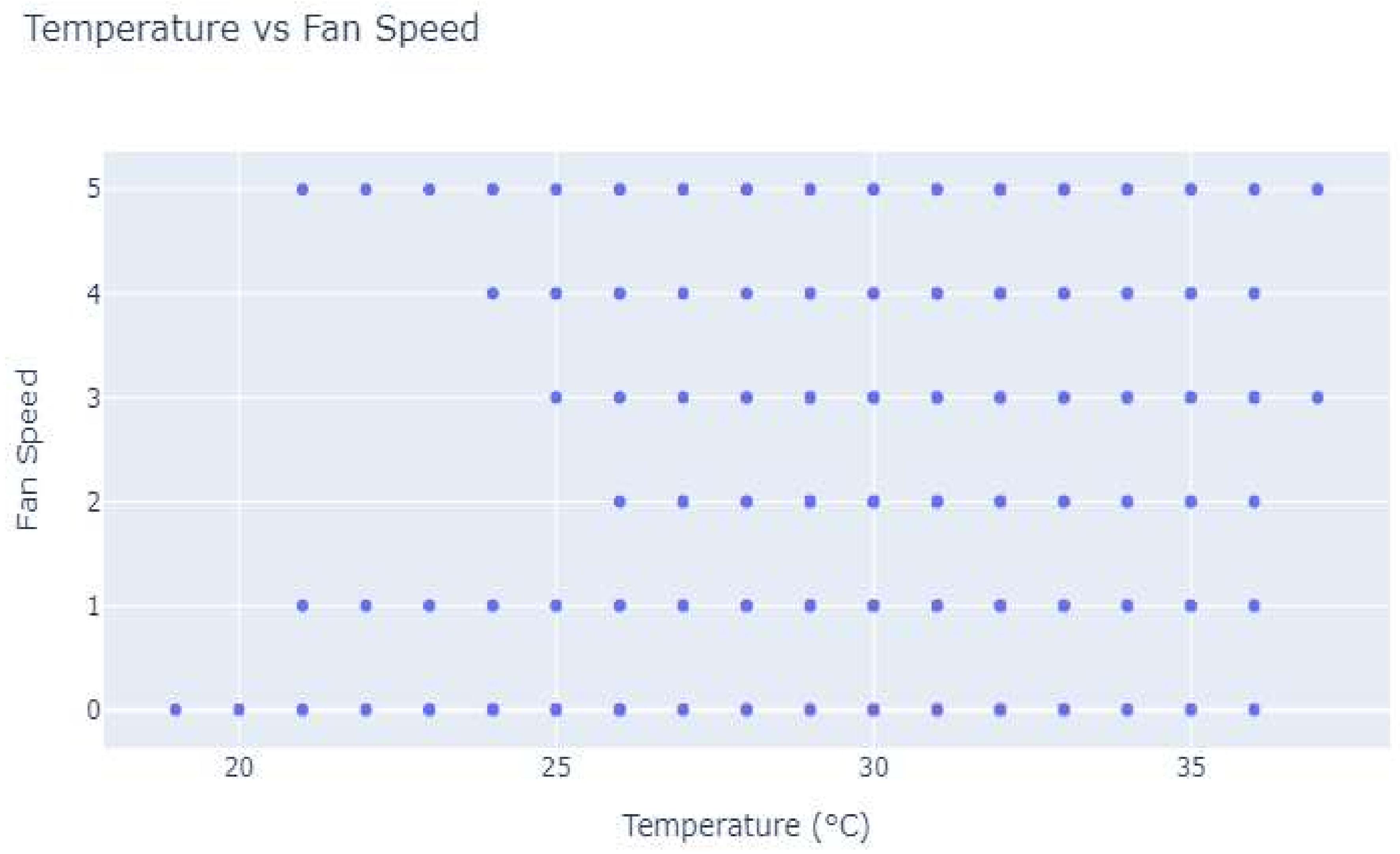
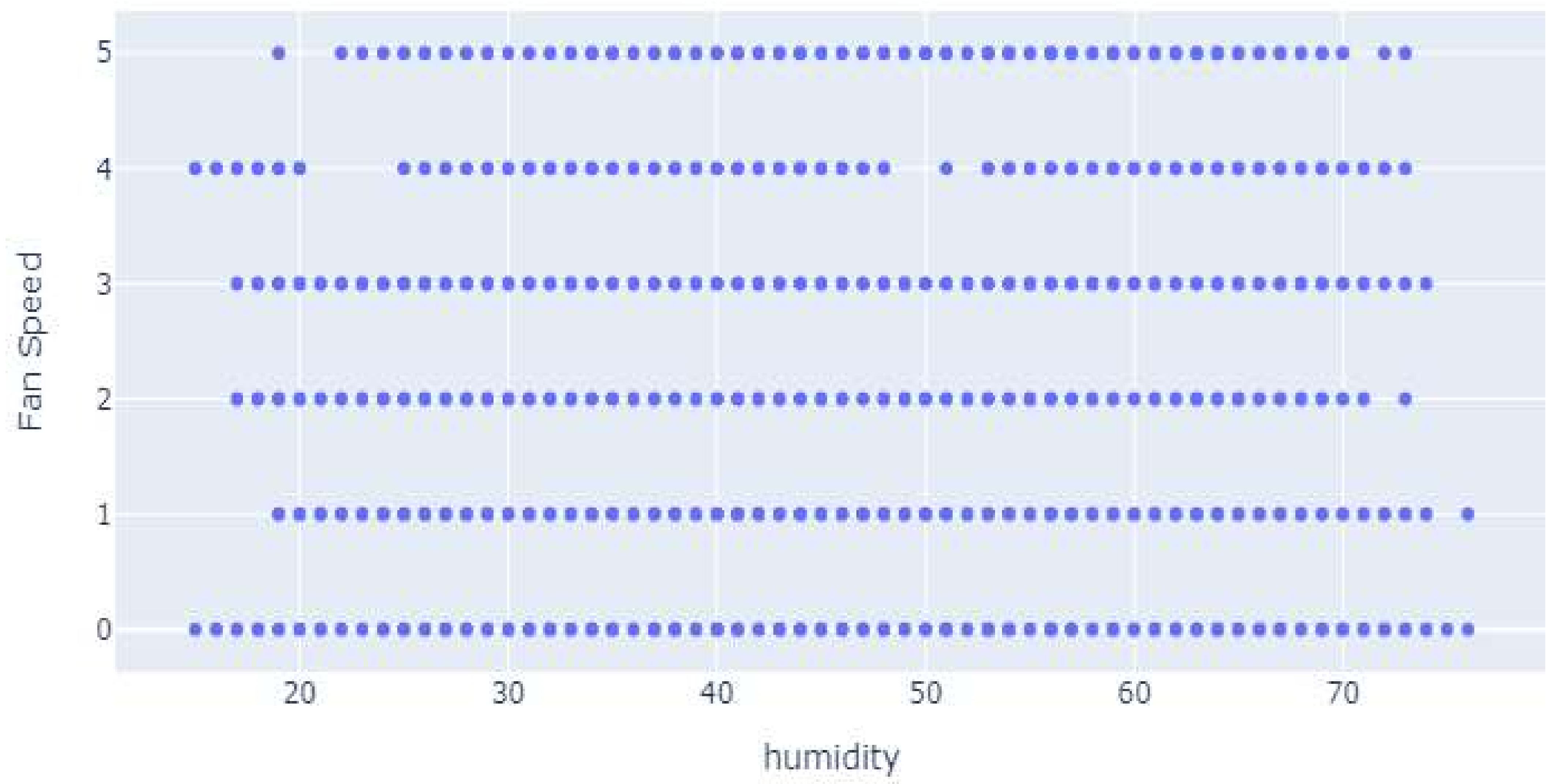
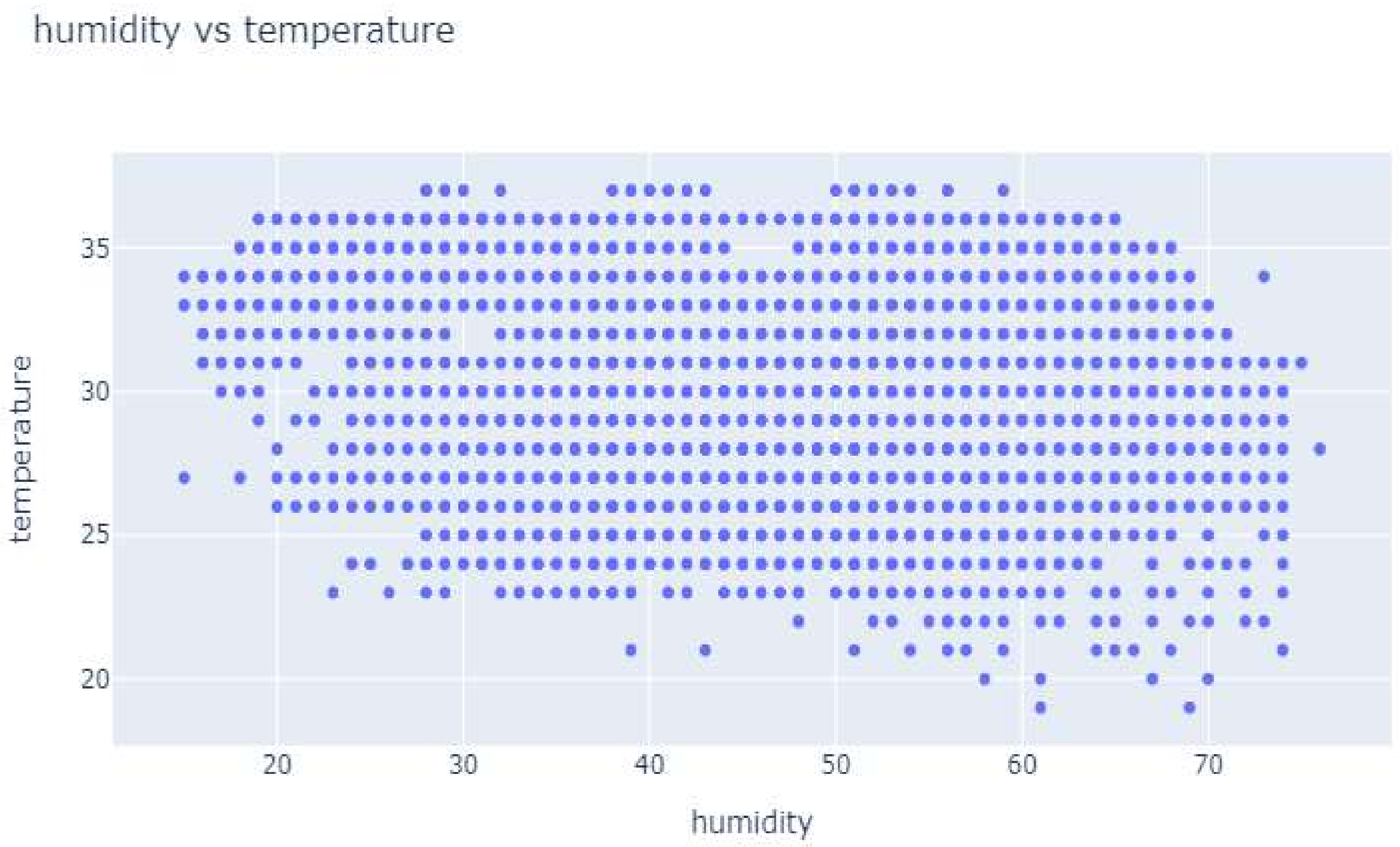
BLDC FAN Data Analysis

