RBDA Project - Data Profiling

Yiyi Tong, yt2239

1. Data Sources

In this project, the air data is downloaded from the EPA for further analysis with COVID data and sentiment data extracted from tweets. The daily air quality index by county (AQI), daily temperature (TEMP), and daily barometric pressure (PRESS) datasets are selected from 2019-2022 and the total size of them is 488MB. Below are the examples of the three datasets:

daily_aqi_by_county_2022

State Name	county Name	State Code	County Code	Date	AQI	Category	Defining Parameter	Defining Site	Number of Sites Reporting
New Mexico	Santa Fe	35	49	2022-01-01	0	Good	PM2.5	35-049-0021	1
Puerto Rico	Caguas	72	25	2022-01-01	0	Good	CO	72-025-0007	1
North Carolir	Northampton	37	131	2022-01-01	1	Good	NO2	37-131-0003	1
Puerto Rico	Bayamon	72	21	2022-01-01	2	Good	CO	72-021-0010	1
Virginia	Pittsylvania	51	143	2022-01-01	2	Good	NO2	51-143-0005	1
Illinois	Macoupin	17	117	2022-01-01	3	Good	NO2	17-117-0002	1
North Carolir	Wake	37	183	2022-01-01	3	Good	NO2	37-183-0021	2
Rhode Islanc	Kent	44	3	2022-01-01	3	Good	PM2.5	44-003-0002	1
Virginia	Norfolk City	51	710	2022-01-01	3	Good	NO2	51-710-0024	1
Texas	Karnes	48	255	2022-01-01	4	Good	NO2	48-255-1070	1
Texas	Wilson	48	493	2022-01-01	4	Good	NO2	48-493-1038	1
Arizona	Mohave	4	15	2022-01-01	5	Good	PM10	04-015-1003	1

daily_temp_2022

State Code	County Code	Site Num	Parameter Code	POC	Latitude	Longitude	Datum	Parameter Name	Sample Duration	Pollutant Standard	Date Local	Units of Measure	Event Type	Observation Count	Observation Percent	Arithmetic Mean
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-01	Degrees Fahrenheit	None	24	100.0	74.925
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-02	Degrees Fahrenheit	None	24	100.0	61.8125
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-03	Degrees Fahrenheit	None	24	100.0	34.425
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-04	Degrees Fahrenheit	None	24	100.0	37.4125
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-05	Degrees Fahrenheit	None	24	100.0	48.45
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-06	Degrees Fahrenheit	None	24	100.0	56.220833
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-07	Degrees Fahrenheit	None	24	100.0	42.058333
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-08	Degrees Fahrenheit	None	24	100.0	50.1625
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-09	Degrees Fahrenheit	None	24	100.0	64.966667
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-10	Degrees Fahrenheit	None	24	100.0	47.8375
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-11	Degrees Fahrenheit	None	24	100.0	41.854167
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-12	Degrees Fahrenheit	None	24	100.0	43.966667
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-13	Degrees Fahrenheit	None	24	100.0	47.275
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-14	Degrees Fahrenheit	None	24	100.0	48.775
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-15	Degrees Fahrenheit	None	24	100.0	52.079167
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-16	Degrees Fahrenheit	None	24	100.0	37.7375
1	53	1000	62101	1	31.0921	-87.5435	NAD 83.00	Outdoor Temperature	1 HOUR		2022-01-17	Degrees Fahrenheit	None	24	100.0	38.866667

