

Configuration Manual

MSc Research Project
Data Analytics

Nayan Sharma

Student ID: X21167818

School of Computing
National College of Ireland

Supervisor: Prof. Paul Stynes

**National College of Ireland
Project Submission Sheet
School of Computing**



Student Name:	Nayan Sharma
Student ID:	X21167818
Programme:	Data Analytics
Year:	2022
Module:	MSc Research Project
Supervisor:	Prof. Paul Stynes
Submission Due Date:	15/12/2022
Project Title:	Configuration Manual
Word Count:	1391
Page Count:	19

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	15th December 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission , to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project , both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Nayan Sharma

ID: 21167818

MSCDAD JAN22 A

December 2022

1 Introduction

This paper has a detailed, step-by-step plan for how to carry out this research study. It will explain everything I did to finish my study and how those steps can be done again.

1.1 Prerequisites for conducting the research project

To conduct this research project, make sure you have an Azure and Datarbricks account set up and a small cost that would require you to use these services. There are very few prerequisites as the main aim of the research project was to make it simpler and automated following the best practices of security. Hence, we will do the most out of Azure services and integrate a few third-party libraries such as scikit-learn-1.1.3, Seaborn-0.12.1, and yellowbrick-1.5. TO integrate these libraries automated code is already written in .ipynb files.

- Using url ¹ create Azure account
- Using url ² create databricks free account where azure account credentials are required.
- Using url ³ create Github free account where code and dataset will be uploaded

2 Setup Environment

2.0.1 Setup Databricks environment

Setup the Databricks cluster as per Table 1 and create a cluster in data bricks as shown in Figure 1.

¹<https://azure.microsoft.com/en-gb/free/>

²<https://www.databricks.com/try-databricksaccount>

³<https://github.com/signup?source=login>

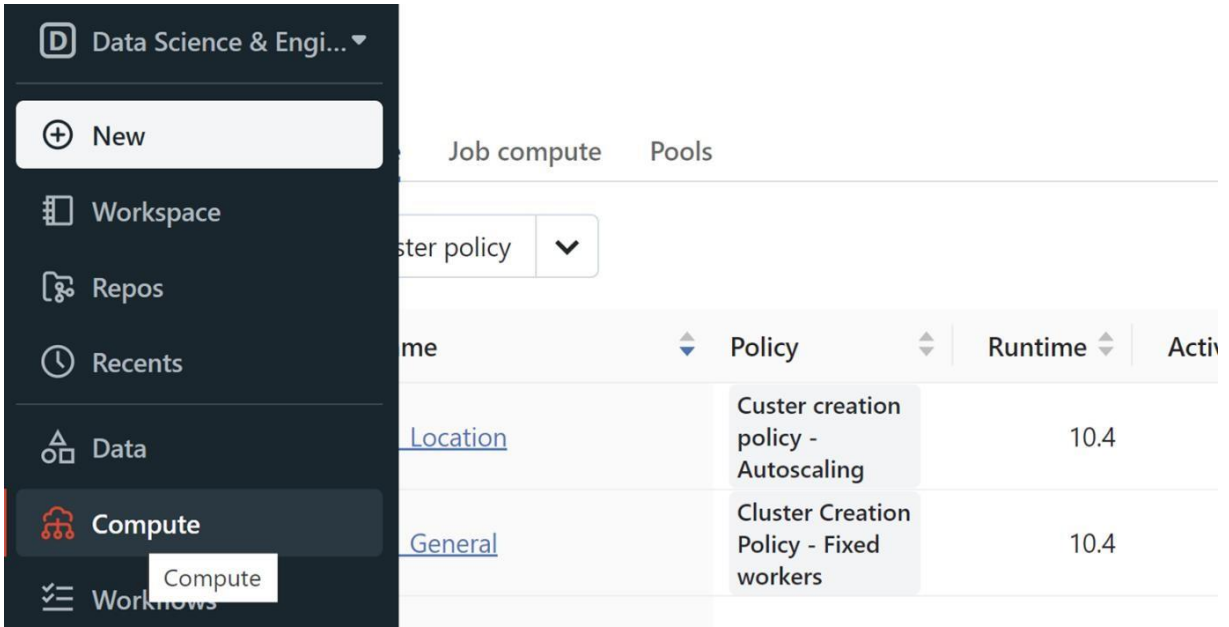


Figure 1: Databricks Cluster

Table 1: Databricks Cluster Specification

Type	Value
Databricks Runtime Version	9.1 LTS (includes Apache Spark 3.1.2, Scala 2.12)
Worker Type	Standard D8s v3 (32 GB, 8 Cores)
Driver Type	Standard D8s v3 (32 GB, 8 Cores)
Number of workers	Autoscale (1-4)

2.0.2 Setup Azure environment

Setup the Azure Key to access the Storage account and contains in Azure Data lake storage as shown in Figure 2.

