HPCC SYSTEMS MACHINE LEARNING

Machine Learning

Machine Learning is the study of computer algorithms that improve automatically through experience.

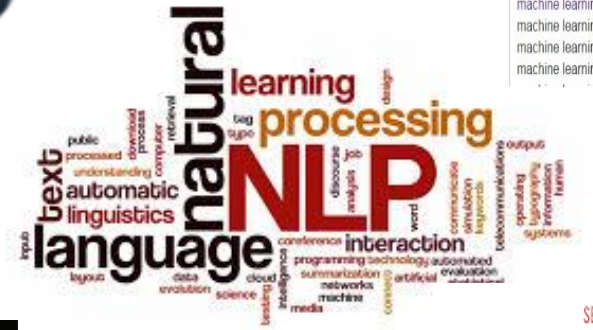
-- Tom Mitchell, 1997

Autonomous Vehicles

AlphaGo

Machine Learning in Daily Life

Google



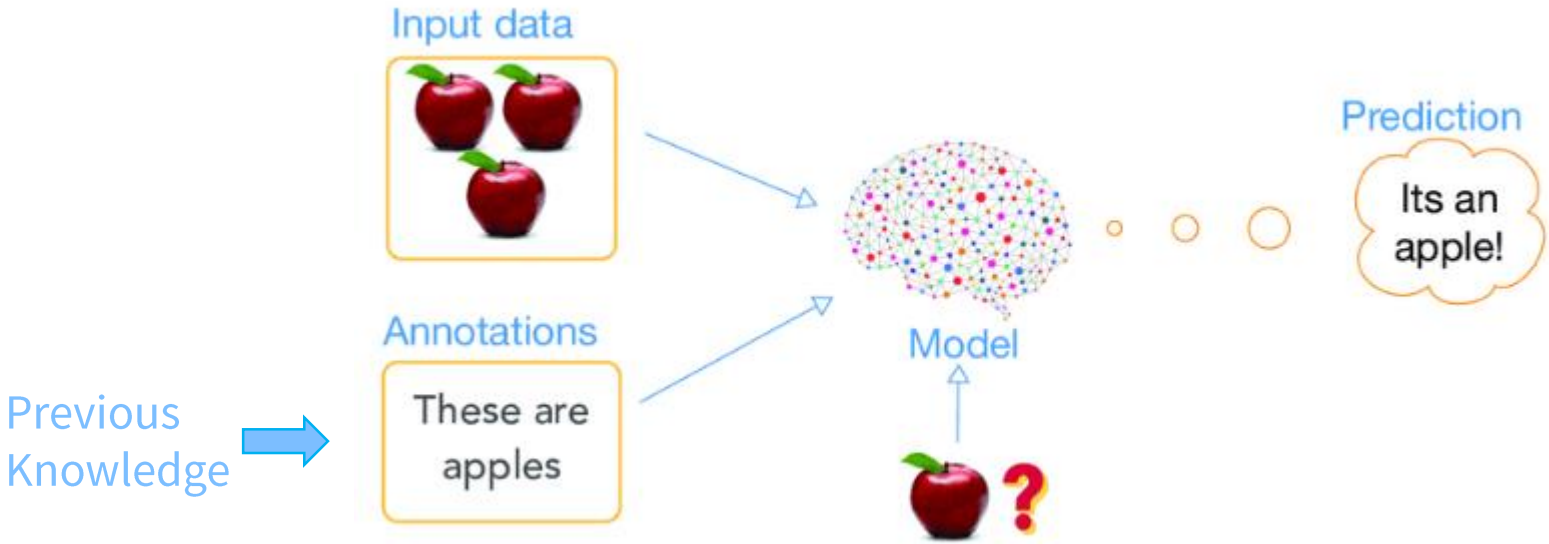
SENTIMENT ANALYSIS

Discovering people opinions, emotions and feelings about a product or service

