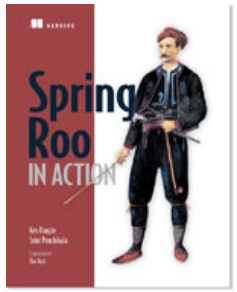
Logistic Regression using Spark Machine Learning

Srini Penchikala
10.04.17

About Me

- Software Architect
- Big Data Processing with Apache Spark book (Q4 2017)
- Co-author of "Spring Roo in Action" book (2012)
- Current Focus:
  - Reactive Microservices
  - Containers
  - ML/Deep Learning



Introductions

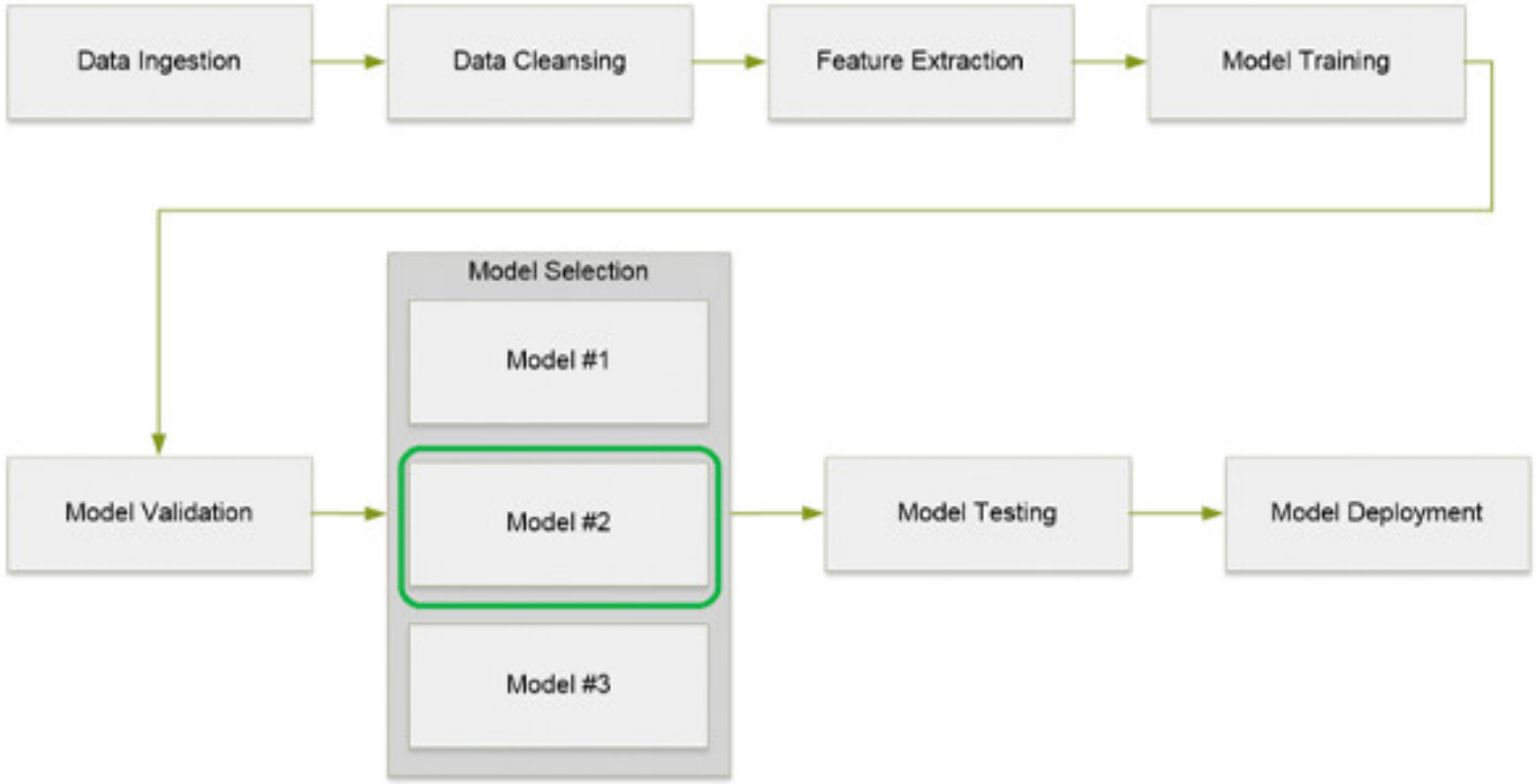
- Role
  - Developers / Architects
  - Data Scientists
  - Data Analysts
  - DBAs, OPS Team
  - Other role?

