University of Exeter

Introduction to Machine Learning

Coding for Reproducible Research

March 2025

Collaborative doc: https://tinyurl.com/fd3cd22b

Course Leader:

- Simon Kirby

Course Helpers:

- Finley Gibson
- Sam Fletcher

Sign in here →

Code of Conduct

- Our ethos is to provide a welcoming and supportive environment for all people, regardless of background or identity. By registering to attend this workshop, participants are agreeing to abide by the Researcher Development Code of Conduct.

- Our goal is to support you to develop your programming skill sets to enable you to do cutting edge research. We want to create a positive and professional learning environment and therefore encourage the following kinds of behaviours:

  - Show courtesy and respect towards all who attend a workshop or engage in community events.
  - Be respectful of different viewpoints and experiences.
  - Gracefully accept constructive criticism.
  - Be patient if there are technical glitches. While we know something about how to use computers, we are not immune to internet or hardware issues.
  - Respect our policy on not recording workshops to protect the nature of the sessions and ensure we are GDPR compliant.

Programme Funding

The CfRR training programme is supported by:

- Research Software Analytics Group
- Institute for Data Science and Artificial Intelligence (IDSAI)
- University of Exeter Reproducibility Leadership Team
- EPSRC Research Software Engineering Fellowship
- Community of academics who volunteer their time to support delivery

To make the case for continued investment, please help us demonstrate the impact of these sessions by attending all courses you register for and providing feedback at the end of the course.

University of Exeter

Intro to Machine Learning

Part 3 - The machine learning pipeline

Course contents

Session 1

- Slides: what is machine learning?
- Tutorial: linear regression
- Slides: model selection and evaluation

Session 2

- Tutorial: model selection and evaluation
- **Slides: the machine learning pipeline**
- Tutorial: machine learning pipeline task

Session 3

- Continue with machine learning pipeline task
- Tutorial: unsupervised learning

The machine learning pipeline

What is it?

- Treating machine learning problems as a pipeline of discrete tasks

Why bother?

- Improves reliability, repeatability and reproducibility of research, by…
    - Making processing, training and analysis steps very clear
    - Enabling testing of modular components
    - Encouraging modularisation and testing of code
    - Encouraging model versioning and tracking
- Leads the way to automation, deployment, and MLOps

The machine learning pipeline

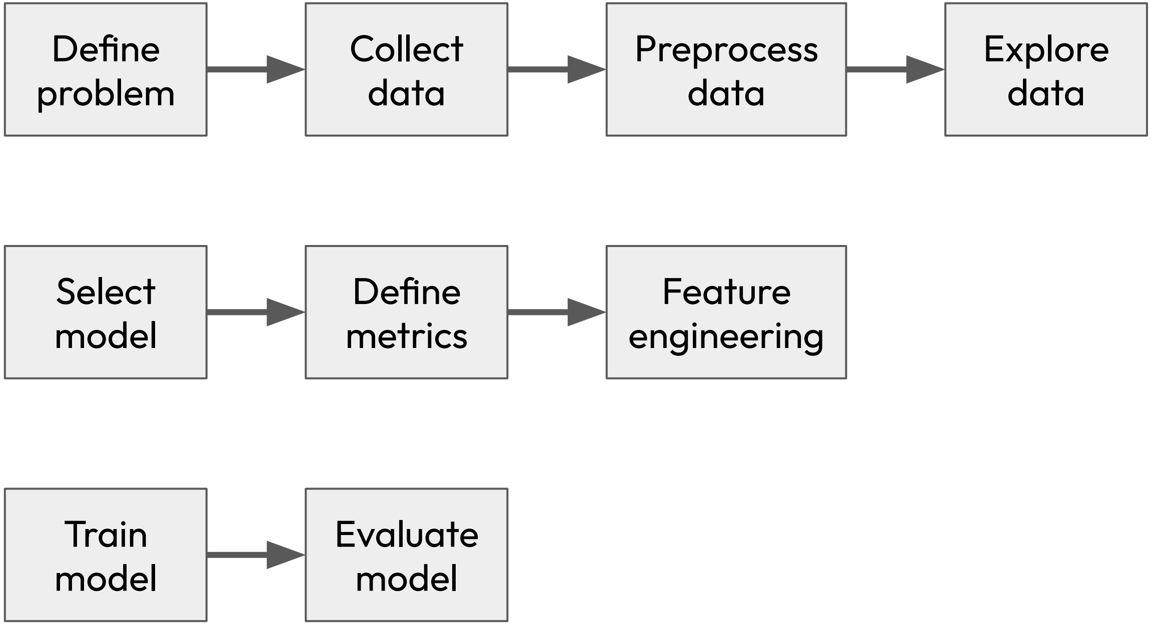
When to use it?