Machine Learning

- **Supervised**
- Unsupervised

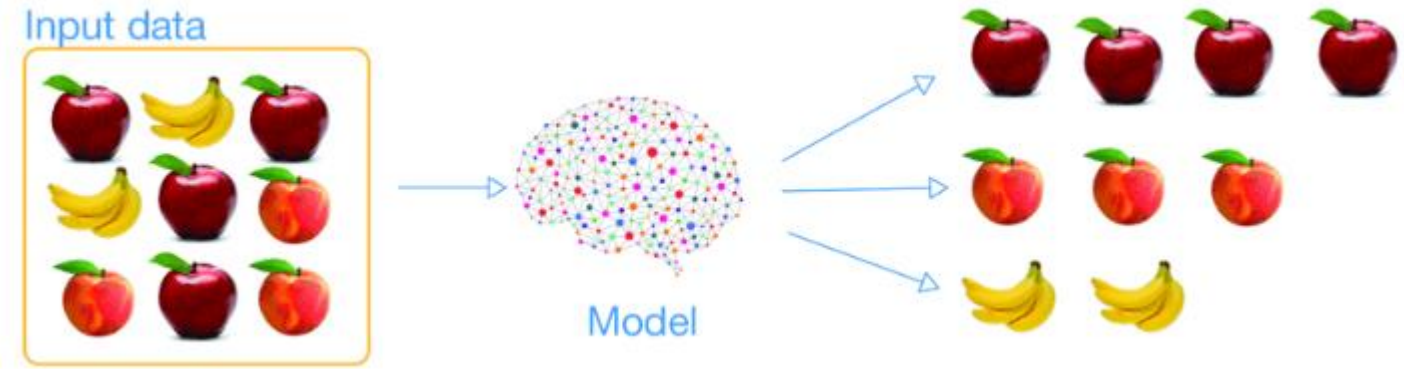
Supervised Learning



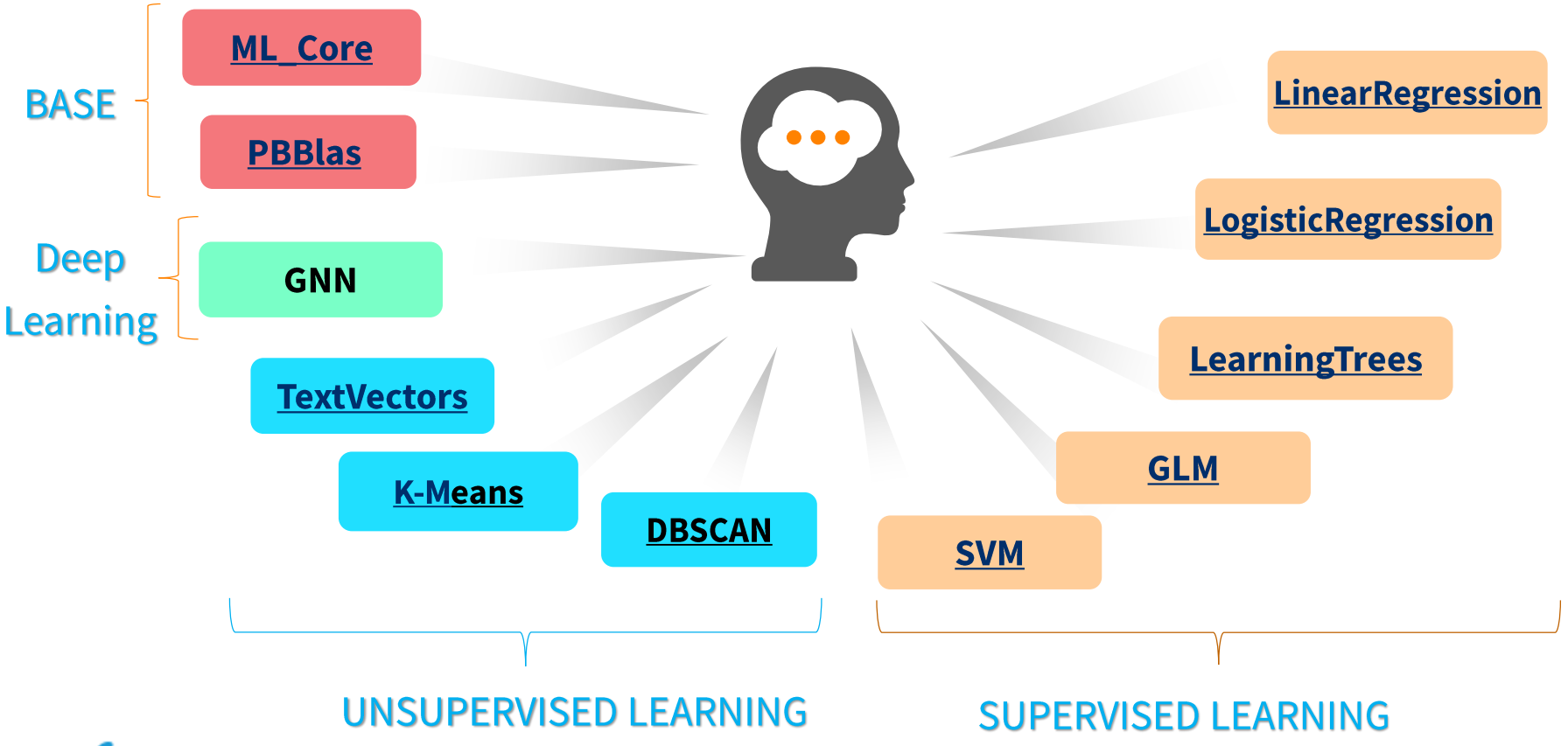
Machine Learning

- Supervised
- **Unsupervised**

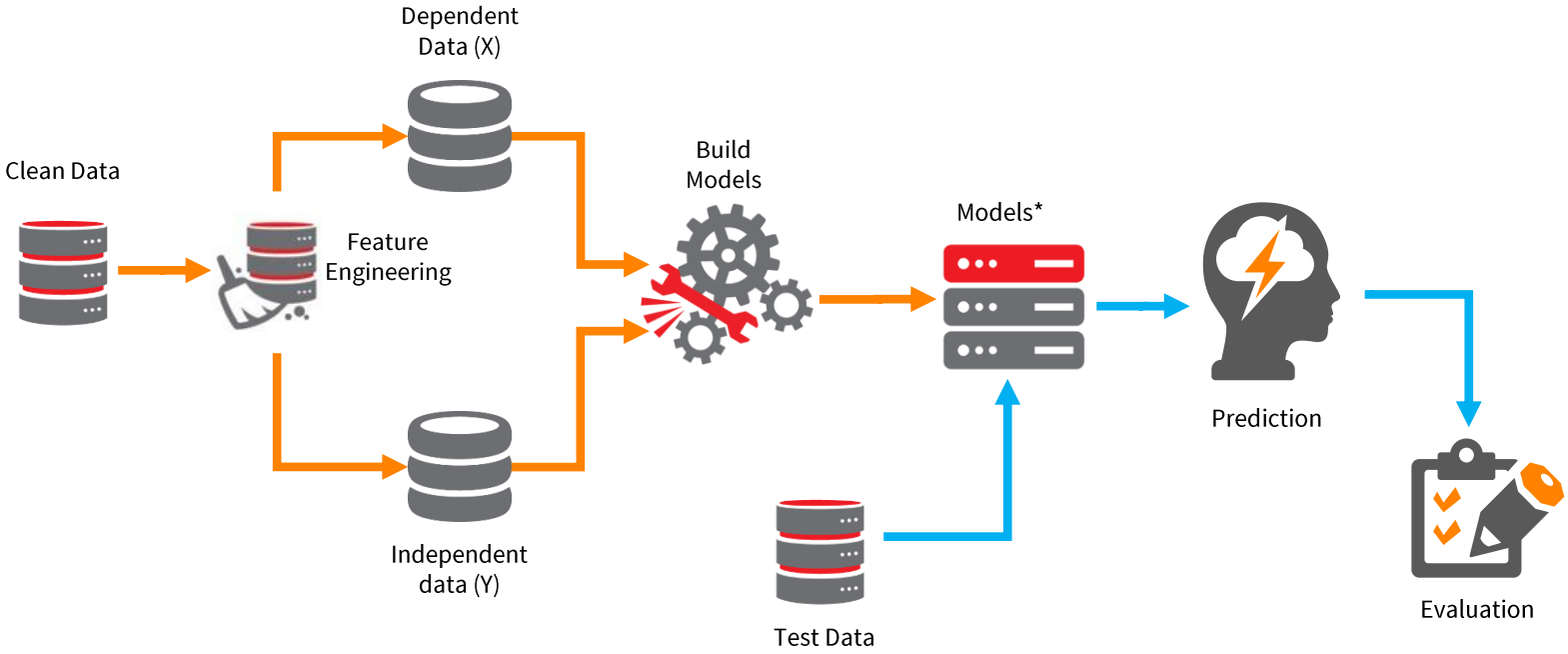
Unsupervised Learning



HPCC Systems Machine Learning Library



Machine Learning Pipeline



Machine Learning

- **Regression**
