

# Viscosity project summary\_Section 2\_ML

---

## Table of content

<b>Standard 398.5 cP</b>	<b>2</b>
Set: Full	2
Set: Full (unorderedT)	9
Set: 1	14
Set: 1 (unorderedT)	21
<b>Standard 1275 cP</b>	<b>25</b>
Set: Full	26
Set: Full (unorderedT)	31
Set: 1	36
Set: 1 (unorderedT)	41

# Standard 398.5 cP

## Machine learning segment

---

### Set: Full

#### Observation of trends

Observation 1: GPR - scaling: Division and LIN - scaling: Multiplication and Division

Once an 'optimal' set of parameters are attained, it will double check one time to confirm, then continue to explore parameters. The 'confirmation' stage usually occurs in trial 5 or 6 of the 10 trials and it can continue to suggest parameters that deviate quite far from the 'optimal' set of parameters.

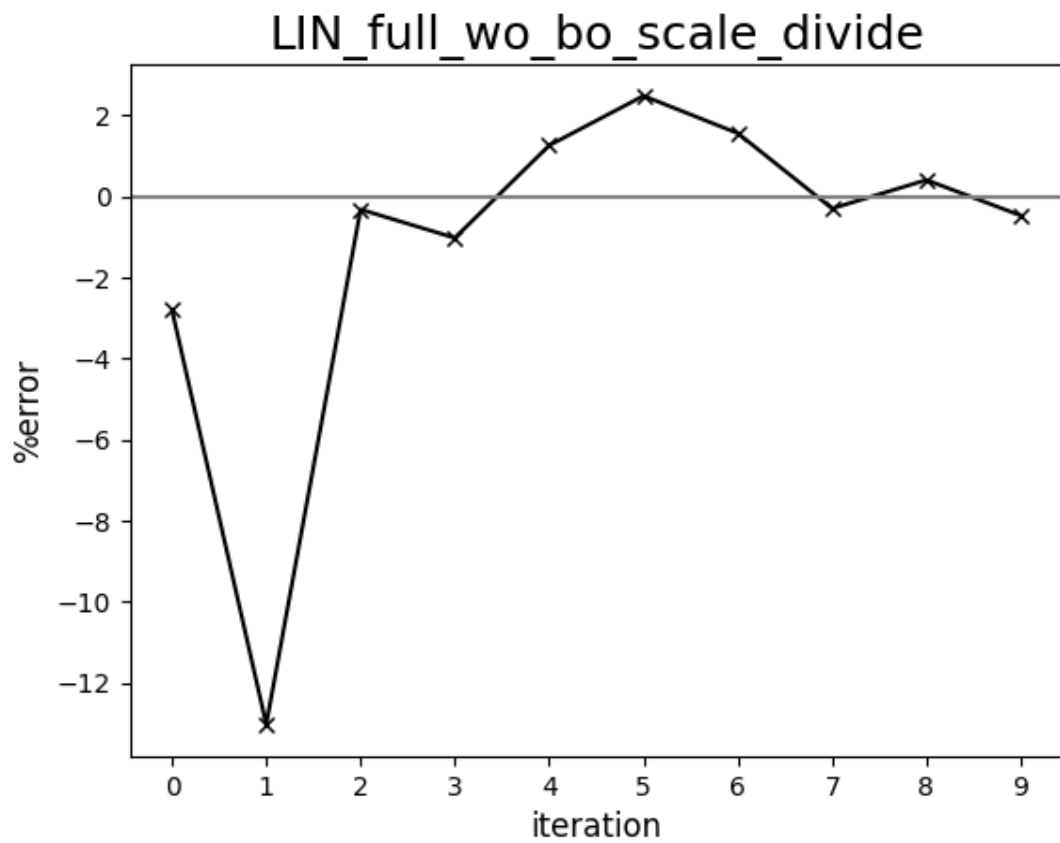
Observation 2: GPR - scaling: Multiplication

Once an 'optimal' set of parameters are attained, it will remain the same for the rest of the 10 trials. The 'optimal' set of data is usually attained in trial 5 or 6 too, similar to when division is used for scaling.

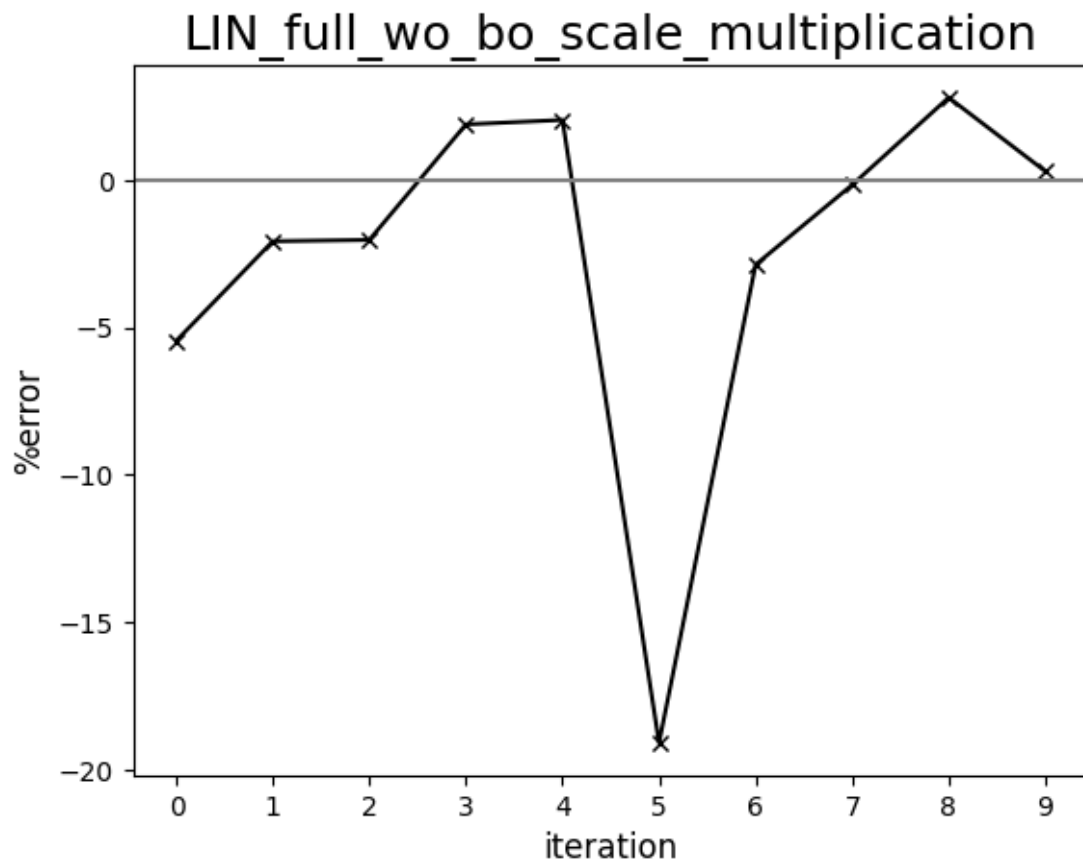
Observation 3: GPR - scaling: Without and LIN - scaling: Without

A converging trend is observed for percentage error. The first 2 trials' %error usually falls between 5-6% but it converges towards 0 from trial 2-9. However, trial 10 deviated from the consistency observed between trial 2-9. Unlike the trials with GPR-multiplication used in the scaling code (stays the same after 'optimal' parameters are found), the parameters continue to change slightly after the 'optimal' set of parameters are attained. For that, GPR without scaling can be said to be somewhere between when division and multiplication are used for the scaling parameters (continues to 'explore' moderately after 'optimal set' is found).