1st Max Value	1st Max Hour	AQI	Method Code	Method Name	Local Site Name	Address	State Name	County Name	City Name	CBSA Name	Date of Last Change
78.9	12		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
74.6	8		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
40.9	14		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
50.8	15		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
62.5	14		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
72.8	13		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
51.9	15		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
61.2	15		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
74.9	14		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
60.7	0		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
53.5	14		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
57.4	14		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
64	14		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
62.3	14		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
59.2	12		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
44.2	0		60	Instrumental - Vaisala 435C RH/AT Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20

daily_press_2022

State Code	County Code	Site Num	Parameter Code	POC	Latitude	Longitude	Datum	Parameter Name	Sample Duration	Pollutant Standard	Date Local	Units of Measure	Event Type	Observation Count	Observation Percent	Arithmetic Mean
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-01	Millibars	None	24	100.0	750.829167
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-02	Millibars	None	24	100.0	750.991667
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-03	Millibars	None	24	100.0	757.941667
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-04	Millibars	None	24	100.0	756.995833
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-05	Millibars	None	24	100.0	753.583333
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-06	Millibars	None	24	100.0	752.179167
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-07	Millibars	None	24	100.0	756.691667
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-08	Millibars	None	24	100.0	758.008333
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-09	Millibars	None	24	100.0	756.575
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-10	Millibars	None	24	100.0	759.3
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-11	Millibars	None	24	100.0	760.445833
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-12	Millibars	None	24	100.0	757.504167
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-13	Millibars	None	24	100.0	753.604167
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-14	Millibars	None	24	100.0	752.783333
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-15	Millibars	None	24	100.0	750.166667
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-16	Millibars	None	24	100.0	749.15
1	53	1000	64101	1	31.0921	-87.5435	NAD 83.00	Barometric pressure	1 HOUR		2022-01-17	Millibars	None	24	100.0	753.695833

1st Max Value	1st Max Hour	AQI	Method Code	Method Name	Local Site Name	Address	State Name	County Name	City Name	CBSA Name	Date of Last Change
751.8	9		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
754.8	22		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
759	20		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
758.7	9		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
754.9	0		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
754.8	22		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
757.9	9		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
759.2	9		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
757.6	0		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
761	22		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
761.8	9		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
759.3	0		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
755.6	0		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
753.9	9		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
752.4	7		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
752.5	22		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
755.4	22		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
756.9	9		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
756	9		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20
756.7	20		60	Instrumental - Vaisala 555B Pressure Sensor	PCI MET1	Jack Springs Rd	Alabama	Escambia	Not in a city		2022-04-20

2. Data Profiling

To make it easier to do SQL queries in HIVE, the three datasets are joined together using the MapReduce framework. The foreign key is the local date combined with the state code and the record is kept only if the AQI data exists. As there may be multiple stations in a state, the AQI,

temperature and pressure are averaged by the state separately. The output record contains the fields: local date, state code, state name, AQI, temperature, pressure.

(1) Mapper

Different mappers are implemented for the three datasets. The input key and value are the offset and the line. The output key is the local date with state code and the output value is a data type flag prepending the corresponding data and the count.

```
@Override
public void map(LongWritable key, Text value, Context context)
        throws IOException, InterruptedException {

    String line = value.toString();
    String[] strArr = line.split( regex: ",");
    try {
        Float.parseFloat(strArr[statCOL]);
        // key: time,location(state code)
        String outKey = String.format("%s,%s", strArr[timeCOL], strArr[posCOL]);
        // value: flag(AQI),location name,AQI,count
        String outVal = String.format("AQI,%s,%s,%d",
                strArr[posNameCol], strArr[statCOL],ONE);
        context.write(new Text(outKey), new Text(outVal));
    }catch (NumberFormatException e){}
}
```

