20% (reward vouchers above 90%), against the low single digits typical of untargeted mailings. In 1995, the year Clubcard launched, Tesco became the UK's largest grocer.

"What scares me about this is that you know more about my customers after three months than I know after 30 years." (Ian MacLaurin, then Tesco chairman, on the first Clubcard analysis)

Case reporting: Marketing Week; Computer Weekly. Redemption figures as reported by dunnhumby.

Market segmentation of London short-let listings

Day 2 built a pricing model for a Hackney flat. The host's next question: which other listings does this flat actually compete with? The comparison group determines the price level, the occupancy estimate, and where to advertise.



The market clearly contains different types of listing, but the data has no column recording the type. With 32,749 listings, manual labelling is not realistic: the groups must come from an algorithm.

London listing data and validation design

We use the London listings data from Day 2, with the same host-level split:

Cases	32,749 fitting listings; 11,063 validation listings from other hosts
Inputs (four)	guest capacity, bedrooms, beds, minimum nights: the physical characteristics recorded for every listing
Typical values	capacity 3.5, 1.6 bedrooms, 2.0 beds, minimum stay 2.0 nights
Excluded on purpose	the quoted nightly price

> **Why exclude price?** We create the groups without using price. If prices later differ across groups, that is independent evidence that they capture meaningful market segments.

The inputs describe the listing; the price is reserved for validation.

Feature-based similarity between listings

K-means groups *similar* listings. Before any algorithm can run, *similar* must be turned into a calculation, because the algorithm never sees flats, only rows of numbers:

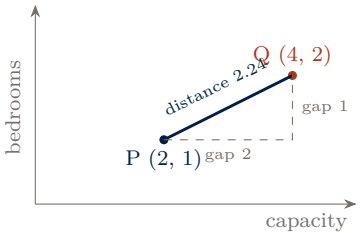
(invented examples)	Capacity	Bedrooms	Beds	Min nights
listing P	2	1	1	2
listing Q	4	2	2	2

- The standard solution: measure **distance** between the two rows. Small distance means similar listings; large distance means different ones.
- The distance must use all four features, and it must treat them fairly. Both points take one slide each.

First a distance for two listings; then the algorithm that applies it to 32,749 at once.

Euclidean distance between listings

With two features the calculation can be drawn: each listing is a point, and distance is the straight line between the points.



$$d(P, Q) = \sqrt{2^2 + 1^2} = \sqrt{5} \approx 2.24.$$

each squared term is one feature's gap between the two listings · the rule is Pythagoras' theorem

- With four features there is no picture, but the calculation is identical: square the four gaps, add them, take the root.
- For P and Q above: $d = \sqrt{2^2 + 1^2 + 1^2 + 0^2} = \sqrt{6} \approx 2.45$.

One formula turns two rows of numbers into one similarity number. Next: why the raw numbers cannot be used as they are.

Why raw distance is scale-dependent

Distance adds gaps in whatever units the data happens to use. Three invented listings show why that is dangerous. B is a bigger flat; C is the same size as A but asks a longer minimum stay:

	Capacity	Min stay	Distance to A (nights)	Distance to A (weeks)
listing A	2	2 nights		
listing B	4	2 nights	$\sqrt{2^2 + 0^2} = 2.0$	$\sqrt{2^2 + 0^2} = 2.0$
listing C	2	5 nights	$\sqrt{0^2 + 3^2} = \mathbf{3.0}$	$\sqrt{0^2 + 0.43^2} = \mathbf{0.43}$

- Recorded in nights, B is A's nearest neighbour ($2.0 < 3.0$). Recorded in weeks, the 3-night gap becomes $3/7 = 0.43$, and now C is nearest ($0.43 < 2.0$).
- Same listings, same facts, opposite grouping. The answer came from a clerical choice of unit, not from the market.

Raw distance is not usable. We need a version of each feature that does not depend on its unit.

Standardising the listing features

Learn one mean and one standard deviation for each feature from the $n = 32{,}749$ fitting listings:

$$\bar{x}_j = \frac{1}{n}\sum_{i=1}^{n} x_{ij}, \qquad s_j = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)^2}, \qquad z_{ij} = \frac{x_{ij} - \bar{x}_j}{s_j}.$$

Quantity (rounded)	Capacity	Bedrooms	Beds	Minimum nights
Mean $\bar{x}_j$	3.5	1.6	2.0	2.0
SD s_j	2.2	1.0	1.4	1.4

One ruler per feature, not per listing. Every listing reuses these four means and spreads and receives four z-values. $z = 0$ means average; $z = \pm 1$ means one spread above or below average.

Standardisation turns every feature into "typical-spread" units, so changing nights to weeks cannot change the grouping.

Worked example: standardised feature values

Use the three invented listings from the units example. The rounded fitting-data rulers are: capacity ($\bar{x} = 3.5,\ s = 2.2$) and minimum nights ($\bar{x} = 2.0,\ s = 1.4$).

Listing	Raw values (capacity, nights)	Capacity z	Minimum-nights z
A	(2, 2)	$(2 - 3.5)/2.2 = -0.68$	$(2 - 2.0)/1.4 = 0$
B	(4, 2)	$(4 - 3.5)/2.2 = 0.23$	$(2 - 2.0)/1.4 = 0$
C	(2, 5)	$(2 - 3.5)/2.2 = -0.68$	$(5 - 2.0)/1.4 = 2.14$

Pair the two z-values to obtain $A = (-0.68, 0)$, $B = (0.23, 0)$, and $C = (-0.68, 2.14)$. Then:

$$d(A, B) = \sqrt{[0.23 - (-0.68)]^2 + [0 - 0]^2} = 0.91,$$
$$d(A, C) = \sqrt{[-0.68 - (-0.68)]^2 + [2.14 - 0]^2} = 2.14.$$

$0.91 < 2.14$, so B is closer to A. Relative to normal market variation, the capacity gap is smaller than the minimum-stay gap.

K-means clustering: the complete algorithm

With a unit-proof distance in place, the grouping algorithm can be stated in full. **K-means** builds k groups around k **centres**:

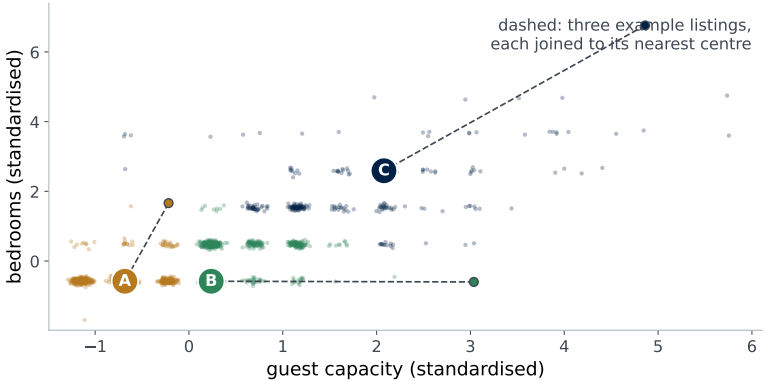
1. Choose the number of groups, k, in advance. (How to choose it is examined later in this part.)
2. Start with k provisional centres, for example k listings picked at random.
3. **Assign**: each listing joins its nearest centre, by the standardised distance.
4. **Update**: each centre moves to the mean of the listings assigned to it.
5. Repeat steps 3 and 4 until no listing changes group.

The next three slides run this loop on real listings.

‣ what k-means minimises: appendix A2

K-means step 1: assign listings to centres

2,000 real listings, two features so the space can be drawn, three centres named A, B and C.