- On real problems, where you do not want to make mistakes

When not to use it?

- Potentially when you are playing around with a new technique or dataset
- However the pipeline stages are still useful to think about!

The machine learning pipeline

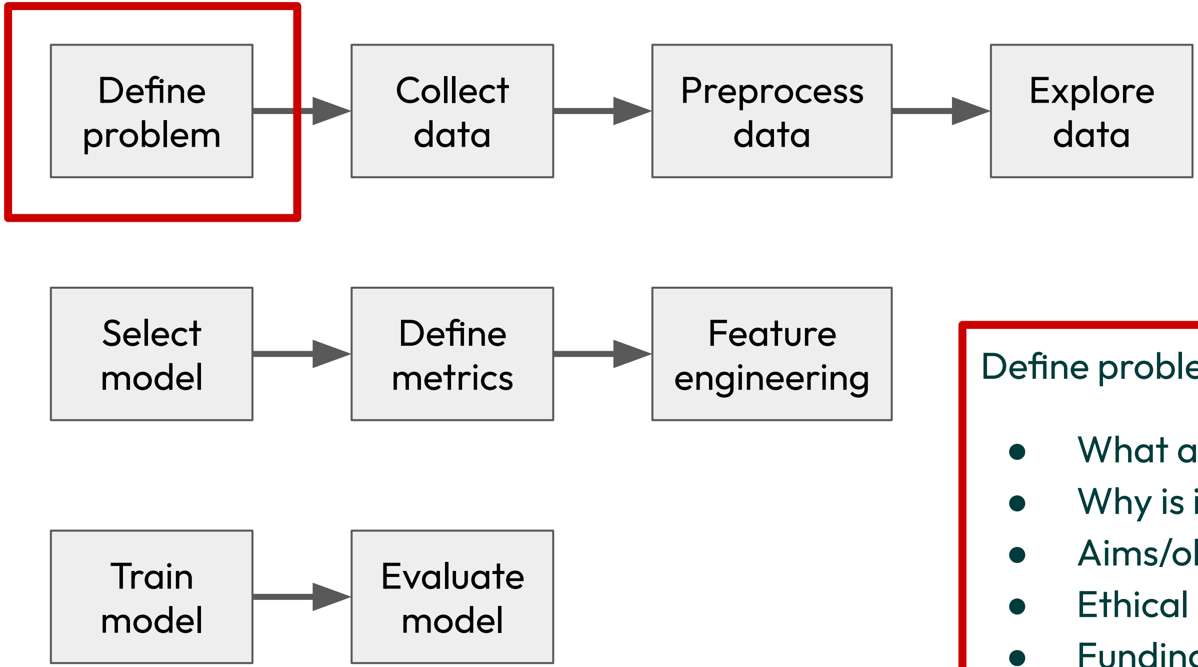


Define problem → Collect data → Preprocess data → Explore data

Select model → Define metrics → Feature engineering

Train model → Evaluate model

