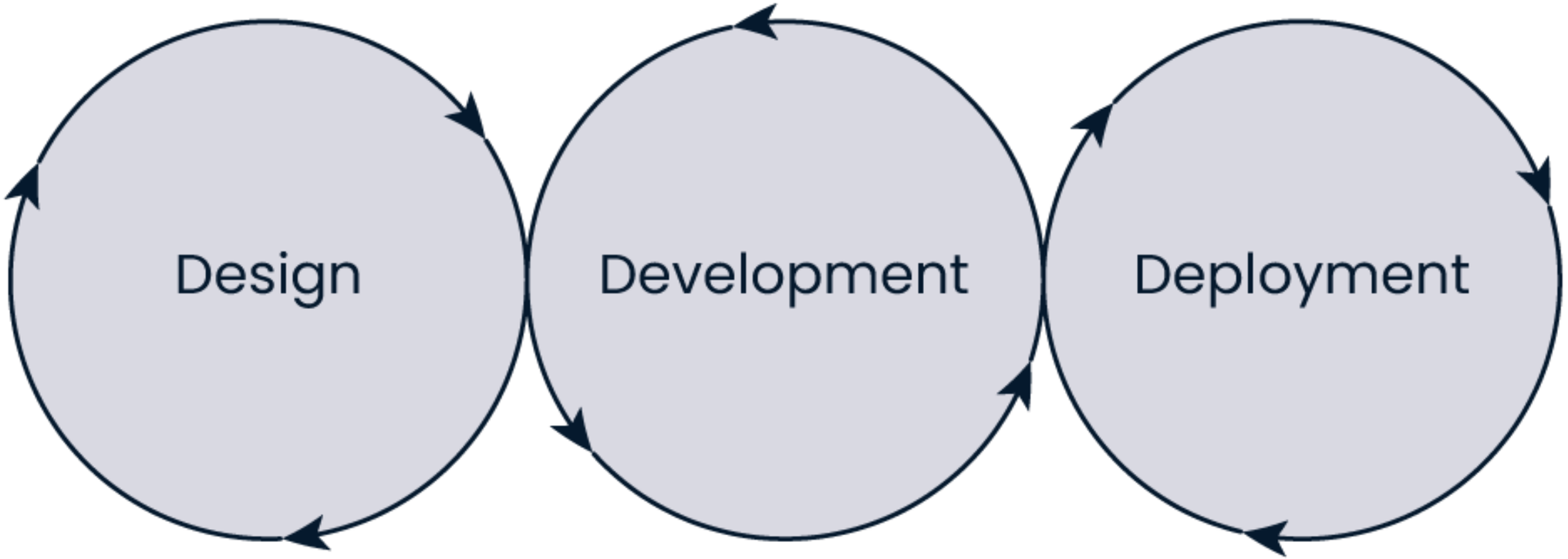
MLOps design

MLOPS CONCEPTS

Machine learning design



- Added value
- Business requirements
- Key metrics
- Data processing

- Feature engineering
- Experiment tracking
- Model training & evaluation

- Runtime environments
- Microservices architecture
- CI/CD pipeline
- Monitoring & retraining

Added value

- Estimate the expected value
- ML is experimental and uncertain
- Aids in resource allocation, prioritization, and setting expectations

Business requirements

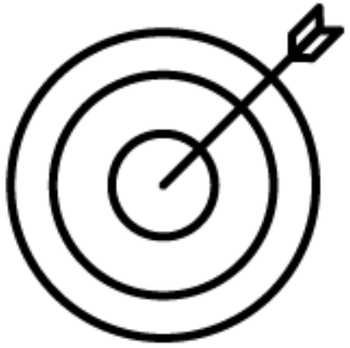
- End user
  - Speed
  - Accuracy
  - Transparency
- Compliance and regulations
- Budget
- Team size

