# [https://avatars2.githubusercontent.com/u/4156894?v=3&s=100](http://www.calstatela.edu/centers/hipic) CIS3200 Term Project Tutorial

#### Authors: Francisco Cortes, Kevin Danao, Riker Santivong, Kevin De La Torre

#### Instructor: [Jongwook Woo](https://www.linkedin.com/in/jongwook-woo-7081a85)

#### Date: 05/22/2021

**Lab Tutorial**

05/22/2021

**NOAA Data Analysis using Elasticsearch & Azure ML**

**Objectives**

**List what your objectives are.** In this hands-on lab, you will learn how to:

* Get data from website and upload to Elasticsearch & Azure ML
* Create an Index Pattern.
* Create Visualizations on Kibana.
* Create a dashboard.
* Create Predict Model in Azure ML.

**Platform Spec**

|  |  |
| --- | --- |
| **ELASTICSEARCH​** |  |
| * Storage | 240 GB |
| * Memory | 8 GB |
| * Master memory | 1 GB |
| **KIBANA** |  |
| * Memory | 1 GB |
| **ML** |  |
| * Memory | 1 GB |
| **APM** |  |
| * Memory | 1 GB |
| **Total** |  |
| * Total Storage | 240 GB |
| * Total Memory | 11.5 GB |
| **Microsoft Azure ML** |  |
| * Max Storage Space | 10 GB |
| * Execution/performance | Single Node |
| * Max number of modules per experiment | 100 |
| * Max experiment duration | 1 hour per experiment |

Step 1: Get data manually by Downloading from the website

**This step is to retrieve the file.**

1. Go to: https://www.kaggle.com/noaa/noaa-global-historical-climatology-network-daily
2. Graphical user interface, text, email, website

   Description automatically generatedOnce there download the 14GB Compressed file.

Step 2: Importing CSV weather data into Kibana of Elasticsearch

**Import the data from CSV**

now that we have data to analyze and an Elasticsearch cluster that can do the analyzing, we need to

import and set up index.

1. Within Kibana, click on Machine Learning.
2. In the subnet, click on Data Visualizer

Graphical user interface, application, Word

Description automatically generated

1. Under Import Data, click Upload File.
2. Drag and drop the **“1894.csv”** file into the importer, which will show the data.

Graphical user interface, text, application, email

Description automatically generated

1. Scroll through the list of contents. Notice that the fields have the types automatically assigned by Kibana.

Graphical user interface

Description automatically generated

1. Click Import to upload and index the data. Uncheck “Create Index Pattern” and add the index name as “milan\_weather1894”, which should look as follows:

Graphical user interface, text, application, email

Description automatically generated

1. Then, select Import. If you don’t have any error, you will see the following pipeline:

Graphical user interface, text, application

Description automatically generated

1. Now import 1764.csv, 1774.csv, 1784.csv, 1794.csv, 1804.csv, 1814.csv, 1834.csv, 1844.csv, 1854.csv, 1864.csv, 1874.csv, 1884.csv and 1894.csv by following the above steps 2: page 3 – page 5.
2. When you finish uploading all 14 files, select **Index Pattern Management** to create index pattern above. And, select “**Create Index pattern**” in the following page.
3. You need to create an index pattern that contains all these 14 data files. For example, the general index pattern name can be milan\_weather\* to include all 14 files.

Graphical user interface, text, application, email

Description automatically generated

1. Now type in your index name: milan\_weathe\*. Then, select Next step:

Graphical user interface, text, application, email

Description automatically generated

1. Then, select Create Index pattern.

Graphical user interface, text, application

Description automatically generated

1. You will see the following page successfully.

Graphical user interface, text, application, email

Description automatically generated

Step 3: Creating Kibana visualizations using Different Graphs.

1. On the left side navigation, click on Visualize. Then click on “Create Visualization.”

Graphical user interface, text, application, Teams

Description automatically generated

1. Select Vertical Bar chart.

Graphical user interface, application

Description automatically generated

1. In the New Line /Choose a source prompt, select milan\_weathe\*

Graphical user interface, text, application, email

Description automatically generated

1. Let’s create **Vertical Bar** chart to analyze **Max Temperatures of Milan**
2. In **metrics** section, input the following:

* **Aggregation:** Max
* **Field:** Temperature

Graphical user interface, application

Description automatically generated

1. In **Buckets** section, Click “add” and select “X-axis”. Then, input the following:

* **Aggregation:** Range
* **Field:** Date

|  |  |
| --- | --- |
| 17640301 | 17640531 |
| 17740301 | 17740531 |
| 17840301 | 17840531 |
| 17940301 | 17940531 |
| 18040301 | 18040531 |
| 18140301 | 18140531 |
| 18240301 | 18240531 |
| 18340301 | 18340531 |
| 18440301 | 18440531 |
| 18540301 | 18540531 |
| 18640301 | 18640531 |
| 18740301 | 18740531 |
| 18840301 | 18840531 |
| 18940301 | 18940531 |

Graphical user interface, application, table

Description automatically generated

1. Then add **two filters**. The add filter button will be locate in upper left-hand side and input the following:

**Filter #1**

* **Field:** Location
* **Operator:** IS
* **Value:** ITE00100554

**Filter #2**

* **Field:** TMAX/TMIN
* **Operator:** IS
* **Value:** MAX

Graphical user interface, text, application, email

Description automatically generated

1. To show the value in each bar, do the following: **Panel setting** > **Show Values on chart**
2. The following image will show up, If no error occur.