The machine learning pipeline

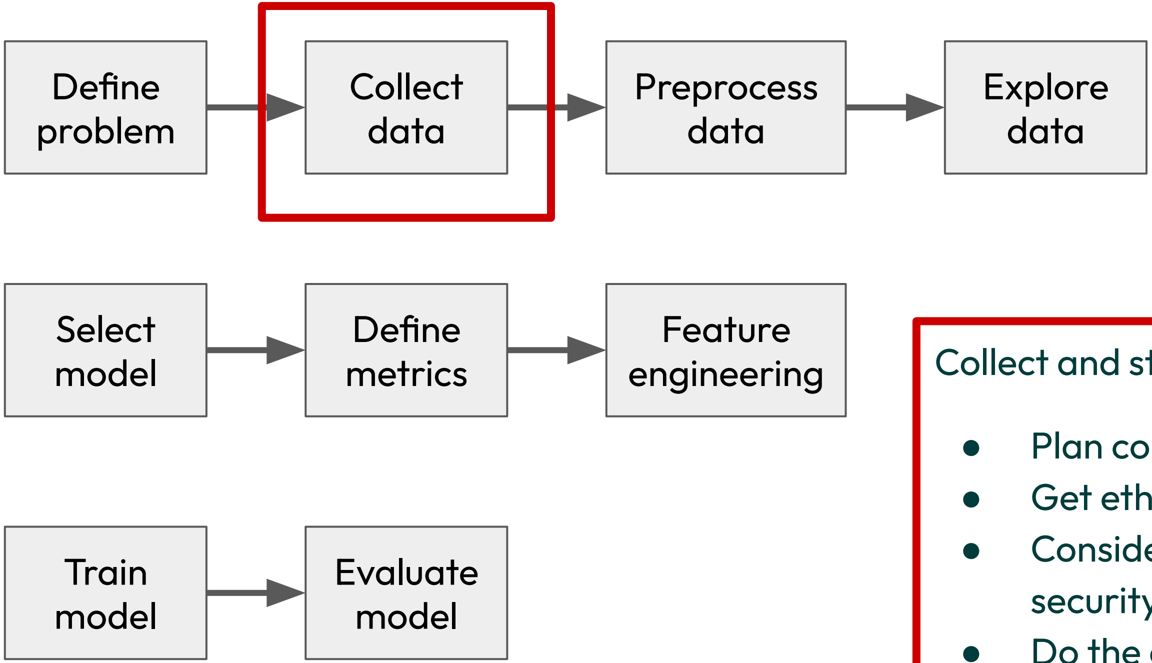


Define problem

- What are we doing?
- Why is it useful?
- Aims/objectives
- Ethical considerations
- Funding, business considerations

The machine learning pipeline

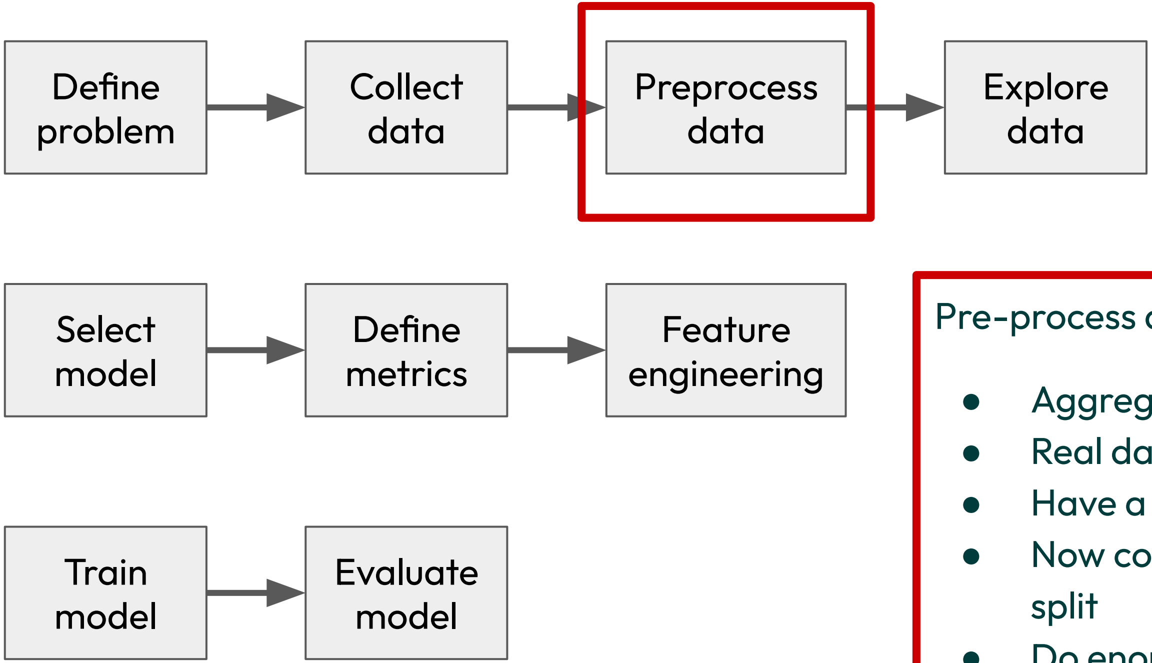
University of Exeter



Collect and store data

- Plan collection details
- Get ethics approval
- Consider data management, storage, security, etc
- Do the collection, store, back it up

