Assignment 2

Jesse Parvess

https://github.com/15091644/MIT-805/tree/main/assignment_2

Part 1: Map Reduce Discussion

Data Set

Weather measurements, across the city of Sofia, where collected. The data set consists of the following field names, data types and explanation:

Table 1: Sofia weather data set, field information.

Name	Data Type	Explanation
Unnamed: 0	Integer	Ordered event identifier
senor_id	Integer	Identifier of the weather sensor
location	Integer	Identifier of the sensor's location
lat	Float	Latitude of the sensor
lon	Float	Longitude of the sensor
timestamp	String	Datetime of recording
pressure	Float	Pressure recording from sensor (Pa)
temperature	Float	Temperature recording of sensor $(°C)$
humidity	Float	Humidity recording of sensor $(\%)$

It is assumed that these sensors are used to generate statistics that inform weather information (daily summaries). Therefore, there is a business requirement to create summary statistics of a day's worth of weather information per sensor that can be fed to a dashboard. This dashboard can then show per sensor information about the day's weather of the location, where the sensor occurs.

Approach

A MapReduce algorithm was used to process batches of weather data, that occurs per day. It is assumed that filtering of the collected data has occurred (based on the timestamp) and that this days' worth of batch of data is fed to the MapReduce algorithms. The algorithms will then process the day's batch and produce statistics that can inform a per sensor account of the day's weather.

This approach is demonstrated on the temperature measurement. This can easily be changed and expanded to the pressure and humidity measurements as well.

The following must be considered:

1. The maximum temperature measured per sensor
2. The minimum temperature measured by the sensor
3. The mean temperature measured by the sensor
4. The mode temperature measured by the sensor (to ascertain if the mode and mean are similar, thus a measure of skewness of the distribution of measurements)
5. The standard deviation of measurements (to ascertain if the maximum and minimum values found are outliers and for a measure of variability of the measurement at each location)
6. The skewness of the distribution to understand the measurements' shape
7. In each case the outputs must be ordered so that quick interpretation of results can be ascertained

Each consideration alters a standard MapReduce algorithm:

1. Each sensor (key) is mapped to its measured temperatures (values)
2. For each key the values are summarised (maximum, minimum, mean, mode, standard deviation, and skewness)
3. For all the keys, the outputs are ordered in ascending order

Reasons for this Algorithm:

1. The algorithm does not require dependencies and can be written with standard built-in functions
2. The summary statistics can be used to get daily information, ascertain the validity of that information (via the standard deviation and outliers); ascertain the shape of the distribution of the information and by using ordering – understand the general weather conditions of sensors located in a region
3. The algorithms can easily be transferred to other measurements when considered necessary (pressure and humidity)

Results Generation:

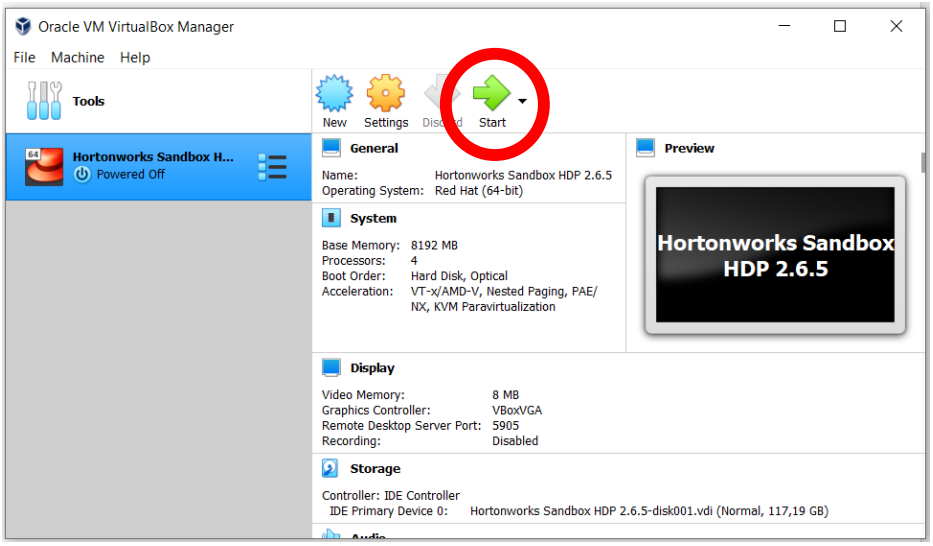
To illustrate the above approach a random day was chosen. In this case the data was filtered for 2017-07-13 as a batch data set used for the MapReduce algorithms to create daily statistics.

A virtual Linux machine was set up using Cloudera's HDP 2.6.5 sandbox. The Hadoop cluster consists of a single node with 8 GB ram.

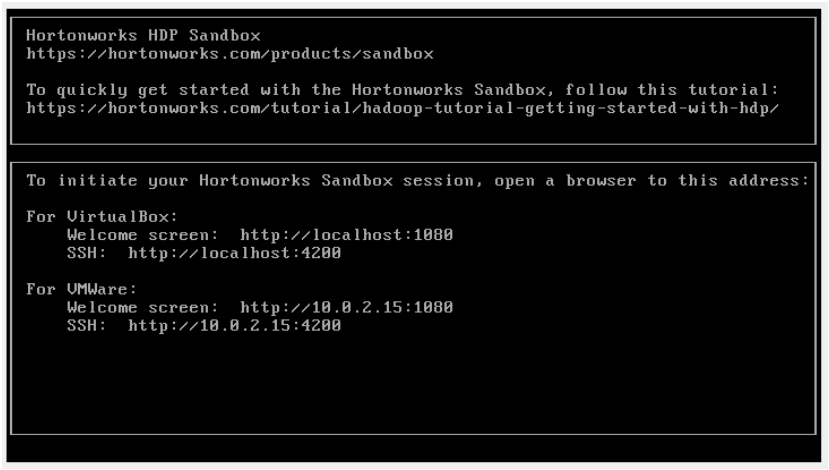
The data set is loaded to the Hadoop File System (HDFS) only with the scripts containing the MapReduce algorithms.