(2) Combiner

As we need to average the data by the states, a combiner helps to reduce the amount of data transferred through the network by averaging the data first. The output value type of the combiner is the same as the mappers.

```
@Override
protected void reduce(Text key, Iterable<Text> values, Context context)
        throws IOException, InterruptedException {
    int[] count = new int[3];   // AQI, TEMP, PRESS
    float[] sum = new float[3];
    String stateName = "";
    for(Text val: values){
        String[] arr = val.toString().split( regex: ",");
        if(arr[0].equals("AQI")){
            count[0]++;
            sum[0] += Float.parseFloat(arr[2]);
            if(stateName.length() == 0){
                stateName = arr[1];
            }
        }else if(arr[0].equals("TEMP")){
            count[1]++;
            sum[1] += Float.parseFloat(arr[1]);
        }else if(arr[0].equals("PRESS")){
            count[2]++;
            sum[2] += Float.parseFloat(arr[1]);
        }
    }
```

```
if(count[0] != 0){
    float meanAQI = sum[0] / count[0];
    String outValue = String.format("AQI,%s,%.4f,%d",
            stateName, meanAQI, count[0]);
    context.write(key, new Text(outValue));
}
if(count[1] != 0){
    float meanTEMP = sum[1] / count[1];
    String outValue = String.format("TEMP,%.4f,%d",
            meanTEMP, count[1]);
    context.write(key, new Text(outValue));
}
if(count[2] != 0){
    float meanPRESS = sum[2] / count[2];
    String outValue = String.format("PRESS,%.4f,%d",
            meanPRESS, count[2]);
    context.write(key, new Text(outValue));
}
```

(3) Partitioner

A custom partitioner is implemented to load balance the work on each reducer. The outputs of the mappers (combiner) are partitioned by the year as the data size is roughly the same each year except for the 2022. And the minimum year can be set through the configurations.

```
public class AirAggPartitioner extends
        Partitioner<Text, Text> implements Configurable {

    private static final String MIN_YEAR = "min.year";
    private Configuration conf = null;
    private int minYear = 0;

    @Override
    public int getPartition(Text key, Text value, int numPartitions) {
        String[] arr = key.toString().split( regex: ",");
        String[] time = arr[0].split( regex: "-");
        return Integer.parseInt(time[0].substring(1)) - minYear;
    }

    public Configuration getConf() {
        return conf;
    }

    public void setConf(Configuration conf) {
        this.conf = conf;
        minYear = conf.getInt(MIN_YEAR,  defaultValue: 2000);
    }

    public static void setMinYear(Job job, int minYear) {
        job.getConfiguration().setInt(MIN_YEAR, minYear);
    }

}
```

(4) Reducer

The join operations are performed at the reducer side. Firstly, the running sum and count of the AQI, temperature and pressure are computed. Then, the average data is written out with the input key (time and location).

```
protected void reduce(Text key, Iterable<Text> values, Context context)
        throws IOException, InterruptedException {
    int[] count = new int[3];   // AQI, TEMP, PRESS
    float[] sum = new float[3];
    String stateName = "";
    for(Text val: values){
        String[] arr = val.toString().split( regex: ",");
        if(arr[0].equals("AQI")){
            int cnt = Integer.parseInt(arr[3]);
            count[0] += cnt;
            sum[0] += Float.parseFloat(arr[2]) * cnt;
            if(stateName.length() == 0){
                stateName = arr[1];
            }
        }else if(arr[0].equals("TEMP")){
            int cnt = Integer.parseInt(arr[2]);
            count[1] += cnt;
            sum[1] += Float.parseFloat(arr[1]) * cnt;
        }else if(arr[0].equals("PRESS")){
            int cnt = Integer.parseInt(arr[2]);
            count[2] += cnt;
            sum[2] += Float.parseFloat(arr[1]) * cnt;
        }
    }
```

```
// left join: keep the record only if the AQI exists
if(stateName.length() > 0){
    // average by state
    float meanAQI = sum[0] / count[0];
    float meanTEMP = count[1] == 0 ? -9999 : sum[1] / count[1];
    float meanPRESS = count[2] == 0 ? -9999 : sum[2] / count[2];
    String outValue = String.format(
            "%s,%.4f,%.4f,%.4f",
            stateName, meanAQI, meanTEMP, meanPRESS);
    context.write(key, new Text(outValue));
}
```