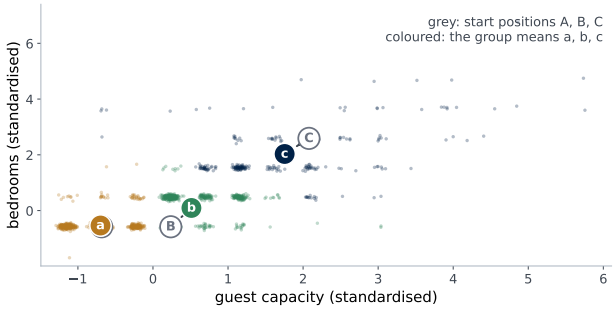


- Each listing takes the colour of its nearest centre.
- The random start is visibly poor. The loop will now repair this.

Standardised capacity and bedrooms; one extreme listing lies above the top edge.

K-means step 2: update the centres

Each centre now moves to the mean of its own group:

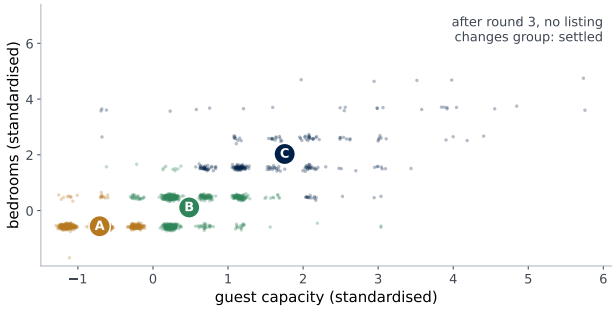


- Each centre is pulled to the average position of its own group: B climbs to b, and C shifts down into its group's middle at c.
- A barely moves: it already sat at its group's mean, so a lands on top of A. A well-placed centre stays put.

Same listings and groups as the previous slide; arrows show each move.

K-means convergence

Assignment and update alternate. When a full round changes no listing's group, the loop stops:



- After round 3 the demonstration is settled: A holds the small listings, B the mid-sized, C the large.
- Settled means one more round would change nothing, not that the grouping is officially correct.

The settled assignment and centres after three rounds of assign and update.

Within-group sum of squares

To score a proposed grouping, look at every listing's standardised distance from the centre of the group to which it was assigned:

$$W = \sum_{\text{all listings}} \big(\text{standardised distance to its own centre}\big)^2.$$

one contribution per listing · a distance of 2 contributes $2^2 = 4$ · distant listings are penalised more

- Add all contributions: small W means compact groups; large W means more spread-out groups.
- Compare W only for the same data and the same k. A small W shows compactness, not that the groups are useful.

K-means repeatedly changes assignments and centres to reduce W.

▸ a worked round: appendix A2b

Why k-means converges

Track W through the demonstration run, after each half-step:

Round	W after assign	W after update	
1	4,237	2,621	the poor start is largely repaired
2	2,407	2,346	smaller corrections
3	2,346	2,346	nothing changes: settled

- **Assign** can only lower W: a listing moves only to a nearer centre.
- **Update** can only lower W: the mean is the point with the smallest squared distance to its group, Day 2's fact.
- W falls at every half-step and cannot fall below zero, so the loop must settle. It settles at *a* low point, not always the lowest: run ten random starts, keep the smallest W.

‣ the two facts, formally: appendix A2

K-means specification for London listings

The demonstration used two features and three centres so it could be drawn. The real question needs the full data:

Listings	the 32,749 fitting listings
Inputs	the four standardised features: capacity, bedrooms, beds, minimum nights
Number of groups	$k = 4$ for now; the choice is examined once the result is on the table
Starts	ten random starts; keep the run with the smallest W

- The question, stated before the answer: given only physical facts, and no labels of any kind, what types of listing will the algorithm propose?

The output is four centres and a group for every listing. The next slide reads them.

Four London listing segments

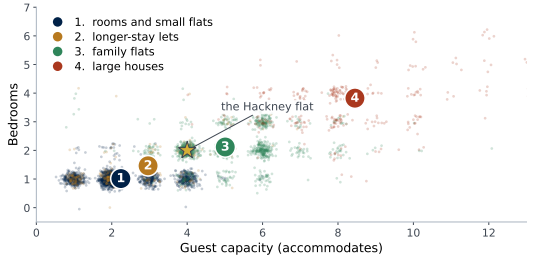
Fitting k-means with four groups on the 32,749 listings gives four centres, reported back in the original units for reading:

Segment	Guests	Bedrooms	Beds	Min nights	Share
Rooms and small flats	2.2	1.0	1.3	1.5	57.0%
Longer-stay lets	3.0	1.5	1.7	5.3	9.4%
Family flats	5.0	2.1	2.7	1.8	26.5%
Large houses	8.4	3.8	5.3	2.4	7.1%

- The names describe each centre's size and stay length. They are added after the fit, never used by it.
- The longer-stay lets stand out on minimum nights (5.3), not on size, so size axes alone will not separate them.

Separation and overlap among the four segments

The averages cannot show whether the groups are compact or blurred. Draw every listing on two of the four inputs, coloured by segment; the numbered badges mark the centres.



- Segments **1**, **3**, **4** (rooms, family flats, large houses) fan out cleanly by size.
- Segment **2**, longer-stay lets, overlaps segment 1: what sets it apart, a five-night minimum stay, is on neither axis. This flat view is a shadow of the full four-input space.
- The gold star is the Hackney flat, sitting inside the family-flats cloud (segment 3).

4,000 fitting listings, jittered; badges 1–4 are the segment centres.

Assigning a new listing to a segment

The section opened by asking which group the Hackney flat belongs to. Placing it uses only the fitted objects: means, spreads, centres.

The flat: entire home, capacity 4, 2 bedrooms, 2 beds, minimum nights 2 (the gold star on the previous slide).

Step	Result
1. Standardise	z: guests 0.24, bedrooms 0.48, beds 0.02, minimum nights 0.01
2. Distance to each centre	segment 1: 1.47; segment 2: 2.50; segment 3: **0.74**; segment 4: 3.71
3. Nearest wins	segment 3, family flats: its centre is closest at 0.74

The competitor set is found: the flat sits among the family flats. Every new listing is placed by this same arithmetic; this is the recipe a deployed system runs.

Held-out validation on unseen hosts

Are the four segments real, or an accident of our sample and recipe? The week's first rule applies unchanged: judge on data the fit never saw.

- **The mechanics:** freeze the four centres, then place each of the 11,063 validation listings by the walkthrough arithmetic. Nothing is refitted.

Segment	Share, fitting	Share, unseen hosts
Rooms and small flats	57.0%	57.2%
Longer-stay lets	9.4%	9.6%
Family flats	26.5%	26.2%
Large houses	7.1%	6.9%

- Every share lands within half a point: the structure travels to hosts the fit never touched.

Held-out structure: passed. Two checks remain.

Stability across samples and random starts

A second doubt: the demonstration showed the start matters, and Day 3 showed small samples changing a tree's rules. Segments that depend on one sample or one seed are not market structure. **Test 1, different data:** re-cluster two random halves from scratch:

	Half A (16,374)		Half B (16,375)	
Segment	cap	share	cap	share
Rooms and small flats	2.3	57.0%	2.2	57.3%
Longer-stay lets	3.0	9.4%	3.0	9.3%
Family flats	5.0	26.6%	5.0	26.3%
Large houses	8.5	7.0%	8.4	7.1%

- **Test 2, different starts:** five fresh random starts (seeds 0 to 4) reproduce the canonical grouping exactly, adjusted Rand index 1.000 each time.

Stability: passed. The last check is the hardest.

External validation using withheld prices

Now the plan from the data slide pays off. Price was excluded so it could serve as an independent check: if the segments are real market types, they should differ on a fact the algorithm never saw.

- Median prices £126, £169, £289, £391:
- Among unseen hosts the same ladder reappears: £116, £170, £285, £381.

