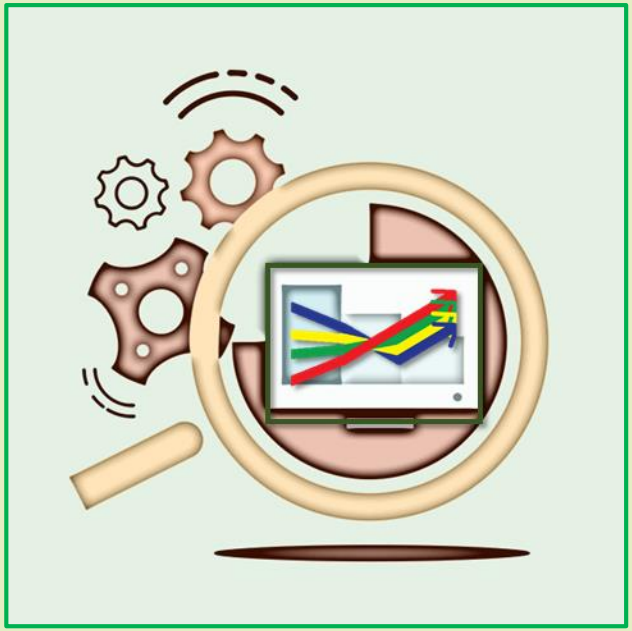
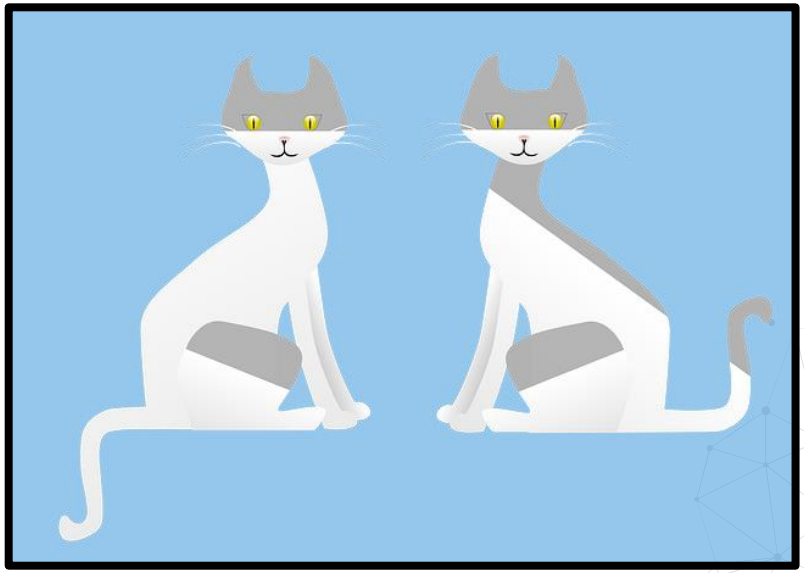
Train In Data



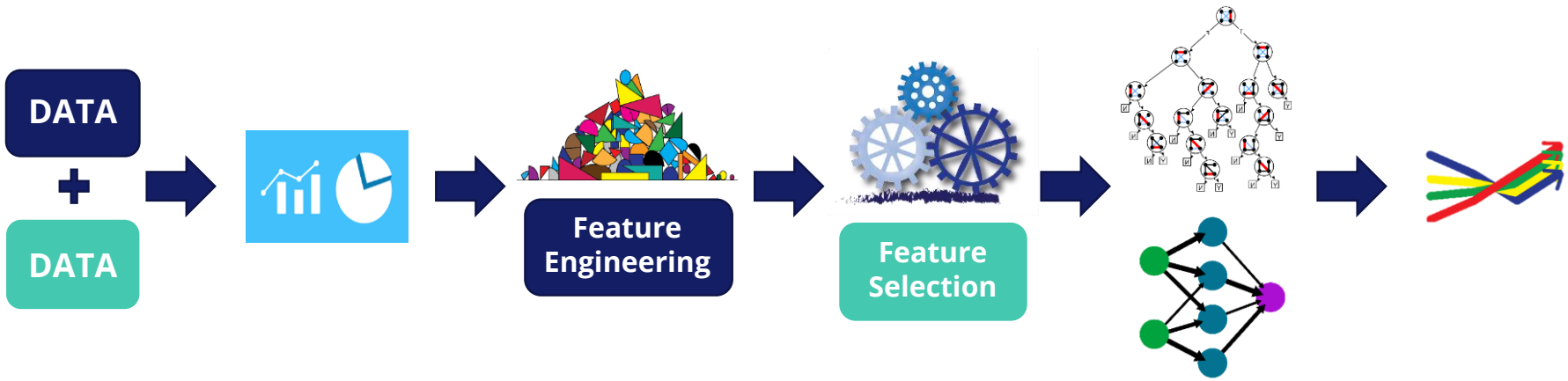
Challenges to Reproducibility

Reproducibility in Machine Learning

Reproducibility is the ability to duplicate a machine learning model exactly, such that given the same raw data as input, both models return the same output.



