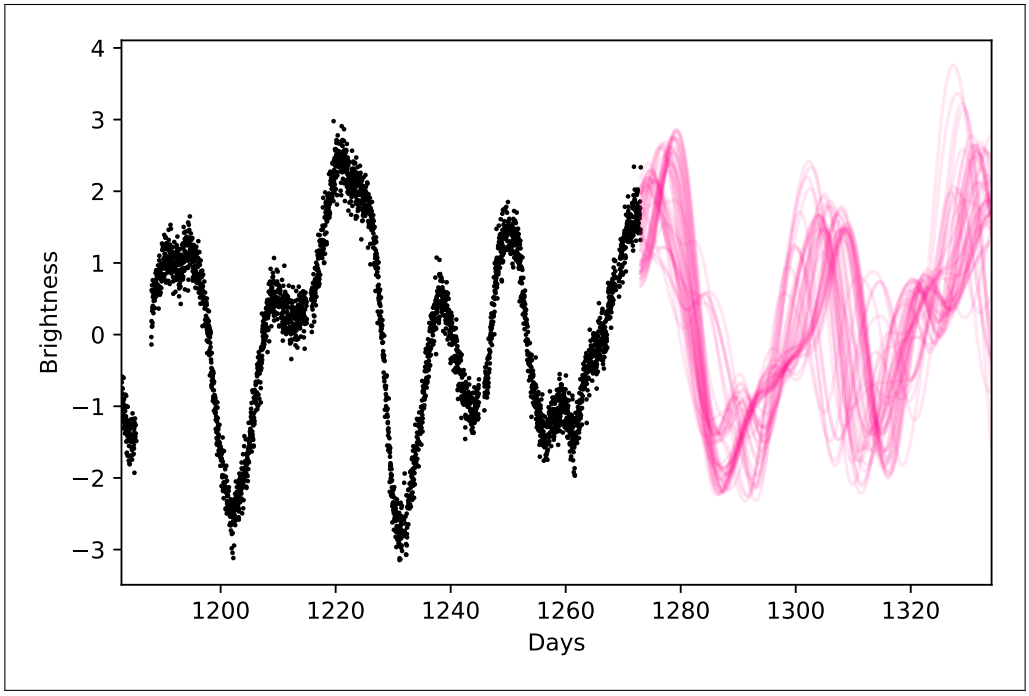


Figure 12

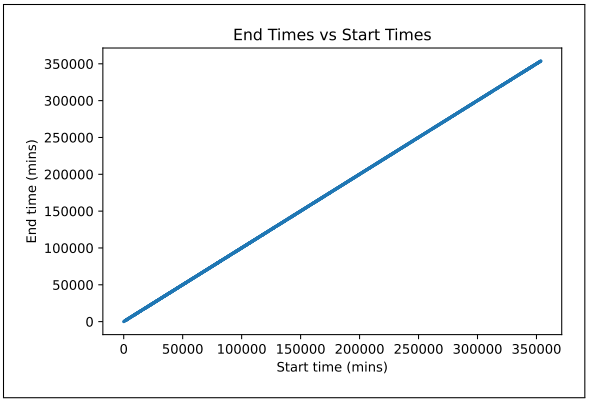
Although this prediction looks messier, it reflects the vague nature of our uncertainty. I note that if I were able to run more samples, the data may seem slightly neater - while I'm aware of this, the process was very time-expensive. Visually, it seems that the chains did provide values that make predictions appear consistent with the data. However, ideally one would conduct more samples (with a better computer than mine) to get a more accurate distribution of possible predictions.

3 Question 5

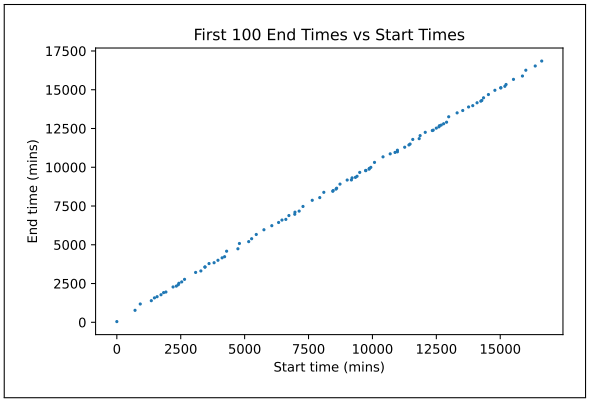
The first thing I noted about this data was that the data was ordered from latest time to earliest. Before even beginning the proper "cleaning" process, I sorted the data to be in chronological order, to make comparisons slightly easier when cleaning the data.

3.1 Before cleaning the data

To get a sense of the data, I first plotted... everything against... everything. The following plots are my initial foray into the (now sorted) dataset, which only consisted of a row of "starting times" and "ending times" over the course of several months. Note that I plotted the "starting time" values in terms of time passed from the first measurement's starting time (i.e. starting from 0). I also plotting the "durations" as the difference between the end and start times, and the "delays" as the time between an end time and the subsequent startime time.


End Times vs Start Times
End time (mins)
Start time (mins)
0
50000
100000
150000
200000
250000
300000
350000

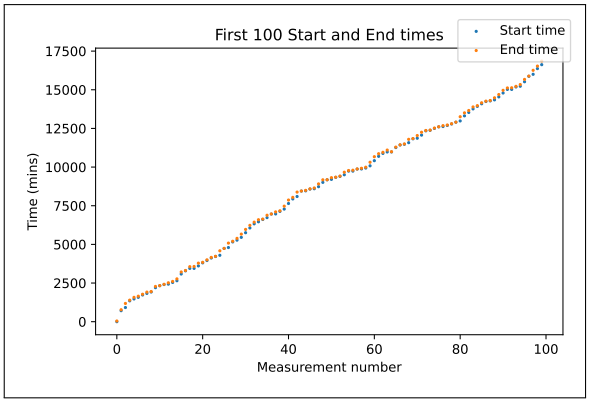
(a)


First 100 End Times vs Start Times
End time (mins)
Start time (mins)
0
2500
5000
7500
10000
12500
15000
17500

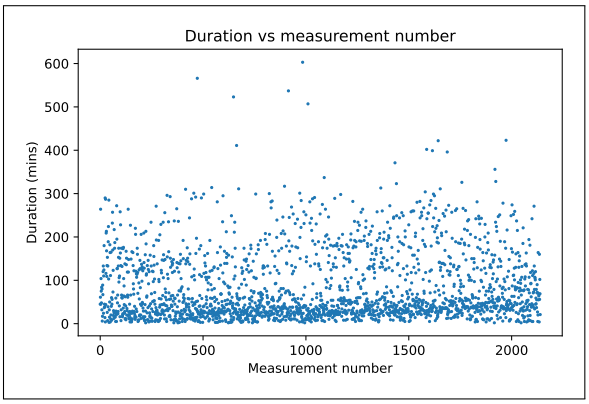
(b)


Plot of start and end times against "measurement number"
Time (mins)
Measurement number
0
50000
100000
150000
200000
250000
300000
350000
500
1000
1500
2000

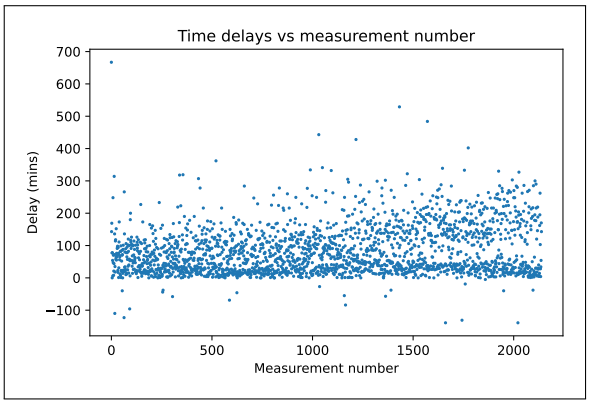
(c)


First 100 Start and End times
Start time
End time
Time (mins)
Measurement number
0
2500
5000
7500
10000
12500
15000
17500
20
40
60
80
100

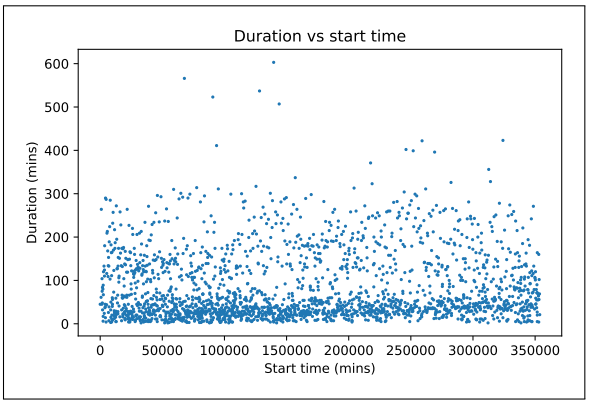
(d)


Duration vs measurement number
Duration (mins)
Measurement number
0
100
200
300
400
500
600
500
1000
1500
2000

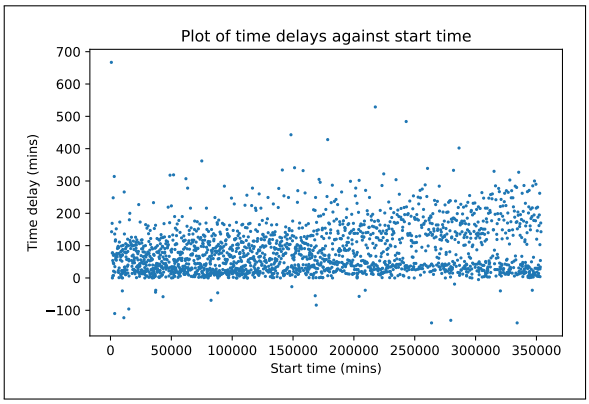
(e)


Time delays vs measurement number
Delay (mins)
Measurement number
−100
0
100
200
300
400
500
600
700
500
1000
1500
2000

(f)


Duration vs start time
Duration (mins)
Start time (mins)
0
100
200
300
400
500
600
50000
100000
150000
200000
250000
300000
350000

(g)


Plot of time delays against start time
Time delay (mins)
Start time (mins)
−100
0
100
200
300
400
500
600
700
50000
100000
150000
200000
250000
300000
350000

(h)

Comments

- Durations and delays appear to have some constant value with spread to higher values;
- There appears to be a very strong linear relationship between end and start time;
- There is a less obvious relationship between start/end time and measurement number;
- Durations don't have a strong relationship with the start time nor experiment number, but there is a gathering of points at values around 40. There appears to be a slight increase in the value of this cluster, but it's not very significant.
- The time delays" - not completely similarly spread, like with duration. There seems to be an increase.
- to model this, I looked at the relationship between the start time and measurement number.

