Milestone 3

Team Name

Urban Analysts

Team Members

Angelina Paredes
Bryton LeValley
Taryn Burns
Melanie Bell

1. Problem Statement and Algorithm Description **3**

a. Give a formal problem statement. It consists of (1) problem input; (2) problem output, and (3) object function/goal for your solution. 3

b. Give a pseudo-code and corresponding description of your sequential algorithms. If your project targets at multiple problem, give the solution for each of the problem. Your description should correspond to the line number in your pseudo-code. 3

2. Discussion of the optimization techniques. You may leave this part for now and take more time to work on this and submit it as the answer to the last question in your homework 3/Milestone 4. **3**

3. Initial experimental plan. At this point you have around 1 months to start implementing and testing your solutions. You should have an initial experimental plan as a part of your final report. The experimental plan has several components you should report: **3**

a. Dataset description (use what you have in M2) 3

b. Algorithm description (specify what are baseline algorithms, and what are your algorithms you will test; give references for each of the baseline algorithm if applicable) 3

c. Setting: CPU and memory information of your machine. If you are using platforms e.g., Hadoop, Spark, give a description of the platform. 4

d. Plan: in sections, you specify (1) the algorithms you want to compare, (2) the performance metric (runtime, data shipment, accuracy, etc), (3) the factor that you are testing that may have an impact of the result (size of datasets, size of query, etc). You should give a statement on what would be the expected result in each subsection you may expect and why. 4

1. Problem Statement and Algorithm Description
    a. Give a formal problem statement. It consists of (1) problem input; (2) problem output, and (3) object function/goal for your solution.

Our project consists of taking WSU CASAS data as input then rearranging their data into multiple sections dependent upon the sensor type, in order to be read easier. The goal is to see how long each sensor is active for in order to better understand how active the residents are in the WSU smart home/apartment.

    b. Give a pseudo-code and corresponding description of your sequential algorithms. If your project targets at multiple problem, give the solution for each of the problem. Your description should correspond to the line number in your pseudo-code.

        i. Query sensor statistics:
        Query user for date and time
        Using SQL find last activation time, battery level, average duration, etc
        Output various statistics
        SQL Query example for last activation time:
        SELECT MAX(datetime) FROM table
        WHERE datetime <= {user_input.datetime} AND sensor_output = 'ON'

        ii. Monitor on/off duration with histogram:

        iii. Monitor battery drain: Appendix a.
        iv. Trajectory identification
        v. Activity recognition

2. Discussion of the optimization techniques. You may leave this part for now and take more time to work on this and submit it as the answer to the last question in your homework 3/Milestone 4.

Another optimization technique that we will be implementing is utilizing indexing to increase the efficiency of our SQL queries.

3. Initial experimental plan. At this point you have around 1 months to start implementing and testing your solutions. You should have an initial experimental plan as a part of your final report. The experimental plan has several components you should report:
    a. Dataset description (use what you have in M2)

Our dataset consists of 2,804,813 tuples, and five attributes. The attributes consist of date, time, sensor, status, and activity. The physical storage size of the file is 100.5 MB.

There are two candidate keys for the dataset, date/time and sensor type. Each sensor activation time is recorded, the sensor type isn't unique but by adding date/time as a key, we can generate a unique identifier for each tuple. Date/time and sensor name functionally determines all attributes in the dataset. Another functional dependency is sensor name determines the type attribute.

    b. Algorithm description (specify what are baseline algorithms, and what are your algorithms you will test; give references for each of the baseline algorithm if applicable)

K-means clustering seems like a great algorithm to use for our data set. This is because k-means creates k groups that contain members that are similar to one another. With an algorithm like this, it can help us combine like data.

Another likely algorithm is association rule mining algorithm. This algorithm is a learning algorithm that looks for associations that co-occur with a high degree of frequency. This will discover interesting relations between variables in our dataset. It often identifies with the if-then association.