Key metrics

Data scientist

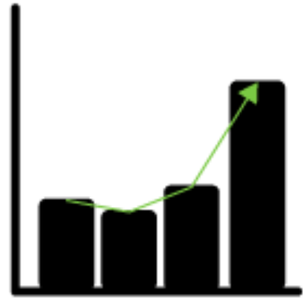
Subject matter expert

Business stakeholder



Accuracy

Customer happiness



Generated revenue

Data quality and ingestion

MLOPS CONCEPTS

Ingestion은 MLOps에서 **데이터를 외부 소스에서 수집하여 머신러닝 파이프라인으로 가져오는 과정**을 의미

이는 머신러닝 모델의 학습, 평가, 배포 및 운영을 지원하기 위한 첫 단계로, 데이터의 품질과 신뢰성이 전체 파이프라인 성능에 큰 영향을 미침

Ingestion의 주요 역할

1.데이터 수집:
    다양한 소스(데이터베이스, API, 파일, 센서 등)에서 데이터를 가져오는 역할.
    예: 로그 파일, IoT 장치에서 실시간 스트림 데이터, 웹 크롤러로 수집된 데이터 등.
2.데이터 준비:
    수집한 데이터를 머신러닝 모델에 적합한 형태로 정리.
    예: 포맷 변환, 중복 제거, 결측값 처리.
3.데이터 전달:
    수집된 데이터를 파이프라인의 다음 단계(전처리, 저장, 분석)로 전달.

Ingestion의 유형

1.배치 데이터 수집 (Batch Ingestion):
    정해진 간격으로 대량의 데이터를 한 번에 수집.
    예: 매일 새벽 1시에 로그 데이터를 데이터 웨어하우스로 로드.
2.스트림 데이터 수집 (Streaming Ingestion):
    실시간으로 데이터를 수집하고 처리.
    예: IoT 센서에서 발생하는 데이터를 실시간으로 수집.

Ingestion의 주요 구성 요소

1.데이터 소스:
데이터가 저장된 원천 (source)
예: 관계형 데이터베이스, NoSQL 데이터베이스, 클라우드 스토리지, API, 메시징 시스템(Kafka).

2.데이터 수집 도구:
데이터를 효율적으로 수집하기 위한 도구나 플랫폼.
예: Apache Kafka, Apache NiFi, AWS Kinesis, Google Cloud Pub/Sub.

3.ETL/ELT 프로세스:
ETL(Extract, Transform, Load): 데이터를 추출하고, 변환 후 저장.
ELT(Extract, Load, Transform): 데이터를 추출 후 저장한 뒤 변환.

4.데이터 품질 보장:
수집 중 데이터의 결측, 오류, 중복 등을 감지하고 처리.

5.데이터 스토리지:
수집된 데이터를 저장하는 위치.
예: 데이터웨어하우스, 데이터레이크.

Ingestion의 중요성

1.데이터 품질 확보:
깨끗하고 정확한 데이터를 확보해야 모델의 신뢰성과 성능을 보장.

2.파이프라인 자동화:
인제스천 단계에서 자동화를 통해 데이터 흐름을 효율화.

3.스케일 확장 가능성:
대규모 데이터와 실시간 데이터 수집을 지원.

4.다양한 데이터 형식 지원:
구조화 데이터(테이블 형태), 반구조화 데이터(JSON, XML), 비구조화 데이터(이미지, 비디오)를 모두 처리.

MLOps에서 Ingestion의 활용 사례

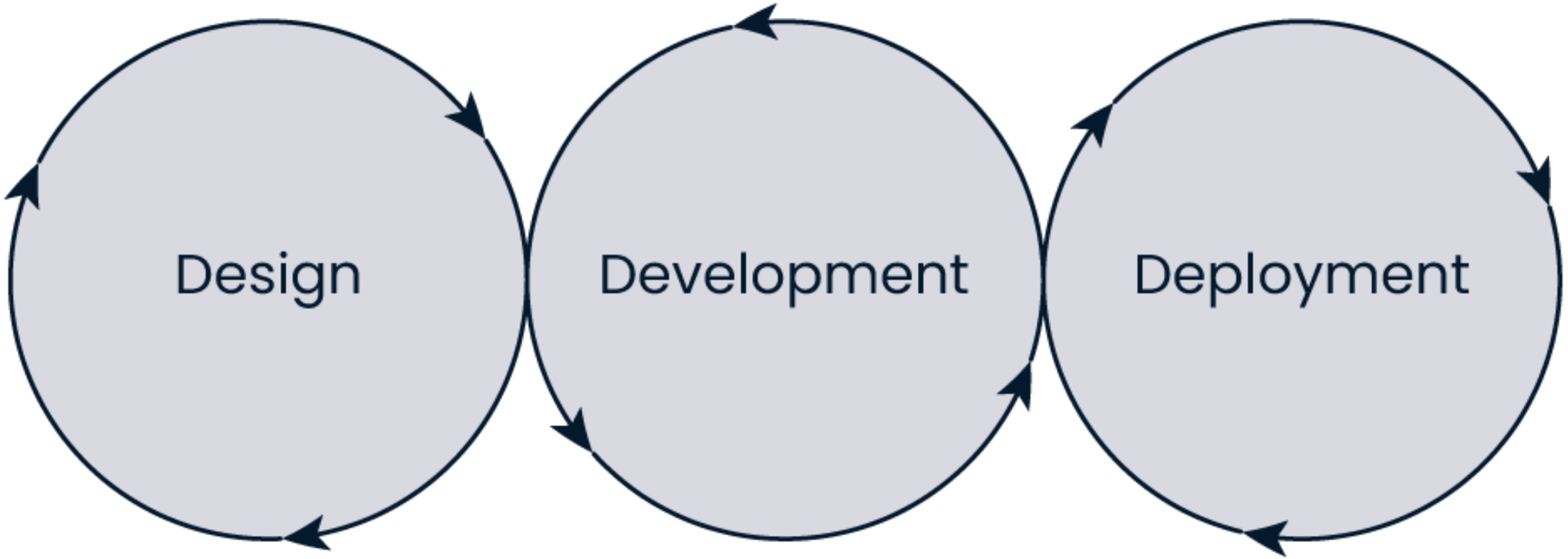
1.모델 학습:
대량의 과거 데이터를 배치 방식으로 수집하여 모델 학습에 사용.

