Data Science in production

Lecture 5: Model monitoring and retraining

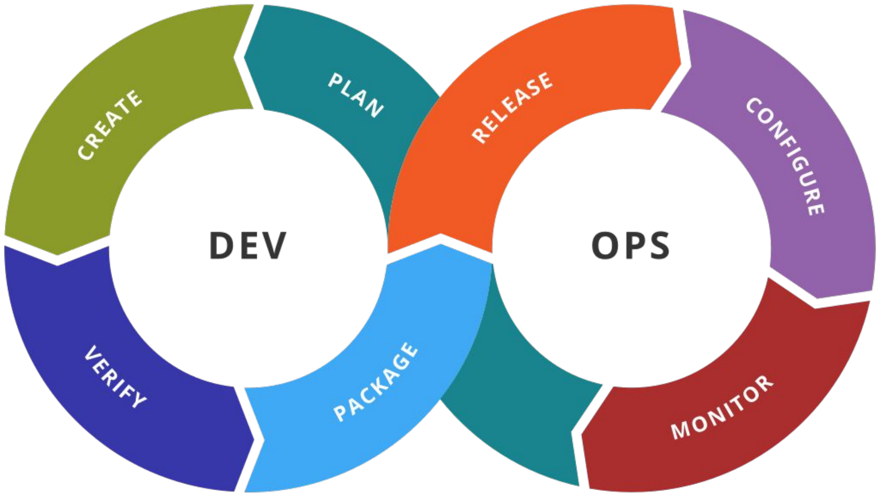
Alaa BAKHTI

Motivation

"The real problems with a ML system will be found while you are continuously operating it for the long term"

DevOps

“DevOps is a set of practices that works to automate and integrate the processes between software development and IT teams, so they can build, test, and release software faster and more reliably.” - Atlassian



source : Wikipedia

MLOps

- Why?
  - Isolation between ML and IT teams
  - ML teams not thinking about production challenges of the models they are producing (training-serving skew, model analysis, etc)

"*An ML engineering culture and practice that aims at unifying ML system development (Dev) and ML system operation (Ops)*" - Google Cloud

- Application of DevOps principles to ML systems (MLOps)
- How to operationalize an ML model?
- How to deploy the model? How to monitor it?

Monitoring

Your model's accuracy will be at its best until you start using it. It then deteriorates as the world it was trained to predict changes - Forbes

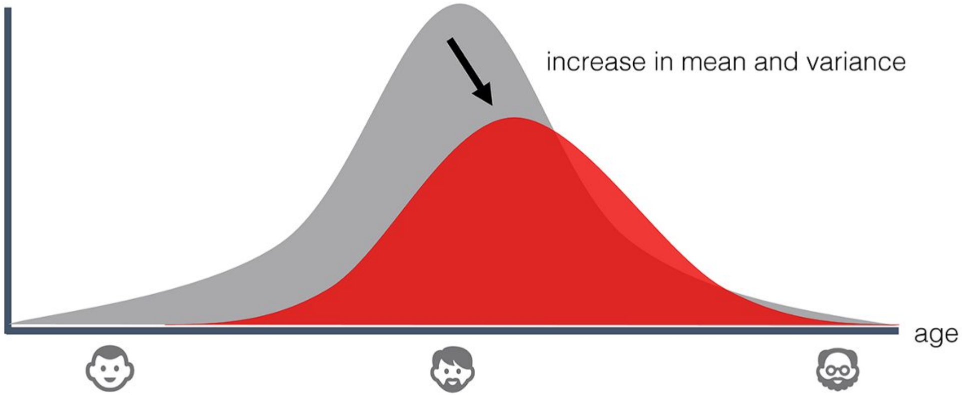
Model drift

- Streaming data
  - Recommender systems, pandemics
  - Credit prediction, economic crisis
- After deployment, the models will start degrading in performance (or not?) because the data is changing over time
- They will get stale (lose freshness) over time because they were trained on past data: Model drift
  - => need to define freshness requirements for the training data
- What are the different types of model drift?

