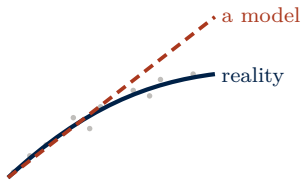
AI and Machine Learning
Foundations

From a question to a prediction you can trust



Fatih Kansoy

Day 1 of 4

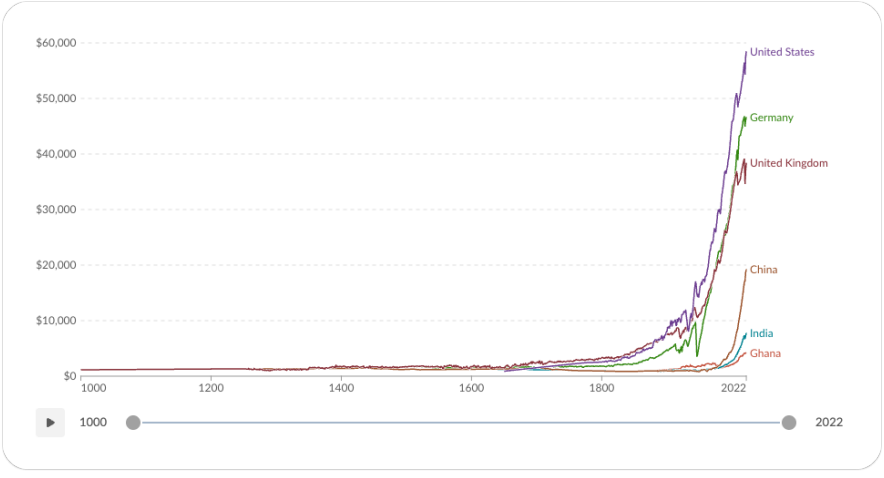
Worcester College, University of Oxford

Outline

1. Artificial intelligence: history and context
2. Business applications of AI
3. Formulating and fitting prediction models
4. Model validation and generalisation
5. Flexible supervised models
6. Unsupervised learning

Part I · Artificial intelligence: history and context

What happened and what is happening?



Source: Our World in Data.

Definition and scope of artificial intelligence

> **Artificial intelligence**: building systems that perform tasks which normally require human intelligence, such as perceiving, reasoning, deciding, or producing language.

- **Narrow AI** performs one task envelope well: translation, chess, credit scoring. Every deployed system today is narrow.
- **General AI** would mean competence across arbitrary tasks. No such system exists, and it is not this week's subject.
- The definition says nothing about method. The methods that now deliver these capabilities are **learned from data**.

When a headline says "AI", ask first: which task, and how is the system judged on it?

Examples of AI in everyday life

Face unlock
recognises your face and unlocks the phone

Maps ETA
predicts when you will arrive

Spam filter
keeps junk out of your inbox

What to watch
suggests the next title for you

Fraud alert
flags an unusual payment

Autocomplete
guesses the next word

It is already running your day, quietly.

AI tasks and prediction output types

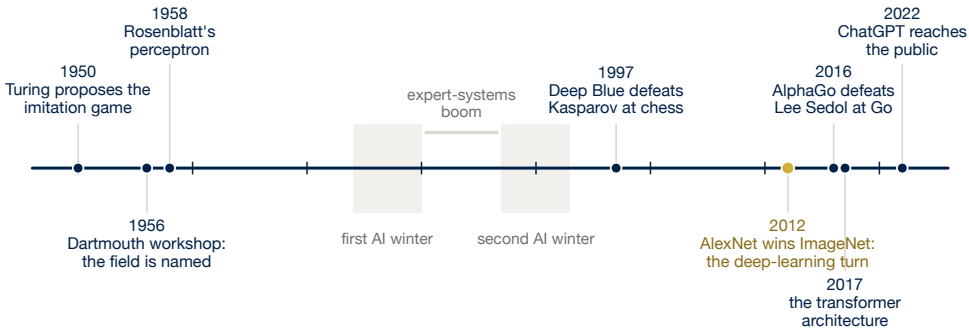
System	Its task, stated as a question	Output type
face unlock	is this the owner's face?	a class
route planner	when will you arrive?	a number
spam filter	is this message junk?	a class
recommendations	which title should come next?	a ranking
fraud alert	how likely is this payment fraudulent?	a probability
autocomplete	which word follows?	a probability

Each system runs quietly, and each is a prediction with one of four output shapes: a **number**, a **probability**, a **class**, a **ranking**.

Keep the four output types in mind. Part III shows how a numerical prediction target is defined and fitted.

History of artificial intelligence

- In 1950 Alan Turing replaced “can machines think?” with a test of written answers; in 1956 a Dartmouth summer workshop named the field **artificial intelligence**.
- Twice, promises outran results and funding collapsed: the **AI winters**.
- The learning era after 2012 is the part this course examines.



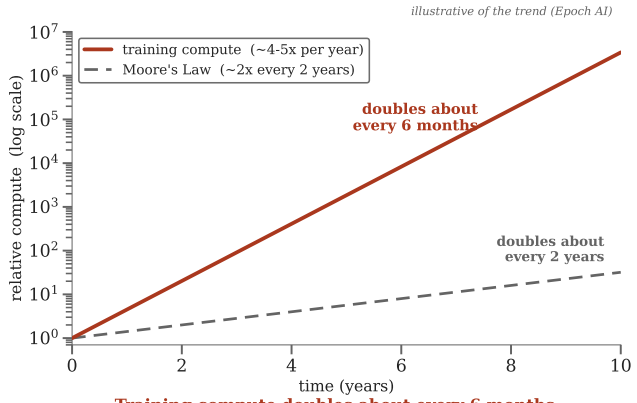
Milestone dates are public record; full citations are in the References.