Median quoted price per night, fitting listings.

Downstream audit: passed. The segments carry real market information, not an echo of the inputs.

Summary of clustering validation

Compact groups are not enough. Before interpreting them, ask whether they travel, repeat and matter:

Check	Concern	Evidence here
Generalisation (held-out)	Specific to the fitting hosts?	Frozen centres give almost identical shares on unseen hosts.
Stability (sample and start)	Would new data or a new start change the groups?	Half-sample refits give similar centres and shares; fresh starts give the same assignments.
Relevance (price audit)	Compact but commercially uninformative?	Withheld median prices differ roughly threefold; the ordering repeats on unseen hosts.

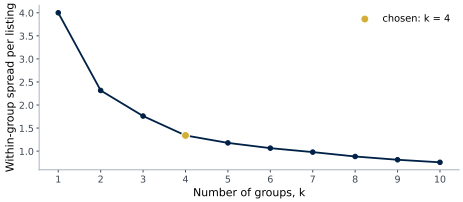
What they support: the segments extend beyond the fitting sample, are reproducible and carry useful market information.

What they do not prove: natural "true" groups, causal effects, or that $k = 4$ is the only defensible choice.

Validation comes before interpretation or use. Next: document the choice of k.

Selecting the number of clusters

Why four groups? No formula fixes k; the loop's own score gives a guide:



- The curve is the loop's own score, W per listing, refit for each k: it falls at *every* k, as more depth always fitted better on Day 3. It suggests, never decides.
- Steep to $k = 4$ (4.0, 2.3, 1.8, 1.3), shallow after (1.2 at $k = 5$): beyond four, extra groups buy little.
- We keep four: the longer-stay segment is commercially meaningful; three or five would be defensible. Documented, revisable.

A silhouette score gives a second view of the same call.

▸ scores for choosing k: appendix A3

Limits of the clustering results

Four segments passing three checks is a strong result. It is still only a description:

- **Descriptions, not prescriptions:** "family flats" names a size, it does not tell a host to raise the price.
- **Soft borders:** a listing sitting between two centres could fall either way; our Hackney flat was clear (0.74 versus 1.47), others are not.
- **Ruler-dependence:** a different feature set or a different scaling can redraw the borders. The standardising choice is part of the result.
- **Not causes:** the segments show which listings resemble each other, not why any price is what it is.

Clustering finds structure worth reviewing, never a rule for action on its own.

Key concepts in k-means clustering

Concept	What it is	In our exercise
Centre	The mean of a group's listings across the four features.	The four crosses in the scatter.
Assignment	Sending each listing to its nearest centre.	Places the Hackney flat in segment 3.
Within-group spread	Total squared distance from listings to their own centre.	Per listing: 4.0 down to 1.3 as k reaches 4.
Standardisation	Rescaling each feature to spread 1 before measuring distance.	So guests do not outweigh bedrooms.
k	The chosen number of groups.	Set to 4, and documented.
Silhouette	A score comparing closeness-in with closeness-out (appendix A3).	A second check on k.

All three checks passed: travelling shares, a stable grouping, a price gap the fit never saw.

Using segments for market comparison

The question that opened this part now has a working answer:

The host's decision	What the segmentation now provides
set the nightly price	compare the quote inside **family flats** (median £289), not against all of London
judge the occupancy	the right peer group: 26.5% of the market, not the 57% of small rooms and flats
aim the advert	the peer group says who the guests are: families and small groups of four or five

- The same recipe serves every host in the file, and any new listing, by the walkthrough arithmetic.
- The evidence behind the answer: three checks, passed and documented.

Part III asks the day's second question: how few forces move a whole curve of interest rates at once?

Part III · Principal component analysis of the yield curve

Principal component analysis

Many datasets have dozens of columns that move together: survey questions, factory sensors, interest rates. Most of that detail is redundant, because a few underlying forces drive all the columns.

Principal Component Analysis replaces many correlated measurements with a few uncorrelated summary numbers (the **principal components**) that keep as much of the variation as possible, ranked from most informative to least.

Three real cases first; then we compress a real yield curve.

PCA for face recognition

The classic image application of PCA.

Problem	A face photo is tens of thousands of pixels; comparing raw pixel grids to recognise a person is slow and fragile.
Approach	Run PCA on many face images. The leading components are themselves face-like images, the **eigenfaces**; any face is then approximated as a weighted sum of about 100 to 150 of them.
Result	A face becomes a short list of about 100 weights instead of 10,000 pixels, roughly a hundredfold compression, powering one of the first near-real-time face-recognition systems.

- The same move we will make on rates: replace many raw numbers with a few weighted blends that keep the essentials.

Turk and Pentland, “Eigenfaces for recognition” (1991).

PCA of genetic variation in Europe

The most striking demonstration that PCA finds real structure.

Problem	Each person carries hundreds of thousands of DNA markers. Is there hidden low-dimensional structure, and what is it?
Approach	PCA of 197,146 DNA markers across 1,387 Europeans; plot each person using only their first two component scores.
Result	The two-number scatter reproduces the map of Europe: the 1st component lines up north to south, the 2nd east to west. Half of people place within about 310 km of their true origin.

- No labels, no target: PCA was told only "find the directions of most variation", and geography fell out.

Novembre et al., "Genes mirror geography within Europe," *Nature* (2008).

PCA of the US yield curve

A treasury or risk desk tracks many US government bond yields every day, from the three-month bill to the thirty-year bond.

- Managing risk maturity by maturity is unwieldy, and the maturities mostly move together, so the numbers repeat each other.
- The desk wants a few named directions it can put in a risk report, stress-test in words ("the curve steepens"), and discuss at a committee.
- In 1991 Litterman and Scheinkman applied PCA to Treasury returns and found that three forces run the whole curve. We reproduce their result on 36 years of data.

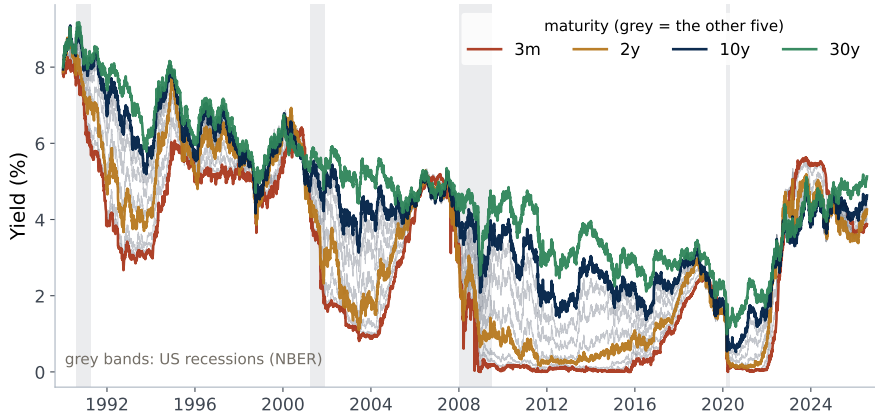
The same tool that compressed faces and the genome, now on a curve of interest rates.

US Treasury yield data

Series	nine constant-maturity Treasury yields: 3m, 6m, 1y, 2y, 3y, 5y, 7y, 10y, 30y
Span	9,143 trading days, January 1990 to July 2026 (the 20-year is excluded: Treasury suspended it from 1986 to 1993)
Measurement	daily *changes* in yield, in percentage points, not the levels
Regimes covered	8% rates of the early 1990s, the 2008 crisis and near-zero era, and the 2022 rate shock

A long window on purpose: if three shapes hold across 36 years of very different rate regimes, they are structural, not a quirk of one calm decade.