3.2 Cleaning the data

I first removed any duplicates from the data and any empty entries in the csv file.

I then looked to filtering out the outliers for the duration values. I did this by implementing a mixture model of two possible models: the relevant model, and the outlier model.

I conducted inference for the proportion of the duration data that were outliers, by using cmdstanpy to sample the parameters using the assumption of a constant value model. As seen in Figure 13e and the apparent linear relationship between start and end times, this is a sensible model to apply.

While no uncertainties were provided in the data, I made the assumption that the time measurements had some uncertainty associated with them. However, given this assumption, I marginalised over these uncertainties such that the the likelihood for this outlier model was:

$$\mathcal{L} = p(y, |x, \mu_{0_y}, \sigma, Q) \propto \prod_{i=1}^{N} \left(\frac{Q}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - \mu_{0_y})^2}{2\sigma^2}} + \frac{(1-Q)}{2\pi\sqrt{\sigma^2 + \sigma_o^2}} e^{-\left(\frac{(y_i - \mu_{0_y})^2}{2(\sigma^2 + \sigma_o^2)}\right)} \right), \tag{45}$$

Where x and y are measurement number and durations respectively, σ is the marginalised uncertainty for the data, and σ_o is the added uncertainty associated with the outlier model (which is "blurrier", or more widely spread).

The stan model for this mixture model was accordingly:

data {

int<lower=0> l;

vector[l] y;

}

parameters {

real<lower=0> sigma_real;

real<lower=0> sigma_out;

real<lower=0,upper=1> q;

real mu_real;

real mu_out;

}

model {

for (i in 1:l){

real log_prob1 = normal_lpdf(y[i] — mu_real, sigma_real);

real log_prob2 = normal_lpdf(y[i] — mu_out, sigma_real+sigma_out);

target+= log_mix(q,log_prob1, log_prob2);

}

}

generated quantities {

vector[l] lp1;

vector[l] lp2;

for (i in 1:l){

lp1[i] = normal_lpdf(y[i] — mu_real, sigma_real);

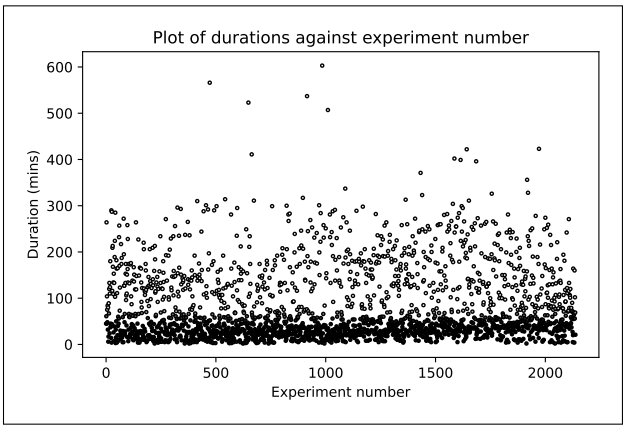
lp2[i] = normal_lpdf(y[i] — mu_out, sigma_real+sigma_out);

}

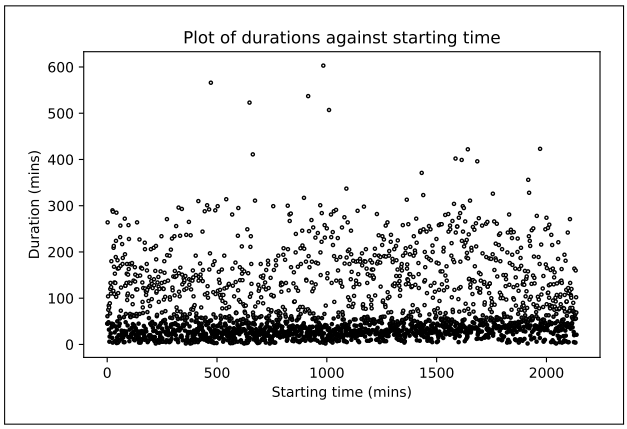
}

It was not possible to really assess the outliers for the delay times, as during the cleaning process, entries in the dataframe were removed. This altered the effective recorded time difference between readings. Therefore, I didn't engage in any "cleaning" based on the distribution of the time delays.

The following plot identifies (by transparency) which duration values are outliers.



(a)



(b)

To be able to remove appropriate points, I chose a threshold for ignored points. Points with a likelihood of belonging to the main model less than 10^{-8} were ignored. The result of this threshold is plotted in the following figure, illustrating the new clean data.

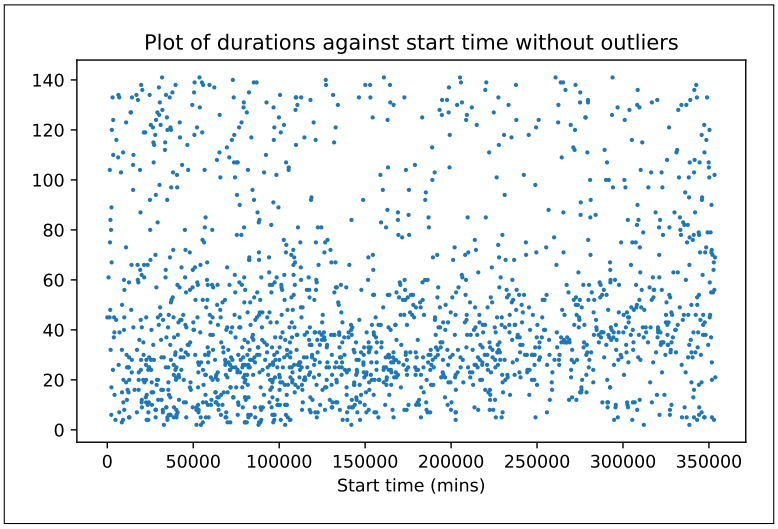


Figure 15

While I did not conduct inference on outliers for the delay times between measurements, Figure 13h shows that some delay times in the raw data are negative. I made the conclusion by inspection that this data must not be "clean" data, as there were very few instances of this occurring, and it was unlikely that such separate durations could possibly occur concurrently. If the data involved the measurement of durations that could be concurrent, more negative values would be anticipated.

Therefore, I removed the data that included start times occuring before the previous measurement (in chronological order) was finished. The result of this, compared with Figure 13h, is plotted below. It can be seen that the sparse negative values have been removed.

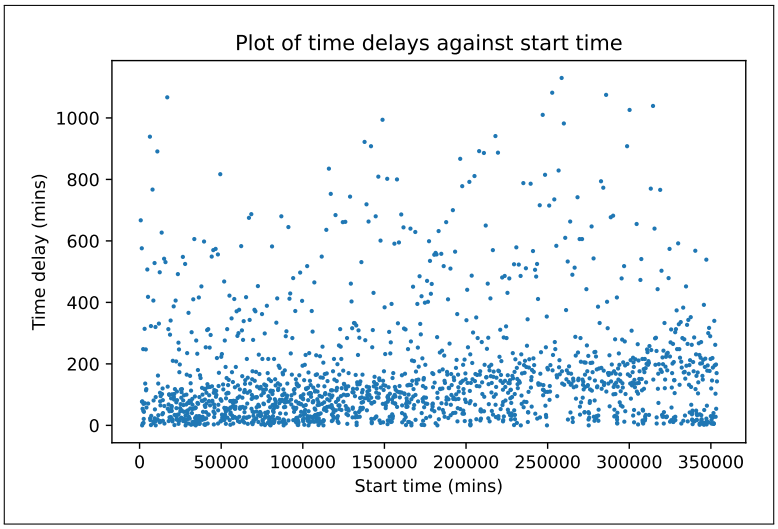
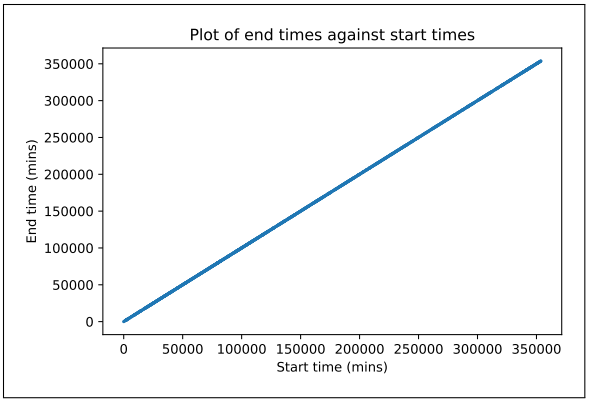


Figure 16

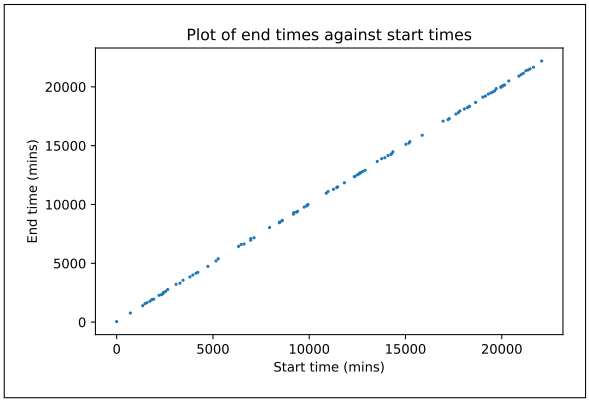
At this point, I declared the data was "clean".