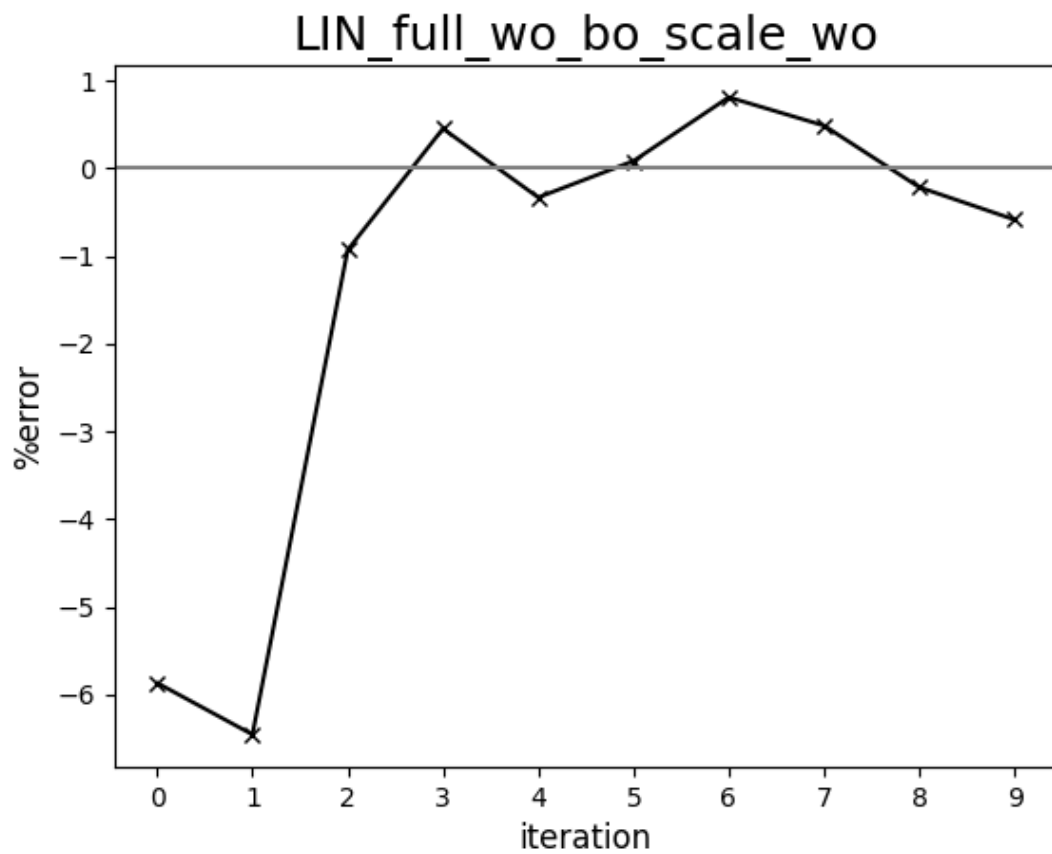
LIN scaling: division



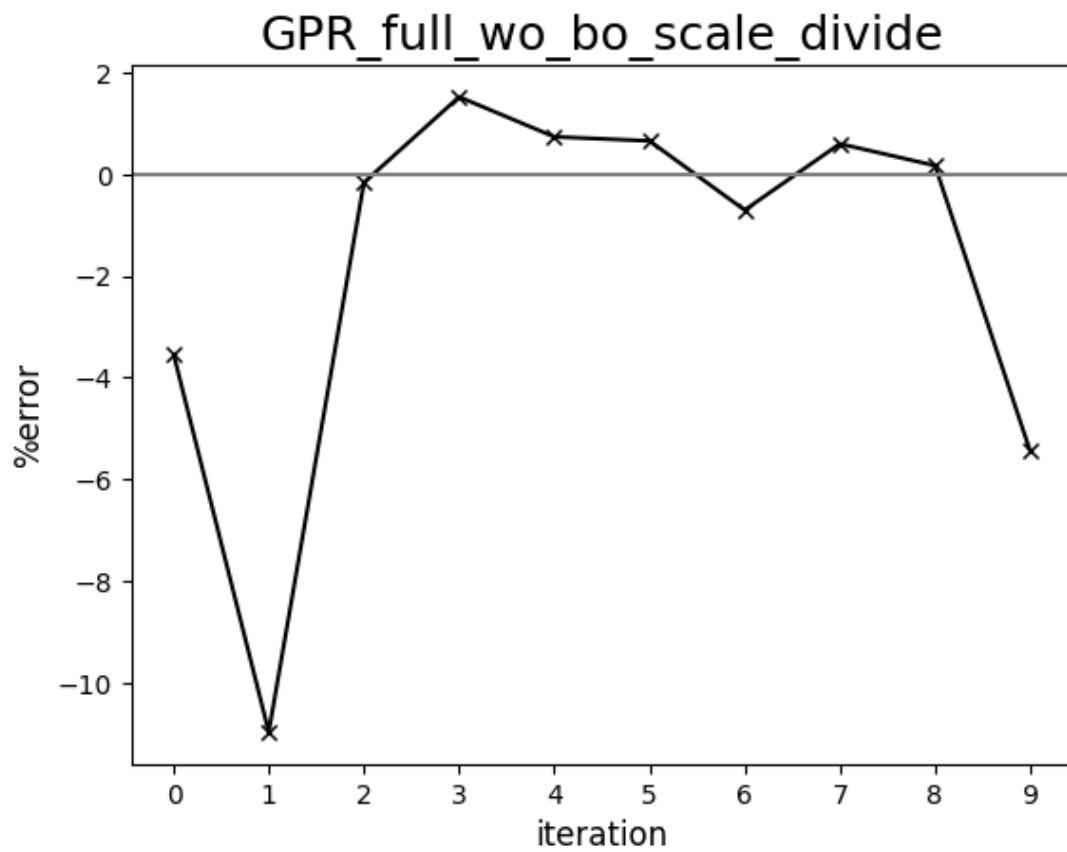
LIN scaling: multiplication



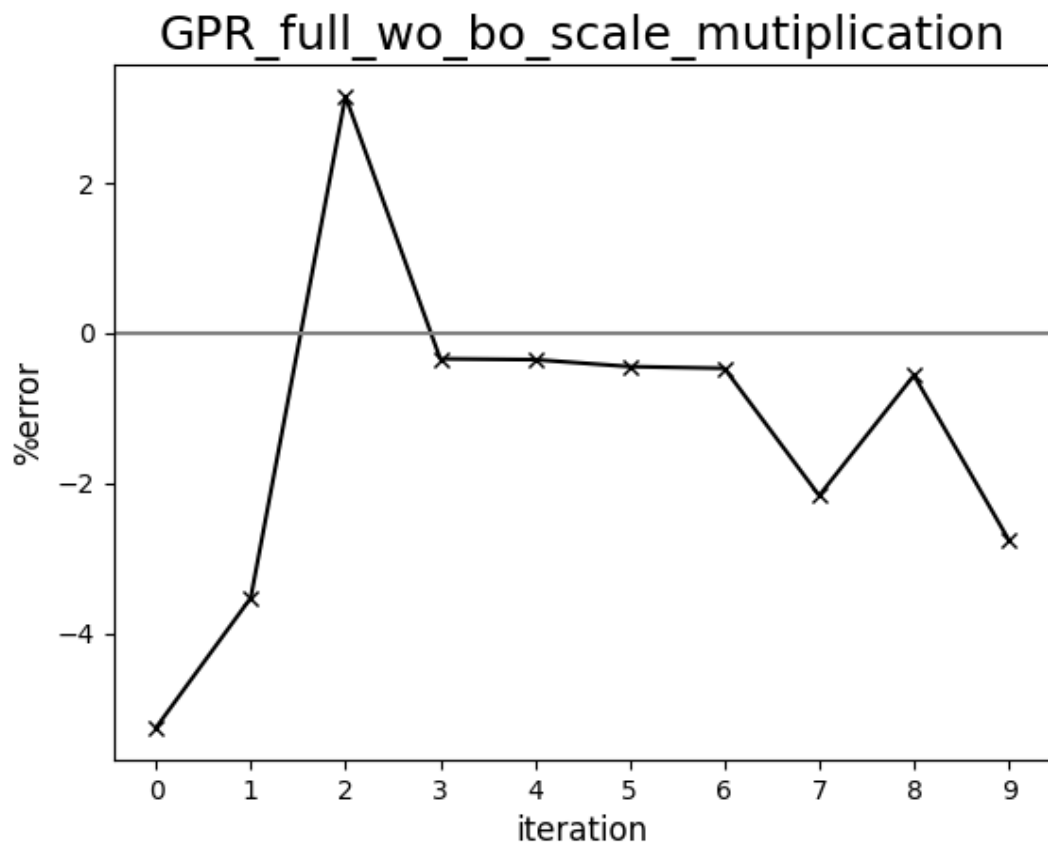
LIN scaling: without



