

# London Weather Prediction using Apache Spark



**COURSE:**

**CSYE 7200: BIG-DATA ENGINEERING USING SCALA**

**PROFESSOR:**

**ROBIN HILLYARD**

**TEAM 9 :**

**RENTENG HUANG, SHUANGSHUANG XU, BALAJI MUDALIYAR**

# Goals of the project

- ▶ To predict the future weather condition of London city.
- ▶ To develop Apache - Spark Scala code to clean, train, model the data.
- ▶ To use Apache - Spark Scala MLlib(Machine Learning Library) to predict the weather.
- ▶ We will be implementing the UI using play framework.
- ▶ Collaborative learning and knowledge sharing
- ▶ Delivering each milestone on time

# Use cases

## Model input

- Date

## Model Output

- Temperature, Pressure

# Use cases

## Model input

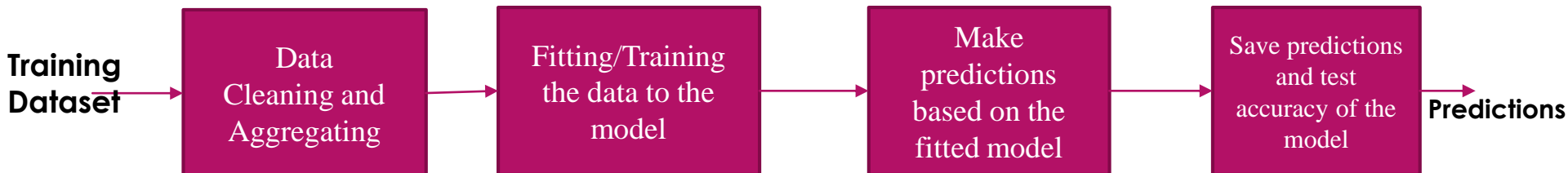
- Date, Mean Humidity, Mean Pressure

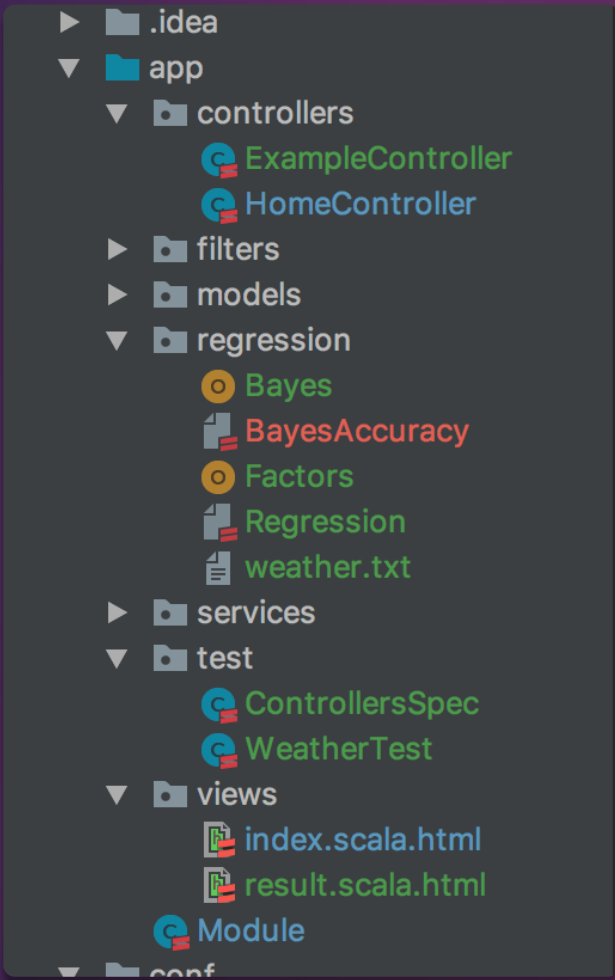
## Model Output

- Weather(Sunny or Rainy)

# Methodology

- ▶ The cleaning of the training dataset
- ▶ Fitting of the data to the model.(Linear Regression)
- ▶ Making predictions based on fitted model
- ▶ Calculating accuracy of the model
- ▶ Save predictions





- ▶ .idea
- ▼ app
  - ▼ controllers
    - ExampleController
    - HomeController
  - ▶ filters
  - ▶ models
  - ▼ regression
    - Bayes
    - BayesAccuracy
    - Factors
    - Regression
    - weather.txt
  - ▶ services
  - ▼ test
    - ControllersSpec
    - WeatherTest
  - ▼ views
    - index.scala.html
    - result.scala.html
  - Module
  - ▼ conf

# Machine Learning Algorithm

# Data source

- ▶ **Weather Underground :**  
([http://api.wunderground.com/api/API\\_KEY/history\\_DATA/q/UK/London.json](http://api.wunderground.com/api/API_KEY/history_DATA/q/UK/London.json))
- ▶ **3700 record dataset containing weather information of London city**

date	meantemp	maxtemp	mintemp	meanpres	maxpressu	minpressu	meanhum	maxhumid	minhumid	rain
1/1/2010	1	3	-1	1005.76	1010	1002	83	93	70	0
1/2/2010	2	5	-1	1014.83	1019	1010	81	93	70	1
1/3/2010	0	2	-2	1022.69	1024	1019	89	93	81	1
1/4/2010	-1	2	-4	1018.05	1023	1012	87	100	70	0
1/5/2010	-1	2	-4	1004.19	1011	1000	86	100	65	0
1/6/2010	0	2	-3	1003.11	1007	1000	91	100	75	0
1/7/2010	-1	1	-3	1009.64	1016	1007	89	93	86	0
1/8/2010	-2	0	-4	1021.53	1024	1017	94	100	87	0
1/9/2010	-1	1	-3	1021.45	1023	1018	90	100	75	0
1/10/2010	1	1	1	1017.12	1018	1017	93	93	93	1
1/11/2010	1	2	0	1018.79	1019	1018	92	100	87	1
1/12/2010	2	2	1	1009.15	1019	1002	86	93	75	0
1/13/2010	0	1	0	1000.69	1002	999	88	93	81	1
1/14/2010	2	4	0	1004.97	1013	1001	97	100	93	1
1/15/2010	4	6	2	1017.97	1020	1013	93	100	87	0