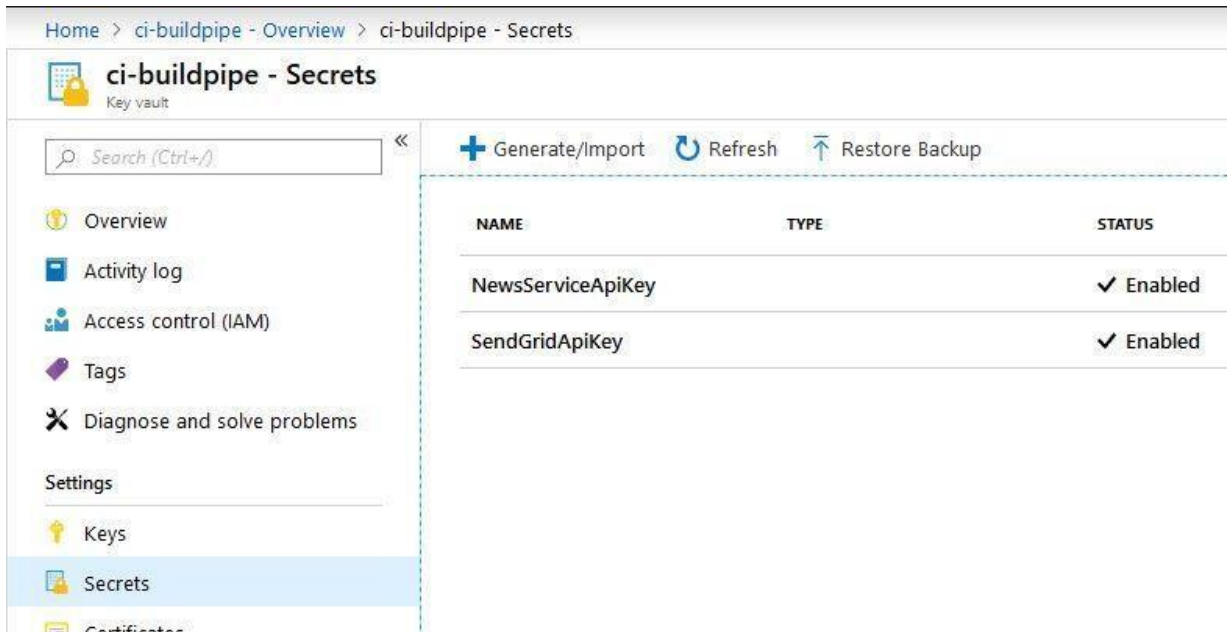


Figure 2: Azure key setup

2.0.3 Setup GitHub

Integrate the GitHub account with Azure data bricks by going into the data bricks user setting as shown in Figure 3. Before that, we need to create a repository for the code and generate a token/key from GitHub to authenticate the user from Databricks. We can follow standard documentation from url ⁴. Either you can upload all .ipynb files into databricks as shown in Figure 4 or upload file in GIT and add repo into Databricks as shown in Figure 5

2.0.4 Setup databricks cli

- Install latest version of python from ⁵ into personal laptop.
- Install data bricks cli by using the command into PowerShell or cmd terminal is shown in Figure 6.

