**ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**



**Báo cáo đồ án môn học**

**DỮ LIỆU LỚN**

*Chủ Đề : Tìm hiểu MLlib của Spark và demo phân tích dữ liệu sử dụng MLlib*

**Giáo viên hướng dẫn : Nguyễn Hồ Duy Tri**

**Lớp : IS405.K11**

**Thực hiện :**

* **PhạmVăn Hữu -15520293**
* **Lê Hồng Ngọc- 15520553**
* **Hoàng Trọng Nghĩa – 16521747**

**MỤC LỤC**

[**I. Lý do chọn đề tài 4**](#_Toc26808622)

[**II. Tổng quan về Mllib – Machine Learning Library 4**](#_Toc26808623)

[**III. Mllib Components 5**](#_Toc26808624)

[**1. Basic Statistics 5**](#_Toc26808625)

[**2. Algorithms 6**](#_Toc26808626)

[**3. Featurization 8**](#_Toc26808627)

[**4. Pipeline 9**](#_Toc26808628)

[**4.1. Các thành phần của Pipeline trong Spark Mllib 9**](#_Toc26808629)

[**4.2. Pipeline và cách hoạt động 10**](#_Toc26808630)

[**4.3. ML Persistence 10**](#_Toc26808631)

[**IV. Demo phân tích dataset với Mllib 11**](#_Toc26808632)

[**1. Thông tin dữ liệu 11**](#_Toc26808633)

[**2. Các yêu cầu cài đặt chung 12**](#_Toc26808634)

[**3. Source code demo phân tích dataset Black Friday 13**](#_Toc26808635)

[**V. Tài liệu tham khảo 13**](#_Toc26808636)