(5) Driver

In the driver code, the mapper, combiner, partitioner and reducer classes are configured. The minimum year is set to 2019 and the number of reduce tasks is set to 4.

```java
public class AirAggregate {
    public static void main(String[] args) throws Exception {
        if (args.length != 4) {
            System.err.println("Usage: AirAggregate <AQI input> <TEMP input> <PRESS input> <output>");
            System.exit( status: -1);
        }

        Configuration conf = new Configuration();
        conf.set("mapred.textoutputformat.separator", ",");

        Job job = Job.getInstance(conf);
        job.setJarByClass(AirAggregate.class);
        job.setJobName("AirAggregate");

        MultipleInputs.addInputPath(job, new Path(args[0]),
                TextInputFormat.class, AirAggAQIMapper.class);
        MultipleInputs.addInputPath(job, new Path(args[1]),
                TextInputFormat.class, AirAggTEMPMapper.class);
        MultipleInputs.addInputPath(job, new Path(args[2]),
                TextInputFormat.class, AirAggPRESSMapper.class);
        FileOutputFormat.setOutputPath(job, new Path(args[3]));
```

```java
job.setCombinerClass(AirAggCombiner.class);
job.setPartitionerClass(AirAggPartitioner.class);
AirAggPartitioner.setMinYear(job, minYear: 2019);
job.setReducerClass(AirAggReducer.class);

job.setMapOutputKeyClass(Text.class);
job.setMapOutputValueClass(Text.class);
job.setOutputKeyClass(Text.class);
job.setOutputValueClass(Text.class);

job.setNumReduceTasks(4);

System.exit(job.waitForCompletion( verbose: true) ? 0 : 1);
```