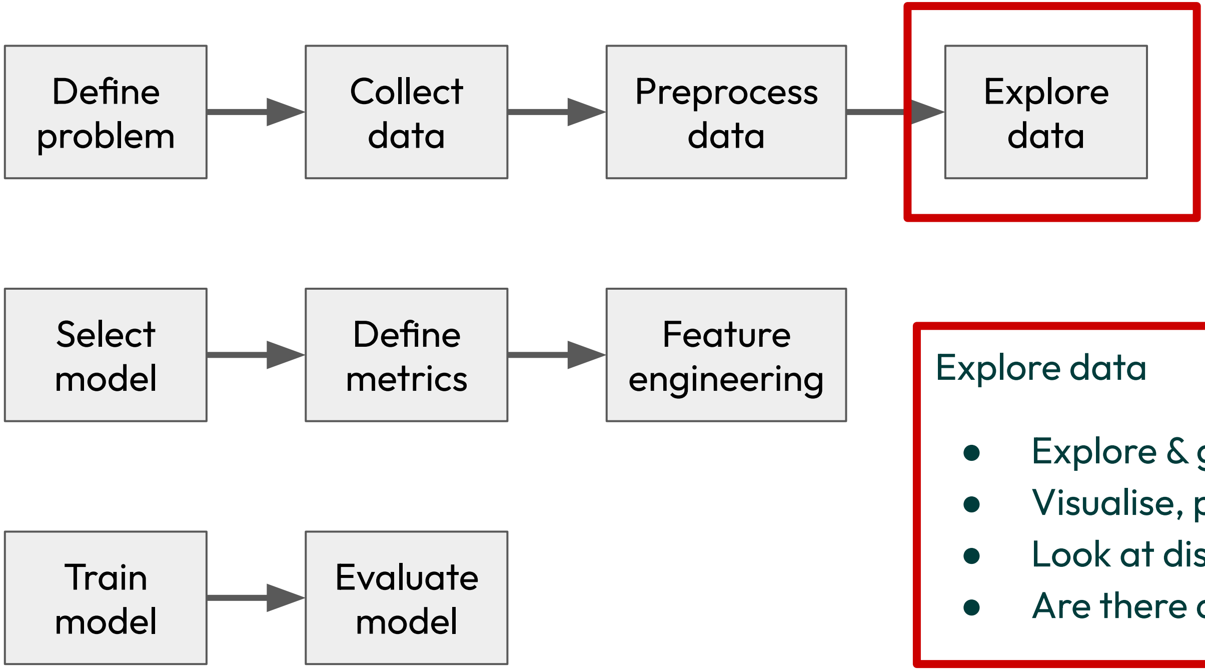
The machine learning pipeline



Pre-process data

- Aggregate files
- Real data is messy: it needs cleaning
- Have a very initial look at the data
- Now could be a good time for train-test split
- Do enough to get insights out the next stage

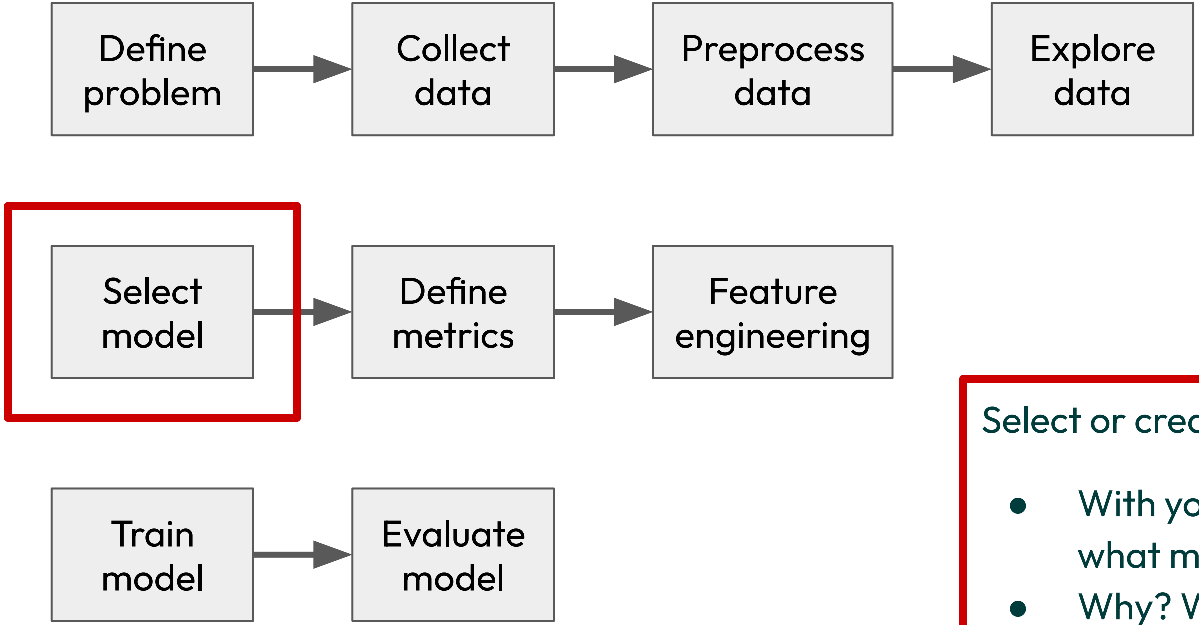
The machine learning pipeline



Explore data

- Explore & get to know your data
- Visualise, plot, transform, etc
- Look at distributions, bias
- Are there any obvious trends?