- Classification
- Clusering

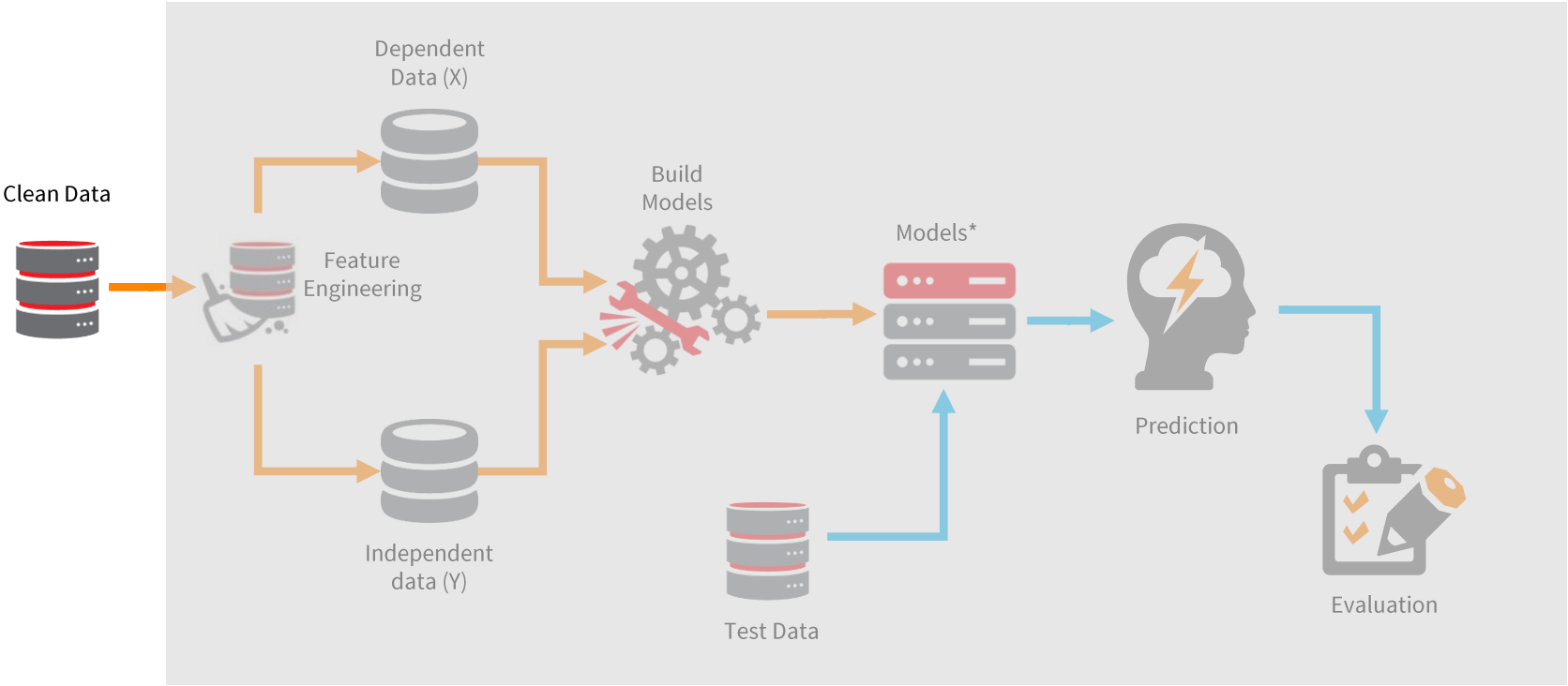
New York City Taxi Trips

NYC Taxi Data

48 GB
241M RECORDS
JAN 2015 – JUN 2016
16 MONTH
W/ WEATHER INFO

HPCC SYSTEMS®

Machine Learning Pipeline



Set up Machine Learning environment

LexisNexis®
RISK SOLUTIONS

Machine Learning Workspace:

https://ide.hpccsystems.com/workspaces/share/4adff453-e8f7-4818-a7fd-1a82cfd0b21c