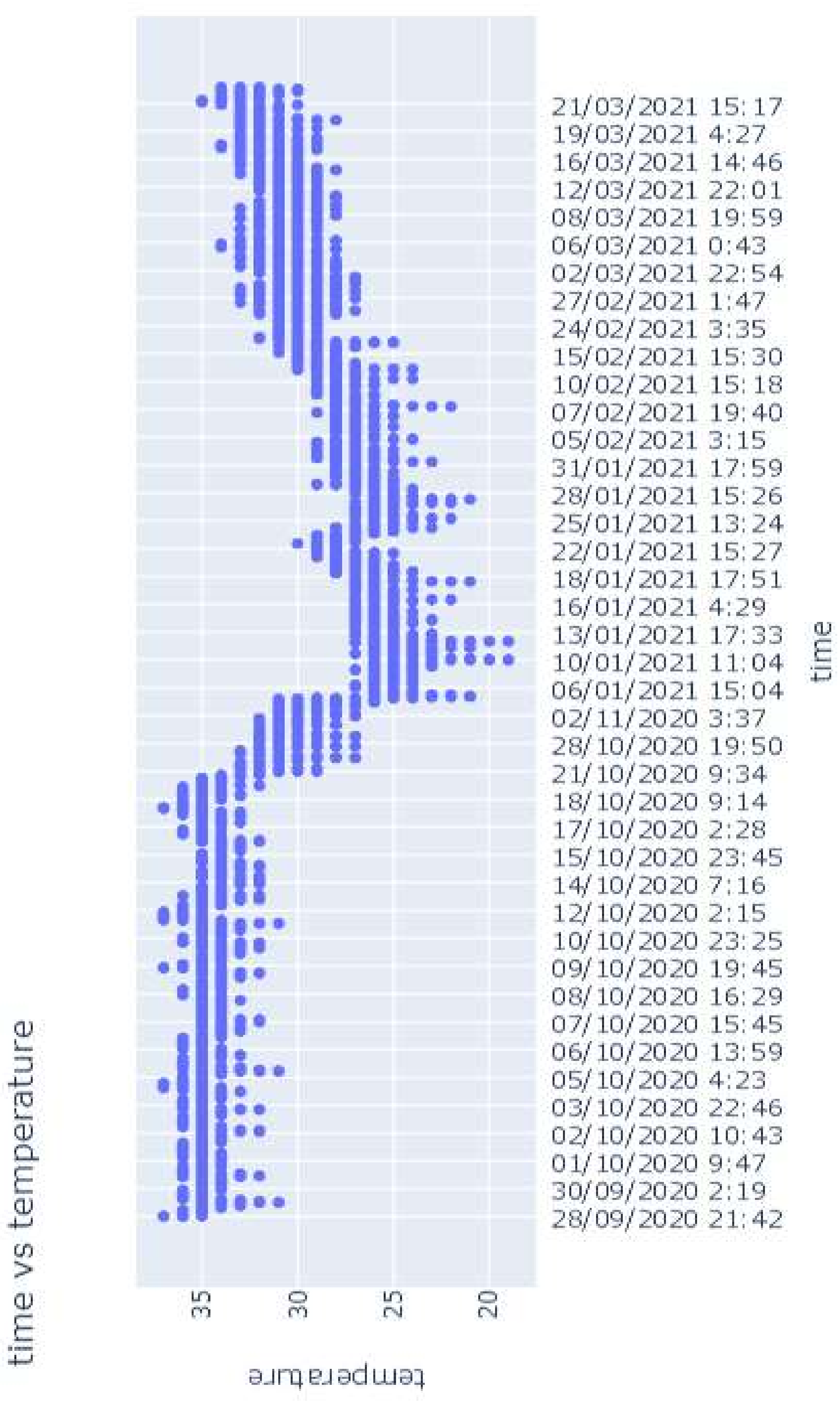
S.No	datetime	temperature	humidity	mode	speed	opTime	eSpent	eSaved
0	28/09/2020 21:42	35	53	0	5	30	1440	1440
1	28/09/2020 21:43	35	52	0	5	30	1440	1440
2	28/09/2020 21:47	34	35	0	5	30	1440	1440
3	28/09/2020 21:53	36	45	0	5	60	2880	2880
4	28/09/2020 21:54	37	49	0	5	180	8640	8640
5	28/09/2020 21:59	34	51	0	5	60	2880	2880
6	28/09/2020 22:05	36	57	0	5	60	2880	2880

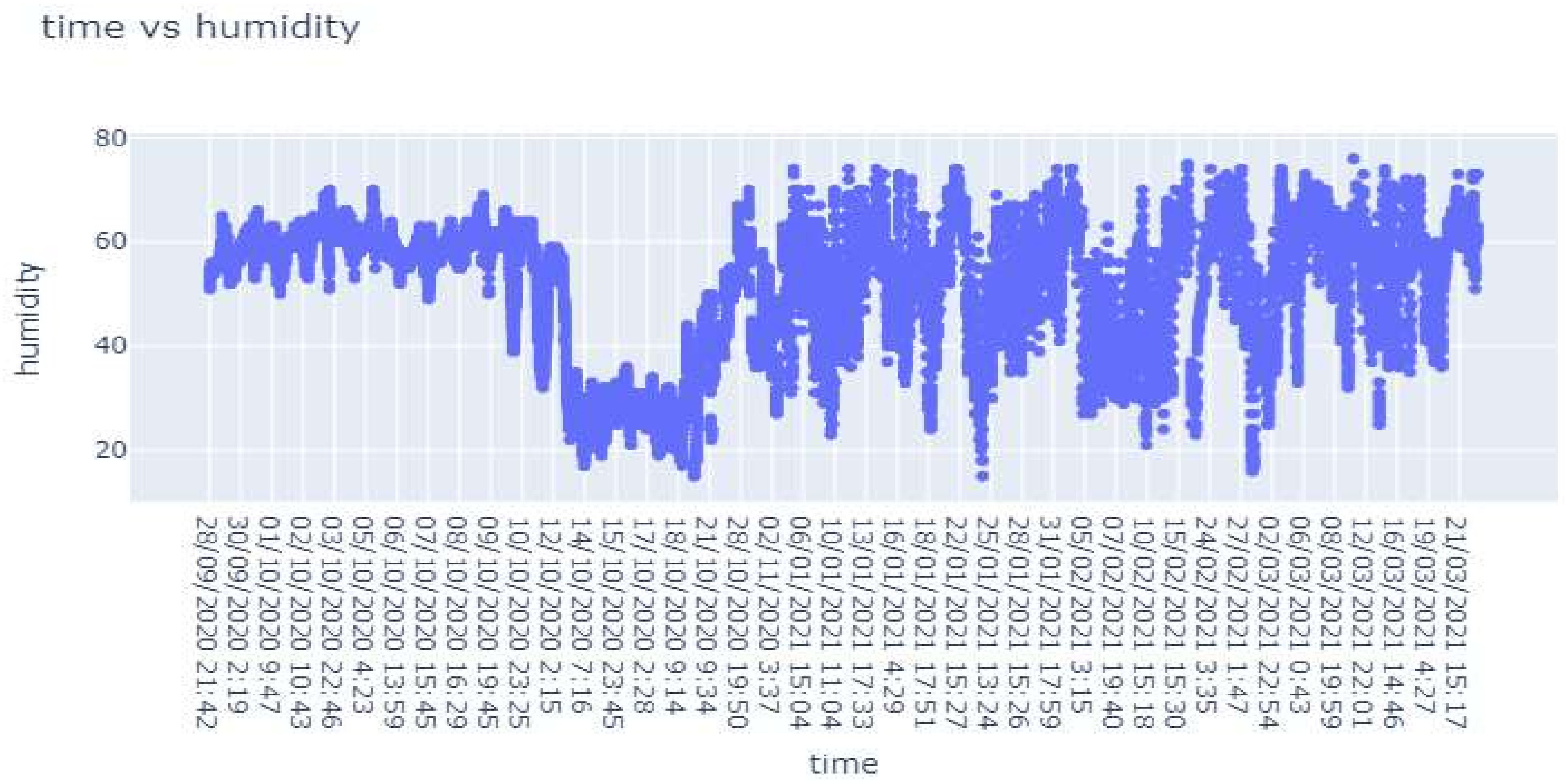

Temperature vs Fan Speed
Fan Speed
5
4
3
2
1
0
20
25
30
35
Temperature (°C)

