Day 4

The Project Lifecycle (Part 2)

Recap

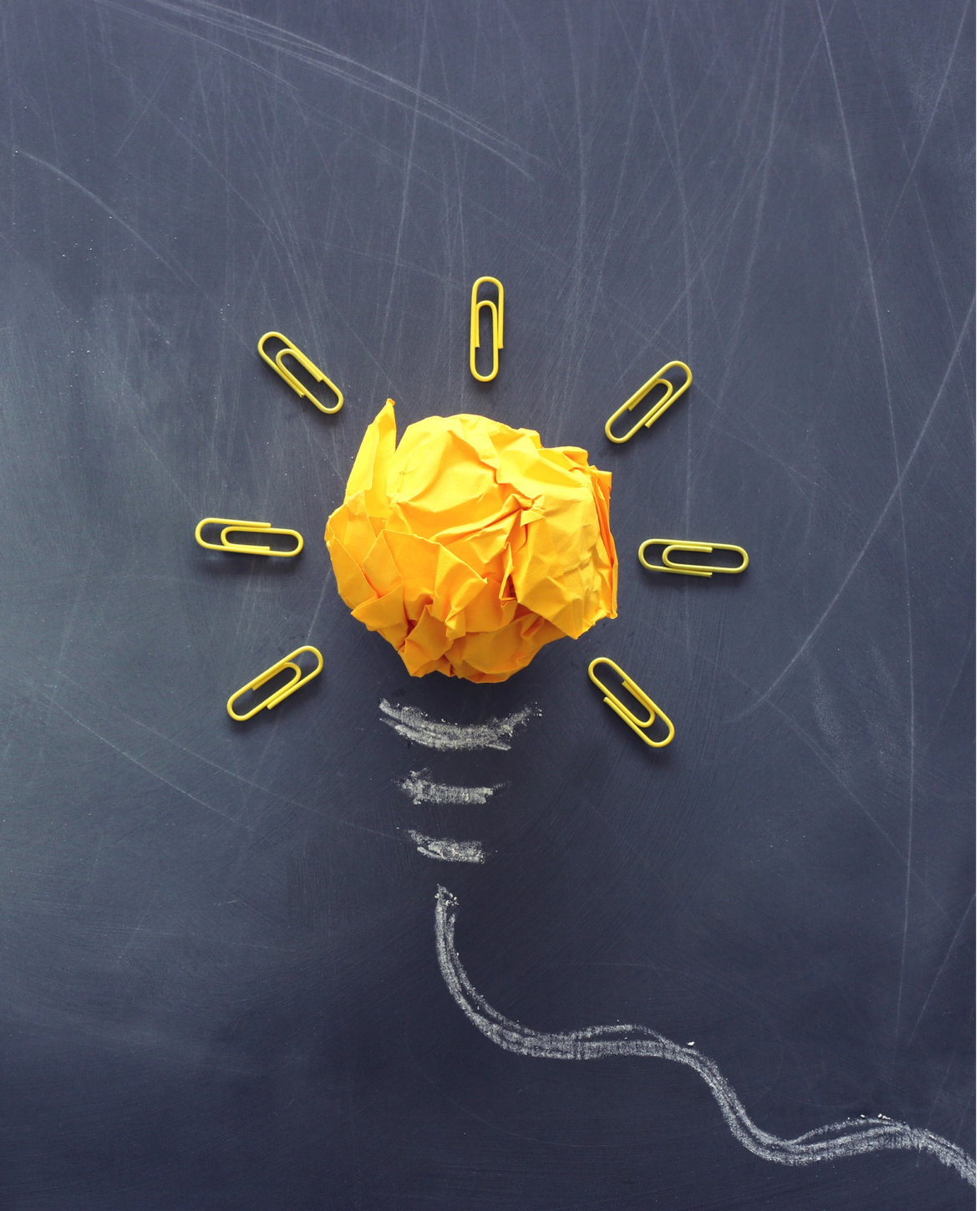
Day 3

- The Project Lifecycle (Project Design)
  - Project Planning
  - Problem Formulation
  - Data Extraction and Procurement
  - Data Analysis
- Guest Lecture (Professor Sabina Leonelli)

Overview

Day 4

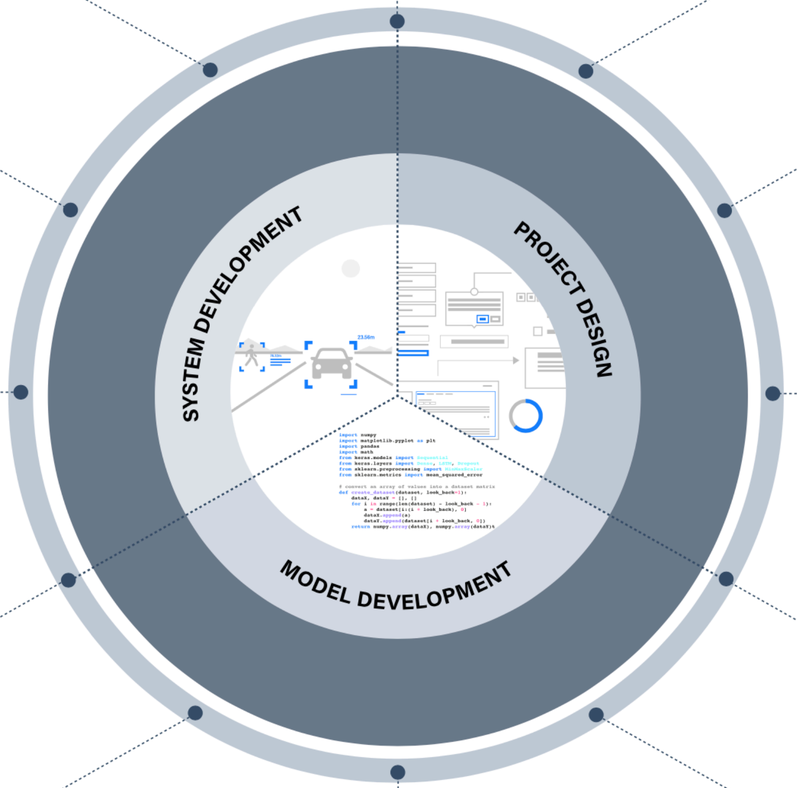
- Model Development
  - Preprocessing & Feature Engineering
  - Model Selection & Training
  - Model Testing & Validation
  - Model Reporting
- System Deployment
  - Model Productionalisation
  - User Training
  - System Use & monitoring
  - Model Updating or Deprovisioning



Day 4

Learning Objectives

- Gain a high-level understanding of the (model) development and (system) deployment stages.
- Explore the activities that are associated with these stages, focusing on salient ethical, social, and legal issues.



