Machine learning for Big Data

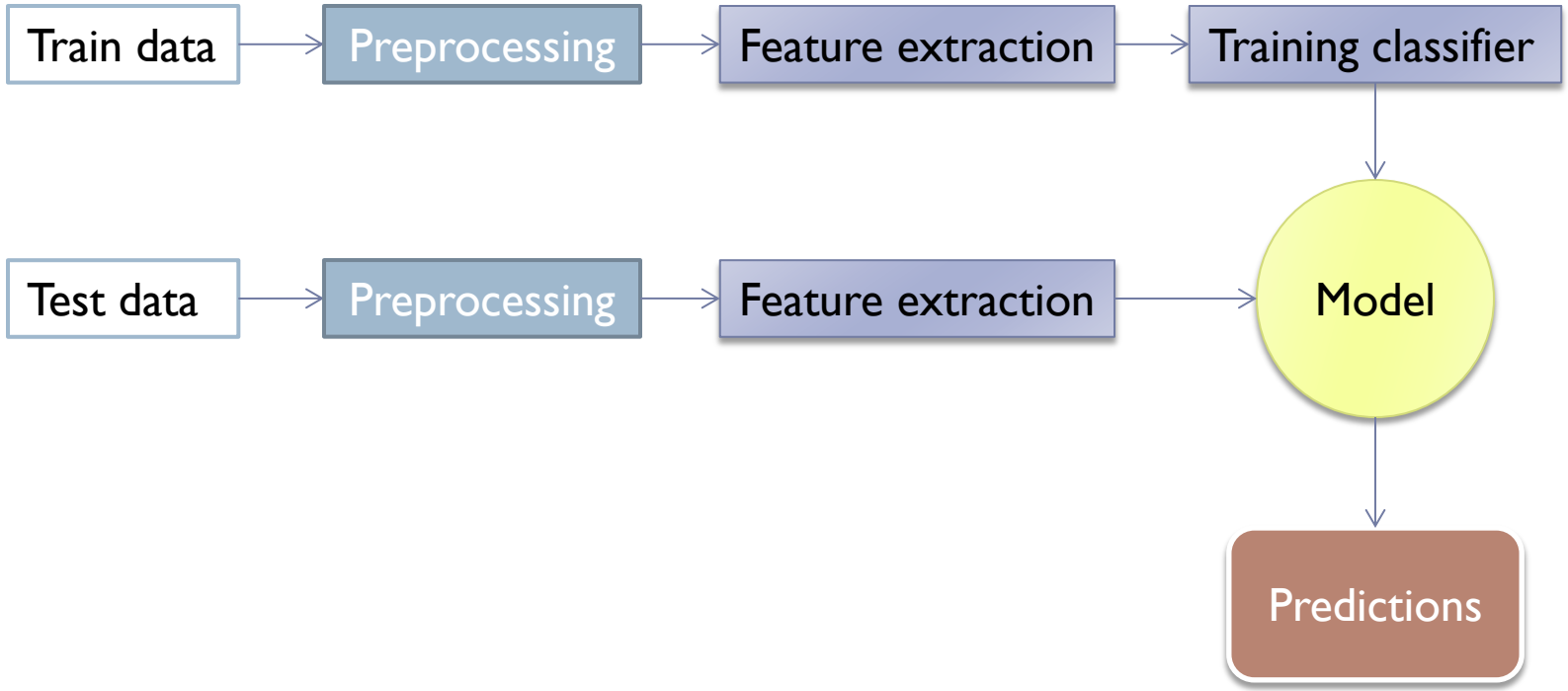
Mai Hai Thanh KIWI @ ETRI

Introduction

- SQL engines help to extract explicit information
  - Selection, join, group by, aggregation,…
- Machine learning (ML) engines helps to extract hidden information
  - Clustering, classification, prediction,…
- Both SQL and ML are important

Introduction

- Machine learning pipeline



Popular ML libraries for big data

- Apache Spark's MLlib
- Apache Mahout
- Apache Storm's Trident ML lib
- Jubatus
- Graphlab
- DistBelief
- H2O
- Scikit-learn
- Weka
- …

Popular ML algorithms

- Spark's MLlib 1.3 (all use Spark engine)
  - Linear SVM and logistic regression
  - Random forest and gradient-boosted trees
  - Recommendation via alternating least squares
  - Clustering via k-means, Gaussian mixtures, and power iteration clustering
  - Topic modeling via latent Dirichlet allocation
  - Singular value decomposition
  - Linear regression with L1- and L2-regularization
  - Isotonic regression
  - Multinomial naive Bayes
  - Frequent itemset mining via FP-growth

Popular ML algorithms

- Mahout 0.10.0

	Single Machine	MapReduce	Spark	
User-Based Collaborative Filtering	x			x
Item-Based Collaborative Filtering	x	x		x
Matrix Factorization with ALS	x	x		
Matrix Factorization with ALS on Implicit Feedback	x	x		
Weighted Matrix Factorization, SVD++	x			
Logistic Regression - trained via SGD	x			
Naive Bayes / Complementary Naive Bayes		x		x
Random Forest		x		
Hidden Markov Models	x			
Multilayer Perceptron	x			
k-Means Clustering	x	x		
Fuzzy k-Means	x	x		
Streaming k-Means	x	x		
Spectral Clustering		x		
Singular Value Decomposition	x	x		x
Stochastic SVD	x	x		x
PCA (via Stochastic SVD)	x	x		x
QR Decomposition	x	x		x
Latent Dirichlet Allocation	x	x		