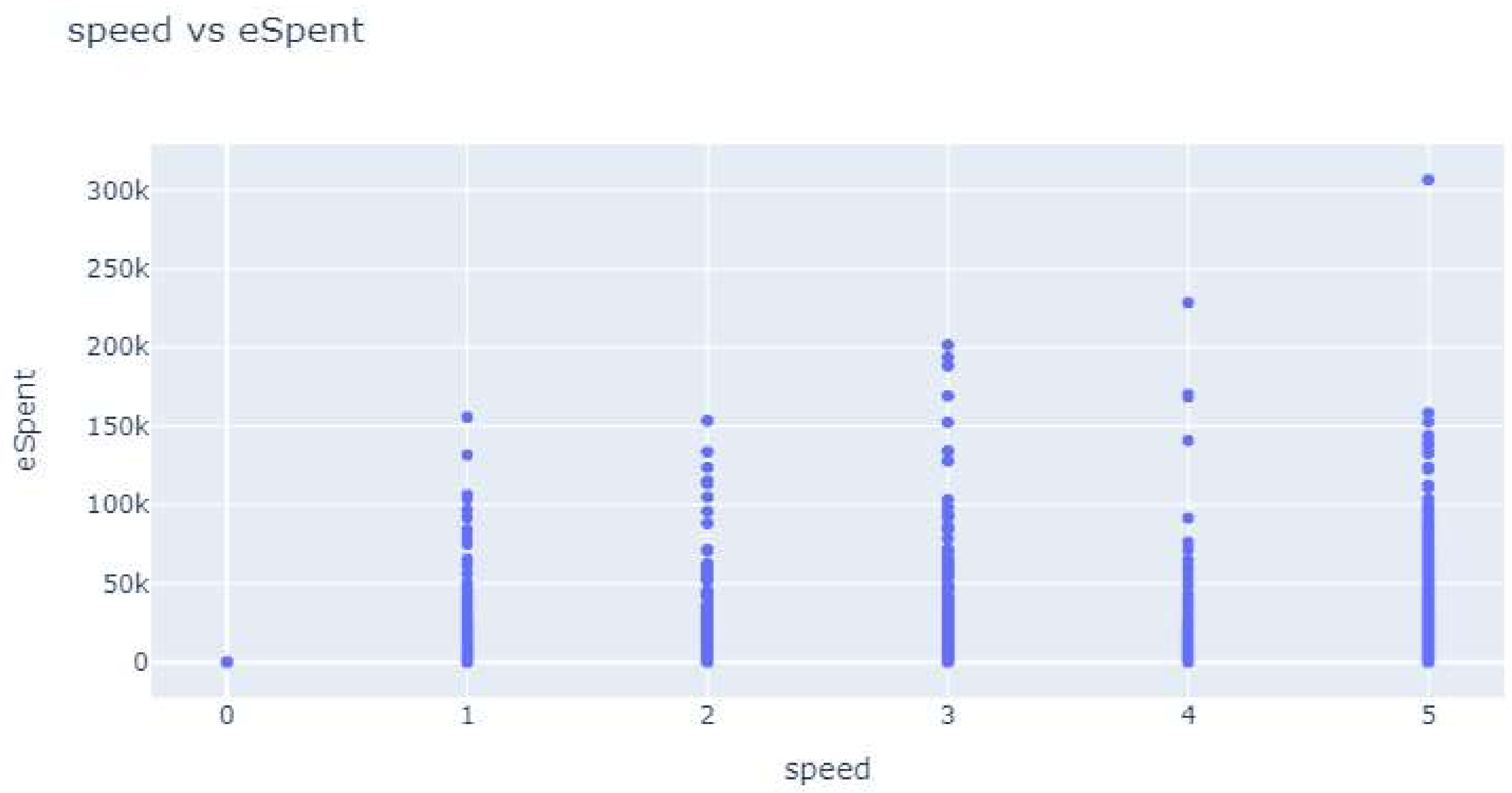
humidity vs Fan Speed

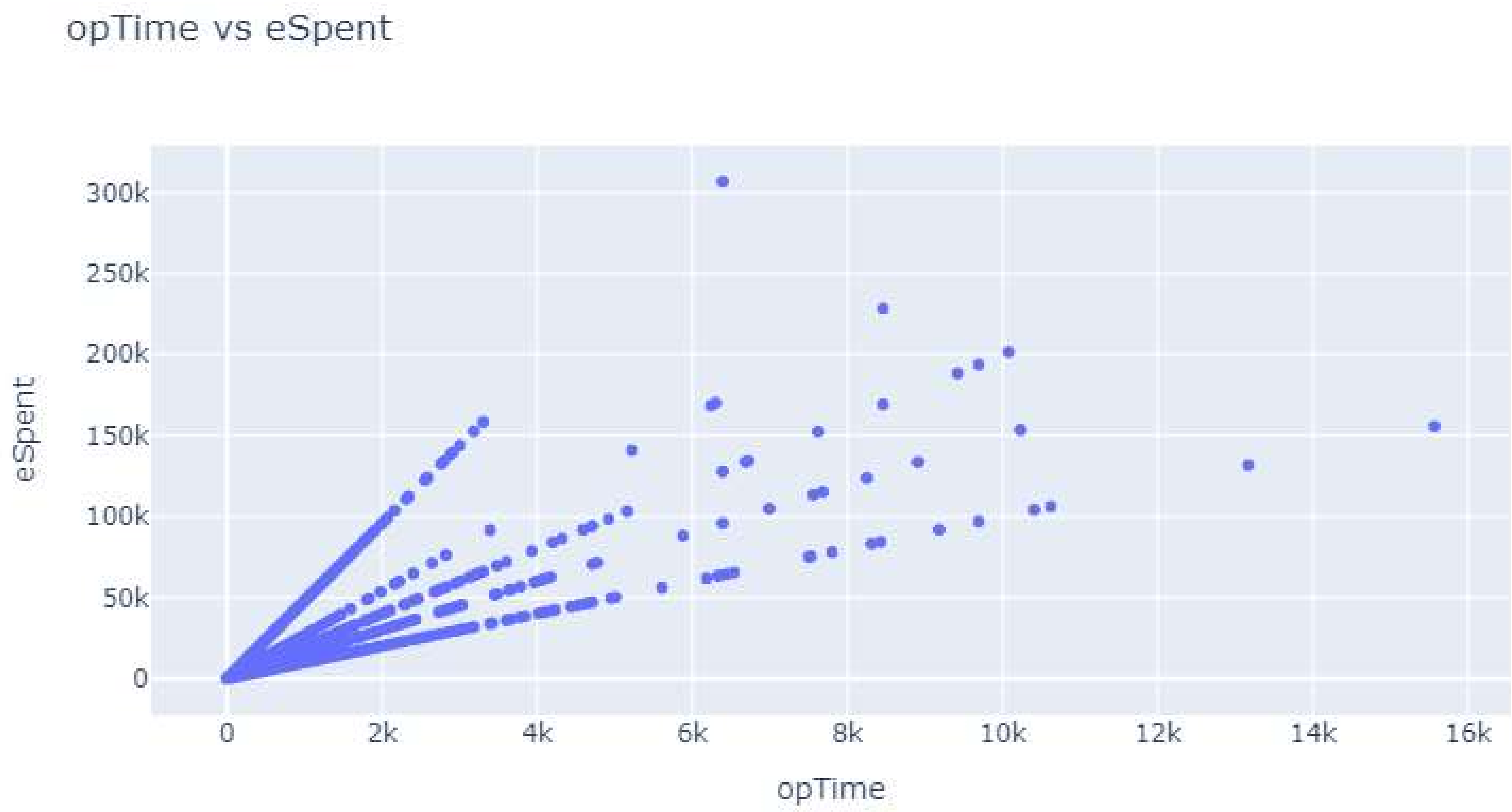


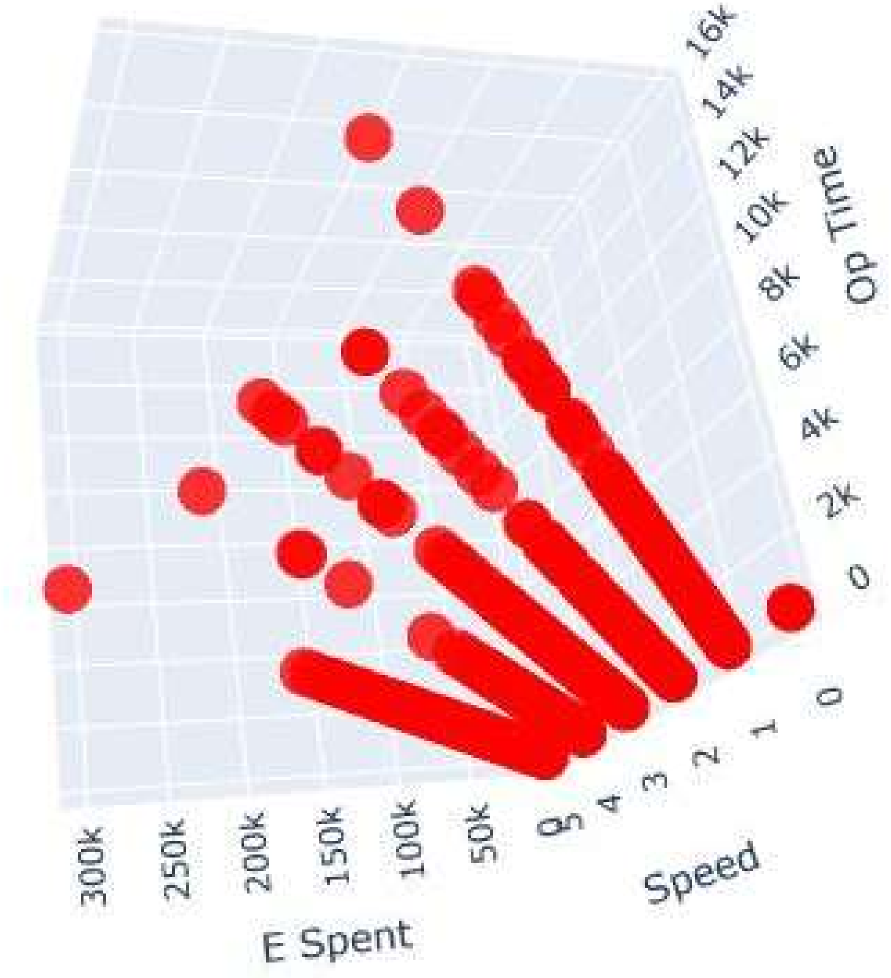

humidity vs temperature
temperature
35
30
25
20
20
30
40
50
60
70
humidity


time vs temperature
temperature
35
30
25
20
time
21/03/2021 15:17
19/03/2021 4:27
16/03/2021 14:46
12/03/2021 22:01
08/03/2021 19:59
06/03/2021 0:43
02/03/2021 22:54
27/02/2021 1:47
24/02/2021 3:35
15/02/2021 15:30
10/02/2021 15:18
07/02/2021 19:40
05/02/2021 3:15
31/01/2021 17:59
28/01/2021 15:26
25/01/2021 13:24
22/01/2021 15:27
18/01/2021 17:51
16/01/2021 4:29
13/01/2021 17:33
10/01/2021 11:04
06/01/2021 15:04
02/11/2020 3:37
28/10/2020 19:50
21/10/2020 9:34
18/10/2020 9:14
17/10/2020 2:28
15/10/2020 23:45
14/10/2020 7:16
12/10/2020 2:15
10/10/2020 23:25
09/10/2020 19:45
08/10/2020 16:29
07/10/2020 15:45
06/10/2020 13:59
05/10/2020 4:23
03/10/2020 22:46
02/10/2020 10:43
01/10/2020 9:47
30/09/2020 2:19