Drivers of recent AI progress

Data
almost everything is now logged

Compute
training power doubles about every 6 months

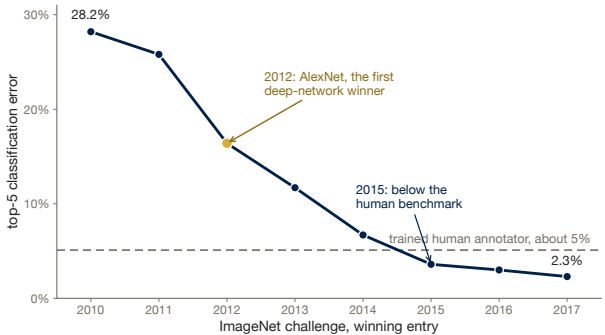
Algorithms
deep learning that scales with data



Epoch AI: compute used per notable training run grew about 4–5x a year. Over a decade, its growth factor was roughly 100,000 times larger than Moore's Law alone (chip density doubling about every two years): the difference came mainly from much larger, costlier training runs, not just better chips. More compute enables progress; it does not guarantee it.

ImageNet classification performance

ImageNet: over a million labelled photographs in a thousand categories, with an annual classification competition from 2010. In 2012 the first deep-network winner cut the error from 25.8% to 16.4% in a single year.



ILSVRC winning top-5 error rates, 2010–2017 (Russakovsky et al., 2015); AlexNet (Krizhevsky, Sutskever, and Hinton, 2012).

AI adoption, use, and investment

72%

of surveyed firms use AI in at least one function

up from 55% a year earlier

~800M

weekly ChatGPT users

reportedly ~100M within two months of launch

$252bn

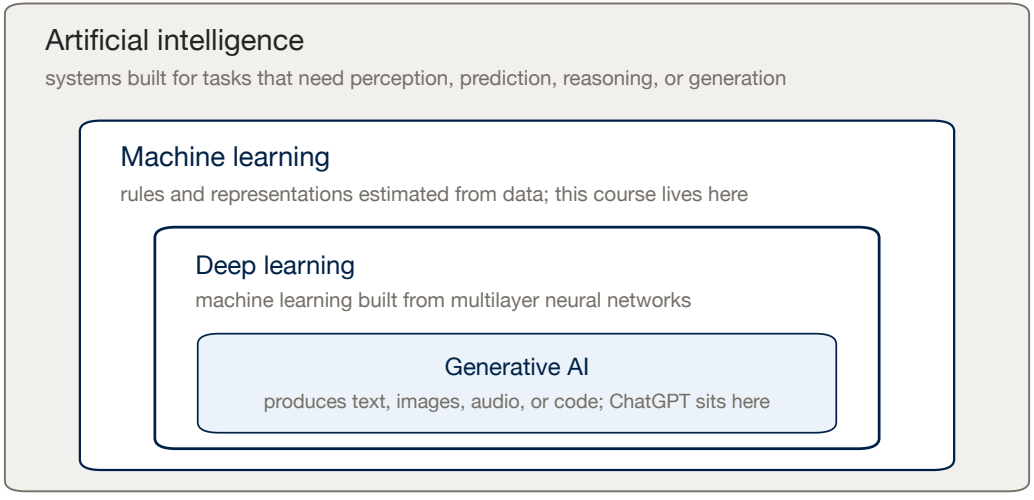
estimated corporate AI investment

in a single year

Source: McKinsey, State of AI 2024 (surveyed firms); OpenAI (Oct 2025); ~100M is a UBS/Similarweb estimate, not an OpenAI figure; Stanford AI Index 2025.

This is not a pilot. It is the operating system of modern business.

AI, machine learning, deep learning, and generative AI



Machine learning settings

Setting	What is given	What is learned	This week
supervised	inputs with outcomes attached	a rule from inputs to outcome	Days 1–3
unsupervised	inputs only	structure: groups, directions	Day 4
reinforcement	actions and their rewards	a policy of action	not covered

Generative assistants are trained in stages. **Self-supervised** pretraining builds targets from the data themselves, for example the next word of a sentence; supervision and preference feedback then tune the result.

The bank task ahead is supervised: the file arrives with the outcomes attached.

Part II · Business applications of AI

Machine learning applications by business function

Marketing
who will leave,
and who to target

Finance
fraud, and
credit risk

Operations
demand forecasts,
and maintenance

Customer
recommendations,
support triage

HR
screening,
handle with care

Risk
early warning
of trouble

This is not one department's tool. It is everyone's.

Netflix recommendation systems

Netflix's own engineers report that recommendations influence about **80%** of the hours watched, and estimate that personalisation saves **more than $1bn a year**, mostly through customers who stay.

Note who is doing the reporting. These are the company's own figures, and no outside party has audited them.

Source: Gomez-Uribe & Hunt, ACM TMIS 2016; Netflix's own ∼2015 estimate.

∼80%

of hours streamed are shaped by what the model recommends

A good recommender is not a feature. It is the business.