⁴<https://docs.databricks.com/repos/index.html>

⁵<https://www.python.org/downloads/>

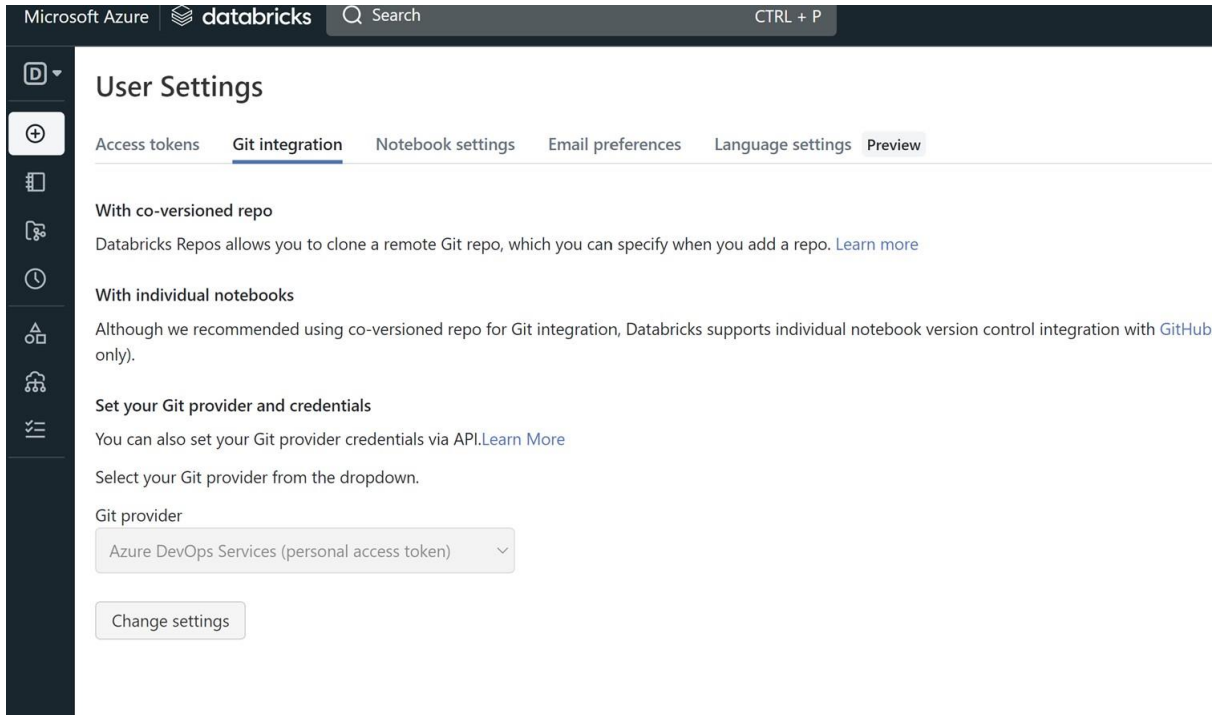


Figure 3: GitHub Integration

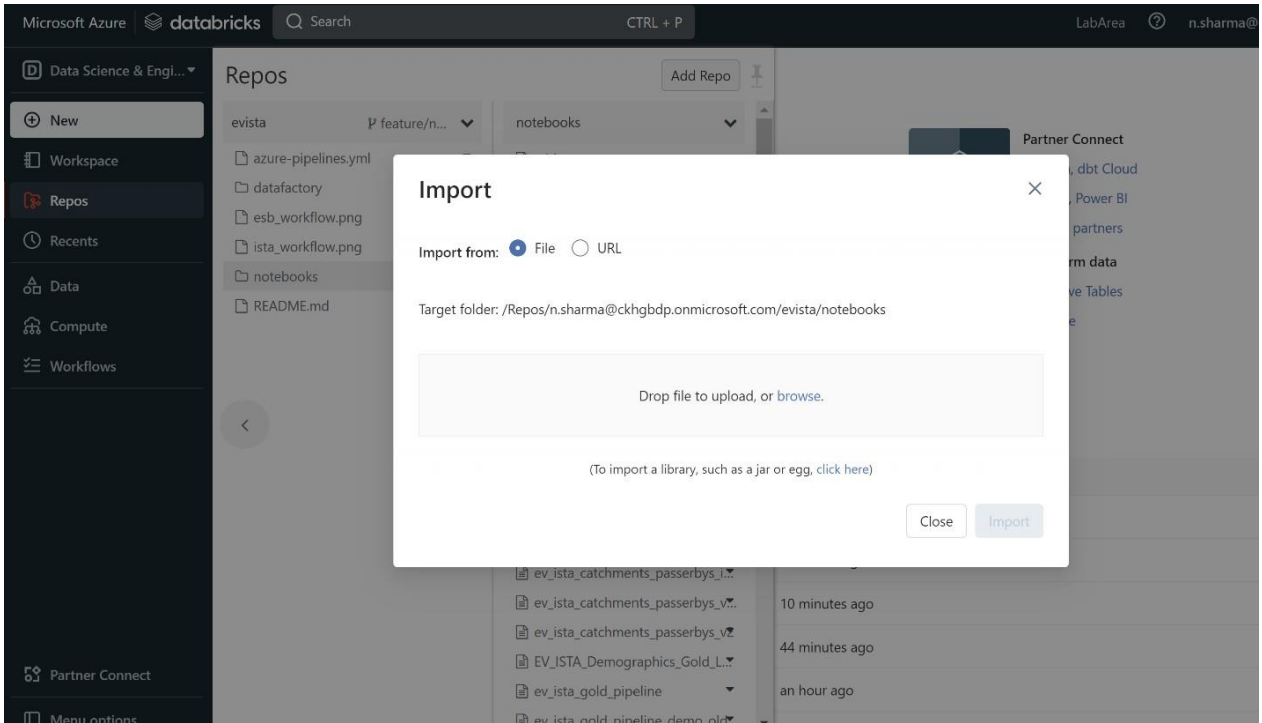


Figure 4: Import .ipynb in Databricks

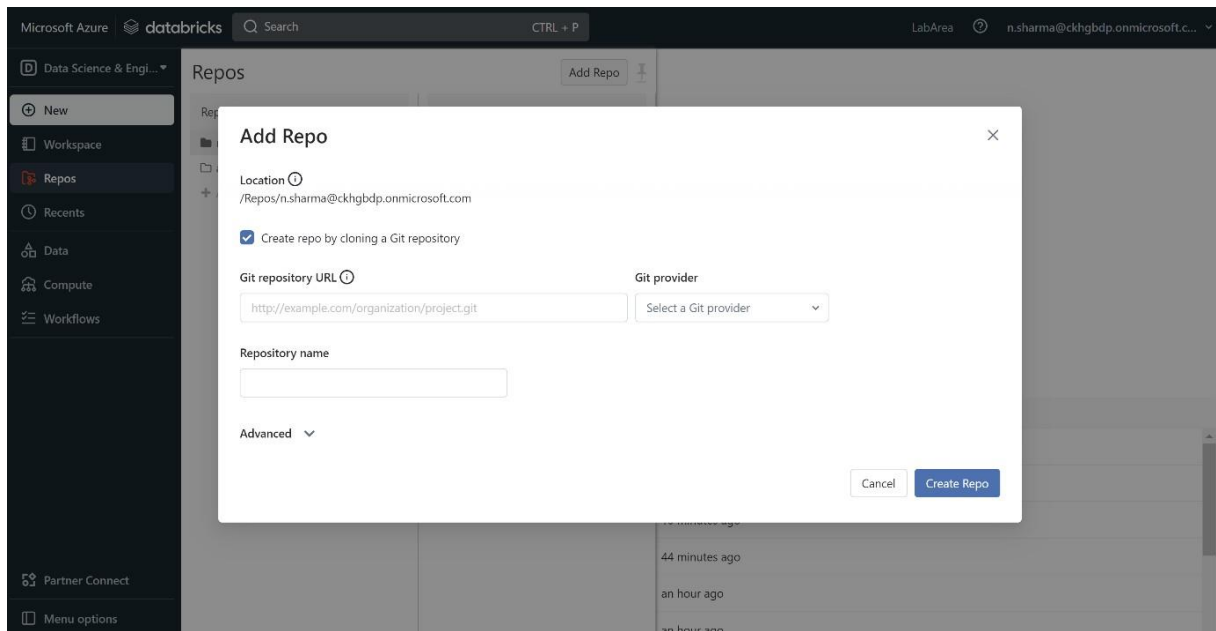


Figure 5: Add GIT Repo into Databricks

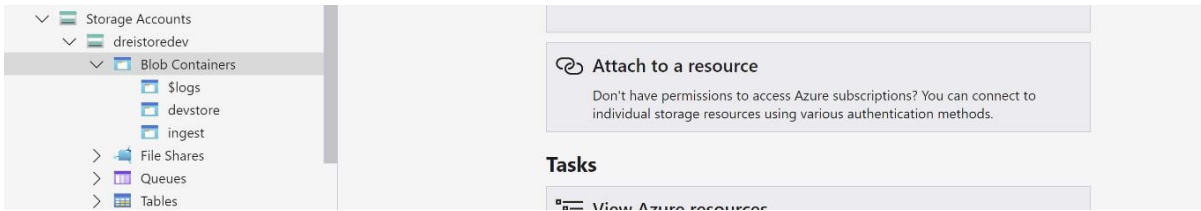


Figure 10: Azure Container creation

3 Collecting data-sets

For this research, we are using seven datasets 1. UK postcode dataset.csv (*uk-postcode-electricity-consumption 2022*) 2. UK Car ownership dataset.csv (*Transport 2022*) 3. Mobility Footfall dataset.csv (*Uk Mobility Footfall (2022)*) 4. Mobility passerby dataset.csv (*Uk Mobility passerby (2022)*) 5. UK traffic count.csv (*Trafficcount—data.gov.uk (2022)*) 6. UK Postcode mapping.csv (*ukpostcode - NSPL (2022)*) 7. UK Outputareas with population.geojson (*ukpostcodegeojson (2022)*). There is a requirement to upload all these datasets to Azure containers as shown in Figure 11 so that Databricks notebook can consume data for processing and feed it into the model.

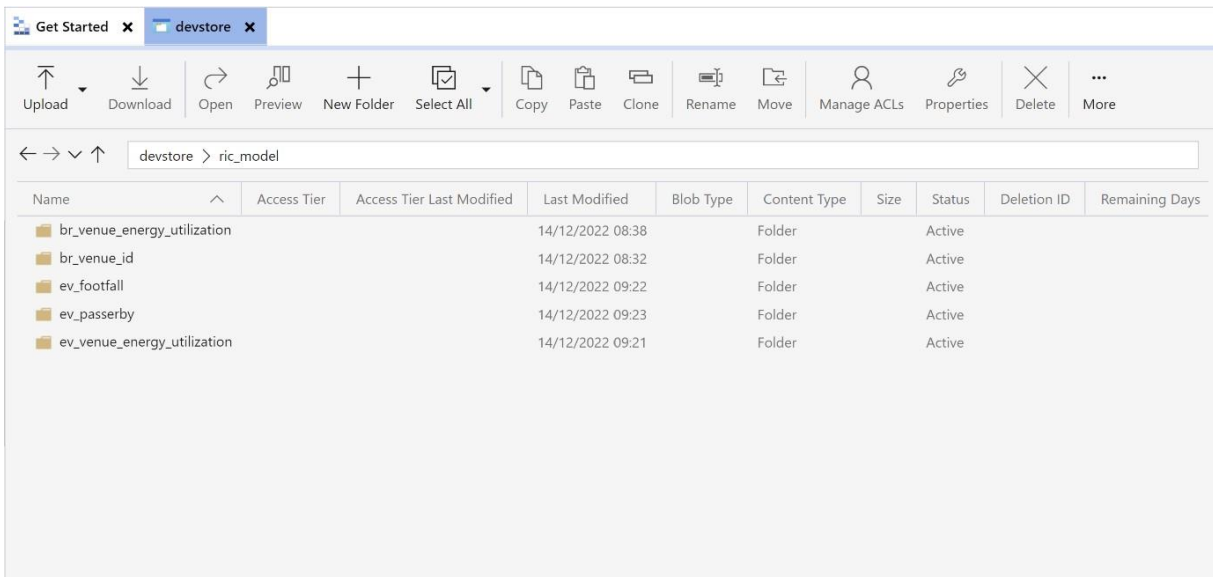


Figure 11: Files at Azure

4 Notebooks

There are eight Notebooks involved in this research which are described below:

- `ev_static_csv_file_load.ipynb`
- `ev_carownership_excel_ingestion.ipynb`
- `ev_outputareas_carownership_ingestion.ipynb`
- `ev_model_dataset_creation.ipynb`
- `ev_create_dataset_traffic_carownershp.ipynb`
- `ev_unsupervised_model_monthly.ipynb`
- `ev_regression_model_monthly_data.ipynb`
- `ev_spatial_model.ipynb`

4.1 `ev_static_csv_file_load.ipynb`

This is the first notebook that should run. The main purpose of this notebook is to create a database in data bricks with name `ric_model` use all the necessary datasets and create a final table that acts as input to `ev_model_dataset_creation.ipynb`. You have to run this notebook multiple times with the table name and path of the files uploaded on Azure as shown in Table 2. Figure 12 shows the notebook where we need to mention the path and table name.

Table 2: Table Creation

File Path	Table Name
/Raw/ric/ev passerby.csv	ric model.ev passerby
/Raw/ric/ev footfall.csv	ric model.ev footfall
/Raw/ric/ev passerby.csv	ric model.ev passerby
/Raw/ric/energy utilization.csv	ric model.ev venue energy utilization
/Raw/ric/post code uk.csv	ric model.br postcodes uk
/Raw/ric/traffic count.csv	ric model.br traffic counts uk

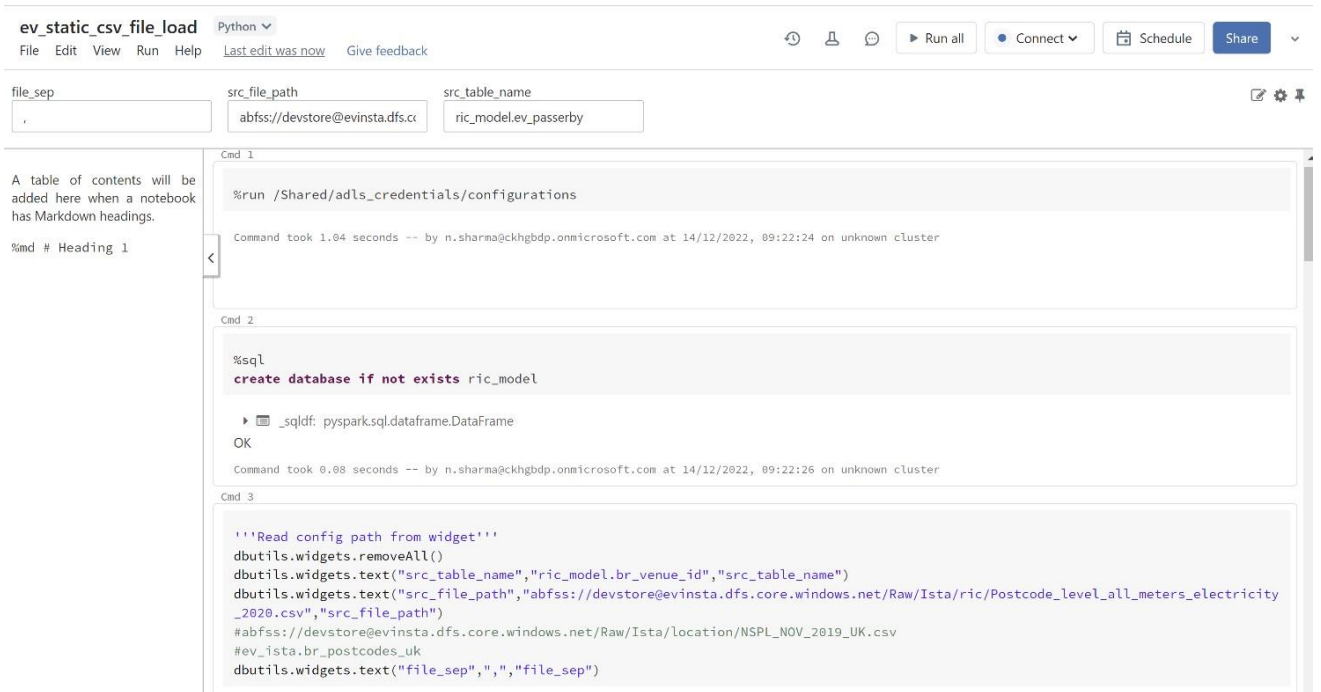


Figure 12: ev_static_csv_file load

4.2 ev_carownership_excel_ingestion.ipynb

The main purpose of this notebook is to download car ownership datasets from the ⁶ and load it into stage table ric_model.br_carownership_stat. The notebook as shown in Figure 13.

