Spark Course
Introduction to the Machine Learning API in Apache Spark

Luca Canali
CERN IT, Data Analytics and Spark Service

Why Spark for ML?

- Data preparation
  - Run distributed data ingestion, feature preparation
  - This step takes a large fraction of effort
- Scale: Hyperparameter search and training
- Familiar APIs similar to popular tools: scikit-learn

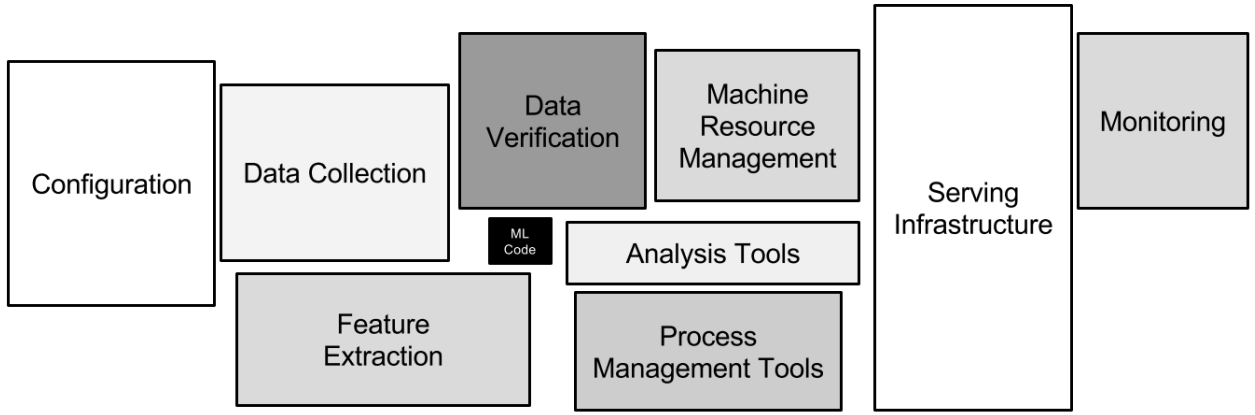


Figure from “Hidden technical debt in machine learning systems”, Google, 2015

Typical Steps in a ML Project

- Explore your dataset
- Prepare the data for ML algorithms
  - Feature engineering
  - Feature preparation
- Model development
  - Experiment with different models for your problem
  - Hyperparameter tuning
- Train the model
  - Test and evaluate
  - Performance metrics
- Move to production and Inference

Spark APIs

- Explore your dataset
- Prepare the data for ML algorithms
  - Feature engineering
  - Feature preparation
- Search for the best model for your problem
- Train (and validate!)
- Inference