Real-time payment fraud detection

Global payment-card fraud losses reached about **$33.4bn** in 2024. Every card payment is scored for risk in **real time**, far faster than any human review could manage.

This is machine learning you never see, running on every card payment, everywhere, all day.

Source: The Nilson Report, card-fraud losses worldwide, 2024 (card fraud specifically).

$33.4bn

global card-fraud losses in a single year, 2024

At this speed and scale, the model is the only thing fast enough.

Email spam, phishing, and malware filtering

Google states that Gmail blocks **more than 99.9%** of spam, phishing and malware, filtering on the order of **15 billion** emails a day.

You never notice the ones it stops. That is the point.

Source: Google Safety Center (a vendor self-report, not an independent audit).

>99.9%

of spam, phishing and malware blocked before you see it

The best ML often shows up as nothing going wrong.

Generative AI use and coding productivity

The newest wave writes and codes. ChatGPT reports about **800 million** weekly users. In **one controlled study**, about 95 developers doing *a single* coding task finished **55% faster** with an AI assistant.

A striking result, but one narrow task run by the vendor. The same rules of evidence still apply.

Source: OpenAI (Oct 2025); GitHub Copilot study, 2022 (∼95 developers, one greenfield task, wide confidence interval).

55%

faster on *one* coding task, in a single vendor study

The newest wave, and the same discipline applies.

Estimated economic impact of generative AI

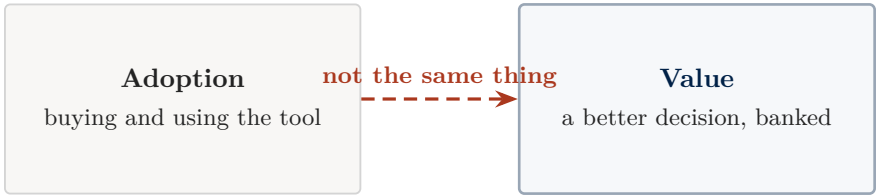
$2.6–4.4 trillion

a year that generative AI *could* add to the world economy, across dozens of use cases: a **projection**, not a measured figure

Source: McKinsey Global Institute, "The economic potential of generative AI," 2023. A modelled estimate, sensitive to adoption; other houses give different numbers.

A real opportunity, and an estimate, not a promise.

AI adoption and realised business value



many pilots never reach production

The winners are disciplined, not just enthusiastic.

Part III · Formulating and fitting prediction models

Conditional mean as a prediction target

For a new case, the features $X = \mathbf{x}$ are known but the outcome Y is not. The prediction target is

$$m(\mathbf{x}) = \mathbb{E}[Y \mid X = \mathbf{x}].$$

- $\mathbf{x}$ describes the case at the time of prediction.
- $m(\mathbf{x})$ is the average outcome among comparable cases.
- This population relationship is unknown; the sample is used to estimate it.

A fitted model produces $\widehat{m}(\mathbf{x})$, an estimate of the unknown target $m(\mathbf{x})$.

Sample data and the design matrix

With p measured features, include the intercept by writing

$$\mathbf{x}_i^\top = (1, x_{i1}, x_{i2}, \ldots, x_{ip}) \in \mathbb{R}^{1\times(p+1)}.$$

Stack the n observed rows and outcomes:

$$\underbrace{\mathbf{X}}_{n\times(p+1)} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1p} \\ 1 & x_{21} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \cdots & x_{np} \end{bmatrix}, \qquad \underbrace{\mathbf{y}}_{n\times 1} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}.$$

- Row i is one observation; column j is one feature.
- The first column carries the intercept β_0.
- The same p features must be constructed for a new $\mathbf{x}_{\text{new}}$.

Linear prediction rule

A linear rule approximates the unknown target using one common set of weights:

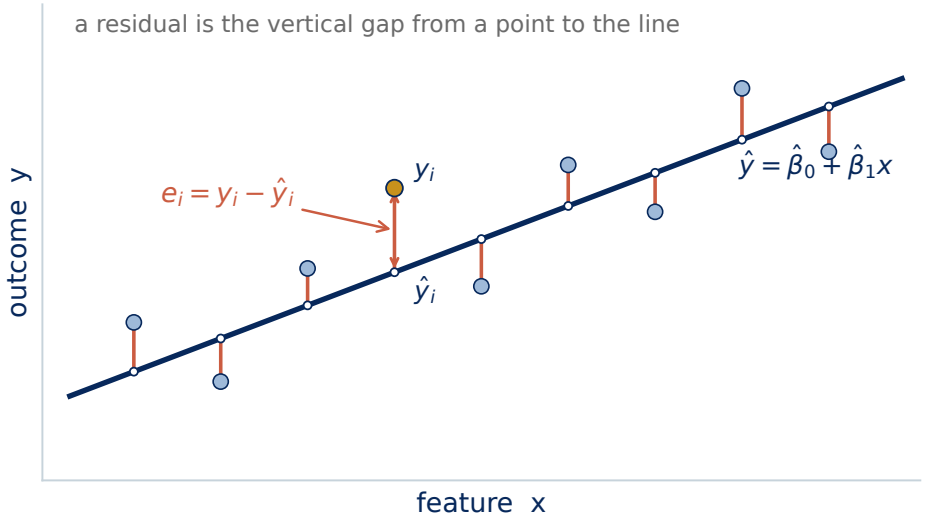
$$\widehat{y}_i = \underbrace{\widehat{\beta}_0}_{\text{baseline}} + \underbrace{\widehat{\beta}_1 x_{i1} + \cdots + \widehat{\beta}_p x_{ip}}_{\text{feature contributions}} = \mathbf{x}_i^\top \widehat{\boldsymbol{\beta}}.$$

- x_{ij} is a recorded feature for observation i.
- The coefficients $\widehat{\beta}_j$ are learned jointly from the training observations.
- The fitted rule combines the feature contributions into one prediction.