3.3 Modelling the clean data

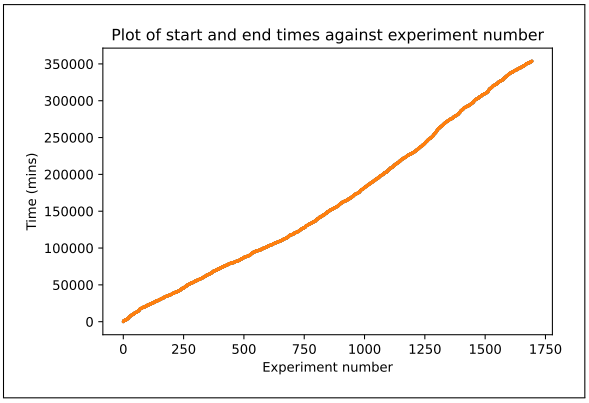
Now using the clean data, the relevant plots that reveal patterns in the data are found below.



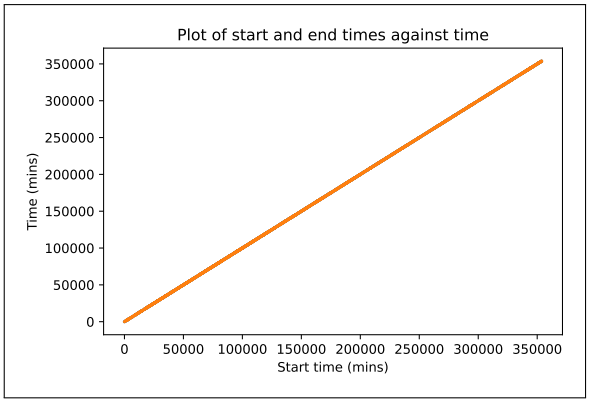
(a)



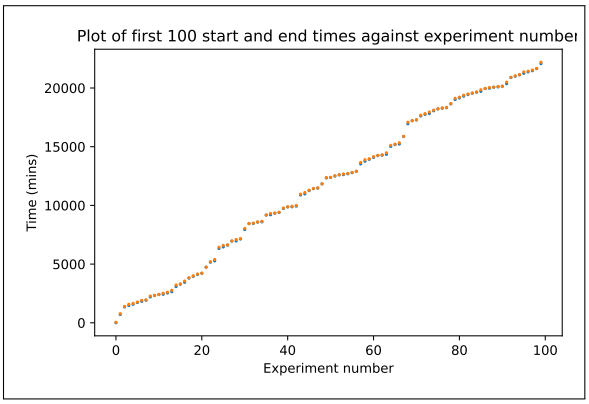
(b)



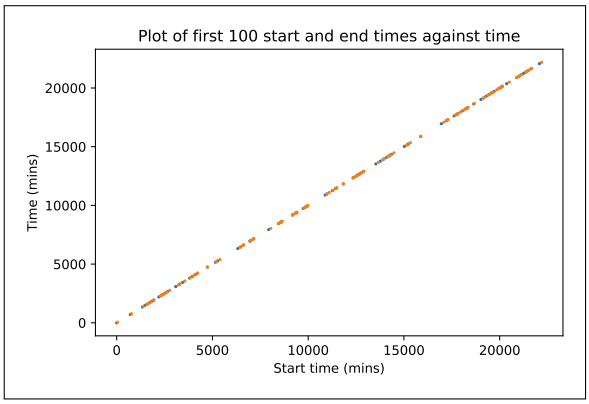
(c)



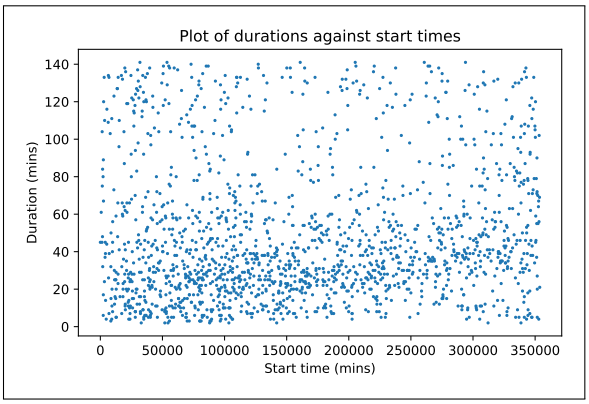
(d)



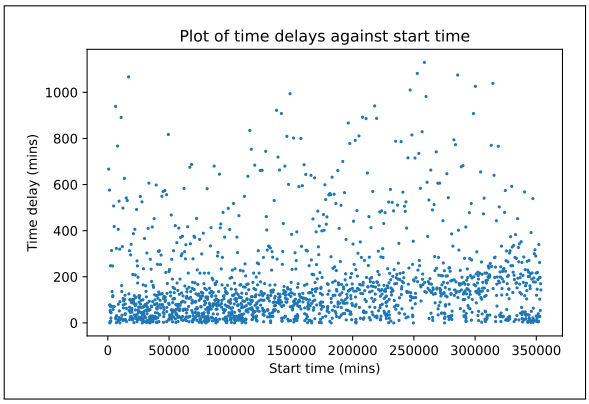
(e)



(f)



(g)



(h)

Figure 17

At this point, I will set out to model three relationships that I can see in the data. Firstly, the relationship between end time and start time, such that an end time can be predicted for a given start time (effectively modelling the duration).

3.3.1 Modelling end time vs start time

As mentioned before, one of the stronger patterns seen in the data is the relationship between start times and end times, seen in 17a. The linearity of this relationship is preserved even when zooming in, as shown in 17b.

If this relationship is linear, that would mean that "duration" is effectively a constant. This somewhat agrees with the plot of durations against start times, shown in Figure 15, which despite having a large variance in the data (ranging from 0 to 140 minutes), appears to be slightly more concentrated around a single value (around 20 mins). This concentration dilutes slightly at higher starting times, but there isn't a particularly distinct increase.

Therefore, it seemed sensible to use a **linear** generative model for the relationship between end and start time.

I modelled the data in a linear fashion, such that:

$$y_i \; \mathcal{N}(mx_i + c, \sigma_i) \tag{46}$$

Where y_i and x_i are the end and start times respectively, m is the gradient and c is the offset, and σ_i represents the uncertainty of the model. I marginalised over this uncertainty, so it can be said that the model is:

$$y_i \; \mathcal{N}(mx_i + c, \sigma), \tag{47}$$

Where σ is the convolved uncertainty of the Gaussian as a result of marginalisation over the system's own uncertainty.

The data itself did not contain uncertainties, but I included them as parameters to be sampled in the inference.

Using conditional probability:

$$p(y_i, \sigma_{y_i}, m, c|x_i) = p(y_i|x_i, \sigma_{y_i}, m, c)p(\sigma_{y_i})p(m, c) \tag{48}$$
$$\tag{49}$$

where

$$p(y_i|x_i, \sigma_{y_i}, m, c) = \frac{1}{2\pi\sigma_{y_i}^2} e^{-\frac{(y_i-(mx_i+c))^2}{2\sigma_{y_i}^2}}. \tag{50}$$

I did not immediately care about finding the uncertainty σ_{y_i} for each data point. Therefore, I marginalised over this parameter using a half-normal prior:

$$f \sim \mathcal{N}(\mu_\sigma, \sigma_\sigma) \tag{51}$$
$$\sigma_{y_i} = |f|. \tag{52}$$

Following from the fact that convolution of Gaussians results in a Gaussian, our marginalised probability then becomes:

$$p(y_i, \sigma_y, m, c|x_i) = \frac{1}{2\pi\sigma_T^2} e^{-\frac{(y_i-(mx_i+c))^2}{2\sigma_T^2}}. \tag{53}$$

Where $\sigma_T^2 = \sigma_{y_i}^2 + \sigma_\sigma^2$.

Further, to ensure the prior distribution of m is equal over all the possible gradients, the prior probability for m and c is given by:

$$p(m, c) = (1 + m^2)^{-3/2}. \tag{54}$$

The total probability of the system is therefore (where N is the number of points):

$$p(y_i, \sigma_y, m, c|x_i) = \prod_{i=1}^{N} (p(y_i|x_i, \sigma_y, m, c)p(m, c)) \tag{55}$$

$$p(y_i, \sigma_y, m, c|x_i) = \prod_{i=1}^{N} \left(\frac{1}{2\pi\sigma_T^2} e^{-\frac{(y_i-(mx_i+c))^2}{2\sigma_T^2}} (1 + m^2)^{-3/2} \right) \tag{56}$$

$$\tag{57}$$

The log probability of the system (where x is the experiment number) is therefore:

$$\ln \mathcal{L} = -\frac{3}{2} N \ln\left(1 + m^2\right) - N \ln 2\pi\sigma_T^2 - \sum_{i=1}^{N} \frac{\left(y_i - (mx_i + c)\right)^2}{2\sigma_T^2}. \tag{58}$$

Coding steps I first used the least squares method in linear algebra to approximate m and c. I then used curve fit to get a refined initialisation for m and c.

The fitted parameter values were:

$$c = 44.09554 \tag{59}$$

$$m = 1.000025. \tag{60}$$

Finally, I used these initialisations to sample the parameters, using cmdstanpy. As the line seems very strongly linear, I could have chosen to not use stan model inference, and instead leave it at curve_fit. However, as I was going to make predictions, it was worth considering the standard deviations in the sampling, and to allow these to propagate into the future prediction.

I used the following stan model to sample the parameters of this system.

data {

int<lower=0> N;

vector[N] x;

vector[N] y;

}

parameters {

real m;

real c;

real<lower=0> sigma;

}

model {

c uniform(-100,100);

sigma normal(0,1000);

y normal(c + x * m, sigma);

}

The resulting chains from this sampling are plotted below, for all three parameters.