GPR scaling: division



GPR scaling: multiplication



GPR scaling: without

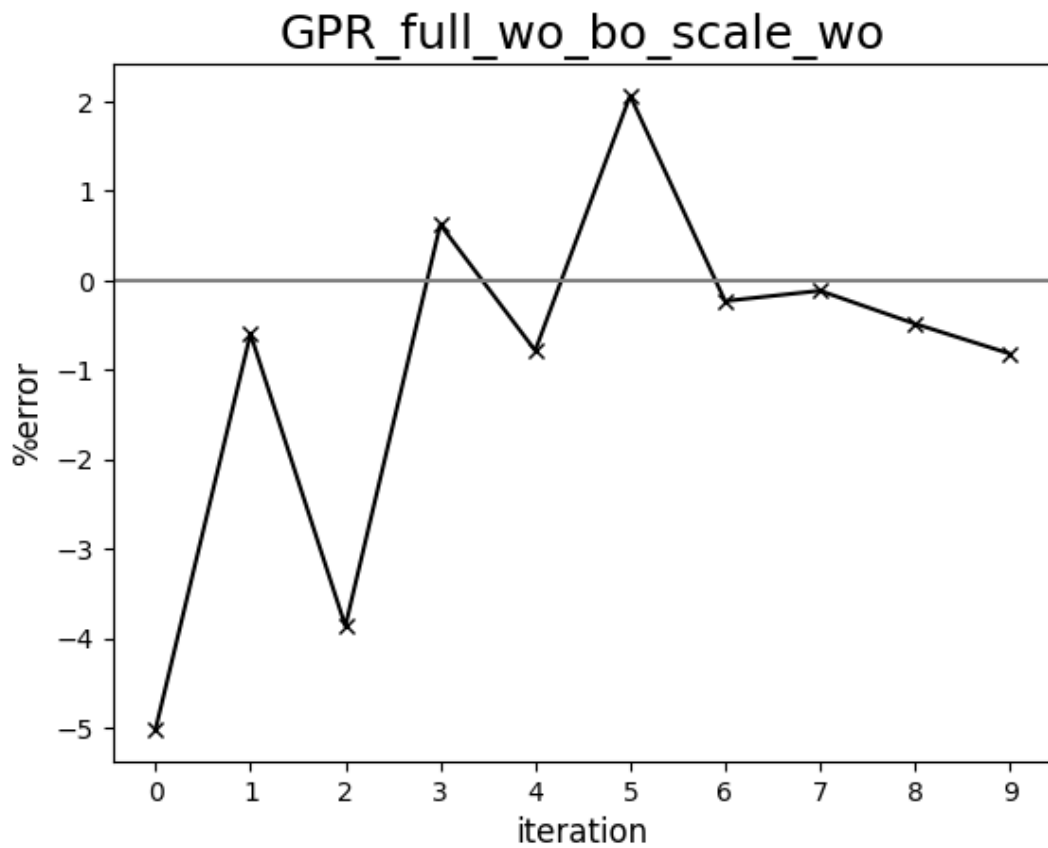
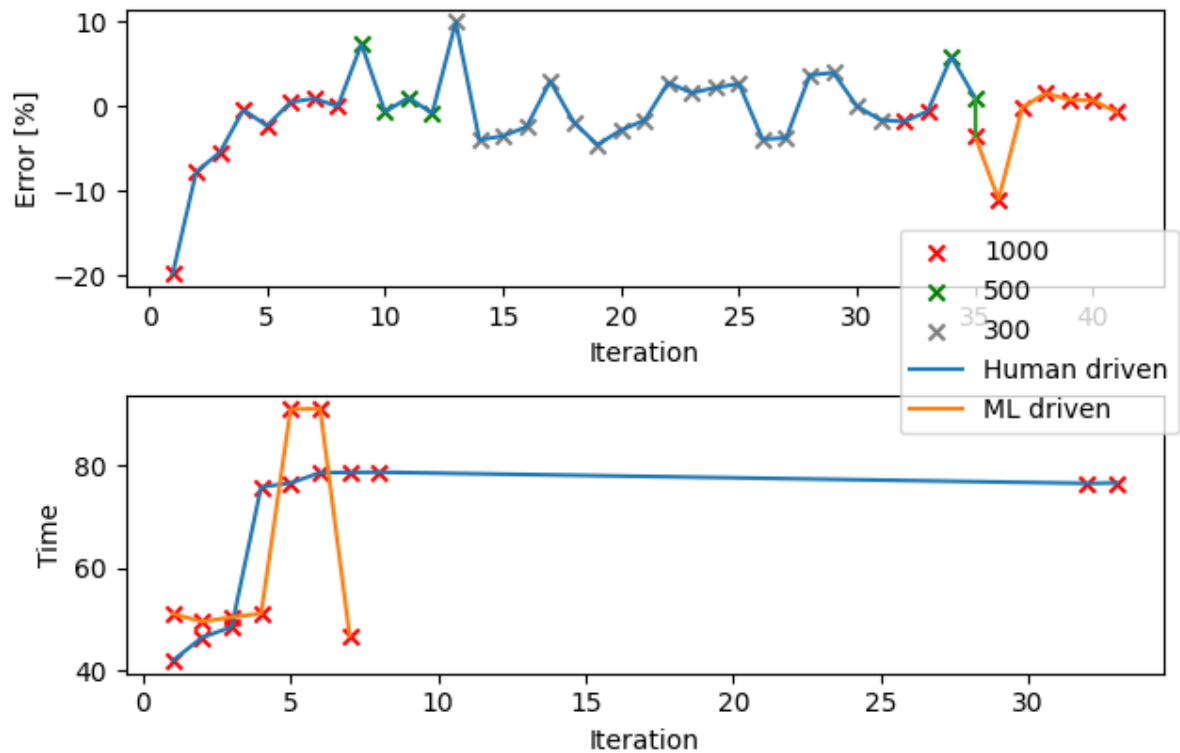
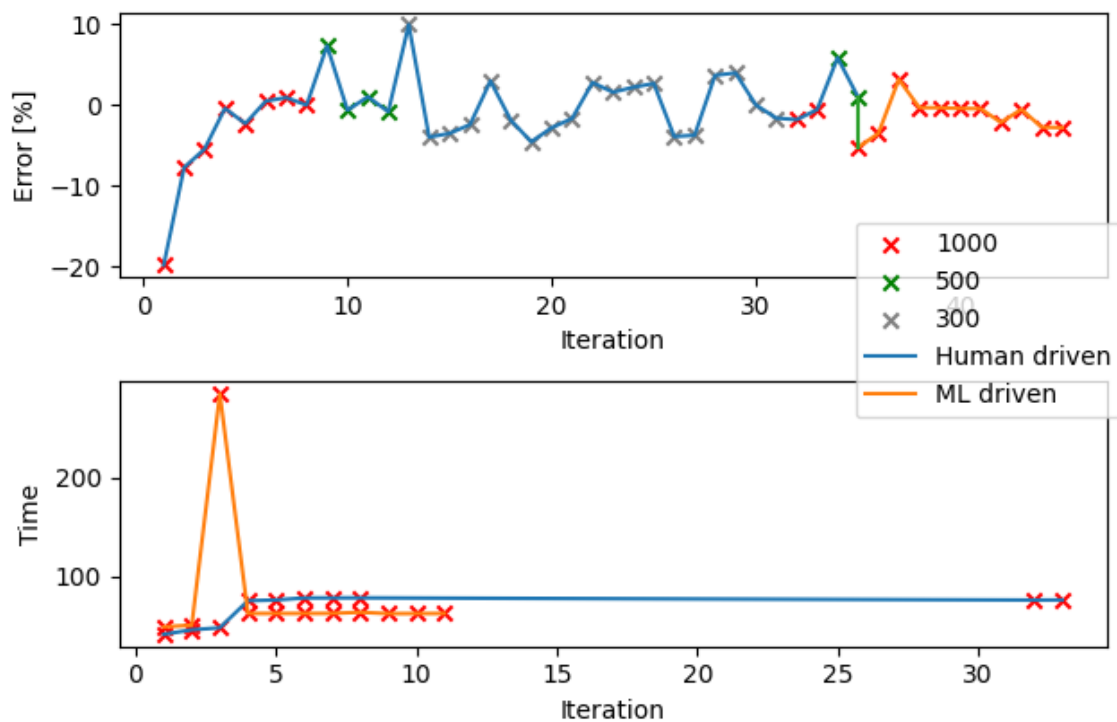