Chart, histogram

Description automatically generated

1. Save the bar chart as **cis3200 Milan Max Temp**
2. Let create another **Vertical Bar** chart to analyze **Min Temperatures**.
3. In **metrics** section, input the following:

* **Aggregation:** Min
* **Field:** Temperature

Graphical user interface, text, application

Description automatically generated

1. In **Buckets** section, Click “**add**” and select “**X-axis**”. Then, input the following:

* **Aggregation:** Range
* **Field:** Date

|  |  |
| --- | --- |
| 17640301 | 17640531 |
| 17740301 | 17740531 |
| 17840301 | 17840531 |
| 17940301 | 17940531 |
| 18040301 | 18040531 |
| 18140301 | 18140531 |
| 18240301 | 18240531 |
| 18340301 | 18340531 |
| 18440301 | 18440531 |
| 18540301 | 18540531 |
| 18640301 | 18640531 |
| 18740301 | 18740531 |
| 18840301 | 18840531 |
| 18940301 | 18940531 |

Graphical user interface, application, table

Description automatically generated

1. Then add **two filters**. The “**add filter**” button will be located in upper left-hand side. Input the following:

**Filter #1**

* **Field:** Location
* **Operator:** IS
* **Value:** ITE00100554

**Filter #2**

* **Field:** TMAX/TMIN
* **Operator:** IS
* **Value:** Min

Graphical user interface, text, application

Description automatically generated

1. To show the **value** in each bar, do the following: **Panel setting** > **Show Values on chart**
2. The following image should show, If no error occur.

Chart, waterfall chart

Description automatically generated

1. Save the bar chart as **cis3200 Milan Min Temp**
2. Next, we are going to create a **Horizontal Bar** chart to visualize the **average Temperatures of Milan**.
3. On the left side navigation, click on **Visualize**. Then click on “**Create Visualization**.”

Graphical user interface, text, application, Teams

Description automatically generated

1. Select “**Horizontal Bar**” chart

Graphical user interface, application

Description automatically generated

1. In the New Line /Choose a source prompt, select milan\_weathe\*

Graphical user interface, text, application, email

Description automatically generated

1. In **metrics** section, input the following:

* **Aggregation:** Average
* **Field:** Temperature

Graphical user interface, text, application

Description automatically generated

1. In **Buckets** section, Click “add” and select “X-axis”. Then, input the following:

* **Aggregation:** Range
* **Field:** Date

|  |  |
| --- | --- |
| 17640301 | 17640531 |
| 17740301 | 17740531 |
| 17840301 | 17840531 |
| 17940301 | 17940531 |
| 18040301 | 18040531 |
| 18140301 | 18140531 |
| 18240301 | 18240531 |
| 18340301 | 18340531 |
| 18440301 | 18440531 |
| 18540301 | 18540531 |
| 18640301 | 18640531 |
| 18740301 | 18740531 |
| 18840301 | 18840531 |
| 18940301 | 18940531 |

Graphical user interface, table

Description automatically generated

1. Then add filter. The add filter button will be locate in upper left-hand side. Input the following:

**Filter #1**

* **Field:** Location
* **Operator:** IS
* **Value:** ITE00100554

Graphical user interface, text, application

Description automatically generated

1. To show the value in each bar, do the following: **Panel setting** > **Show Values on chart**
2. The following image should show, If no error occur.

Chart, bar chart

Description automatically generated

1. Save the bar chart as **cis3200 Milan Avg Temp**
2. Now, we can create a **dashboard**. Where we can visualize all three charts.
3. On the left side navigation, click on **Dashboard**. Then click on “**Create Dashboard**.”

Text

Description automatically generated with medium confidence

1. Select “**add an existing**” and look for the three charts by their names.
2. Your dashboard should look like the following image:

Graphical user interface, application

Description automatically generated with medium confidence

1. Save the dashboard as **cis3200 Milan dashboard**.

Step 4: Create a predict model in Azure ML Studio

Please note because of the amount of time it takes to run Azure ML Studio with the 1884 and 1894 datasets as well as the inaccuracy of the models, those datasets are not included in this lab.

1. Upload the datasets to Microsoft Azure ML Studio by going to the dataset tab and then clicking new at the bottom. Repeat until all datasets are uploaded.
2. Create a new experiment and name it “Weather model”.
3. Drag all datasets to the screen from 1764-1874.