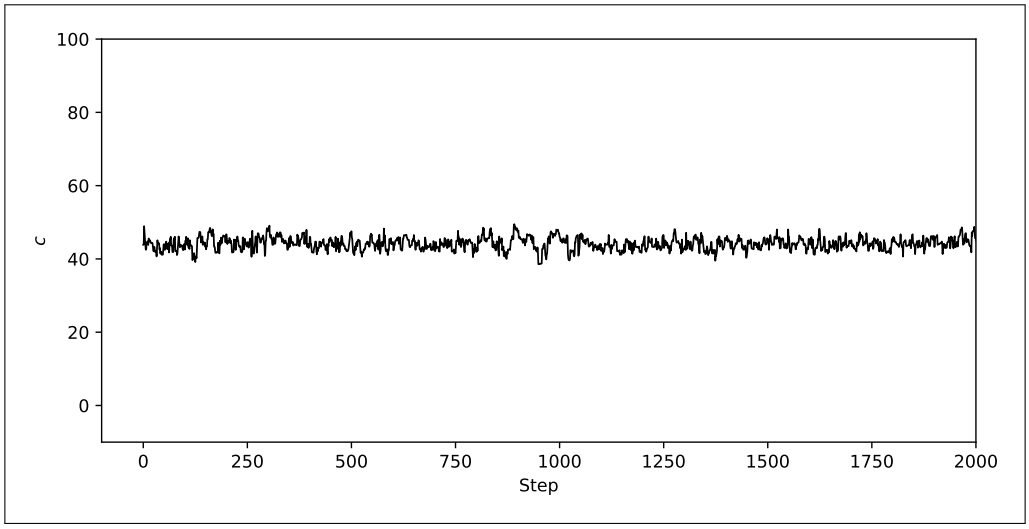


Figure 18

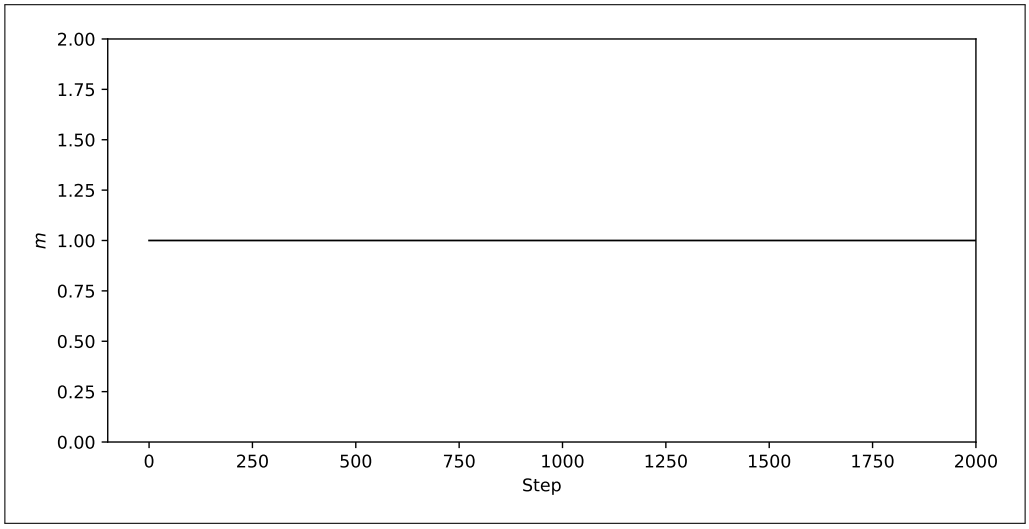


Figure 19

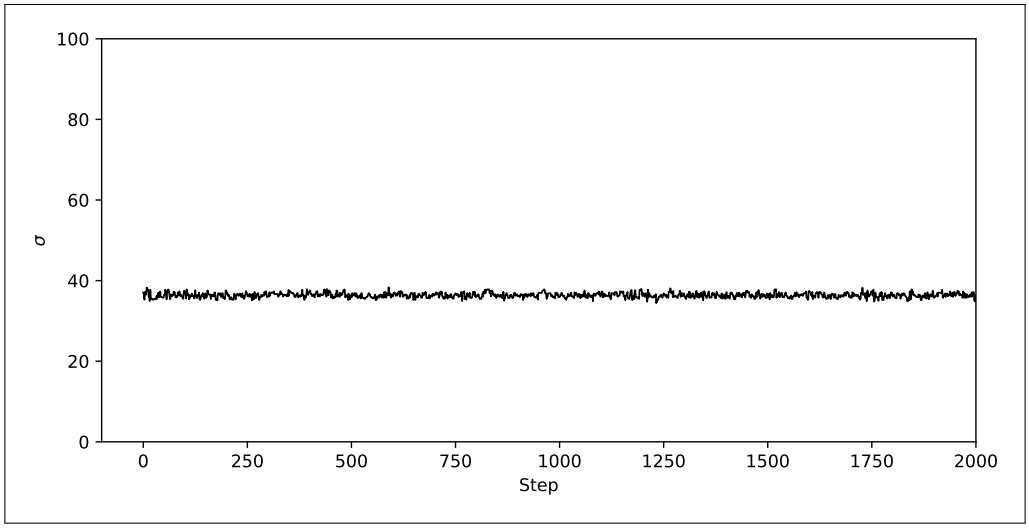


Figure 20

The above chains not only appear to have converged, but their $\hat{R}$ values are each close to 1. The resultant parameters from sampling are:

$$c = 44.2601 \text{ with a standard deviation: } 1.7776 \tag{61}$$

$$m = 1.000025 \text{ with a standard deviation: } 9.6987 \times 10^{-6} \tag{62}$$

$$\sigma_T = 36.3364193 \text{ with a standard deviation: } 0.61837. \tag{63}$$

We will return to this model later, when making general predictions.

3.3.2 Modelling the start time (AKA time-delays)

I thought it would be useful to be able to predict the start time on its own, in addition to the end times. For example, if this data were recorded for something turning on an off, it would be useful to be able to model the delay between these "ons" and "offs" (or simply, when the "ons" occur).

For this purpose, I plotted starting times alone against the experiment number, such that it can be predicted that the n-th starting time would be a certain value.

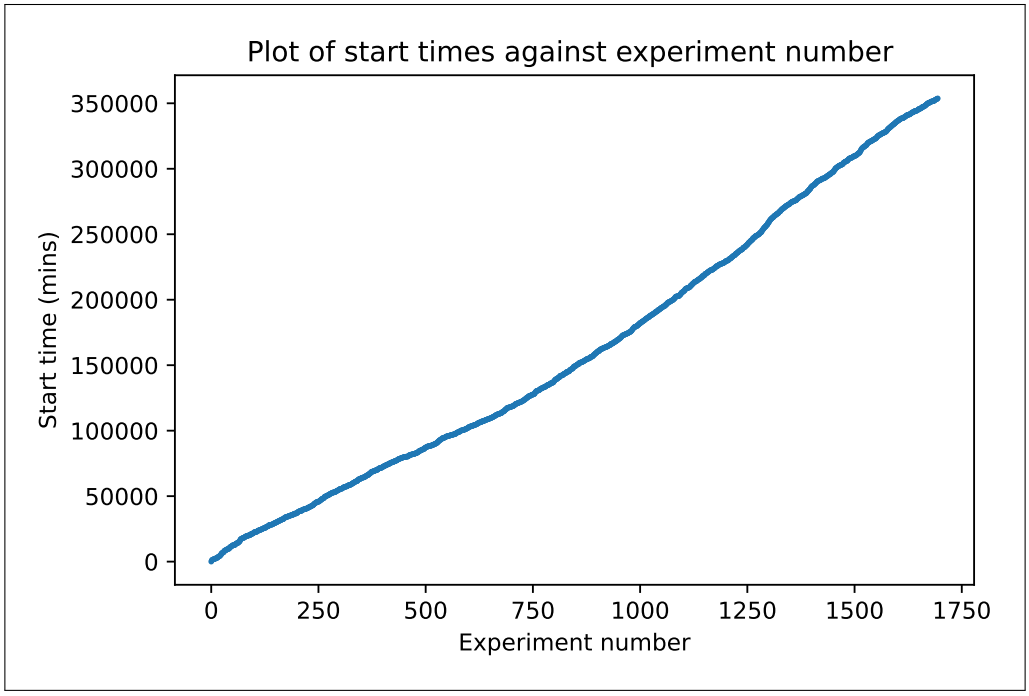


Figure 21

It is not immediately obvious what model would fit this data. In the interests of applying the simplest model possible as a starting point, I decided to only explore polynomial trends.

As a first point of call, I applied the Bayesian Information Criterion to assess which polynomial model would be ideal.

The formula for BIC is given by:

$$BIC = D \ln N - 2 \ln \hat{\mathcal{L}}, \tag{64}$$

Where D is the number of parameters in the model, theta are the parameters and $\mathcal{L}$ is the likelihood.

To engage in the linear algebra approach to point estimating the parameters, I assumed that there was some uncertainty associated with the starting times (again, despite the lack of given uncertainties in the data). I assumed that both start and end times would share similar uncertainties. To be safe, I still only imposed their uncertainties to be a fifth of the uncertainties previously established in the previous sampling method (σ).

Applying the likelihoods for the different polynomial models, the BIC for each number of parameters is plotted below.