The same rule is applied to a new feature vector:

$$\widehat{y}_{\text{new}} = \mathbf{x}_{\text{new}}^\top \widehat{\boldsymbol{\beta}}.$$

Observed outcomes, predictions, and residuals



Ordinary least squares estimation

For observation i, the residual is the observed outcome minus its prediction: $e_i = y_i - \widehat{y}_i$.
Ordinary least squares chooses the coefficient values that give the smallest mean squared residual on the training observations:

$$\text{MSE}_{\text{train}} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2 = \frac{1}{n}\sum_{i=1}^{n} e_i^2.$$

- Squaring stops positive and negative residuals from cancelling.
- A residual of 4 counts more heavily than a residual of 2.

Low training MSE means a close fit to these observations. Part IV asks whether the fitted pattern survives on new observations.

Probability prediction for binary outcomes

Code the outcome as

$$Y = \begin{cases} 1, & \text{the event occurs,} \\ 0, & \text{the event does not occur.} \end{cases}$$

For a binary outcome, the average among comparable cases is their event rate:

$$p(\mathbf{x}) = \mathbb{E}[Y \mid X = \mathbf{x}] = \Pr(Y = 1 \mid X = \mathbf{x}).$$

$$\frac{3}{10} = 0.30$$

A prediction $\widehat{p}(\mathbf{x}) = 0.30$ means that about 30% of comparable cases are expected to have $Y = 1$. An individual realised outcome is still either 0 or 1.

Logistic regression probability model

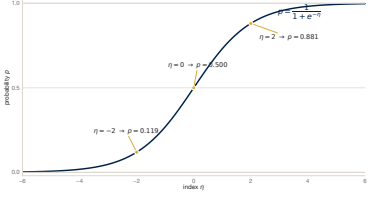
- **Calculate a raw score:**

$$z = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p.$$

  Each x_j is a feature value and each β_j is a weight learned from training data. The score z adds their contributions; it can be any real number and is not yet a probability.

- **Convert the score into a probability:**

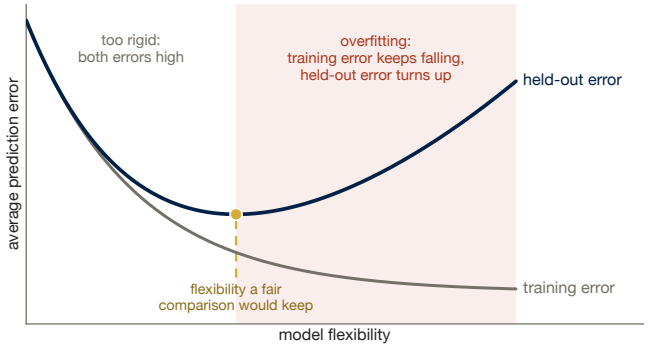
$$\widehat{p}(\mathbf{x}) = \frac{1}{1 + \exp(-z)}.$$



Part IV · Model validation and generalisation

Generalisation and overfitting

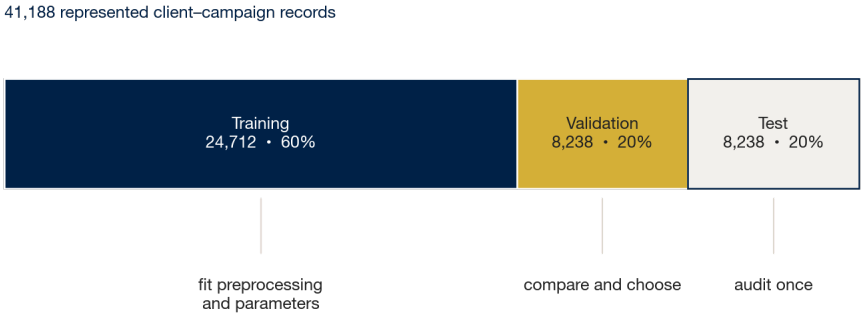
Training observations shaped the rule. A low training loss is necessary, but it is not evidence that the learned pattern will travel.



The job is not to remember the past; it is to perform on the next case.

Training, validation, and test samples

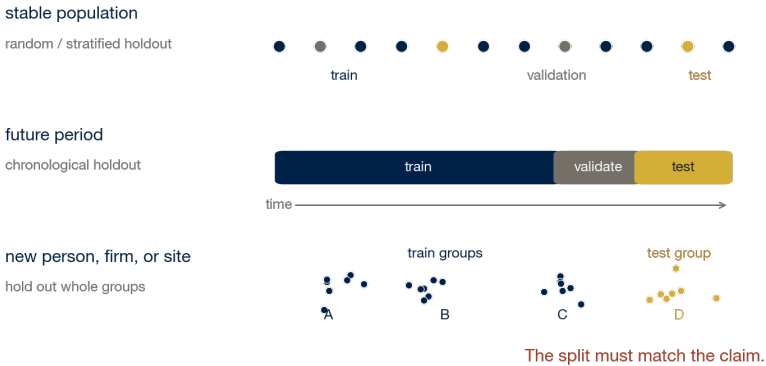
Training data fit candidate workflows; validation data choose among them; the test set audits the frozen choice.