Background pattern

Description automatically generated with medium confidence

1. Search for **Add Rows** and drag it to the screen. Repeat until there are enough modules for every pair of datasets.
2. Connect the 1764 and 1774 datasets to the Add Row module. Repeat for all other datasets by connecting every pair of datasets **Add Row** module.

Diagram

Description automatically generated

1. Drag in another **Add Row** module and connect the **Add Row** modules connected to the datasets with each other. Repeat for every pair of datasets.

Diagram

Description automatically generated

1. Continue to **Add Row** modules and connect **Add Row** modules with each other until there all the datasets have been combined. Graphical user interface, text

   Description automatically generated
2. Search for the **Clip Values Module** and drag it under the last **Add Rows** module. On the right sidebar set the values as follows.
   1. Set of thresholds: ClipPeaksandSubpeaks
   2. Threshold: Constant
   3. Constant value of upper threshold: 50
   4. Constant value of lower threshold: -89
   5. Substitute value for peaks: Missing.
   6. Substitute value for subpeaks: Missing.
   7. List of Columns: Select Temperature
   8. Overwrite flag: checked
   9. Added indicator columns: unchecked.

This removes temperature outliers from the dataset.

1. Search for the **Clean Missing Data** module and drag it to the screen. Configure it as follows:
   1. Columns to be cleaned: Temperature, Location
   2. Minimum missing value ratio: 0
   3. Maximum missing value ratio: 1
   4. Cleaning mode: Remove entire row

This removes the rows that contained the outliers.

1. Search for the **Select Columns in Dataset** module and drag it to the screen. Connect the Clean Missing Data Module to it. Select all columns but exclude Column 4, Column 5, and Column 6.

Diagram

Description automatically generated

1. Search for the **Split Data** module and drag it to the screen under the Select Columns in Dataset. Configure it as follows:
   1. Splitting mode: Split rows
   2. Fraction of rows in the first output dataset: 0.8
   3. Randomized split: Checked
   4. Random seed: 0
   5. Stratified split: False
2. Search for another **Split Data** module and drag it to the screen under the previous Split Data module. Configure it as follows:
   1. Splitting mode: Split rows
   2. Fraction of rows in the first output dataset: 0.75
   3. Randomized split: Checked
   4. Random seed: 0
   5. Stratified split: False
3. Search for the **Bayesian Linear Regression** module and drag it to the left of the Split Data module. Configure it as follows:
   1. Regularization weight: 1
   2. Allow unknown categorical levels: Checked
4. Search for the **Train Model** module and drag it to the left of the Split Data module. Connect the Bayesian Linear Regression to the left port and the right port of the bottommost Split Row module to the right port. Select the temperature column.

Graphical user interface, text, chat or text message

Description automatically generated

1. Search for the **Score Model** module and drag it to below the Train Model module. Connect the Train Model module to the left port and the right port the right port of the bottommost Split Row module to the right port. Ensure that the Append score columns to output is checked.

Note the output after the program is run:

Graphical user interface, table

Description automatically generated

1. Search for the **Permutation Feature Importance** module and drag it below the Score module. Connect the Train model module to the left port and the right port of the bottommost Split Row module to the right port. Configure it as follows:
   1. Random seed: 1234
   2. Classification: Mean Absolute Error

Diagram

Description automatically generated

1. Search for the **Decision Forest Regression** module and drag it to the right of the Split Data module. Configure it as follows:
   1. Resampling method: Bagging
   2. Create trainer mode: Single Parameter.
   3. Number of decision trees: 8
   4. Maximum depth of the decision trees: 32
   5. Number of random splits per node: 128
   6. Minimum number of samples per leaf node: 1
   7. Allow unknown values for categorical features: Checked
2. Search for the **Train Model** module and drag it to the left of the Split Data module. Connect the Bayesian Linear Regression to the left port and the right port of the bottommost Split Row module to the right port. Select the temperature column.
3. Search for the **Score Model** module and drag it to below the new Train Model module. Connect the Train Model module to the left port and the right port the right port of the bottommost Split Row module to the right port. Ensure that the Append score columns to output is checked.

Note the output after it is run:

Graphical user interface, table

Description automatically generated

It should look as follows:

Graphical user interface, text

Description automatically generated

1. Search for the **Evaluate Model** module and drag it below and between the Score Model modules. Connect both score model modules on the left and the right to the left and right ports respectively.

Diagram

Description automatically generated

1. When the Evaluate Model module is run, it compares the two models with each other. The model attached to the left port is presented first (Bayesian), followed by the metrics for the model attached on the right port (Forest)​

Graphical user interface

Description automatically generated with medium confidence

1. The lab is complete.

References

* 1. URL of Data Source, https://www.kaggle.com/noaa/noaa-global-historical-climatology-network-daily
  2. URL of your Github: https://github.com/fcortes19/CIS-3200-Project