2.모델 운영:
실시간 스트리밍 데이터를 사용하여 모델 입력 값 생성 및 예측.

3.데이터 모니터링:
운영 중 발생하는 데이터를 수집하여 모델 성능 모니터링 및 드리프트 감지.

Ingestion은 MLOps에서 머신러닝 파이프라인의 시작점이자 핵심 단계로, 모델의 신뢰성과 성능을 결정짓는 중요한 역할을 수행
적절한 데이터 수집 전략과 도구를 사용하면, 고품질 데이터를 효율적으로 관리하고 모델 운영을 안정적으로 지원할 수 있음

Data quality and ingestion



- Added value
- Business requirements
- Key metrics
- Data processing

- Feature engineering
- Experiment tracking
- Model training & evaluation

- Runtime environments
- Microservices architecture
- CI/CD pipeline
- Monitoring & retraining

What is data quality?

- Data quality is a measure of how well data serves its intended purpose
- Evaluated through various dimensions
- Quality of ML model depends on data

Data quality dimensions

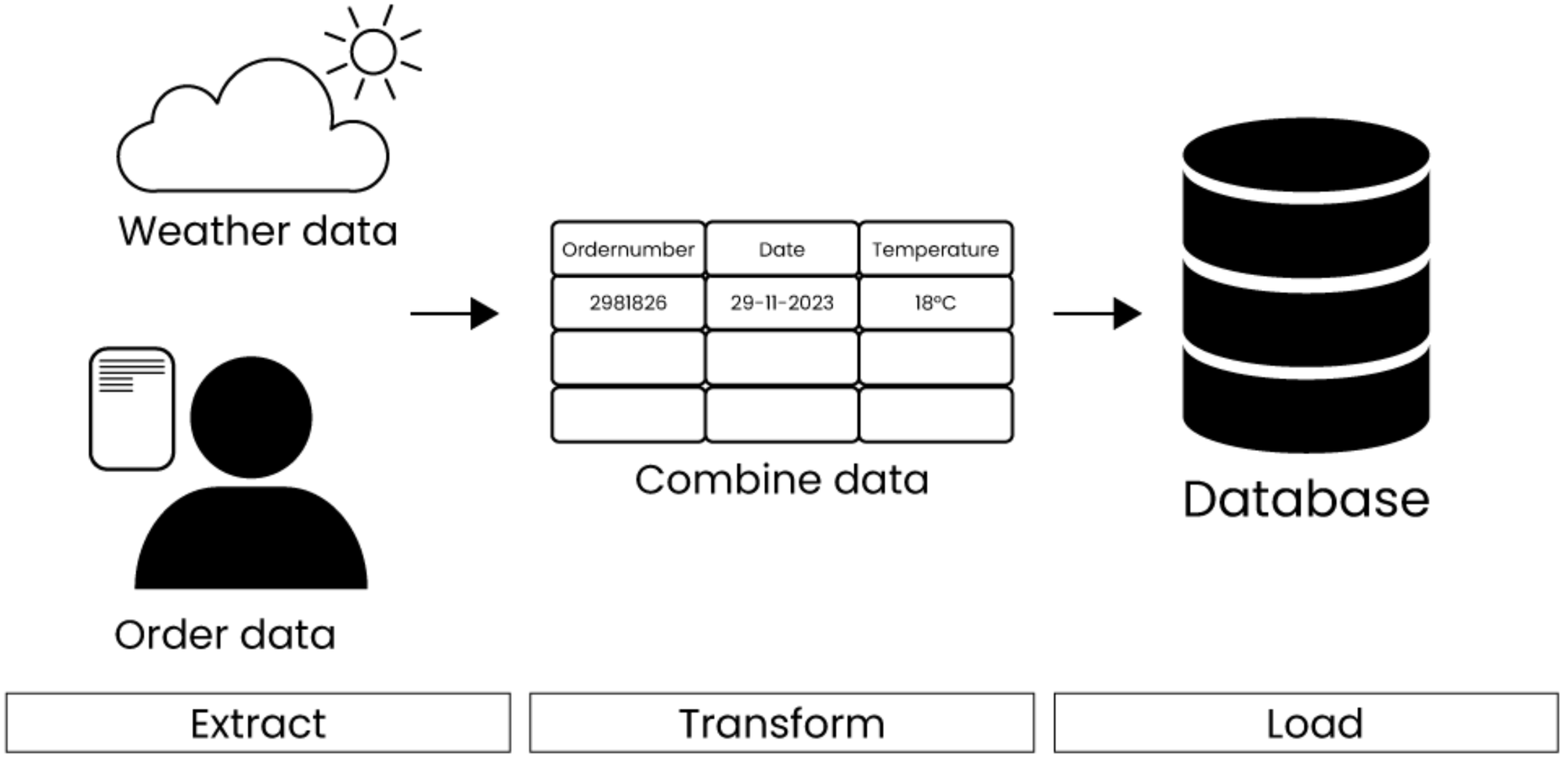
- **Accuracy**
- **Completeness**
- **Consistency**
- **Timeliness**