Validation chooses; the test set gives one final audit of the frozen procedure.

Data splitting for deployment settings

A split answers a specific deployment claim: comparable new records, future records, or entirely new people and organisations.



The evaluation design is part of the claim, not an afterthought.

Choosing a holdout design

Think of the holdout as a rehearsal for the real prediction task. “New” can mean different things:

- **Another similar customer** $\longrightarrow$ hide random customers.
- **Customers next year** $\longrightarrow$ hide the latest time period.
- **Customers at a new bank branch** $\longrightarrow$ hide an entire branch.

If the task is to predict for a new branch, but customers from that branch appear in training, the test is too easy and its result is unrealistic.

What will be new at deployment: records, time, or groups?

Testing is a rehearsal for deployment, so the rehearsal must resemble the real event.

Bank marketing prediction task

Each row is one historical client-campaign contact. The task is to predict whether the client subscribes to a term deposit, $y \in \{\text{no}, \text{yes}\}$, using information available before a new call.

Contract item	Teaching data
Source	UCI Bank Marketing, `bank-additional-full.csv`
Unit and scope	41,188 client-campaign records; May 2008–November 2010
Inputs	20 recorded inputs: client, contact, campaign-history, and economic context
Target	y: subscription after the contact
Prediction moment	before placing a prospective call

UCI Bank Marketing; Moro, Cortez, and Rita (2014).

Bank marketing dataset: example records

A simplified view of three completed historical marketing calls:

Contact	Age	Job	Previous outcome	**Duration**	**Outcome y**
A	31	admin	no previous call	96 sec	no
B	47	technician	success	420 sec	yes
C	55	retired	failure	185 sec	no

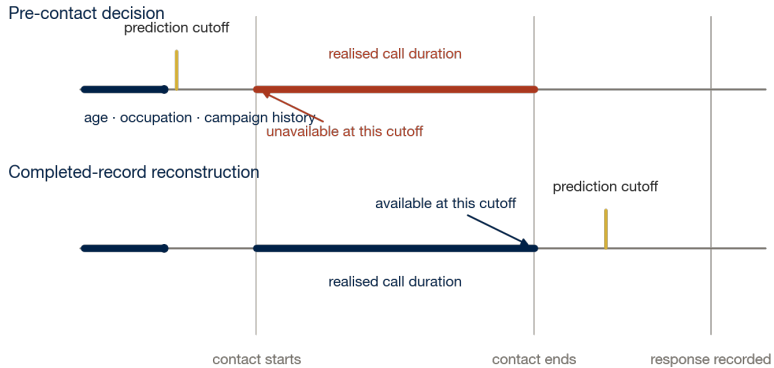
Features (X) are the columns offered to the model, such as age, job, previous outcome, and call duration.

The **target (y)** is the answer we want to predict: did the customer subscribe?

Timing check: before a new call begins, which of these feature values can the model actually know?

Feature availability at prediction time

The bank file contains `duration`: how long the completed call lasted. It is strongly informative after the fact, but is not known when deciding whom to call. UCI explicitly warns against using it for a realistic model.



The valid feature set is determined by the prediction moment, not by the score it produces.

Validation prediction procedure

- **Data and split.** The 41,188 historical contacts were divided into 24,712 training, 8,238 validation, and 8,238 test contacts, while keeping the subscription rate similar in each part.
- **Training.** Preprocessing and model coefficients were fitted only on the training contacts. Logistic regression was used as a simple probability model for the yes/no target.
- **Inputs.** Only valid pre-call features were used; `duration` was excluded.
- **Validation.** For each validation contact, the model produced $\widehat{p}$. Values at least 0.50 were labelled yes; the rest were labelled no and compared with the recorded outcome.
- **Test.** The 8,238 test contacts remained untouched for the final audit.

Validation outcomes by actual and predicted class

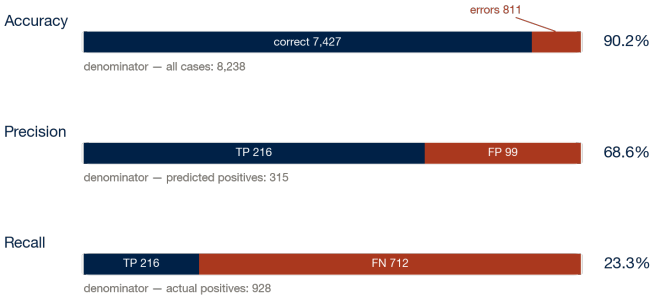
	Predicted yes	Predicted no	Actual total
Actual yes: subscribed	**216 correct**	712 missed	928
Actual no: did not subscribe	99 false yes	**7,211 correct**	7,310
Predicted total	315	7,923	8,238

- The 216 contacts subscribed and were correctly predicted yes.
- The 7,211 contacts did not subscribe and were correctly predicted no.
- The other $99 + 712 = 811$ predictions were wrong.

$$\text{accuracy} = \frac{216 + 7{,}211}{8{,}238} = \frac{7{,}427}{8{,}238} = 90.16\%.$$

Classification outcomes and evaluation metrics

At threshold 0.50, the frozen validation predictions give $TP = 216$, $FP = 99$, $FN = 712$, and $TN = 7{,}211$.