ev_carownership_excel_ingestion Python

File Edit View Run Help Last edit was 27 minutes ago Give feedback

Note: This notebook must...
All vehicles veh0122.ods
Low emission vehicles ve...

```
Command complete

Cmd 37

%sql

CREATE TABLE IF NOT EXISTS ric_model.br_carownership_stat
(postcode_dist string,cars_count int,car_type string,year int,quarter string,country string,date string,load_datetime timestamp);

MERGE INTO ric_model.br_carownership_stat a
USING carownership
ON a.postcode_dist = carownership.postcode_dist and
a.car_type = carownership.car_type and
a.year = carownership.year and
a.quarter = carownership.quarter and
a.country = carownership.country

WHEN MATCHED THEN
UPDATE SET a.cars_count = carownership.cars_count,
a.load_datetime = carownership.load_datetime

WHEN NOT MATCHED THEN
INSERT (postcode_dist, car_type,year,quarter,country,cars_count,date,load_datetime) VALUES (carownership.postcode_dist,
carownership.car_type,carownership.year,carownership.quarter,carownership.country,carownership.cars_count,carownership.date,carowne
rship.load_datetime);

Command complete

Cmd 38

%sql
drop table carownership

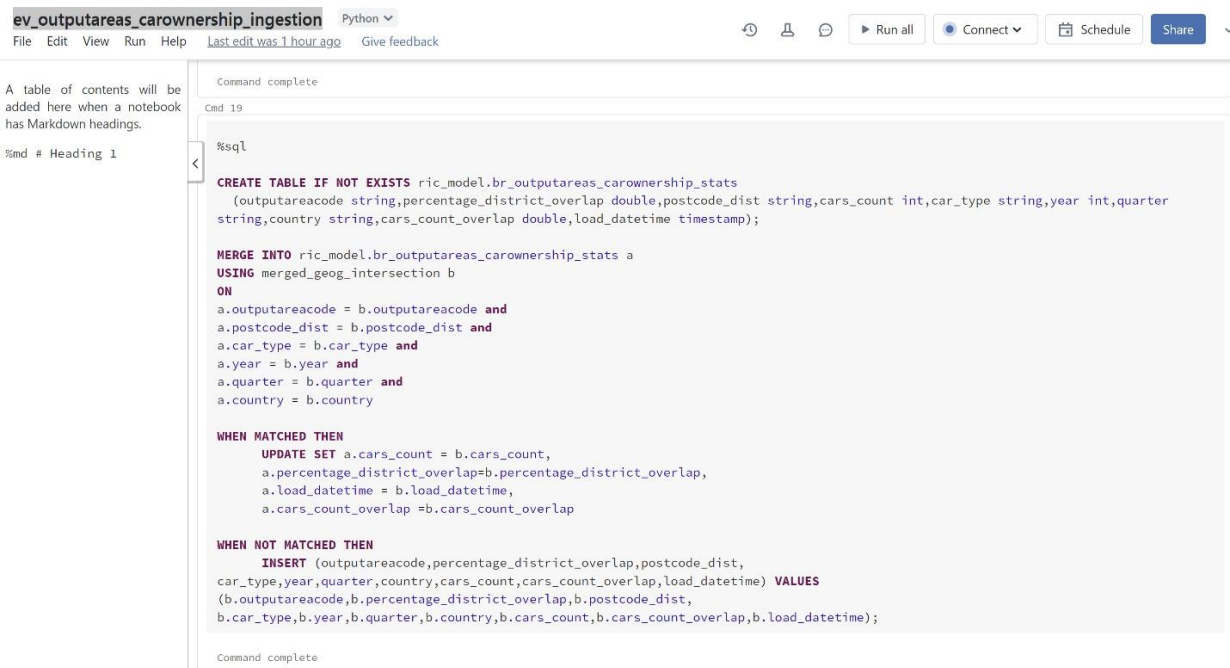
Command complete
```

Figure 13: ev car ownership excel ingestion

⁶<https://www.gov.uk/government/collections/vehicles-statistics>

4.3 ev_outputareas_carownership_ingestion.ipynb

The main purpose of this notebook is to use ric_model.br_carownership_stat table and load it into the final table ric_model.br_outputareas_carownership_stats after mapping it with the postcode. The notebook as shown in Figure 14.



The screenshot shows a Jupyter Notebook titled "ev_outputareas_carownership_ingestion" with a Python kernel. The notebook content includes a table of contents placeholder and a SQL query cell. The query is as follows:

```
%%sql
CREATE TABLE IF NOT EXISTS ric_model.br_outputareas_carownership_stats
(outputareacode string,percentage_district_overlap double,postcode_dist string,cars_count int,car_type string,year int,quarter
string,country string,cars_count_overlap double,load_datetime timestamp);

MERGE INTO ric_model.br_outputareas_carownership_stats a
USING merged_geog_intersection b
ON
a.outputareacode = b.outputareacode and
a.postcode_dist = b.postcode_dist and
a.car_type = b.car_type and
a.year = b.year and
a.quarter = b.quarter and
a.country = b.country

WHEN MATCHED THEN
UPDATE SET a.cars_count = b.cars_count,
a.percentage_district_overlap=b.percentage_district_overlap,
a.load_datetime = b.load_datetime,
a.cars_count_overlap =b.cars_count_overlap

WHEN NOT MATCHED THEN
INSERT (outputareacode,percentage_district_overlap,postcode_dist,
car_type,year,quarter,country,cars_count,cars_count_overlap,load_datetime) VALUES
(b.outputareacode,b.percentage_district_overlap,b.postcode_dist,
b.car_type,b.year,b.quarter,b.country,b.cars_count,b.cars_count_overlap,b.load_datetime);
```

Figure 14: ev_outputareas_carownership_ingestion

4.4 ev_model_dataset_creation.ipynb

This is the second notebook that should run. The main purpose of this notebook is to use energy consumption, passerby, and footfall, and aggregate them all and create a single output table with the name ric_model.ev_model_dataset_monthly that acts as input to various machine-learning models. The notebook is shown in Figure 15.

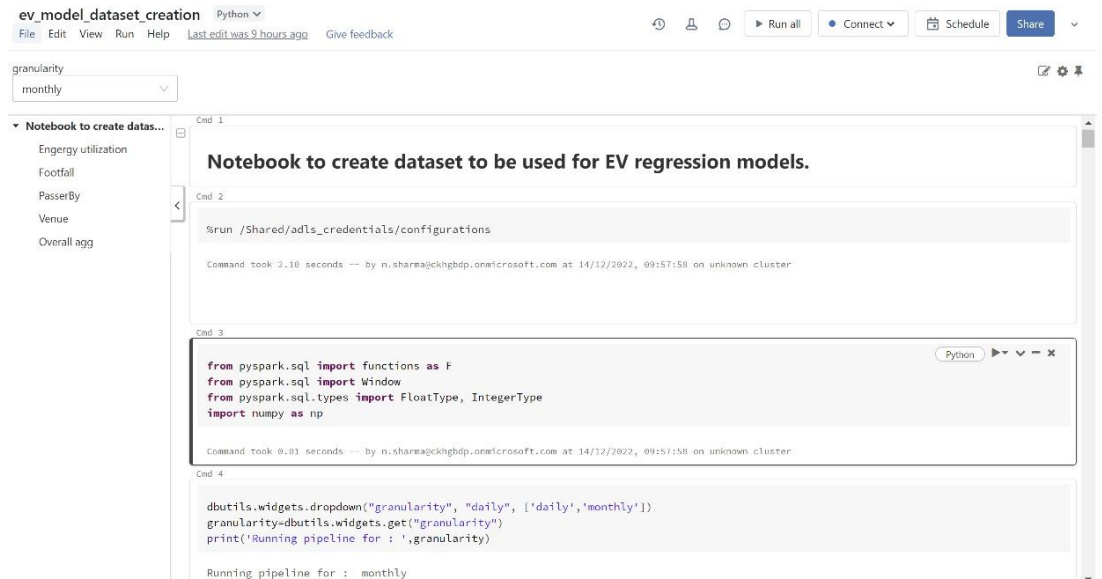


Figure 15: ev_model_dataset_creation

4.5 ev_create_dataset_traffic_carownershp.ipynb

This is the third notebook that should run. The main purpose of this notebook is to combine traffic count, car ownership, and output table from the second notebook i.e ric_model.ev_model_dataset_monthly. Output is ric_model.br_ev_model_dataset which contains the total motor and total car count. The notebook is shown in Figure 16.