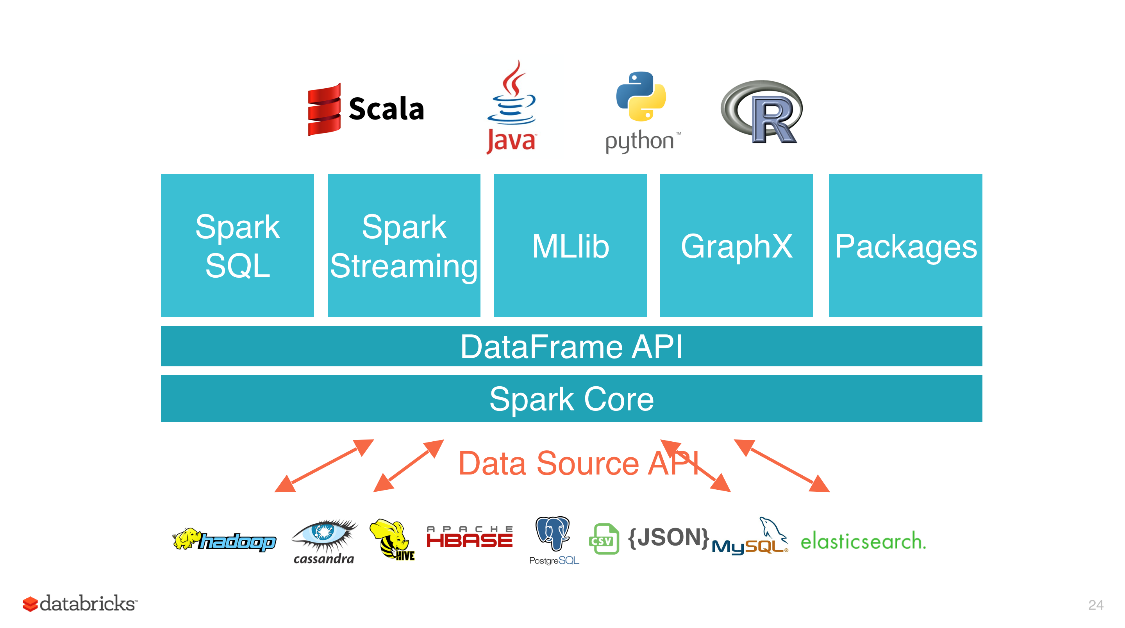
# Lý do chọn đề tài

Cùng với sự phát triển của công nghệ hiện nay, một số lượng dữ liệu khổng lồ được sinh ra và vấn đề giải quyết lượng dữ liệu đó trở nên quan trọng hơn bao giờ hết. Hạn chế về não bộ con người và nhân lực khiến cho các nhà khoa học hướng tới một công nghệ tiên tiến hơn – **Trí Tuệ Nhân Tạo** (Artificial Intelligence – AI). Trong quá trình học máy nói chung, phần lớn hai công đoạn **data preparation và modelling** là tốn khá nhiều thời gian .Để rút ngắn thời gian xử lý, việc cải thiện các thuật toán, hay nâng cấp phần cứng cũng là một phương pháp nhưng không mang lại hiểu quả cao khi xử lý với lượng dữ liệu lớn. Khi tập dữ liệu của AI đa số là dữ liệu độc lập nên việc chia nhỏ và xử lý song song là có khả năng. Lợi dụng đặc điểm này, có hai phương pháp được sinh ra làm tăng tốc độ xử lý **ML/DL** với **Big Data** hiệu quả.

Mllib và Spark ML là thư viện cung cấp các thuật toán ML hỗ trợ MapReduce. Vì có sử dụng MapReduce nên công cụ xử lý hiệu quả với dữ liệu lớn, và được tích hợp trong Apacha Spark ecosystem. Các thuật toán đa số đã được tối ưu hóa và tiền xử lý cho Hadoop.

Trong phạm vi môn học Bigdata cùng với những thông tin bên trên nhóm đã quyết định chọn đề tài là tìm hiểu về Mllib – Machine Learning Library.

# Tổng quan về Mllib – Machine Learning Library



* Spark Mllib là một thư viện về Machine Learning của Spark Apache.
* Với mục đích lớn nhất của Spark Mllib là có làm cho việc thực hành học máy có thể mở rộng và dễ dàng hơn.
* Đơn giản hóa việc phát triển và triển khai các vấn đề máy học với sự mở rộng của các Pipeline của Machine Learning.

# Mllib Components

**Utilities**

* Linear algebra
* Statistics

**Pipeline**

* Constructing
* Evaluating
* Tuning
* Persistence

**Algorithms**

* Classification
* Regression
* Clustering
* Collaborative Filtering

**Featurization**

* Extraction
* Transformation

## Basic Statistics

Bao gồm các kĩ thuật cơ bản nhất của Machine Learning như :

* **Correlations: Là một kĩ thuật phổ biến trong thống kê, dùng để tính mối tương qua giữa hai chuỗi dữ liệu. Các phương thức được hỗ trợ trong hiện tại như Pearson và Pearman.**
* **Hypothesis Testing: hay còn gọi là kiểm định giả thuyết thống kê, là công cụ mạnh mẽ trong thống kê dùng để quyết định xem một kết quả là có đủ độ tin cậy mang tính thông kê hay không.**  Các phương pháp được hỗ trợ như Pearson’s Chi-squared (X2) tests.
* **Summary Statistics: là phương pháp dùng để cung cấp các số liệu thống kê cho Dataframe. Các cột được thể hiện trong Dataframe như** mean, variance, count, max, min and numNonZeros.
* **Ngoài ra thì còn một số kĩ thuật khác như Stratified Sampling (**sampleBykey, sampleByKeyExact**)**, **Random Data Generation (**RandomRDDs, Normal and Poisson**).**

## Algorithms

* **Regression**

Việc phân tích sử dụng các thuật toán Regression là một qui trình thống kê nhằm xây dựng một mô hình mối quan hệ giữa các biến với nhau.Trong nhóm thuật toán Regression bao gồm nhiều kĩ thuật xây dựng mô hình và phân tích như Linear Regression, Generaralized Linear Regression, Decision Tree Regression, Random Forest Regression, [Gradient-boosted tree regression](https://spark.apache.org/docs/latest/ml-classification-regression.html#gradient-boosted-tree-regression), [Survival regression](https://spark.apache.org/docs/latest/ml-classification-regression.html#survival-regression), [Isotonic regression](https://spark.apache.org/docs/latest/ml-classification-regression.html#isotonic-regression).Các kĩ thuật tập trung vào việc tìm ra mối quan hệ giữa một biến phụ thuộc với một hay nhiều biến độc lập.Hơn thế nữa ,với việc phân tích Regression sẽ giúp người phân tích có thể hiểu được các trường hợp thay đổi giá trị của một biến phụ thuộc khi bất kì một biến độc lập nào đó thay đổi trong khi các biến độc lập còn lại vẫn giữ cố định.

