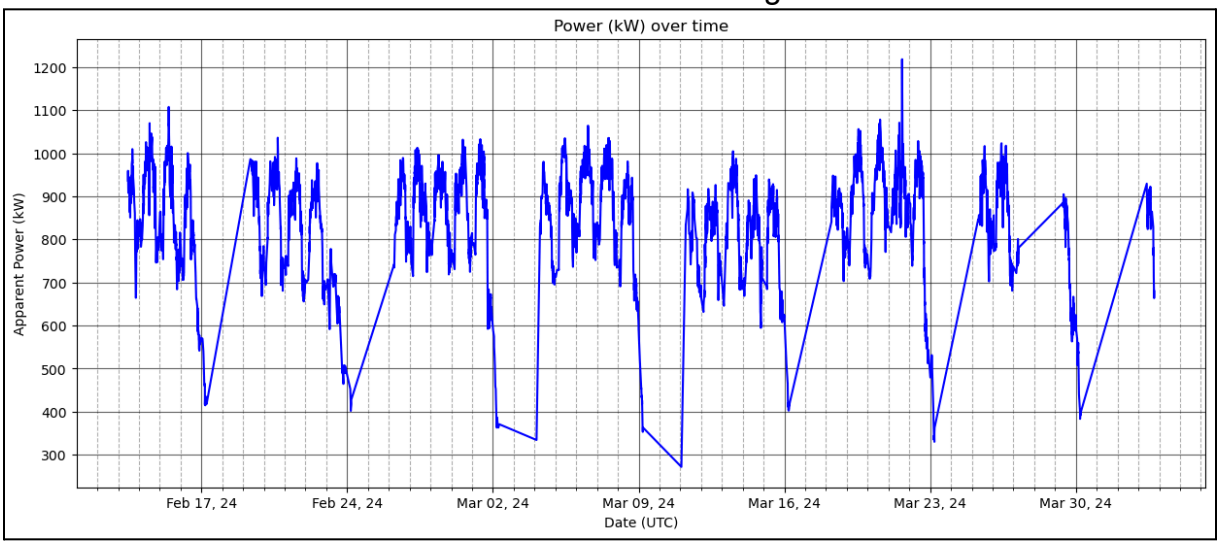
Introduction

I merged the energy consumption, temperature, and production datasets into one dataset by matching the date variable ‘at’. However, the energy consumption dataset is reported with increments by minutes, the temperature dataset is reported with increments in hours, and the production dataset is reported with time containing seconds.

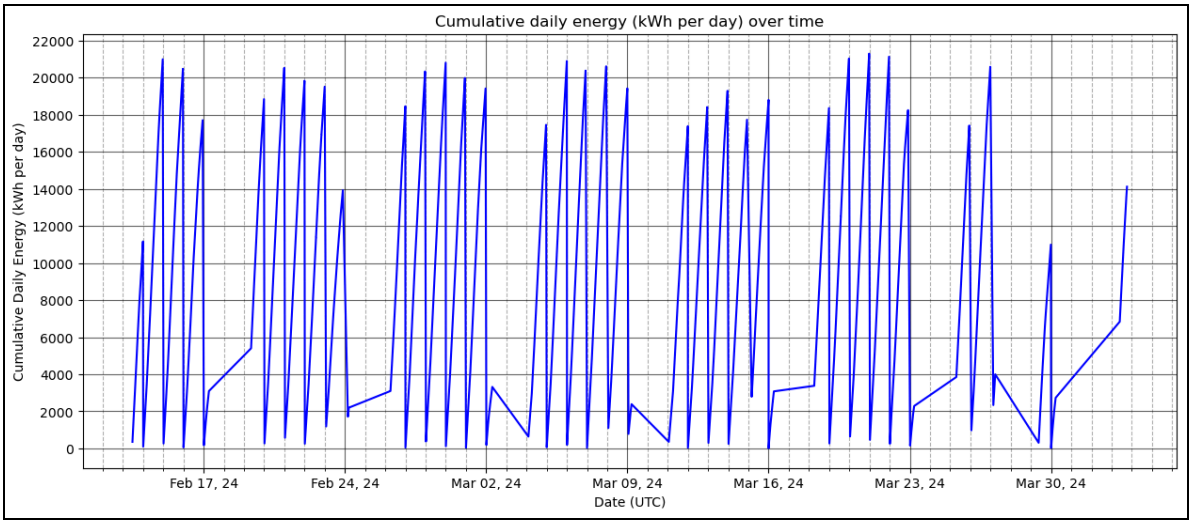
In order to merge the 3 datasets, I rounded the dates in the energy dataset to the nearest hour and matched those dates to the dates in the temperature dataset. Then, I rounded the dates in the production dataset to the nearest minute and matched those dates to the dates in the energy dataset. Through this method, all 3 datasets are able to be merged together into one dataset. Note that while the energy and temperature datasets are large-sized (i.e. over 70,000 observations), the production dataset is medium-sized (i.e around 9,000 observations) so the final merged dataset is only medium-sized (i.e. around 8,000 observations) since there are dates in energy and temperature that cannot be matched in the production dataset and vice-versa. I also calculated the daily energy consumption through sum(power * time) in the energy dataset.