Support Vector machine will be used to implement the activity recognition for each event that is triggered based on prior research. Prior research from activity recognition "found that SVMs achieve consistently stronger performance than other approaches (Cook 2013).The activity discovery algorithm was used in the WSU CASAS case. This is because it was "one way to handle unlabeled data is to design an unsupervised learning algorithm to discover activities from unlabeled sensor data." (Cook 6)

c. Setting: CPU and memory information of your machine. If you are using platforms e.g., Hadoop, Spark, give a description of the platform.

We are using PostgreSQL, which is also known as Postgres. It is a free and open-source relational database system, which emphasizes technical standards compliance and extensibility. It gives the user a range of use, whether it be on their desktop or their browser. Additionally, it can handle a range of workloads.

Additionally we are using Amazon Cloud Services with our Postgres SQL. Amazon Cloud Services, also known as AWS, Amazon Web Services. AWS is the world's most comprehensive and broadly adopted cloud platform that offers many different services and tools to use. It is a subsidiary of Amazon that provides on-demand cloud computing platforms and APIs to everyone, including individuals and companies.

d. Plan: in sections, you specify (1) the algorithms you want to compare, (2) the performance metric (runtime, data shipment, accuracy, etc), (3) the factor that you are testing that may have an impact of the result (size of datasets, size of query, etc). You should give a statement on what would be the expected result in each subsection you may expect and why.

Algorithms we want to compare are K-means, association rule mining algorithm, Support Vector Machine (SVM)

Performance metric: Runtime will be measured to understand the performance of each of the algorithms. There will be different implementations of each algorithm to look at due to the size of the dataset we are using. Different python libraries will be compared to determine which function is the most efficient. There are different distance metrics we can use for k-means that we must consider due to the large dataset that we're using. Accuracy will be measured to ensure that our algorithms are labeling activities correctly. Accuracy can be determined by generating training and testing sets from the results of k-means to determine the most efficient number of groups we should use.
To measure the accuracy of our SVM, we'll manually label a couple weeks of activities, this was performance metric was completed in prior research by evaluating "performance as the percentage of sensor events that were correctly labeled across all of the apartments, with no additional customized training for each apartment" (Cook 2013). There is previous data that we can analyze to determine that our labels are correct.

The factor that we are testing that may have an impact on the result is dataset size. We expect that the runtime is going to be significant factor due to the large dataset. The

size of the data will play a large factor into the runtime of each algorithm. We'll also need to utilize indexes to determine if this will improve the run time of our query results.

References:

Cook, D. J., Crandall, A. S., Thomas, B. L., & Krishnan, N. C. (2013). CASAS: A Smart Home in a Box. *Computer*, *46*(7), 62–69. https://doi.org/10.1109/MC.2012.328

Appendix:

A.

in python:
Load BattVolt.csv as BV
open file Batt_drain.csv as BD
Batt_list = [list of all unique batt_volt sensors]

```
for all item in Batt_list:
        BD.write(f"{item},")
        first_charge.level = 0
        first_charge.time = 0
        last_charge.level = 0
        last_charge.time = 0

        For all rows in BV:

                if item == row[1]
                        if row[2] > (last_charge + 100)
# the 100 is to mitigate minor increases in battery charge due to inconsistencies in monitoring
                                power_drain = (first_charge.level - last_charge.level)\
/(first_charge.time - last_charge.time)
                                first_charge.level = row[2]
                                first_charge.time = row[0]
                                power_drain_list.add(power_drain)
                        last_charge.level = row{2]
                        last_charge.time = row[0]
        average_power_drain = Average of power_drain_list[1:] #minus the first result
        BD.write(f"{average_power_drain}/n")
```

In SQL:
CREATE TABLE sensor_drain
INSERT INTO sensor_drain Batt_drain.csv

in app:
Query user for datetime
show battery level for sensors, projected drain time, and highlight red if sensor will run out of battery within the month