To run the MapReduce algorithm on the machine, one must SSH into the virtual box using puTTY. The user than navigates to the HDFS folder containing the data set and MapReduce code. This is illustrated as follows:

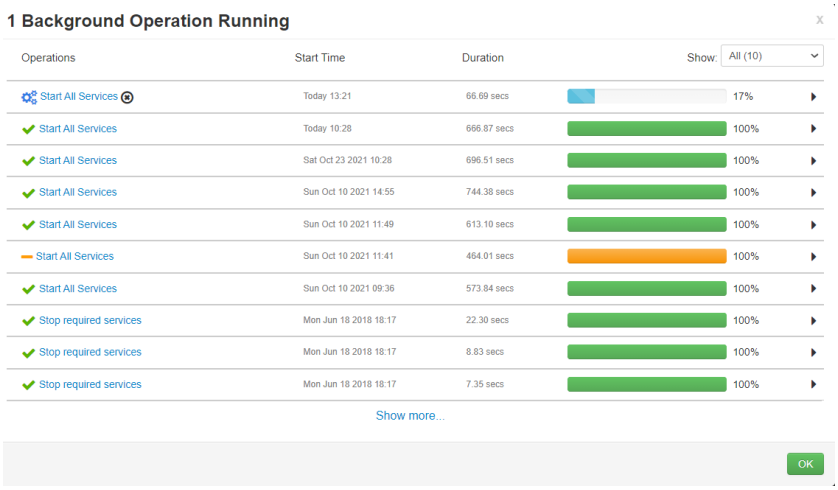
1. Launch the virtual Linux Machine:



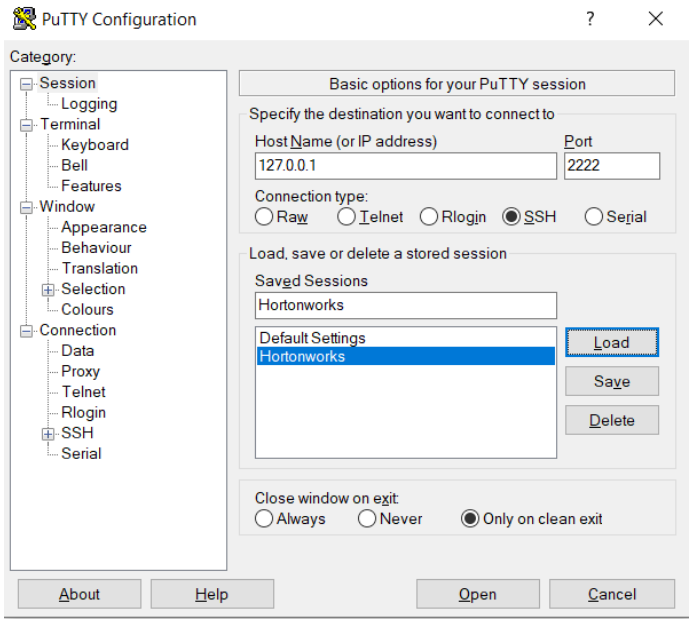
2. Check that all systems are loaded on Ambari by going to the local host specified:



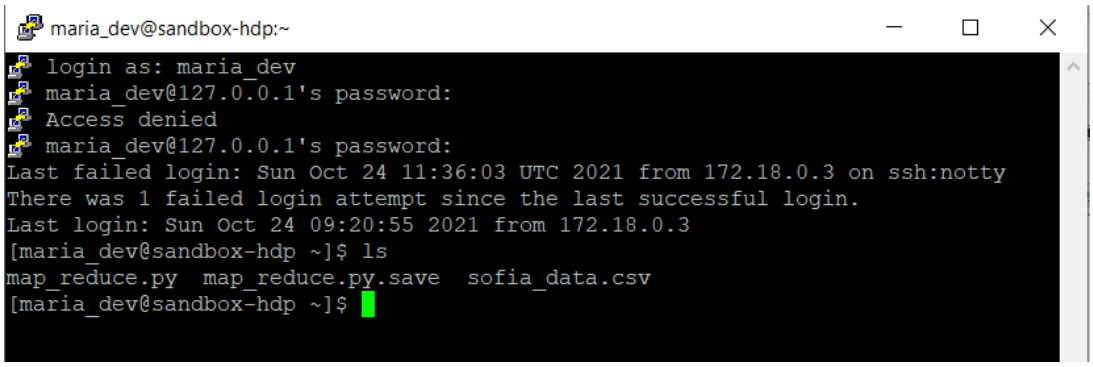
3. Check Ambari and wait for the Hadoop ecosystem to start completely:



4. SSH into the virtual box using puTTY:



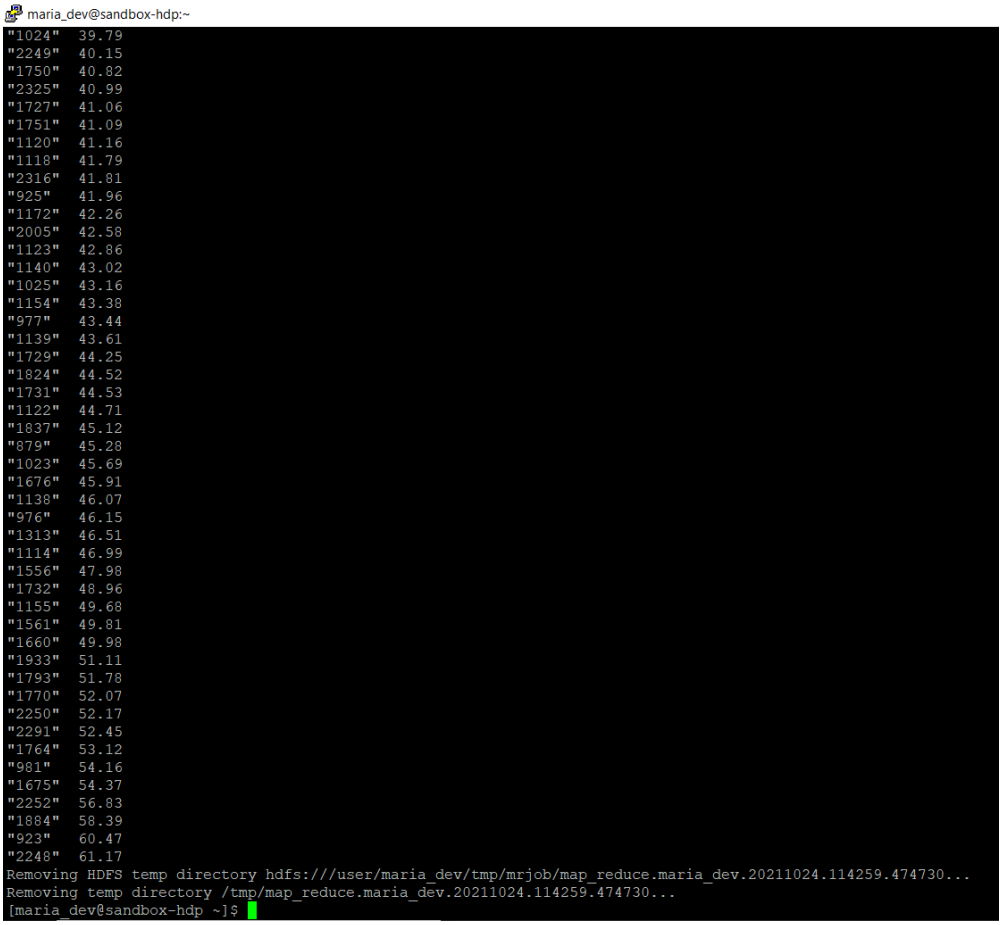
5. Ensure the data and MapReduce code are in the same HDFS directory:



6. Use the following command to run the script:

```
python MapReduce.py -r hadoop --hadoop-streaming-jar
/usr/hdp/current/hadoop-mapreduce-client/hadoop-streaming.jar
sofia_data.csv
```

7. Here the algorithm used was to find the maximum temperature on a sensor per day. All other MapReduce algorithms are specified in GitHub. The output (keys are sensor IDs and values are maximum temperatures of the sensor on the day) is as follows:


maria_dev@sandbox-hdp:~
"1024" 39.79
"2249" 40.15
"1750" 40.82
"2325" 40.99
"1727" 41.06
"1751" 41.09
"1120" 41.16
"1118" 41.79
"2316" 41.81
"925" 41.96
"1172" 42.26
"2005" 42.58
"1123" 42.86
"1140" 43.02
"1025" 43.16
"1154" 43.38
"977" 43.44
"1139" 43.61
"1729" 44.25
"1824" 44.52
"1731" 44.53
"1122" 44.71
"1837" 45.12
"879" 45.28
"1023" 45.69
"1676" 45.91
"1138" 46.07
"976" 46.15
"1313" 46.51
"1114" 46.99
"1556" 47.98
"1732" 48.96
"1155" 49.68
"1561" 49.81
"1660" 49.98
"1933" 51.11
"1793" 51.78
"1770" 52.07
"2250" 52.17
"2291" 52.45
"1764" 53.12
"981" 54.16
"1675" 54.37
"2252" 56.83
"1884" 58.39
"923" 60.47
"2248" 61.17
Removing HDFS temp directory hdfs:///user/maria_dev/tmp/mrjob/map_reduce.maria_dev.20211024.114259.474730...
Removing temp directory /tmp/map_reduce.maria_dev.20211024.114259.474730...
[maria_dev@sandbox-hdp ~]$

Results

Table 1: Generated output on temperature per sensor using the map reduce algorithms on a random day (2017-05-18)