Trong Machine Learning thì các phương pháp Regression cũng được sử phổ biến cho các nhu cầu dự đoán và dự báo một biến số có giá trị liên tục nào đó từ việc tìm ra các mối quan hệ giữa các biến.

* **Classification**

Là việc xác định một quan sát mới thuộc vào loại nào từ việc phân tích một tập dữ liệu có chứa các trường hợp xảy ra của vấn đề đó.Ví dụ như các vấn đề nhận dạng giọng nói,khuôn mặt,vấn đề xác định đâu là email “spam” và “non spam”,….

Các thuật toán của nhóm Classification được hỗ trợ như : [Logistic regression](https://spark.apache.org/docs/latest/ml-classification-regression.html#logistic-regression) ([Binomial logistic regression](https://spark.apache.org/docs/latest/ml-classification-regression.html#binomial-logistic-regression), [Multinomial logistic regression](https://spark.apache.org/docs/latest/ml-classification-regression.html#multinomial-logistic-regression)), [Decision tree classifier](https://spark.apache.org/docs/latest/ml-classification-regression.html#decision-tree-classifier), [Random forest classifier](https://spark.apache.org/docs/latest/ml-classification-regression.html#random-forest-classifier), [Gradient-boosted tree classifier](https://spark.apache.org/docs/latest/ml-classification-regression.html#gradient-boosted-tree-classifier), [Multilayer perceptron classifier](https://spark.apache.org/docs/latest/ml-classification-regression.html#multilayer-perceptron-classifier), [Linear Support Vector Machine](https://spark.apache.org/docs/latest/ml-classification-regression.html#linear-support-vector-machine), [One-vs-Rest classifier (a.k.a. One-vs-All)](https://spark.apache.org/docs/latest/ml-classification-regression.html#one-vs-rest-classifier-aka-one-vs-all), [Naive Bayes](https://spark.apache.org/docs/latest/ml-classification-regression.html#naive-bayes),…

* **Clustering**

Là công việc gom nhóm các đối tượng chưa được gán nhãn có tính tương đồng (ở đây tính tương đồng được định nghĩa tùy thuộc vào mục đích,lĩnh vực,…của người thực hiện công việc gom nhóm) lại với nhau để tạo thành các nhóm các đối tượng.Là một kĩ thuật được sử dụng phổ biến trong nhiều lĩnh vực máy học,nhận dạng,phân tích hình ảnh,truy xuất thông tin,sinh học,…

Các kĩ thuật hỗ trợ cho việc gom nhóm như : [K-means](https://spark.apache.org/docs/latest/ml-clustering.html#k-means), [Latent Dirichlet allocation (LDA)](https://spark.apache.org/docs/latest/ml-clustering.html#latent-dirichlet-allocation-lda), [Bisecting k-means](https://spark.apache.org/docs/latest/ml-clustering.html#bisecting-k-means), [Gaussian Mixture Model (GMM)](https://spark.apache.org/docs/latest/ml-clustering.html#gaussian-mixture-model-gmm),..

* **Colaborative filtering**

Được sử dụng cho hệ thống khuyến nghị - Recommandation System.Mục đích chính của kĩ thuật này là điền vào các giá trị còn thiếu trong ma trận kết hợp giữa user và item.

## Featurization

Trong giai đoạn này là sẽ làm việc với các thuộc tính trong tập dữ liệu để hỗ trợ cho các thuật toán.Được chia thành các nhóm như :

* **Extraction**

Ở nhóm Extraction sẽ thực hiện việc trích xuất các thuộc tính/đặc trưng – Features từ tập dữ liệu thô ban đầu.Các phương thức hỗ trợ cho việc trích xuất thuộc tính như : TF-IDF, Word2Vec, [CountVectorizer](https://spark.apache.org/docs/latest/ml-features.html#countvectorizer), [FeatureHasher](https://spark.apache.org/docs/latest/ml-features.html#featurehasher),..

