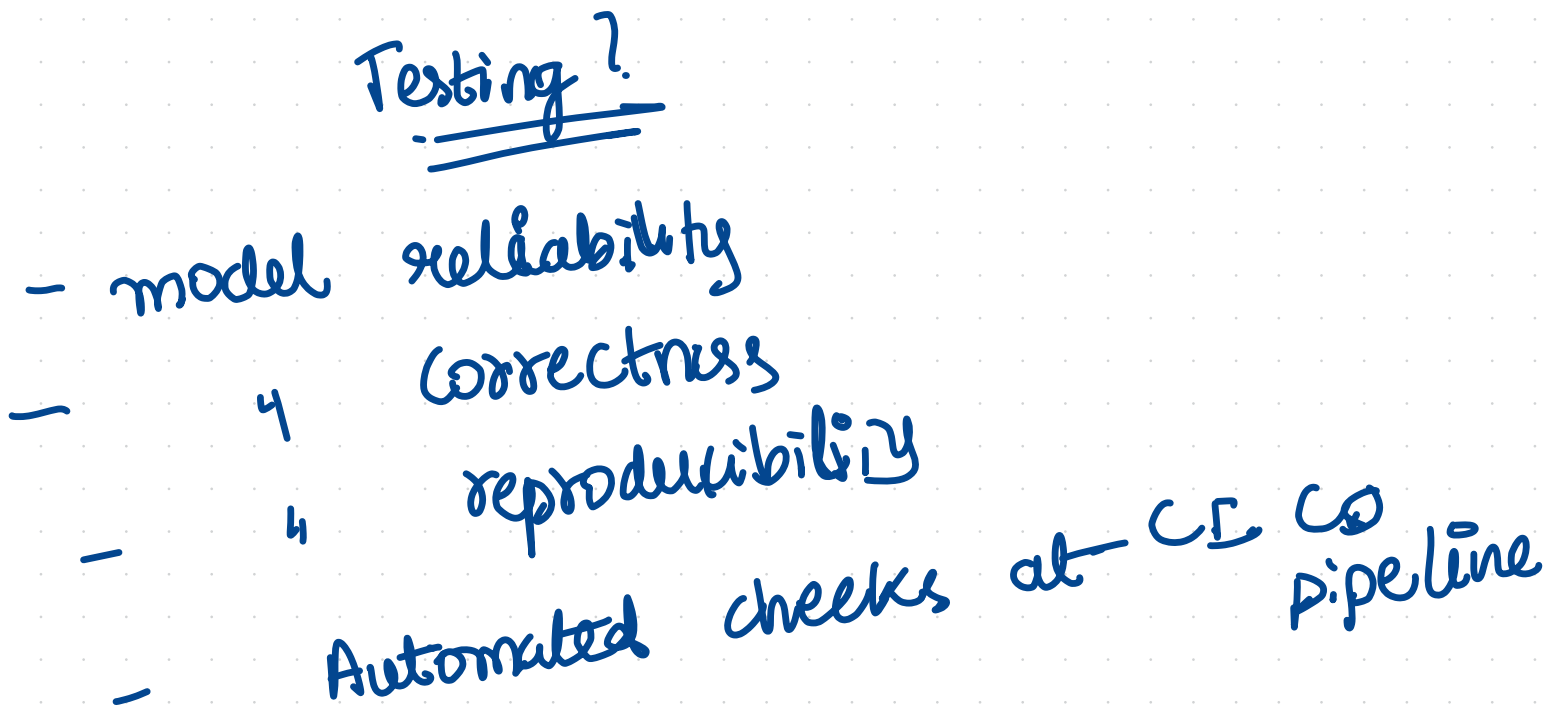
Testing in MLOps



Types of Testing in MLOps:

1. Unit Testing: *Test Individual functions . components (data processing, metrics, etc.)*
2. Integration Testing: *Test Interaction between the components (eg: pipeline stages)*
3. Data Validation: *Check for schema consistency, missing values, Anomalies in input data*
4. Model Testing: *Evaluate model accuracy, fairness, stability using test datasets*
5. Regression Testing: *Ensure changes dont break existing functionalities*
6. Performance Testing: *Test Response time, throughput under load, latency of the model*
7. Drift Detection: *Monitor the changes in data distribution over time (data/model drift)*

Tools Used:

1. Pytest, Unittest - Unit Testing & Integration Testing
2. Pandera, Pydantic - Data Validation
3. scikit-learn - Model Evaluation
4. EvidentlyAI - Drift Detection
5. MLFlow - for reproducibility & evaluation tracking

Pytest

Testing Framework for python to perform unit test, functionality test & implement automation as part of CI CD pipelines

Structure and Naming Conventions

Test File - starts with **test_** or ends with **_test.py**
Test Function - start with **test_**
Test Class - start with **Test** and has no **__init__**