- Experience in:
  - Machine Learning
  - Apache Spark
  - Spark MLlib
  - Scala

Agenda

- Machine Learning
- Classification & Regression
- Spark MLlib
- Sample Application
- Demo
- Conclusions
- Q&A

Machine Learning Data Pipeline



*Source: https://www.infoq.com/articles/apache-sparkml-data-pipelines

Machine Learning Categories

Supervised Learning

- Make predictions based on a set of examples
- Look for patterns in the value labels
- Task driven
- Examples: Classification, Regression, Anomaly Detection

Unsupervised Learning

- Data points have no labels associated with them
- Goal is to organize data in some way or to describe its structure (e.g. group data into clusters)
- Data driven
- Examples: Clustering, Dimensionality Reduction, Recommender Systems, Deep Learning

Reinforcement Learning

- Learns by interacting with its environment and observing the results of these interactions
- Is a form of unsupervised learning
- Find balance between "exploration" (of uncharted territory) and "exploitation" (of current knowledge)
- Use cases: Game Theory, Robotics, Computer Networking, Vehicle Navigation, AlphaGo

Machine Learning Algorithms

Classification & Regression

- Logistic Regression
- Decision Tree
- Random Forest
- Gradient Boosted Tree

Collaborative Filtering

- Alternating Least Squares (ALS)