* **Transformation**

Ở nhóm Transformation thì sẽ thực hiện các nhiệm vụ trên các thuộc tính như nhân rộng,thu nhỏ,giảm kích thước,chuyển đổi,sửa đổi các thuộc tính theo mong muốn của người thực hiện.Có rất nhiều phương thức được hổ trợ thực hiện các nhiệm vụ này,một số phương thức phổ biển kể đến như : [Tokenizer](https://spark.apache.org/docs/latest/ml-features.html#tokenizer), [StopWordsRemover](https://spark.apache.org/docs/latest/ml-features.html#stopwordsremover), [PCA](https://spark.apache.org/docs/latest/ml-features.html#pca), [StringIndexer](https://spark.apache.org/docs/latest/ml-features.html#stringindexer),[IndexToString](https://spark.apache.org/docs/latest/ml-features.html#indextostring),[OneHotEncoderEstimator](https://spark.apache.org/docs/latest/ml-features.html#onehotencoderestimator), [VectorIndexer](https://spark.apache.org/docs/latest/ml-features.html#vectorindexer), [VectorAssembler](https://spark.apache.org/docs/latest/ml-features.html#vectorassembler), [Imputer](https://spark.apache.org/docs/latest/ml-features.html#imputer),….

* **Selection**

Ở nhóm Selection sẽ hỗ trợ cho việc trích xuất hay chọn lựa ra tập thuộc tính con từ một tập thuộc tính lớn.Các phương thức hỗ trợ ở nhóm này kể đến như : [VectorSlicer](https://spark.apache.org/docs/latest/ml-features.html#vectorslicer), [RFormula](https://spark.apache.org/docs/latest/ml-features.html#rformula), [ChiSqSelector](https://spark.apache.org/docs/latest/ml-features.html#chisqselector) ,..

* **Locality Sensitive Hashing (LSH)**

Ở nhóm này thì sẽ hỗ trợ việc kết hợp tất cả các khía cạnh của việc biến đổi các thuộc tính với các thuật toán khác.Là một nhóm quan trọng của kĩ thuật băm,thường được sử dụng trong bài toán gom cụm (Clustering) dùng để tìm hàng xóm gần nhất và phát hiện ra các ngoại lệ của bộ dữ liệu.

## Pipeline

### Các thành phần của Pipeline trong Spark Mllib

* **Transformer**

Là một khái niệm trừu tượng đại diện cho bước tiền xử lý dữ liệu với các công việc biến đổi thuộc tính và học mô hình thông qua phương thức transform()

Ví dụ :

* + Việc biến đổi các thuộc tính được lấy từ Dataframe ban đầu và nó sẽ thực hiện việc đọc các cột và ánh xạ thành một cột mới và đầu ra là một Dataframe mới đã được thêm một cột thuộc tính được ánh xạ từ các thuộc tính của Dataframe ban đầu.
  + Việc học mô hình được lấy từ Dataframe ban đầu và nó sẽ thực hiện đọc cột chứa các vector thuộc tính và dự đoán các nhãn ứng với mỗi vector thuộc tính và đầu ra sẽ là một Dataframe mới với một cột giá trị các nhãn đã dự đoán được thêm vào Dataframe.
* **Estimator**

Cũng là một dạng khác của Transformer.Bằng việc thực hiện phương thức fit() một thuật toán bất kì lên một tập dữ liệu để tập luyện cho thuật toán.Ví dụ như việc học thuật toán LogisticRegression là một Estimator và gọi phương thức fit() để tập luyện cho mô hình LogisticRegressionModel.

* **Evalutor**

Là một thành phần dùng để đánh giá hiệu suất của mô hình tạo ra dự trên một số độ đo nhất định như ROC , RMSE,…Giúp cho việc tự động hóa điều chỉnh mô hình bằng việc so sánh các mô hình và chọn ra mô hình tốt nhất cho việc tạo ra dự đoán.Ví dụ như BinaryClassificationEvaluator,Cross-validatior,..