Visuals

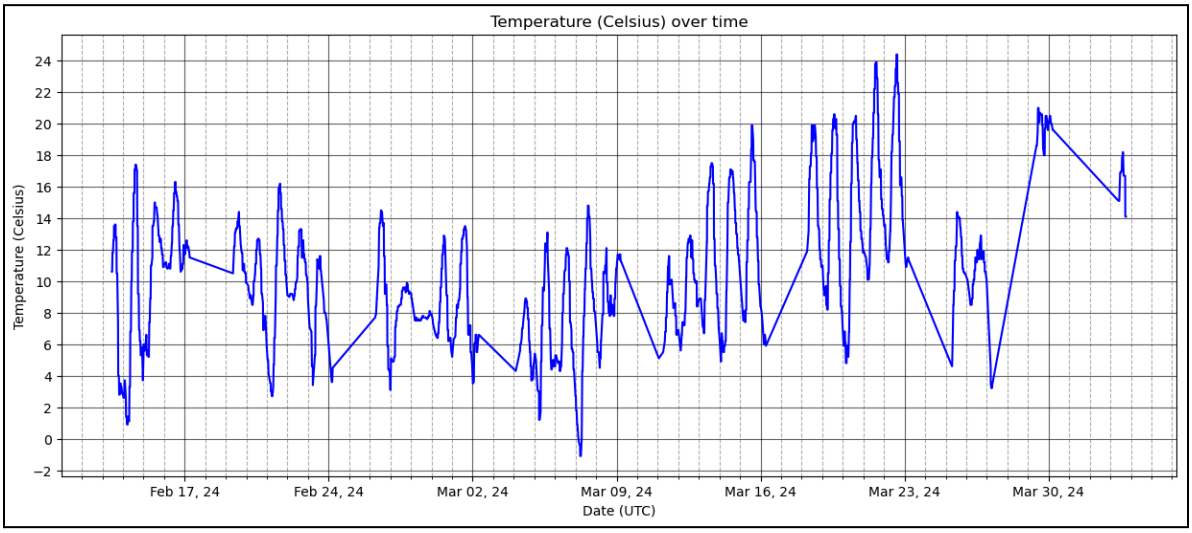
Below are visuals for some variables in the merged dataset.



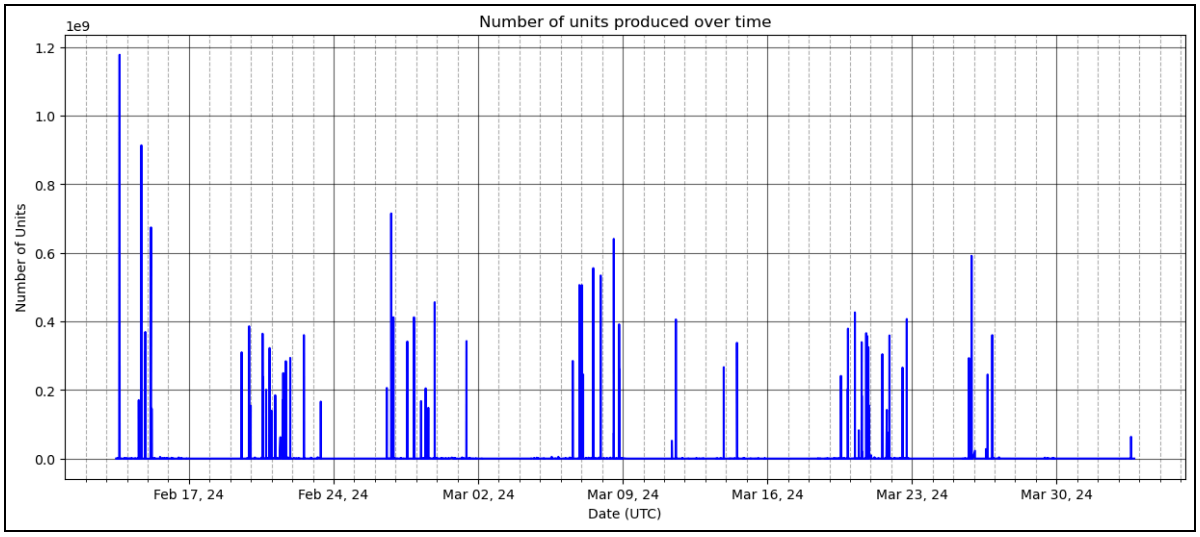
Power appears to still follow the same pattern as before following some sort of cyclic cycle every week (i.e 7 days). Note that almost all the anomalies since the dates where the anomalies occurred in the energy dataset were not able to be matched to the dates in the production dataset.