Spark SQL/DataFrame

spark.ml

API: spark.ml

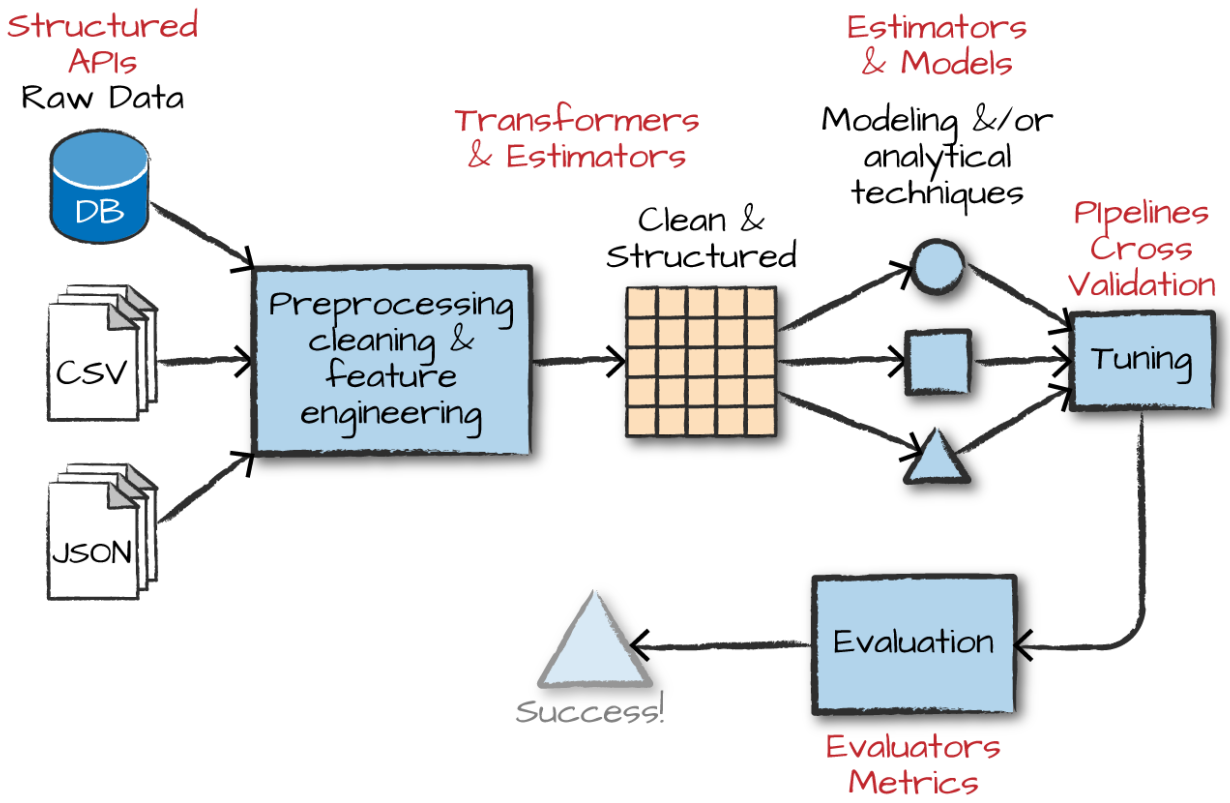


Figure from: Chambers, Zaharia, “Spark: The Definitive Guide”, O’Reilly 2018

Key Components of spark.ml

- DataFrames:
  - Data processing with Spark's main API
- Transformers:
  - Algorithms that transform DFs (from input data to transformed data)
  - They have a *transform* method
- Estimators:
  - Algorithms that build custom Transformers based on data
  - They have a *fit* method.
- Pipelines:
  - Allow to put together multiple processing steps
  - Build ML workflows composed of Transformers and Estimators

Key Steps with Spark ML

- Create a pipeline

```
from pyspark.ml import Pipeline

# the pipeline prepares the features and feeds them to a model
pipeline = Pipeline(stages = [mytransform1,..., mymodel] )
```

- Fit the pipeline to the training dataset

```
training_data, test_data = df.randomSplit([0.8, 0.2], 4242)
model_transformer = pipeline.fit(training_data)
```

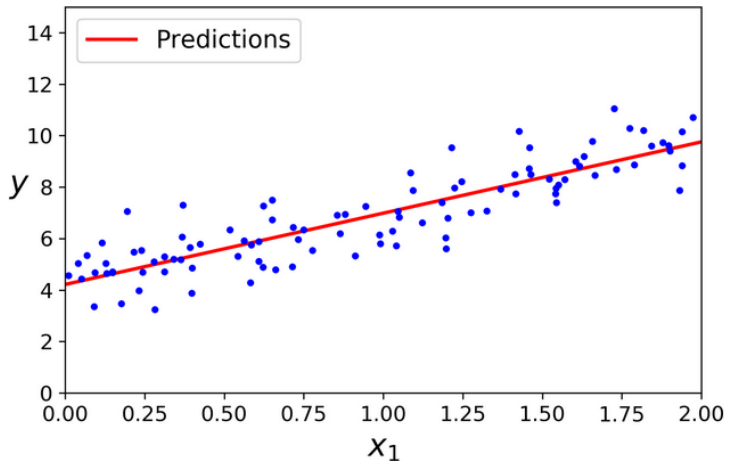
- Use the model against the test dataset

```
predicted_data = model_transformer.transform(test_data)
```

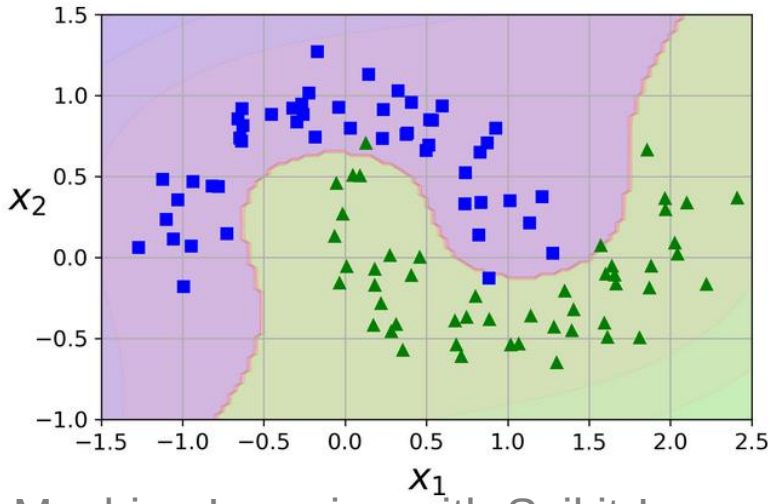
ML Model Training

- Supervised learning models
  - Classification: predict a label
  - Regression: predict a quantity
- Fitting a model to training data
  - Finding a (complex) function, learnt from data, to characterize the inputs

Regression



Classification



Figures from: Aurelien Geron, “Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition”, O’Reilly 2019

Machine Learning with Spark, Demo (Notebook)

Spark and Deep Learning

- Use Spark for data preparation at scale
  - Data exchange formats
  - Parquet, TFRecord, etc
- Run TensorfFlow/PyTorch on GPU
  - GPUs on SWAN: https://swan-k8s.cern.ch
- Tools with 3rd party work on integrating Spark with TensorFlow and Pytorch
  - Horovod, BigDL

Key Learning Points

- Spark can run data preparation at scale for ML projects
  - Profit of Spark DataFrame API to read and process data from different sources and data formats

- Spark Machine Learning
  - Use Spark ML libraries to scale ML model training and hyper parameter tuning
  - Use Spark DataFrames and UDF for model inferencing