Data drift

- Change in the input data distribution
- The statistical properties of the input data features (eg: age) used to train the production model change
- Possible causes: seasonality, trends, etc
- Feature drift, covariate shift, etc

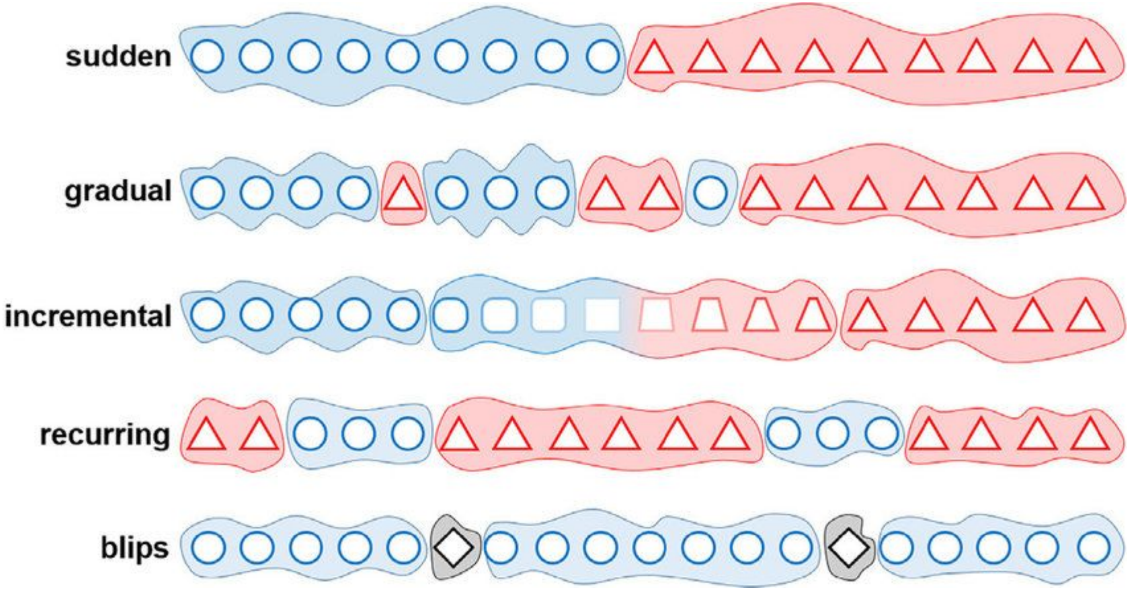
=> The model trained on the past data is no longer relevant on the new upcoming data



Sources: https://evidentlyai.com/blog/machine-learning-monitoring-data-and-concept-drift

Concept drift

- *“The statistical properties of the target variable, which the model is trying to predict, change over time in unforeseen ways”* - Wikipedia
- the pattern the model learned is no longer valid
- external factors that change the relationship between the features and the labels we want to predict
- Example: Fraud detection



Sources

Monitoring

What?

- Basic summary statistics of features and target
- Distributions of features and target
- Model performance metrics
- Business metrics

How?

- Versioning and logging (models, data, inference data, ...)
- Statistical tests
- dashboards (Grafana for example)

Some monitoring packages

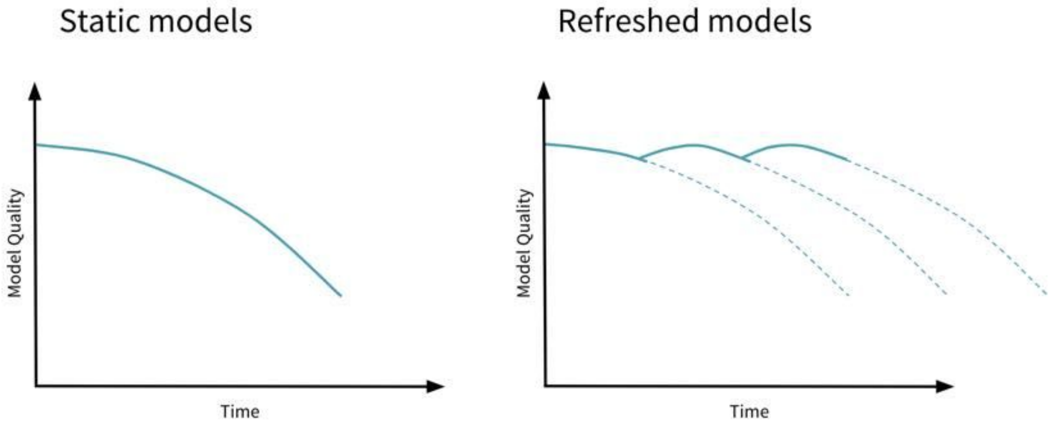
- [Evidently AI]
- [Data Drift Detector]
- [Alibi Detect]
- [scikit-multiflow]

Retraining

Retraining

Monitor the model performances, when it starts degrading, a retraining is needed (not always)

1. Collect fresh data
2. Label it if needed
3. Train a model on this data
4. Validate that it respects the defined requirements (performance, inference time, ...)
5. Compare it to the production model
6. Deploy it to production