Project Planning

Problem Formulation

Data Extraction & Procurement

Data Analysis

Preprocessing & Feature Engineering

Model Selection & Training

Model Testing & Validation

Model Reporting

Model Production-alisation

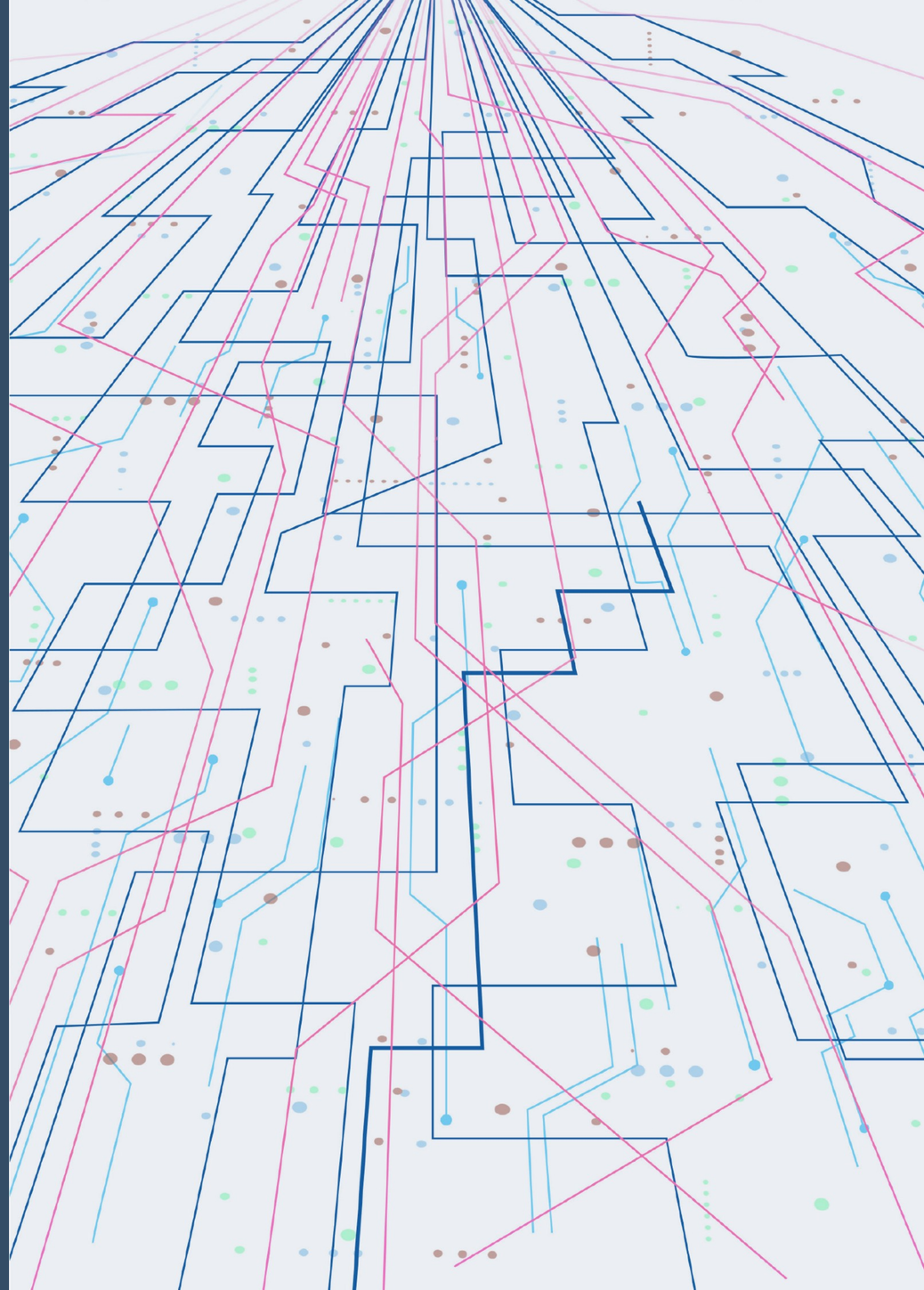
User Training

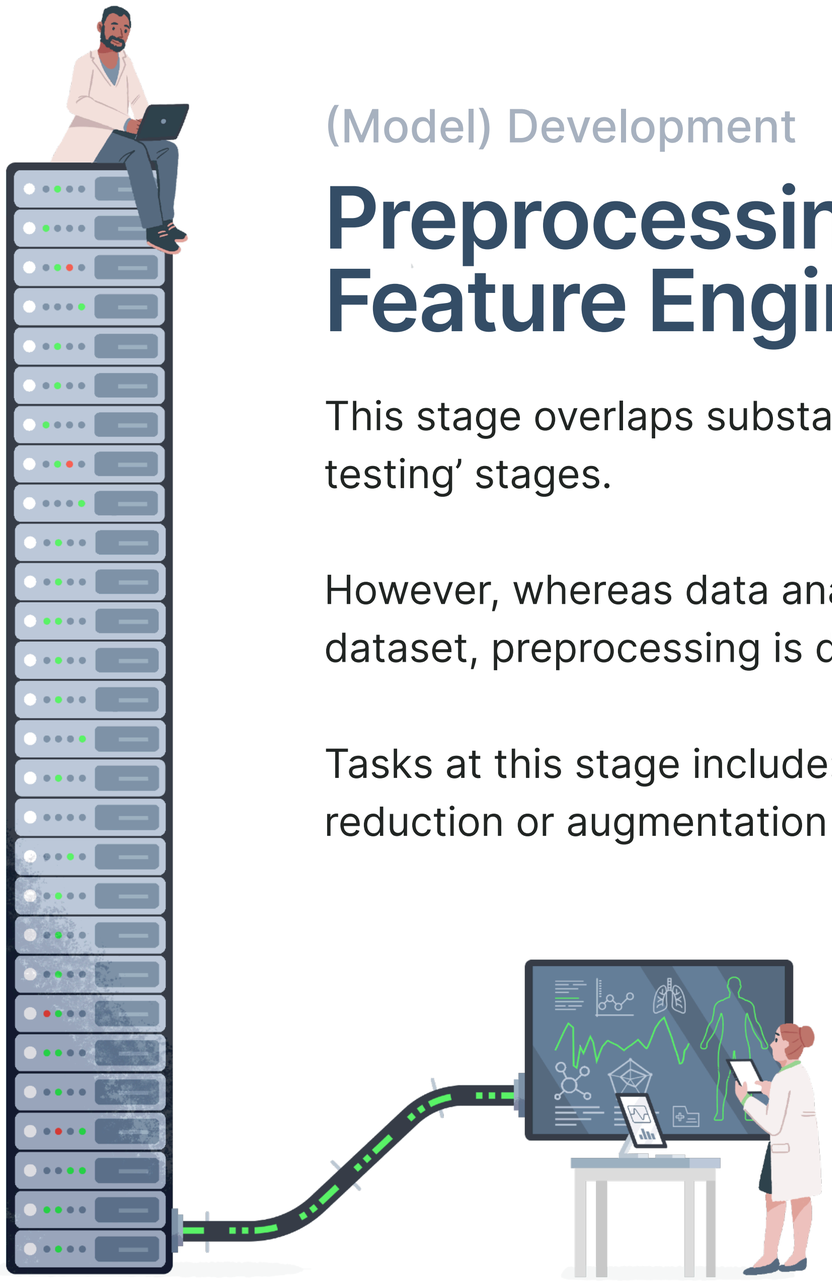
System Use & Monitoring

Model Updating & Deprovisioning

Day 4

(Model) Development





(Model) Development

Preprocessing & Feature Engineering

This stage overlaps substantially with the ‘data analysis’ and ‘model selection, training, and testing’ stages.