Co-movement across Treasury maturities

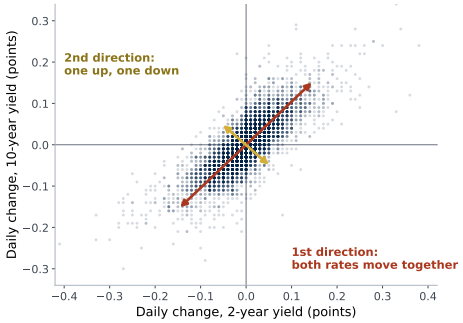


- The nine lines rise and fall mostly together; the average correlation of their daily is 0.67.
- Nine numbers arrive each day, but plainly fewer than nine *things* are happening.

FRED constant-maturity Treasury yields; 9,143 trading days, January 1990 to July 2026.

Principal components as directions of maximum variance

Start with just two of the nine: the daily changes of the 2-year and the 10-year yields.



- The cloud is a tilted ellipse: most days both rates move together (long axis); now and then they part (short axis).
- **PCA** finds these directions, for all nine maturities at once, ordered by how much movement each carries. The first is the long axis.

9,142 daily changes; arrows are the two principal directions, computed from the cloud.

Loadings and component scores

A **component** is a fixed recipe of weights, one per maturity. Each day it turns the nine changes into a single number, the **score**:

$$\text{score}_t = w_1\,\Delta_{1t} + w_2\,\Delta_{2t} + \cdots + w_9\,\Delta_{9t}, \qquad w_1^2 + \cdots + w_9^2 = 1.$$

Δ_{jt}: today's change in maturity j · w_j: the fixed weight (**loading**) the recipe puts on maturity j · score_t: today's one-number reading

- The weights are chosen once and reused every day; only the daily changes vary. The weights are the recipe; the score is the reading.
- Unit length ($\sum_j w_j^2 = 1$) fixes the overall scale, so scores from different recipes are comparable.

▸ how the weights are computed: appendix A4

Variance maximisation and orthogonality

Two rules pin down the recipes, in order:

1. **Most spread first.** The 1st recipe is the one whose score varies as much as possible across the days: the single direction that captures the most daily movement.
2. **Right angles after.** Each later recipe is chosen at right angles (**orthogonal**) to all earlier ones, so it captures variation the earlier ones missed, never repeats it.

- Right angles have a payoff: the scores are *uncorrelated*, so their variances simply add up, with no double-counting.
- Three orthogonal scores can therefore be read as three independent dials.

Ordered by how much they explain, and independent by construction.

‣ checked on real recipes: appendix A4b

Covariance PCA without standardisation

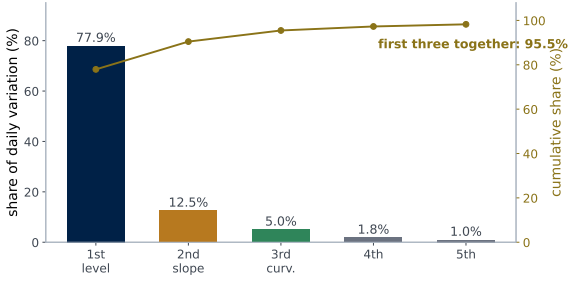
In the clustering we standardised, because guests and bedrooms used different units. Here the decision goes the other way.

- All nine rates are already in one unit, percentage points, so a one-point move means the same thing at every maturity.
- We therefore do *not* standardise: we let a maturity that genuinely moves more count for more (covariance PCA, not correlation PCA).
- Had the inputs mixed units, say yields with trading volumes, we would standardise first, or the largest-numbered column would dominate every component by scale alone.

The scaling choice is a modelling decision, made on purpose and documented: the same discipline as clustering, the opposite call.

Selecting the number of components

Each component reports the share of daily variation it captures. List them in order, the largest drop tells you how many to keep:



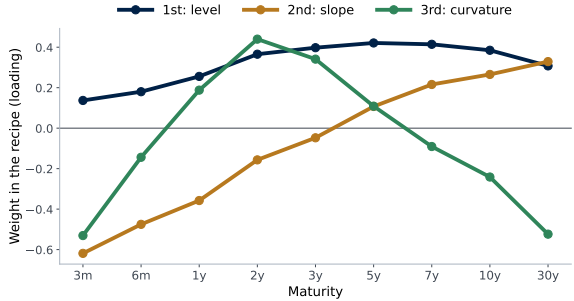
- The 1st holds 77.9%, the 2nd 12.5%, the 3rd 5.0%: together **95.5%**. Components 4 and beyond each hold under 2%.
- Nine dials collapse to three numbers a day, with 4.5% left behind. Day 1 previewed this compression.

Covariance PCA on 9,142 daily changes.

‣ formal selection rules: appendix A4e

Yield-curve loadings: level, slope, and curvature

Draw each recipe's weights across the maturities and the shapes name themselves:

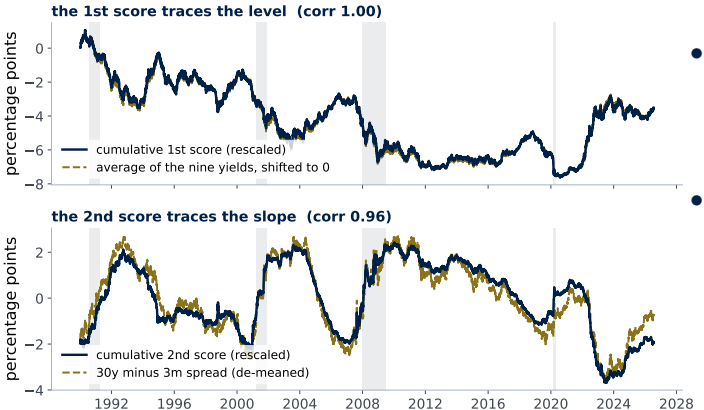


- **Level** (77.9%): all maturities the same sign, the whole curve shifts up or down together.
- **Slope** (12.5%): short and long ends opposite signs, the curve steepens or flattens.
- **Curvature** (5.0%): the belly moves against the two ends, the curve bows.

Litterman and Scheinkman (1991) named these shapes; our 36-year panel reproduces them. Signs are oriented only for readability.

Component scores over time

A component is not just algebra. Cumulate each daily score and it redraws a curve summary a desk already knows:



- The 1st score, added up, traces the average **level** of the nine yields almost exactly (correlation 1.00).
- The 2nd score traces the **slope**, the gap between the 30-year and 3-month rates (0.96).

Cumulative daily scores, rescaled to their comparators for display; grey bands are US recessions.

Yield-curve scores on 19 September 2008

The single largest level day in 36 years fell in the week Lehman Brothers failed. All nine yields jumped the same way:

maturity	3m	6m	1y	2y	3y	5y	7y	10y	30y
change (points)	+.76	+.75	+.52	+.38	+.37	+.34	+.29	+.24	+.22

- Run them through the level recipe, weight times change summed across the nine: $0.14(0.76) + 0.18(0.75) + \cdots + 0.31(0.22) = 1.08$.
- That 1.08 is the largest level score in all 9,142 days: one number captures “the whole curve lurched up on the same day”.

Every day is scored by this same recipe.

‣ the weights at work: A4c ‣ rebuild the day: A4d

Stability of PCA results across periods

Are level, slope and curvature real, or an artefact of one period? Split the 36 years into two eras and refit PCA in each from scratch:

Era	Variance shares (level / slope / curv.)	Loading agreement with full sample
1990–2007 (high rates)	79.9 / 12.1 / 3.8	0.98 / 0.98 / 0.98
2008–2026 (zero, then 2022)	76.7 / 12.7 / 5.9	0.98 / 0.97 / 0.97

- Rates went from 8% to near zero and back, yet all three shapes reproduce with 0.97 to 0.98 agreement in *both* eras.
- This is the payoff of the long window: strong evidence the three shapes are structural, not a quirk of one calm decade.

