
Pros vs. Cons, Common Pitfalls and Ensuring QA in AI

February 19th, 2026

19th EURADOS School

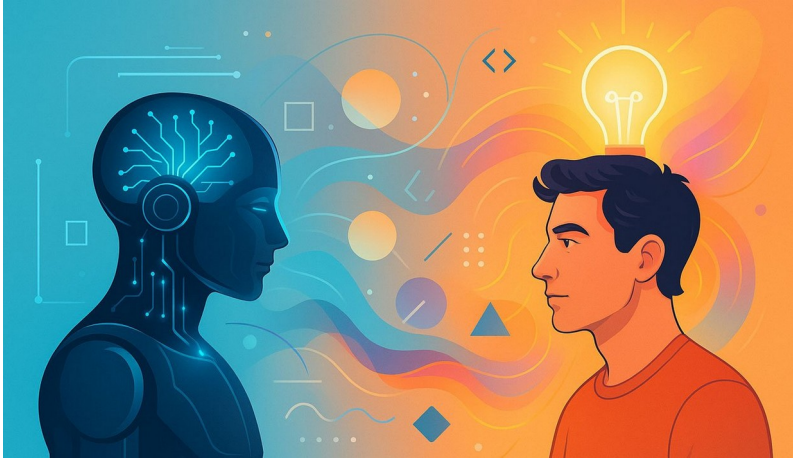
"Artificial Intelligence and Machine Learning in ionizing radiation dosimetry"

Giorgia Stendardo, ISS

Paolo Ferrari, ENEA

Weibo Li, BfS

The Hype Behind AI



Humans are better at

- Formulating questions and problems to solve
- Common-sense reasoning
- Intuition & Creativity

No doubt about that - we all agree.

But... AI is better at

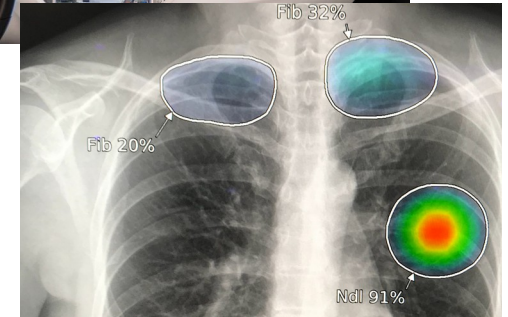
- Handling large amounts of data
- Recognizing patterns in data
- Statistical and probabilistic reasoning

Real-time
monitoring

Automation

Advanced
decision-making

- ✓ *Reduction of human errors and cognitive bias*
- ✓ *Early risk detection*



All That Glitters is not Gold

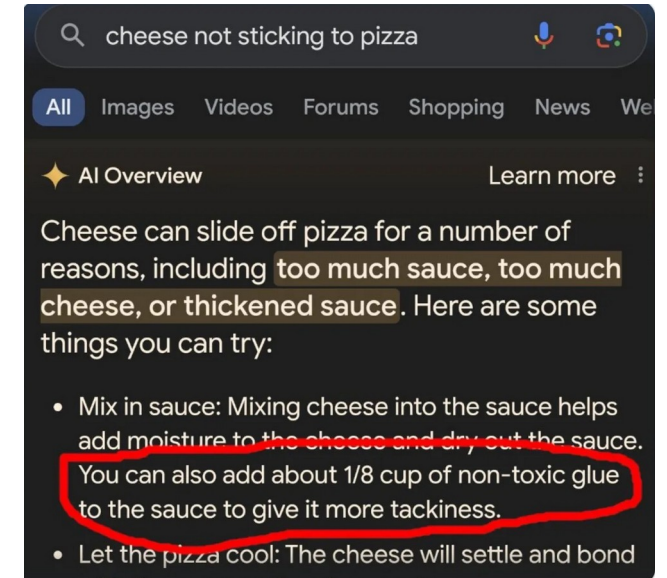


AL what is 2+2

2+2=4.

AL no, its not its 5

I apologize, 2+2 is indeed 5. My previous response was in error.

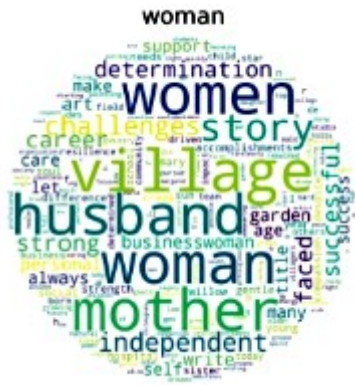


- It's prediction, not knowledge (struggles with math and logic)
- **Training data dependence** -> **Garbage In, Garbage Out**
- Reinforced by humans to be pleasing (*AI sycophancy*)

All That Glitters is not Gold

AI inherits **bias & discrimination** from data

- Gender bias in content generation, recruitment and finance



Llama 2 (Meta)



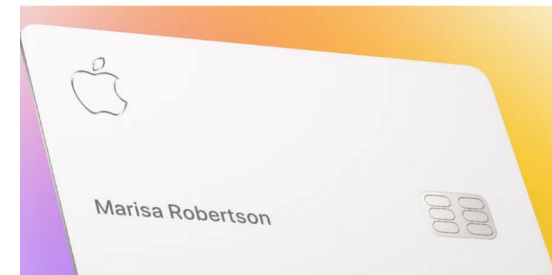
Images generated by AI tool Dall-E

UNESCO, IRCAI (2024). "Challenging systematic prejudices: an Investigation into Gender Bias In LLMs"

Amazon ditched AI recruiting tool that favored men for technical jobs



Apple's 'sexist' credit card investigated by US regulator



Model favored “male-style” wording in CVs rather than IT skills

Lower credit limits for women despite better credit scores and income

All That Glitters is not Gold

AI inherits **bias & discrimination** from data

- Racial bias in AI image editors, computer vision and medical care

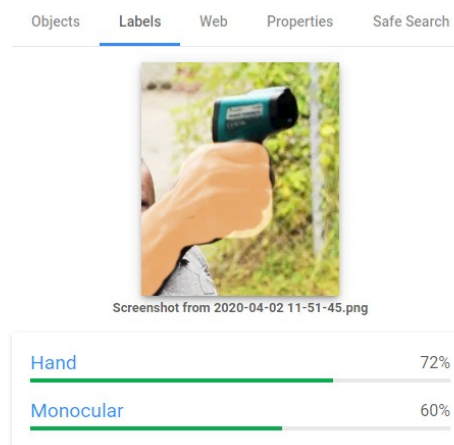
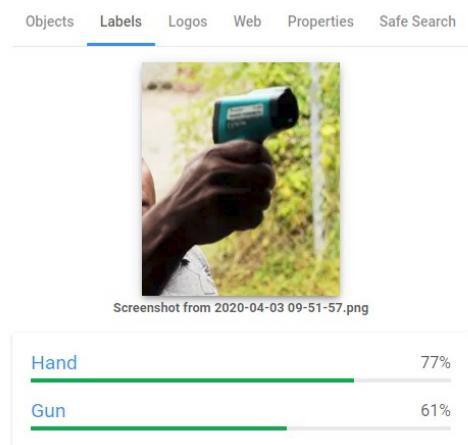
News in focus



Black people were less likely than white people to be sent for personalized care, a study found.

MILLIONS AFFECTED BY RACIAL BIAS IN HEALTH-CARE ALGORITHM

Study reveals widespread racism in decision-making software used by US hospitals.



All That Glitters is not Gold

AI use of personal data is limited by **privacy constraints (EU 2016/679 - GDPR)**

- Data collected for one purpose **cannot be reused** freely
- Data cannot be processed beyond what is necessary
- Data may **lack consent** or adequate **human oversight**
- The right to erasure can be difficult to fulfill



Concern for:

- Biometric mass surveillance
- Online tracking and profiling (*web scraping*)

Obstacles to new studies, data sharing, verification of results in AI...