ev_create_dataset_traffic_carownership Python

File Edit View Run Help Last edit was 5 hours ago Give feedback

▶ Run all ■ Terminated ▾ 📅 Schedule Share ▾

Combining carownership ...
venue table
Adding Mobility Data

```
%sql
create or replace table ric_model.br_ev_model_dataset as
select a.*,b.total_motor,b.car_counts
from ric_model.ev_model_dataset_monthly a left outer join ric_model.br_venue_traffic_carownership b on a.venue_id=b.postcode
--where b.year=2021
```

Notebook detached
cluster not in usable state

▶ (6) Spark Jobs
▶ _sqlidf: pyspark.sql.dataframe.DataFrame = [num_affected_rows: long, num_inserted_rows: long]
Query returned no results
Command took 11.88 seconds -- by n.sharma@ckhgbdp.onmicrosoft.com at 14/12/2022, 15:32:11 on DE_General

Cmd 14

```
%sql
select * from ric_model.br_ev_model_dataset
```

▶ (3) Spark Jobs
▶ _sqlidf: pyspark.sql.dataframe.DataFrame = [venue_id: string, year: integer... 74 more fields]

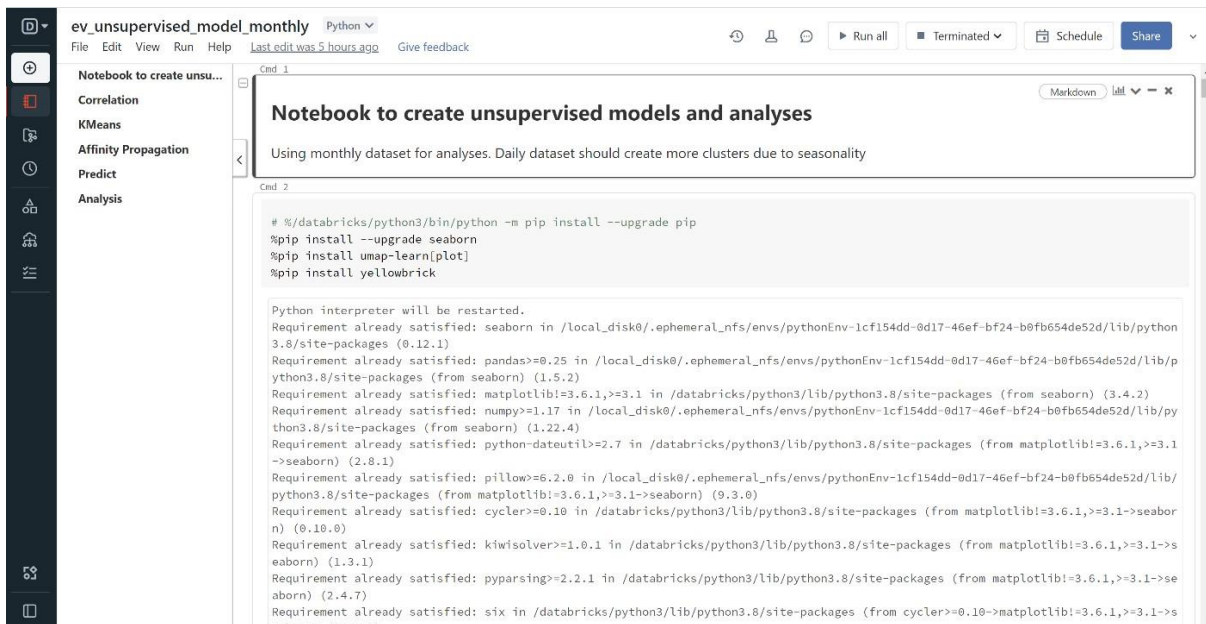
	venue_id	year	month	energy_utilization	total_footfall	age_band_18_24	age_band_25_34	age_band_35_4
1	BD18 3ST	2022	9	229.33	144988	0.10329130686677518	0.18471873534361463	0.14377741606
2	BD18 3ST	2022	12	148.49	null	null	null	null
3	BD18 3ST	2022	8	291.28	164001	0.10604813385284236	0.1867854464302047	0.14424302290
4	BD18 3ST	2022	10	197.14	148782	0.09980373969969486	0.1866153163689156	0.14281969593
5	BD18 3ST	2022	11	121.98	126363	0.09409399903452752	0.18556064670829278	0.14330144108
6	BS1 6HY	2022	2	388.43	624580	0.19901373723141952	0.2209164558583368	0.14471164622
7	BS1 6HY	2022	6	1864.88	738566	0.18546209817402914	0.2287161878559262	0.15339861264

ENG 20:58

Figure 16: ev_create_dataset_traffic_carownership

4.6 ev_unsupervised_model_monthly.ipynb

The main purpose of this notebook is to take the ric_model.br_ev_model_dataset table created and feed into the KMean clustering model. Before feeding it into the Machine learning model EDA and min max scaler have been performed. The notebook is shown in Figure 17.



```
ev_unsupervised_model_monthly Python
File Edit View Run Help Last edit was 5 hours ago Give feedback

Notebook to create unsu...
Correlation
KMeans
Affinity Propagation
Predict
Analysis

Notebook to create unsupervised models and analyses
Using monthly dataset for analyses. Daily dataset should create more clusters due to seasonality

# /databricks/python3/bin/python -m pip install --upgrade pip
%pip install --upgrade seaborn
%pip install umap-learn[plot]
%pip install yellowbrick

Python interpreter will be restarted.
Requirement already satisfied: seaborn in /local_disk0/.ephemeral_nfs/envs/pythonEnv-1cf154dd-0d17-46ef-bf24-b0fb654de52d/lib/python3.8/site-packages (0.12.1)
Requirement already satisfied: pandas>=0.25 in /local_disk0/.ephemeral_nfs/envs/pythonEnv-1cf154dd-0d17-46ef-bf24-b0fb654de52d/lib/python3.8/site-packages (from seaborn) (1.5.2)
Requirement already satisfied: matplotlib!=3.6.1,>=3.1 in /databricks/python3/lib/python3.8/site-packages (from seaborn) (3.4.2)
Requirement already satisfied: numpy>=1.17 in /local_disk0/.ephemeral_nfs/envs/pythonEnv-1cf154dd-0d17-46ef-bf24-b0fb654de52d/lib/python3.8/site-packages (from seaborn) (1.22.4)
Requirement already satisfied: python-dateutil=2.7 in /databricks/python3/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (2.8.1)
Requirement already satisfied: pillow>=6.2.0 in /local_disk0/.ephemeral_nfs/envs/pythonEnv-1cf154dd-0d17-46ef-bf24-b0fb654de52d/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (9.3.0)
Requirement already satisfied: cycler>=0.10 in /databricks/python3/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (0.10.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /databricks/python3/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (1.3.1)
Requirement already satisfied: pyparsing>=2.2.1 in /databricks/python3/lib/python3.8/site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (2.4.7)
Requirement already satisfied: six in /databricks/python3/lib/python3.8/site-packages (from cycler>=0.10->matplotlib!=3.6.1,>=3.1->seaborn) (1.16.0)
```