Diagram comparing human-driven and ML test trials (398 cP, set:full)

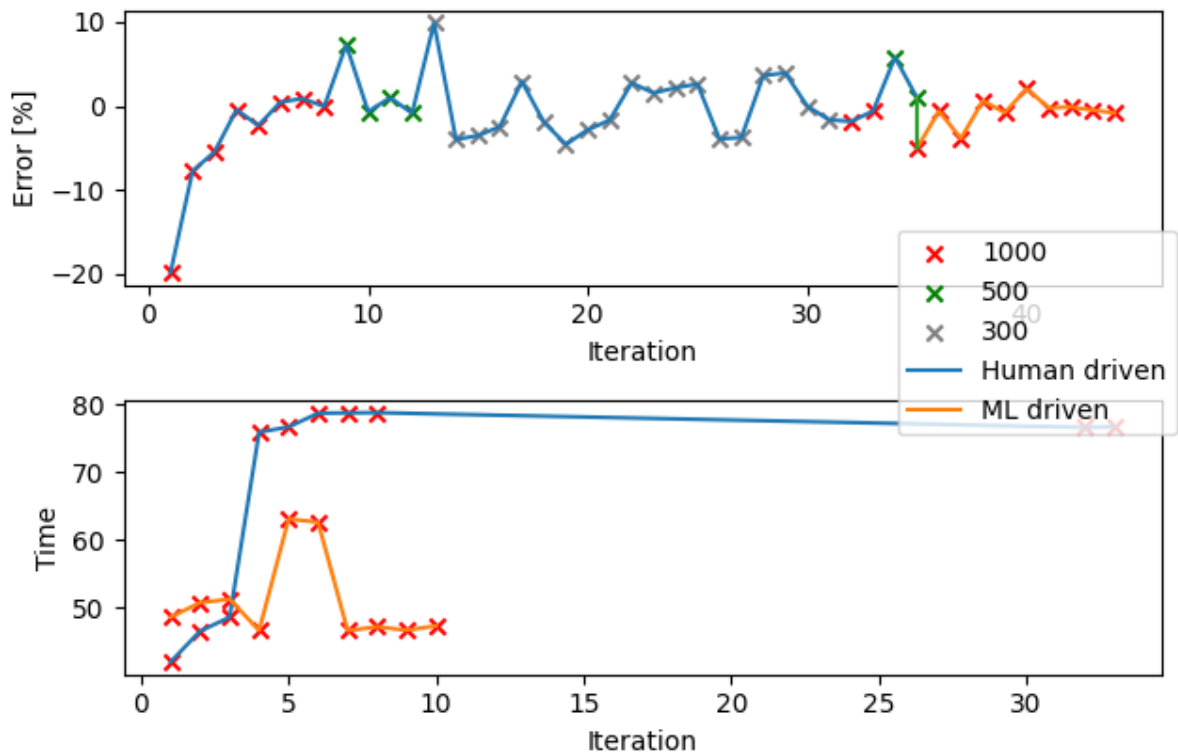
Gaussian Process Regression model with fast transfer penalization,  
trained with full data set