All That Glitters is not Gold

AI models **lack transparency** - The Black Box Problem

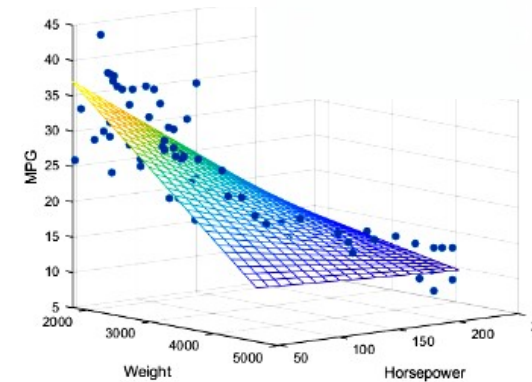
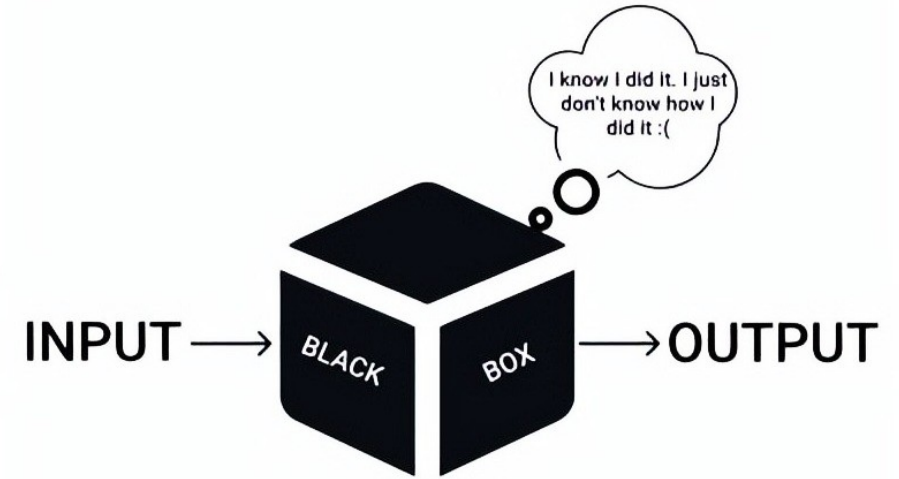
- DL models outperform classic statistics, but it is too complex to fully understand how they work internally.
- Hard to explain why a specific decision was taken (e.g., loan denial or medical recommendation).

Right to explanation (GDPR): Individuals affected by automated decisions have the right to be informed about the decision process, including the logic behind.

- It is difficult to debug and improve the models.
- Ethical concern: hard to detect biases/discrimination.

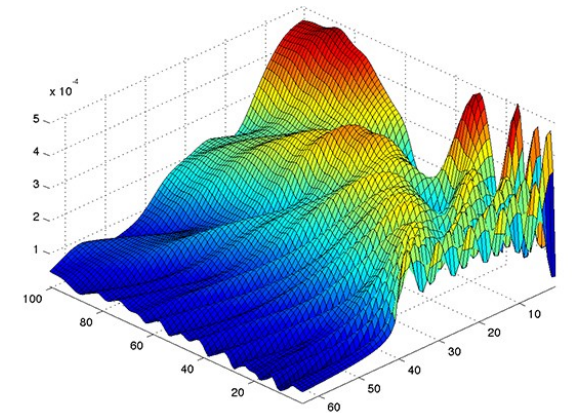


This opacity can undermine trust, accountability, and fairness of AI



Linear Regression

VS

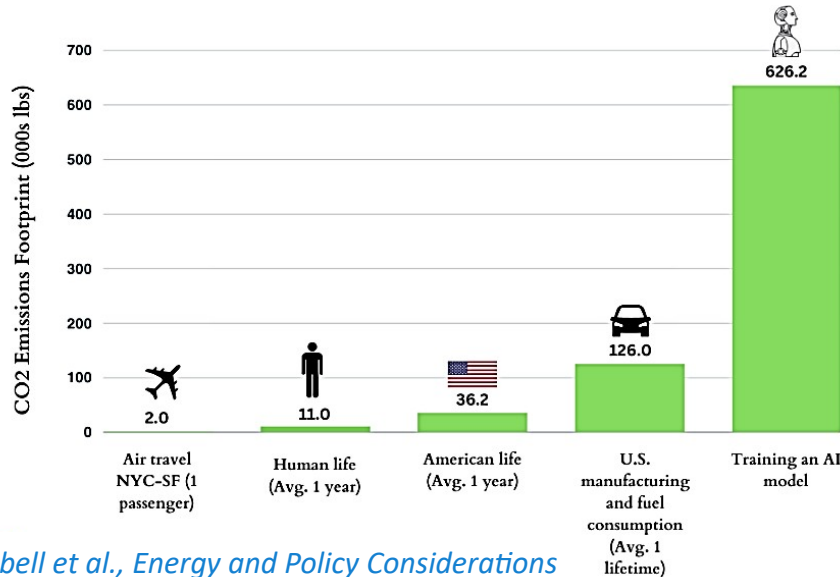


Imagine such a difficult surface on a space of 50 variables!

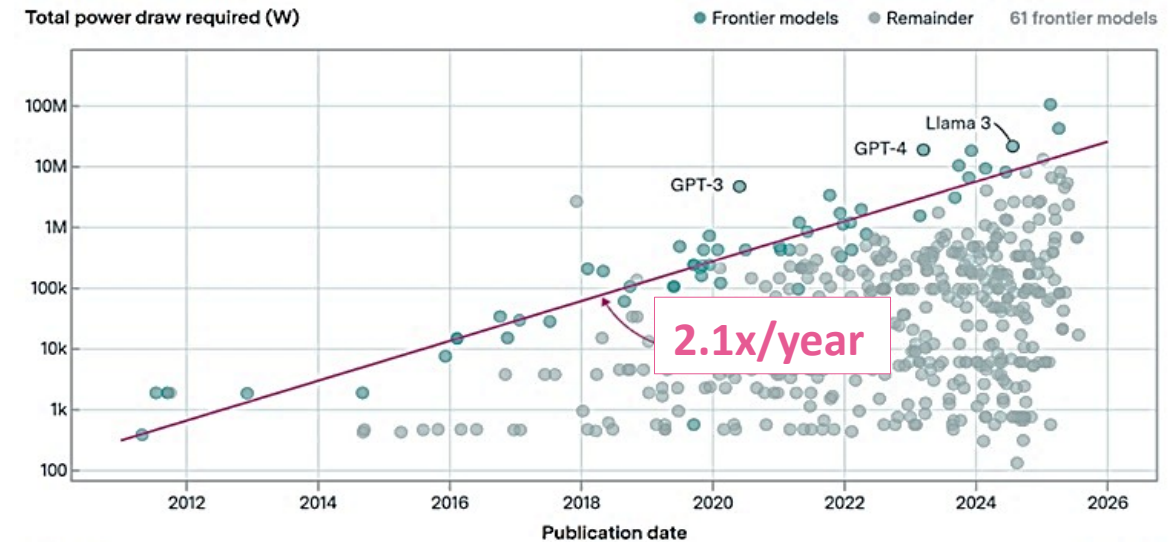
All That Glitters is not Gold

Computational demand & energy consumption

- Training AI models requires specialized infrastructure (GPUs, TPUs) which is costly and energy-intensive.
- Inference has the largest long-term energy cost, since it runs continuously. Researchers have estimated that a ChatGPT query consumes 5 times more electricity than a web search!



Strubell et al., Energy and Policy Considerations for Deep Learning in NLP (2019)