Data quality dimensions example

Dimension	Example question to answer	Example of dimension quality
Accuracy	Does our data correctly describe the customer?	The customer's age in the data is 18, but is actually 32.
Completeness	Is there any customer data missing?	For 80% of the customers, we don't have a last name.
Consistency	Is the definition of the customer synchronized throughout the company?	The customer is stated as active in one database but not active in another.
Timeliness	When is the customer ordering data available?	The customer orders are synchronized at the end of the day but are not available in real-time.

Low data quality is not the end of the project!

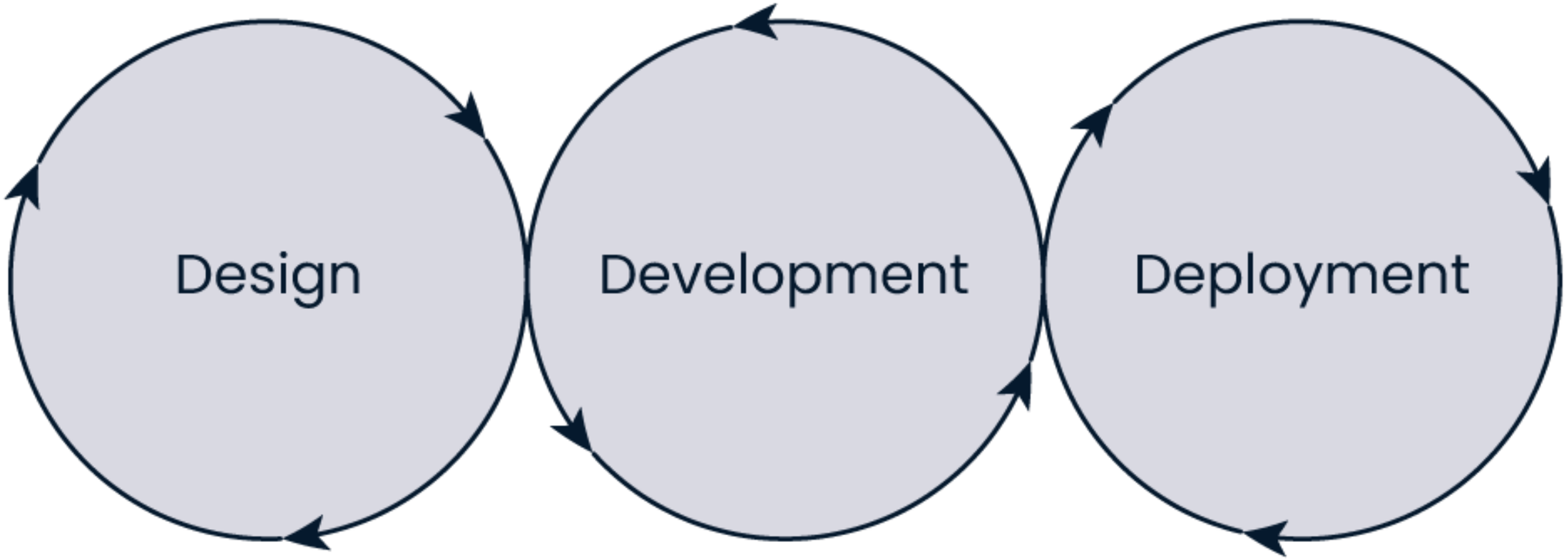
Data ingestion



Feature engineering

MLOPS CONCEPTS

Feature engineering



- Added value
- Business requirements
- Key metrics
- Data processing

- Feature engineering
- Experiment tracking
- Model training & evaluation

- Runtime environments
- Microservices architecture
- CI/CD pipeline
- Monitoring & retraining

Feature Engineering(특성 공학)은 **머신러닝 모델의 성능을 높이기 위해 데이터에서 유의미한 특성(Feature)을 생성, 변환, 선택, 또는 조합하는 과정**을 의미

- MLOps에서는 이 과정이 데이터 처리 파이프라인의 중요한 부분으로 포함되어, 모델의 품질과 예측 정확도를 결정하는 핵심 요소로 작용

Feature Engineering의 역할

1.모델 성능 향상:
고품질의 특성을 생성하면 머신러닝 모델이 데이터를 더 잘 이해하고, 높은 정확도를 제공할 수 있음

2.데이터 이해도 증가:
데이터를 변환하거나 새 특성을 생성함으로써 데이터의 구조와 패턴을 더 명확히 파악할 수 있음

3.효율적인 데이터 표현:
복잡하거나 다차원적인 데이터를 간소화하여 모델의 학습 속도와 효율성을 향상시킴