HPCC Systems Cluster: http://40.76.26.67:8010

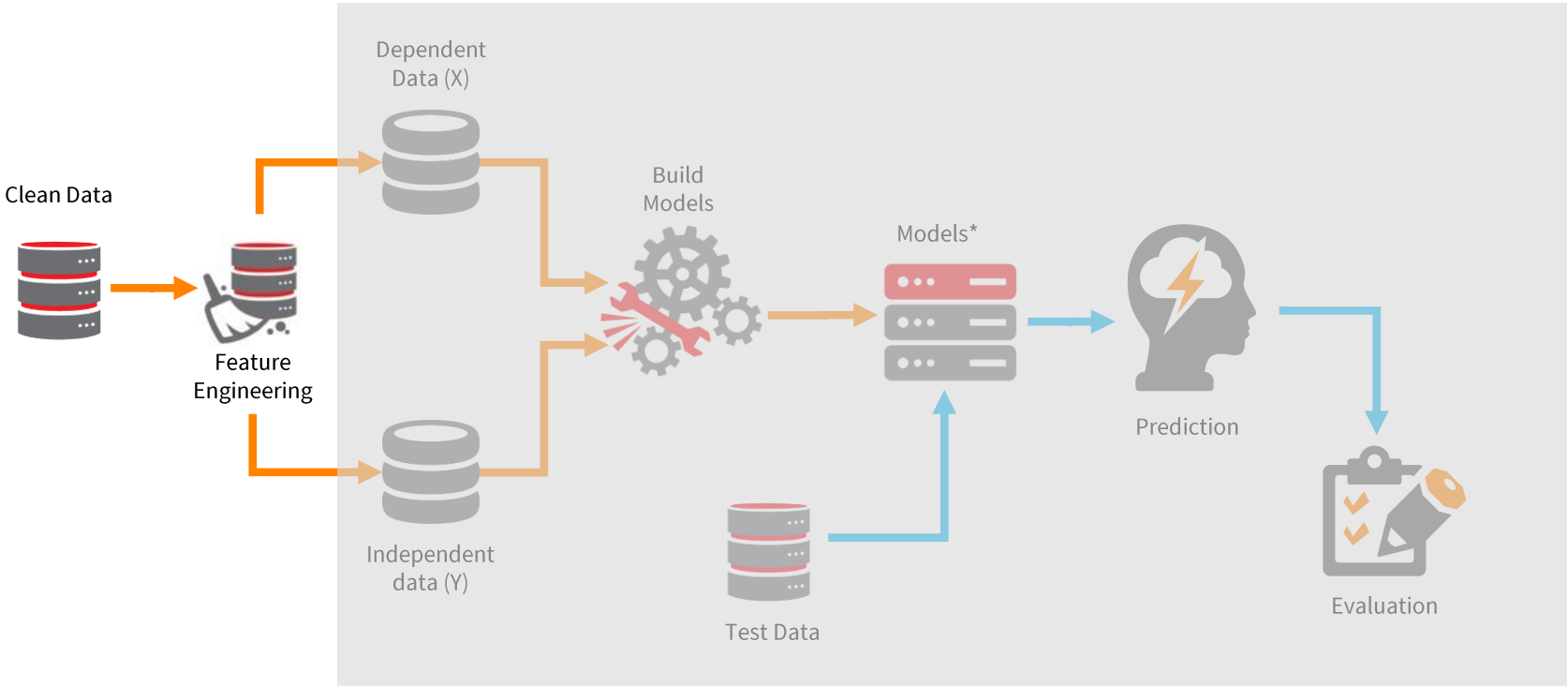
Machine Learning – Task 1

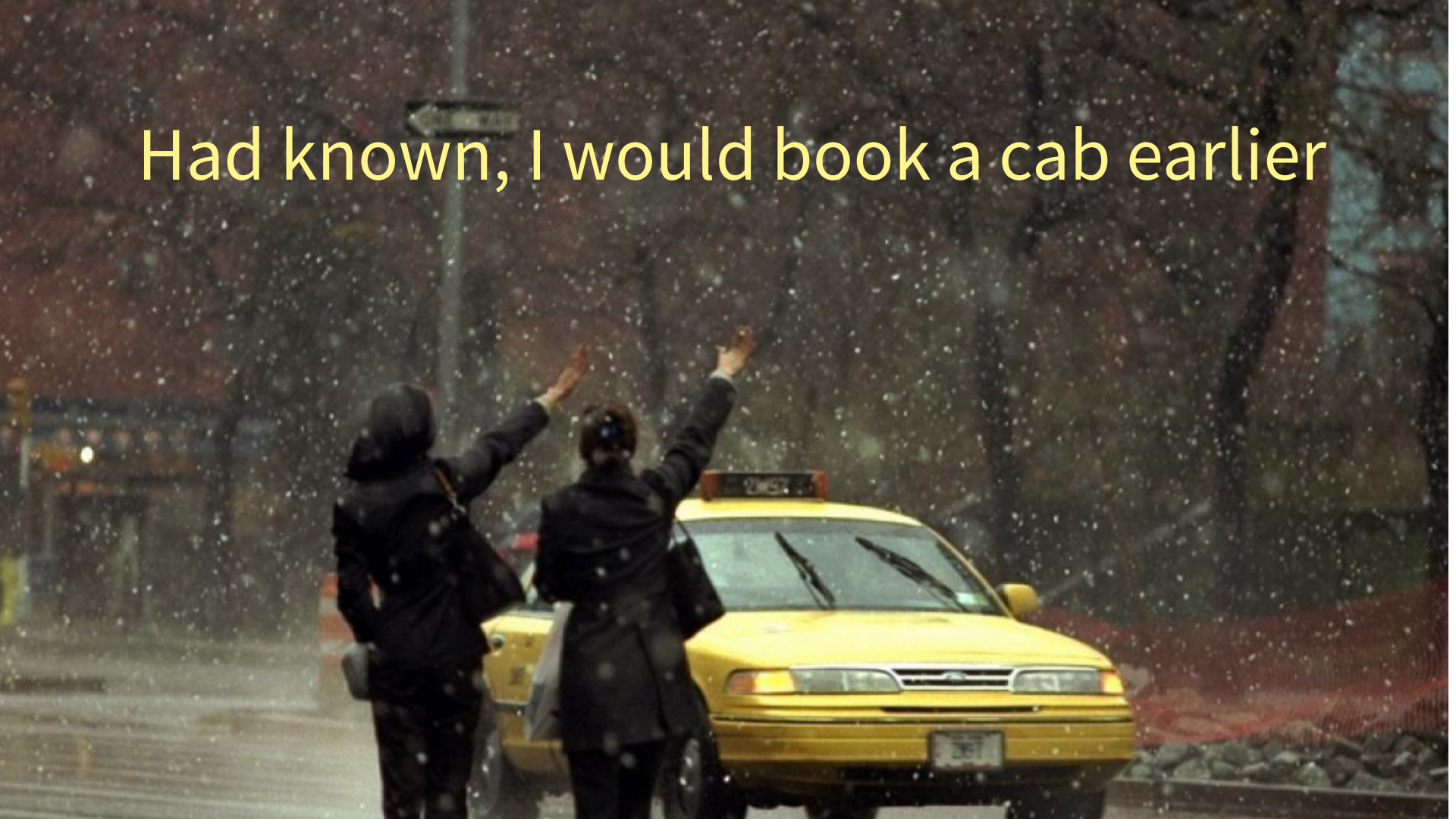
LexisNexis®
RISK SOLUTIONS

Based on A_Read_Data.ecl:

1. How many records in the raw New York Taxi Trips dataset?
2. Which day has the most taxi trips?

Machine Learning Pipeline




Had known, I would book a cab earlier

Feature Engineer

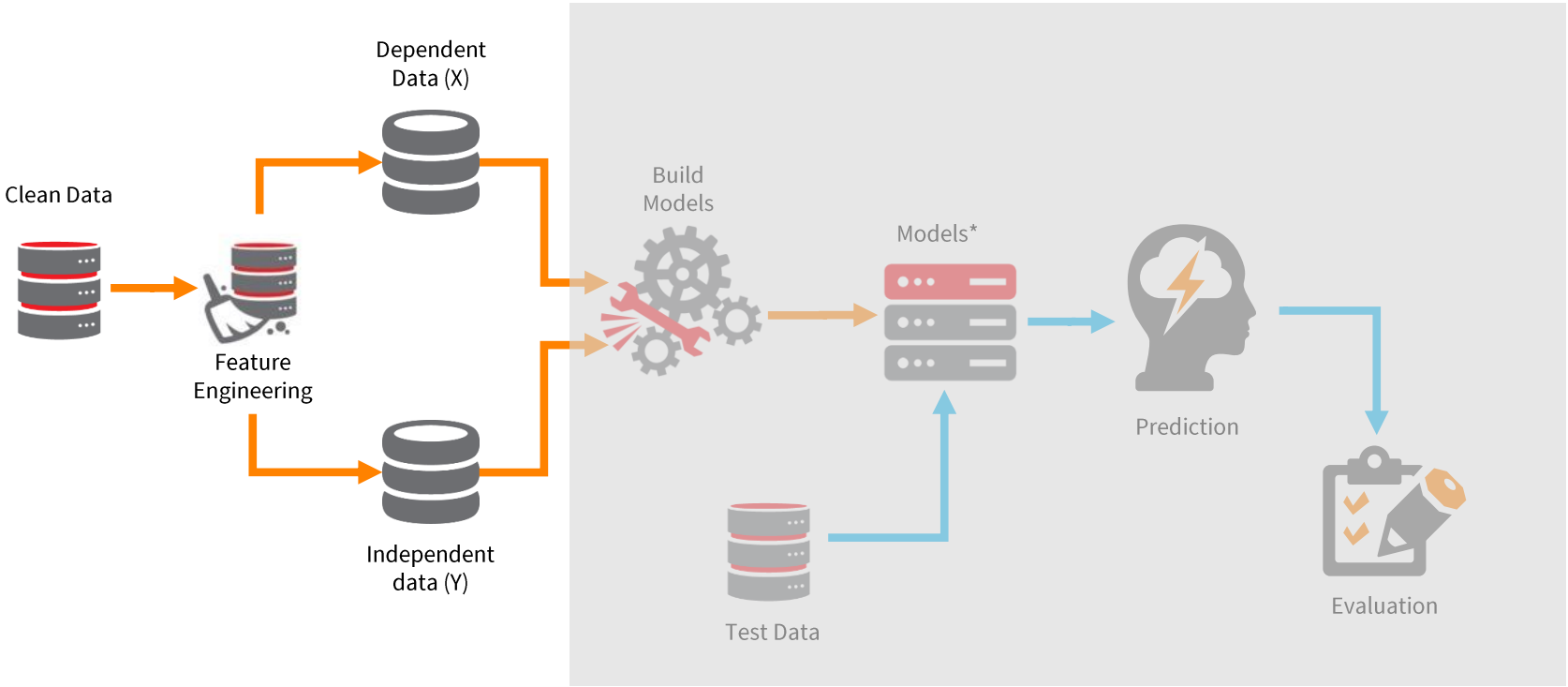
Machine Learning – Task 2

LexisNexis®
RISK SOLUTIONS

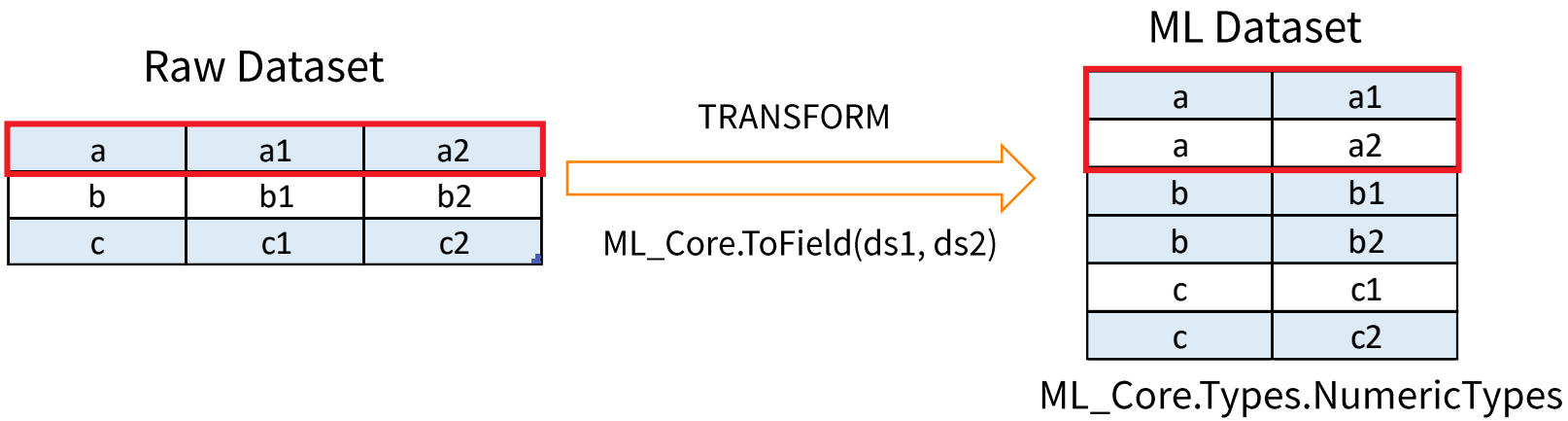
Based on B_Feature_Enginner.ecl:

1. How many records in the engineered dataset?
2. How many taxi trips on July 4th, 2015?

Machine Learning Pipeline



Machine Learning Data Structures



ML_Core Bundle

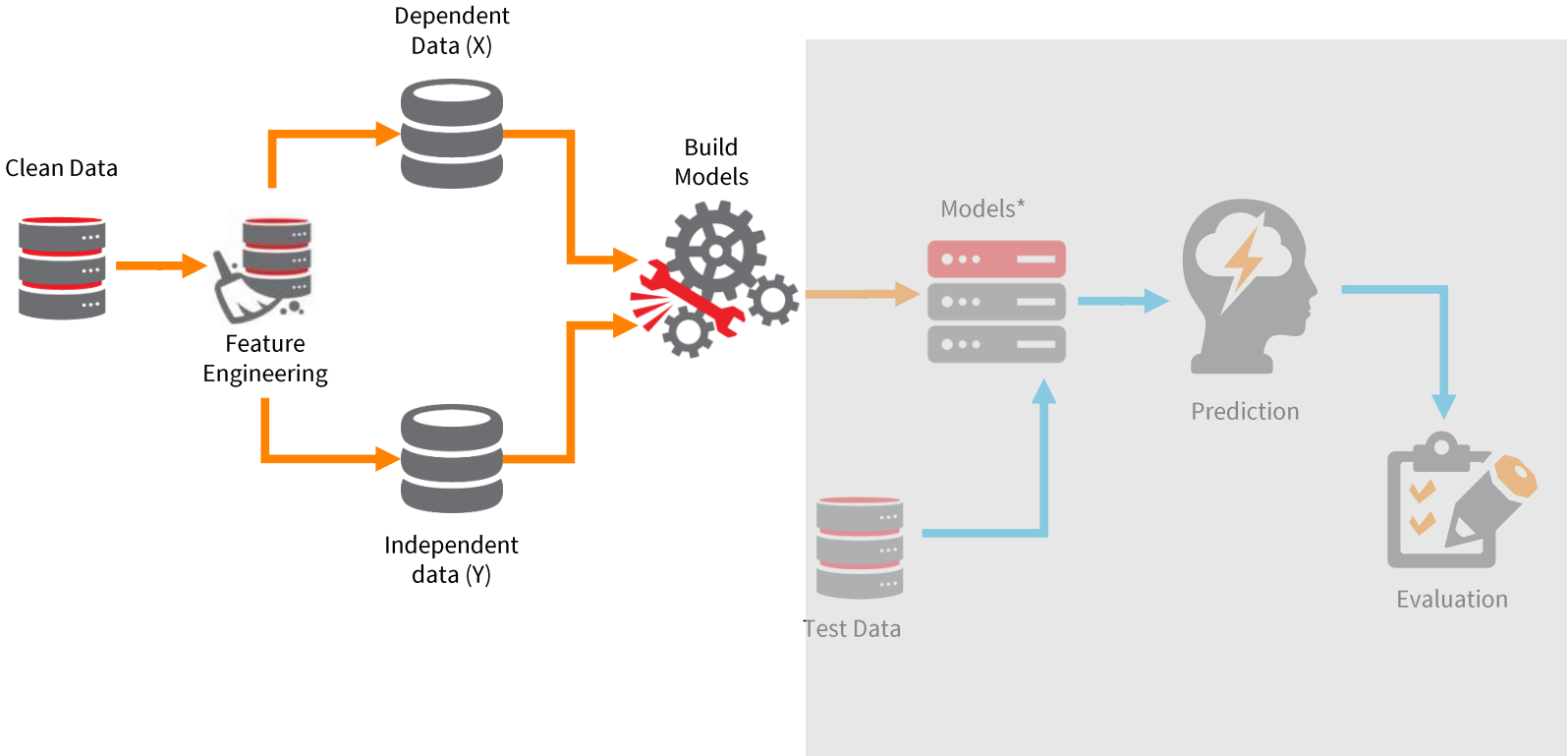
- Prerequisite for all HPCC Systems production machine learning bundles
- Main attributes:
  - Definitions for common data types
    - ML_Core.Types
  - Data manipulation utilities
    - ToField()
    - Discretize()
  - Data examination
    - FieldAggregates(): min, max, mean, var, std
- ML_Core Bundle: https://github.com/hpcc-systems/ML_Core

Machine Learning – Task 3

LexisNexis®
RISK SOLUTIONS

Fix the error in C_Transform.ecl to correctly transform training data to NumericField format.

Machine Learning Pipeline



Linear Regression

```
//Reading enhanced data
enhancedData := D_Data_Enhancement.enhancedData;

//Transform to Machine Learning Dataframe, such as NumericField
ML_Core.ToField(enhancedData, train);

// split into input (X) and output (Y) variables
X := train(number < 4);
Y := train(number = 4);

//Training LinearRegression Model
lr := LROLS.OLS(X, Y);

//Prediction
predict := lr.predict(X);
OUTPUT(predict);
```

wi	id	number	value
1	1	1	1
1	1	2	3
1	1	3	0.001289982354828361
1	1	4	374040
1	2	1	1
1	2	2	1
1	2	3	0.05718114840201266
1	2	4	416962
1	3	1	1
1	3	2	2
1	3	3	0.008881908280789124
1	3	4	224097