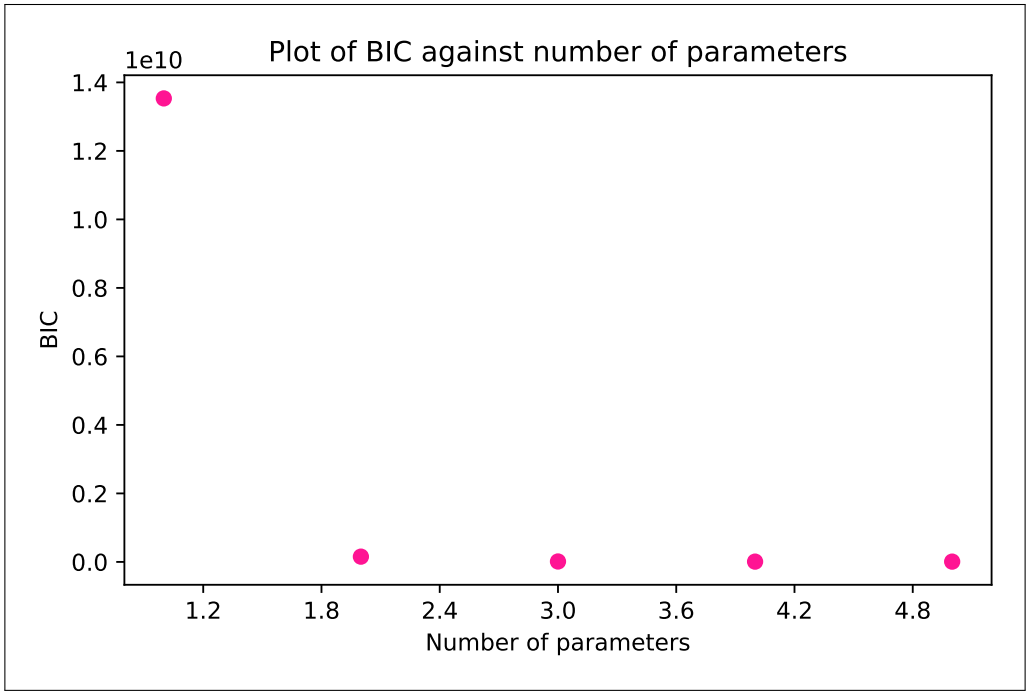


Figure 22

The number of parameters with the lowest BIC (therefore the optimal model) was 5 parameters (a quartic polynomial), with the BIC value $\sim 8.5 \times 10^6$.

While this does provide some guidance, we have found this BIC value using an arbitrarily chosen uncertainty in the data. Further, we do not have a grasp of what model may be more appropriate in this case (one with more or less parameters). In the interests of simplicity ("always start with the simplest model!") I first conducted inference using a linear model, and confirmed this was likely not correct.

The BIC values for a quadratic, cubic and quartic model were $1.25587245e{+}07, 8.49188002e{+}06, 8.48986786e{+}06$ respectively. I first noticed that the cubic and quartic BIC were nearly the same (differing by $\sim 0.02\%$), meaning I was more inclined to determine the cubic model than the quartic (simper is better). Further, the quadratic BIC was sufficiently close for me to consider it, and determine its suitability as a model accordingly.

LINEAR MODEL

Using a similar model to the previous inference, just for completeness, I used a linear model on the data for start times vs measurement number.

Using the curve_fit function, the initialisation values for the parameters m (gradient) and c (offset) were:

$$c = -13908.21 \text{ with a standard deviation: } 531.853 \tag{65}$$

$$m = 208.63 \text{ with a standard deviation: } 0.5437188688468032. \tag{66}$$

The stan model for the linear modelling of start time against measurement number is the following.

data {

int<lower=0> N;

vector[N] x;

vector[N] y;

}

parameters {

real m;

real c;

real¡lower=0¿ sigma;

}

model {

c uniform(-100000000,100);

sigma normal(0,1000);

y normal(c + x * m, sigma);

}

The chain plots for this linear model are plotted below.

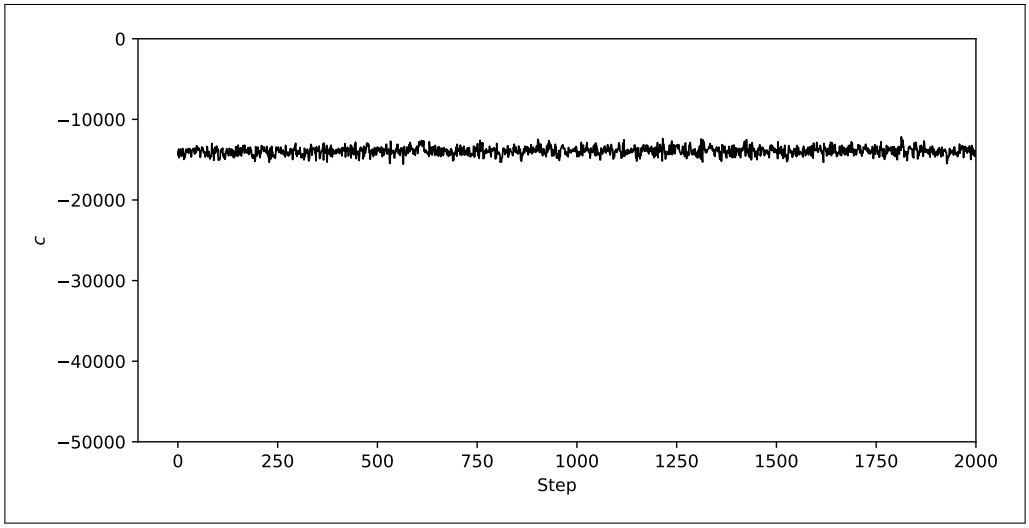


Figure 23

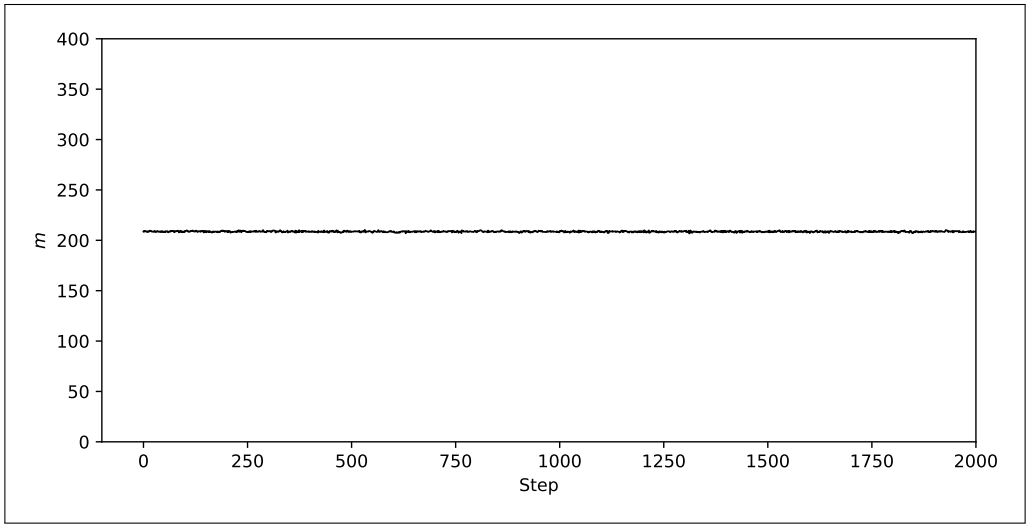


Figure 24

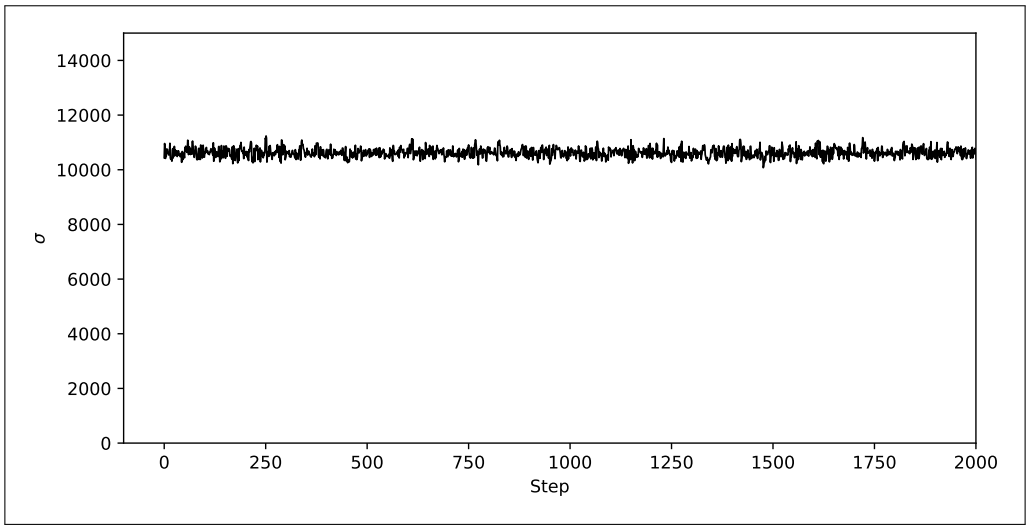


Figure 25

These chains all appear to converge, supported by their $\hat{R}$ values close to 1.

The resultant mean values for the parameters and their standard deviations, derived from the chain plots, were:

$$c = -13921.1376 \pm 507.8847 \tag{67}$$

$$m = 208.6395 \pm 0.5173 \tag{68}$$

$$\sigma_T = 10616.492 \pm 170.255. \tag{69}$$

The extended prediction for the starting times using this model is plotted below, with the actual data. The plot is filled within the first two standard deviations σ_T.