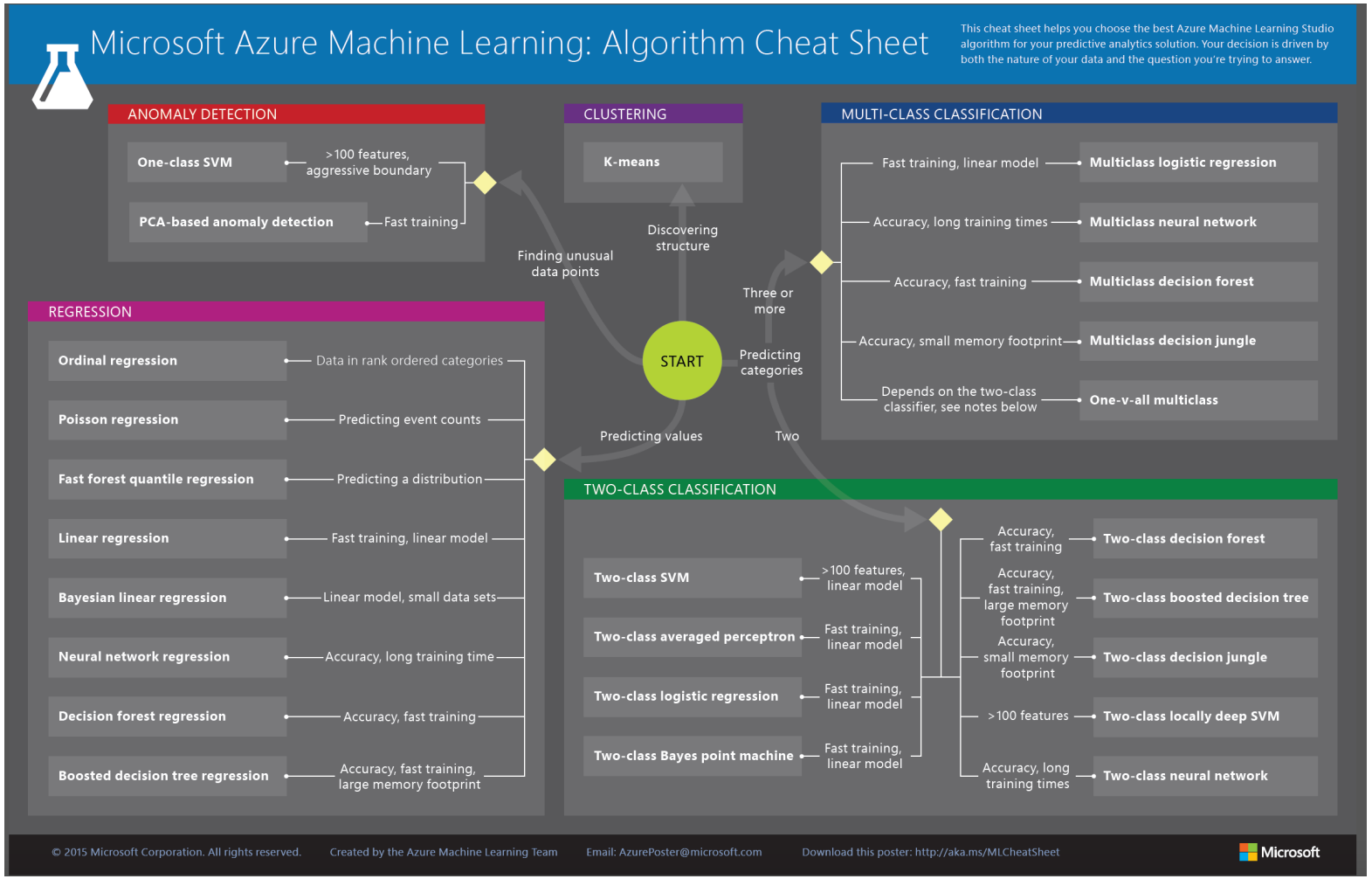
Frequent Pattern Mining

- FP-growth

Clustering

- K-Means
- LDA

ML Algorithm Cheat Sheet



Source:
http://download.microsoft.com/download/A/6/1/A613E11E-8F9C-424A-B99D-65344785C288/microsoft-machine-learning-algorithm-cheat-sheet-v6.pdf

Classification Use Cases

- Spam detection
- Google news classification
- Cancer cell classification (Benign, Malignant)
- Fraud detection
- Weather prediction
- Credit scoring
- Ad targeting
- Image classification

Logistic Regression

- Measures the relationship between categorical dependent variable & one or more independent variables
- Developed by statistician David Cox in 1958
- Outcome is usually coded as 0 or 1 (success=1, failure=0)
- Function:

$$P(y = 1|x) = h_\theta(x) = \frac{1}{1 + \exp(-\theta^\top x)} \equiv \sigma(\theta^\top x),$$
$$P(y = 0|x) = 1 - P(y = 1|x) = 1 - h_\theta(x).$$

Logistic Regression Types

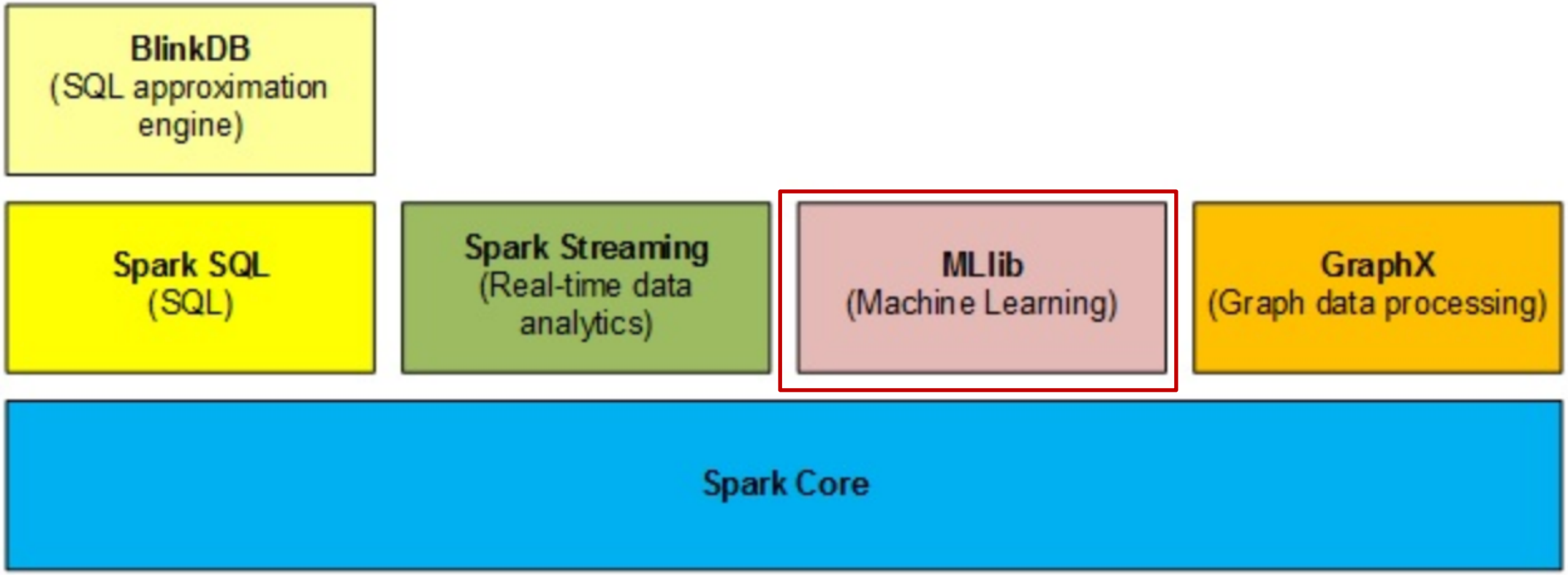
- Logistic regression can be binomial, multinomial or ordinal
- Binomial: observed outcome for a dependent variable can have only two possible types: 0 & 1
- Multinomial: outcome can have three or more possible types that are not ordered
- Ordinal: deals with dependent variables that are ordered