Applications of PCA to yield-curve risk

Three scores a day replace nine correlated rates, and each buys something concrete:

- **Risk in three numbers.** A day's curve exposure reads as level, slope and curvature, not nine overlapping figures.
- **Stress tests in plain words.** "The curve shifts up", "it steepens", "it bows" are directions a committee can name and shock, not nine separate what-ifs.
- **Cleaner inputs.** A downstream model takes three near-independent directions instead of nine rates that mostly repeat each other.

The value is a shared language for curve risk that a person can read and a model can reuse.

Limits of principal component analysis

The three shapes are a fair summary of the daily variation, and no more:

- **Movements, not causes.** They describe how the curve moves, not what makes it move.
- **Variance is not importance.** A large share means good compression; a small component can still carry the risk that matters on an extreme day.
- **Signs are arbitrary.** Flip every weight and every score and the component is unchanged; the up/down orientation is a convention.
- **Panel-specific.** The shapes hold for these nine maturities on this window; a different panel or period can reweight them.

Compression is task-neutral. Whether a direction matters belongs to the later decision, tested against its own scenario.

Key concepts in principal component analysis

Concept	What it is	In our exercise
Component	A fixed recipe of weights across the maturities.	Three of them summarise the curve.
Loading (weight)	The weight a component puts on one maturity.	Its shape names the force: level, slope, curvature.
Score	Today's changes run through the recipe: one number.	The daily reading a desk can monitor.
Variance share	Fraction of daily variation a component holds.	77.9 / 12.5 / 5.0%; compression, not importance.
Same-units rule	Same-unit inputs need no standardising.	Bigger-moving maturities weigh more, by design.

Found, read as level/slope/curvature, and stable across 36 years of very different rates.

Yield-curve PCA: nine rates to three components

The desk's problem that opened this part now has a working answer:

The desk needed	What the three components provide
a short risk report	level, slope, curvature: three numbers holding 95.5% of the daily action
stress tests in words	shift the curve, steepen it, bow it: one component moved at a time
trust to reuse it	the same three shapes across 1990–2007 and 2008–2026, agreement 0.97 to 0.98

- The same tool compressed a face to about 100 numbers and a genome to a two-number map. Here it is nine rates to three shapes.

Two unsupervised tools now in hand: clustering finds groups, PCA finds directions. Part IV takes both to their hardest ground: models pointed at people.

Part IV · Risks in AI-assisted decisions

Consequences of errors in AI-assisted decisions

Everything this week scored things: prices, rates, response records. The same machinery is now routinely pointed at people, at scale:

- About 99% of Fortune 500 companies filter job applicants through automated tracking systems; 88% of employers say qualified candidates get screened out before a human looks (Harvard Business School, 2021).
- In McKinsey's 2024 survey, 65% of organisations were already using generative AI in at least one business function.
- A mispriced flat costs pounds; a misjudged person can lose a loan, a job, or years of their life. The 712 missed subscribers we tolerated were a budget line; on people, the same error rates become questions of justice.

Nothing in the mathematics changes. Every duty does. This part is a casebook: real systems, real people, and the lesson each case leaves behind.

Historical bias in model targets

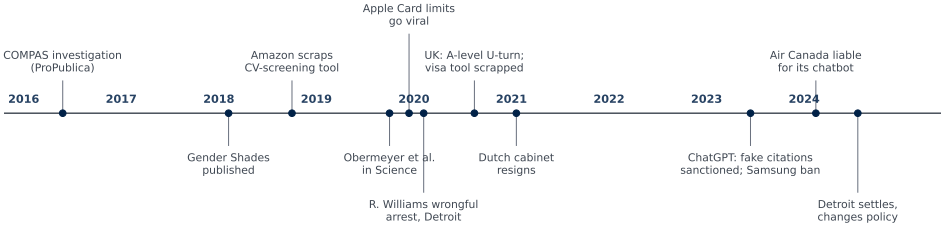
The week's quiet lesson, stated plainly: a model learns the target it is given, and the target carries the past.

- Our own target was "said yes in 2008 to 2010", not "should be called": the model inherited the old campaigns' choices.
- Day 1's recruiting case, in full: Amazon trained an experimental CV screener on a decade of its own hires; it learned the past's imbalance, marked down CVs containing "women's", and was scrapped (Reuters, 2018).
- Day 1's healthcare case: "cost" stood in for "need" in a system applied to about 200 million people a year; fixing the proxy would nearly triple the share of black patients flagged for extra care, 17.7% to 46.5% (Obermeyer et al., *Science*, 2019).

No malice is required anywhere in the pipeline. The target does the damage on its own.

Documented failures of automated decision systems

None of what follows is hypothetical. Every event on this line has a named person, a court ruling, a regulator's report, or a resignation attached:

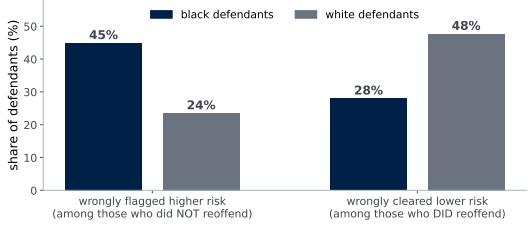


- The pattern repeats across justice, hiring, credit, education, immigration, welfare and customer service, under different laws.
- The next slides take six apart, each with the same question: what exactly failed, and what would have caught it?

Each event is sourced on its case slide and in the Sources pages.

Fairness trade-offs in the COMPAS score

COMPAS: *Correctional Offender Management Profiling for Alternative Sanctions.* It estimates reoffending risk. ProPublica compared 7,000 Florida scores with two-year outcomes:



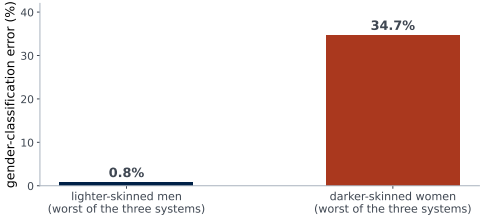
- Overall accuracy was similar by race; the *mistakes* were not: they fell on opposite groups, in opposite directions.
- The vendor's defence was also true: a given score meant the same reoffending rate in both groups. Both properties cannot hold at once when base rates differ, so fairness is a *choice*, and someone must own it.

ProPublica, "Machine Bias" and methodology (2016); impossibility: Kleinberg et al. (2016), Chouldechova (2017).

‣ the arithmetic: appendix A5

Subgroup error rates in face analysis

In 2018 Buolamwini and Gebru tested three commercial face-analysis systems (Microsoft, IBM, Face++), each with a high advertised overall accuracy, on a balanced set of faces. By subgroup:



- Error rates ran from under 1% for lighter-skinned men to 34.7% for darker-skinned women, a gap invisible in the single headline number.
- The audit worked: vendors cut the gaps within months, and in June 2020 IBM withdrew from general-purpose face recognition altogether.

Buolamwini and Gebru, *Gender Shades*, PMLR 81 (2018). Test one confusion matrix per group: this week's tool, applied.

Wrongful arrests from face-recognition matches

What happened	January 2020: Detroit police ran a grainy CCTV still through face recognition; it returned Robert Williams' licence photo. He was arrested in front of his family and held about 30 hours. He had nothing to do with the crime.
Not a one-off	In 2023 Porcha Woodruff, eight months pregnant, was wrongly arrested the same way. At least three such wrongful arrests are documented in Detroit alone.
The outcome	The ACLU sued; in 2024 Detroit paid $300,000 and adopted one of the strictest US police policies: a face match alone can never justify an arrest, only a lead to corroborate.

- Three numbers were confused: the vendor's lab accuracy, the subgroup error rate (previous slide), and what the output *is*: a lead, not an identification.