However, whereas data analysis is oriented towards a provisional understanding the dataset, preprocessing is directed towards the model development tasks.

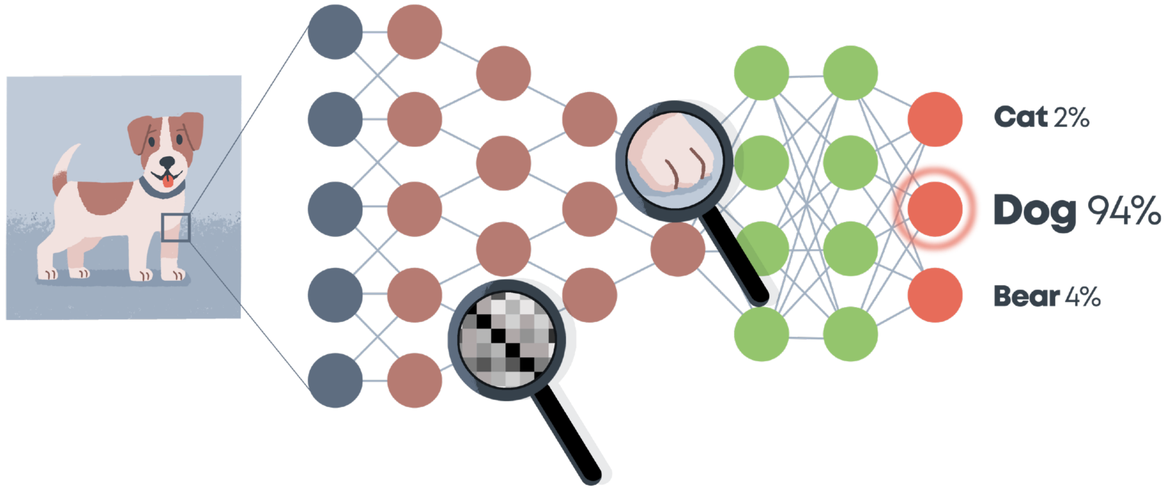
Tasks at this stage include: additional) data cleaning, data wrangling or normalisation, data reduction or augmentation

(Model) Development

Model Selection

Types of machine learning:

- Supervised
  - Classification (e.g. Neural Networks)
  - Prediction (e.g. Linear Regression)
- Unsupervised
  - Pattern Recognition (e.g. K-Means Clustering)
  - Dimensionality Reduction (e.g. PCA)
- Reinforcement
  - Optimal Action/Policy Selection (e.g. Q-Learning)



