training code: DBX-DE-ML2D / ENG DL 2d / EN

Databricks Data Engineering & Data Preparation for ML

This two-day training guides participants through the entire process of data preparation and transformation in the Databricks environment for machine learning purposes. Day one is a data engineering crash course – from data loading and basic transformations to the use of Delta Lake, Auto Loader, and Medallion architecture with declarative Delta Live Tables pipelines. Day two focuses purely on data preparation for models – exploration, cleaning, feature engineering, splitting into training and test sets, and logging/sharing features with MLflow and Feature Store. The training is based on Databricks Academy official materials and documentation.

Training recipients

The training is intended for:

– Data scientists and ML engineers who want to build their own ML pipelines in Databricks.

– AI/ML and DataOps teams responsible for data quality and preparation.

– Data engineers preparing datasets for ML models.

– Participants familiar with Python/PySpark and ML modeling basics.

Benefits

– Solid foundation in data engineering in Databricks: data loading and validation, use of DataFrame API, ACID operations in Delta Lake, versioning, and time travel.

– Knowledge of tools for streaming and incremental data loading (Auto Loader) and benefits such as scalability, schema control, and exactly-once file processing.

– Understanding of Medallion architecture (Bronze→Silver→Gold) and building declarative pipelines with Delta Live Tables that automatically manage infrastructure and ensure reliability.

– Ability to explore, clean, and enrich data for machine learning (EDA, imputation, encoding, scaling, window functions).
– Learning to build complete ML pipelines with Spark MLlib and MLflow, and manage features in Feature Store for training and serving consistency.

Training program

Day 1 – Data Engineering for Machine Learning (crash course)
– Introduction to the Databricks platform: overview of Lakehouse architecture, workspace, clusters, notebooks, DBFS.
– Ingest and basic transformations:
  – Creating DataFrames in SQL and PySpark; select, filter, join, groupBy operations.
  – Loading data from CSV, JSON, Parquet formats and reading/writing Delta tables.
  – Schema management, ACID, time travel, MERGE, UPDATE, DELETE in Delta Lake.
– Streaming and incremental data loading:
  – Introduction to Auto Loader and cloudFiles for automatic detection of new files and exactly-once processing.
  – Comparison of "trigger once" and "continuous" modes; schema evolution handling and multiple file formats.
  – Structured Streaming: readStream / writeStream, checkpointing, fault tolerance.
– Medallion architecture and Delta Live Tables:
  – Concept of Bronze, Silver, Gold layers and their application in data preparation.
  – Working with Delta Live Tables – declarative table/view definitions, pipeline creation in GUI, scheduling, data quality expectations, monitoring, and lineage.
  – Combining DLT with Auto Loader and Structured Streaming in one flow.
– Orchestration and governance:
  – Using Databricks Workflows for jobs and multitask jobs.
  – Basics of access management and Unity Catalog.
  – Best practices for code versioning (Repos) and performance monitoring (Spark UI).

Day 2 – Data Preparation & Feature Engineering
– Data exploration (EDA):
  – Using display, summary, and profile commands to explore distributions and dependencies.
  – Creating visualizations in Databricks notebooks; identifying outliers.
– Data splitting:
  – Splitting into training, validation, and test sets; random and stratified sampling.
  – Cross-validation and time-based split concept.
– Imputation of missing values:
  – Methods for dropping (dropna) and filling (fillna) missing values.

– Using MLlib Imputer to replace with mean/median and create missing flags.
– Feature encoding and transformation:
– Encoding categories with StringIndexer and OneHotEncoder; introduction to target encoding.
– Scaling numeric variables with StandardScaler, MinMaxScaler, RobustScaler.
– Window functions: lag, lead, row_number, rolling average for sequential features.
– Feature engineering and selection:
– Difference between extraction and selection; creating new variables via aggregations, log transforms, interactions.
– Combining variables into a vector with VectorAssembler.
– Building ML pipelines:
– Defining Pipeline stages (imputation, encoding, scaling, model).
– Fitting models and evaluation; logging metrics, hyperparameters, and models in MLflow.
– Feature Store & MLflow:
– Creating and registering feature tables in Databricks Feature Store; versioning and sharing features for multiple models.
– Using mlflow.log_model and mlflow.register_model to track experiments and ensure consistency between training and serving.
– Best practices and testing:
– Modular notebook design and unit testing of transformation functions.
– Documenting pipelines, monitoring data quality, and managing lineage.

Expected preparation of the participant

Participant Prerequisites
Participants should know basic ML concepts (e.g., regression and classification), use PySpark or SQL, and have experience working with data or pipelines.

Training Includes

access to Altkom Akademia student

Duration

2 days / 14 hours

Language

Training: English

- Materials: English