Sensor (Keys)	Maximum Temperature	Minimum Temperature	Mean Temperature	Mode Temperature	Percentage Difference	Maximum Outlier	Minimum Outlier	Skewness
1731	30.95	21.55	25.75	25.77	0.08%	31.49	20.01	-0.09
1121	31.55	22.98	27.75	28.01	0.94%	33.41	22.09	-0.05
1120	31.56	21.37	27.75	27.91	0.58%	34.45	21.05	-0.31
1172	31.85	20.17	26.75	26.31	1.64%	34.17	19.33	-0.08
925	32.21	21.15	26.75	27.42	2.50%	34.29	19.21	-0.13
1154	32.92	19.7	26.75	27.11	1.35%	35.09	18.41	-0.2
1024	33.28	23.51	28.75	28.3	1.57%	34.67	22.83	-0.1
2005	33.31	20.94	27.75	27.8	0.18%	34.83	20.67	-0.09
1118	34.6	21.01	28.75	30.01	4.38%	37.33	20.17	-0.24
1558	35.18	23.83	29.75	29.55	0.67%	37.31	22.19	-0.02
1729	35.36	21.14	28.75	28.17	2.02%	37.93	19.57	-0.05
1751	35.83	22.05	28.75	28.77	0.07%	36.03	21.47	-0.03
1998	36.02	23.17	28.75	28.19	1.95%	37.09	20.41	-0.21
1727	36.06	22.76	30.75	31.77	3.32%	37.79	23.71	-0.55
1023	36.42	18.7	26.75	26.3	1.68%	36.33	17.17	-0.09
1750	36.61	21.38	28.75	28.07	2.37%	37.61	19.89	-0.26
1313	37.02	20.02	27.75	28.53	2.81%	36.93	18.57	-0.09
1556	37.02	21.75	27.75	26.85	3.24%	36.03	19.47	-0.22
1138	37.31	20.64	27.75	26.91	3.03%	36.77	18.73	-0.13
1732	37.65	20.03	27.75	27.37	1.37%	39.51	15.99	-0.25
1123	37.66	22.62	29.75	28.81	3.16%	39.65	19.85	-0.27
977	37.81	21.52	29.75	29.37	1.28%	40.37	19.13	-0.05
1140	38	19.18	28.75	29.21	1.60%	39.99	17.51	-0.1
1676	38.04	21.29	24.75	22.96	7.23%	34.67	14.83	-1.88
1139	38.55	22.24	30.75	30.31	1.43%	40.85	20.65	-0.04
976	39.11	19.59	27.75	29.17	5.12%	38.05	17.45	-0.16
1122	39.51	23.21	29.75	29.91	0.54%	39.85	19.65	-0.16
1933	40.23	21.09	28.75	27.39	4.73%	41.17	16.33	-0.45
1824	40.28	21.34	29.75	29.88	0.44%	39.87	19.63	-0.17
1155	40.47	18.52	26.75	24.81	7.25%	38.29	15.21	-0.46
879	40.87	19.57	27.75	26.97	2.81%	39.69	15.81	-0.26
981	41.79	20.73	28.75	27.58	4.07%	40.61	16.89	-0.57
1561	43.08	20.99	30.75	29.95	2.60%	43.91	17.59	-0.2
1764	44.9	21.39	29.75	27.27	8.34%	42.53	16.97	-0.65
1884	45.27	23.02	28.75	28.08	2.33%	38.51	18.99	-1.16
1660	45.59	21.17	28.75	26.46	7.97%	42.33	15.17	-0.76
1770	46.91	20.78	30.75	30.27	1.56%	44.83	16.67	-0.54
923	47.55	19.54	27.75	25.92	6.59%	40.77	14.73	-0.77
1675	52.27	19.45	32.75	28.66	12.49%	48.83	16.67	-1.17
Daily Mean	38.22	21.16	28.52	28.0026				

Table 1 indicates the following:

1. The sensors that read the highest temperatures generally do not also read the lowest temperatures.
2. The mean temperature and mode temperatures generally correspond. However, the percentage difference between these two indicates a correlation with the maximum temperature found.
3. Furthermore, when using $\mu \pm 2 * \sigma$ to detect maximum and minimum outliers, as the temperature increases more of the maximum temperatures found can be considered outliers ($\mu + 2 * \sigma$).
4. Maximum outliers on some sensors indicates that some sensors may have registered unnatural temperatures or are faulty.
5. Most minimum temperatures are within the $\mu - 2 * \sigma$ threshold.
6. Maximum temperatures are more outlier prone compared to minimum temperatures.
7. High percentage differences in mean and mode corresponds to high skewness towards higher temperatures (as expected).

Table 1 indicates that a user can quickly ascertain whether readings are unusual, what the distribution of the readings are, and what the general maximum, minimum, average, and modal temperatures for a day are.

Improvements:

1. This data should incorporate the proximity of sensors so that spatial relationships between readings can be ascertained.

Part 2: Visualisation Discussion

Introduction

Weather data for the town of Sofia was collected. The data set consists of the following:

Table 1: Sofia weather data set, field information.

Name	Data Type	Explanation
Unnamed: 0	Integer	Ordered event identifier
senor_id	Integer	Identifier of the weather sensor
location	Integer	Identifier of the sensor's location
lat	Float	Latitude of the sensor
lon	Float	Longitude of the sensor
timestamp	String	Datetime of recording
pressure	Float	Pressure recording from sensor (Pa)
temperature	Float	Temperature recording of sensor $(°C)$
humidity	Float	Humidity recording of sensor $(\%)$

Here, for practical reasons, the temperature measurement is used to extract useful visualisations. The approaches discussed can easily be expanded to the humidity and pressure measurements.

It is assumed that these sensors were placed around the city by scientists who wish to monitor the local weather differences across the city. To ensure effective monitoring occurs, measurements are checked once per day to gain insights and detect issues.

These insights are the maximum, minimum, and mean registered temperature per sensor per day. This will give a detailed understanding of the daily temperature registered by the sensors. Additionally, because there are multiple sensors spread out spatially, understanding these measurements with respect to the locations of the sensors is important. Furthermore, it is important to check the validity of these measurements. Thus, visualisations that check the measurements of the closest sensors with respect to a reference sensor of interest is important. This is because if sensor is reading vastly different readings to its closest neighbours, it indicates a potential issue on that sensor (especially if its neighbours read similar readings). Furthermore, since data is registered every few seconds, understanding the progression of a measurement over time can allow for temporal patterns to be extracted for that day. This can be overlayed with the sensor's closest neighbours to gain insight about the temporal trend of a measurement across a local region.

Methodology

To simulate the above scenario a random day was chosen from the data set to develop various daily visualisations. Here the day of 2018-05-01 was considered.