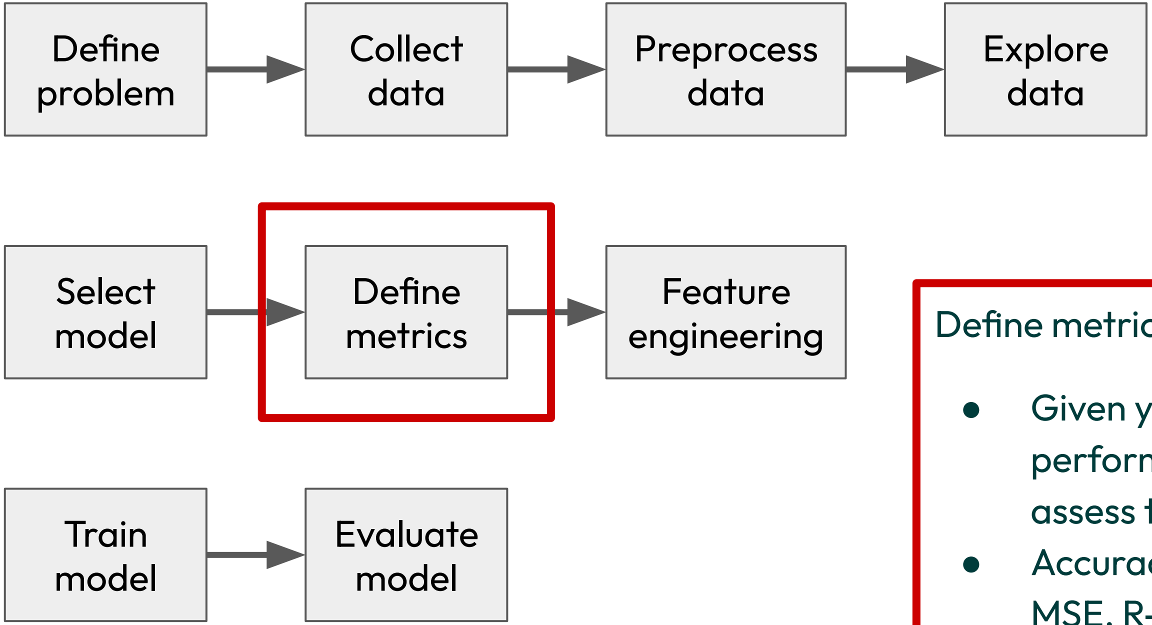
The machine learning pipeline



Select or create model

- With your new insights about your data, what model can you select to start?
- Why? What reasoning?

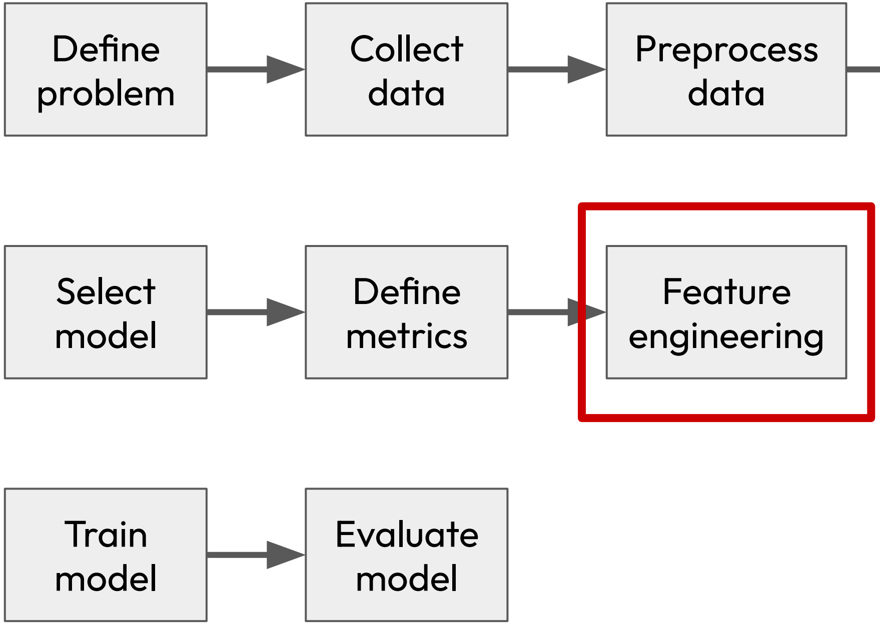
The machine learning pipeline



Define metrics and plan validation

- Given your initial chosen model, what performance metrics will you select to assess this model?
- Accuracy, f1-score, recall, precision, RMSE, MSE, R-squared etc
- Plan validation strategy here: i.e. cross validation, hyperparameter tuning