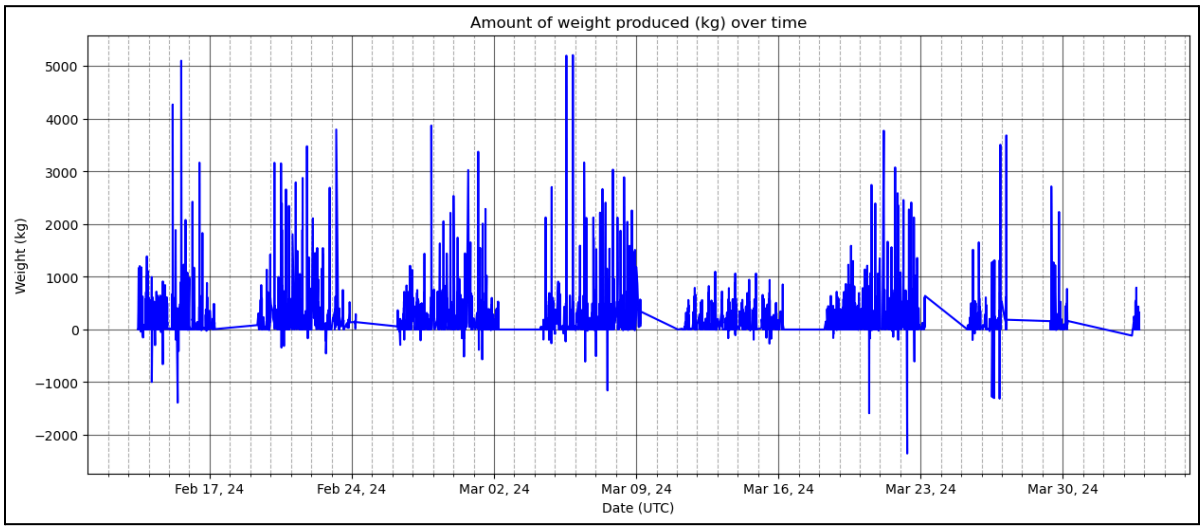
Daily energy consumption might seem like it is just jumping from 0 to ~20,000. Since I calculated energy consumption as daily, energy consumption always resets to 0 after every day. In fact, in the power over time graph above, the power generally stays around 700 to 1,000 kW. This means over a short time horizon, the power is somewhat constant so the energy consumption follows a linear trend (i.e. $E = \int P(t)\,dt = \int \text{constant}\,dt$ = linear), which is why it appears to be jumping from 0 to ~20,000. In actuality, energy is just increasing linearly quickly.