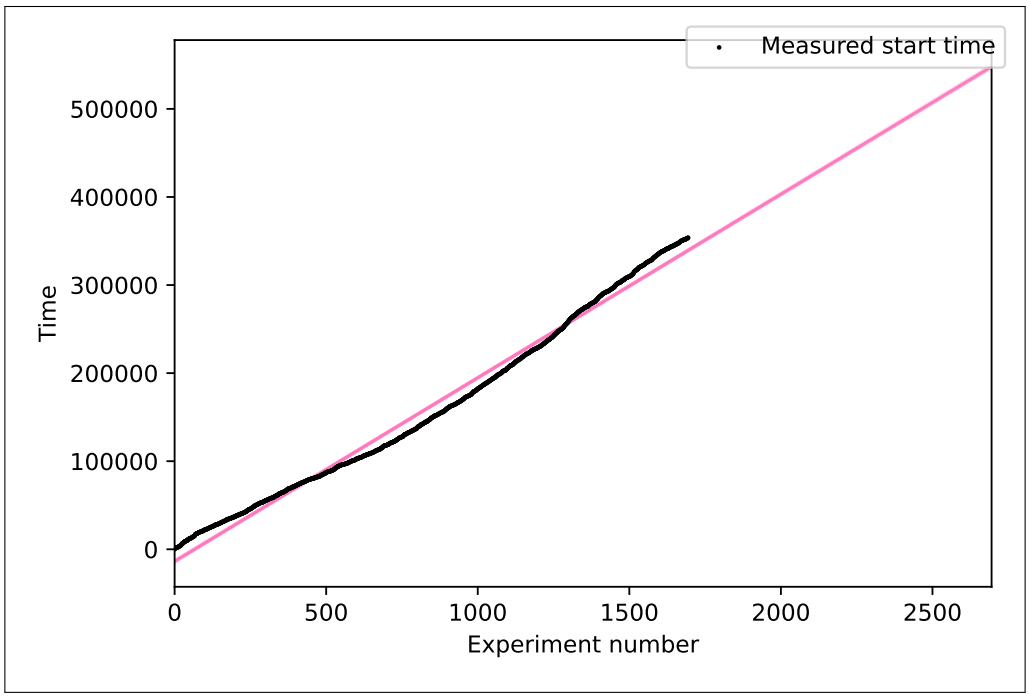


Figure 26

Visually, we can inspect the graph and conclude that our BIC value was rightfully higher for this model, as it fails to incorporate the deviations from a straight line that are clearly not due to noise. Rather, the relationship between start time and measurement number is clearly not linear. This is particularly notable when considering the sheer number of points that have followed a similar curve - if this were simply an artefact of the normal distribution of the data probability, one would expect the spread of data to be more random, rather than following a curved line. Therefore, it is not accurate to chalk those trends up to a large standard deviation, and the actual model is more likely to be a higher order polynomial.

QUADRATIC MODEL

Similar to the linear model, the quadratic likelihood is simply:

$$p(y_i, \sigma_y, m, c|x_i) = \prod_{i=1}^{N} \left(\frac{1}{2\pi\sigma_T^2} e^{-\frac{\left(y_i - (ax_i^2 + bx_i + c)\right)^2}{2\sigma_T^2}} (1+m^2)^{-3/2} \right). \tag{70}$$

I again used curve_fit to initialise the values, and implemented the following stan model to conduct sampling for the parameters.

The stan model for the quadratic model is as below.

data {

int<lower=0> N;

vector[N] x;

vector[N] y;

}

parameters {

real a; real b; real c; real<lower=0> sigma; }

model {

sigma normal(0,1000);

y normal(a * x .* x + b * x + c, sigma);

}

The chain plots for parameters a, b, c and σ are given below. There all appear to converge, and have $\hat{R}$ values around 1.00.

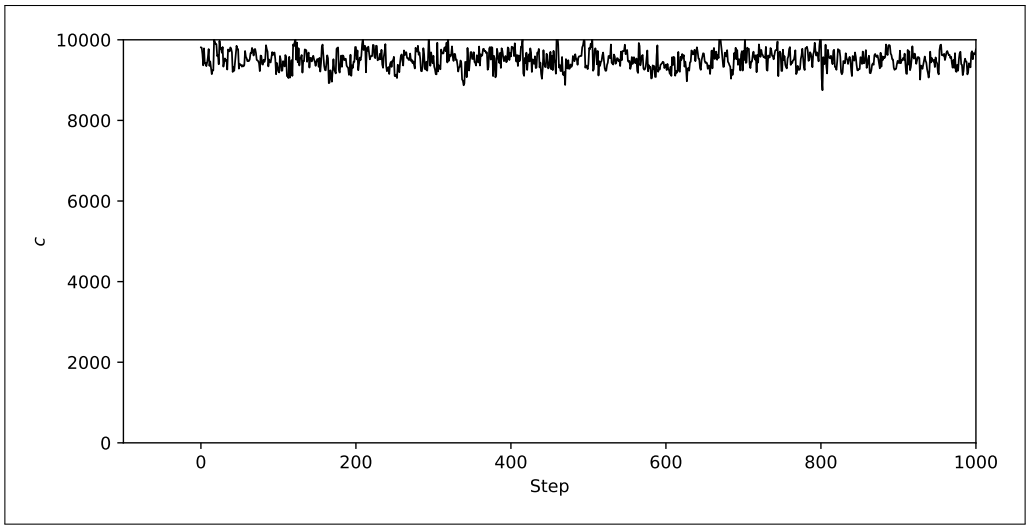


Figure 27

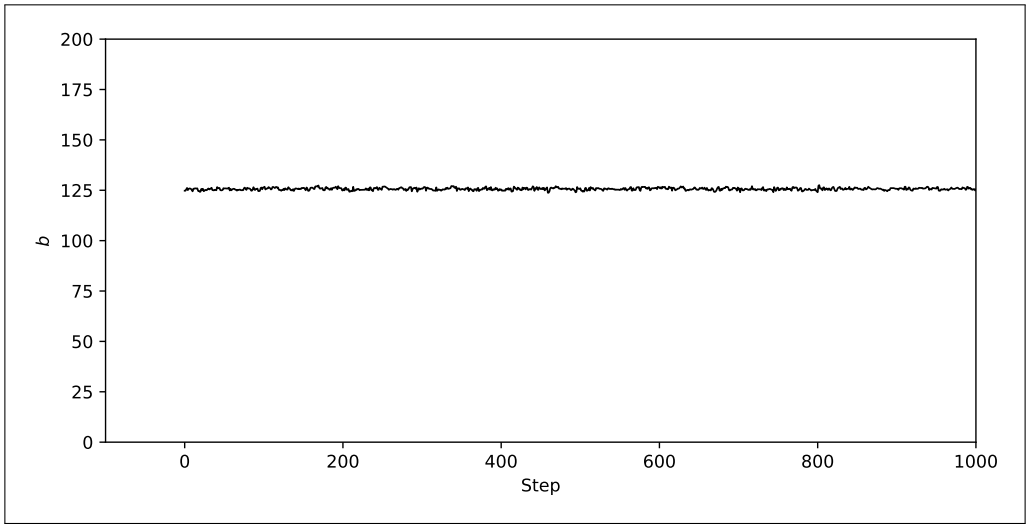


Figure 28

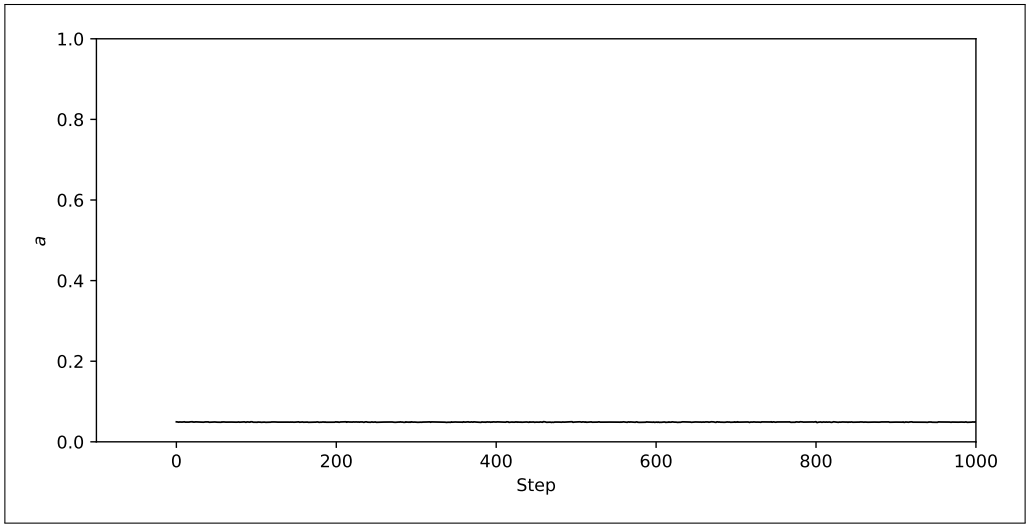


Figure 29

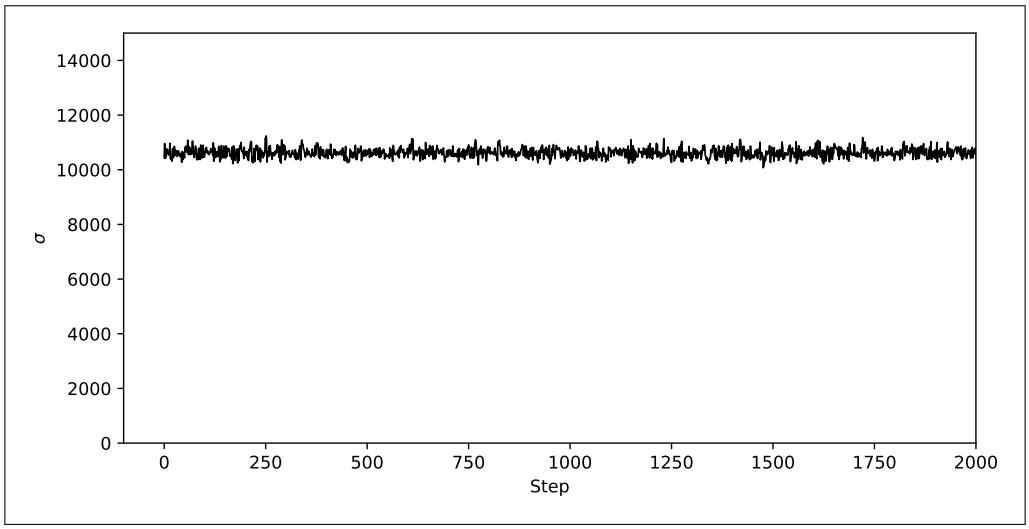


Figure 30

The final parameter values (their mean and standard deviations) were:

$$a = 0.0489911638 + / - 0.00035053 \tag{71}$$

$$b = 125.6456955 + / - 0.610032796 \tag{72}$$

$$c = 9503.207445 + / - 221.7859999 \tag{73}$$

$$\sigma_T = 3122.110465 + / - 54.0340. \tag{74}$$

(I note that while σ_T appears large, this is still on the order of the "thickness" of the depicted data line as the axes scale is very large. I have no reason for why I chose to do this in minutes, I made that decision at the beginning and never got around to changing it).