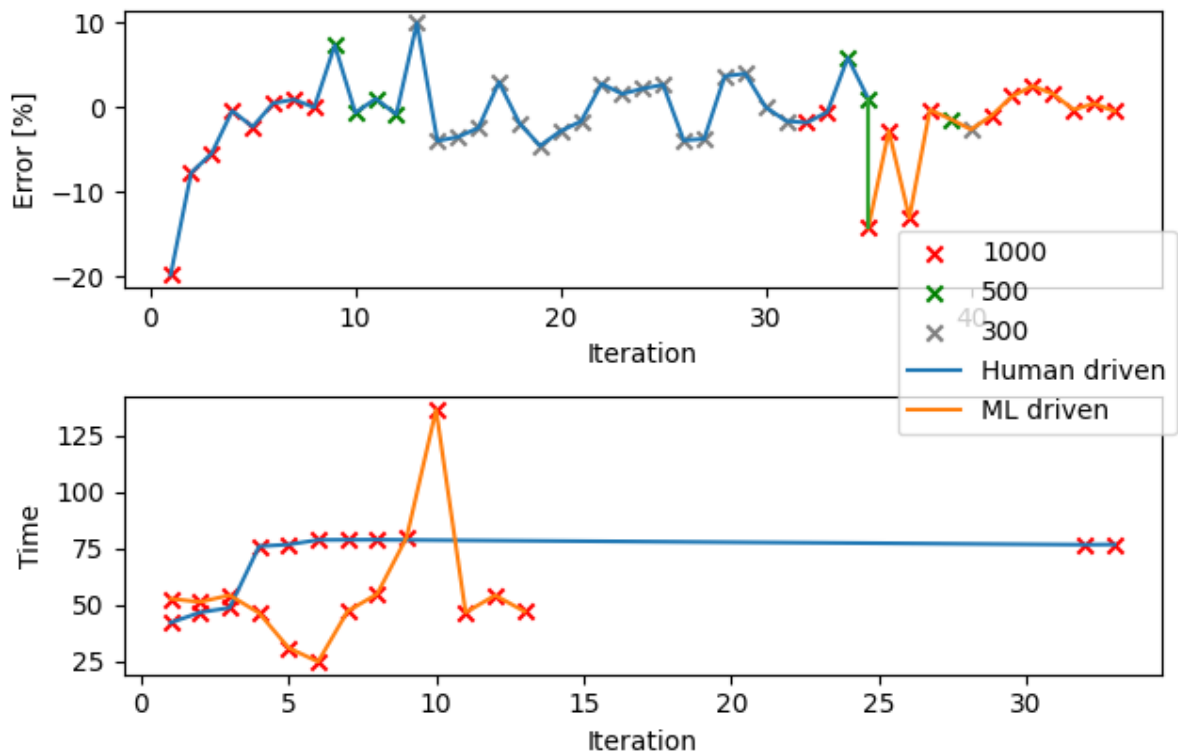
Gaussian Process Regression model with slow transfer penalization,  
trained with full data set



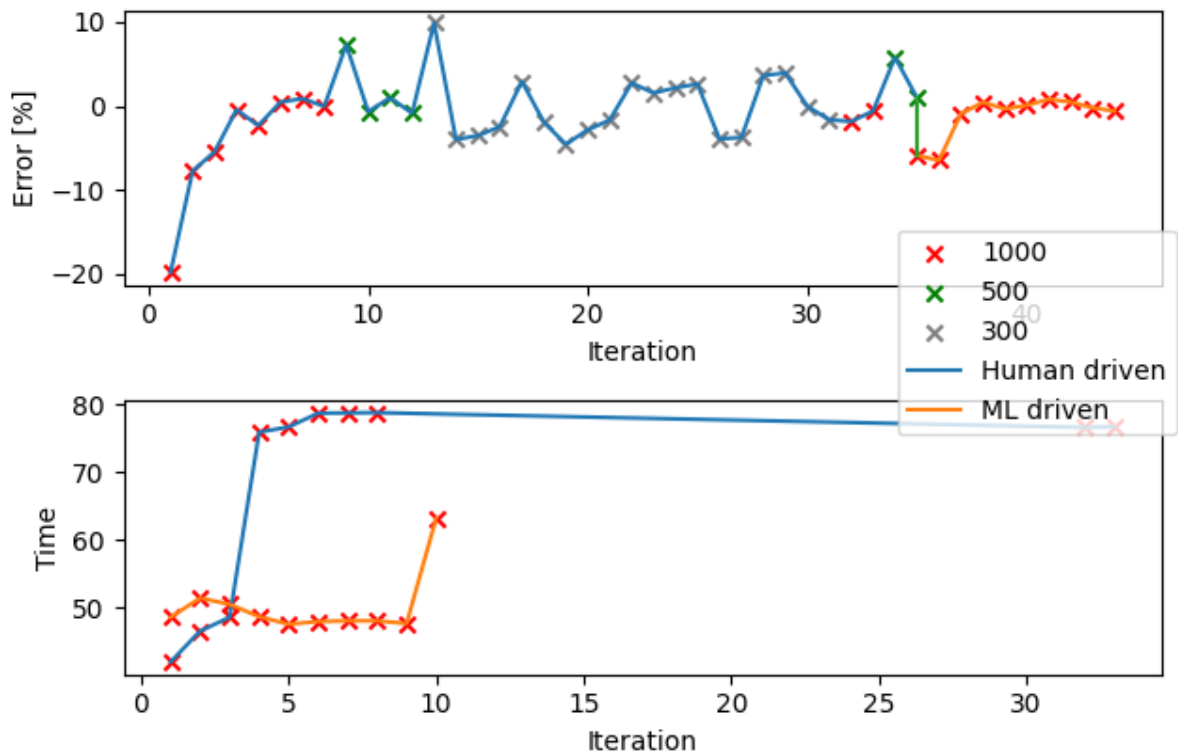
Gaussian Process Regression model with no penalization,  
trained with full data set