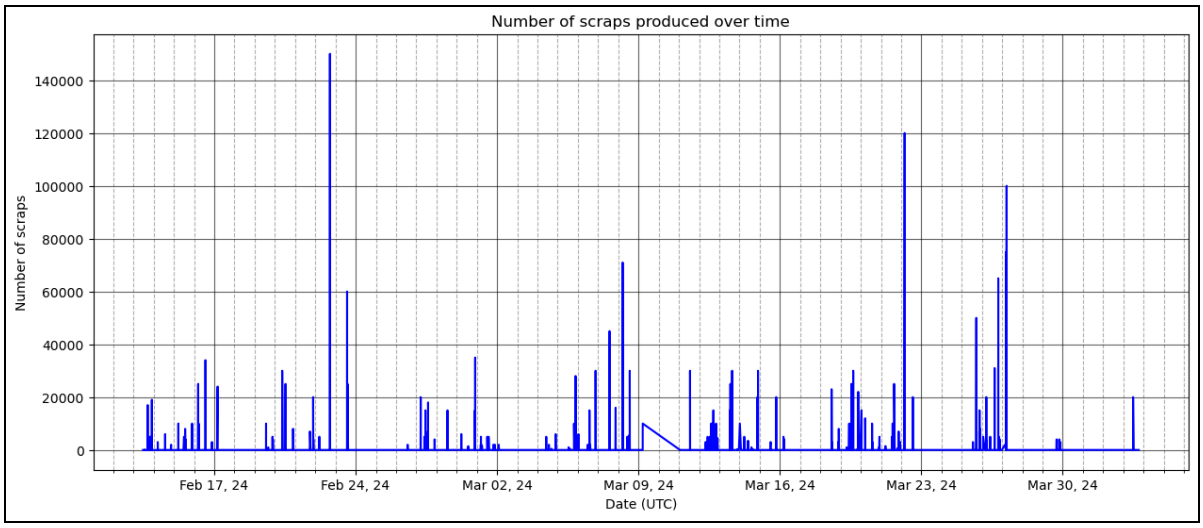
Compared to temperature following a clear wave-like pattern previously, over a short time horizon, temperature is generally sporadic although it tends to not deviate too much in value as they appear to be within 10 degrees or so as shown in the graph.



The number of units produced still follows the same trend as before where the number of units are just randomly produced in time.



The amount of weight produced still follows the same trend as before where the amount appears to be produced randomly in some days and never in others.



The number of scraps produced appears to follow the same trend as before where the number of scraps are just randomly produced in time.

Analysis

For this short report, I only performed analysis on energy consumption and amount of weights produced with only a Linear model, Decision Tree model, and Recurrent Neural Net.

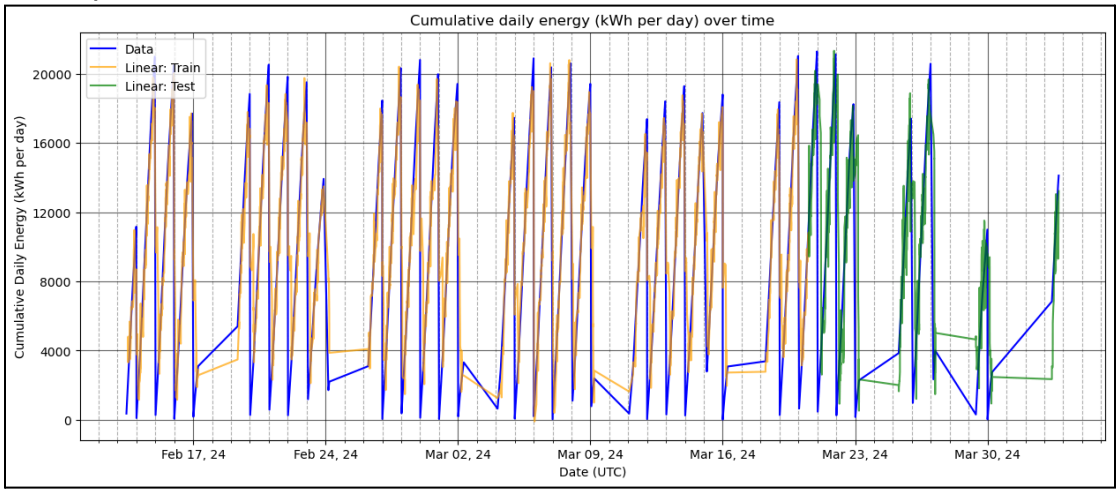
First, let's cover the analysis for energy consumption. Below are the results.