I plotted the quadratic model using the parameters' mean values, filled within **two** standard deviations, with the original data.

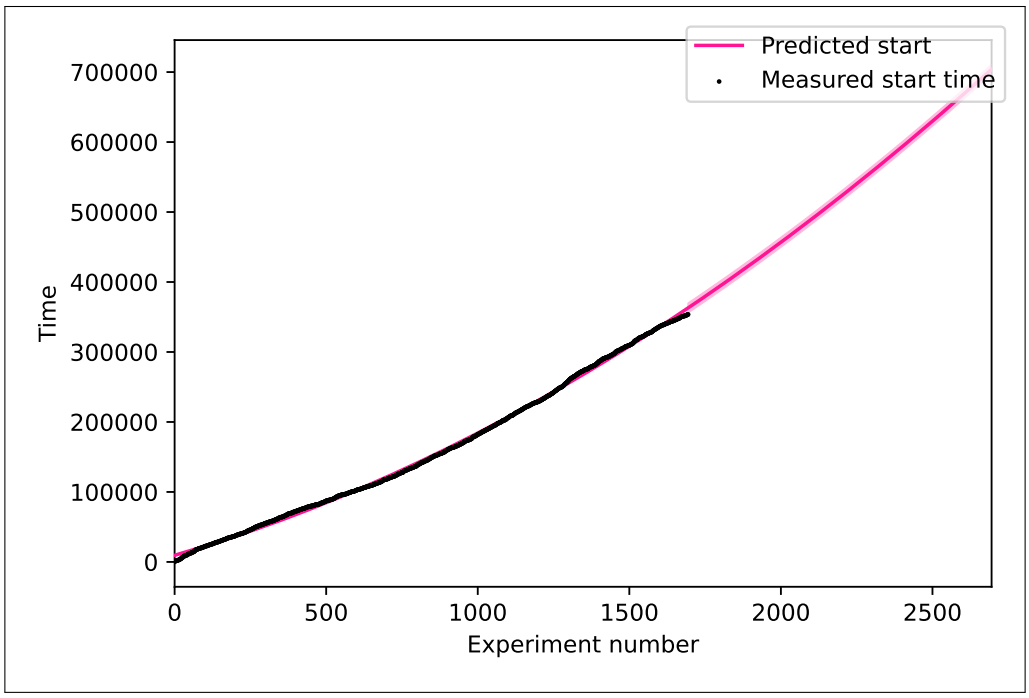


Figure 31

This fit appears much better than the linear fit. Further, the standard deviations appear to incorporate the full "thickness" of the line, corresponding to the smaller-scale variance. However, there are two places the data seems to coherently deviate from the quadratic trend - at the beginning, and at the end of the data. Therefore, there is cause to explore a cubic trend, although a quadratic model appears to be in good standing to be the generative model for this data.

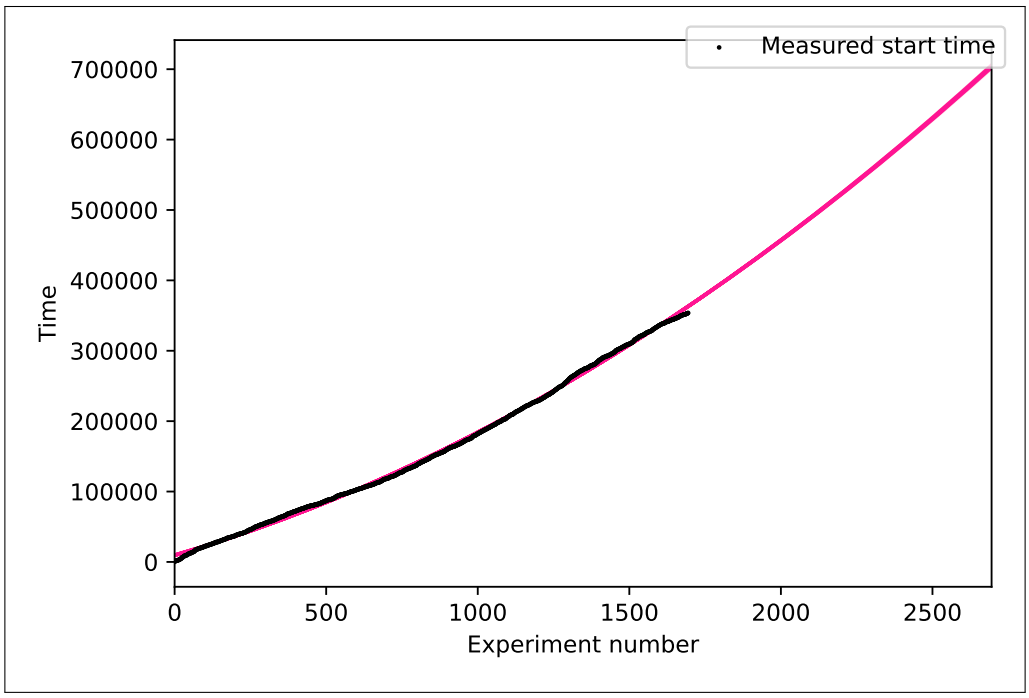


Figure 32

For this reason, I only used curve_fit to explore the cubic model.

CUBIC MODEL

I used the following cubic model, and applied it to the data using scipy.curve_fit to find the parameters.

$$y = ax^3 + bx^2 + cx + d. \tag{75}$$

The resultant parameters were:

$$d = 8595.61758733002 + / - 301.5555543832021 c = 132.1208993314244 + / - 1.54210069083153 16 b = 0.03943043034902863 + \tag{76}$$

The result is plotted in pink below.

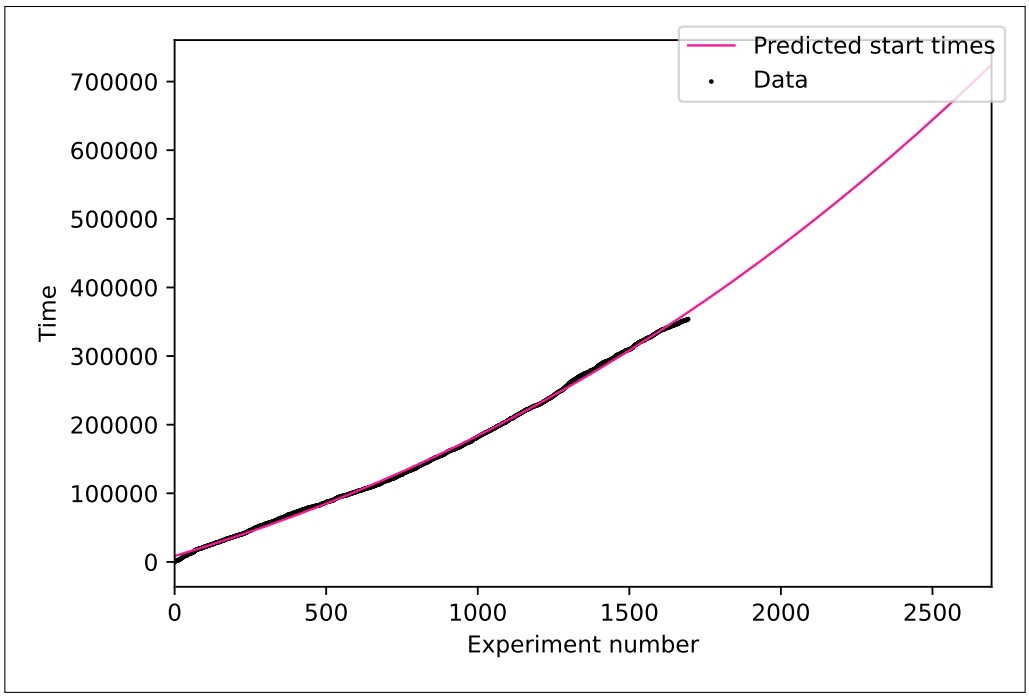


Figure 33

When compared with Figure 31 and 32, the cubic model does not seem to incorporate any major differences to the quadratic model - in fact, the final value for "a" is relatively small, seeming to prefer a quasi-quadratic fit.

Without further information about the source or expected trend of the data, this model does not seem to provide a big advantage over the quadratic model. Therefore, I would choose the model the data with the quadratic model, in the interest of sticking to the simplest model.

3.4 Final predictions

Through the above methods, we have developed a quadratic model for the starting time as a function of "measurement number", and a linear model for the ending time as a function of starting time. Therefore, both of these can be predicted for future measurements.

It is easier to explain these predictions in the context of something turning "on" at the start time, and "off" at the end time. Using the quadratic model, we are able to predict when it will turn "on" next (ie the start time of the following "measurement number". The general trend is that it will increase quadratically.

A visualisation of this prediction is given below, using the method of plotting 100 lines with parameters taken from random points in the chain plots. This was done to provide a representation of the spread of possible predictions, which was possible due to the sampling method.