28/09/2020 21:42

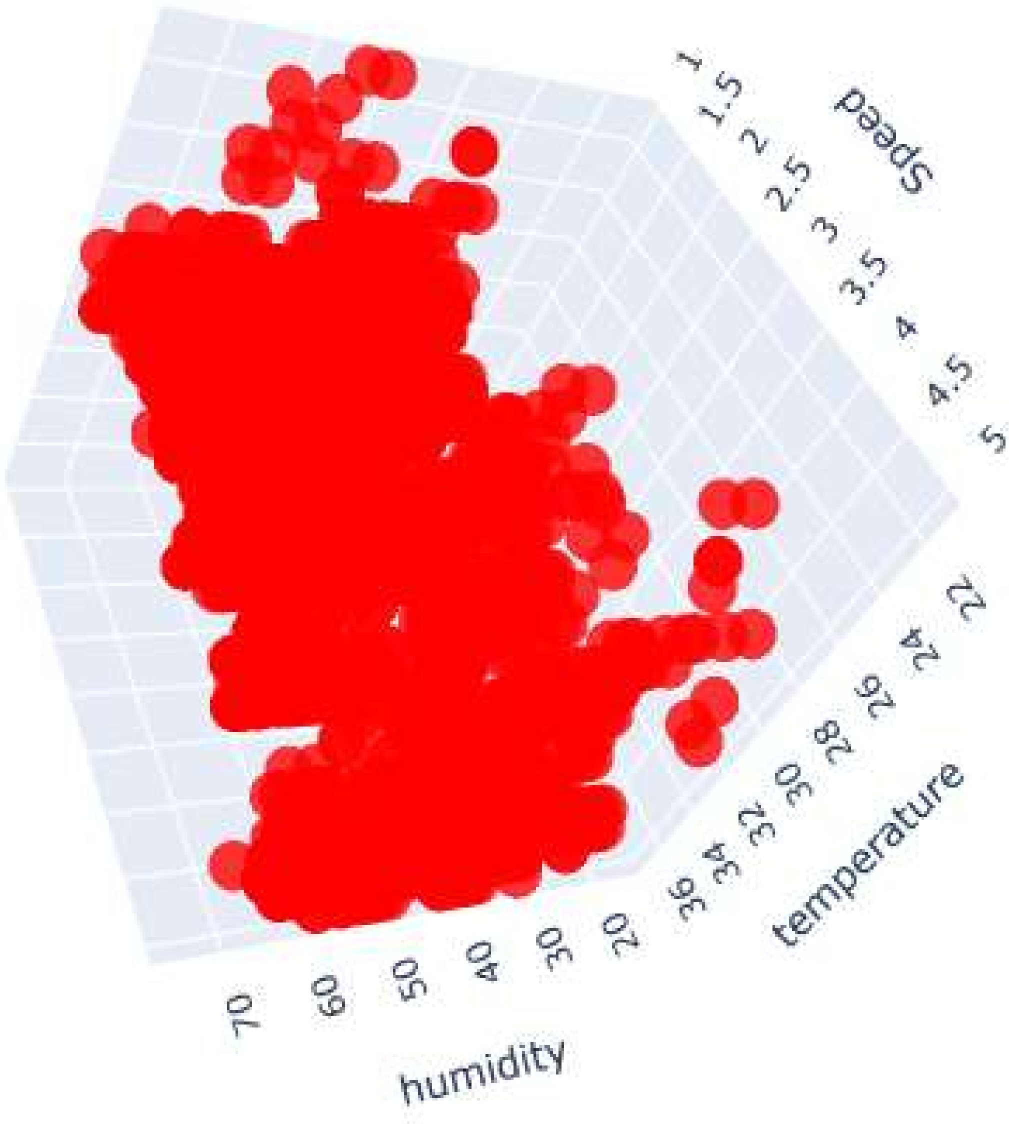

time vs humidity
humidity
80
60
40
20
time
28/09/2020 21:42
30/09/2020 2:19
01/10/2020 9:47
02/10/2020 10:43
03/10/2020 22:46
05/10/2020 4:23
06/10/2020 13:59
07/10/2020 15:45
08/10/2020 16:29
09/10/2020 19:45
10/10/2020 23:25
12/10/2020 2:15
14/10/2020 7:16
15/10/2020 23:45
17/10/2020 2:28
18/10/2020 9:14
21/10/2020 9:34
28/10/2020 19:50
02/11/2020 3:37
06/01/2021 15:04
10/01/2021 11:04
13/01/2021 17:33
16/01/2021 4:29
18/01/2021 17:51
22/01/2021 15:27
25/01/2021 13:24
28/01/2021 15:26
31/01/2021 17:59
05/02/2021 3:15
07/02/2021 19:40
10/02/2021 15:18
15/02/2021 15:30
24/02/2021 3:35
27/02/2021 1:47
02/03/2021 22:54
06/03/2021 0:43
08/03/2021 19:59
12/03/2021 22:01
16/03/2021 14:46
19/03/2021 4:27
21/03/2021 15:17


speed vs eSpent
eSpent
300k
250k
200k
150k
100k
50k
0
0
1
2
3
4
5
speed


opTime vs eSpent
eSpent
300k
250k
200k
150k
100k
50k
0
0
2k
4k
6k
8k
10k
12k
14k
16k
opTime