Spatial Visualisations

The mean, maximum, and minimum temperatures on the day for each sensor was determined and overlayed on the positional data given (latitude and longitude):

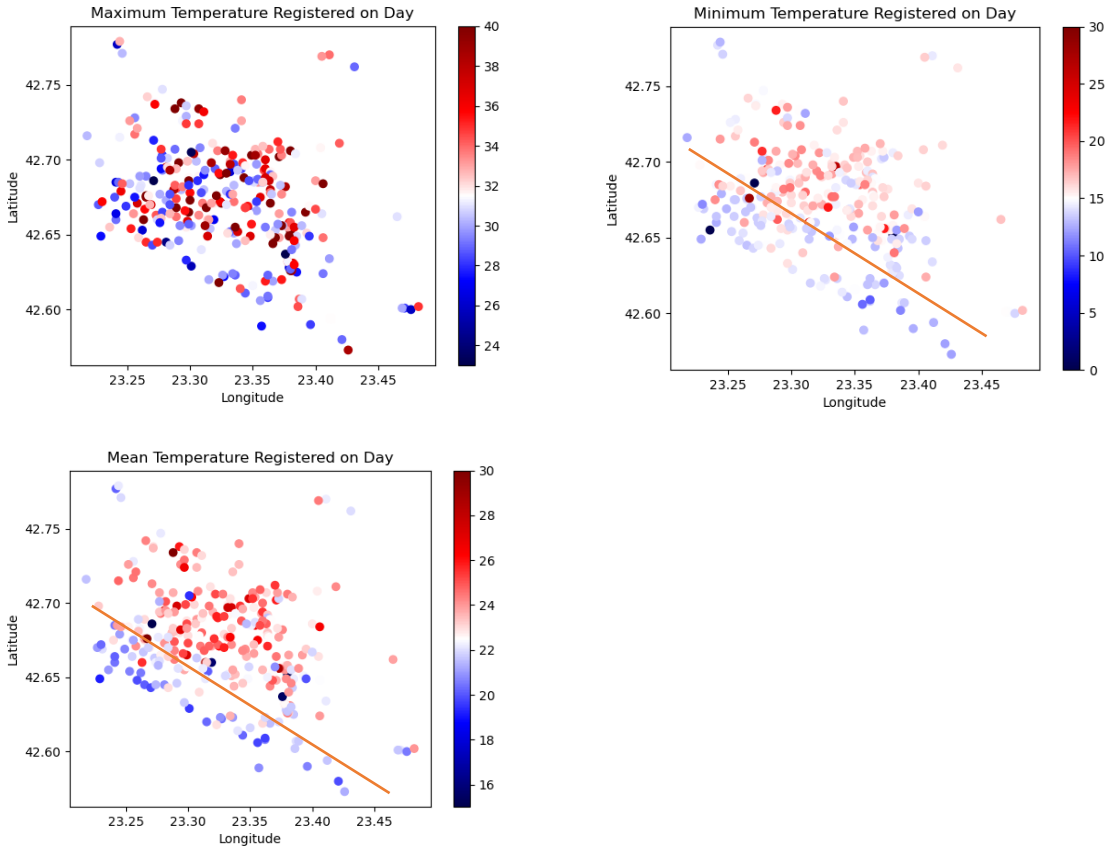


Figure 1: Maximum, Minimum and Mean Temperature Readings per Sensor on Day

Figure 1 indicates a clear difference between sensors located in the north of the city compared to the south of the city. Clearly the north-east of Sofia is on average warmer than its south-west region.

Looking at the maximum temperature scatter plot, it indicates that there are more uniform maximum readings across the city. Maximums on average were around $35°C$. Some maximum readings exceed $38°C$ and should be investigated as outliers.

Looking at the minimum temperature scatter plot, it indicates that there are less uniform minimum readings across the city (a clear north-westerly divide, shown with an orange line). Minimums on average were around $15°C$. Some minimum readings were as low as $5°C$ and should be investigated as outliers.

Looking at the average temperature scatter plot, it indicates that there are less uniform mean readings across the city (a clear north-westerly divide, shown with an orange line). Means on average were around $25°C$for the day. The clear spatial divide is confirmed by the mean temperature because it summarises a range of temperature values (and is therefore more temporally robust). This stands in contrast to the minimum temperature reading which may occur at a certain time of day when this pattern is prominent (say early morning)

An explanation for the higher northern temperatures could be due to the topography of the region:

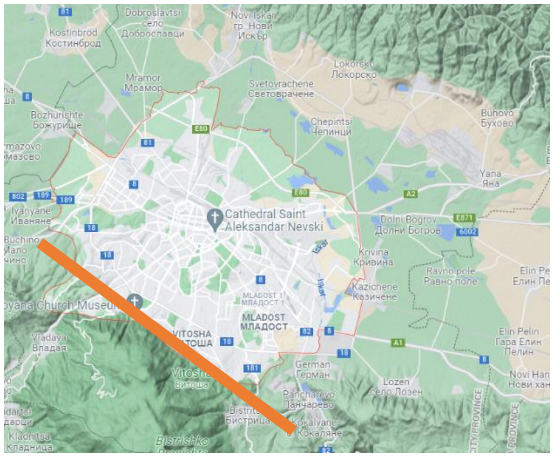


Figure 2: Topographical map of Sofia, Bulgaria

Figure 2 indicates an orange line showing the prominence of a mountain that runs in a north-westerly direction. Compare this to the lines in figure 1. Clearly the prominence of the mountain influences the temperature measurements. Here the mountain could influence wind flows and heating patterns that cause lower temperature measurements in sensors along this direction in the city.

Outlier Detection

To detect outliers among sensors a heatmap can be used to compare temperature readings across a single day and across time (across days) to detect sudden sensor degradation.