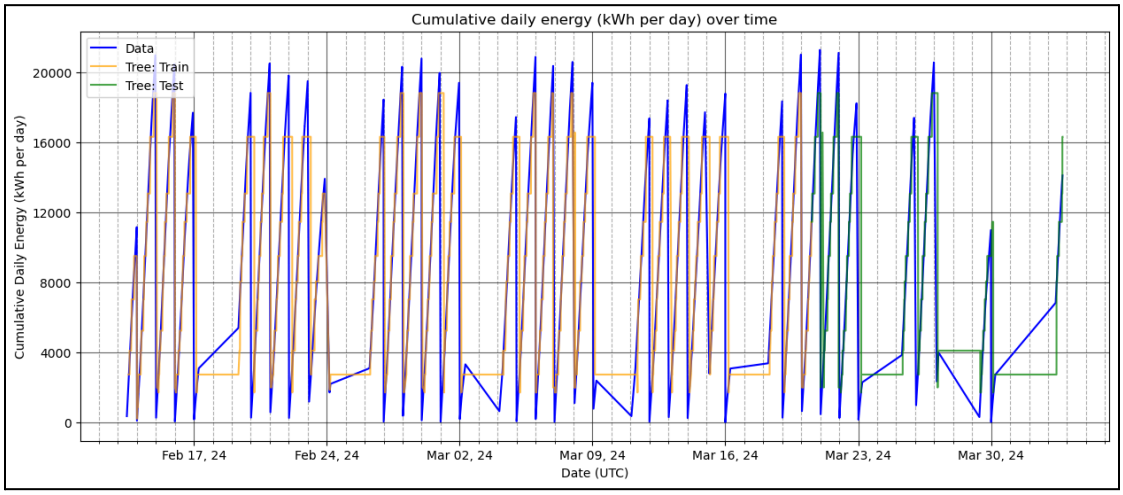
Lower is better	Linear	Tree	RNN
Training Error	1498	1993	609
Testing Error	2603	1942	2802
1st important	cumul_daily_energy at lag 1 (0.27)	cumul_daily_energy at lag 1 (0.97)	
2nd	cumul_daily_energy at lag 2 (0.17)	cumul_daily_energy at lag 8 (0.02)	
3rd	cumul_daily_energy at lag 3 (0.11)	Anything else	
4th	cumul_daily_energy at lag 4 (0.08)	Anything else	
5th	cumul_daily_energy at lag 5 (0.05)	Anything else	

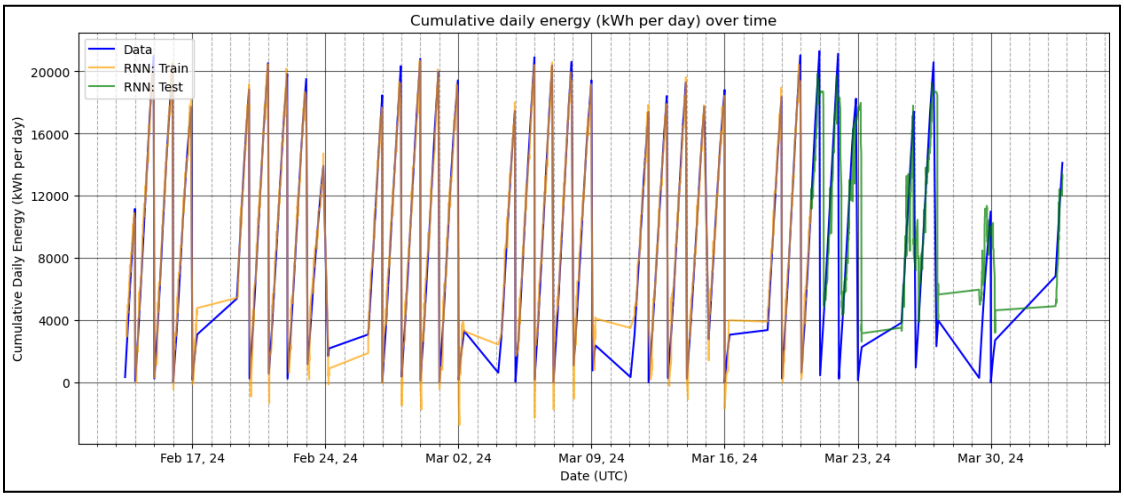
It appears that instead of all models being roughly equally accurate, the Decision Tree model is clearly the most accurate. The most important variable in forecasting energy consumption in both models is the information on energy consumption that is most recently available.

However, it is important to note that all previous models with only the energy consumption data performed much better than the merged dataset. This is probably because it already followed a clear trend without having information from production. Since a lot of data was removed due to the lack of matching dates in production, the model appears to have suffered.

Below are the predictions made by each model. All models appear to predict the trend quite well.