A score treated as proof: the exact failure the Dutch affair repeats at national scale.

The 2020 A-level grading model

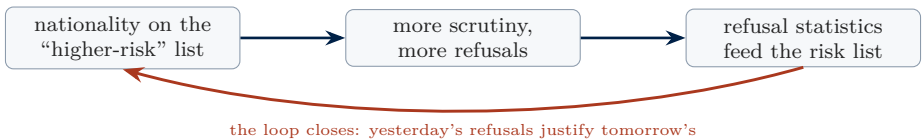
The task	Covid cancelled England's 2020 exams. Ofqual's model adjusted teacher-assessed grades to keep national results in line with history, leaning on each school's past performance.
What happened	About 39% of teacher-assessed A-level grades were pulled down. Small cohorts, often private schools, largely kept teacher grades; strong pupils in large state schools were capped by their school's history. Street protests followed.
The outcome	On 17 August 2020 the government reversed course and restored teacher grades; the chief regulator resigned within days; the Prime Minister blamed a "mutant algorithm".

- The model did roughly what it was designed to do. The design goal, hold the national distribution steady, was itself the injustice: a pupil was graded by her school's past, not her own work.

Accurate on average is not acceptable case by case. For a policy maker, the algorithm *is* the policy.

Nationality and feedback loops in visa screening

The system	From 2015 the UK Home Office streamed every visa application red, amber or green. Nationality was an input; a secret list of "higher-risk" nationalities pushed applicants into the red stream.
The outcome	Challenged as discriminatory by JCWI and Foxglove, the Home Office scrapped the tool on 7 August 2020, days before the case would have been heard. Foxglove's director called it "speedy boarding for white people".



- The Dutch affair's two ingredients, nationality as an input and a self-reinforcing loop, caught earlier because someone sued. External scrutiny, not internal accuracy metrics, stopped the harm.

Transparency and the Apple Card investigation

What happened	In November 2019 complaints went viral, including from Apple's co-founder, that wives received far lower credit limits than husbands with shared finances. New York's financial regulator investigated Goldman Sachs, the issuer.
The finding	After reviewing about 400,000 applications, the regulator found *no unlawful discrimination*: similar credit profiles got similar limits, and the card scored individual, not household, credit histories.
The critique	The same report faulted the programme anyway: customers could not get an intelligible reason for their limit, and "the algorithm decided" satisfied no one.

- Two lessons in one case: a viral anecdote is not an audit; and an audit is not an excuse for opacity. An unexplainable decision is a business failure even when it is lawful.
- The questions stay open at scale: across two million 2019 US mortgage applications, lenders were about 80% more likely to deny black applicants than similar white ones, 17 factors held constant (The Markup, 2021; an estimate, not a ruling).

Organisational liability for generative AI outputs

Three documented failures from the generative-AI era, each with a price attached:

Case	What happened	The bill
Mata v. Avianca (US court, 2023)	lawyers filed six case citations ChatGPT had invented; asked whether they were real, ChatGPT said yes	$5,000 sanction, careers dented
Moffatt v. Air Canada (2024)	the airline's chatbot invented a refund policy; the airline argued the bot was "a separate legal entity" and lost	CAD $812, and the precedent
Samsung (2023)	engineers pasted confidential source code into ChatGPT to debug it; the material could not be recalled	a company-wide ban

- A fluent answer is not a sourced answer; asking the model to check itself is not verification. Your organisation owns what its AI says, and what staff type into a public tool leaves the building.

Dutch childcare benefits scandal: case overview

The Netherlands, 2005 to 2019. The tax authority screened childcare benefit claims with a self-learning risk model, and treated high risk scores as grounds for action.

- Around 26,000 parents were wrongly accused of fraudulent claims, and many were ordered to repay years of benefits in full, immediately.
- Among the inputs was the claimant's nationality. Families with dual citizenship were flagged disproportionately.
- The consequences ran through debt, lost homes and broken families; reporting has linked more than a thousand children's foster placements to the affair, with some estimates higher.
- In January 2021 the Dutch government resigned over the scandal.

Parliamentary inquiry *Unprecedented Injustice* (2020); Amnesty International, *Xenophobic Machines* (2021).

Dutch childcare benefits scandal: system failures

Set the scandal against this week's own checklist:

The week's tool	What it would have asked
error rates by group (Day 1)	who is being falsely flagged, and is any group carrying the mistakes?
asymmetric costs (Day 2)	a wrongly accused family is not a €4 mistake; what threshold do the real costs imply?
drift and monitoring (Day 3)	is the model still valid on today's claimants?
audit before scale (today)	was the score ever checked against verified outcomes before it was acted on at scale?

- The deepest failure sat outside the model: **a risk score was treated as proof**, and families had to prove their innocence against it.

Common failure patterns and safeguards

The pattern	Seen in	The defence, from this week
a proxy target carries the past	Amazon hiring; Obermeyer; the bank's own target	interrogate the target column before fitting
an average hides the subgroups	COMPAS; Gender Shades	one confusion matrix per group
a score treated as proof	Williams arrest; the Dutch affair	a score is a lead; humans must stay able to say no
secret criteria, feedback loops	visa streaming; the Dutch affair	document inputs; audit against verified outcomes
right on average, wrong per person	the A-level algorithm	individual appeal routes; legitimacy is not a metric
fluent fabrication, owned words	Avianca brief; Air Canada bot	verify against sources; you own your AI's statements

Every defence in the right column is a tool this course already taught. The failures were failures of use, not of exotic mathematics.

Safeguards for high-stakes AI systems

- **Document** the target, the data's origin, the split, the scores, and the costs behind every threshold: this week's habits, written down.
- **Test by group**, not only on average: one confusion matrix per group the decision could harm.
- **Keep a human path**: a score triggers review by someone empowered to disagree, and the subject can appeal.
- **Monitor drift**: Day 3 showed models age; on people the ageing is silent until it is a scandal.
- **Know the law**: the EU AI Act classifies systems that assess creditworthiness, make employment decisions, or decide access to essential public services as high risk, with duties close to this list.

Predicting risk about a person is not evidence of wrongdoing by that person. Build every process as if that sentence will be tested in court, because it will be.

‣ the risk ladder: appendix A6

Checklist for evaluating AI-assisted decisions

The casebook compresses into questions anyone can ask, whichever seat you sit in:

As a…	Ask…
consumer	Can I get a reason for this decision, and is there a route of appeal to a human? (Apple Card; A-levels)
buyer or builder	What was the target, who chose it, and what history does it carry? What are the error rates *by group*, on held-out data, not the headline average? (Amazon; Obermeyer; Gender Shades)
user of AI output	Is this answer sourced, or only fluent? Who is accountable when it is wrong, and what am I typing into it? (Avianca; Air Canada; Samsung)
policy maker	Is the score used as a lead or as proof? Are the criteria inspectable, is there a feedback loop, and who audits outcomes at scale? (Williams; visa tool; the Dutch affair)

The questions cost nothing and would have caught every case in this part. Asking them is what "responsible AI" means in practice.

Part V · Course review and conclusions

Day 4 summary

Topic	What it delivered	How it was checked
Clustering (k-means)	four London market segments, built from four physical facts	unseen hosts, a change of sample, and the withheld price
Principal components	nine yields compressed to three named shapes carrying 95.5%	scores audited against the level and the spread; stable across two eras
Models on people	six documented failures and one national scandal	the failure patterns, and the questions that catch them

- Two unsupervised methods, and the discipline for using any model, supervised or not, on people.