4.특성 관리의 자동화:
MLOps 파이프라인에서는 Feature Engineering 과정이 자동화되어 재현성과 확장성을 제공

MLOps에서 Feature Engineering의 중요성

MLOps에서는 Feature Engineering이 단순한 데이터 전처리를 넘어, **재현성**과 **자동화**가 핵심이 됨

1.자동화된 파이프라인:
데이터 수집부터 특성 생성, 모델 학습까지의 과정을 자동화하여 반복 작업을 줄이고 일관성을 유지.

2.재현성 보장:
동일한 데이터와 설정으로 항상 동일한 특성 변환 결과를 얻을 수 있도록 파이프라인을 설계.

3.특성 관리:
특성 버전 관리 및 특성 저장소(Feature Store)를 활용해 다양한 모델과 프로젝트 간 특성을 재사용.

4.실시간 Feature Engineering:
스트리밍 데이터에 대해 실시간으로 특성을 생성하고 처리.

Feature Engineering의 주요 과정

1.특성 선택 (Feature Selection):
모델 학습에 중요한 특성만 선택하여 차원을 축소하고 불필요한 노이즈를 제거.
예: 상관관계 분석, Lasso Regression, 중요도 순위에 따른 특성 제거.
2.특성 변환 (Feature Transformation):
기존 특성을 변환하여 모델이 데이터를 더 잘 학습할 수 있도록 변형.
예: 로그 변환, 정규화(Scaling), 표준화(Standardization), Box-Cox 변환.
3.특성 생성 (Feature Creation):
기존 데이터를 조합하거나 새 변수를 만들어 추가적인 정보를 생성.
예: 날짜 데이터를 "요일" 또는 "월"로 변환, 속도 = 거리/시간 계산.
4.결측치 처리 (Handling Missing Values):
결측 데이터를 대체하거나 제거하여 데이터의 완전성을 유지.
예: 평균, 중앙값으로 결측치 대체 또는 KNN Imputation.
5.범주형 데이터 처리 (Categorical Feature Handling):
범주형 데이터를 모델에 적합한 형식으로 변환.
예: 원-핫 인코딩(One-Hot Encoding), 레이블 인코딩(Label Encoding).
6.시간 데이터 처리 (Time Series Features):
시간 데이터를 활용해 특성을 생성.
예: 트렌드, 계절성, 이동 평균.

Feature Engineering의 한계 및 도전 과제
1.고비용:
고품질 특성을 생성하기 위해서는 도메인 지식과 많은 자원이 필요.
2.복잡성:
데이터가 복잡하거나 대규모일수록 Feature Engineering 과정이 까다로움.
3.자동화의 어려움:
비정형 데이터(예: 이미지, 텍스트)에서는 자동화가 어렵고 수작업이 필요한 경우가 많음.

Feature engineering

... is the process of selecting, manipulating, and transforming raw data into features.