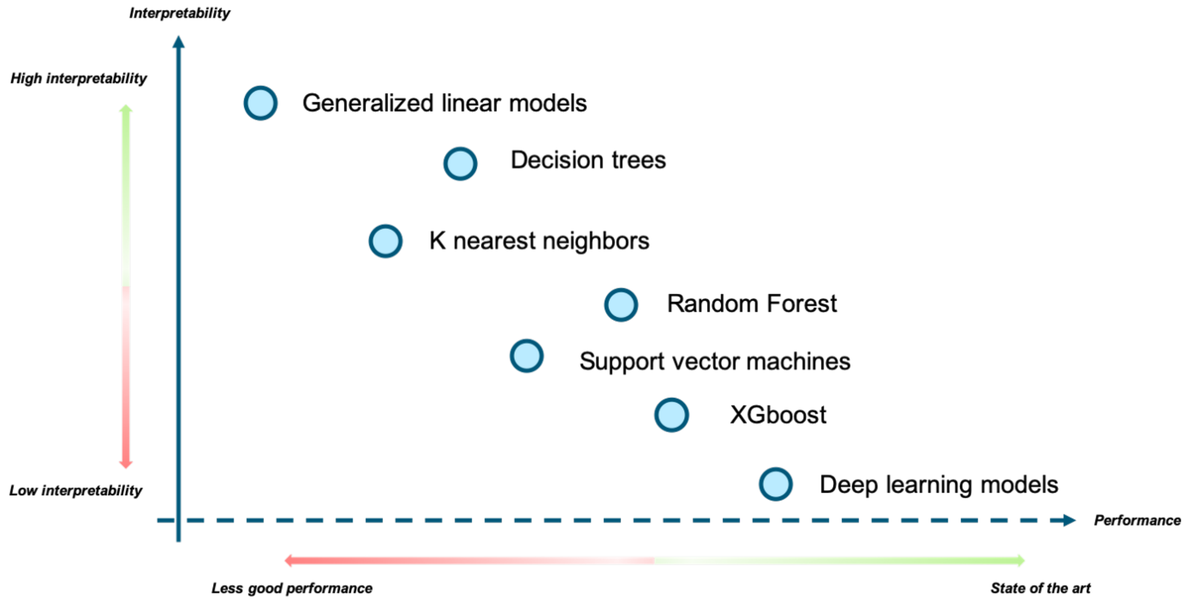
Model Selection

Interpretability

There is a well-known trade-off between the inherent interpretability of an algorithm and its predictive performance.

It gives rise to one of the biggest ethical, legal, and social challenges associated with (model) development:

- how to determine the acceptable threshold for accuracy while meeting requirements for explainability?



Model Selection

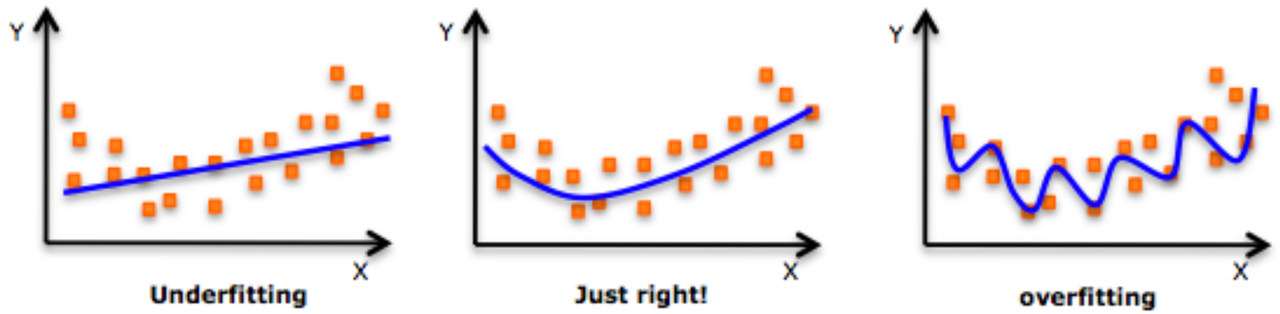
Explainability

(Model) Development

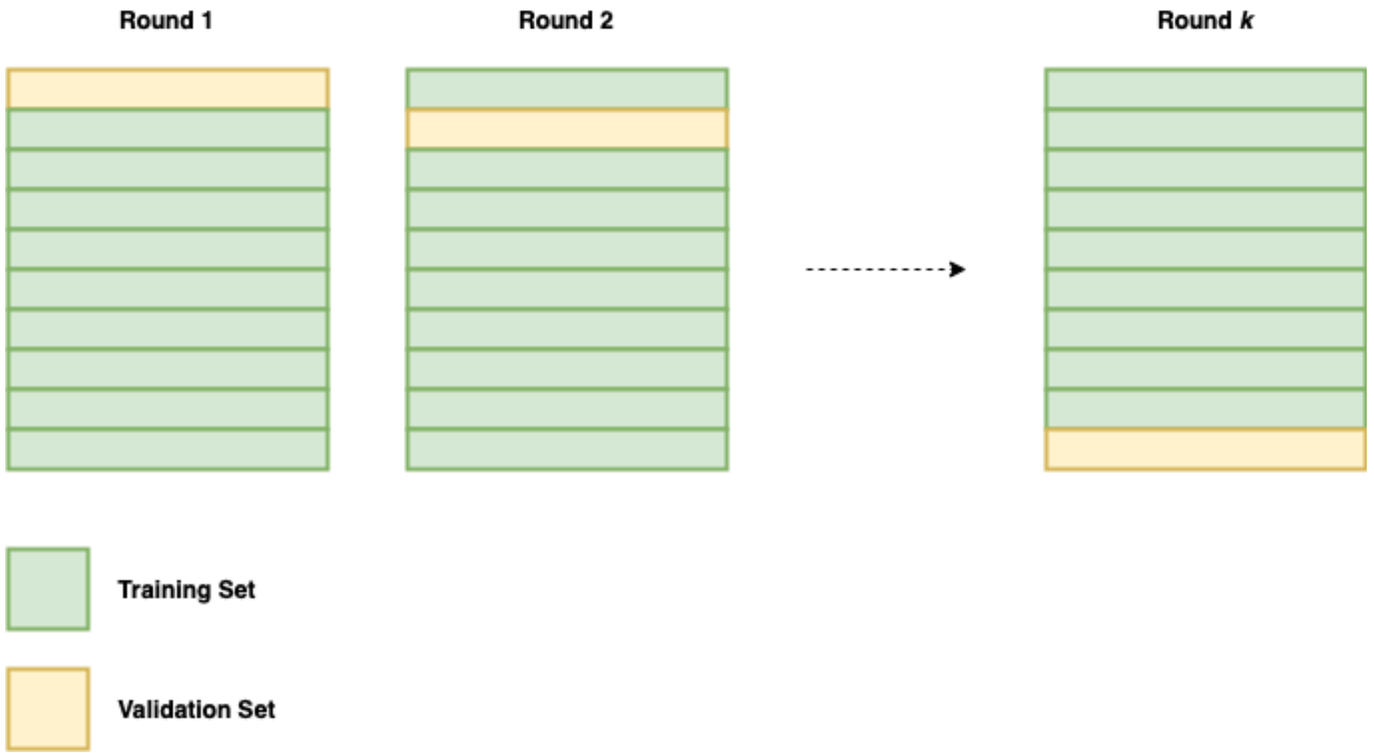
Model Training, Testing and Validation

Two important (but non-exhaustive) considerations:

- Underfitting and Overfitting—achieving desired level of model generalisability
- Fairness optimisation



Cross-Validation



Model Training, Testing and Validation

Fairness Optimisation

Famous case from ProPublica exposing variation in risk score accuracy between ‘White’ and ‘African American’ defendants:

	WHITE	AFRICAN AMERICAN
Labeled Higher Risk, But Didn’t Re-Offend	23.5%	44.9%
Labeled Lower Risk, Yet Did Re-Offend	47.7%	28.0%

False Positive

Model Training, Testing and Validation

Fairness Optimisation

Famous case from ProPublica exposing variation in risk score accuracy between ‘White’ and ‘African American’ defendants:

	WHITE	AFRICAN AMERICAN
Labeled Higher Risk, But Didn’t Re-Offend	23.5%	44.9%
Labeled Lower Risk, Yet Did Re-Offend	47.7%	28.0%

False Negative

Fairness Optimisation

Due to the precision-recall trade-off, however, it’s not possible—except in the most trivial of cases—to simultaneously optimise for all values.

	High Risk	Low Risk
Did Reoffend	True Positive	False Positive
Did Not Reoffend	False Positive	True Negative

Fairness Optimisation

Due to the precision-recall trade-off, however, it's not possible—except in the most trivial of cases—to simultaneously optimise for all values.

	High Risk	Low Risk
Did Reoffend	True Positive	False Positive
Did Not Reoffend	False Positive	True Negative

Precision = number of accurate predictions as a fraction of all positive predictions

Fairness Optimisation

Due to the precision-recall trade-off, however, it's not possible—except in the most trivial of cases—to simultaneously optimise for all values.

	High Risk	Low Risk
Did Reoffend	True Positive	False Positive
Did Not Reoffend	False Positive	True Negative

Precision = number of accurate predictions as a fraction of all positive predictions

Recall = number of accurate predictions as a fraction of actual cases of interest

Fairness Optimisation

Due to the precision-recall trade-off, however, it's not possible—except in the most trivial of cases—to simultaneously optimise for all values.

	High Risk	Low Risk
Did Reoffend	True Positive	False Positive
Did Not Reoffend	False Positive	True Negative

But different stakeholders or affected users are likely to care more about which of these values is optimised for during model development!

Society or Victim ≠ Defendant

(Model) Development

Model Reporting

Model reporting is a process of developing and integrating documentation and evidence about the processes by which your model was designed and developed (e.g., trained, tested, and evaluated), as well as how it ought to be used or deployed.

(Model) Development

Model Reporting

Several options exist for systematising the process:

- Data Nutrition Project
- TRIPOD+ Statement
- Model Cards for Model Reporting

Data Facts

RadAbd Summary

Machine learning-based decision support software to augment medical imaging-related diagnosis of abdominal CT scans