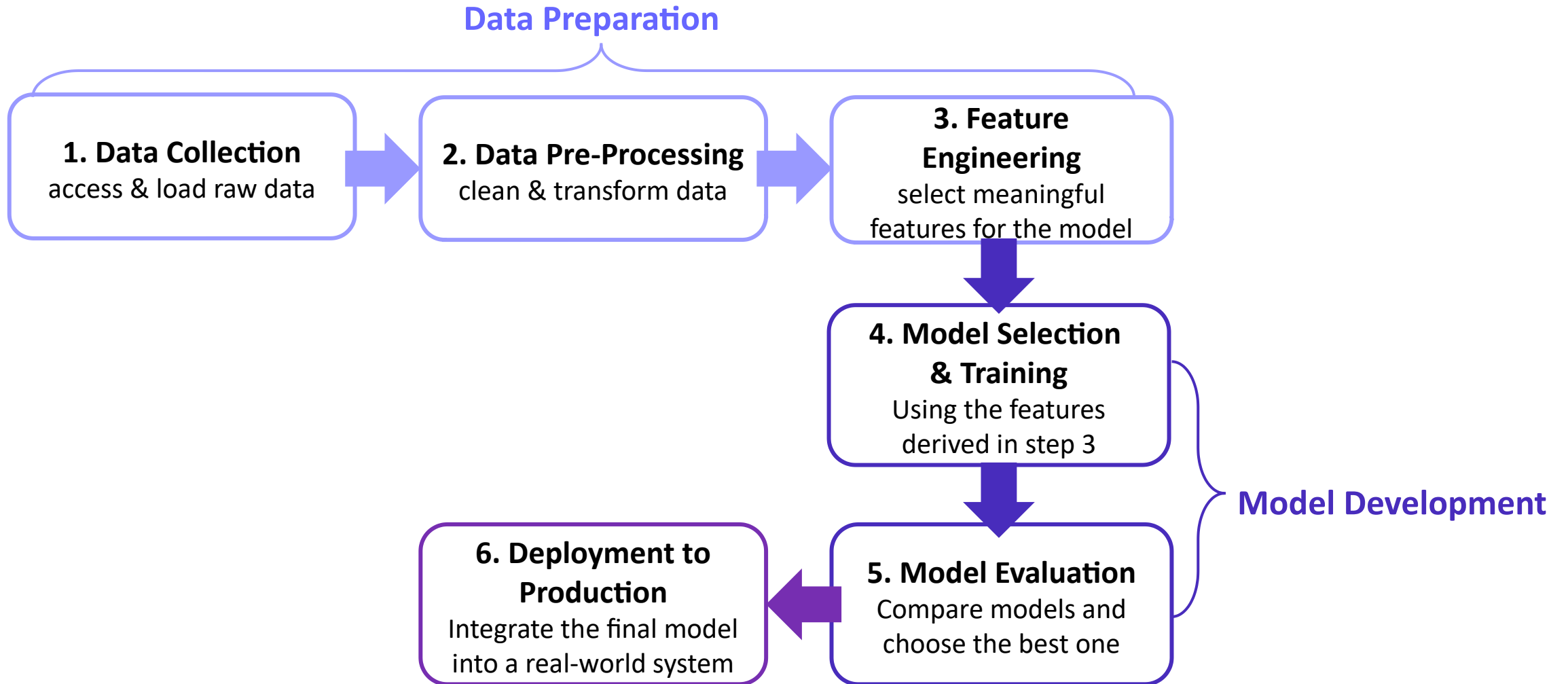


Scaling Intelligence: The Exponential Growth of AI's Power Needs. EPRI White Paper (2025)

Environmental impact

- Carbon dioxide emissions and pressures on the electric grid.
- Cooling AI hardware requires large amounts of water.
- E-waste can contaminate soil and water affecting human health.

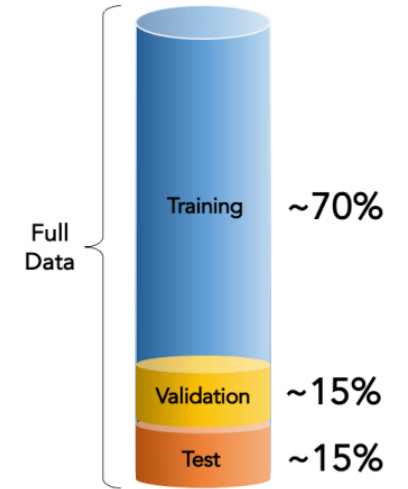
AI Pipeline: Key Stages to Focus On



Common Pitfalls in Data Preparation

Rule #1: Always Split your Data First

- Training with a dedicated **training set**, to make the model learn parameters
- Evaluate the model on a **validation set**, to monitor and optimize the training process
- Testing (*inference*) with a **new and unseen test set**, for the final model evaluation



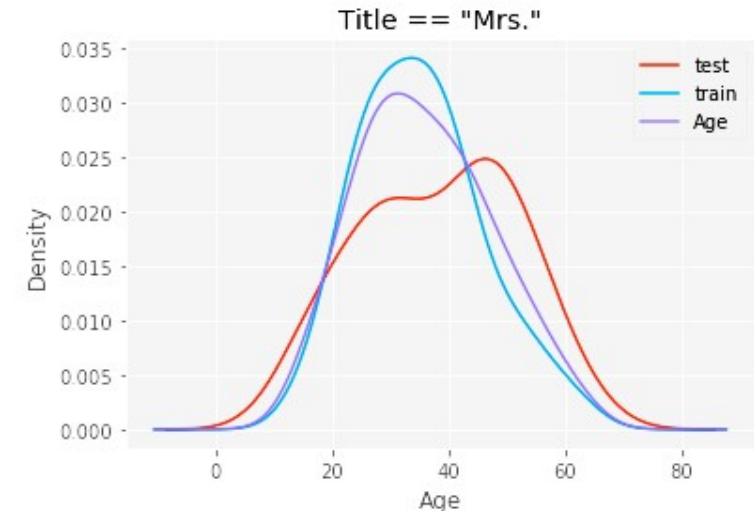
Goal: Low error on the difference between the training and test set prediction (*generalization*)



Pre-processing the entire dataset before proper splitting leads to a form of **data leakage** known as **train-test contamination**

For example, using mean imputation to handle missing values, if computed from the entire dataset would compromise the training, because in reality we would not know the true distribution in unseen data!

✗ over-optimistic performance
during testing/ real-world



Common Pitfalls in Data Preparation

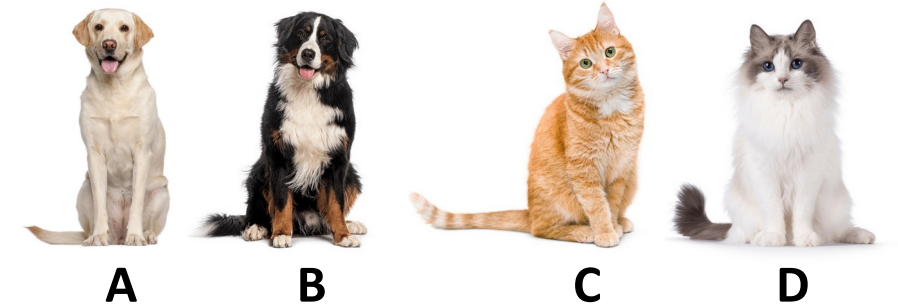
Rule #2: Split the Right Way

Let's say that I have multiple entities (e.g, cat/dog), and each entity is associated with several images.

I mix all images and take 70% for training and 30% for testing.

Wrong! I could have 7 images of dog "A" in training, and 3 images of dog "A" in testing. *How do I know if I'm learning features that are discriminative for the class dog, or for the specific dog "A"?*

I have to split **at entity level**, not at image level, to prevent bias!



- Dataset: {10 images of dog "A", 10 of dog "B", 10 of cat "C", 10 of cat "D"}
- Aim: classify if an image shows a cat or a dog

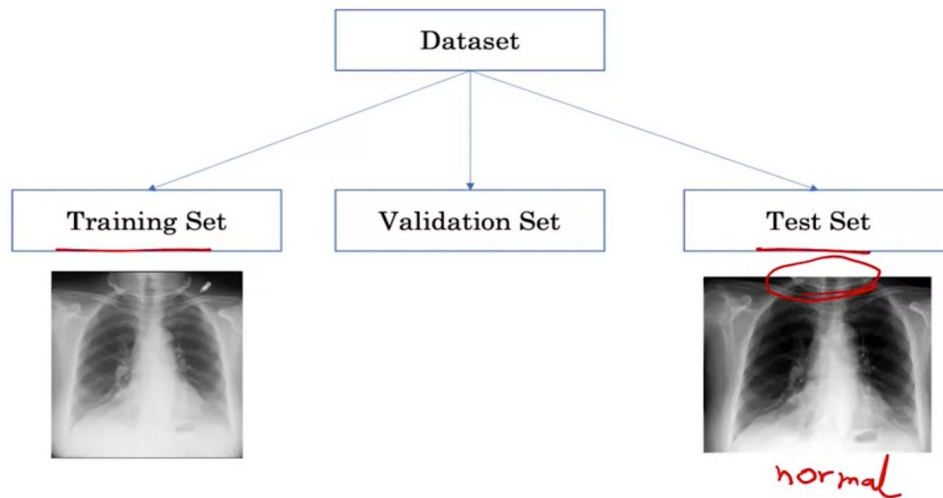
- ✓ Correct procedure: use all images of dog "A" and cat "C" for training, the rest for testing (**group-based splitting**)

Common Pitfalls in Data Preparation

Rule #2: Split the Right Way

Patient Overlap – the importance of group-based splitting in healthcare

If images from the **same patient** appear in both the training and test sets, the model may memorize patient-specific features instead of learning general patterns of the disease.



In this case, the patient was wearing a necklace.
The model could mistakenly learn to associate the necklace with the presence or absence of disease.

Patient-specific features may include:

- Scanner or device differences
- Noise or image artifacts
- Posture/positioning
- Individual anatomy

✓ Split By Patient

Training Set				Validation Set				Test Set			
Image	Patient id	Label (Mass)	...	Image	Patient id	Label (Mass)	...	Image	Patient id	Label (Mass)	...
xray1.jpg	20	1		xray7.jpg	32	1		xray4.jpg	17	0	
xray2.jpg	20	1		xray8.jpg	32	0		xray3.jpg	38	0	
xray0.jpg	20	1		xray9.jpg	32	0		xray5.jpg	11	0	
...				...				...			