- A feature is a variable, such as the column in a table
- We can use raw data, but also create our own

Customer data

Customer ID	Number of orders	Total expenditure
0	4	$1982
1	2	$8545
2	8	$102
...	...	...

Average expenditure
$495.50
$4272.50
$12.75
...

Feature engineering

- Goal is to enhance model performance
- Tools and techniques help to process, select, and maintain features:
  - Feature selection
  - Feature store
  - Data version control

Feature selection

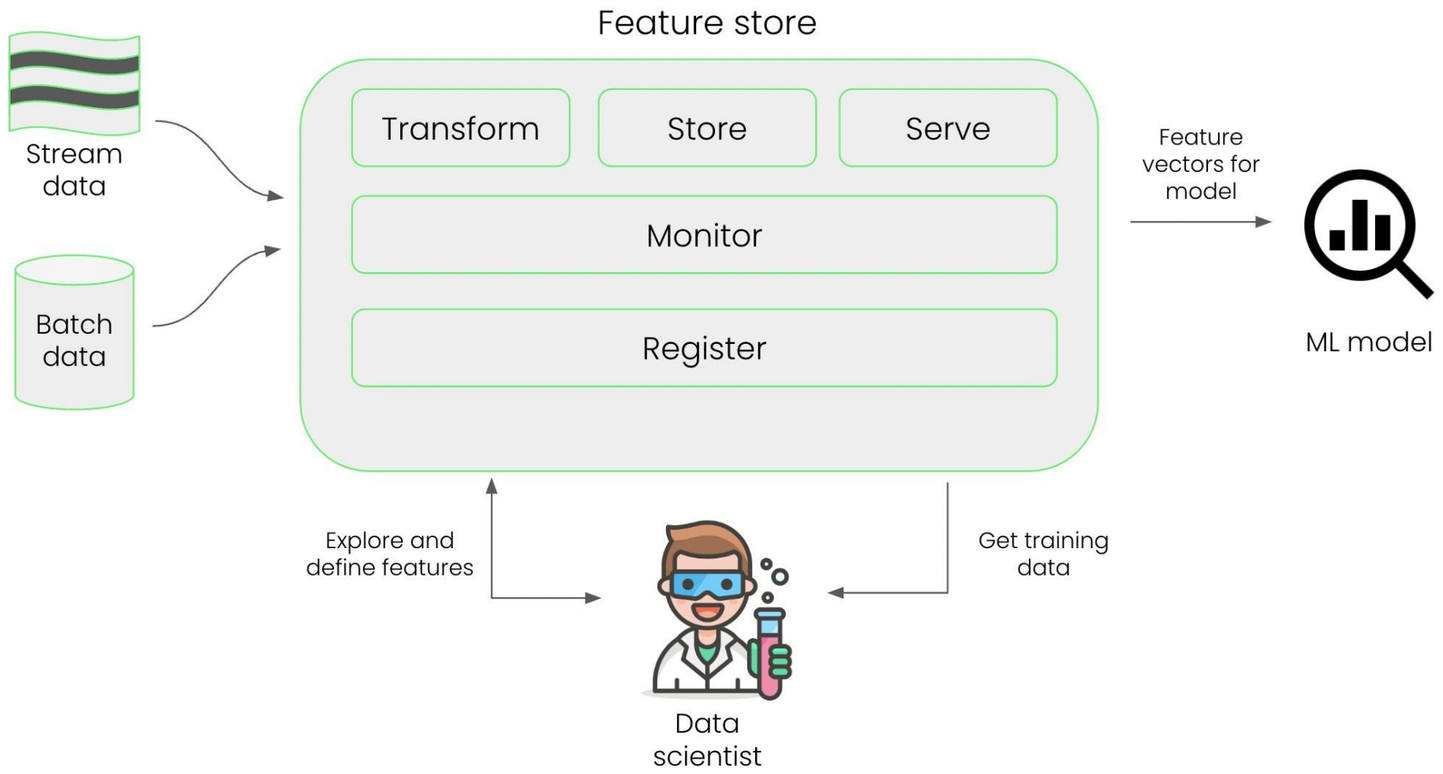
- Domain-specific knowledge
- Correlation
- Feature importances
- Other methods: univariate selection, Principal Component Analysis (PCA), Recursive Feature Elimination (RFE)

Univariate Selection은 **특성(Feature) 선택 기법** 중 하나로, 데이터셋의 각 독립 변수(Feature)와 종속 변수(Target) 간의 **통계적 관계**를 개별적으로 평가하여 가장 관련성이 높은 특성을 선택하는 방법