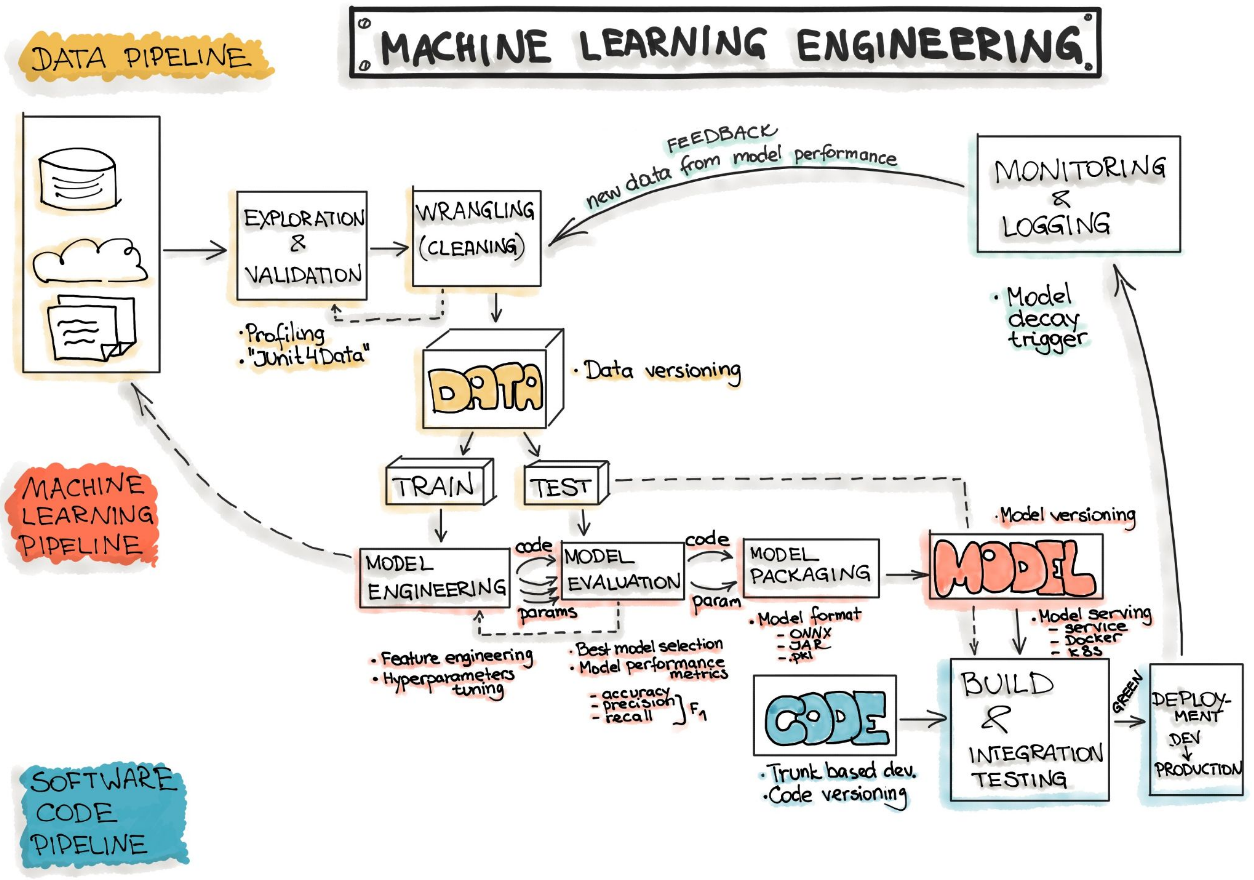
Source: Databricks blog

Model comparison

- Use a ground truth dataset composed of fresh data
- Compute the models performance on this dataset (the new model & the production model)

⚠️ Never compare the performances of 2 models if these performances:

- Were not computed on the same dataset
  - Model A evaluated on February data
  - Model B evaluated on March data
- If some of the dataset records were used in the training set of one of the 2 models.
  - Train a model on the past month data and compute its performance using last week's data



Retraining pipeline

Source: https://ml-ops.org/content/end-to-end-ml-workflow

Retraining strategies

Basic

- Retrain the model each period of time (1 month? 2 months?) on the new data
- Determine this period by
    - Asking your subject matter expert: how much time needed for the data to become non representative?
    - Doing some analysis on the past data

Advanced

- Trigger the model retraining when a drift is detected in the data

Validate the deployment of the newly trained model manually by your product owner (or not? use case specific)