Six principles for applied machine learning

1. Judge a model only on data the fit never saw.
2. The ideal prediction is the average among comparable cases; every model is a way of building "comparable".
3. Use only what is known at the prediction moment.
4. Every score needs a baseline, and every gain a price in the decision's own currency.
5. The split must rehearse the deployment: rows, groups, or calendar.
6. A prediction is not a cause, and predicting people is not proof about them.

These principles are model-independent: they apply unchanged to whatever method you meet next.

Further study and next steps

- **The book behind this course**: James, Witten, Hastie, and Tibshirani, *An Introduction to Statistical Learning*, free online, with labs. Chapters map almost day for day.
- **What was deliberately left out**: neural networks and deep learning, reinforcement learning, and causal inference; each deserves its own course, and the six principles apply unchanged in all of them.
- **The fastest way to learn more**: take one real dataset you care about and walk it through this week's route: splits first, baseline second, discipline throughout.

Thank you.

Data and methods sources

- Inside Airbnb, *London detailed listings*, snapshot of 19 June 2026 (CC BY 4.0). UCI Machine Learning Repository, *Bank Marketing*; Moro, Cortez, and Rita (2014).
- Federal Reserve Bank of St. Louis, FRED constant-maturity yield series; nine maturities (3-month to 30-year), frozen daily panel, January 1990 to July 2026. Recession bands: NBER dates via FRED series USREC.
- PCA origin: K. Pearson (1901); H. Hotelling (1933). R. Litterman and J. Scheinkman (1991), "Common factors affecting bond returns," *Journal of Fixed Income* 1(1), 54–61.
- M. Turk and A. Pentland (1991), "Eigenfaces for recognition," *J. Cognitive Neuroscience* 3(1), 71–86. J. Novembre et al. (2008), "Genes mirror geography within Europe," *Nature* 456, 98–101.
- Unstructured-data share: IDC, *Data Age 2025* (2018). ImageNet labelling: Deng et al., CVPR (2009); Gershgorn, *Quartz* (2017).
- G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*, 2nd edition (2021), chapter 12 for today.
- All fitted numbers are recomputed by the course build script (`build_day4_v4_figures.py`) from the frozen files, under the week's splits.

Case study and governance sources

- Office for National Statistics and University College London, *Output Area Classifications* (2011; 2021), ons.gov.uk.
- Netflix: C. Gomez-Uribe and N. Hunt (2015), "The Netflix recommender system," *ACM TMIS* 6(4); "taste communities" press reporting, 2017 to 2018.
- Tesco Clubcard (launched 1995) and dunnhumby: case reporting in *Marketing Week* and *Computer Weekly*.
- McKinsey & Company (2021), *Next in Personalization*.
- Parliamentary Interrogation Committee on Childcare Allowances (2020), *Unprecedented Injustice*; Amnesty International (2021), *Xenophobic Machines*.
- Z. Obermeyer, B. Powers, C. Vogeli, and S. Mullainathan (2019), "Dissecting racial bias in an algorithm used to manage the health of populations," *Science* 366(6464).
- Regulation (EU) 2024/1689 (the EU AI Act), Annex III.

Sources for automated-decision case studies

- COMPAS: J. Angwin et al., “Machine Bias” and methodology, *ProPublica*, 23 May 2016. Impossibility: J. Kleinberg, S. Mullainathan, M. Raghavan (2016), arXiv:1609.05807; A. Chouldechova (2017), *Big Data* 5(2).
- J. Buolamwini and T. Gebru (2018), “Gender Shades,” *Proceedings of Machine Learning Research* 81, pp. 1–15. IBM withdrawal: letter to US Congress, 8 June 2020.
- Wrongful arrests: ACLU, *Williams v. City of Detroit* (settled 28 June 2024, $300,000 and policy change); Porcha Woodruff coverage, August 2023.
- A-levels: A. Kelly (2021), *British Educational Research Journal* 47(3) (39.1% of grades downgraded); BBC News, August 2020.
- Visa streaming: JCWI and Foxglove judicial review; Home Office suspension, 7 August 2020 (TechCrunch; Computer Weekly).
- Apple Card: New York State Department of Financial Services, report on Apple Card and Goldman Sachs, 23 March 2021. Mortgages: E. Martinez and L. Kirchner, “The Secret Bias Hidden in Mortgage-Approval Algorithms,” *The Markup*/AP, 25 August 2021.
- LLMs: *Mata v. Avianca*, S.D.N.Y., order of 22 June 2023; *Moffatt v. Air Canada*, 2024 BCCRT 149, 14 February 2024; Samsung ban: Bloomberg, 2 May 2023.
- Adoption: Harvard Business School and Accenture (2021), *Hidden Workers: Untapped Talent*; McKinsey (2024), *The State of AI in Early 2024*.

Appendix · Technical details

A1a. Standardisation and unit invariance

Each feature is rescaled with the fitting listings' own mean and spread:

$$z = \frac{x - \bar{x}}{s}, \qquad \text{fitting spreads: } s_{\text{capacity}} = 2.2,\ s_{\text{bedrooms}} = 1.0,\ s_{\text{beds}} = 1.4,\ s_{\text{nights}} = 1.4.$$

x: a listing's raw value on one feature · $\bar{x}$, s: that feature's mean and spread on the fitting listings · after scaling, every feature has spread 1

Why the unit cancels. Changing the unit multiplies every value by a constant c (nights to weeks: $c = 1/7$). The mean and the spread are multiplied by the same c, so $z = (cx - c\bar{x})/(cs) = (x - \bar{x})/s$: the c divides out. In numbers: nights $3/1.4 = 2.1$; weeks $0.43/0.2 = 2.1$.

- Means and spreads come from the fitting listings only; the validation listings are transformed with the same numbers, like every preprocessing step this week.

The standardised value is identical in any unit: the units flip cannot happen after scaling.

‣ the distances, worked ‣ back to standardising

A1b. Worked standardised-distance calculation

The three listings from the units slide, taken through the full recipe (capacity spread 2.2, nights spread 1.4):

Pair	Raw gaps (cap, nights)	Standardised gaps	Distance
A to B	$2,\ 0$	$2/2.2 = 0.9,\ 0$	$\sqrt{0.9^2 + 0^2} = \mathbf{0.9}$
A to C	$0,\ 3$	$0,\ 3/1.4 = 2.1$	$\sqrt{0^2 + 2.1^2} = 2.1$

- B is A's nearest neighbour, $0.9 < 2.1$, and by A2a's algebra this ordering is the same in nights, weeks, or any other unit.
- Scaling does not erase genuinely large differences: a 28-night gap would be $28/1.4 = 20$ spreads, still enormous. Units stop pretending; real differences remain.

Without scaling, the widest raw feature sets the ruler and quietly dominates every distance.

back to standardising

A2a. K-means objective function

K-means looks for the grouping that makes the total within-group squared distance as small as it can:

$$W = \sum_{g=1}^{k} \sum_{i \in C_g} \|\mathbf{x}_i - \boldsymbol{\mu}_g\|^2.$$

C_g: the listings placed in group g · $\boldsymbol{\mu}_g$: that group's centre · $\|\cdot\|^2$: squared distance in the standardised space

Two facts make the alternation safe:

- **Assignment step:** moving each listing to its nearest centre can only lower W or leave it unchanged; no listing's term rises.
- **Update step:** replacing each centre by its group's mean can only lower W: the mean is the closest point in squared distance, Day 2's least-squares fact.

W never rises, so the run settles at *a* low point: ten random starts, keep the smallest.

‣ one worked round ‣ back to k-means

A2b. Worked k-means iteration

Invented one-dimensional example: five values $1, 2, 3, 8, 10$ with two starting centres at 1 and 4. Every number below is exact.