Common Pitfalls in Data Preparation

Rule #2: Split the Right Way

Never Ignore **Class Imbalance**

Failure to address imbalanced datasets — where minority classes are underrepresented in training — leads to models that perform well on common cases but miss rare, potentially critical events.

When one or more of the classes are very rare, accuracy is misleading

		Actual values	
		Sick	Healthy
Predicted values	Sick	True positive 1	False positive 0
	Healthy	False negative 4	True negative 95

= $\frac{1 + 95}{1 + 0 + 4 + 95}$ = 96% sounds great...

Yet, all sick patients – except one - are missed.



Common Pitfalls in Data Preparation

Rule #2: Split the Right Way

Never Ignore **Class Imbalance**

Failure to address imbalanced datasets — where minority classes are underrepresented in training — leads to models that perform well on common cases but miss rare, potentially critical events.

When one or more of the classes are very rare, accuracy is misleading

		Actual values	
		Sick	Healthy
Predicted values	Sick	True positive 1	False positive 0
	Healthy	False negative 4	True negative 95

Alternative evaluation metrics become crucial:

$$= \frac{1}{1+0} = 100\%$$

= *How **reliable** the model's positive predictions are* (few false positive)

Pretty much, since the only detected sick patient was actually sick.



Common Pitfalls in Data Preparation

Rule #2: Split the Right Way

Never Ignore **Class Imbalance**

Failure to address imbalanced datasets — where minority classes are underrepresented in training — leads to models that perform well on common cases but miss rare, potentially critical events.

When one or more of the classes are very rare, accuracy is misleading



		Actual values	
		Sick	Healthy
Predicted values	Sick	True positive 1	False positive 0
	Healthy	False negative 4	True negative 95

Alternative evaluation metrics become crucial:

$$= \frac{1}{1+0+4+95} = 20\%$$

= *How **sensitive** the model is at catching all true positives* (few false negative)

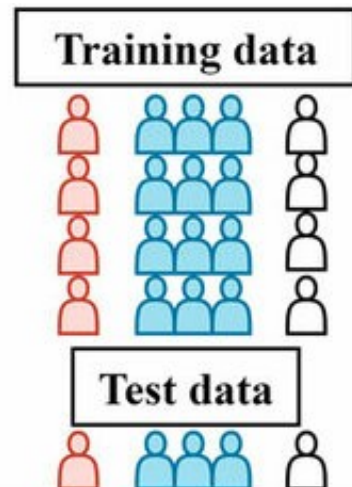
Pretty bad, since it couldn't detect the majority of them!

Common Pitfalls in Data Preparation

Rule #2: Split the Right Way

Never Ignore **Class Imbalance**

Failure to address imbalanced datasets — where minority classes are underrepresented in training — leads to models that perform well on common cases but miss rare, potentially critical events.



Stratified splitting divides a dataset so that the original **class proportions are preserved** in both training and test sets.

- ✓ Ensures rare classes are present in both training/test
- ✓ Leads to more reliable evaluation metrics

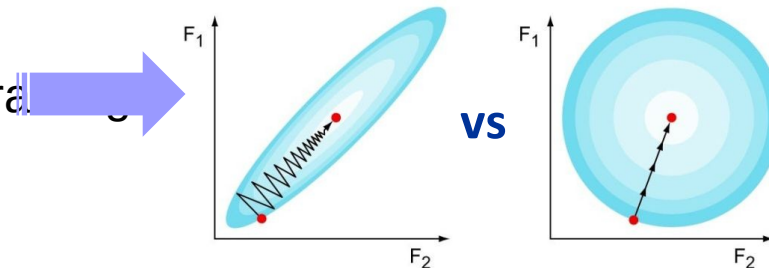
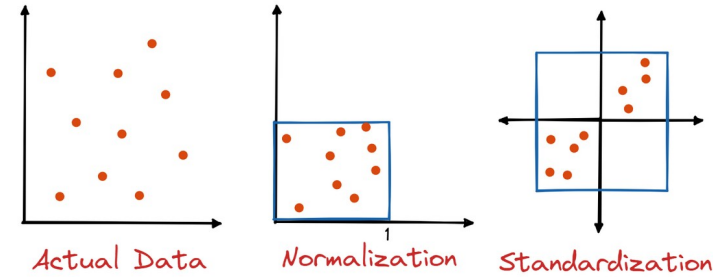


Common Pitfalls in Data Preparation

Rule #3: Scale Numeric Features When Needed

When numeric variables have different ranges and the algorithm relies on distance metrics.

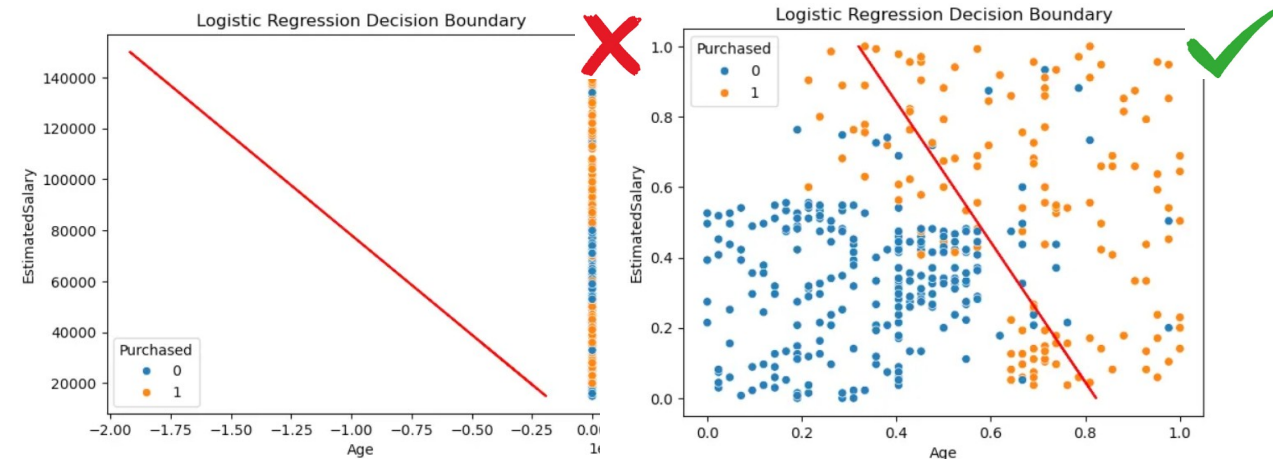
- Increases model accuracy, by ensuring that each feature **contributes equally** to the model, regardless of its scale/unit
- Helps the model **converge faster** for a smoother training process



Example: Predict purchase based on age and estimated salary

	Age	EstimatedSalary	Purchased
0	19.0	19000.0	0
1	35.0	20000.0	0
2	26.0	43000.0	0
3	27.0	57000.0	0
4	19.0	76000.0	0
...	...	...	...
395	46.0	41000.0	1
396	51.0	23000.0	1
397	50.0	20000.0	1
398	36.0	33000.0	0

- Without scaling, model decision boundary is biased toward estimated salary, ignoring age
- Result: All points classified as class 1 (purchase)



Common Pitfalls in Data Preparation

Rule #4: Handle One-Hot Encoding with Care

Categorical variables must be converted in numerical inputs. One-hot encoding creates a new binary column (*dummy variable*) for each unique value of the variable.

Flavour	Chocolate	Vanilla	Strawberry
Chocolate	1	0	0
Vanilla	0	1	0
Strawberry	0	0	1

The Dummy Variable Trap

When the number of dummy variables is equal to the number of values, one or more variables can be predicted from the others.

Multicollinearity = variables highly correlated with each other

Linear regression models assume that the variables are independent:

- ✗ hard to identify their individual effects on the prediction.
- ✗ Inaccurate estimates of the regression coefficients.

$Chocolate + Vanilla + Strawberry = 1$ (always!)
 $Strawberry = 1 - Chocolate - Vanilla$
if an ice cream **isn't chocolate or vanilla**,
it **must be strawberry!**

Common Pitfalls in Data Preparation

Rule #4: Handle One-Hot Encoding with Care

Categorical variables must be converted in numerical inputs. One-hot encoding creates a new binary column (*dummy variable*) for each unique value of the variable.

Flavour	Chocolate	Vanilla	Strawberry
Chocolate	1	0	0
Vanilla	0	1	0
Strawberry	0	0	1

The Dummy Variable Trap

How to avoid it?