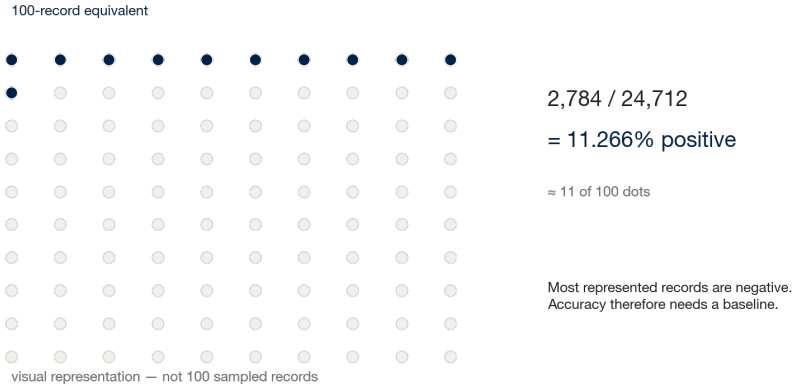
$$\text{accuracy} = 90.16\%, \qquad \text{precision} = 68.57\%, \qquad \text{recall} = 23.28\%.$$

Same frozen validation ledger; classification threshold 0.50.

The denominator decides the question: correct overall, credible alarms, or cases found.

Majority-class accuracy baseline

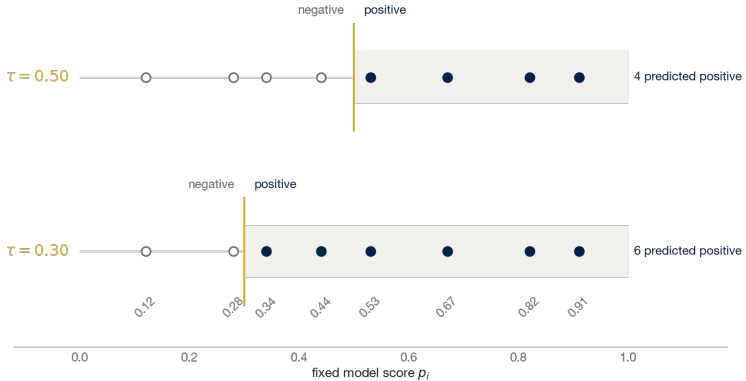
With `duration` removed and validation observations held out, begin with the obvious rule: predict "no subscription" for everyone. It already achieves $7{,}310/8{,}238 = 88.74\%$ accuracy and finds no subscribers.



Frozen bank ledger: stratified 60/20/20 split (seeds 42/43); validation $n = 8{,}238$; `duration` excluded; train-only scaling and encoding; unpenalised logistic regression.

Classification threshold selection

The probabilities do not change when the classification threshold changes. What changes is the balance between misses and false alarms.



Choose the threshold on validation data using the decision's costs, capacity, and harms; then freeze it before the final test.

Prediction versus causal effects

After validation selects the workflow, one untouched test set audits it. That audit supports a narrow claim: how well the frozen procedure predicts response for records like those observed.

Prediction: Who is likely to subscribe among historical contacts represented in this file?

Causal effect: Who subscribes because a call is made rather than not made? This needs a comparable no-call strategy.

The model can rank likely responders; it cannot infer the benefit of calling from contacts alone.

Part V · Flexible supervised models

Why move beyond one global rule?

Regression is a valuable benchmark: it applies one consistent rule across all cases. Business behaviour, however, often depends on cutoffs and combinations.

One global rule
Each feature has a stable relationship with the outcome.

Useful when clarity and consistency matter most.

Flexible rules
"Channel matters more for new customers" or "risk changes after a threshold."

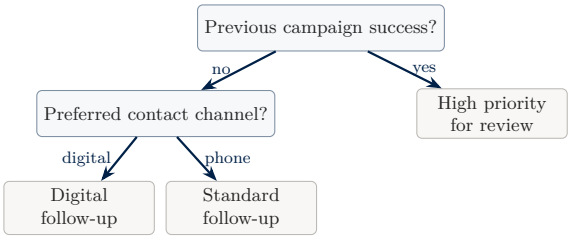
Useful when the same action works differently across cases.

Can it capture the real pattern? → **Can people explain** the decision? → **Does it still work** on unseen cases?

Flexibility is valuable only when it improves a decision outside the training data.

A decision tree is a sequence of business questions

A tree sends each case through a short series of if–then questions. The final group receives a predicted outcome or recommended next step.

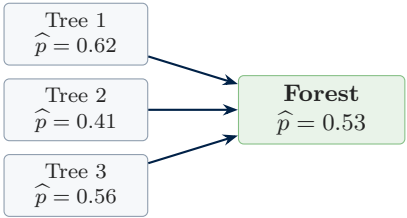


- **What it does:** creates understandable case groups.
- **Where used:** customer triage, preliminary risk review, and service routing.
- **Watch-out:** one tree can change when the sample changes.

The value is not the drawing. It is a visible decision path that managers can question, test, and revise.

A random forest is a committee of trees

A decision tree follows one if–then path. A random forest averages predictions from many deliberately different trees.



1. Train varied trees using resampled rows and random feature subsets.
2. Send the same new case through all the trees.
3. Average probabilities, or take a majority vote.