Step	Left group (centre)	Right group (centre)	Total spread
round 1, assign	$1, 2$ (at 1)	$3, 8, 10$ (at 4)	54
round 1, update	$1, 2$ (at 1.5)	$3, 8, 10$ (at 7)	26.5
round 2	$1, 2, 3$ (at 2)	$8, 10$ (at 9)	4
round 3	$1, 2, 3$ (at 2)	$8, 10$ (at 9)	4, settled

- **Check one row:** with centres 1.5 and 7, the five squared misses are $0.25 + 0.25 + 16 + 1 + 9 = 26.5$.
- In round 2 the value 3 switches to the left group, the centres move to 2 and 9, and the spread falls to 4. In round 3 nothing moves, so the run has settled.

back to k-means

A3a. Criteria for selecting k

Aid	What it measures	Its limit
Elbow	the within-group spread W_k as k rises	W_k always falls; the bend is a judgement
Silhouette	how well each point sits in its group versus the nearest other group	internal evidence only, in the stated geometry

$$s = \frac{b - a}{\max(a, b)}.$$

a: mean distance to points in its own group · b: mean distance to the nearest other group · s near 1 is well grouped, near 0 is borderline

Invented worked values

- $a = 2,\ b = 8:\ s = 6/8 = 0.75$, comfortably grouped.
- $a = 4,\ b = 5:\ s = 1/5 = 0.20$, borderline.

Our listings (real)

- $k = 3$: 0.498; $k = 4$: 0.470; $k = 5$: 0.427.
- The score mildly prefers three groups; we keep four because the longer-stay segment is commercially meaningful and stable on unseen hosts.

A documented judgement, disagreement reported.

‣ the full elbow table: A3b ‣ back to choosing k

A3b. Within-group spread for $k = 1$ to 10

The within-group spread per listing, refit at every k from 1 to 10:

k	1	2	3	4	5	6	7	8	9	10
W_k per listing	4.00	2.32	1.76	1.34	1.18	1.07	0.98	0.88	0.81	0.76

- The fall never stops: each extra centre can only tighten the groups, so W_k decreases at every k by construction.
- The large steps end at $k = 4$ (drops of 1.68, 0.56, 0.42); from $k = 5$ onwards each extra group buys 0.16 or less.
- The bend is where explanation ends and bookkeeping begins; reading it is a judgement, which is why the choice is documented.

‹ back to choosing k

A4a. Computing PCA loadings and scores

With centred data $\mathbf{X}$ and sample covariance $\mathbf{S}$, the first loading is the unit recipe that carries the most variation:

$$\max_{\|\mathbf{w}\|=1} \mathbf{w}^\top \mathbf{S}\mathbf{w} \quad \Longrightarrow \quad \mathbf{S}\mathbf{w} = \lambda \mathbf{w}.$$

$\mathbf{w}$: the unit-length weights · λ: the variation the recipe carries · loadings are covariance eigenvectors, ordered by λ

- Software reads loadings and scores from the singular value decomposition $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$: loadings are $\mathbf{V}$, scores are $\mathbf{X}\mathbf{V}$.
- Keeping the first r recipes retains a share $\sum_{m \le r} \lambda_m / \sum_j \lambda_j$ of the total variation.
- Multiplying one recipe and its scores by -1 leaves the component unchanged; signs are set only for reading.

The yields work uses covariance PCA on daily changes, so wider-moving maturities carry more weight.

‣ one score worked ‣ back to components

A4b. Orthogonality of the yield-curve loadings

Two recipes are at right angles when their weights, multiplied maturity by maturity and added up, give zero. Check level against slope with the printed weights:

Maturity	3m	6m	1y	2y	3y	5y	7y	10y	30y
level w_j	.14	.18	.26	.37	.40	.42	.41	.38	.31
slope v_j	−.62	−.48	−.36	−.16	−.05	.11	.22	.27	.33

$$\sum_{j=1}^{9} w_j v_j = -0.005 \approx 0 \qquad \text{(exactly 0 before rounding).}$$

- The negative products at the short end cancel the positive ones at the long end, term by term.
- Because the scores share nothing, their variance shares add: $77.9 + 12.5 + 5.0$ sums without double-counting.

back to the two rules

A4c. Worked calculation of a component score

Take the level score of 19 September 2008 apart. The level weights are *not* equal: they are smallest at the short end and largest in the belly.

Maturity	3m	6m	1y	2y	3y	5y	7y	10y	30y
change Δr_j	.76	.75	.52	.38	.37	.34	.29	.24	.22
weight w_j	.14	.18	.26	.37	.40	.42	.41	.38	.31

$$\text{score} = \sum_{j=1}^{9} w_j \, \Delta r_j = 1.08, \qquad \underbrace{0.318 \times 3.87}_{\text{equal-weight guess}} = 1.23.$$

- The equal-weight guess overshoots because that day's *biggest* moves were at the short end (0.76, 0.75), where the level recipe puts its *smallest* weights.
- So the score is a genuine weighted blend, not a plain total: the loadings decide how much each maturity counts.

‹ back to the worked day

A4d. Reconstructing the yield curve on 19 September 2008

Compression runs both ways: keep only the day's three scores ($+1.08$, -0.85, -0.28) and rebuild the nine changes as score $\times$ recipe, summed over the three recipes:

Maturity	3m	6m	1y	2y	3y	5y	7y	10y	30y
actual change	.76	.75	.52	.38	.37	.34	.29	.24	.22
rebuilt from 3	.83	.64	.53	.40	.37	.33	.29	.26	.20

- The largest miss is 0.11 points (at 6 months); the three scores capture 99.1% of the day's squared movement.
- This is what "95.5% of the variation" means in practice: three numbers rebuild the whole curve's move, almost exactly, even on the wildest day in the panel.

‹ back to the worked day

A4e. Rules for selecting the number of components

Standard rules pick the number of components from the **eigenvalues**:

Rule	Statement	On our data
Kaiser criterion	keep components whose correlation-matrix eigenvalue exceeds 1 (ours: 6.55, 1.49, then 0.49); covariance version: above the average eigenvalue	keep 2
cumulative share	keep enough for a declared threshold: two reach 90.5%, three reach 95.5%	2 or 3
scree elbow	keep the components before the curve flattens, read by eye	1 to 3

- The rules disagree: two is fully defensible here. We keep three because curvature is a named direction desks actively hedge, and it is stable across both eras.
- As with choosing k (A3a): rules narrow the range; the final call is a documented judgement.

back to variance explained

A5. Incompatible group-fairness criteria

An invented, exact example: two groups, different base rates, one score equally precise and equally complete in both:

	Group A	Group B
will actually reoffend (of 100)	50	10
flagged by the score	50	10
flagged and correct	40	8
precision (correct among flagged)	$40/50 = 0.80$	$8/10 = 0.80$
recall (caught among actual)	$40/50 = 0.80$	$8/10 = 0.80$
false positive rate	$10/50 = \mathbf{0.20}$	$2/90 = \mathbf{0.02}$

- Same precision and recall, yet group A's innocents are flagged *nine times* as often: only the base rates differ.
- This is general (Kleinberg et al. 2016; Chouldechova 2017): with different base rates, equal calibration and equal false positive rates cannot both hold. Fairness needs an explicit choice.

‹ back to the COMPAS case

A6. EU AI Act risk categories

Regulation (EU) 2024/1689 sorts AI systems by risk, with duties rising up the ladder:

Tier	Examples	What the law requires
prohibited	social scoring by public authorities; exploitative manipulation	banned outright
high risk	creditworthiness, employment decisions, access to essential public services (Annex III), among other categories	documented data and design, human oversight, accuracy and monitoring duties
limited risk	chatbots, generated content	transparency: people must know they are dealing with an AI
minimal risk	most other uses	no new duties

- The high-risk duties mirror Part IV's safeguards almost line by line: documentation, group testing, a human path, monitoring.

back to the safeguards