3. Running the Program

The shell commands to run the program can be seen in the workflow.txt. And below is the log of running the MR job.

```
yt2239_nyu_edu@nyu-dataproc-m:~/RBDA/project/src$ hadoop jar AirAggregate.jar AirAggregate project/data/AQI project/data/TEMP proj
ect/data/PRESS project/aggregate
2022-11-21 17:42:26,048 INFO client.RMProxy: Connecting to ResourceManager at nyu-dataproc-m/192.168.1.39:8032
2022-11-21 17:42:26,219 INFO client.AHSProxy: Connecting to Application History server at nyu-dataproc-m/192.168.1.39:10200
2022-11-21 17:42:26,387 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool i
nterface and execute your application with ToolRunner to remedy this.
2022-11-21 17:42:26,406 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/yt2239_nyu
_edu/.staging/job_1668611926937_0321
2022-11-21 17:42:26,686 INFO input.FileInputFormat: Total input files to process : 23
2022-11-21 17:42:26,718 INFO input.FileInputFormat: Total input files to process : 23
2022-11-21 17:42:26,731 INFO input.FileInputFormat: Total input files to process : 23
2022-11-21 17:42:26,851 INFO mapreduce.JobSubmitter: number of splits:69
2022-11-21 17:42:26,932 INFO Configuration.deprecation: mapred.textoutputformat.separator is deprecated. Instead, use mapreduce.ou
tput.textoutputformat.separator
2022-11-21 17:42:27,028 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1668611926937_0321
2022-11-21 17:42:27,029 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-11-21 17:42:27,199 INFO conf.Configuration: resource-types.xml not found
2022-11-21 17:42:27,199 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-11-21 17:42:27,376 INFO impl.YarnClientImpl: Submitted application application_1668611926937_0321
2022-11-21 17:42:27,410 INFO mapreduce.Job: The url to track the job: http://nyu-dataproc-m:8088/proxy/application_1668611926937_0
321/
2022-11-21 17:42:27,411 INFO mapreduce.Job: Running job: job_1668611926937_0321
2022-11-21 17:42:34,522 INFO mapreduce.Job: Job job_1668611926937_0321 running in uber mode : false
2022-11-21 17:42:34,523 INFO mapreduce.Job:  map 0% reduce 0%
2022-11-21 17:42:43,604 INFO mapreduce.Job:  map 4% reduce 0%
2022-11-21 17:42:45,614 INFO mapreduce.Job:  map 10% reduce 0%
2022-11-21 17:42:49,636 INFO mapreduce.Job:  map 13% reduce 0%
2022-11-21 17:42:50,643 INFO mapreduce.Job:  map 17% reduce 0%
2022-11-21 17:42:51,648 INFO mapreduce.Job:  map 26% reduce 0%
2022-11-21 17:42:52,654 INFO mapreduce.Job:  map 36% reduce 0%
2022-11-21 17:42:53,660 INFO mapreduce.Job:  map 49% reduce 0%
2022-11-21 17:42:54,666 INFO mapreduce.Job:  map 58% reduce 0%
2022-11-21 17:42:56,676 INFO mapreduce.Job:  map 61% reduce 0%
2022-11-21 17:42:57,682 INFO mapreduce.Job:  map 62% reduce 0%
2022-11-21 17:42:58,687 INFO mapreduce.Job:  map 65% reduce 0%
2022-11-21 17:42:59,691 INFO mapreduce.Job:  map 67% reduce 0%
2022-11-21 17:43:00,695 INFO mapreduce.Job:  map 68% reduce 0%
2022-11-21 17:43:02,704 INFO mapreduce.Job:  map 70% reduce 0%
2022-11-21 17:43:03,709 INFO mapreduce.Job:  map 77% reduce 0%
2022-11-21 17:43:04,714 INFO mapreduce.Job:  map 80% reduce 0%
2022-11-21 17:43:05,718 INFO mapreduce.Job:  map 83% reduce 0%
2022-11-21 17:43:06,723 INFO mapreduce.Job:  map 86% reduce 0%
2022-11-21 17:43:07,729 INFO mapreduce.Job:  map 91% reduce 0%
2022-11-21 17:43:08,734 INFO mapreduce.Job:  map 100% reduce 0%
2022-11-21 17:43:14,764 INFO mapreduce.Job:  map 100% reduce 17%
2022-11-21 17:43:15,769 INFO mapreduce.Job:  map 100% reduce 35%
2022-11-21 17:43:16,774 INFO mapreduce.Job:  map 100% reduce 52%
2022-11-21 17:43:17,779 INFO mapreduce.Job:  map 100% reduce 65%
2022-11-21 17:43:18,784 INFO mapreduce.Job:  map 100% reduce 78%
2022-11-21 17:43:19,788 INFO mapreduce.Job:  map 100% reduce 96%
2022-11-21 17:43:20,792 INFO mapreduce.Job:  map 100% reduce 100%
2022-11-21 17:43:21,804 INFO mapreduce.Job: Job job_1668611926937_0321 completed successfully
2022-11-21 17:43:21,894 INFO mapreduce.Job: Counters: 57
        File System Counters
                FILE: Number of bytes read=44696807
                FILE: Number of bytes written=112213389
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=3158544657
                HDFS: Number of bytes written=24438590
                HDFS: Number of read operations=322
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=69
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Killed map tasks=1
                Killed reduce tasks=1
                Launched map tasks=69
                Launched reduce tasks=23
                Data-local map tasks=21
                Rack-local map tasks=48
                Total time spent by all maps in occupied slots (ms)=3300468
                Total time spent by all reduces in occupied slots (ms)=350536
```

```
        Total time spent by all map tasks (ms)=825117
        Total time spent by all reduce tasks (ms)=87634
        Total vcore-milliseconds taken by all map tasks=825117
        Total vcore-milliseconds taken by all reduce tasks=87634
        Total megabyte-milliseconds taken by all map tasks=3379679232
        Total megabyte-milliseconds taken by all reduce tasks=358948864
    Map-Reduce Framework
        Map input records=14751837
        Map output records=14751768
        Map output bytes=537861367
        Map output materialized bytes=44706191
        Input split bytes=18860
        Combine input records=14751768
        Combine output records=1107201
        Reduce input groups=435902
        Reduce shuffle bytes=44706191
        Reduce input records=1107201
        Reduce output records=435494
        Spilled Records=2214402
        Shuffled Maps =1587
        Failed Shuffles=0
        Merged Map outputs=1587
        GC time elapsed (ms)=24513
        CPU time spent (ms)=848020
        Physical memory (bytes) snapshot=80260046848
        Virtual memory (bytes) snapshot=443563794432
        Total committed heap usage (bytes)=86339223552
        Peak Map Physical memory (bytes)=1192259584
        Peak Map Virtual memory (bytes)=4886904832
        Peak Reduce Physical memory (bytes)=527159296
        Peak Reduce Virtual memory (bytes)=4841041920
    Shuffle Errors
        BAD_ID=0
        CONNECTION=0
        IO_ERROR=0
        WRONG_LENGTH=0
        WRONG_MAP=0
        WRONG_REDUCE=0
    File Input Format Counters
        Bytes Read=0
    File Output Format Counters
        Bytes Written=24438590
```