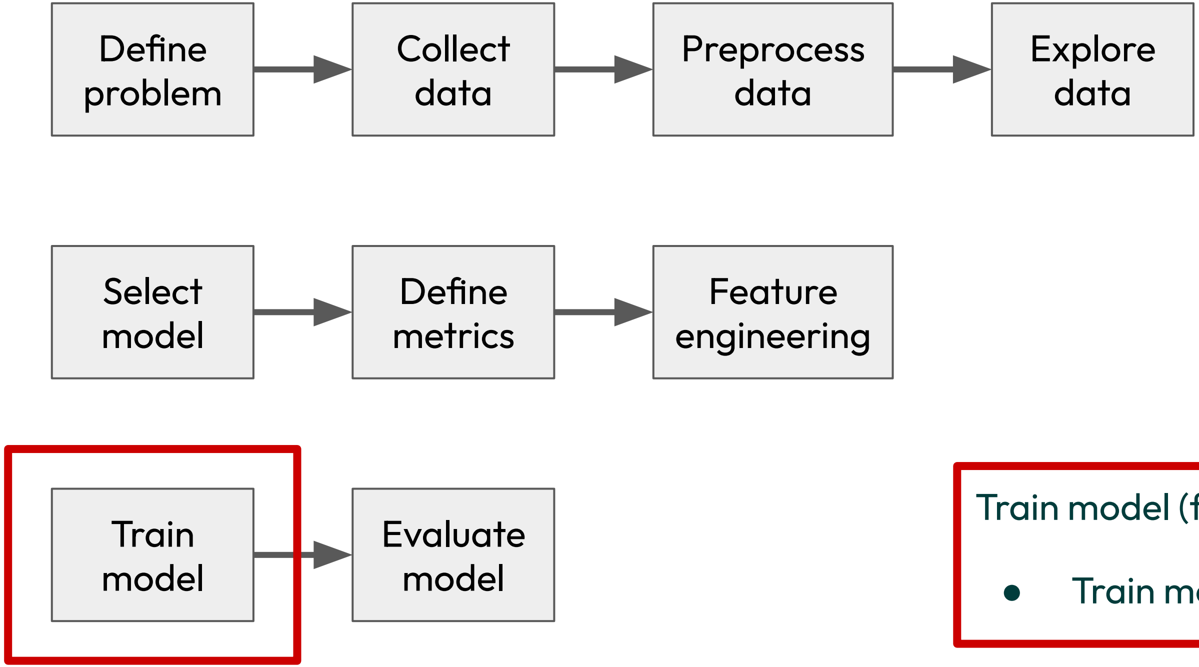
The machine learning pipeline



Engineer features

- Ensure data is encoded correctly for models.
- Things like one-hot encoding, scaling, normalisation, outliers, vectorisation, tokenisation, image processing, etc.
- Basically get the data ready for training.