Popular optimization techniques

- Often used for machine learning
  - Gradient descent
  - Stochastic gradient descent
    - Faster than gradient descent
  - L-BFGS
    - Faster than gradient descent

Iterative methods

  - Normal equation

Non-iterative methods

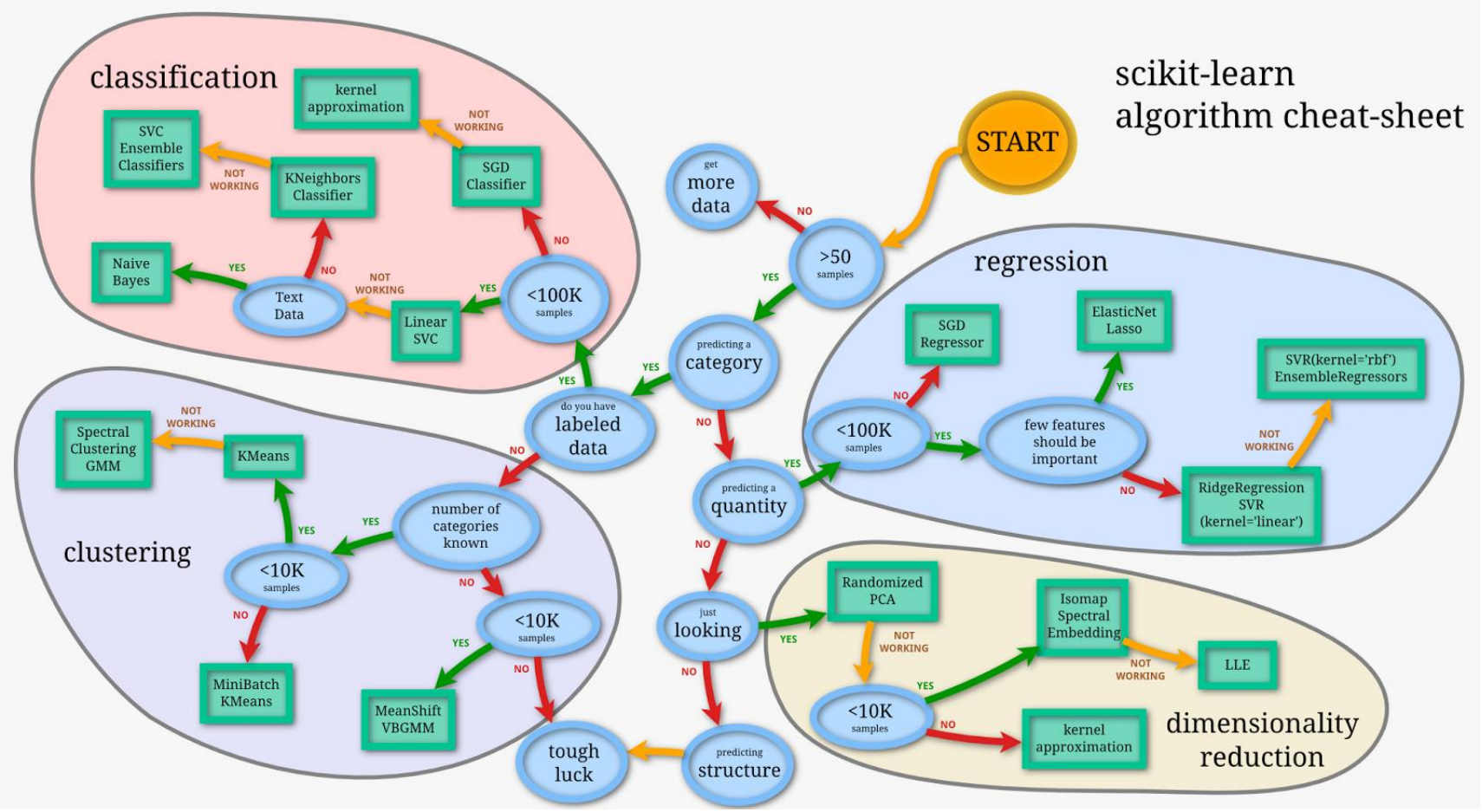
Spark MLlib

- Built on Spark in-memory compute engine
- Very fast, scalable, widely used
- User-friendly APIs
- Developers write applications in Scala, Java, Python
- Use any Hadoop data source (HDFS, HBase, local files,…)
- Interact with other Spark's components (SparkSQL, GraphX, …) seamlessly
- Example

```
points = spark.textFile("hdfs://...").map(parsePoint)
model = KMeans.train(points, k = 10)
```

ML algorithm selections

- Wisdom of the crowd



ML algorithm selections

- Wisdom of the crowd