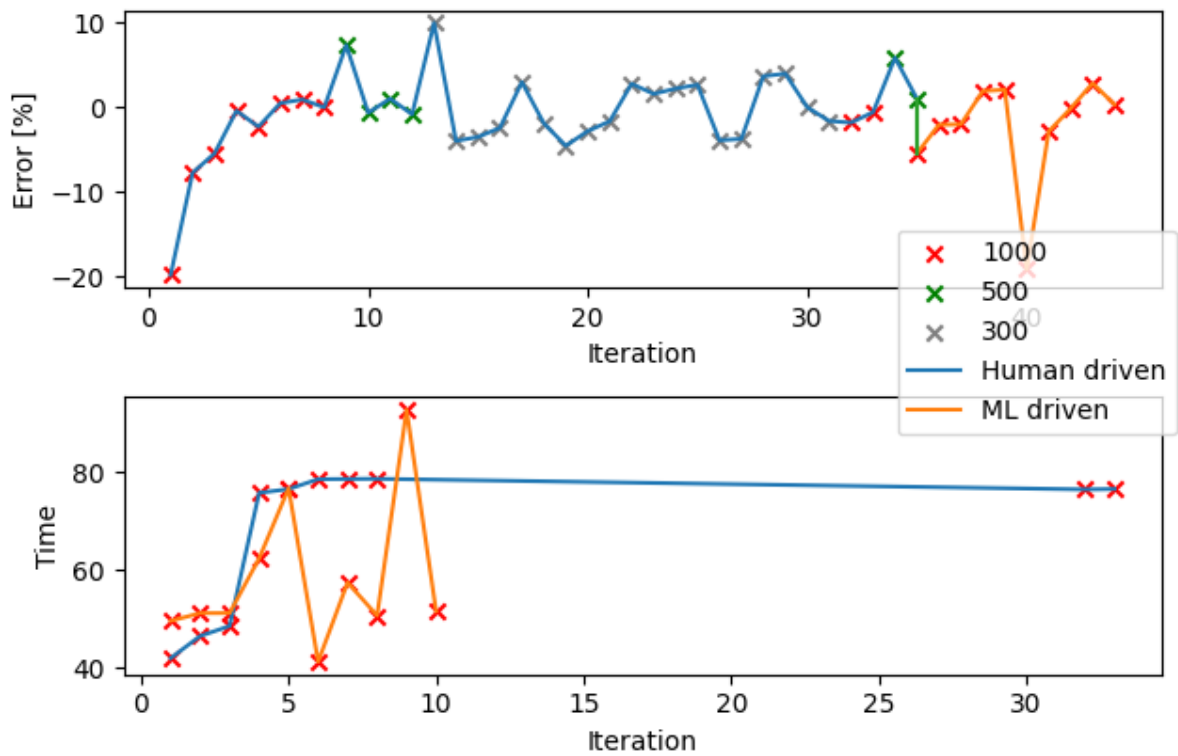
Linear Regression model with fast transfer penalization,  
trained with full data set



Linear Regression model with no penalization,  
trained with full data set



Linear Regression model with slow transfer penalization,  
trained with full data set





## Set: Full (unorderedT)

### Observation of trends

Observation 1: GPR - scaling: multiply

Some repetitions in the sets of parameters generated but it does show a decent amount of variation (set: 1 trials often have more repetition, GPR especially - sometimes all 10 sets of parameters are the same). However, the sets of parameters generated do not seem to align with the penalization used. When it is changed to multiply, it should penalise slower transfer times but this iteration has more long time trials as compared to GPR - none, which does not have the penalization. The percentage error is not so bad,

Observation 2: GPR - scaling: none

In general, fast and relatively accurate trials, around 7 out of 10 trials are within tolerance of 1000uL. The time taken for transfer is similar to the average time taken for human driven section trials (standard calibration).

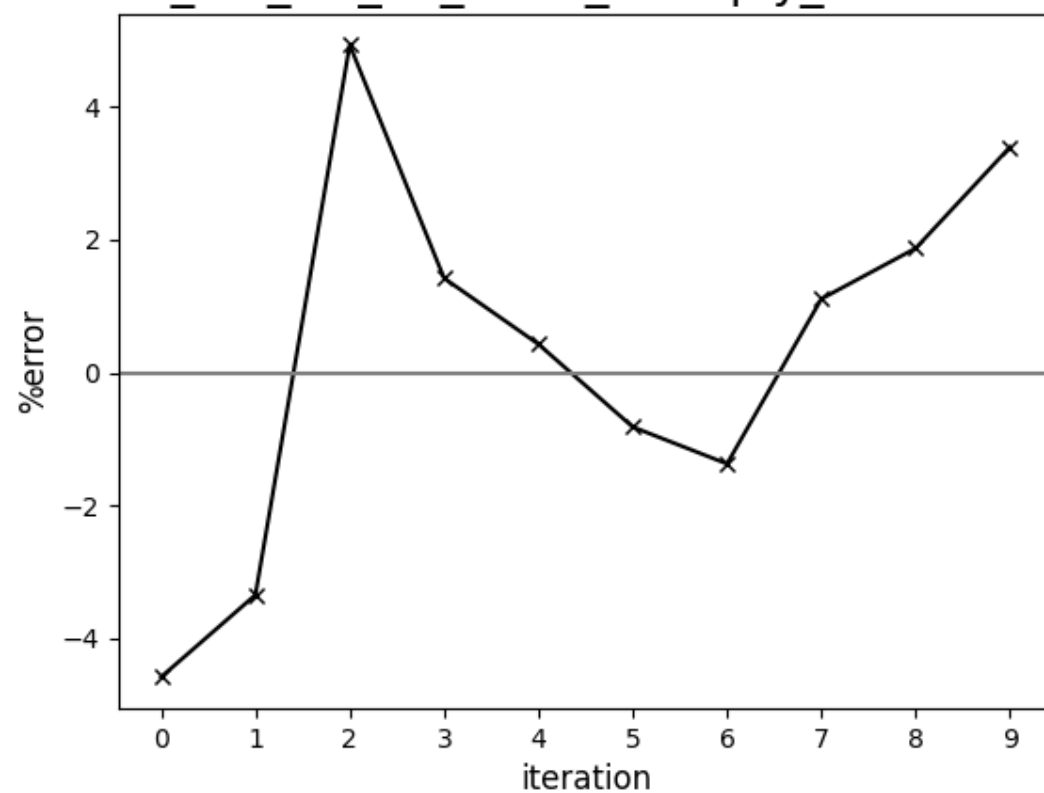
Observation 3: LIN - scaling: multiply

Slow transfer penalization seems to work on LIN. Overall short transfer time, not very accurate as seen in the data, only 5 to 6 out of 10 trials are within tolerance of 1000uL.

