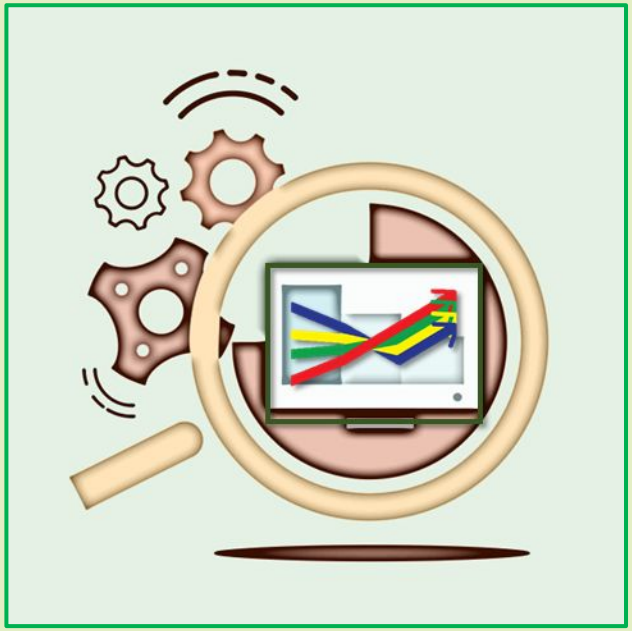
Train In Data



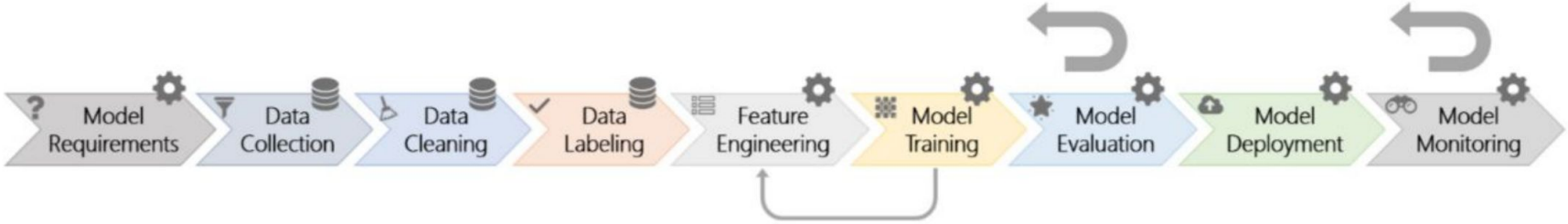
Key Principles for ML Systems

Useful Resources

1. "The ML Test Score: A Rubric for ML Production Readiness and Technical Debt Reduction" (2017) Breck *et al*. *IEEE International Conference on Big Data* (Google)

2. "Software Engineering for Machine Learning: A Case Study" (2019) Amershi *et al*. (Microsoft)

End-to-End Pipelines

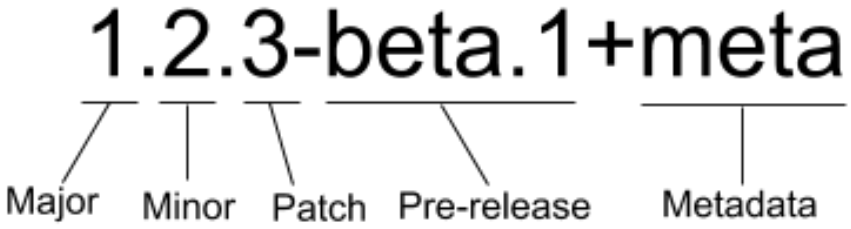


- Automation of all stages of the ML workflow

Reproducibility

- Training is reproducible
- Every model specification undergoes a code review and is checked into a repository
- Models can be quickly and safely rolled back to a previous serving version

Versioning



- Each model is tagged with a provenance tag that explains with which data it has been trained on and which version of the model
- Each dataset is tagged with information about where it originated from and which version of the code was used to extract it (and any related features).

Testing

- The full ML pipeline is integration tested
- All input feature code is tested
- Model specification code is unit tested
- Model quality is validated before attempting to serve it

Infrastructure

- Models are tested via a shadow and/or canary process before they enter production serving environments
- Monitor the model performance

Run Through the Checklist

It's also useful to observe the links and dependencies across different best practices:

- Without model specification review and version control, it would be hard for reproducible training
- Without reproducible training, the effectiveness and predictability of canary releases are significantly reduced.
- Without knowing the impact of model staleness, it's hard to implement effective monitoring