Type of Algorithm employed

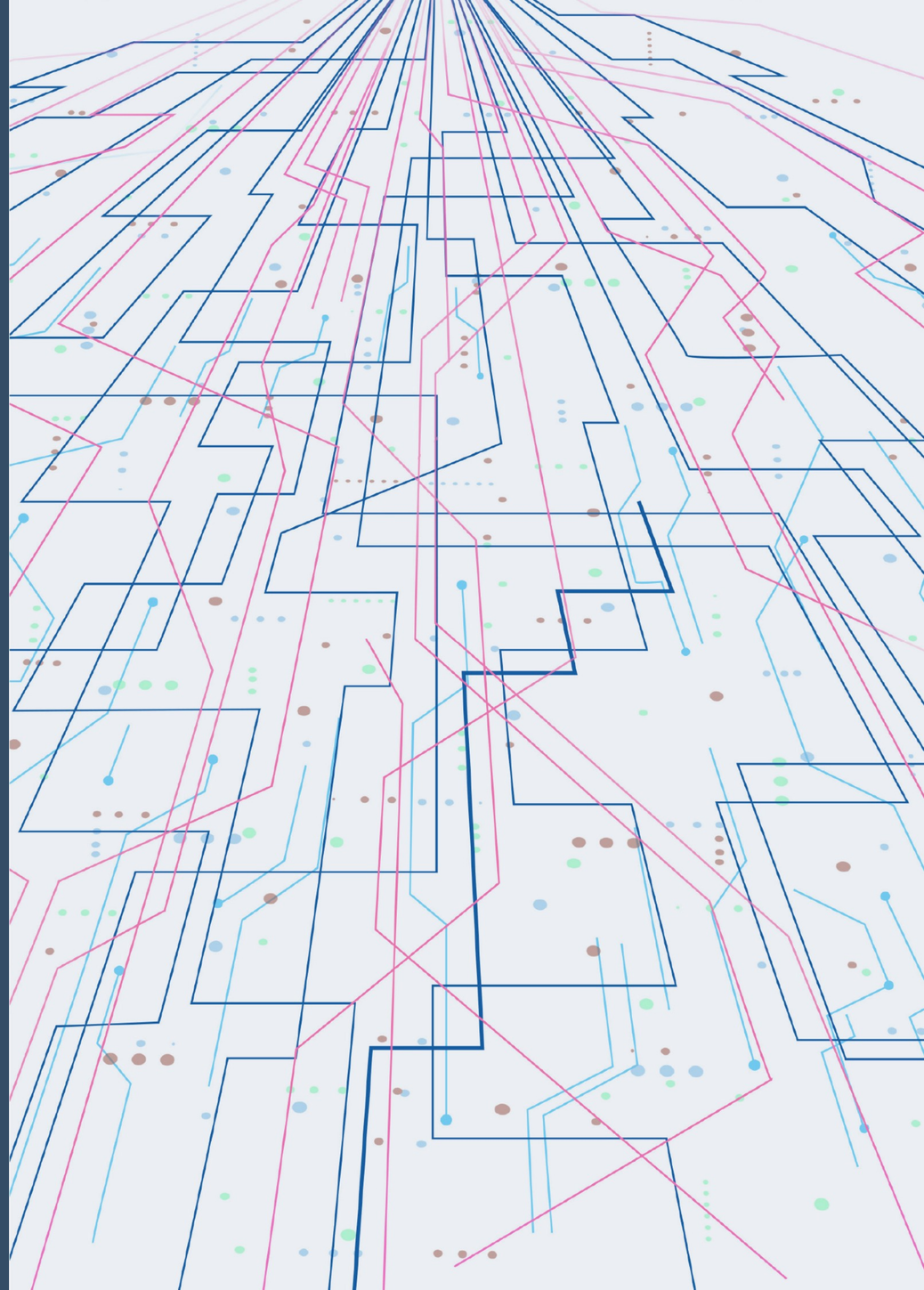
Convolutional neural network

Population composition

Ethnic composition	
Non-Hispanic White	**60%**
Hispanic and Latino	**18%**
Black or African American	**13%**
Asian	**6%**
Other	**3%**
Gender balance	
Male/Female	**55/45%**

Activity 9

Designing Model Reports

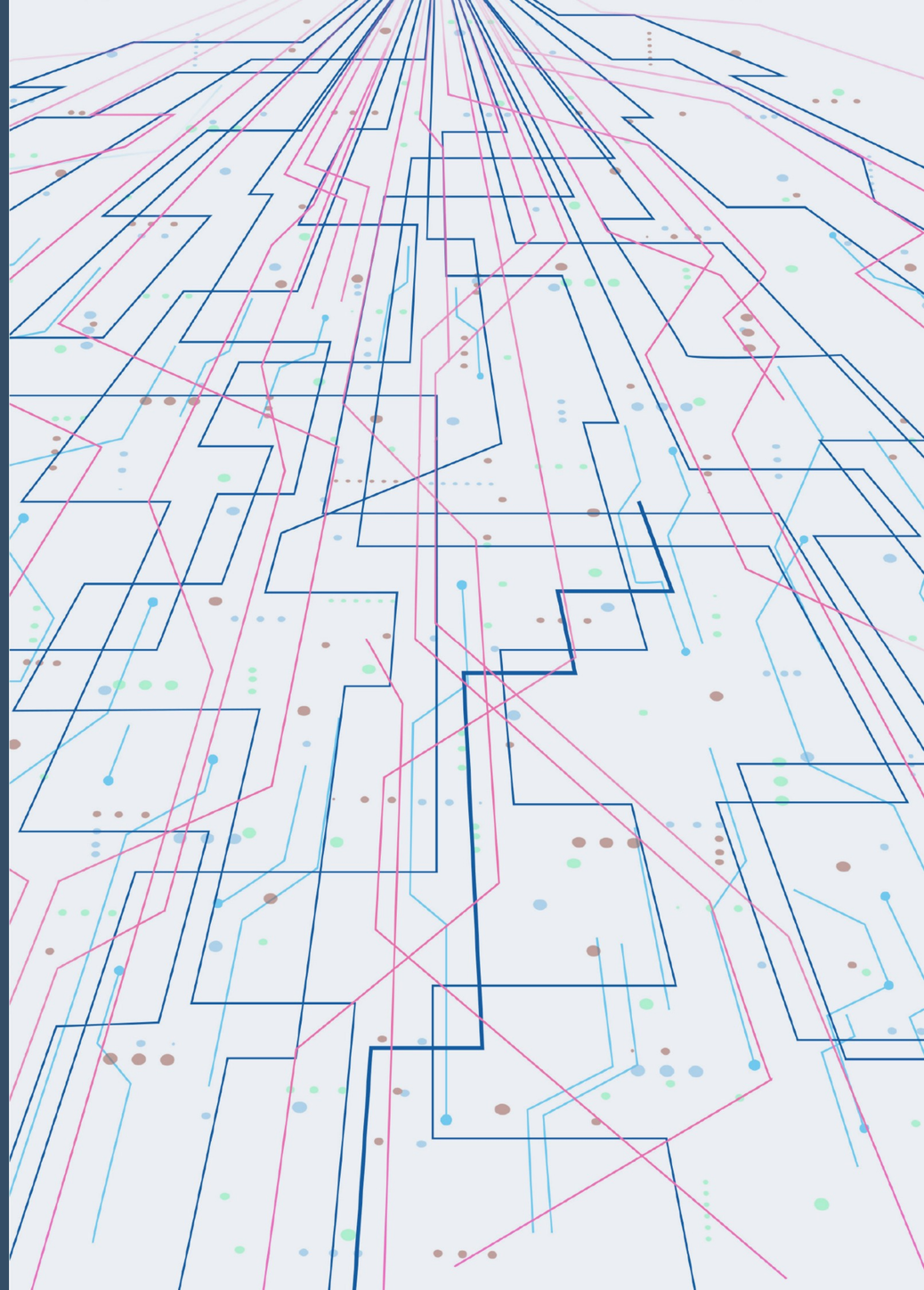


Breakout Groups

Plenary

Day 4

(System) Deployment



(System) Deployment

Model Productionalisation

Unless the end result of the project is the model itself, which is perhaps more common in scientific research, it is likely that the model will need to be implemented within a larger system.

This process requires understanding:

- how the model is intended to function in the proximate system
- how the model will impact—and be impacted by—the functioning of the wider sociotechnical environment

(System) Deployment

User Training

The performance of your system and its ability to achieve the intended or desired outcome is dependant upon **human factors.**

Human factors is a field in which researchers and practitioners are interested in both understanding the interaction of people and technology and in making that interaction more efficient, safer, and pleasant. (Durso et al., 2014)

User Training

Human Factors

You are in charge of deploying an automated facial recognition system that is used to verify that people who are attempting to enter a secure building are using the appropriate identity card. Upon swiping their identity card, the system scans the face of the user and matches it to the expected image from a database of authorised people. If the person matches their card and is also allowed access to the building, they are automatically granted access.

After a week or so of deploying the system, you find out that the security guard in charge of the building has been overriding the system. You go to speak with the security guard, and he claims that the system has been refusing entry to people who clearly match their identity badge.