### Pipeline và cách hoạt động

* Pipeline đại diện cho một quy trình thực hiện học máy bao gồm nhiều các giai đoạn ( giai đoạn ở đây là các bước Transformer và Estimator) được thực hiện tuần tự để xử lý và học từ tập dữ liệu.
* Cách hoạt động của pipeline: Khi một pipline được xác định là một tuần tự các giai đoạn với mỗi giai đoạn có thể là Transformer hoặc là Estimator.Những giai đoạn này sẽ được thực hiện theo thứ tự và Dataframe đầu vào sẽ được biến đổi qua mỗi giai đoạn.Đối với mỗi giai đoạn Transformer thì phương thức transform( ) sẽ được gọi lên Dataframe.Đối với mỗi giai đoạn Estimator thì phương thức fit( ) sẽ được gọi để tạo ra một Transformer (là một phần của PipelineModel) và phương thức transform() của Transformer sẽ được gọi lên Dataframe.

### ML Persistence

* Thực hiện việc lưu mô hình hoặc một pipeline đã được xây dựng và khi cần sử dụng lại có thể gọi lại mô hình hay pipeline đã lưu.

# Demo phân tích dataset với Mllib

## Thông tin dữ liệu

* Tên dataset: Black Friday
* Nguồn dataset: <https://www.kaggle.com/mehdidag/black-friday>
* Mô tả dữ liệu:

Dataset này là dữ liệu các giao dịch được thu thập trong một cửa hàng bán lẻ. Cửa hàng muốn biết thêm về hành vi mua hàng của khách hàng đối với những sản phẩm khác nhau.

* Thông số dataset:
  + Số thuộc tính:12
  + Số dòng:537577
  + Dung lượng file:5MB
  + Bảng thông tin thuộc tính

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Loại dữ liệu** | **Mô tả thuộc tính** | **Phân khúc** |
| 1 | User\_ID | Numeric | Discrete | Id khách hàng | Khách hàng |
| 2 | Product\_ID | Numeric | Discrete | Id sản phẩm | Sản phẩm |
| 3 | Gender | Categorical | Nominal | Giới tính | Khách hàng |
| 4 | Age | Categorical | Ordinal | Tuổi | Khách hàng |
| 5 | Occupation | Categorical | Nominal | Nghề nghiệp  (Được giấu) | Khách hàng |
| 6 | City\_Category | Categorical | Ordinal | Danh mục thành phố (A, B, C) | Thành phố |
| 7 | Stay\_In\_Current\_City\_Years | Categorical | Ordinal | Số năm ở tại thành phố hiện tại | Thành phố |
| 8 | Marital\_Status | Categorical | Ordinal | Tình trạng hôn nhân | Khách hàng |
| 9 | Product\_Category\_1 | Categorical | Nominal | Danh mục sản phẩm (được giấu) | Sản phẩm |
| 10 | Product\_Category\_2 | Categorical | Nominal | Sản phẩm cũng có thể thuộc danh mục khác (được giấu) | Sản phẩm |
| 11 | Product\_Category\_3 | Categorical | Nominal | Sản phẩm cũng có thể thuộc danh mục khác (được giấu) | Sản phẩm |
| 12 | Purchase | Numeric | Continuous | Số tiền mua | Sản phẩm |

## Các yêu cầu cài đặt chung

* Python và các thư viện hỗ trợ như pandas , p4j , pyspark , findspark,pip
* Jupyter
* Spark version 2.4.4 trở lên
* Chạy trên hệ điều hành Ubuntu

## Source code demo phân tích dataset Black Friday

Bạn có thể nhìn thấy source code tại đường link bên dưới :

<https://github.com/phamvanhuu0903/Bigdata-Project/blob/master/Pre-Report.ipynb>

# Tài liệu tham khảo

* <https://viblo.asia/p/tong-quan-ve-apache-spark-cho-he-thong-big-data-RQqKLxR6K7z>
* <https://www.edureka.co/blog/spark-mllib/>
* <https://www.kaggle.com/sdolezel/black-friday/kernels>
* <https://spark.apache.org/docs/latest/ml-guide.html>

*The end*