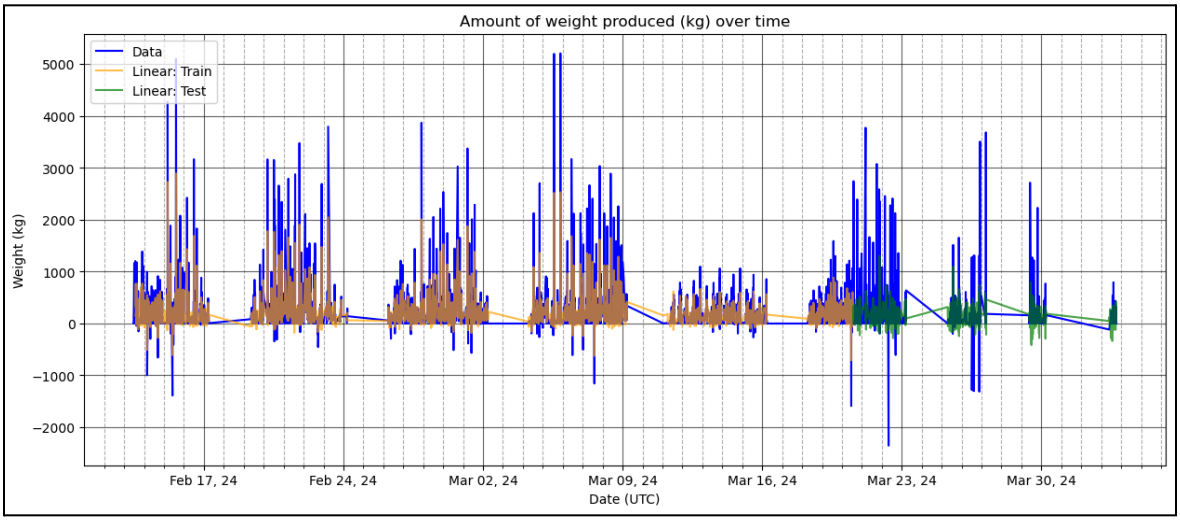
Next, below are the results of my models on the amount of weight produced by treating the production dataset as a time series instead of before where I was treating the production dataset as just a regression.

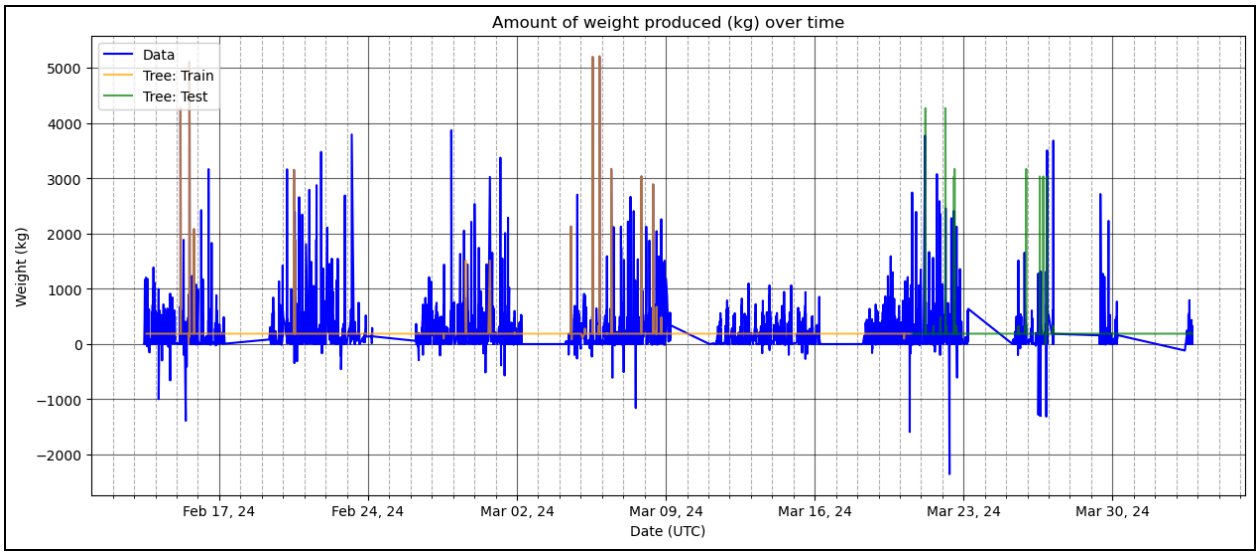
Lower is better	Linear	Tree	RNN
Training Error	162	262	110
Testing Error	375	437	371
1st important	sku_1575763 at lag 5 (0.15)	group_4G140400 at lag 2 (0.13)	
2nd	sku_1208324 at lag 43 (0.15)	sku_762385 at lag 55 (0.13)	
3rd	Poste de travail_40418 at lag 2 (0.14)	weight_kg at lag 33 (0.13)	
4th	Poste de travail_40146 at lag 5 (0.14)	papp at lag 15 (0.06)	
5th	sku_1753241 at lag 21 (0.13)	sku_1635500 at lag 50 (0.06)	