4. Results

The results can be seen from the aggregate directory. Below is a sample of the outputs.

```
"2022-01-01","01","Alabama",24.4286,74.9250,750.8292
"2022-01-01","02","Alaska",34.0000,-21.1250,-9999.0000
"2022-01-01","04","Arizona",38.6154,33.4236,976.3958
"2022-01-01","05","Arkansas",28.0000,-9999.0000,-9999.0000
"2022-01-01","06","California",45.4865,42.9166,938.6042
"2022-01-01","08","Colorado",39.1500,11.4917,816.2525
"2022-01-01","09","Connecticut",25.0000,-9999.0000,-9999.0000
"2022-01-01","12","Florida",38.1000,72.5417,-9999.0000
"2022-01-01","13","Georgia",27.3889,71.4643,997.0750
"2022-01-01","15","Hawaii",9.0000,60.4583,-9999.0000
"2022-01-01","16","Idaho",24.5000,4.0875,977.0834
"2022-01-01","17","Illinois",34.6667,38.0694,989.4584
"2022-01-01","18","Indiana",23.5000,34.0833,-9999.0000
"2022-01-01","19","Iowa",42.7000,11.4962,984.2220
"2022-01-01","21","Kentucky",30.7500,59.5000,979.0833
"2022-01-01","22","Louisiana",32.4211,77.1667,1017.3333
"2022-01-01","23","Maine",27.1667,43.2500,1004.6250
"2022-01-01","24","Maryland",18.3750,53.5629,991.4408
"2022-01-01","25","Massachusetts",43.3077,47.2411,995.5744
"2022-01-01","26","Michigan",31.5000,-9999.0000,-9999.0000
"2022-01-01","27","Minnesota",27.0000,-20.5000,-9999.0000
"2022-01-01","28","Mississippi",33.2500,75.0917,-9999.0000
"2022-01-01","29","Missouri",30.4545,29.5293,980.4441
"2022-01-01","30","Montana",45.2667,-7.7969,-9999.0000
"2022-01-01","31","Nebraska",25.2857,2.7333,967.7083
"2022-01-01","32","Nevada",40.0000,35.1230,957.9583
"2022-01-01","33","New Hampshire",7.0000,-9999.0000,-9999.0000
"2022-01-01","34","New Jersey",12.0000,-9999.0000,-9999.0000
"2022-01-01","35","New Mexico",15.8750,33.5625,-9999.0000
"2022-01-01","36","New York",32.5769,-9999.0000,-9999.0000
"2022-01-01","37","North Carolina",21.5556,69.3056,978.9271
"2022-01-01","38","North Dakota",30.3333,-15.2833,951.0125
"2022-01-01","39","Ohio",29.1739,51.6208,968.5083
"2022-01-01","40","Oklahoma",33.1000,33.0262,-9999.0000
"2022-01-01","41","Oregon",46.6667,14.3333,918.4764
"2022-01-01","42","Pennsylvania",25.5000,54.2931,974.1166
"2022-01-01","44","Rhode Island",13.6667,49.4167,1001.1041
"2022-01-01","45","South Carolina",27.8571,73.3583,-9999.0000
"2022-01-01","47","Tennessee",29.2857,65.1667,869.7083
```

The results are tested against a local program without using the MapReduce framework (AirAggregateLocal.java). It shows that the two results have some inconsistent precision at the fourth decimal place but the error is acceptable.