ML Dataframe: NumericField

wi	id	number	value
1	1	4	383492.0584366489
1	2	4	358001.6615743856

Linear Regression Result Example

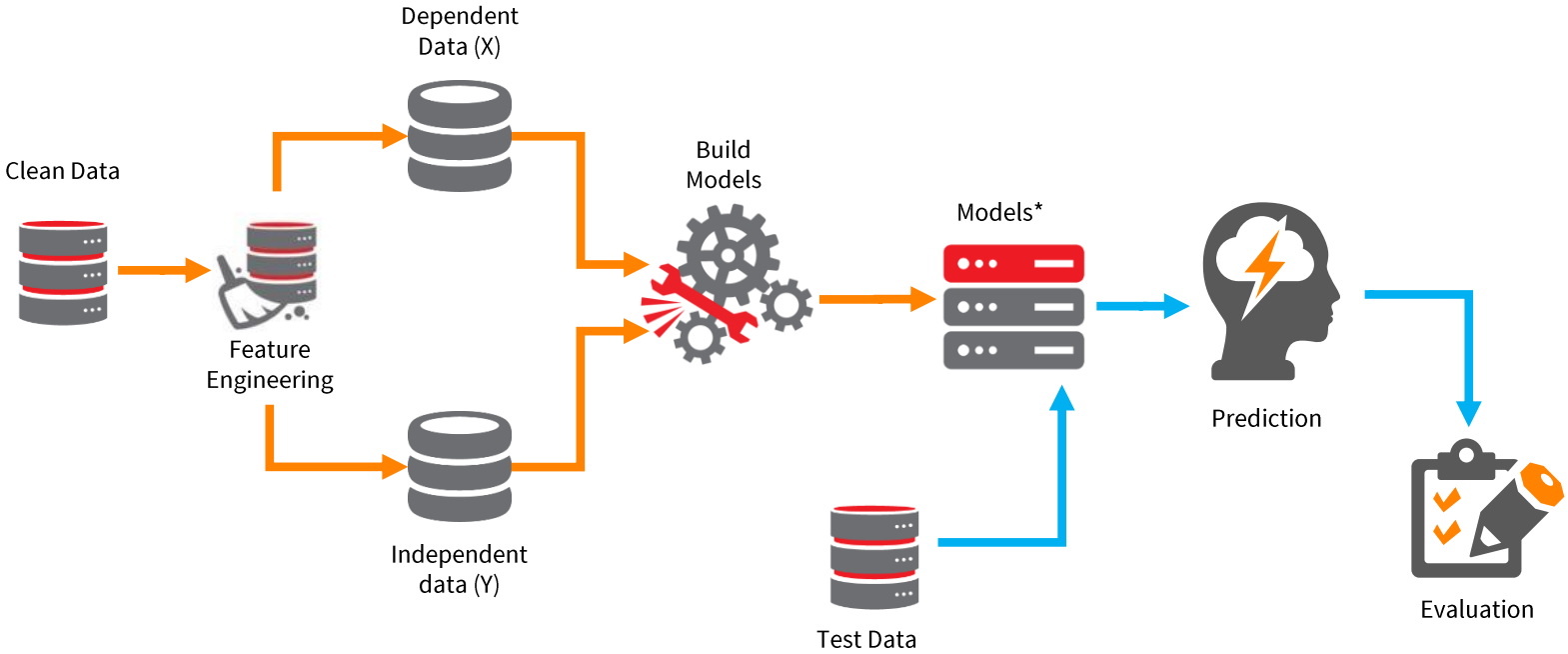
Machine Learning – Task 4

LexisNexis®
RISK SOLUTIONS

Complete the tasks in D_Training.ecl to train a Linear Regression model

```
IMPORT ML_Core;
IMPORT ML_Core.Types AS Types;
IMPORT ML_Core.Analysis AS Analysis;
IMPORT LinearRegression AS LROLS;


// Read training data
NFTrain  := DATASET('~NCF2021::ML::NFTrain', Types.NumericField, FLAT);
// Independent and dependent split
trainInd := NFtrain(number < 5 );
trainDep := PROJECT(NFtrain(number = 5 ), TRANSFORM(Types.NumericField, SELF.number := 1, SELF := LEFT));

// Train Linear Regression model
m := LROLS.OLS(trainInd, trainDep);

// Read test data
NFTest   := DATASET('~NCF2021::ML::NFTest',  Types.NumericField, FLAT);
// Independent and dependent split
testInd := NFtest(number < 5 );
testDep := PROJECT(NFtest(number = 5 ), TRANSFORM(Types.NumericField, SELF.number := 1, SELF := LEFT));

// Predict with test data
result := m.Predict(testIND);
OUTPUT(result[1..100]);

// Evaluate model
evaluation :=  Analysis.Regression.Accuracy(result, TestDep);
OUTPUT(evaluation);
```

Machine Learning Pipeline



Machine Learning

- Regression
- Classification
- **Clustering**

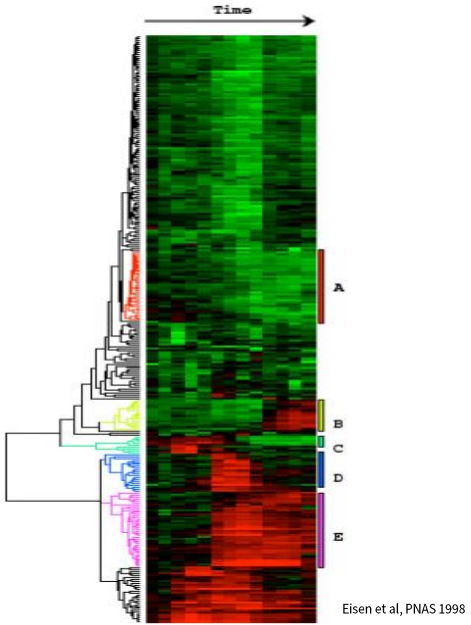
K-Means

- Unsupervised Machine Learning (ML) algorithms
- Automatically find the clusters/groups of the data without previous knowledge
- Highly Scalable Parallelized for Big Data machine learning challenge

Applications

Claim\Customer segmentation

Image segmentation

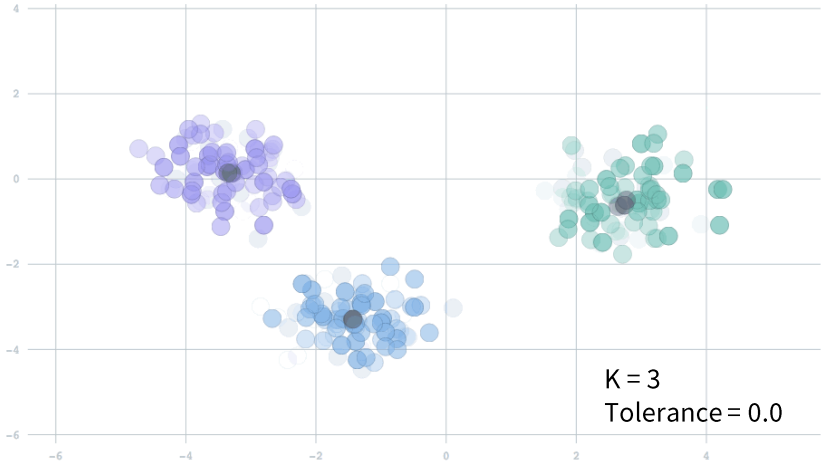


Clustering gene expressions

K-Means

➢ **KMEANS**

- Most popular clustering method [3]
- Highly Scalable Parallelized
- Parametric: K, Tolerance
- Sensitive to Initialization
- Spherical Clusters
- Sensitive to Outliers
- Curse of Dimensionality