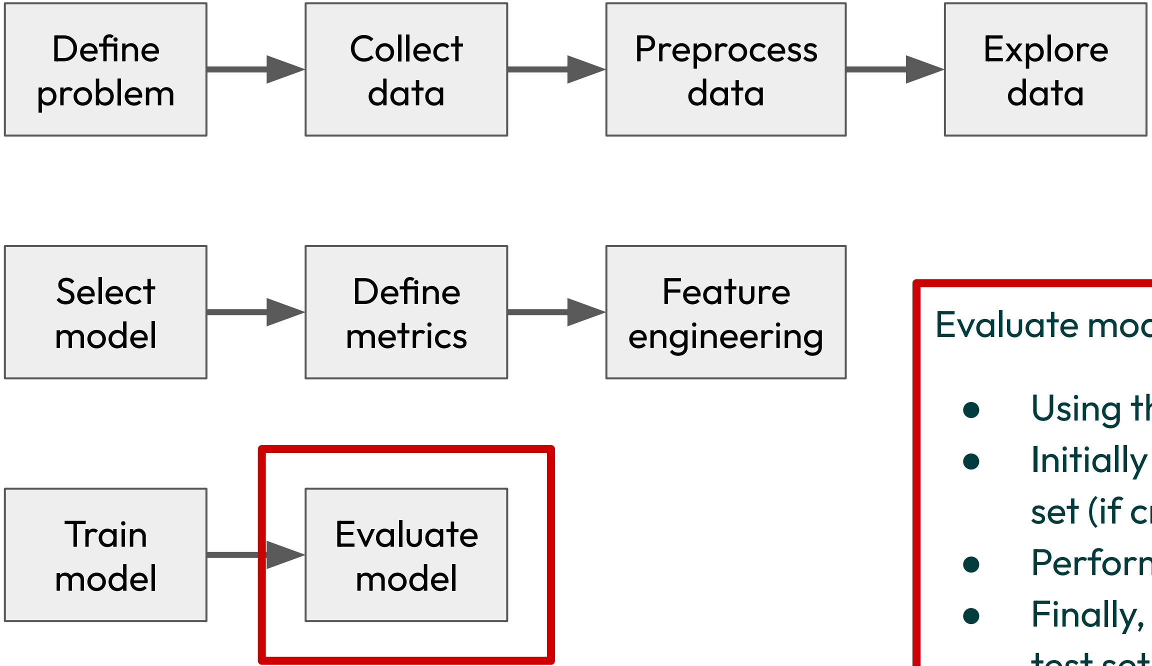
The machine learning pipeline



Train model (finally!)

- Train model, saving as you go

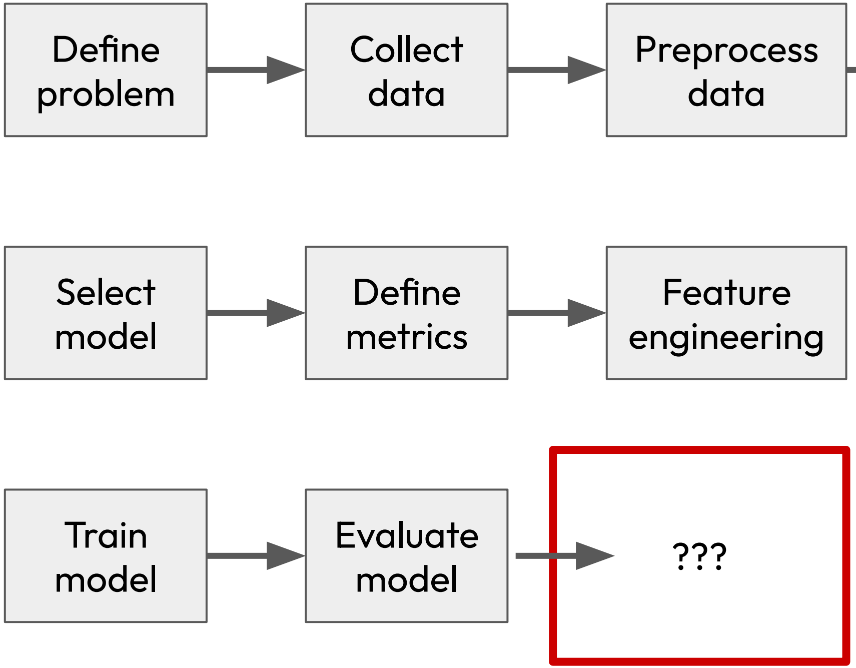
The machine learning pipeline



Evaluate model performance

- Using the metrics decided earlier
- Initially validate on your held out validation set (if cross validating)
- Perform fine tuning
- Finally, evaluate the model on the held out test set.