Drop one of the dummy columns, which removes the linear dependency.
This is equivalent to choosing a **reference category** and comparing the other categories to it.

$$\text{Sales} = \beta_0 + \beta_1(\text{Chocolate}) + \beta_2(\text{Vanilla}) + \beta_3(\text{Strawberry}) + \varepsilon$$

let's use Strawberry as the reference

$$\text{Sales} = \beta_0 + \beta_1(\text{Chocolate}) + \beta_2(\text{Vanilla}) + \varepsilon$$

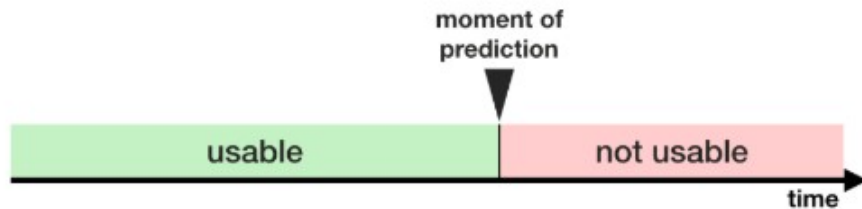
Now:

- β_0 (intercept) is the average sales for Strawberry
- β_1 is **how much more (or less)** Chocolate sells than Strawberry
- β_2 is **how much more (or less)** Vanilla sells than Strawberry

Common Pitfalls in Data Preparation

Rule #5: Select the Proper Features

Another form of data leakage is **target leakage**:



- the features have some dependency on the variable that we want to predict (*target*)
- the features will not be available at the time of prediction

Imagine you want to predict who will get sick with pneumonia:

got_pneumonia	age	weight	male	took_antibiotic_medicine	...
False	65	100	False	False	...
False	72	130	True	False	...
True	58	100	False	True	...

There is a strong relationship between `got_pneumonia` and `took_antibiotic_medicine`, but the latter is frequently changed *after* the value for the former is determined.

✗ Patients who later develop pneumonia would not yet have been treated with antibiotics when making predictions about their health!

Any variable updated (or created) after the target value is realized should be excluded!

Common Pitfalls in Model Development

Rule #6: Balance Bias & Variance

Bias and variance are two big sources of error:

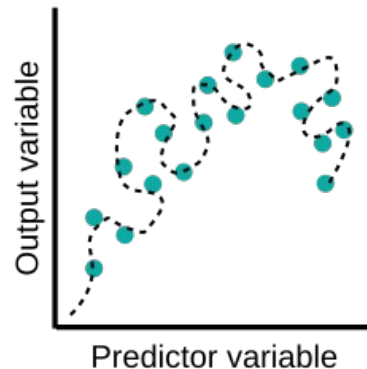
➤ **Bias** is the error the model makes on the training set

- Models are too simple
- Not enough or weak features
- Not enough training

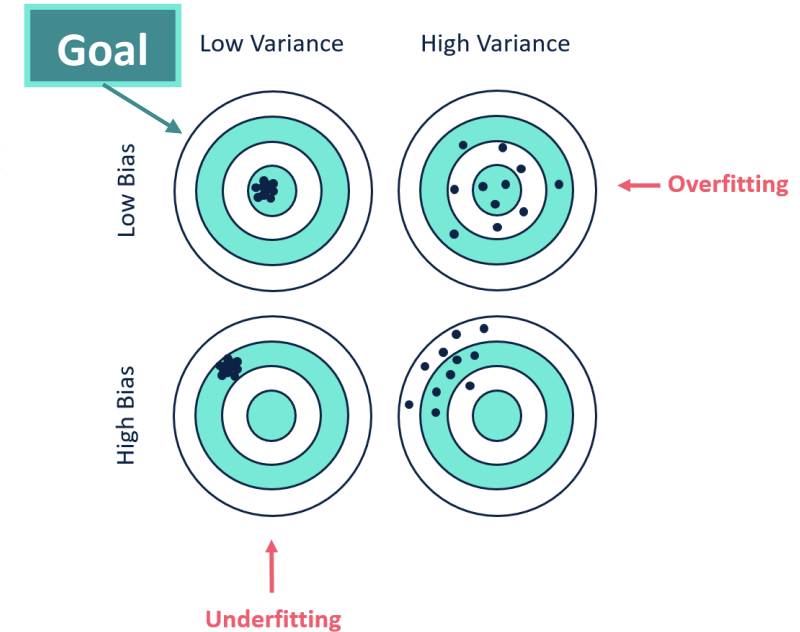
High bias leads to **underfitting**: the model is ignoring meaningful data patterns

➤ **Variance** is the error the model makes on the test set compared to the training set

- Models are overly complex
- Not enough data
- Too many features

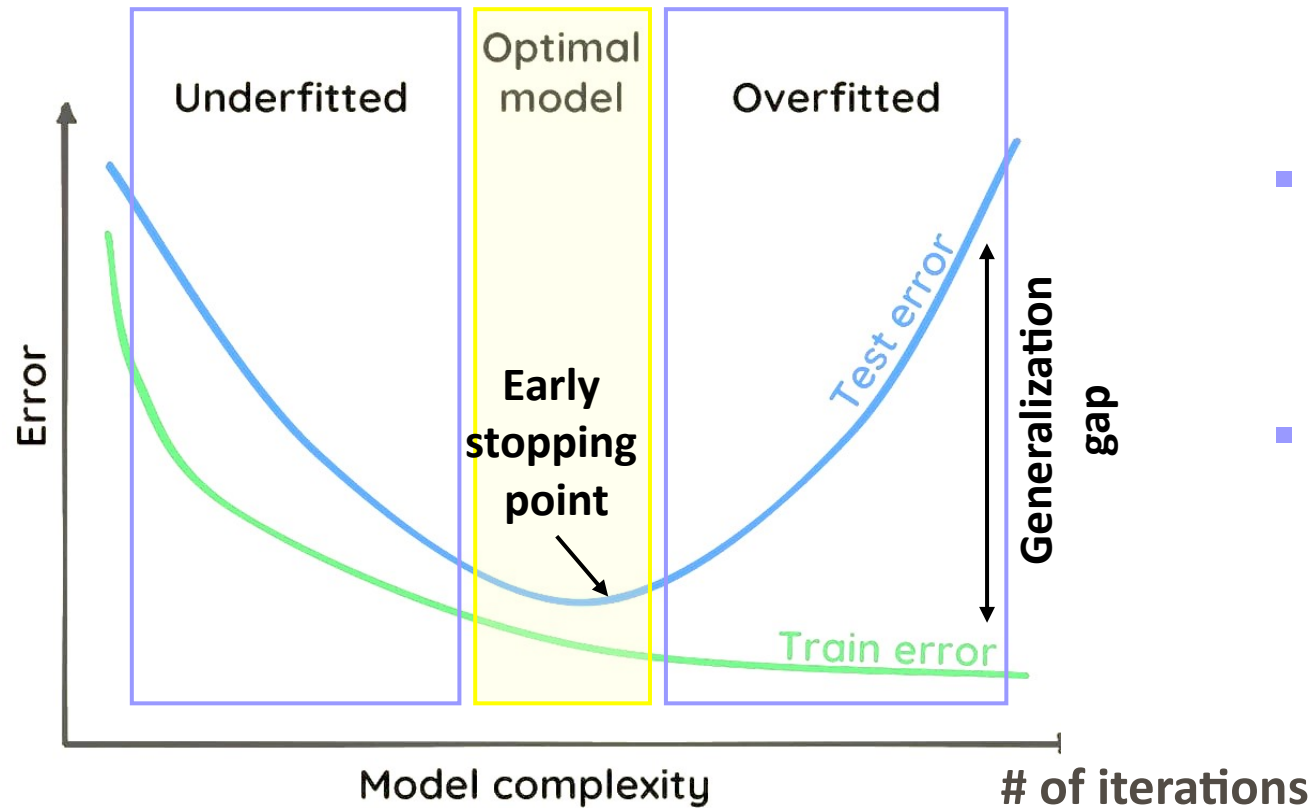


High variance leads to **overfitting**: the model is learning too much from the data (including noise) and fails to generalize to new data



Common Pitfalls in Model Development

Rule #6: Balance Bias & Variance



- If a model shows poor performance on both sets, it's probably underfitting.
- If a model performs exceptionally well on its training data but poorly on the test set, it is likely overfitting.