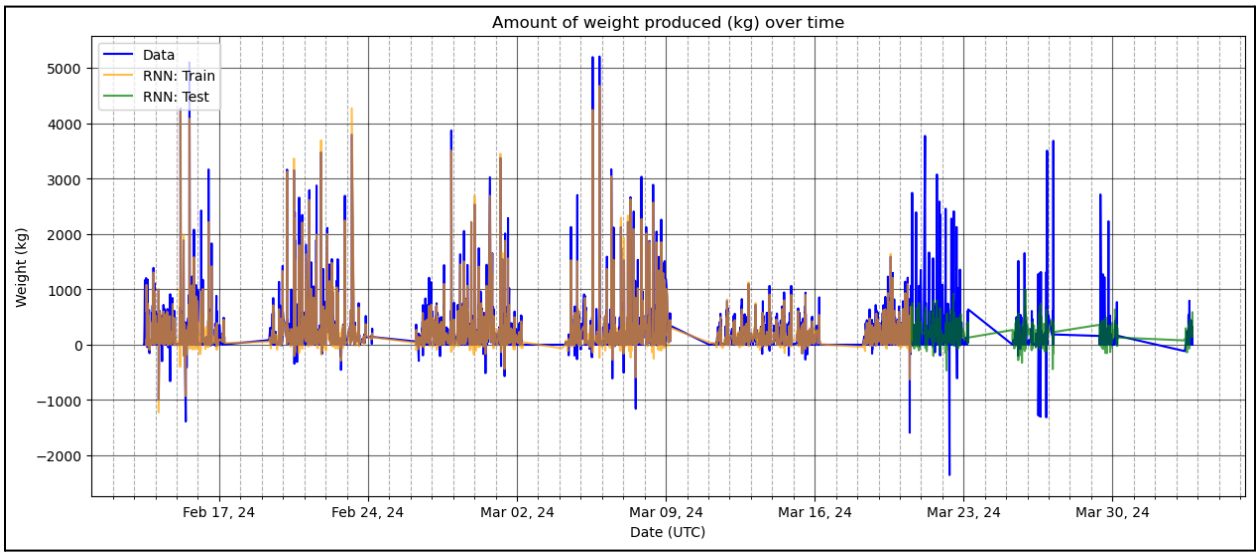
It appears that the most accurate models are the linear model and the recurrent neural net. Additionally, it appears that when trying to predict the amount of weight produced, the information about its product reference (i.e. sku) and Poste de travail are both roughly equally as important. Also, some groups are important, but Centre de coûts does not appear to be important.

The results somewhat match the trend when I found the most important variables when treating the production data as just a regression. Some sku are important and Centre de coûts are not as important.

Below are the predictions made by each model. The RNN model appears to be the most accurate in predicting the trend, but all models fail to perform well on the testing data as noted by the green note matching the blue very well. This is most likely due to the fact that even though the amount of weight produced is within certain intervals, the amount produced appears to be sporadic within each interval and lack a clear trend.




Amount of weight produced (kg) over time
Data
Tree: Train
Tree: Test
Weight (kg)
5000
4000
3000
2000
1000
0
−1000
−2000
Feb 17, 24
Feb 24, 24
Mar 02, 24
Mar 09, 24
Mar 16, 24
Mar 23, 24
Mar 30, 24
Date (UTC)


Amount of weight produced (kg) over time
Data
RNN: Train
RNN: Test
Weight (kg)
5000
4000
3000
2000
1000
0
−1000
−2000
Feb 17, 24
Feb 24, 24
Mar 02, 24
Mar 09, 24
Mar 16, 24
Mar 23, 24
Mar 30, 24
Date (UTC)