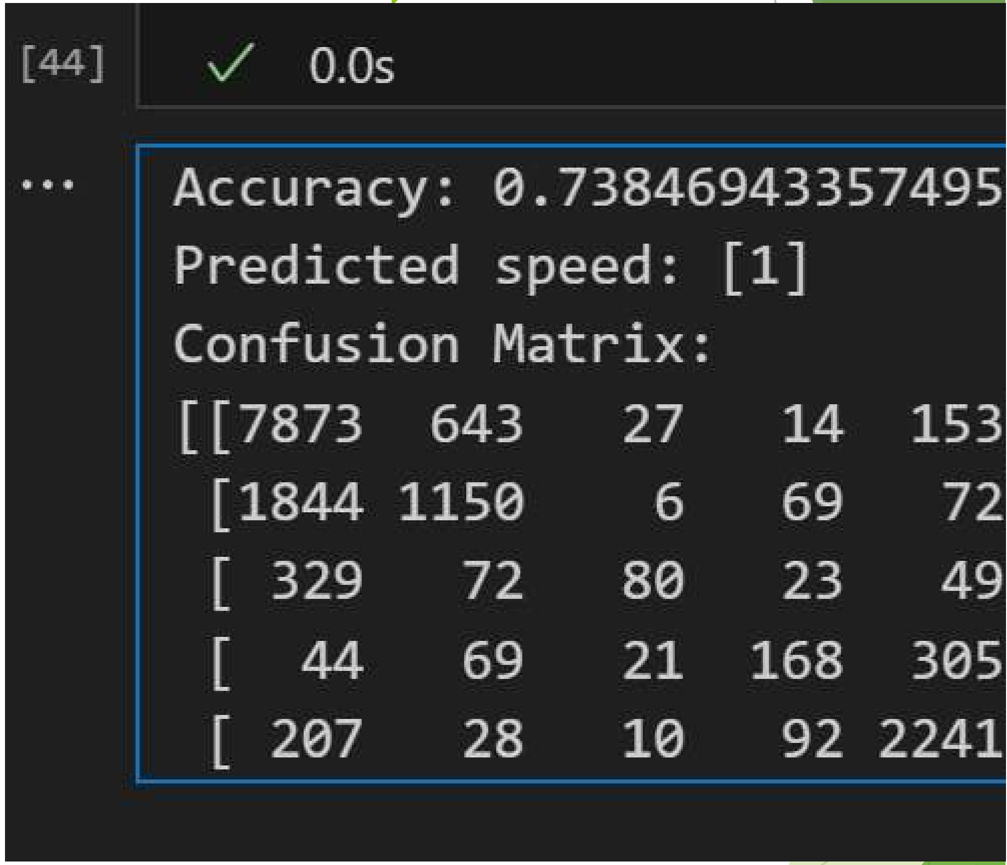


E Spent
300k
250k
200k
150k
100k
50k
0
Speed
5
4
3
2
1
0
Op Time
16k
14k
12k
10k
8k
6k
4k
2k
0


Speed
1
1.5
2
2.5
3
3.5
4
4.5
5
temperature
22
24
26
28
30
32
34
36
humidity
20
30
40
50
60
70

SPEED CLASSIFIER (DECISION TREE)

- TEMPERATURE HUMIDITY
- DECISION TREE CLASSIFIER
-
- SPEED
- (five classes 1-5)

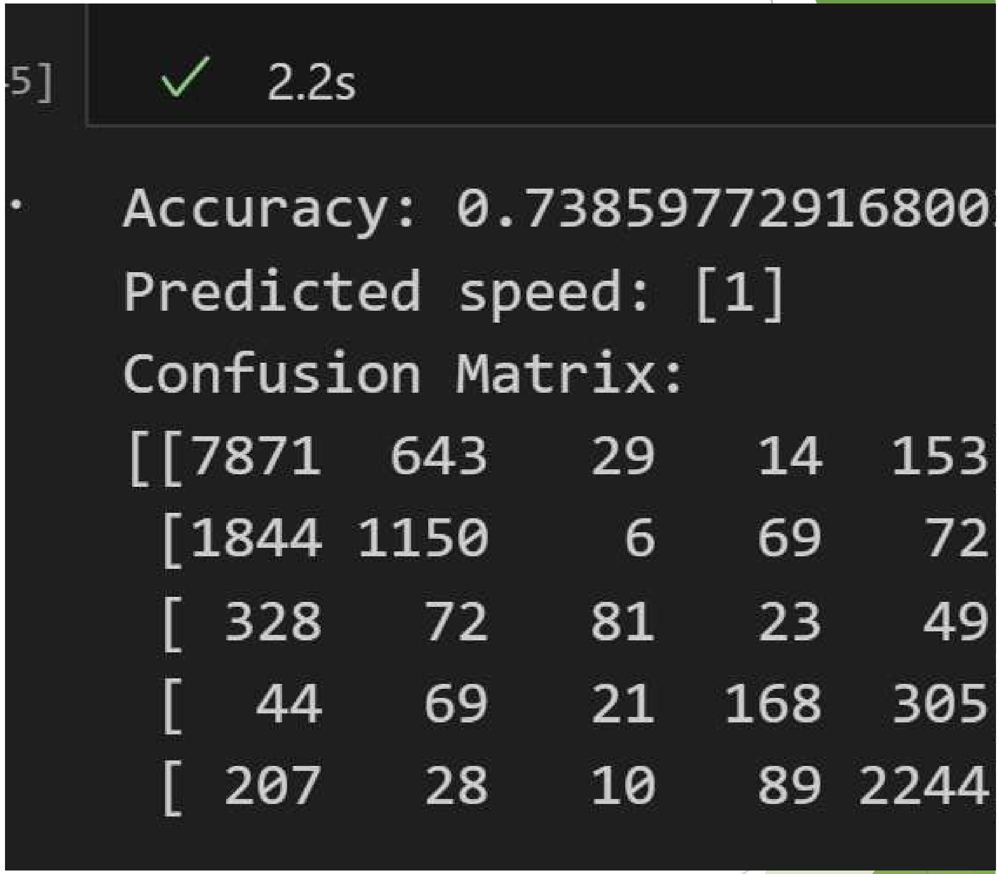


```
[44]  ✓  0.0s

...   Accuracy: 0.73846943357495
      Predicted speed: [1]
      Confusion Matrix:
      [[7873  643   27   14  153
       [1844 1150    6   69   72
       [ 329   72   80   23   49
       [  44   69   21  168  305
       [ 207   28   10   92 2241
```

SPEED CLASSIFIER (RANDOM FOREST)

- TEMPERATURE HUMIDITY
- RANDOM FOREST CLASSIFIER
-
- SPEED
- (five classes 1-5)

- RANDOM FOREST IS BEST AMOUNG DECISION TREE
- AND KNEIGHBOR.



```
5]    ✓  2.2s

.   Accuracy: 0.738597729168003
    Predicted speed: [1]
    Confusion Matrix:
    [[7871  643   29   14  153
     [1844 1150    6   69   72
     [ 328   72   81   23   49
     [  44   69   21  168  305
     [ 207   28   10   89 2244
```

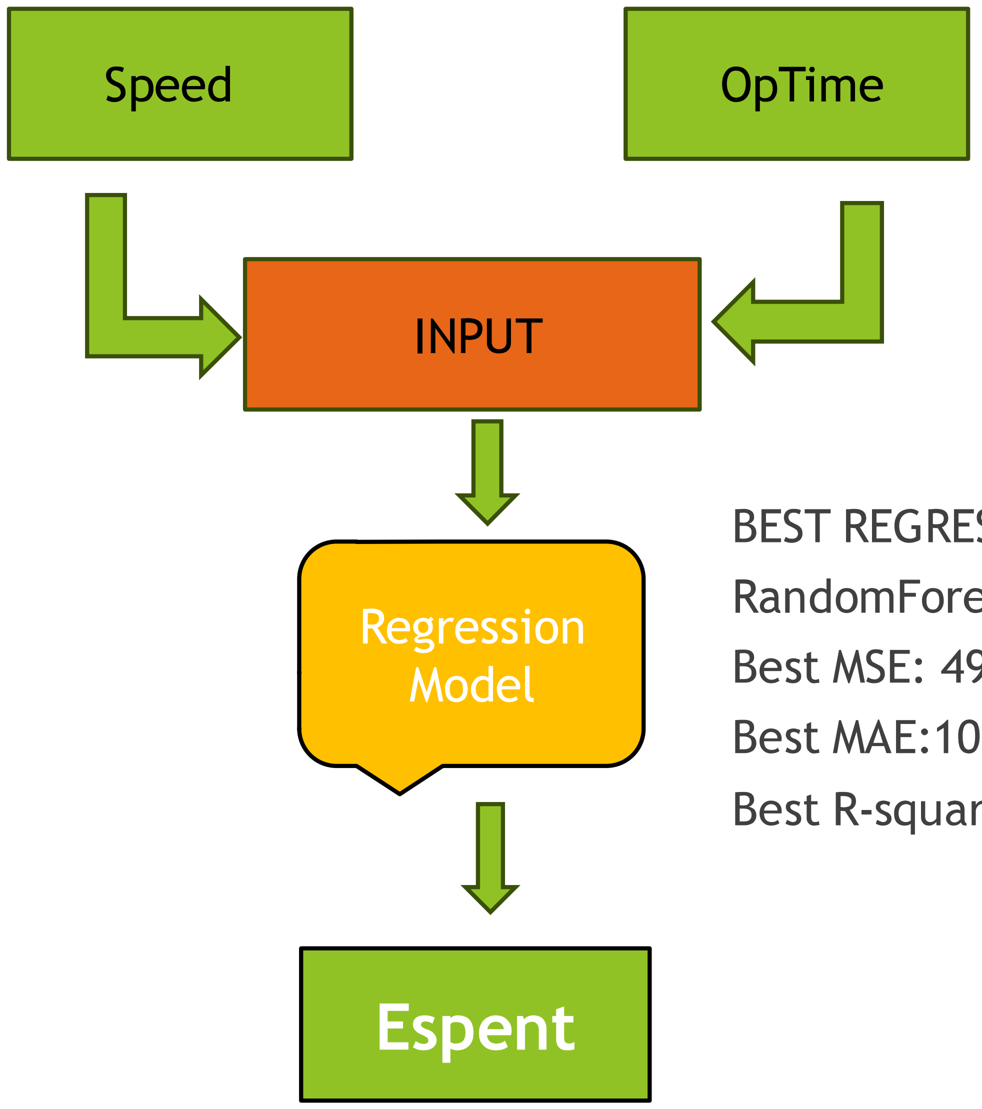
OPTIME PREDICTOR REGRESSION MODEL:

- AFTER LOOPING THROUGH LinearRegression ,DecisionTreeRegressor
- ,RandomForestRegressor:

- Best regressor: RandomForestRegressor
- Best MSE: 163.33234519212255

  Best MAE: 0.5547616909359159
- Best R-squared: 0.9997020512203401

SK learn Regression Model



BEST REGRESSOR FOR eSPENT

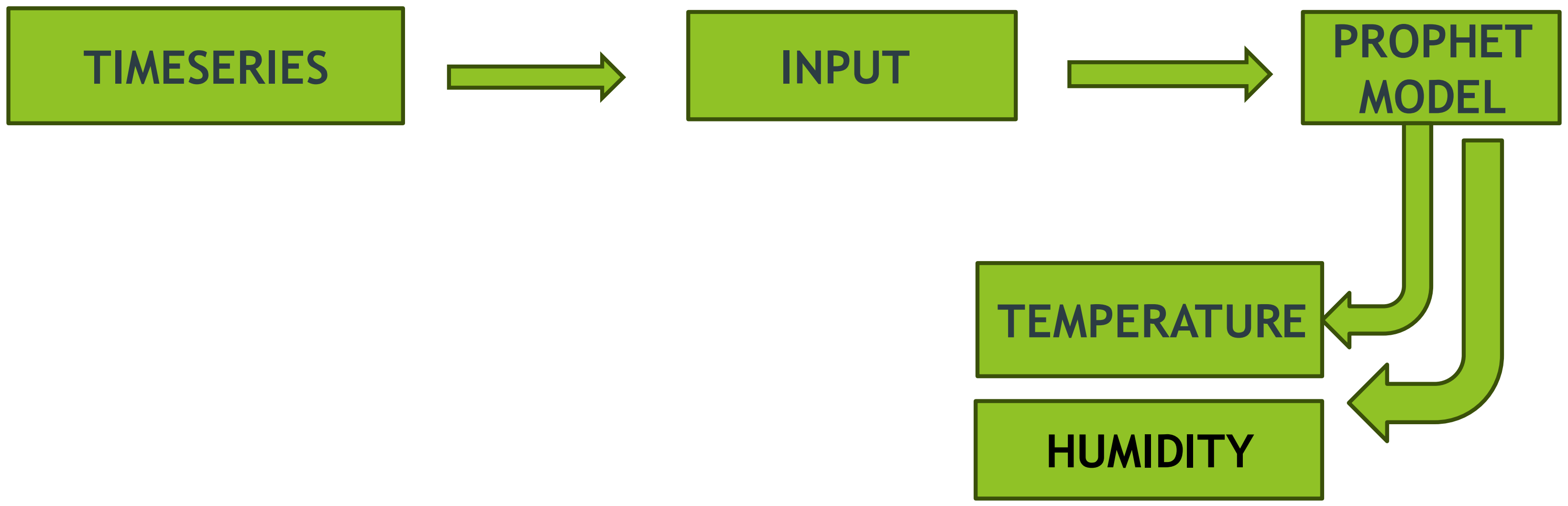
RandomForestRegressor:

Best MSE: 49178.90099762005

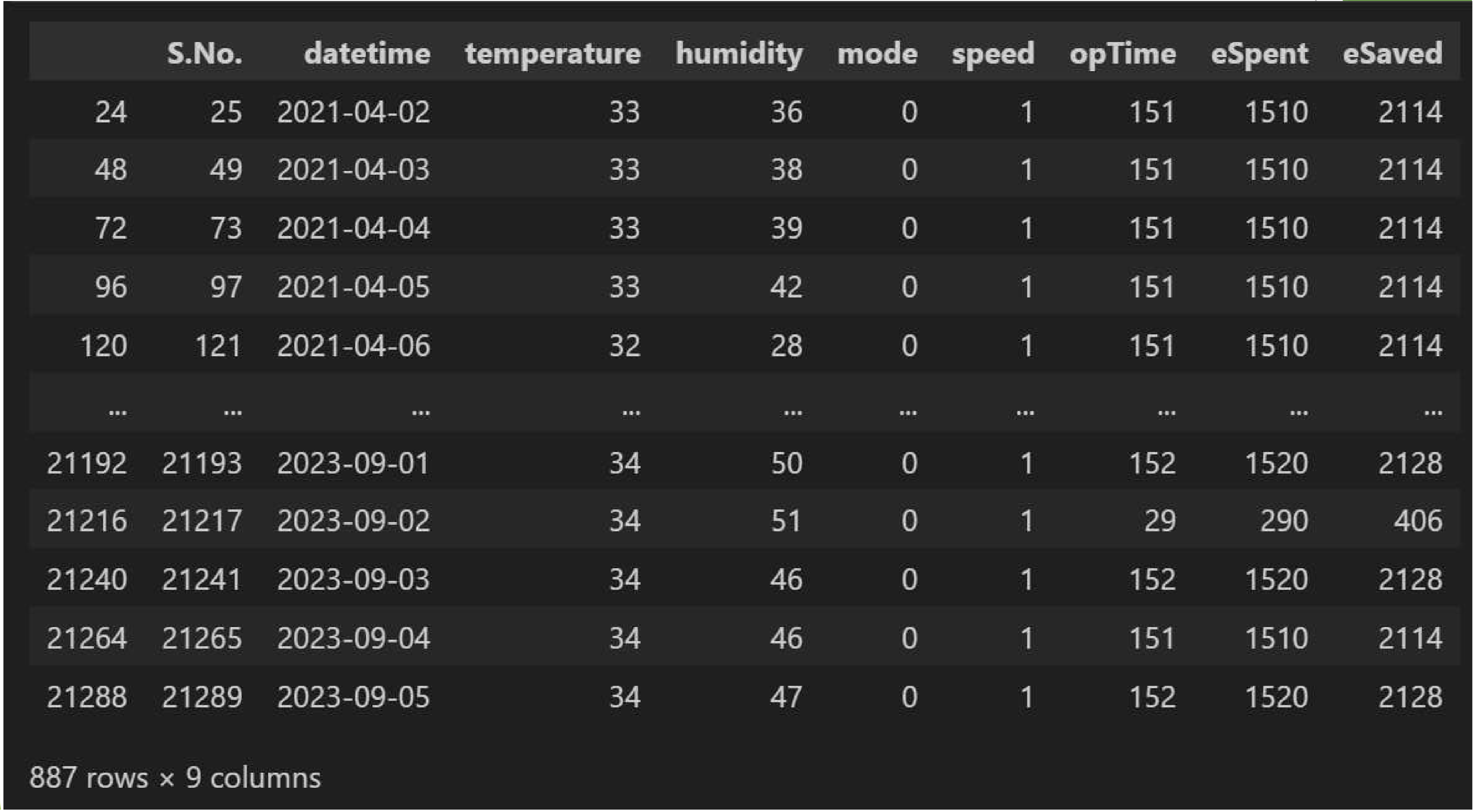
Best MAE:10.057831162999555

Best R-squared: 0.9998327648503513

FACEBOOK PROPHET MODEL TIME SERIES ANALYSYS



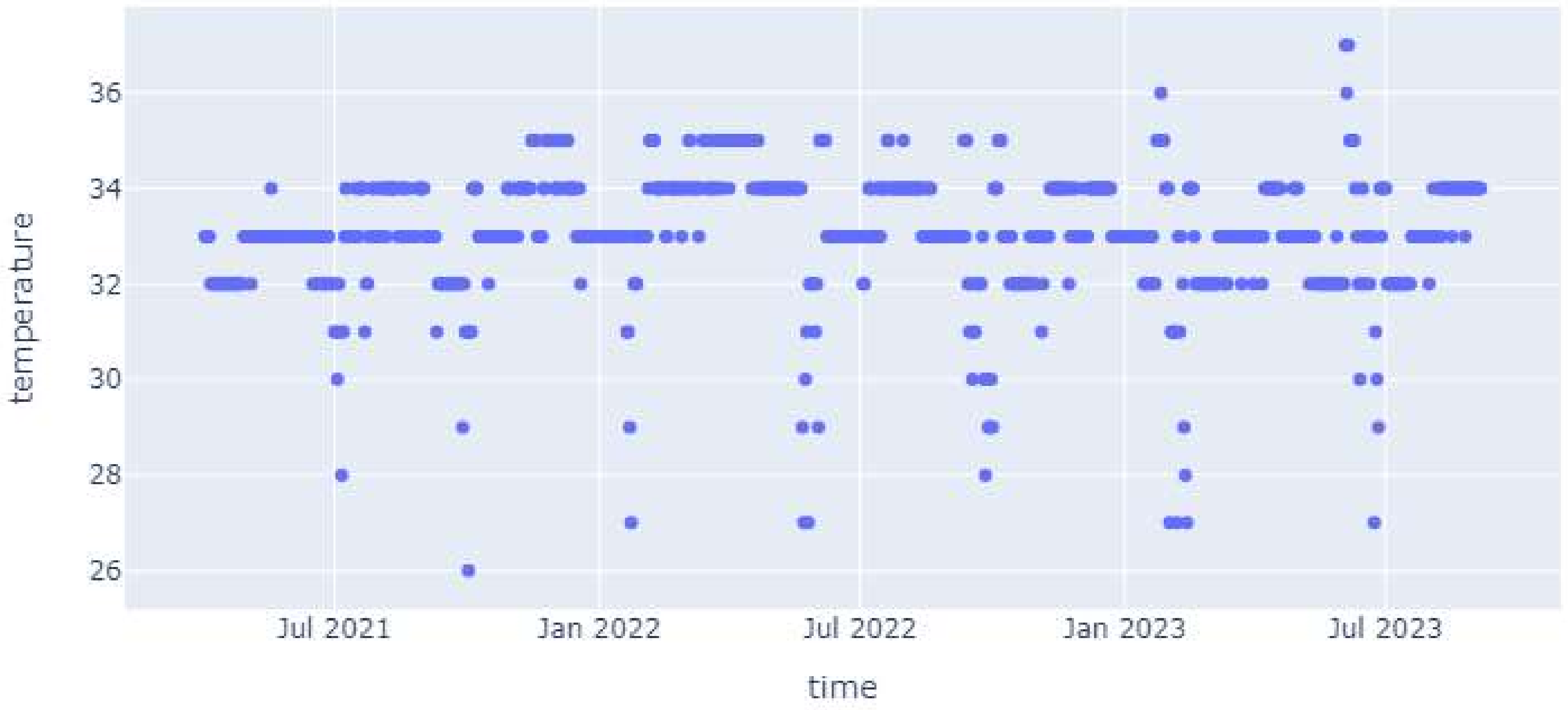
Filtered Data for time series analysis:

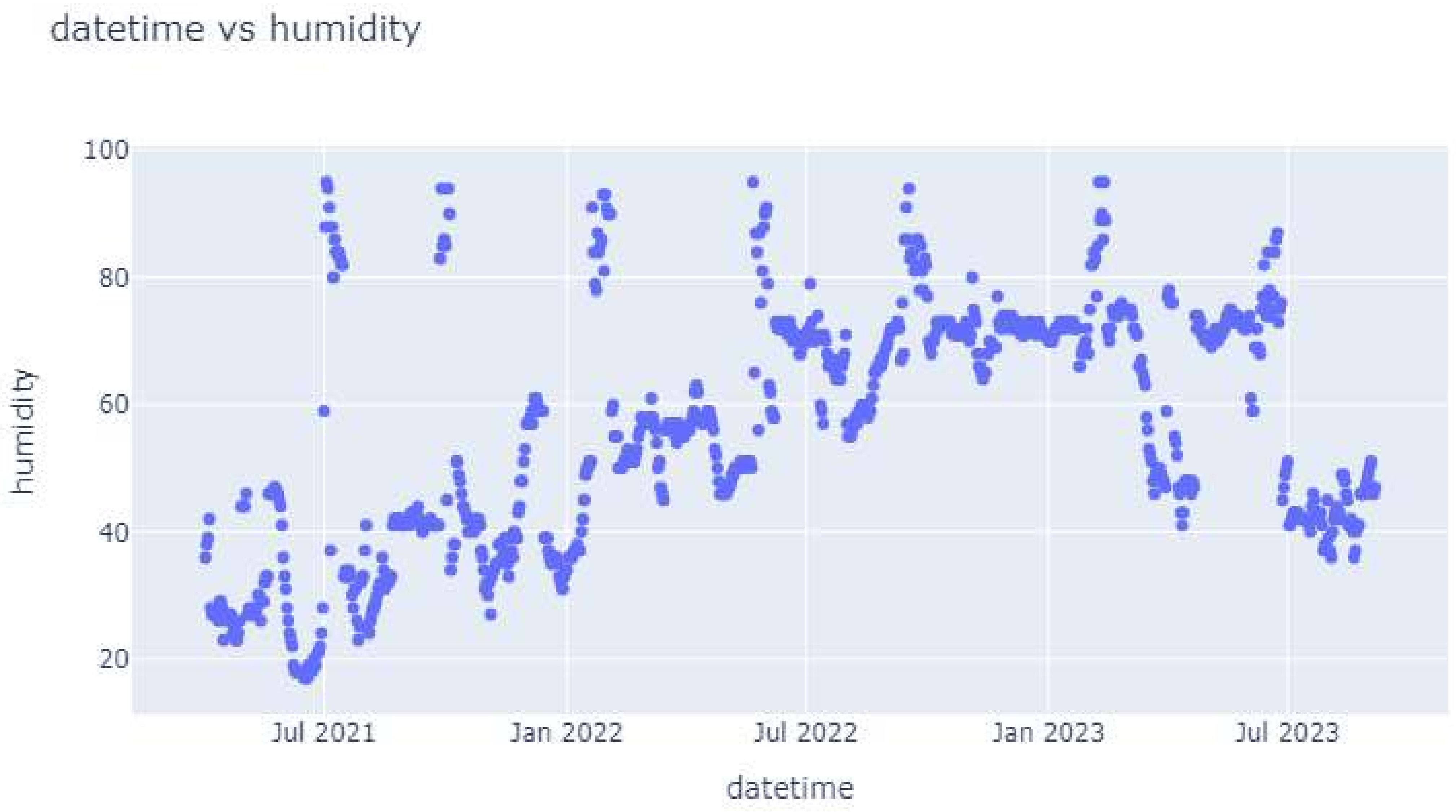


	S.No.	datetime	temperature	humidity	mode	speed	opTime	eSpent	eSaved
24	25	2021-04-02	33	36	0	1	151	1510	2114
48	49	2021-04-03	33	38	0	1	151	1510	2114
72	73	2021-04-04	33	39	0	1	151	1510	2114
96	97	2021-04-05	33	42	0	1	151	1510	2114
120	121	2021-04-06	32	28	0	1	151	1510	2114
...	...	...	...	...	...	...	...	...	...
21192	21193	2023-09-01	34	50	0	1	152	1520	2128
21216	21217	2023-09-02	34	51	0	1	29	290	406
21240	21241	2023-09-03	34	46	0	1	152	1520	2128
21264	21265	2023-09-04	34	46	0	1	151	1510	2114
21288	21289	2023-09-05	34	47	0	1	152	1520	2128