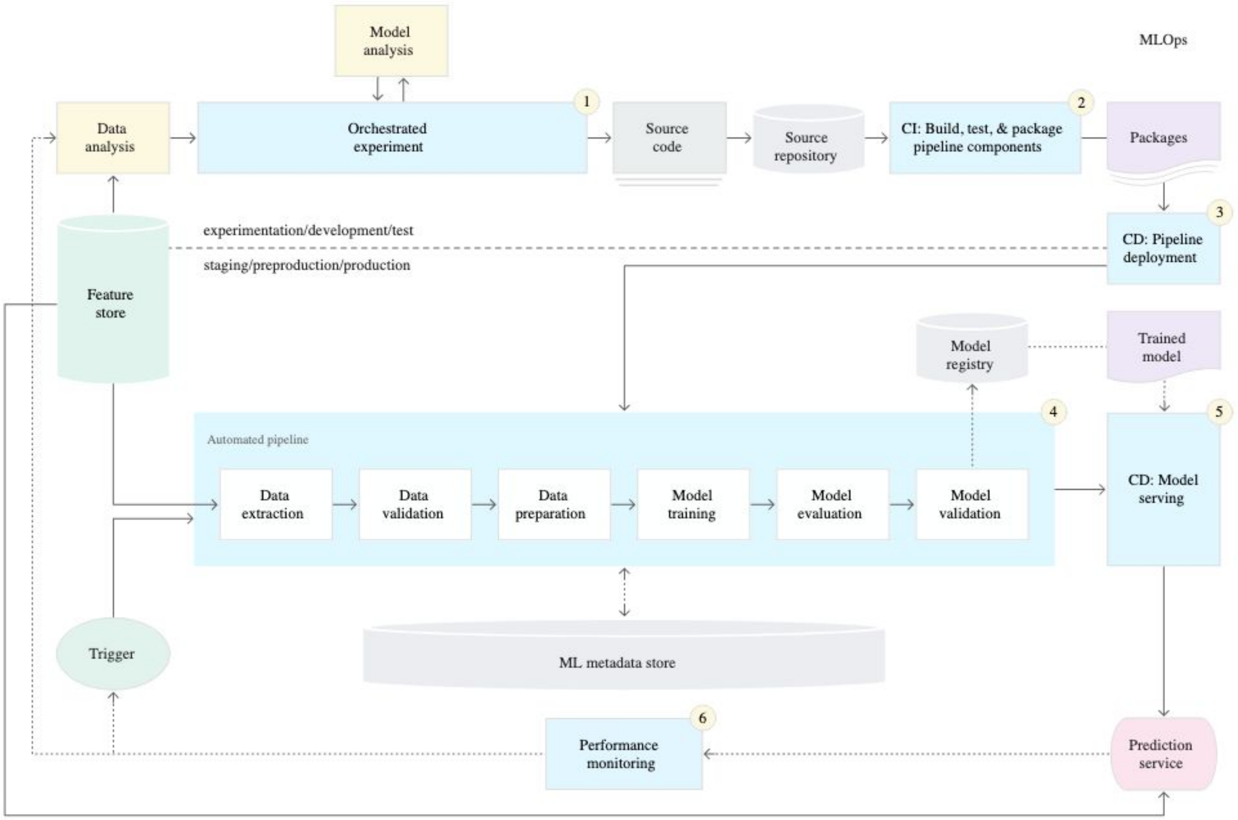
- Shadow production, ...

Model error analysis

- Identify the cause for the deviation (external data source problem, seasonality behavior (Black friday, New Year eve, etc))
- Understand why the model performance degraded

Model registry

- Centralized model repository to govern the lifecycle of ML models
    - Register, organize, track, and version trained and deployed models
    - Review, approve, release and rollback models
- In MLflow Model Registry, each model is characterized by
    - Name
    - Version: incremented each time the model is trained (if the model name is the same)
    - Stage: None, Staging, Production and Archived
- Models stages
    - Staging for model testing
    - Production for models that have completed the testing or review processes and have been deployed to applications
- The models will transition between different stages



CI/CD and automated ML pipeline - Google ([source]())

Ressources

- Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift
- Monitoring and explainability of models in production
- Challenges in Deploying Machine Learning: a Survey of Case Studies
- MLOps: Continuous delivery and automation pipelines in machine learning

Paper presentations

- **Google Developers Rules of Machine Learning: | ML Universal Guides**
- Continuous Delivery for Machine Learning (CD4ML)
- MLOps: Continuous delivery and automation pipelines in machine learning
- Paper: Hidden Technical Debt in Machine Learning Systems
- **Continuous Integration and Deployment for Machine Learning Online Serving and Models**