Different errors partly cancel, so the combined result is usually more stable. Validation is still required.

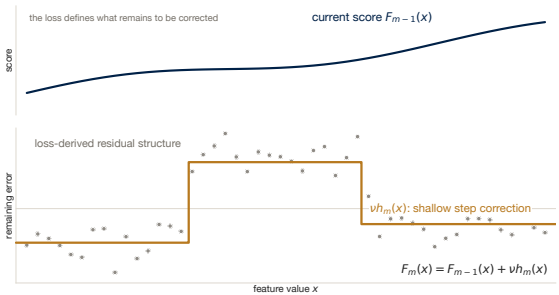
Historical application: Microsoft Kinect research (2011)

- **Input:** a depth camera measures the distance of each pixel.
- **Trees:** pixels follow many trees and vote for body parts or possible three-dimensional joint locations.
- **Output:** the combined votes estimate body pose for real-time interaction; the research system reported about 200 frames per second.

Shotton et al., CVPR (2011); Girshick et al., ICCV (2011). The reported speed describes the historical research system.

Gradient-boosted decision trees (GBDT)

GBDT adds shallow trees sequentially to reduce the error left by the current model.



Training sequence

1. Start with a simple initial prediction.
2. Identify the remaining prediction errors.
3. Fit a small tree to part of those errors.
4. Add a cautious correction and repeat.

Learning rate η: controls how much influence each new correction receives.

Schematic illustration: the next shallow tree targets structured error remaining from the current prediction.

Decision tree, random forest, and GBDT

All three models use decision trees. They differ in how the trees are trained and combined.

	Decision tree	Random forest	GBDT
Structure	one tree	many trees trained independently	many shallow trees trained in sequence
Combination	no combination	average or majority vote	add a small correction at each stage
Main strength	clear decision path	stable, flexible prediction	strong prediction on structured data
Main caution	can be unstable	less transparent than one tree	requires careful tuning and monitoring

Practical distinction: a random forest reduces instability by averaging; GBDT improves the current fit by correcting errors sequentially.

Application: Facebook notification ranking

Facebook used boosted trees to rank notifications under a strict response-time limit.

Prediction task

- **Case:** a candidate notification for a person.
- **Target:** whether that person clicks.
- **Inputs:** prior notification clicks and story engagement.

Operational use

1. Estimate click probability for each candidate.
2. Rank or filter the candidates using those probabilities.
3. Return the result within the live product's latency budget.

Documented system scale (2017)

- 256 trees with 32 leaves per tree; roughly 1,000 candidates evaluated in an average batch.
- Evaluation was made up to five times faster than the flat compiled baseline, allowing more candidates or a larger model with the same computing budget.

Management implication: relevance, response time, and computing cost must be managed together.

Meta Engineering (2017), "Evaluating boosted decision trees for billions of users." Historical system.

Choose by business need, not model fashion

No model is automatically “best.” Compare candidates on the same unseen cases, then consider the operating environment.

Business need	Sensible starting point	Main trade-off
Transparent benchmark	Logistic regression	clear and stable, but may miss complex patterns
Explainable decision path	Decision tree	easy to follow, but can be unstable
Stable flexible prediction	Random forest	strong general-purpose option, but less transparent
Strong structured-data prediction	Gradient boosting	often powerful, but needs more tuning and monitoring

prediction | explanation | speed | maintenance | risk

The winning model is the one that improves the decision and can be operated responsibly.

Part VI · Unsupervised learning

Supervised and unsupervised learning

Part V assumed that each training case had a target y. Part VI removes that target, so the task changes from predicting an outcome to identifying structure in the measured features X.

Element	Supervised learning	Unsupervised learning
Available data	features X and a known target y	features X, but no target variable
Main question	What outcome should we predict for a new case?	What groups, shared patterns, or unusual cases exist?
Typical output	predicted value, class, or probability	clusters, principal components, or anomaly scores
Evaluation	performance on unseen cases with known outcomes	stability, interpretation, and usefulness in a later decision

Advantages
Uses data without costly labels; simplifies complex data; generates hypotheses and exploratory segments.

Limitations
Results depend on features, scaling, and method; there is no single accuracy score; discovered patterns are not causal conclusions.

Common applications: customer segmentation, grouping related documents, detecting unusual transactions, and summarising correlated risk indicators.

Unsupervised learning without a target variable

Suppose a retailer has customer behaviour data but no outcome such as “will subscribe” or “will churn.”

Customer	Visits in 90 days	Average basket	Days since purchase	Digital share
A	12	£38	5	90%
B	11	£41	8	85%
C	2	£160	20	10%
D	1	£25	120	30%

Clustering groups rows
Which customers have similar behaviour?

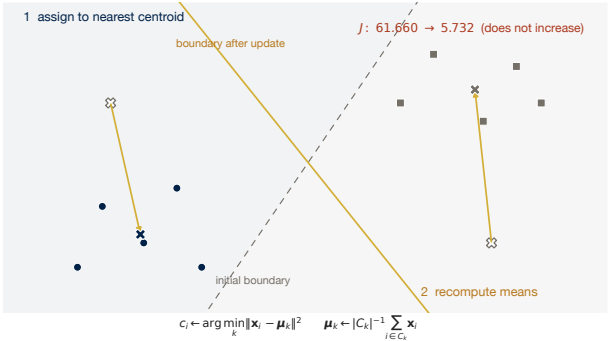
PCA summarises columns
Which measures contain similar information?