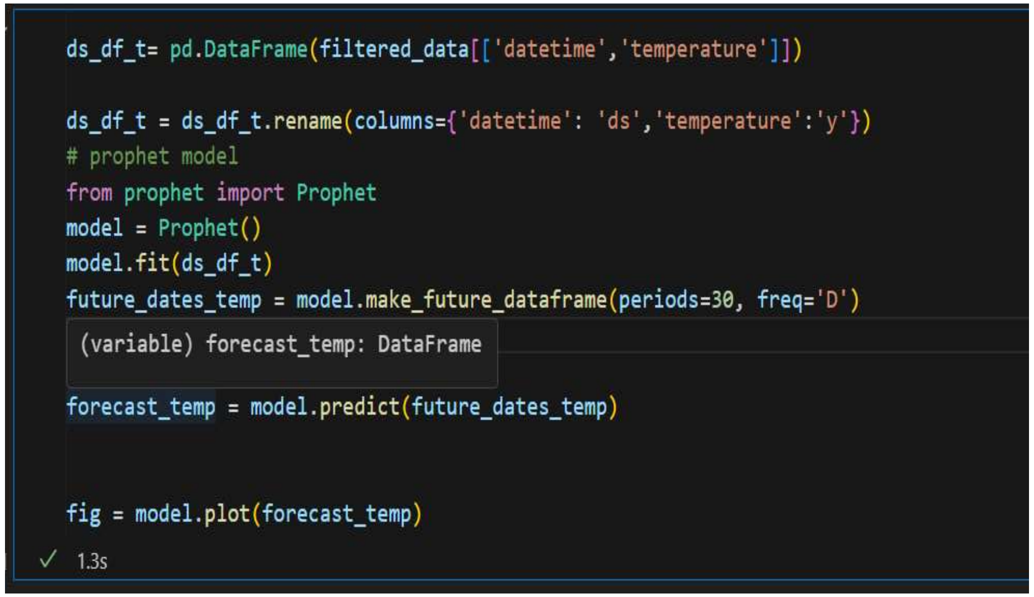
887 rows × 9 columns

time vs temperature

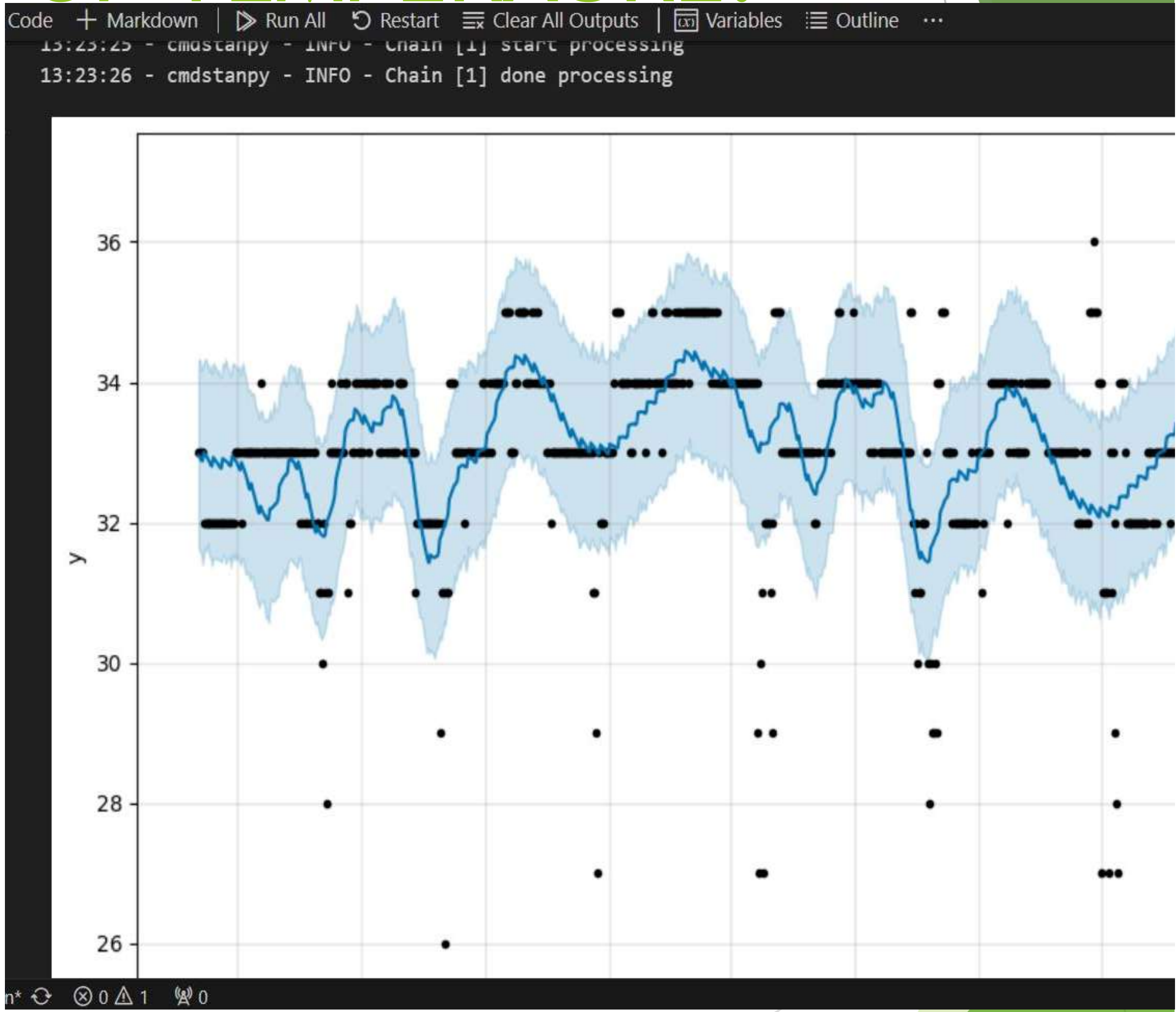



datetime vs humidity
humidity
100
80
60
40
20
Jul 2021
Jan 2022
Jul 2022
Jan 2023
Jul 2023
datetime

TIMESERIES ANALYSIS OF TEMPERAURE:

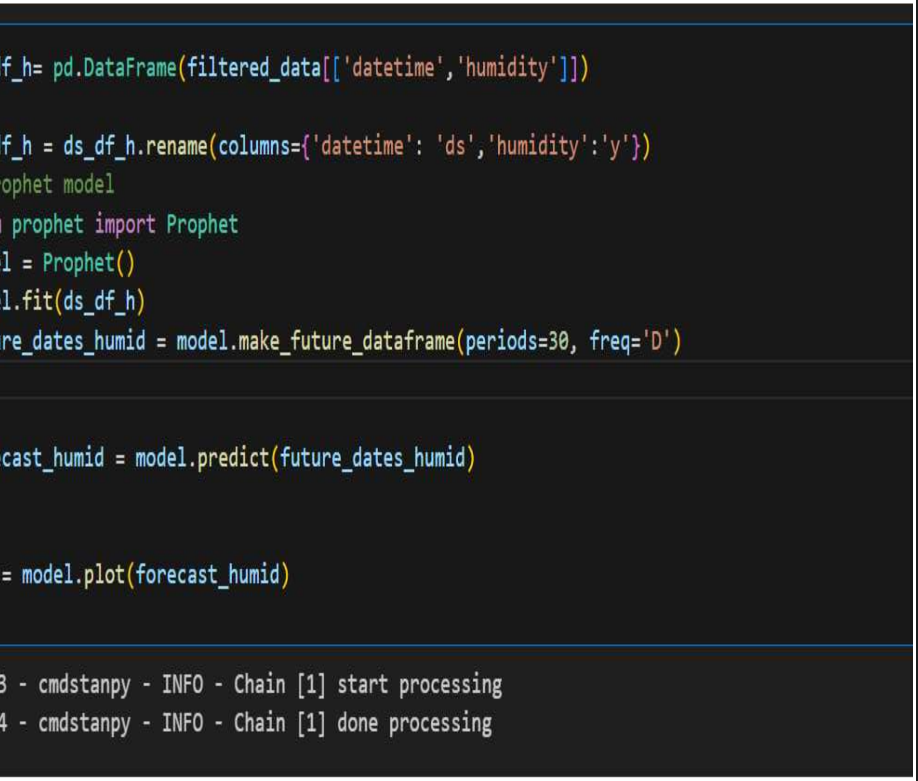


```
ds_df_t= pd.DataFrame(filtered_data[['datetime','temperature']])

ds_df_t = ds_df_t.rename(columns={'datetime': 'ds','temperature':'y'})
# prophet model
from prophet import Prophet
model = Prophet()
model.fit(ds_df_t)
future_dates_temp = model.make_future_dataframe(periods=30, freq='D')
(variable) forecast_temp: DataFrame

forecast_temp = model.predict(future_dates_temp)


fig = model.plot(forecast_temp)
✓ 1.3s
```

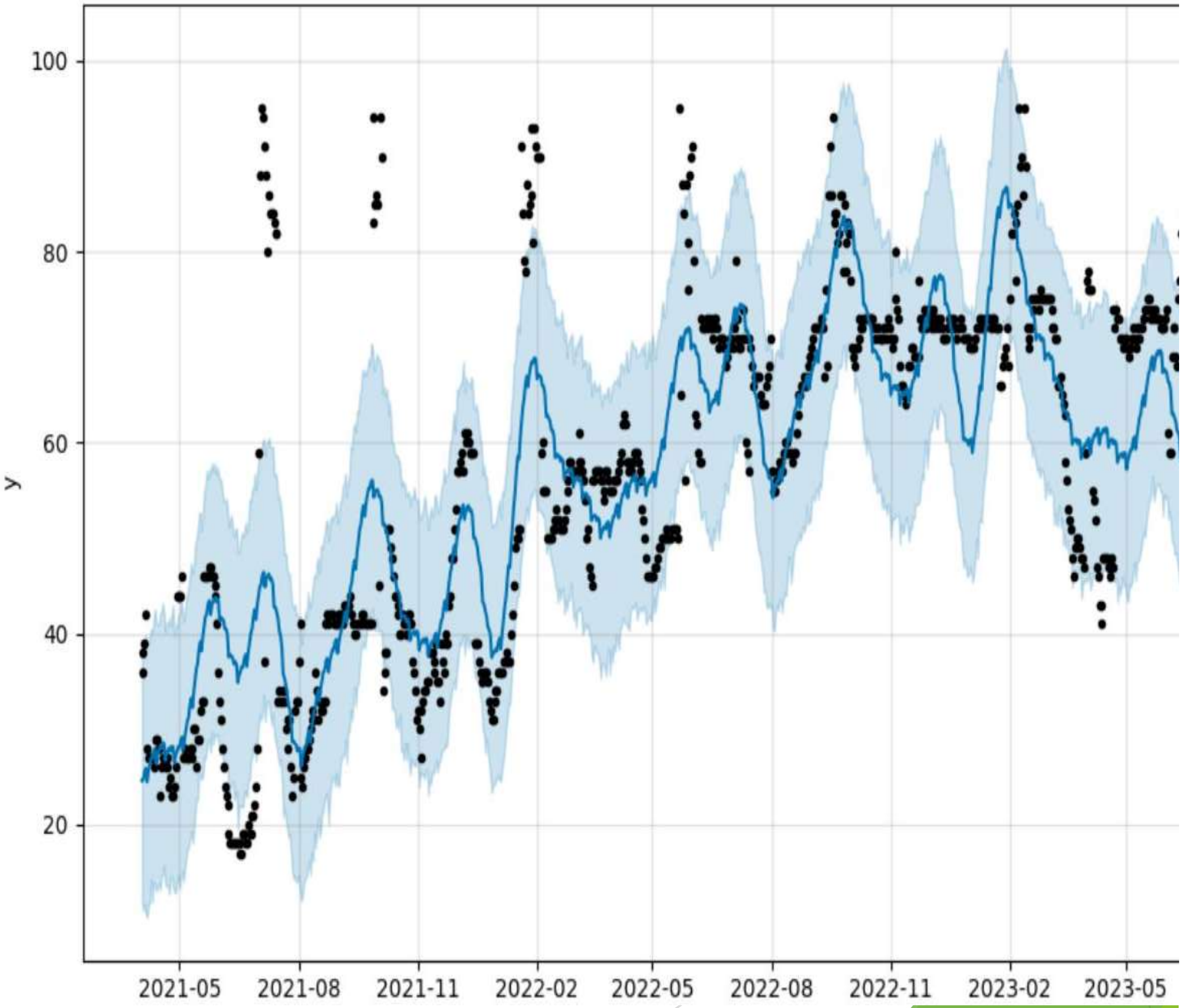


```
13:23:25 - cmdstanpy - INFO - Chain [1] start processing
13:23:26 - cmdstanpy - INFO - Chain [1] done processing
```

TIMESERIES ANALYSIS OF HUMIDITY:



```
f_h= pd.DataFrame(filtered_data[['datetime','humidity']])

f_h = ds_df_h.rename(columns={'datetime': 'ds','humidity':'y'})
ophet model
 prophet import Prophet
l = Prophet()
l.fit(ds_df_h)
re_dates_humid = model.make_future_dataframe(periods=30, freq='D')


cast_humid = model.predict(future_dates_humid)


= model.plot(forecast_humid)

3 - cmdstanpy - INFO - Chain [1] start processing
4 - cmdstanpy - INFO - Chain [1] done processing
```



100
80
60
40
20
y
2021-05
2021-08
2021-11
2022-02
2022-05
2022-08
2022-11
2023-02
2023-05