Use Cases

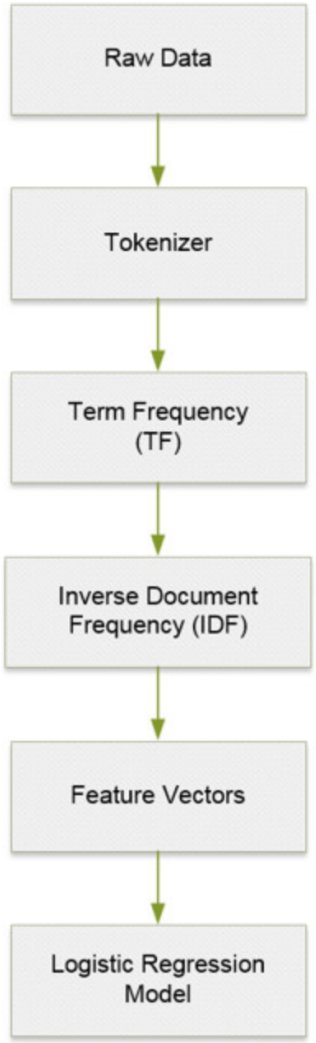
- Probability of failure of a given process, system or product
- Trauma & Injury Severity Score (TRISS), which is used to predict mortality in injured patients
- Predict:
  - whether a patient has a given disease based on observed characteristics of the patient (age, gender, body mass index, results of blood tests)
  - whether an American voter will vote for one party or another, based on age, income, gender, race, state of residence, votes in previous elections
  - if a customer would purchase a product or halt a subscription
  - the likelihood of a homeowner defaulting on a mortgage

Source: https://en.m.wikipedia.org/wiki/Logistic_regression

Spark Ecosystem with Spark MLlib



Text Classification



TF-IDF

Term Frequency - Inverse Document Frequency (TF-IDF)

Statistical measure to evaluate how important a word is to a document in a given corpus

Used to rank how important a word is to a collection of documents

TF: If a word appears frequently in a doc, it's important. This is calculated as:

```
TF = (# of times word X appears in a document) / (Total # of words in the document)
```

IDF: used to diminish the weight of terms that occur very frequently in the document set and increases the weight of terms that occur rarely (e.g "the")