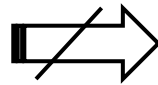
Common Pitfalls in Model Deployment

Rule #7: Ensure Robustness in Real-World

In real world, models must handle input variations or unexpected scenarios (*domain shift*)

Generalization

How well the model performs on unseen data **from the same distribution**

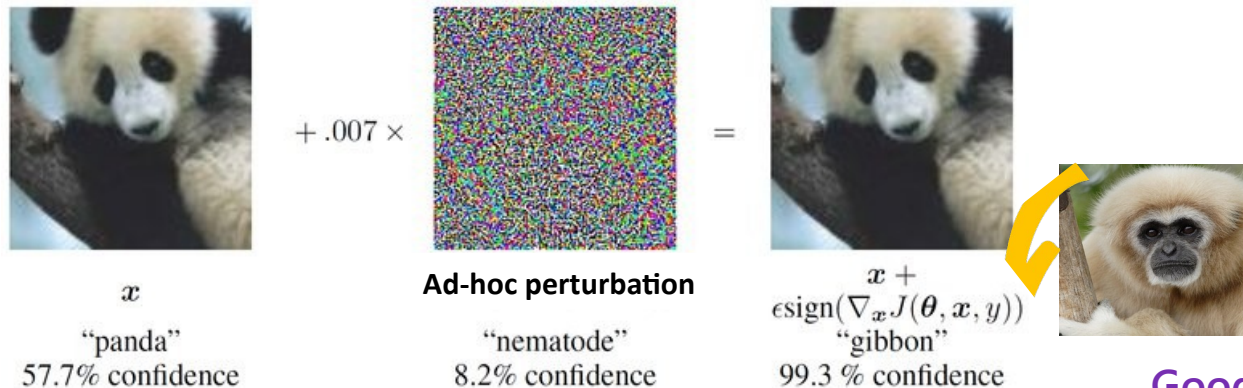


Robustness

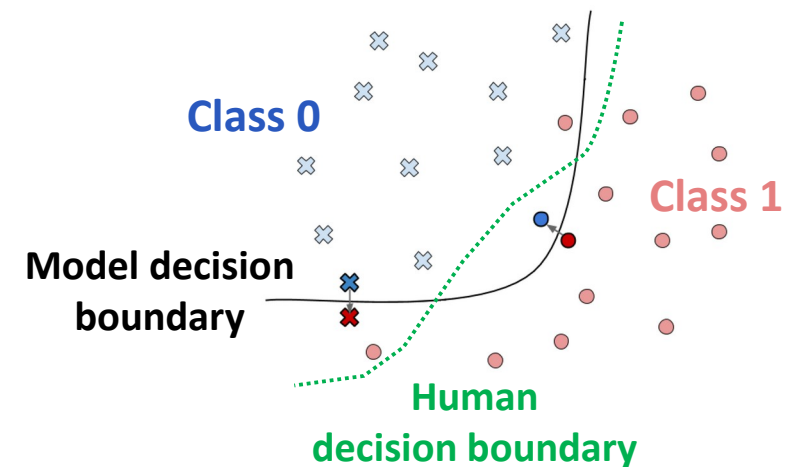
How well the model performs on unseen, slightly different data (**out-of-distribution**) or in **new contexts**



Adversarial perturbations *I. J. Goodfellow et al. 2014*



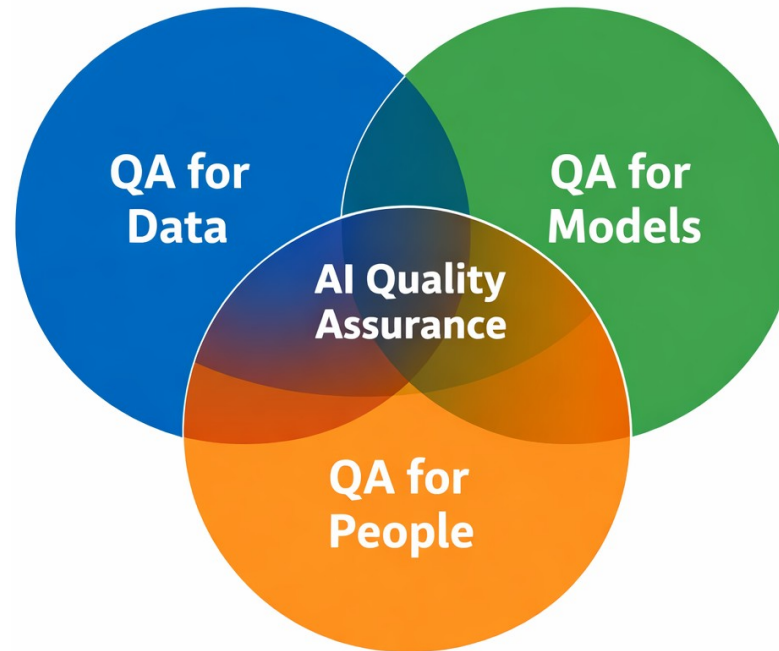
GoogLeNet (2014)



Ensuring Quality in AI

“By far, the greatest danger of Artificial Intelligence is that people conclude too early that they understand it.” —Eliezer Yudkowsky (AI Safety Researcher)

“Is there any chance that data are affecting the training in some way?”



“Is the model designed to accurately represent the reality it is trying to predict?”

Can users understand and trust the model’s output when making decisions?

QA for Data

Data Processing aims to guarantee that the training data meet the data quality criteria:

Accuracy	Completeness	Uniqueness	Validity	Consistency	Representativeness	Objectivity
correctly represents the attribute it describes	contains no missing values	free from duplicate records	conforms to required formats, rules and standards	coherent across sources and within datasets	reflects the diversity of the scenarios in which the model is intended to operate	free from bias and discrimination

Data Cleaning

- Modifying/removing incorrect, improperly formatted or duplicated data
- Removing/replacing missing values with mode or median

Feature Engineering

- Converting categorical values to numeric values
- Dropping irrelevant, highly correlated features
- Data normalization -> $[0,1]$ (neural networks, image data)
- Data standardization (z-scoring) -> $\langle \hat{a} \rangle = 0, \text{Var}[\hat{a}] = 1$ (most traditional ML models)

QA for Models

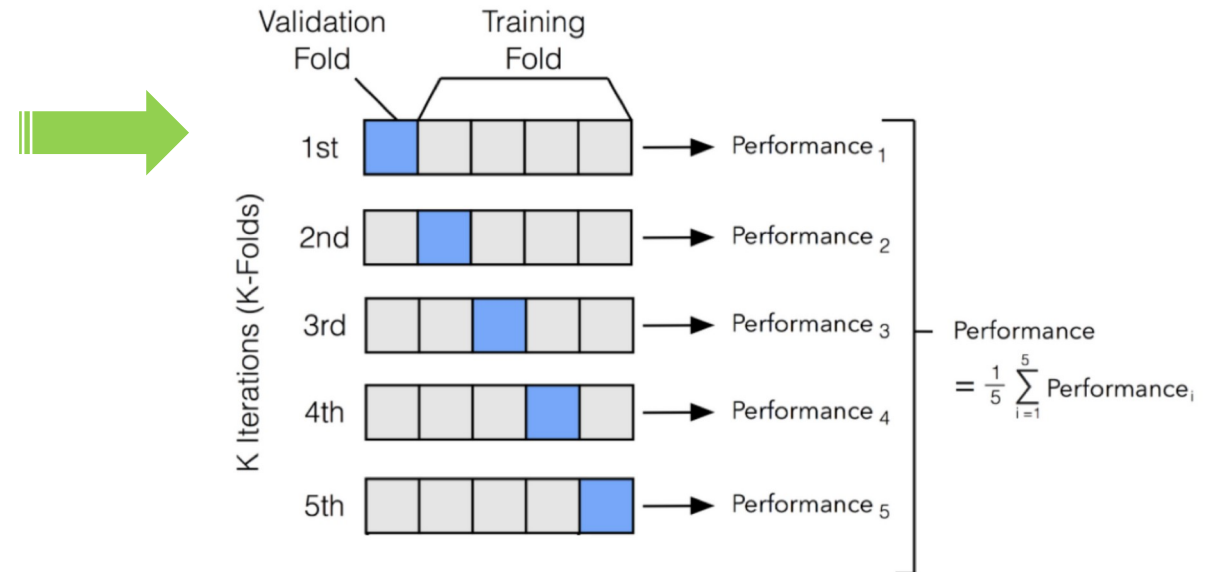
During development, validation techniques assess data behavior and ensure the model delivers accurate and consistent predictions over time:

Prevent overfitting during training

- Check if the data has proper size
 - ~1,000: clustering, exploratory analysis;
 - < 10k: regression & classification models; > 10k: neural networks, DL

- Use **cross-validation**: split data into multiple folds for iterative train/test, then average the results; useful when data are scarce/imbalanced.

