# Limitations and Opportunities for Improvement

## Data

### ChatGPT challenges

I had some difficulties with generating the data using ChatGPT. See the “ChatGPT Challenges” folder for failed versions.

Compared to the documentation from AEMO about the NEM12 file format, there seems to be some data misplaced or missing. However, I was able to generate a file with an overall format that matches the majority of the NEM12 specifications. This is what I used in the model.

### Reading/parsing NEM12 dataset

The way I parse the NEM12 data into a usable format relies on some “hardcoded” values (see In[3] line 3). This will not work for datasets with a different metering interval than mine. It also doesn’t allow for reading of some of the other information that might be important, such as meter number or any of the data quality flags. Nor does it have a way to account for bad or missing data.

There is definitely an opportunity to build a more robust parsing tool that can take in NEM12 files of all types (e.g. different metering intervals, multiple meters, etc). I was able to find some open-source tools online, and I imagine Flow Power has tools of their own as well.

### Electric pricing dataset

I could only find real time electric pricing data for a single day, which was just the previous 24 hours from when I accessed the website. Though this dataset was used to represent the daily electric price in the model, it is not actually accurate as prices will change day to day.

In future iterations of the model, actual historic pricing data across longer durations should be used to forecast future performance.

## Battery

### Assumptions

Much of the battery parameters used in the model were assumptions based on my knowledge (for example the battery charge/discharge limits, see In[12] lines 7-8). Ideally, these should be sourced from actual data or manufacturer specifications.

### Functionality

Right now, the battery is only expected to perform one charge and discharge per day, and the model doesn’t allow for more complicated battery set ups or operations. It also doesn’t provide the user with any information on the ancillary equipment required for an actual installation (e.g. inverters, etc).

Customers installing batteries might also have concurrent plans for solar, which is not accounted for with this model

Developing this model to account for greater battery functionality or integration with other systems is probably a good idea.

## Model

### Simplicity

The model only examines the performance of the battery given an arbitrarily set offset period that is the same throughout the year. It doesn’t account for any automation or price signalling to make decisions.

Further improvement ideas/questions:

* A simple extension of this model would be to compare performance of different offset periods (e.g. 2-6pm or 5-9pm).
* In the model, the battery is discharging 17% of the time. It would be interesting to see how much additional cost savings the battery can achieve if it maintains this 17% capacity factor, but instead of a predetermined offset period, it offsets energy use during the most expensive 17% of prices.
* This model is based on completely offsetting usage during discharge. Batteries could also discharge during times where its full capacity would not be enough to meet the entirety of the site’s demand.

Building in some functionality to ramp up and down the discharge rate could be helpful, for example if a customer expresses the desire to run just a small subset of equipment off the batteries as opposed to the entire site.

In this case, modelling equipment usage would also be required to match the battery performance to the equipment operation.

* The model doesn’t account for the industry’s ability to forecast prices. I think that there is a lot of opportunity for advancement here. One idea is to enable “premature” discharging, even during times where prices are relatively low or normal, IF there is a confident forecast of even lower or negative pricing coming up soon.

### Usability

The model doesn’t allow for any user input, so changing any of the modelling parameters might require some python knowledge, which might be an issue if this was intended to be a customer-facing tool.

Development of the model should prioritize usability, meaning being able to easily change battery and modelling parameters without having to go through the code.

### Coding

As I don’t have much experience with python, I am sure there are plenty of areas in my code where I do not follow best practices, or I might not have analysed the data or structured them in the most optimal ways. Improvements could be made to align my code style to best or common practice as well as runtime optimisation for larger datasets or more complicated battery set ups.

### Financials

In the model, I have presented a simple payback period calculation based on the capital expense of purchasing the batteries and the annual electric bill cost savings. I don’t think this is an investment ready presentation.

Further improvement should be made to account for operating costs, maintenance costs, construction costs, expected future electric prices, inflation, etc. Unrelated to financials, but projecting performance beyond just the first year will also introduce some complexity in making sure the model accounts for battery wear and deterioration in efficiency over time.