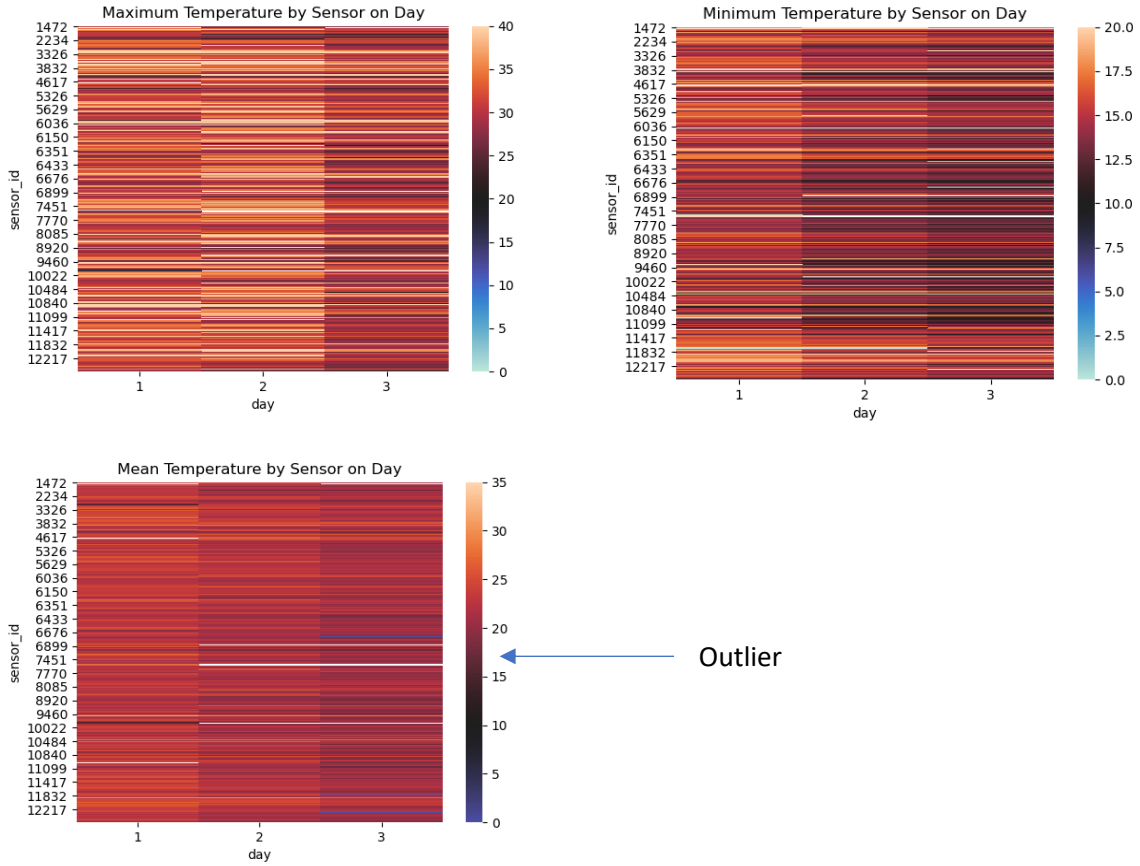


Figure 3: Temperature measurement heatmaps for 3 days

Figure 3 provides easy interpretation of the state of sensors in the system. For instance, what most readings are on a day. One can detect, in general, that from the 1st to the 3rd of May 2018 the temperature decreased (prominence of shaded blue readings). Additionally, by looking at the mean temperature heatmap, sensor 7451 suddenly started reading high values when clearly most temperature readings decreased. This potentially indicates an outlier (indicated by an arrow) and a faulty sensor.

To incorporate spatial information, the distribution of temperature readings of the closest sensors to a sensor of interest can be plotted. Vastly different temperature measurements of the reference sensor to its closest neighbours potentially indicates a sensor issue. Here three sensors (IDs: 1472, 1954, 1962) were chosen to demonstrate the algorithm:

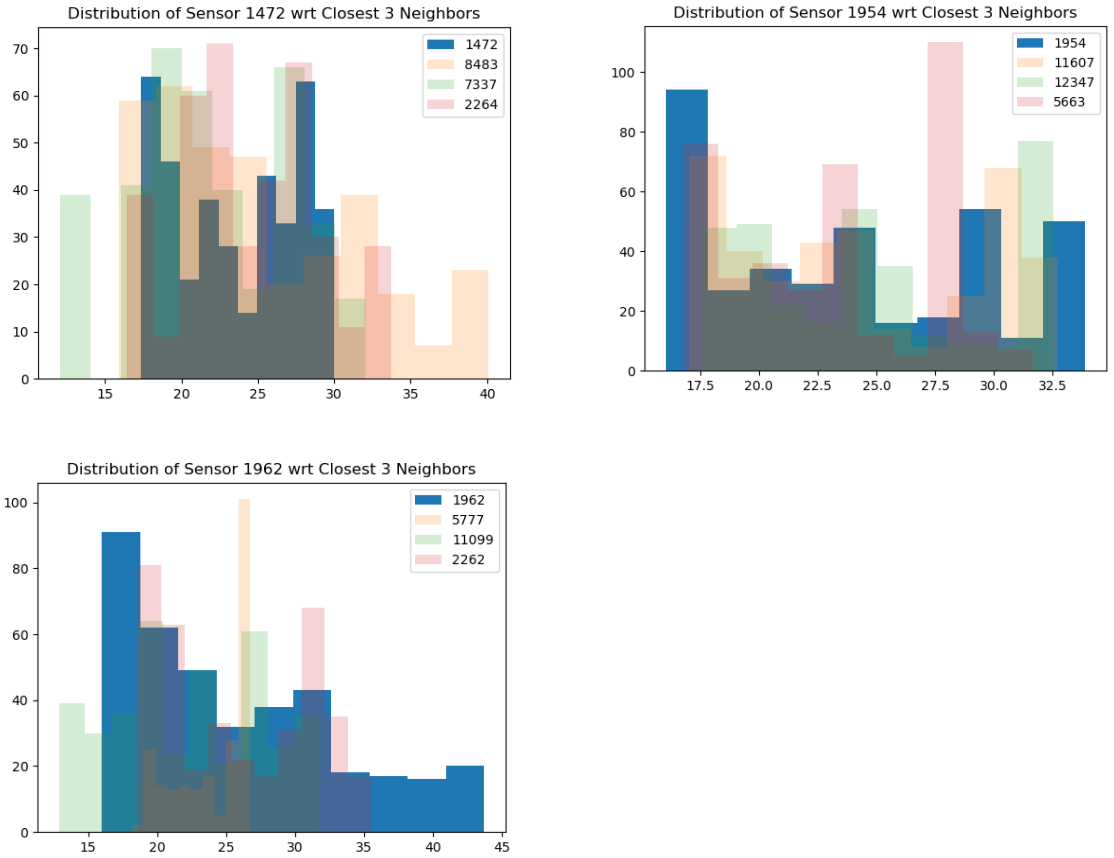


Figure 4: Distribution of a reference sensor's (blue) temperature measurements for a single day compared to its three closest neighbours (orange, pink and green)

Figure 4 indicates that all reference sensors (in blue) have similar distributions to the three closest neighbours to that sensor. There are some differences in measurement (at the extremes of measurement). This approach can be used to quickly ascertain if a sensor's measurements can be trusted by looking at the distribution of its neighbours.