When you investigate the issue, you find out that the system is functioning as expected and that no errors with the automated facial recognition system have been logged. However, every one of the attempts that the security guard has overridden are for people with expired identity cards. Although they match their cards, they should not have access to the building.

User Training

Cognitive Biases

In the context of RRI, user training related to how an algorithmic system should be operated may include:

a) conveying basic knowledge about the nature of machine learning (e.g. probabilistic results or outcomes),
b) explaining the limitations of the system,
c) educating users about the risks of AI-related biases, such as **decision-automation bias** or **automation-distrust bias**, and
d) encouraging users to view the benefits and risks of deploying these systems in terms of their role in helping humans to come to judgements, rather than replacing that judgement.

(System) Deployment

System Use & Monitoring

You cannot step twice into the same rivers; for fresh waters are ever flowing in upon you.

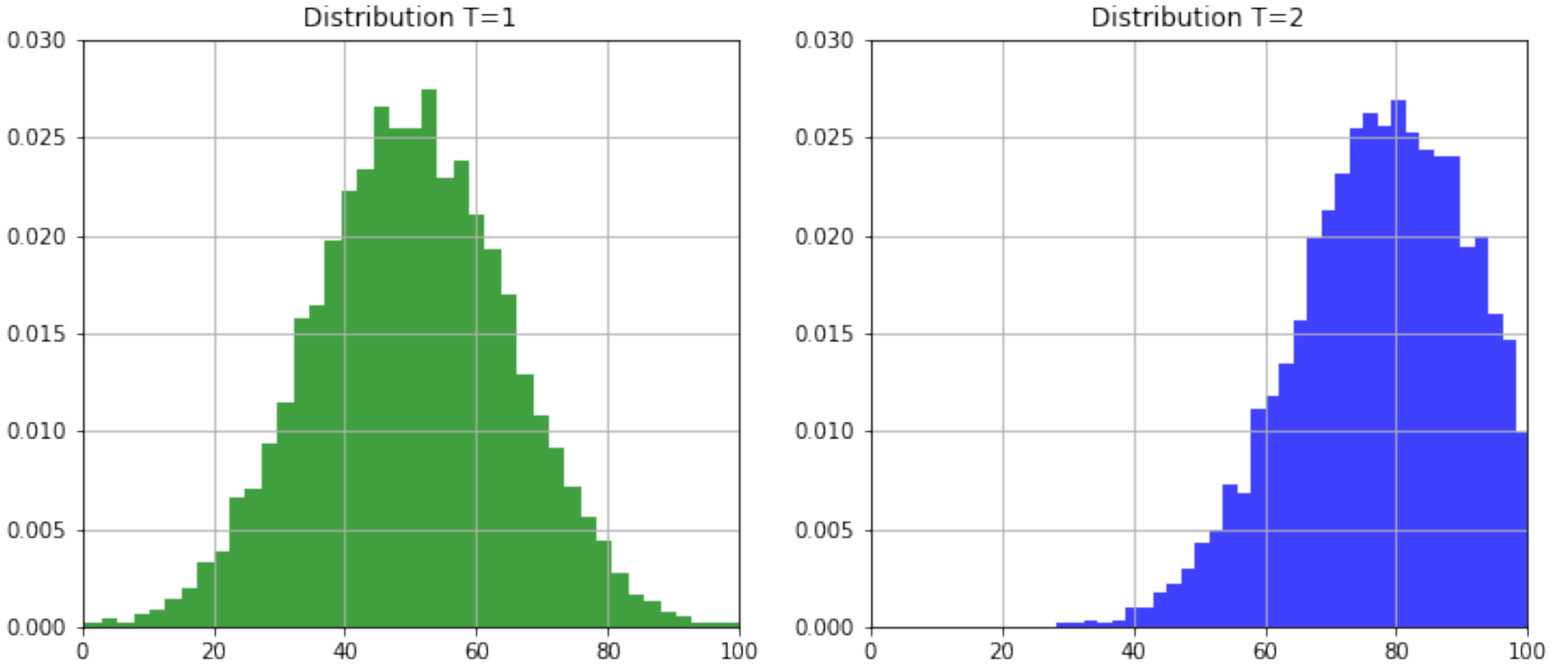
—Heraclitus, Fragment 12

System Use & Monitoring

Model Drift

The world changes, and as it does the ability for the model to capture the underlying data generating process may also drift.

This limits the generalisability of a model across time due to deteriorating performance.