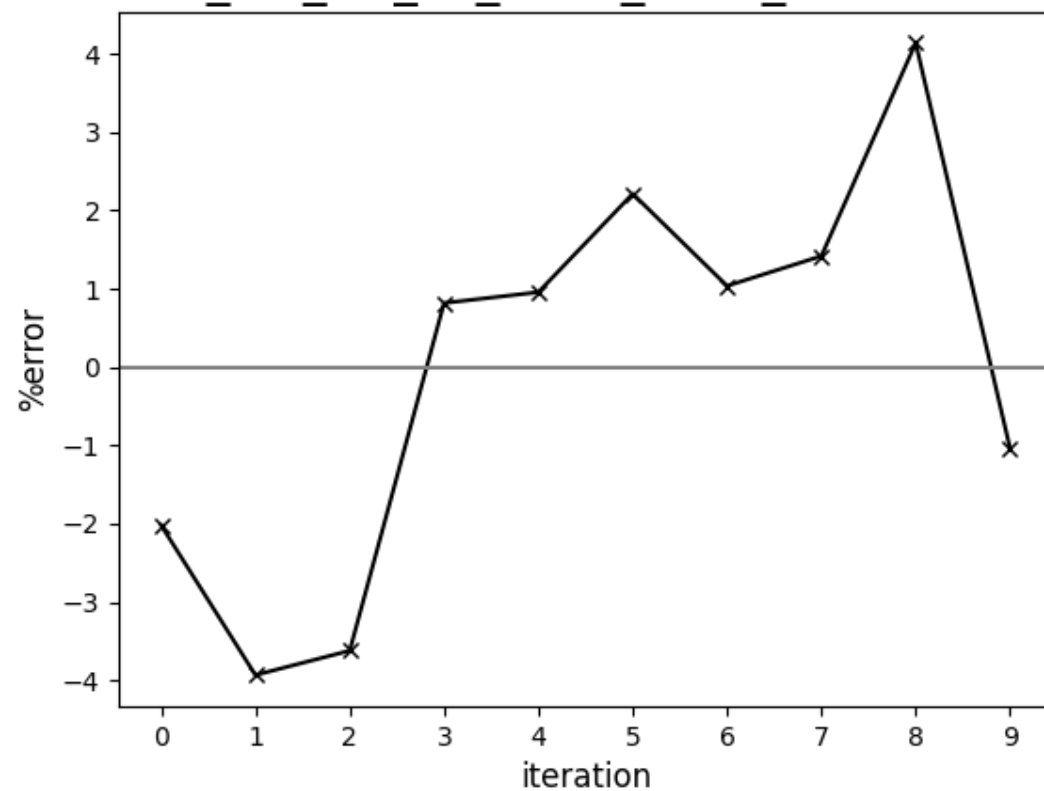
Observation 4: LIN - scaling: none

Accuracy is similar to LIN - multiply but much longer trials on average, a few trials even reached 200 seconds even though the standard calibration indicates around 80 seconds. However, at the last 5 trials, the model is able to discover sets of parameters that give very accurate transfer even though the 5 sets of parameters are very different, time ranges from 56s to 200s. (This shows that this iteration is prioritising low percentage error instead of shortening transfer times) Thus, one can suspect, in the long run this model can provide interesting parameters that are accurate.

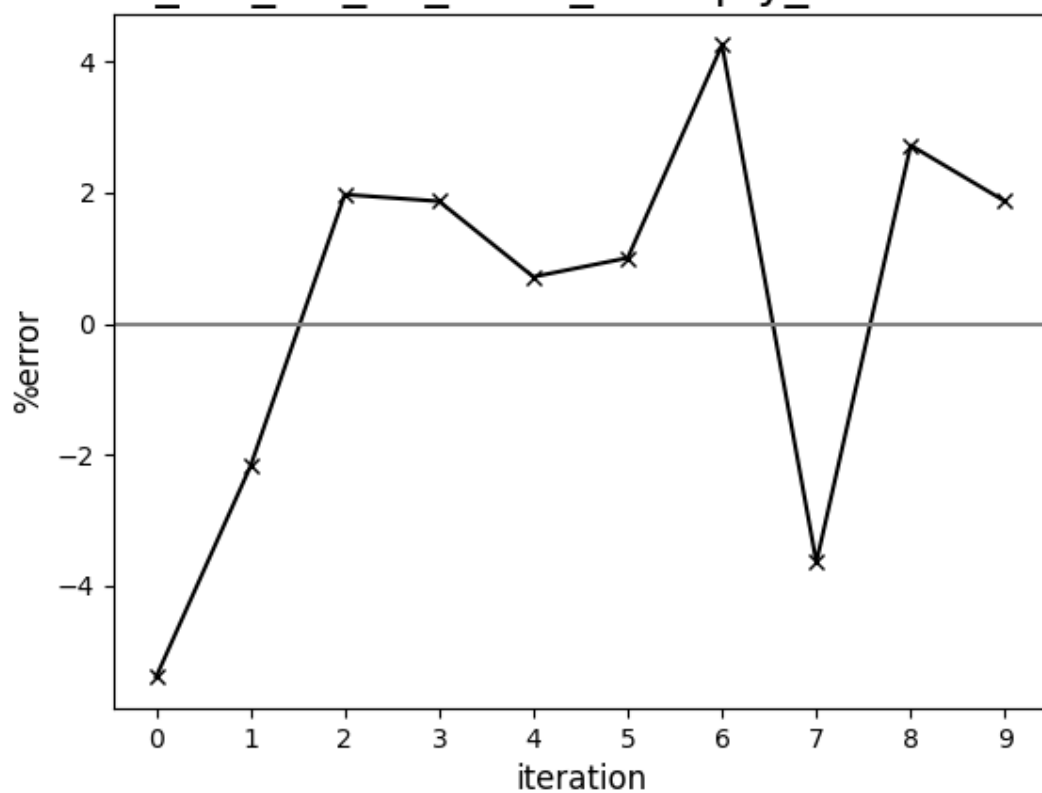
GPR\_full\_wo\_bo\_scale\_multiply\_unorderedT