Figure 17: ev_unsupervised_model_monthly

4.7 ev_regression_model_monthly_data.ipynb

The main purpose of this notebook is to take the ric_model.br_ev_model_dataset table created in the first step as input, feed it into six machine-learning models, and save the output into a table that acts as input for the spatial model table ric_model.regression_model_output. The notebook is shown in Figure 18.

```
ev_regression_model_monthly_data Python
File Edit View Run Help Last edit was 5 hours ago Give feedback
Run all Terminated Schedule Share

Linear Regression with all...
Predict
Linear Regression with to...
Regression model with tr...

Cmd 1
Python
# /databricks/python3/bin/python -m pip install --upgrade pip
%pip install --upgrade seaborn
%pip install umap-learn[plot]
%pip install yellowbrick

Python interpreter will be restarted.
Requirement already satisfied: seaborn in /local_disk0/.ephemeral_nfs/envs/pythonEnv-3f4488e5-cf6e-4a18-8f54-937ead82fcf0/lib/python3.8/site-packages (0.12.1)
Requirement already satisfied: pandas>=0.25 in /local_disk0/.ephemeral_nfs/envs/pythonEnv-3f4488e5-cf6e-4a18-8f54-937ead82fcf0/lib/python3.8/site-packages (from seaborn) (1.5.2)
Requirement already satisfied: matplotlib>=3.6.1,>=3.1 in /databricks/python3/lib/python3.8/site-packages (from seaborn) (3.4.2)
Requirement already satisfied: numpy>=1.17 in /local_disk0/.ephemeral_nfs/envs/pythonEnv-3f4488e5-cf6e-4a18-8f54-937ead82fcf0/lib/python3.8/site-packages (from seaborn) (1.22.4)
Requirement already satisfied: python-dateutil>=2.7 in /databricks/python3/lib/python3.8/site-packages (from matplotlib>=3.6.1,>=3.1->seaborn) (2.8.1)
Requirement already satisfied: pillow>=6.2.0 in /local_disk0/.ephemeral_nfs/envs/pythonEnv-3f4488e5-cf6e-4a18-8f54-937ead82fcf0/lib/python3.8/site-packages (from matplotlib>=3.6.1,>=3.1->seaborn) (9.3.8)
Requirement already satisfied: cycler>=0.10 in /databricks/python3/lib/python3.8/site-packages (from matplotlib>=3.6.1,>=3.1->seaborn) (0.10.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /databricks/python3/lib/python3.8/site-packages (from matplotlib>=3.6.1,>=3.1->seaborn) (1.3.1)
Requirement already satisfied: pyparsing>=2.2.1 in /databricks/python3/lib/python3.8/site-packages (from matplotlib>=3.6.1,>=3.1->seaborn) (2.4.7)
Requirement already satisfied: six in /databricks/python3/lib/python3.8/site-packages (from cycler>=0.10->matplotlib>=3.6.1,>=3.1->seaborn) (1.15.0)
Requirement already satisfied: uvloop>=2020.1 in /databricks/python3/lib/python3.8/site-packages (from pandas>=0.25->seaborn) (2020.5.4)
Command took 19.59 seconds -- by n.sharmackh@dp.omicrossoft.com at 14/12/2022, 15:34:03 on DE_Generator

Cmd 2
import pandas as pd
```

Figure 18: ev_regression_model_monthly_data

4.8 ev_spatial_model.ipynb

The main purpose of this notebook is to take the final output of the model and run the spatial model. The output of this notebook is a JPEG file that contains the locations where to place Electric Vehicle charging stations as per the energy utilization ranking. The notebook is shown in Figure 19.