## PROJECT DETAILS

DATE	MILESTONE
15-Mar	Project Start Planning & Data work
23-Mar	Data finding & Cleaning(complete)
24-Mar	Spark Self-Learning, Mlib study
1-Apr	Coding
5-Apr	Implementation
9-Apr	Testing
13-Apr	Final Presentation & Documentation
15-Apr	Project End

Milestones/sprints



# Programming in Scala and code repository

- ▶ **Data extraction part was done in Python.**
- ▶ **Most part of the project will be programmed in Scala including**
  - ▶ Cleaning
  - ▶ Splitting(Training& Testing)
  - ▶ Fitting/Training Data to Model
  - ▶ Predictions
  - ▶ Accuracy Calculation
- ▶ **Code repository : GitHub**

<https://github.com/001239511ShuangShuangXu/csye7200-spring2018-group9>

```
1.0 0.0 [84.0, 1029.6]
0.0 0.0 [75.0, 1012.58]
1.0 1.0 [81.0, 1009.56]
1.0 0.0 [80.0, 1018.69]
1.0 1.0 [92.0, 1016.6]
0.0 0.0 [66.0, 1022.84]
0.0 0.0 [71.0, 1026.65]
0.0 0.0 [60.0, 1018.78]
0.0 1.0 [74.0, 1019.49]
1.0 1.0 [77.0, 1003.2]
0.0 0.0 [54.0, 1020.91]
0.0 0.0 [69.0, 1011.86]
0.0 0.0 [67.0, 1012.76]
1.0 1.0 [86.0, 1016.08]
Accuracy=0.6097560975609756
```

```
Process finished with exit code 0
```

# Accuracy

Accuracy of average weather condition for the test dataset

# 11/18/2017

weather	meanTemperature	maxTemperature	minTemperature	meanPressure	maxPressure	minPressure
Sunny	8	12	4	1021	1023	1018
Sunny	7	11	3	1020	1023	1017
Sunny	5	9	2	1019	1022	1017
Sunny	7	10	4	1016	1020	1014
Sunny	9	12	6	1014	1017	1011
Sunny	11	14	9	1009	1013	1004
Sunny	12	15	9	1006	1011	999

0	11/18/2017	6	9	3	1021.23	1025	1019	80	87	65	1
1	11/19/2017	4	7	2	1022.56	1023	1020	73	87	61	0
2	11/20/2017	9	13	5	1014.14	1022	1012	83	93	72	1
3	11/21/2017	13	15	11	1008.93	1012	1005	77	88	63	1
4	11/22/2017	14	16	12	997.86	1004	990	64	77	55	0
5	11/23/2017	12	16	7	998.6	1006	987	64	82	44	1
6	11/24/2017	3	8	-2	1010.19	1012	1005	76	93	66	0

# Play Framework Display

## Welcome to Weather Prediction Application

**4/11/2018**

weather	meanTemperature	maxTemperature	minTemperature	meanPressure	maxPressure	minPressure
Rainy	9	12	7	1008	1011	1005
Rainy	9	11	7	1008	1011	1005
Rainy	9	12	7	1008	1010	1005
Rainy	9	12	6	1009	1012	1005
Rainy	10	13	6	1011	1014	1008
Rainy	11	14	7	1012	1015	1009
Sunny	11	15	7	1013	1016	1010

4/11/2018	9	11	7	1006	1007	1005	91	100	87	1
4/12/2018	9	11	7	1002.73	1005	1001	94	100	87	0
4/13/2018	10	13	6	1006.91	1013	1002	86	93	71	1
4/14/2018	11	15	7	1015.98	1017	1014	77	93	55	0
4/15/2018	12	16	8	1009.47	1014	1007	85	100	59	1
4/16/2018	12	16	7	1013.81	1018	1009	68	93	48	0
4/17/2018	14	20	9	1020.49	1024	1018	60	82	45	0
4/18/2018	16	21	10	1026.74	1029	1025	71	88	56	0

# Root Mean Square Error

```
"D:\Program Files\Java\jdk1.8.0_152\bin\java" ...
```

```
The mean error of meanTemperature is :2.4698894590847393
```

```
The mean error of maxTemperature is :2.871180961448972
```

```
The mean error of minTemperature is :2.9325955676733706
```

```
The mean error of meanPressure is :7.313482280982774
```

```
The mean error of maxPressure is :6.746302467395636
```

```
The mean error of minPressure is :7.812621360990136
```



```
The mean error of meanHumidity is :7.62271988060607
```

```
The mean error of maxHumidity is :5.386243710795707
```

```
The mean error of minHumidity is :11.637982634403196
```

```
Process finished with exit code 0
```

# Acceptance criteria

- ▶ **We had a target of 4/7(day/day) accuracy. We ended up having model with 61% accuracy.** 
- ▶ **We had Targeted a Root Mean Square Error (RMSPE) of 20 but we ended up achieving a RMSE of 3 for temperature and 7 for pressure.** 



# THANK YOU

FOR LISTENING

End