- **Univariate**라는 용어는 하나의 변수(특성)를 한 번에 평가한다는 것을 의미
- 이는 각 특성을 독립적으로 검토하며, 다른 특성 간의 상관관계나 상호작용은 고려하지 않음

Recursive Feature Elimination (RFE)은 **특성 선택 기법**으로, 머신러닝 모델의 성능을 기준으로 불필요하거나 덜 중요한 특성을 점진적으로 제거하여 **가장 유용한 특성만 선택**하는 방법
RFE는 다음과 같은 과정을 반복하여 특정 개수의 중요한 특성을 선택:
1. 모델을 학습시킨 후 각 특성의 중요도를 평가.
2. 가장 중요도가 낮은 특성을 제거.
3. 남은 특성으로 모델을 다시 학습.
4. 원하는 특성 개수가 남을 때까지 반복.

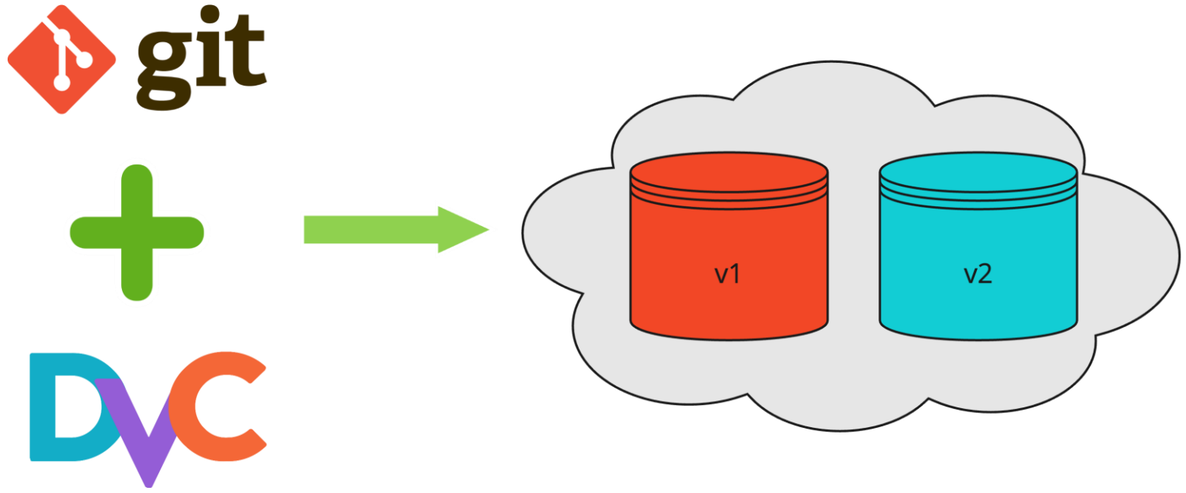
The feature store



Only relevant for large teams working on multiple projects that use the same features

Data version control

- Tracking dataset changes
- Maintaining consistency throughout the development lifecycle