Sample Application

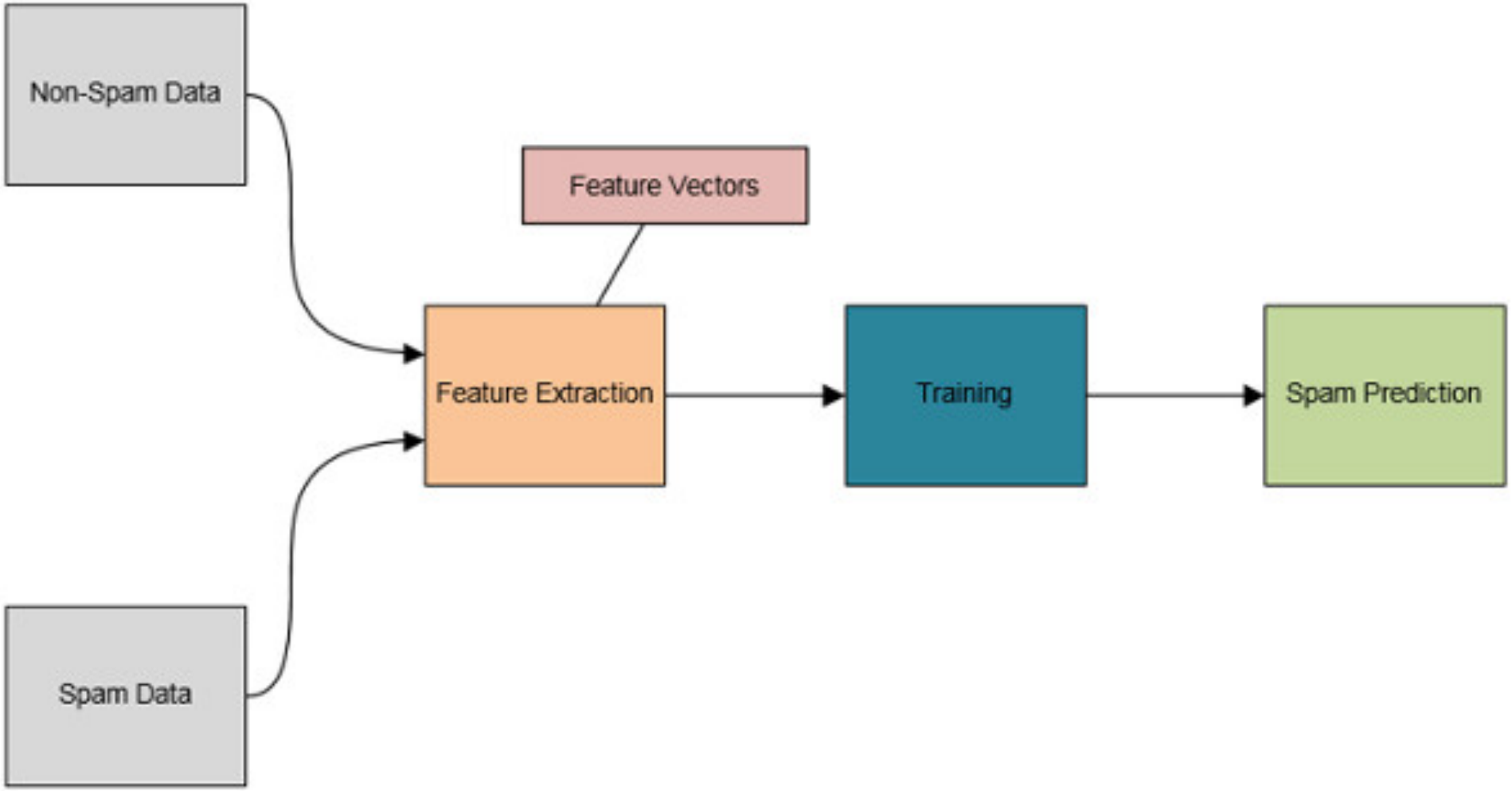
Use Case:

- Spam Detection

Technologies

- Spark MLlib (v 2.2.0)
- Scala
- Spark Shell (CLI)

Spam Detection Process



Spark MLlib Classification API

- Tokenizer
- HashingTF
- IDF
- LogisticRegression
- Pipeline
- BinaryClassificationEvaluator

*Package: `org.apache.spark.ml`

Demo

- Logistic Regression
- Reference: Building machine-learning apps with Spark
- Github Project
- Scala Example
- Datasets*
  - Not spam (3,600 files)
  - Spam (1,500 files)
- Training Dataset
  - Iteration #1: Small (~10 files)
  - Iteration #2: Large (~3k files)

* Caution about the spam file content

Next Steps/Enhancements

- Streaming data analytics
- Kafka & Spark Streaming
- Deep Learning & NLP (Tensorflow)

Conclusions

- Classification
- Logistic Regression
- Spark MLlib framework

References

- Apache Spark main website
- Spark Machine Learning Programming Guide
- Logistic Regression
- Apache Spark Data Pipelines article on InfoQ
- Apache Spark article series on InfoQ

Thank You

- Contact Information
  - http://www.infoq.com/author/Srini-Penchikala
  - srinipenchikala@gmail.com
  - @srinip
- Big Data Processing using Apache Spark
- Spring Roo in Action Book

Questions?