The machine learning pipeline



Define problem

Collect data

Preprocess data

Explore data

Select model

Define metrics

Feature engineering

Train model

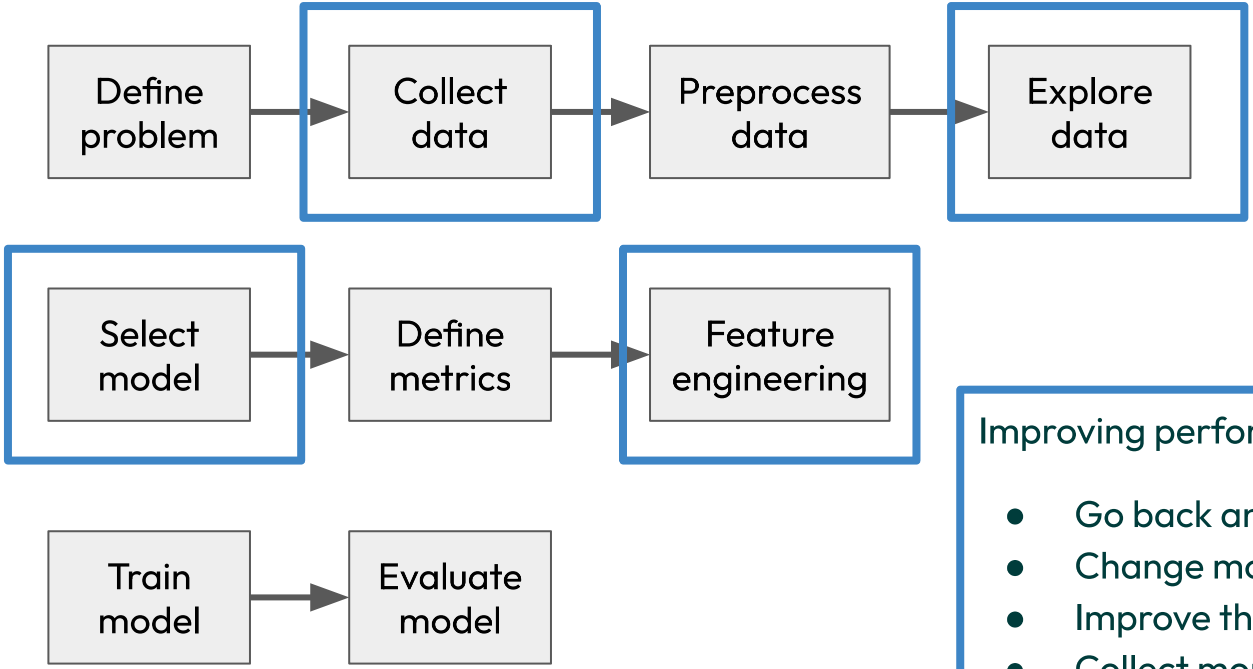
Evaluate model

???

What next?

- Is performance good enough?
- Yes: launch model, product, research
- No: what can we do to improve performance?

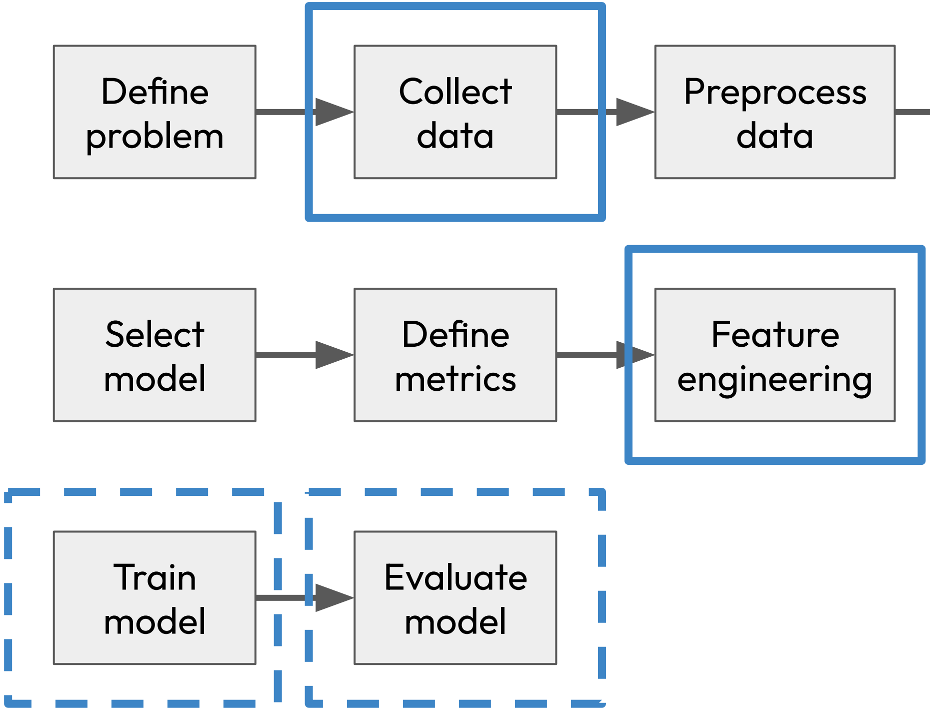
Improving model performance



Improving performance (selected blocks)

- Go back and explore the raw data further
- Change model/approach
- Improve the feature engineering
- Collect more data

Where is most effort spent?



High effort:

- Data collection, feature engineering
- Maybe evaluation/fine tuning

Then less effort:

- Data exploration/understanding problem
- Model training might also be hard, if high compute or RAM requirements.

Thank you!

Post-workshop Anonymous Feedback Form 2024-25
https://tinyurl.com/2d8fys7e