Each row is a customer and each measured column is a feature. There is no target variable supplying a correct answer.

Hypothetical customer data for illustration.

K-means clustering

K-means separates observations into k groups by repeatedly updating representative group centres.



Schematic two-dimensional example. The chosen features and distance measure define what counts as similar.

Training sequence

1. Choose k and initial centres.
2. Assign each observation to its nearest centre.
3. Recalculate the centres and repeat until assignments settle.

Output

A cluster label for each observation and a centre describing each cluster.

Application: customer segmentation

Applied to the customer features, clustering might produce the following descriptive groups.

Analyst's label	Observed pattern	Business question to test
Frequent digital	frequent and recent purchases; high digital share	Would personalised app offers improve conversion?
High-value occasional	fewer visits; larger average basket	Would service or retention benefits increase frequency?
Inactive	few visits; long time since last purchase	Is a low-cost reactivation campaign worthwhile?

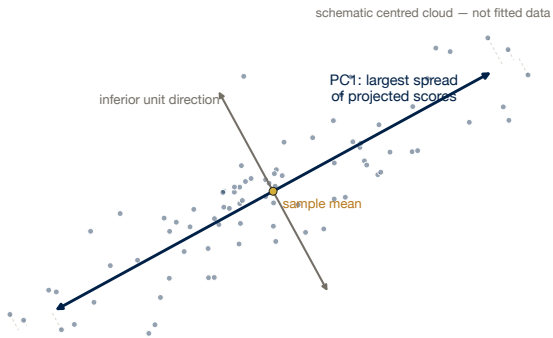
What the algorithm supplies
Group assignments and numerical group centres.

What managers and analysts add
Segment names, commercial interpretation, and controlled tests of the proposed actions.

Clusters describe patterns in the selected data; they are not permanent customer types and do not prove which action will work.

Hypothetical segments for illustration.

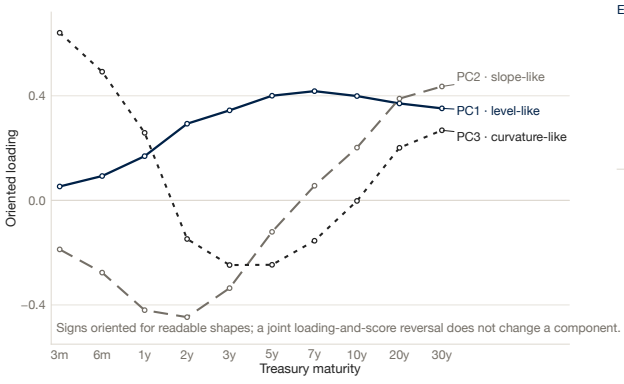
Principal component analysis (PCA)



Each dot is one observation. PCA rotates the view: **PC1 follows the greatest spread** and becomes one summary score; PC2 captures what remains.

Schematic centred data; not fitted data.

Application: interest-rate risk



Ten rates become three movements—**level, slope, and curvature**—retaining 95.7% of observed variation for interest-rate risk monitoring and stress tests.

Frozen FRED sample: 4,130 daily changes. Interpretation follows Litterman and Scheinkman (1991).

Part VII · Responsible AI and governance

Case study: gender bias in recruitment

Amazon developed an experimental machine-learning tool to rank technical-job applications from one to five stars.

Intended purpose	Identify promising applicants more quickly.
What went wrong	The training applications covered ten years and came mostly from men. The system learned to favour some male-associated patterns and penalised references to women's activities and graduates of two women's colleges.
Response	Engineers tried removing specific terms, but other proxy signals could remain. Amazon abandoned the project; recruiters had not relied solely on its rankings.

Management lesson: historical data can reproduce past inequality. Removing gender from the data does not necessarily remove gender-related signals, so outcomes must be tested by relevant groups.

Reuters, Jeffrey Dastin (2018), "Amazon scraps secret AI recruiting tool that showed bias against women."

Case study: racial bias in healthcare

A widely used US healthcare algorithm helped decide which patients should receive additional care.

Intended purpose	Identify patients with the greatest health needs.
What the model predicted	Future healthcare cost—a convenient but imperfect proxy for health need.
What went wrong	Unequal access meant less was spent on Black patients with the same level of illness. At the same risk score, Black patients were therefore considerably sicker than White patients.

Correcting the disparity was estimated to increase the share of Black patients receiving additional help from 17.7% to 46.5%.

Management lesson: a model can predict its chosen target accurately and still support the wrong decision. Bias can enter through the target even when race is not an input.

Obermeyer, Powers, Vogeli, and Mullainathan, *Science* 366 (2019), 447–453.

Case study: privacy and data security in an AI service

In March 2023, ChatGPT was taken offline after a bug in an open-source Redis client affected separation between users.

Information exposed	Some users could see titles from another active user's chat history.
Payment information	Payment-related information for 1.2% of Plus subscribers active during a nine-hour window may have been visible. Full card numbers were not exposed.
Response	OpenAI took the service offline, patched the bug, notified affected users, and added further checks and logging.

Management lesson: the model was not the failure. AI governance must cover the whole service, including third-party software, access control, monitoring, data minimisation, and incident response.

OpenAI (2023), "March 20 ChatGPT outage: Here's what happened."