**ML Pipelines**

**ML Pipelines** provide a uniform set of high-level APIs built on top of [DataFrames](http://spark.apache.org/docs/latest/sql-programming-guide.html) that help users create and tune practical machine learning pipelines.

# Main concepts in Pipelines

MLlib standardizes APIs for machine learning algorithms to make it easier to combine multiple algorithms into a single pipeline, or workflow.

* [**DataFrame**](http://spark.apache.org/docs/latest/ml-pipeline.html#dataframe): This ML API uses DataFrame from Spark SQL as an ML dataset, which can hold a variety of data types
* [**Transformer**](http://spark.apache.org/docs/latest/ml-pipeline.html#transformers): A **Transformer** is an algorithm which can transform one DataFrame into another DataFrame. E.g., an ML model is a Transformer which transforms a DataFrame with features into a DataFrame with predictions.
* [**Estimator**](http://spark.apache.org/docs/latest/ml-pipeline.html#estimators): An Estimator is an algorithm which can be fit on a DataFrame to produce a Transformer. E.g., a learning algorithm is an Estimator which trains on a DataFrame and produces a model.
* [**Pipeline**](http://spark.apache.org/docs/latest/ml-pipeline.html#pipeline): A Pipeline chains multiple Transformers and Estimators together to specify an ML workflow.
* [**Parameter**](http://spark.apache.org/docs/latest/ml-pipeline.html#parameters)**:** All Transformers and Estimators now share a common API for specifying parameters.

## Pipeline components

### Transformers

* A Transformer is an abstraction that includes feature transformers and learned models. A Transformer implements a method transform(), which converts one DataFrame into another, generally by appending one or more columns.

**For example:**

* A feature transformer might take a DataFrame, read a column (e.g., text), map it into a new column (e.g., feature vectors), and output a new DataFrame with the mapped column appended.
* A learning model might take a DataFrame, read the column containing feature vectors, predict the label for each feature vector, and output a new DataFrame with predicted labels appended as a column

### Estimators:

* An Estimator abstracts the concept of a learning algorithm or any algorithm that fits or trains on data. an Estimator implements a method fit(),which accepts a DataFrame and produces a Model, which is a Transformer. For example, a learning algorithm such as LogisticRegression is an Estimator, and calling fit() trains a LogisticRegressionModel, which is a Model and hence a Transformer.

## Pipeline

In machine learning, it is common to run a sequence of algorithms to process and learn from data. E.g., a simple text document processing workflow might include several stages:

* Split each document’s text into words.
* Convert each document’s words into a numerical feature vector.
* Learn a prediction model using the feature vectors and labels.

A Pipeline is specified as a sequence of stages, and each stage is either a Transformer or an Estimator. These stages are run in order, and the input DataFrame is transformed as it passes through each stage For Transformer stages, the transform() method is called on the DataFrame. For Estimator stages, the fit() method is called to produce a Transformer (which becomes part of the PipelineModel, or fitted Pipeline), and that Transformer’s transform() method is called on the DataFrame.