Apply K-Means

Step 1 Import K-Means bundle

```
IMPORT KMeans as KM;
```

Step 2 Train K-Means Model

```
Model := KM.KMeans(Max_iterations,Tolerance).Fit( Samples, InitialCentroids));
```

Optional

Required

Step 3 Predict the cluster index of the new samples (Optional)

```
Labels := KM.KMeans().Predict(Model, NewSamples);
```

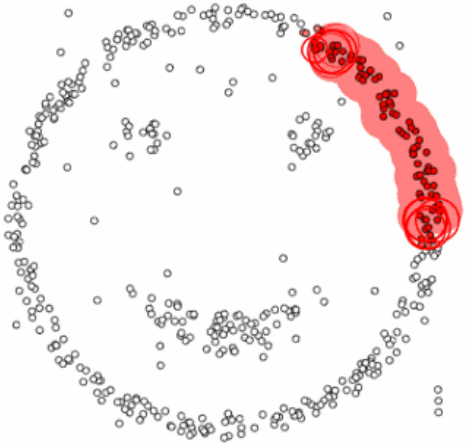
Let’s Play With The Code

LexisNexis®
RISK SOLUTIONS

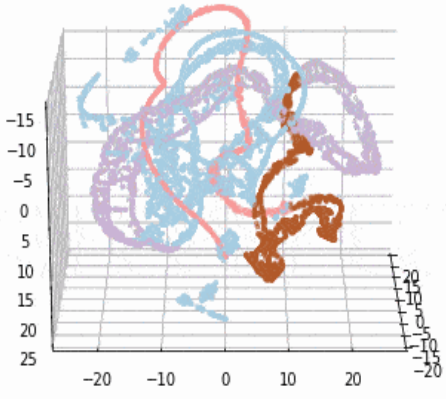
DBSCAN

➢ DBSCAN

- Density-Based Clustering Metho
- Highly Scalable Parallelized
- Parametric: epsilon, minPoints
- Sensitive to Initialization
- Random Shapes Clusters
- Outliers Detection
- Sensitive to Density Variance
- Curse of Dimensionality

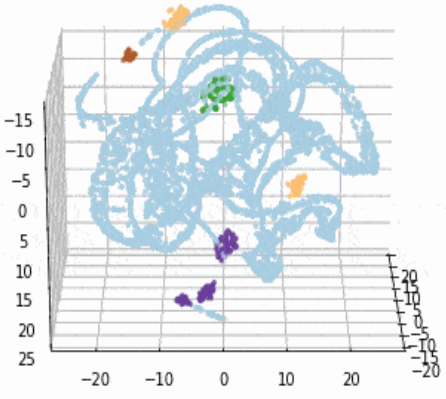


KMeans vs. DBSCAN



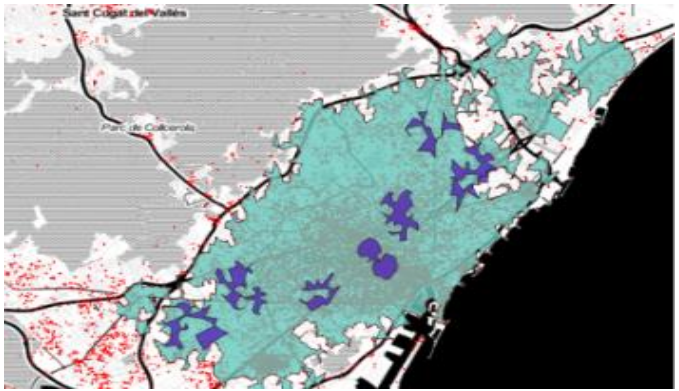
KMeans

- Clusters Shape
- Cluster Size
- Model Parameters
- Number of Clusters (Fixed vs. Variable)
- Outlier Detection
- Curse of Dimensionality

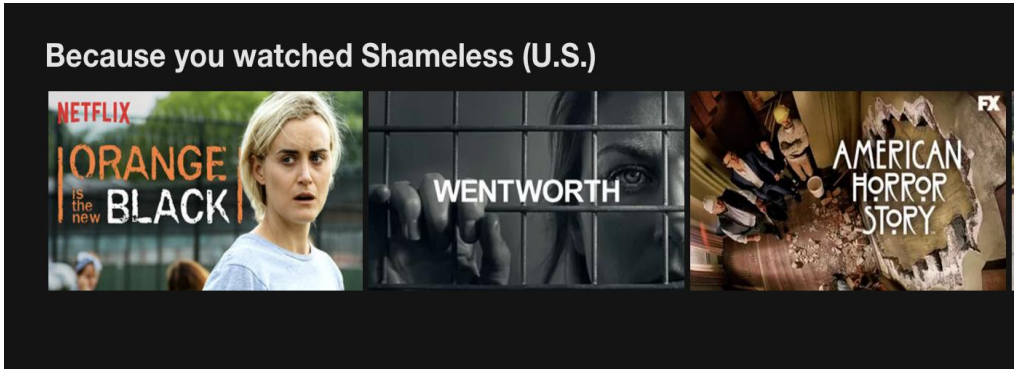


DBSCAN

Application Domains



Clustering Demographic/Geospatial Data



Recommendation System

Machine Learning – Task 5

LexisNexis®
RISK SOLUTIONS

Complete the code in DBSCAN_Clustering.ecl to train a DBSCAN model

Can you apply the Machine Learning models you just learnt to Flight Data?

LexisNexis®
RISK SOLUTIONS

More Information

1. Introduction of HPCC Systems Machine Learning Library
2. HPCC Systems Machine Learning Library on Github
3. Myriad Interface Tutorial
4. LearnECL

Q&A

Lili Xu
Lili.xu@Lexisnexisrisk.com