Machine Learning Pipeline



Gathering Data Sources

Data Analysis

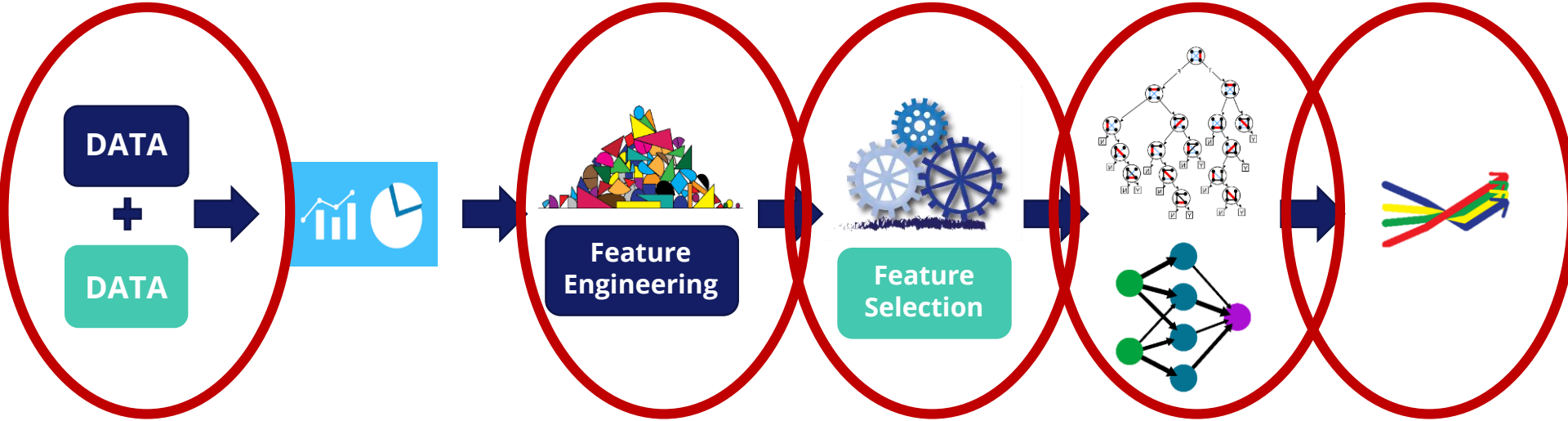
Data Pre-processing

Variable selection

Machine Learning Model building

Model deployment

Machine Learning Pipeline



Gathering Data Sources

Data Analysis

Data Pre-processing

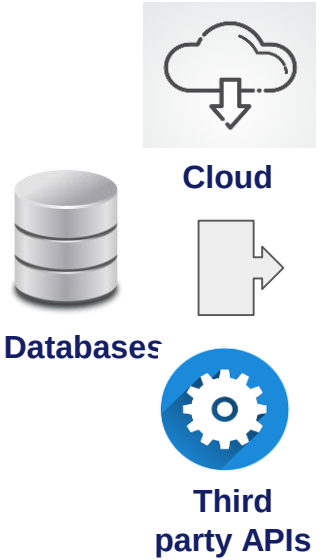
Variable selection

Machine Learning Model building

Model deployment

Reproducibility during Data Gathering

Data ➡ **The most difficult challenge to reproducibility**



Challenges

- ❖ Training dataset can't be reproduced
- ➢ Databases are constantly updated and overwritten.
- ➢ Order of data while loading is random (SQL).

Solutions

- ❖ Save a snapshot of training data
  - ✓ Simple
  - ○ Potential conflict with GDPR
  - o Not suitable for big data
- ❖ Design data sources with accurate timestamps.
  - ✓ Ideal situation
  - ○ Big effort to (re)design the data sources

Reproducibility during Feature Creation

Feature Engineering

Data Pre-processing

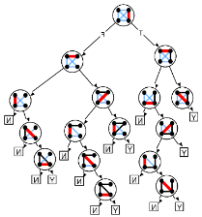
Challenges

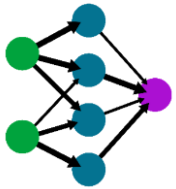
- Replacing missing data with random extracted values
- Removing labels based on percentages of observations
- Calculating statistical values like the mean to use for missing value replacement
- More complex equations to extract features, e.g., aggregating over time

Solutions

- Code on how a feature is generated should be tracked under version control.
- Many of the parameters extracted for feature engineering depend on the data used for training ➔ ensure data is reproducible
- If replacing by extracting random samples, always set a seed

Reproducibility during Model Training





Challenges

- Machine learning models rely on randomness for training
  - Data and feature extraction for trees
  - Weight initialisation for neural nets, etc.
- Machine Learning model implementations work with arrays agnostic to feature names
  - Need to be careful to feed data in the correct order

Solutions

- Record the order of the features
- Record applied feature transformations
- Record hyperparameters
- For models that require randomness always set a seed.
- If the final model is a stack of models, record the structure of the ensemble.

Reproducibility during Model Deployment:

Model Deployment

Challenges

- A feature is not available in the live environment
- Different programming languages
- Different software
- Live populations don't match those used for training

Solutions

- Software versions should match exactly - applications should list all third party library dependencies and their versions
- Use a container and track software specifications
- Research, develop and deploy utilising the same language, e.g., python
- Prior to building the model, understand how the model will be integrated with other systems