If we compare the same sensors to their **furthest** neighbours, we find:

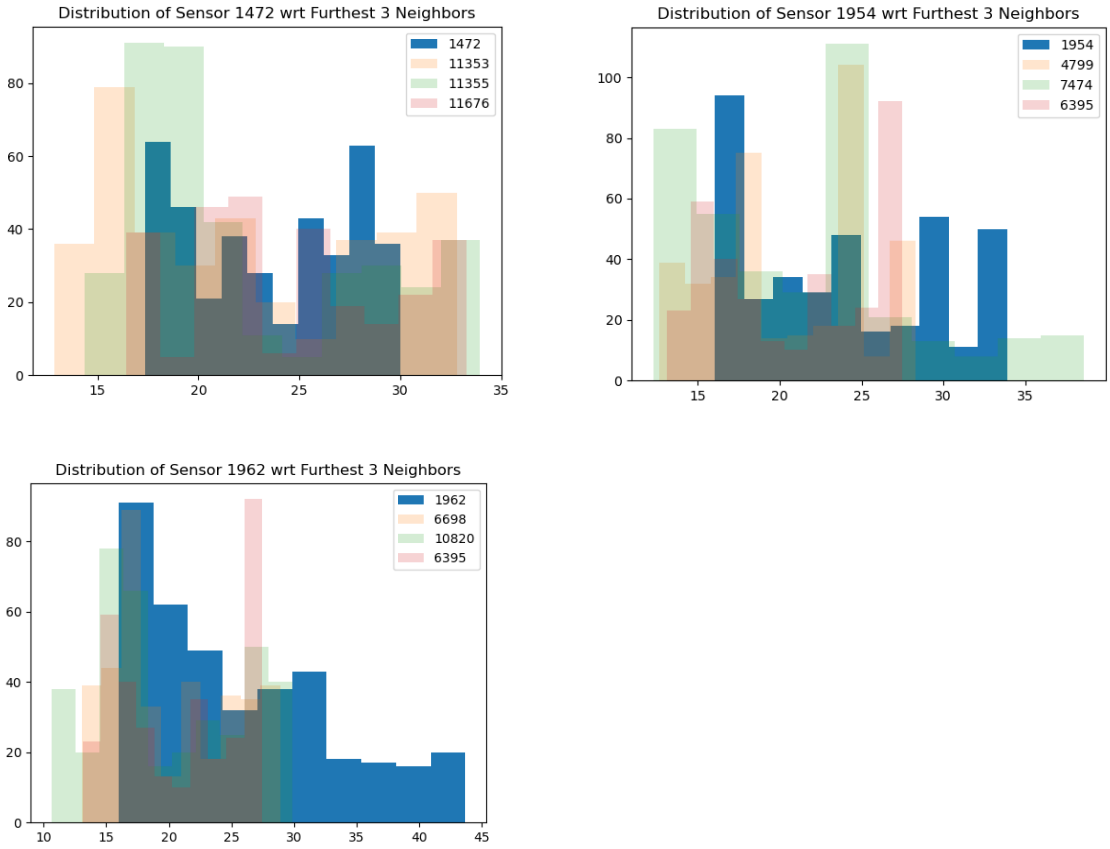


Figure 5: Distribution of a reference sensor's (blue) temperature measurements for a single day compared to its three furthest neighbours (orange, pink and green)

Figure 5 indicates that, when considering the reference sensor to its furthest sensor, very different distributions can be detected (see subfigure 3). This proves that the above method can be effectively used to tack outliers and sensor degradation when considering the reference sensor's closest neighbours (figure 4). This is because the furthest sensors should have different temperature distribution profiles.

Temporal Considerations

Since the data is updated every 20 minutes, and because weather changes with time, tracking the temporal movement of measurements is important to monitoring. Here, the above outlier approach can be employed where the closest sensors to a reference sensor are overlayed temporally. This allows for faulty measurements to be detected (if the reference sensor does follow similar measurements to its closest neighbours) and for the general temperature trend in a locality to be validated by several close sensors.

Here the reference sensor is plotted over time in blue, and its closest neighbours are overlayed over the same period of time:

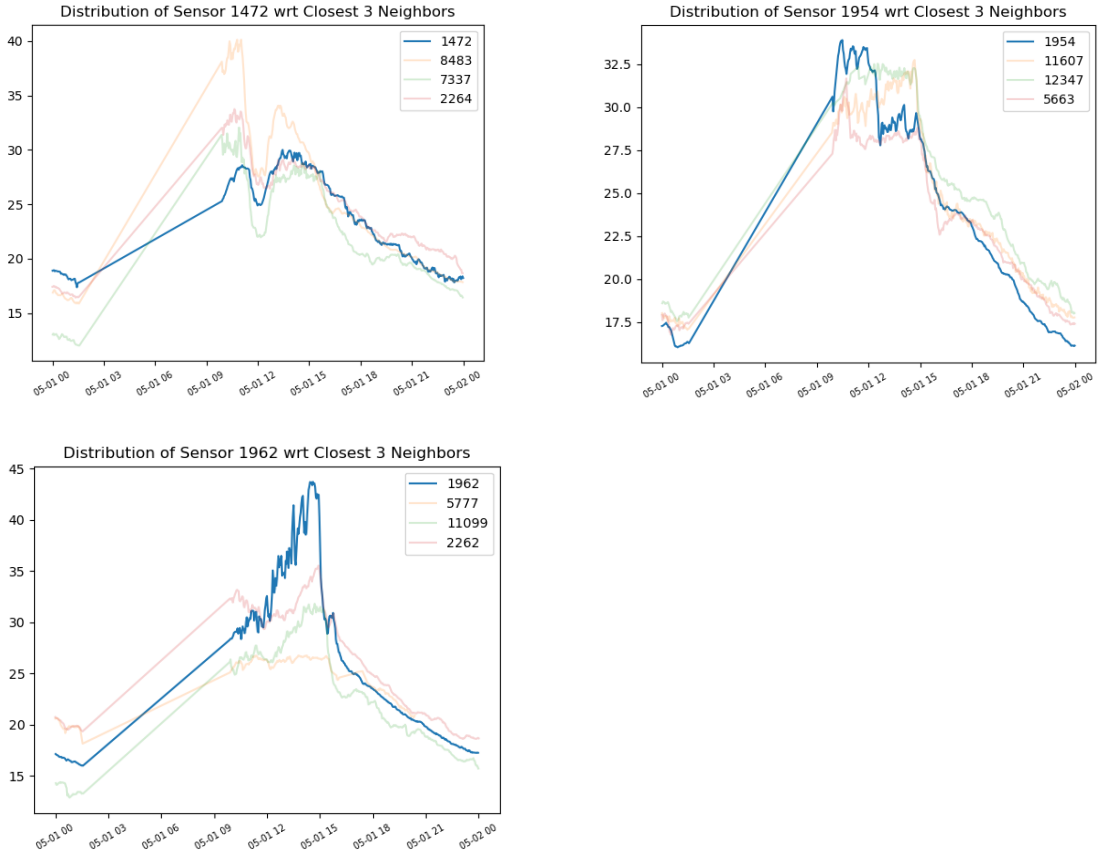
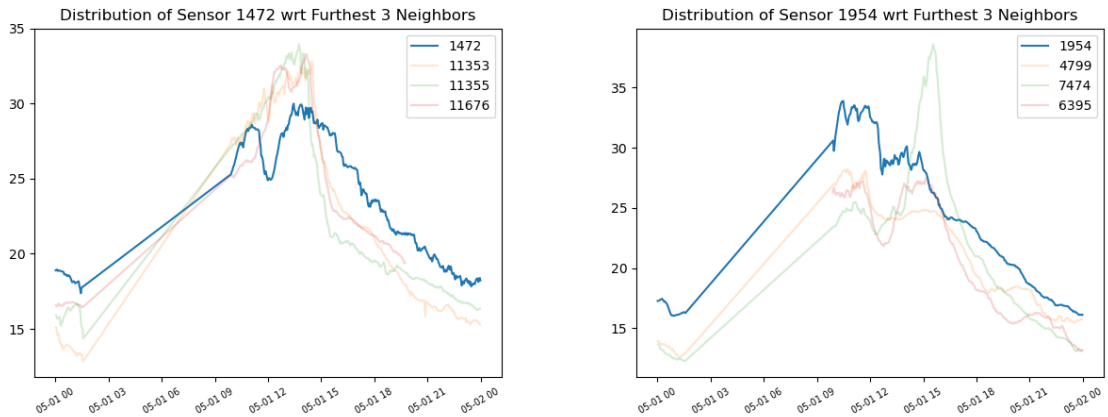


Figure 6: Distribution of a reference sensor's (blue) temperature measurements for a single day compared to its three closest neighbours (orange, pink and green) over time.

Figure 6 indicates that all the closest sensors to a reference sensor follow a similar trend (if not in magnitude then in general profile). All three subfigures represent the same period of time, and thus show the variability of measurement profiles across localities.

Additionally, a period (indicated by a straight gradient from one measurement to the next) indicates that these sensors did not register information and were off. This indicates another form of error detection in the system.

If we compare the same sensors to their **furthest** neighbours, we find:



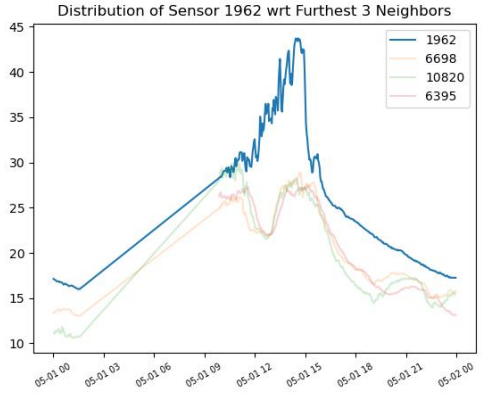


Figure 7: Distribution of a reference sensor's (blue) temperature measurements for a single day compared to its three furthest neighbours (orange, pink and green) over time.

Figure 7 indicates that all the furthest sensors to a reference sensor follow a different trend (in magnitude and in general profile). This indicates that above method is an effective means to detect sensor degradation or faulty readings.

Conclusion

The above approach only considered a single parameter (temperature), this can be expanded to the humidity and pressure measurements (other **varieties** of data in the set). The approach was only focused on daily tracking, such that a user could find:

1. Spatial differences in measurements (the **volume** of data is summarised to daily statistics)
2. Outliers in sensors (the **veracity** of the data is measured in the visualisations)
3. Change in time in sensor measurements (the **velocity** and change in time is accounted for)

The approach was heavily focused on outlier detection and sensor degradation since the purpose of these sensors is for scientific study. Thus, the researchers could know when to service the system and the veracity of the data they collected can be upheld.

To improve the approach, one can summarise the data over larger periods of time (like for a month in a year) and gather summary statistics.