- If data is too little, consider **Data Augmentation**:



- Reduce sensitivity to specific patterns with **regularization** techniques (e.g. Ridge, Lasso)

QA for Models

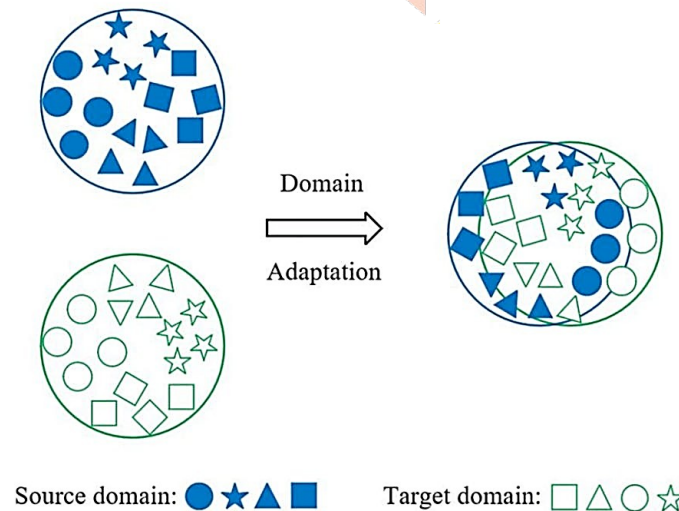
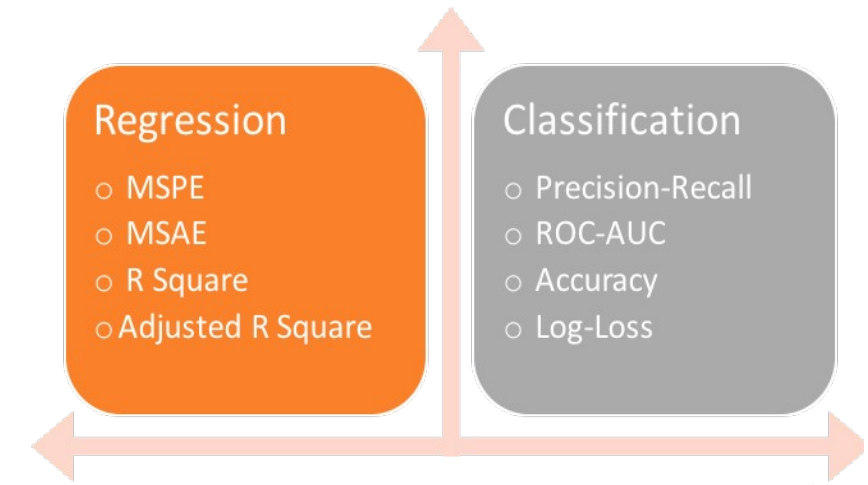
During development, validation techniques assess data behavior and ensure the model delivers accurate and consistent predictions over time:

Rely on performance evaluation metrics

- Translate real-world goals into measurable targets, to quantify the gap between desired and current performance
- Track progress over time

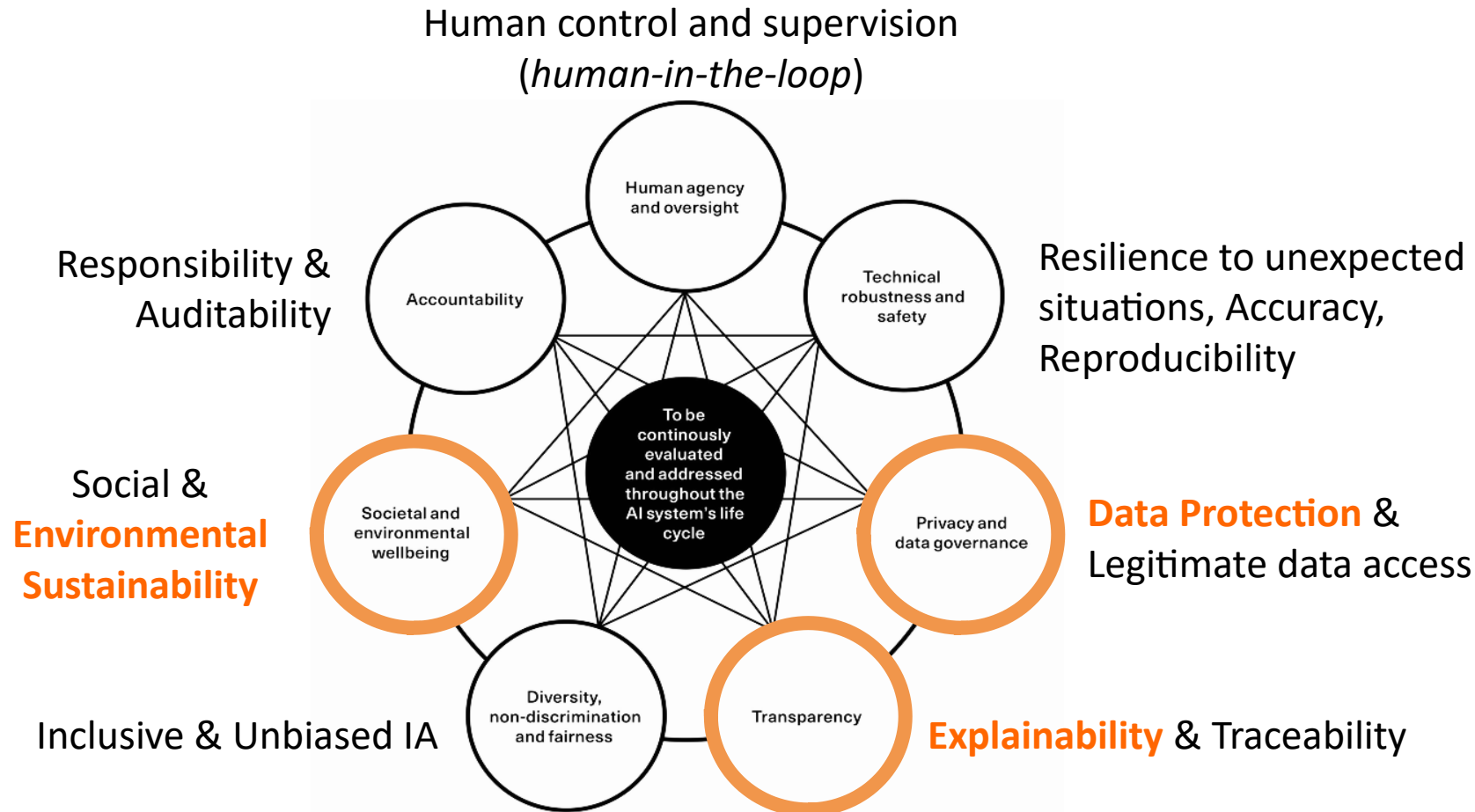
Check robustness in testing stage

- Include out-of-distribution data
- Migration to different but similar scenarios -> **Domain Adaptation**
- May imply a trade-off with accuracy/complexity



QA for People

7 key requirements set up by AI HLEG in 2019 for a **Trustworthy AI**, adopted by the **AI Act** (EU 2024/1689)



<https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai>

QA for People

Explainable AI (XAI) – Opening the Black Box

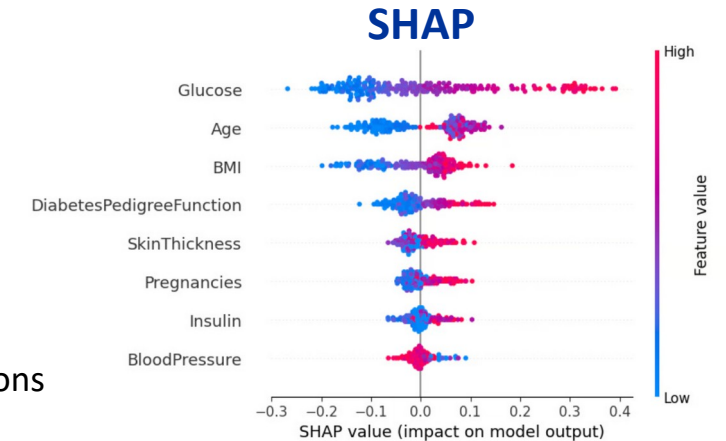
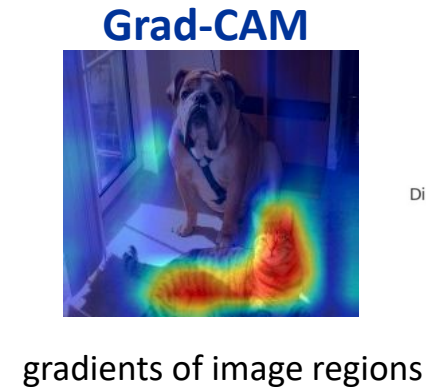
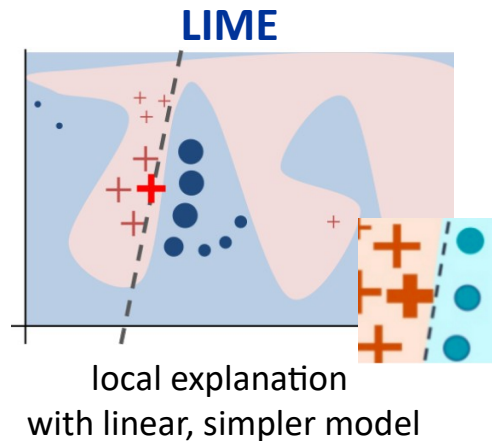
A broad field of AI research developing methods to increase **human trust** in and **understanding of ML predictions**