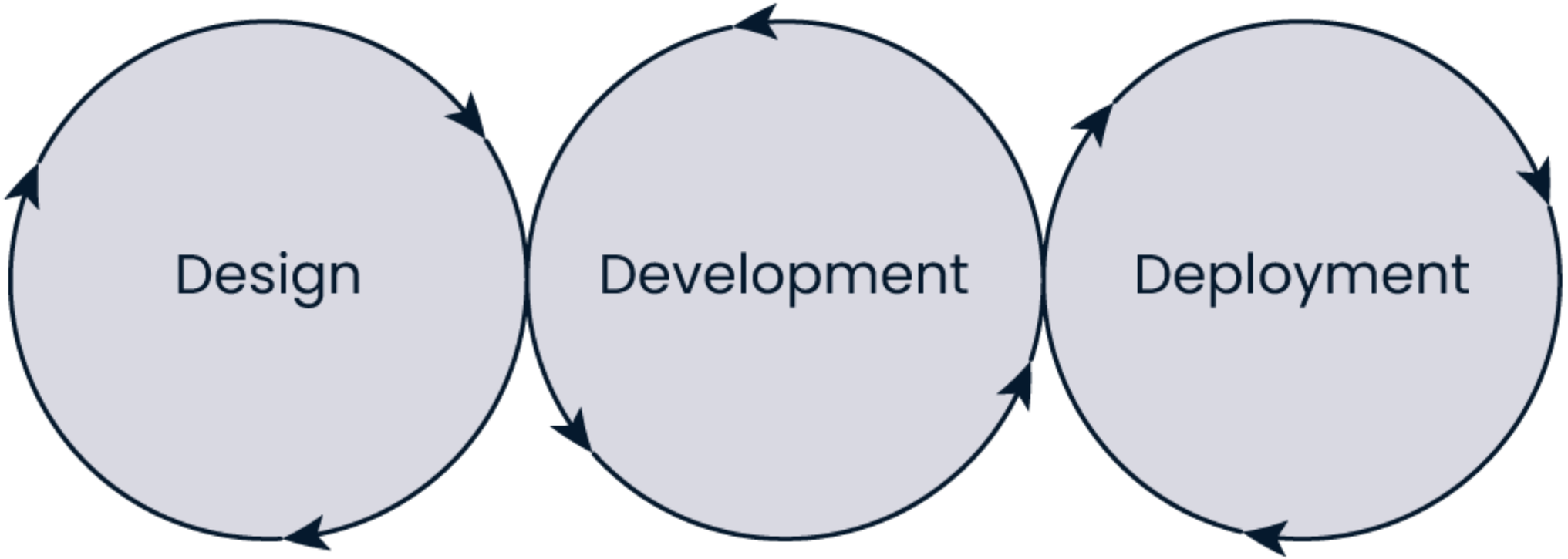


[1] https://www.datacamp.com/courses/cicd-for-machine-learning

Experiment tracking

MLOPS CONCEPTS

The machine learning experiment



- Added value
- Business requirements
- Key metrics
- Data processing

- Feature engineering
- Experiment tracking
- Model training & evaluation

- Runtime environments
- Microservices architecture
- CI/CD pipeline
- Monitoring & retraining

Why is experiment tracking important?

In each experiment, the following factors can be configured:

Machine learning models

Model hyperparameters

Versions of data

Execution scripts

Environment configurations

Experiment Tracking(실험 추적)은 머신러닝 모델 개발 과정에서 수행한 모든 실험의 **입력 변수, 실행 환경, 결과, 메트릭 등을 기록하고 관리**하는 프로세스를 의미

- MLOps의 중요한 부분으로, 실험 추적은 모델 개발의 **재현성**과 **효율성**을 높이는 데 필수

Using experiment tracking in the ML lifecycle

Experiment tracking can help to:

- Compare results
- Reproduce past experiments
- Collaborate with developers and stakeholders
- Report on results to stakeholders

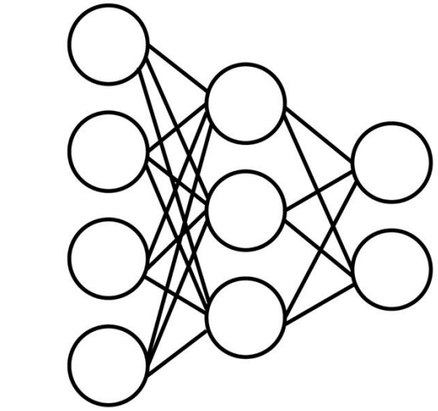
How to track experiments?

Tool	Pro	Con
Spreadsheet	Straightforward, easy to use	Require a lot of manual work
Proprietary platform	Custom solution specific for our process	Require time and effort
Experiment tracking tool	Specifically designed for experiments	Requires getting familiar with the tool

A machine learning experiment

Experiment 1

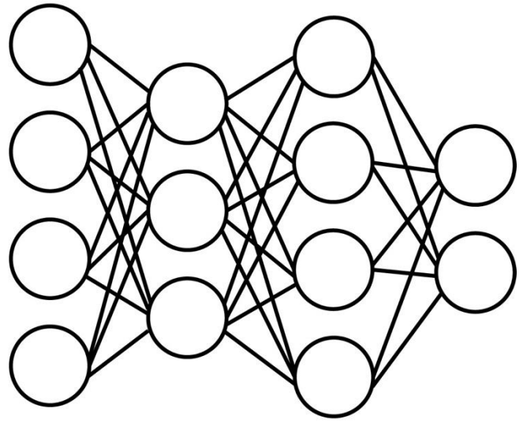
1.000 images of cats & dogs



A neural network with 1 hidden layer

Experiment 2

2.000 images of cats & dogs (including puppies & kittens)



A neural network with 2 hidden layers

The experiment process

1. Formulate a hypothesis: *"We expect that..."*
2. Gather images and labels
3. Define experiments, e.g., types of models, hyperparameters, datasets
4. Setup experiment tracking
5. Train the machine learning model(s)
6. Test the models on a hold-out test set
7. Register the most suitable model
8. Visualize and report back to team and stakeholders, and determine next steps