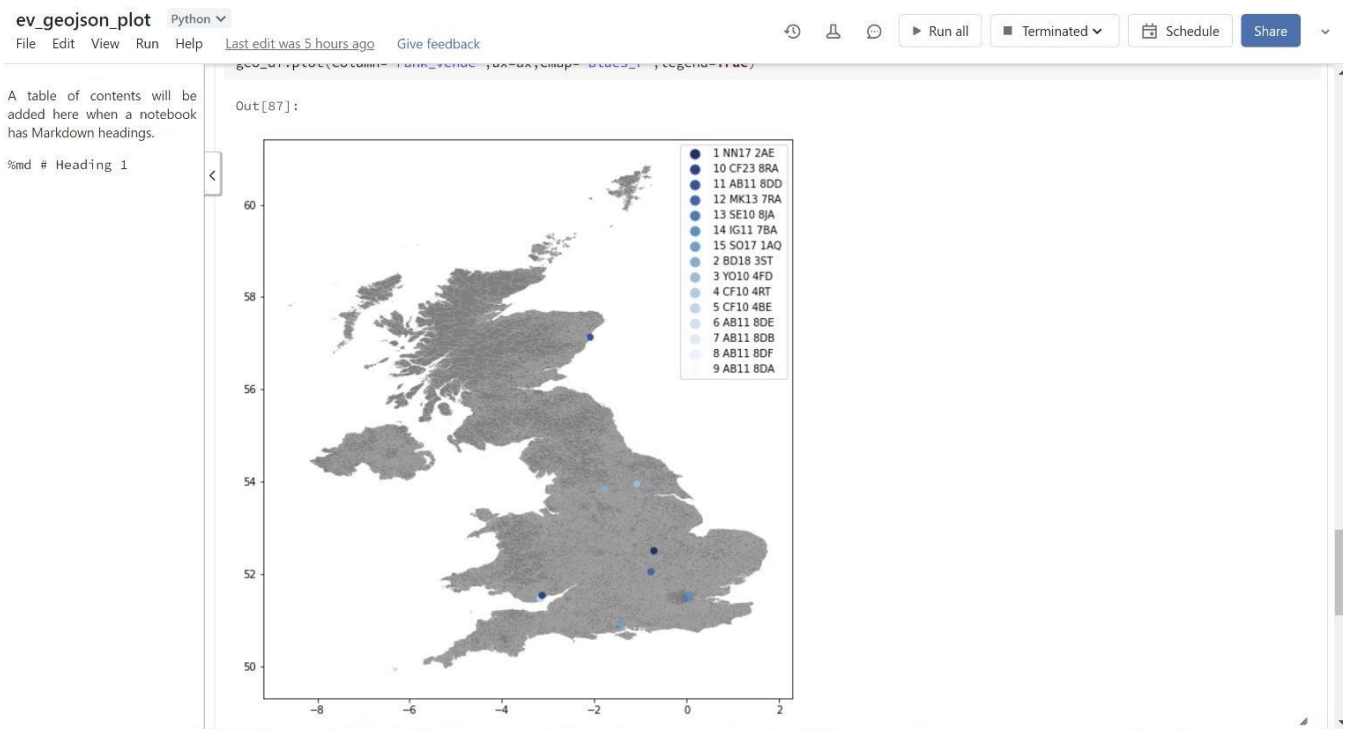


Figure 19: ev spatial model

5 Pipeline Setup

The pipeline of notebooks is set up in the Databricks workflow manager as shown in Figure 20. The pipeline contains all the notebooks in the below order:

- `ev_static_csv_file_load.ipynb`
- `ev_model_dataset_creation.ipynb`
- `ev_create_dataset_traffic_carownership.ipynb`
- `ev_unsupervised_model_monthly.ipynb`
- `ev_regression_model_monthly_data.ipynb`
- `ev_spatial_model.ipynb`

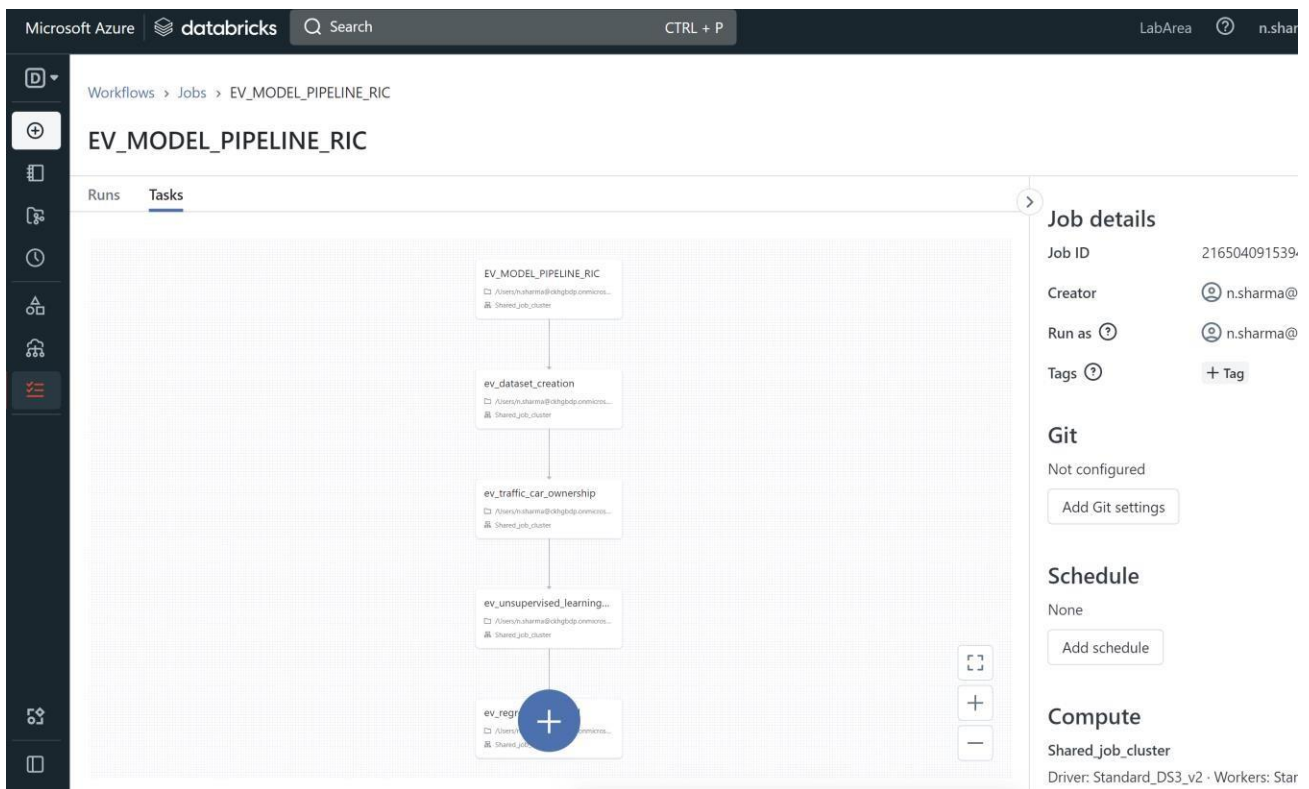


Figure 20: Databricks Workflow

6 Resources URL

Artefact:

https://studentncirl-my.sharepoint.com/:f:/g/personal/x21167818_student_ncirl_ie/EqSYdxrs8rBDjjO7278KkT8BSAsbnFseK5UclUafd4LTUA

References

Trafficcount-data.gov.uk(2022), <https://www.data.gov.uk/dataset/208c0e7b-353f-4e2d-8b8b-8b8b8b8b8b8b> [Accessed 24-Nov-2022].

Transport, D. f. (2022), ‘Vehicles statistics’.

URL: <https://www.gov.uk/government/collections/vehicles-statistics>

Uk Mobility Footfall (2022).

URL: <https://my.api.mockaroo.com/evfootfall.json?key=9eb8e7e0>

Uk Mobility passerby (2022).

URL: <https://api.mockaroo.com/api/67042b30?count=1000key=9eb8e7e0>

uk-postcode-electricity-consumption (2022), <https://www.data.gov.uk/dataset/e7d4c1cf-45a0-4070-878f-24ad9641f655/domestic-electricity-and-gas-estimates-by-postcode-in-great-britain>. [Accessed 24-Nov-2022].

ukpostcodegeojson(2022), <https://geoportal.statistics.gov.uk/>. [Accessed 24-Nov-2022].

ukpostcode-NSPL(2022), <https://geoportal.statistics.gov.uk/datasets/national-statistics> [Accessed 24-Nov-2022].