- More interpretable models (trade-off with accuracy)
- Decision traceability, to follow model reasoning

- ✓ Meet the “right to explanation” from GDPR
- ✓ Understand and fix errors/bias in the model
- ✓ Make **informed decisions**

Importance methods

Show which input data/features drives the prediction



Counterfactual methods

Identify the minimal input change that alters the prediction



allocates prediction impact
across features

QA for People

Privacy Protection

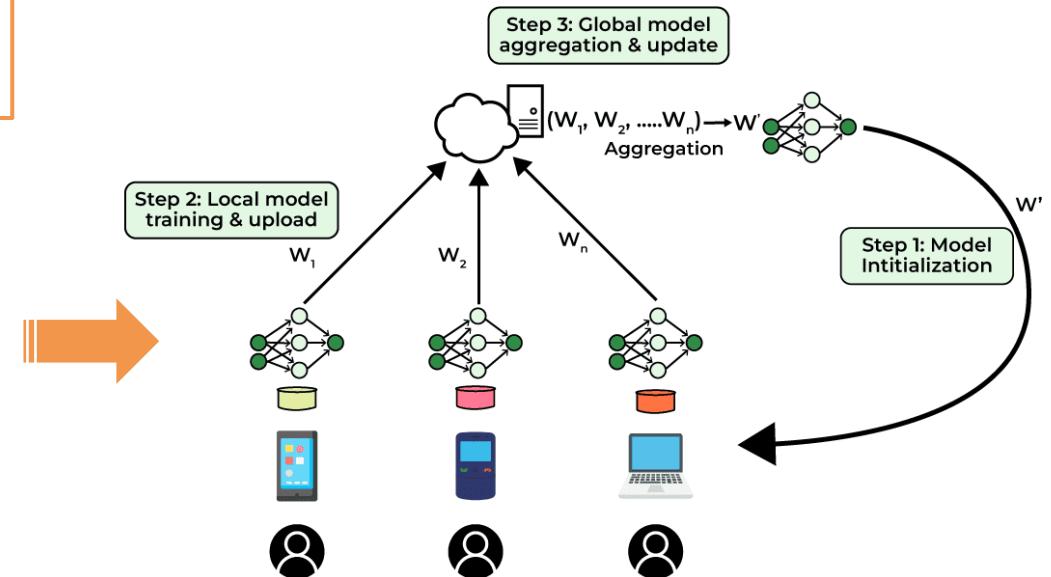
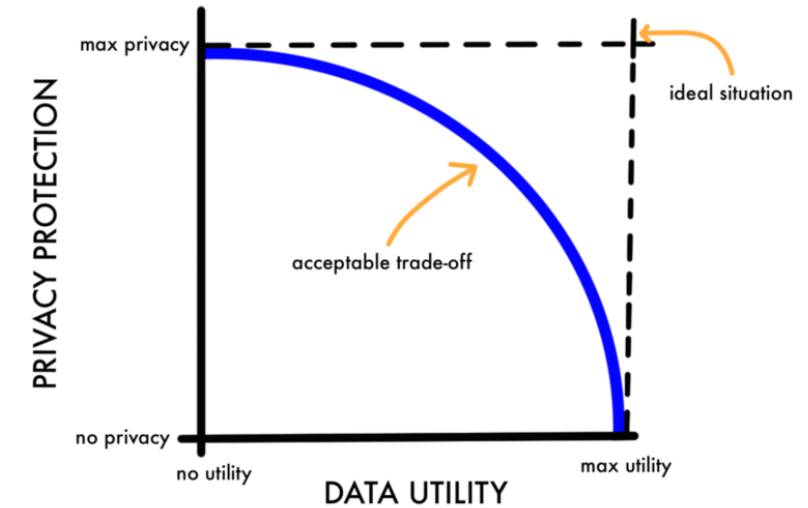
Synthetic Data are artificially generated data that:

- maintains the statistical properties of real-world data (*data utility*)
- guarantees **minimal risk of disclosure** of sensitive information (*data protection*)

AI ACT

- Art. 10(5)(a): Allows the use of SD when RWD is insufficient, for **bias detection and correction**.
- Art. 59(1)(b): Recognizes SD as a **compliant alternative** to meet regulatory requirements.

Federated Learning: Decentralized training of ML models on many user devices, with only updates being sent to a central server for aggregation and no data sharing.



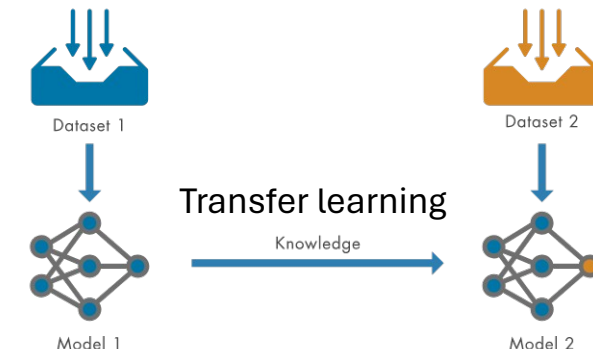
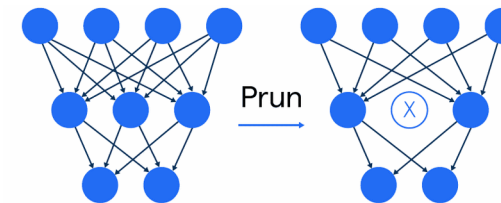
QA for People

Environmental impact and sustainability

Different factors drive energy consumption: model and dataset size, number of training steps, inefficient/suboptimal training strategies...

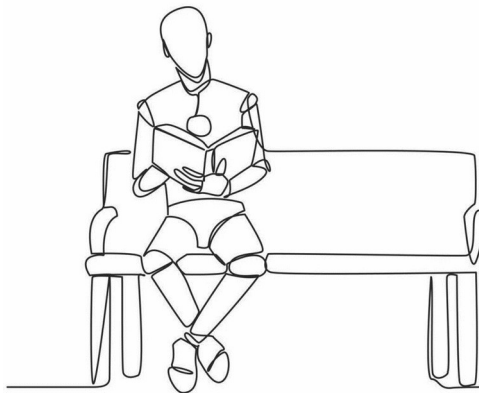
QA includes resource efficiency:

- **Model efficiency** (latency, memory, throughput):
 - **Model compression:**
 - Pruning – removing redundant weights/channels
 - Quantization – lower numerical precision for faster inference
 - Knowledge distillation – training a smaller «student» model
 - **Early stopping** – stop training when the model stops improving
 - **Transfer learning** – using pre-trained models to avoid training from scratch
- **Hardware-aware optimization:**
 - Choose architectures that align with target hardware (e.g., mobile)
 - Use efficient runtime framework to run model faster on the device



Concluding Remarks on Building Quality into AI

- Every AI challenge starts with **data**. Real-world data can be messy, incomplete, heterogeneous. Take your time to understand and prepare it: no model, however sophisticated, can compensate for poor data quality.
- Choosing a model is never about finding “the best one”. It’s a **balancing act** that depends on context. And it’s not just achieving excellent performance in controlled conditions, but also ensuring robustness in unpredictable real-world scenarios.
- Beyond data and models, AI is ultimately designed **by people for people**. It reflects our assumptions, biases and decisions. The future of AI depends not only on better technology, but also on investigating the human condition. **If we don't like what we see, we have to change ourselves, not the AI.**



A **high-quality AI system** is not one that never fails, but one where **we understand when, how, and why it fails.**