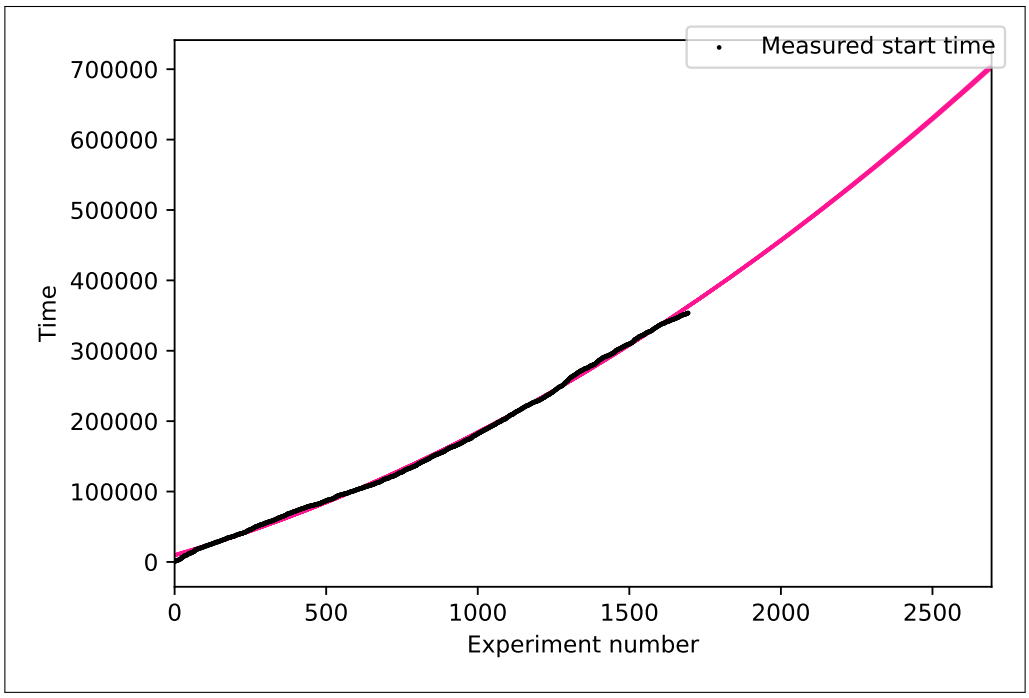


Figure 34

I do strongly note that there seems to be a deviation from this predicted increase near the end of the data. If this is due to an upcoming down-turn in the true data values, a different model may very well be more accurate. However, as we do not have insight into this "future" data, the quadratic model suffices.

Finally, from this, we can also predict when the thing will turn "off", as this value is linearly dependent on when it turns "on". This prediction for end times is plotted below, again as a function of measurement number, and again using the method of picking 100 sets of parameters from random points in the sampled chain plots. This is effectively predicted to be some constant duration after whatever the starting time was.

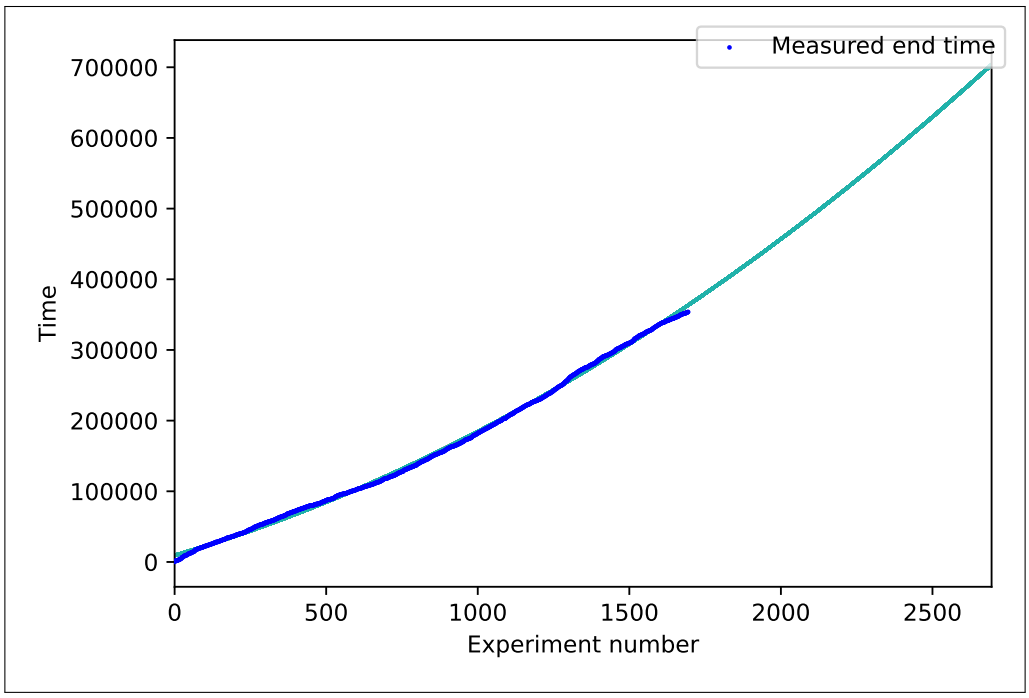


Figure 35

3.5 Further ideas

I was limited in this question by time (and skill), but there are a few things I noted that I would've liked to attempt to model further.

Firstly, there appears to be a periodic element in the data, as shown by the "zoomed in" plot in Figure 17e. There are repeated "bumps" to the start (and therefore end) times. I chose to discount these "bumps" as being encompassed in the uncertainty of a linear model. However, there may be scope to explore periodic trends. For example, inspired by Question 2, one could implement Gaussian processing to model this periodicity.

Another formulation of this artefact is by considering a histogram of the number of "starting times" for each hour of the day. This is shown below, and on first inspection, it appears possible this could be fitted by some periodic function.

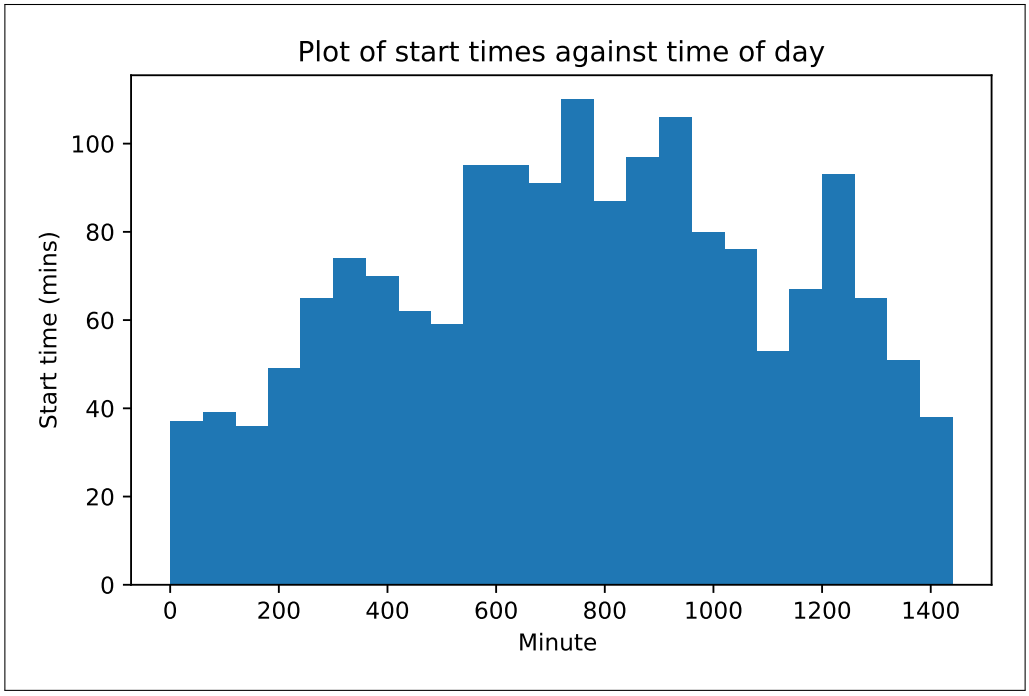


Figure 36

In addition, I plotted the total number of starting times (so, measurements) per day in the data, as seen below.

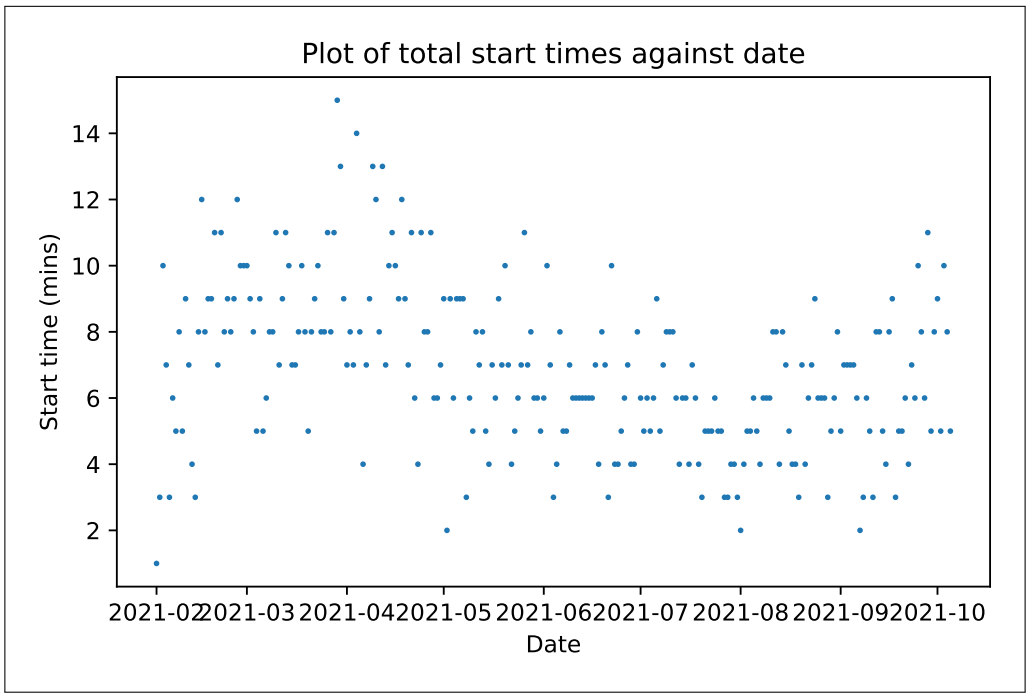


Figure 37

The number of start times per day appears to be experiencing a slight decreasing trend over the course of the months. This could be another modelled trend, to predict the number of measurements that would occur

in future days.