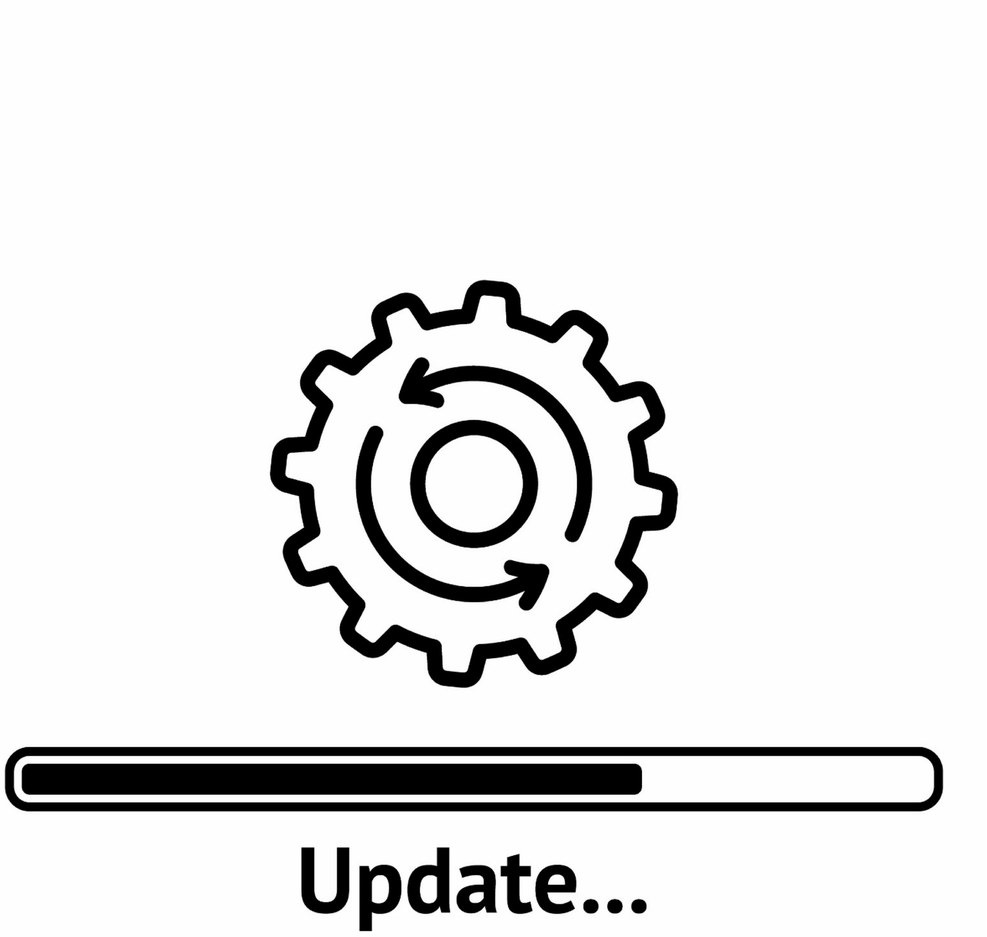


System Use & Monitoring

Feedback Loops

Systems that learn about behaviour and adjust their responses in (near) real-time can also create feedback loops.

For example, social media recommendations that show users progressively more extreme content.



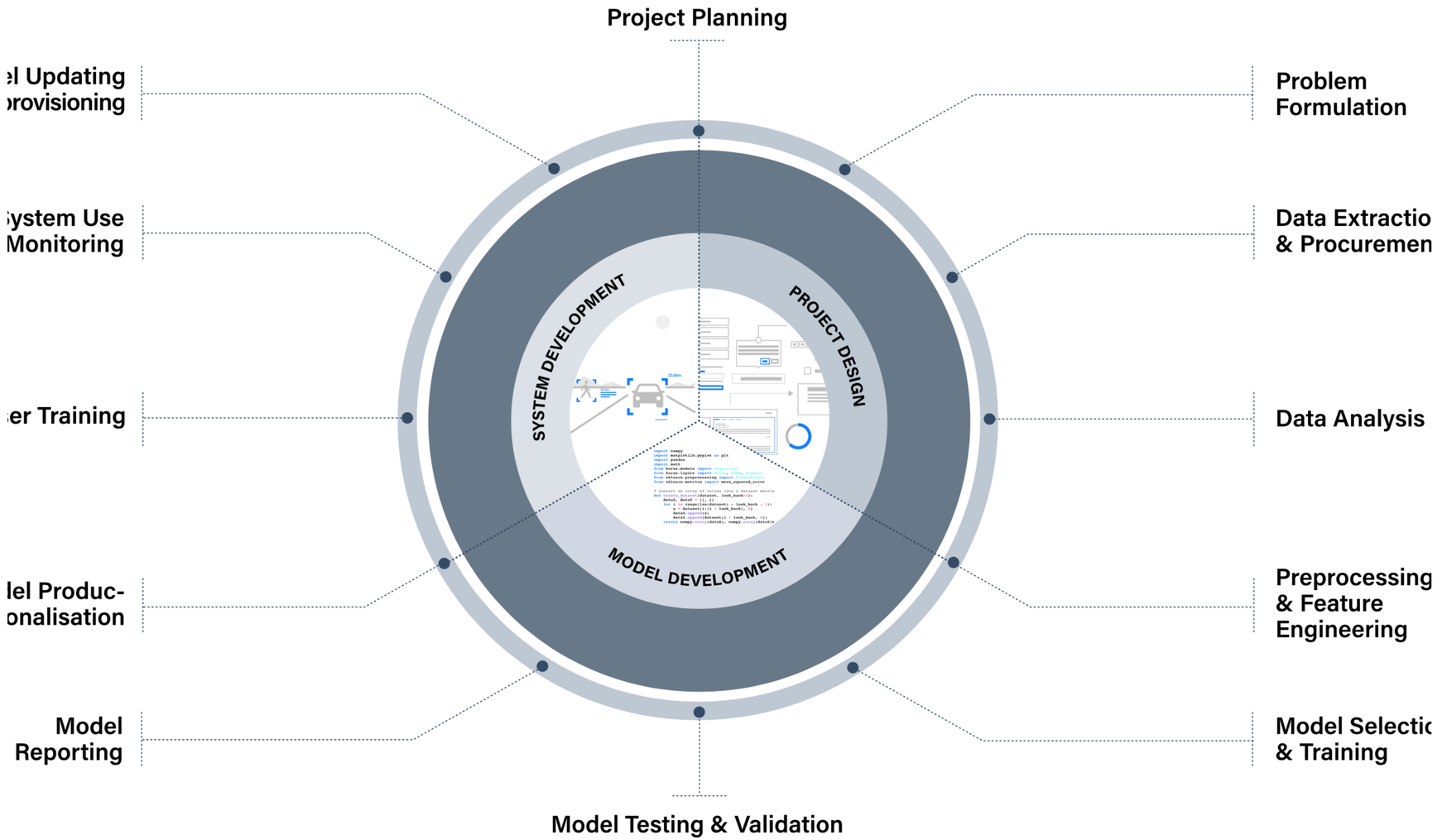
(System) Deployment

Model Updating or Deprovisioning

When should a model be updated, and when should it be deprovisioned?

What metrics or performance indicators can be monitored at the previous stage that can inform this decision?

Risk of creating legacy systems if unable to update or deprovision, but what if the model is business critical?



Full Circle

Next Steps

How do you **responsibly communicate** the actions and decisions undertaken throughout the project lifecycle stages to an audience with diverse needs and expectations?

Q&A

Day 5

Tomorrow

- What is Argument-Based Assurance?
- Assurance and Responsible Communication
- Goals, Properties, and Evidence
- Wrap-Up

Thank you!

See you tomorrow for the final

The Alan Turing Institute

Dr Christopher Burr
cburr@turing.ac.uk
@chrisdburr