GPR\_full\_wo\_bo\_scale\_none\_unorderedT



LIN\_full\_wo\_bo\_scale\_multiply\_unorderedT



LIN\_full\_wo\_bo\_scale\_none\_unorderedT

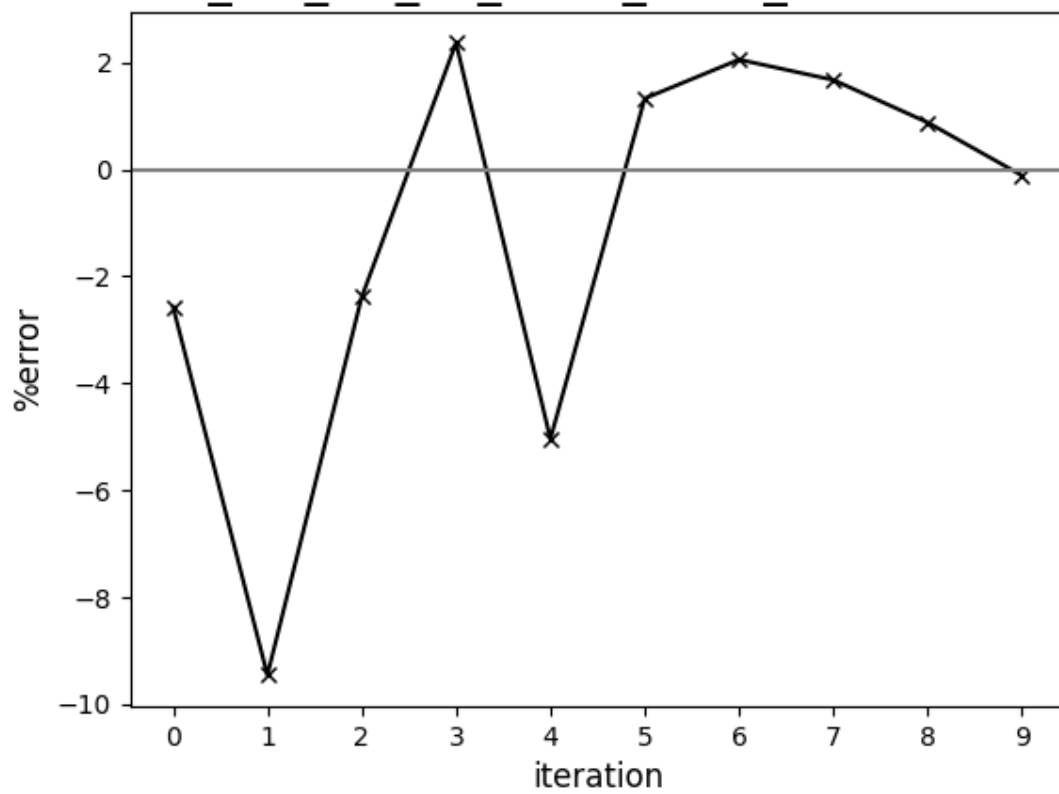
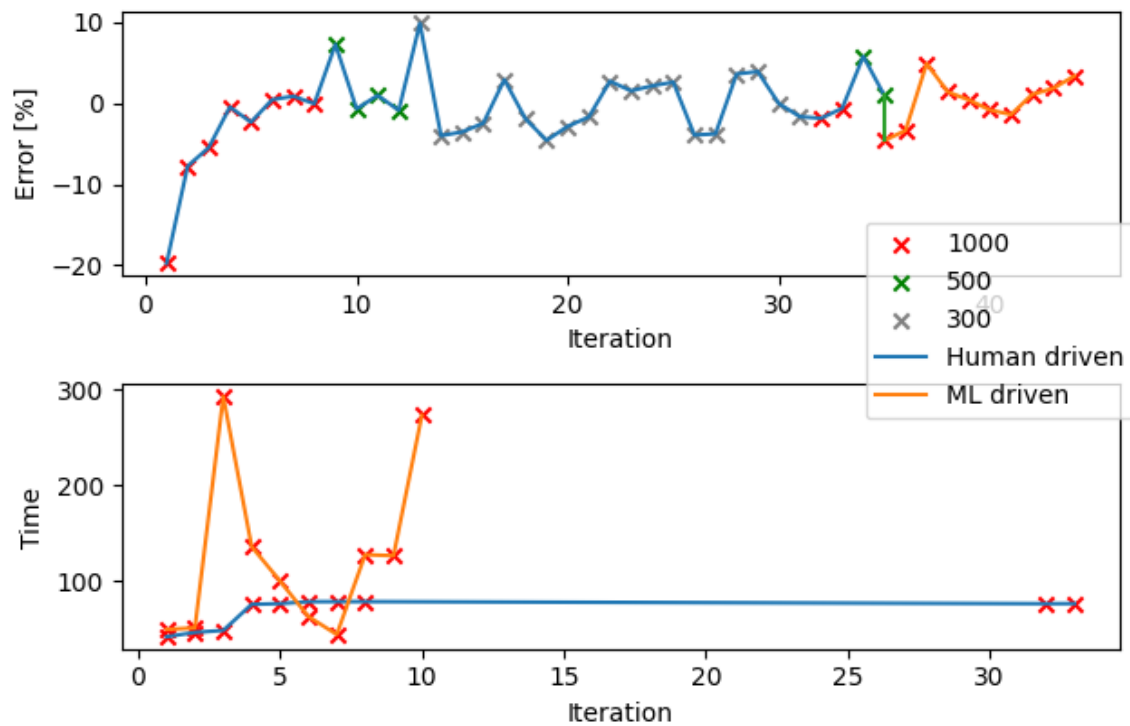
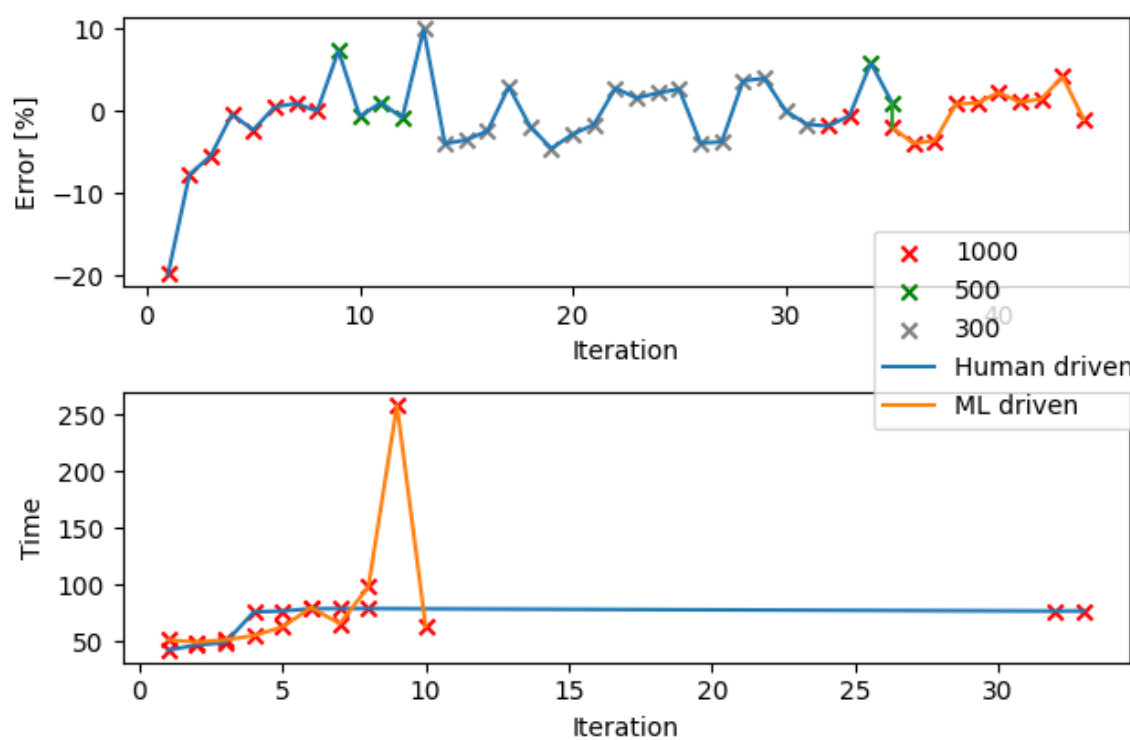


Diagram comparing human-driven and ML test trials (398 cP, set:full, unorderedT)

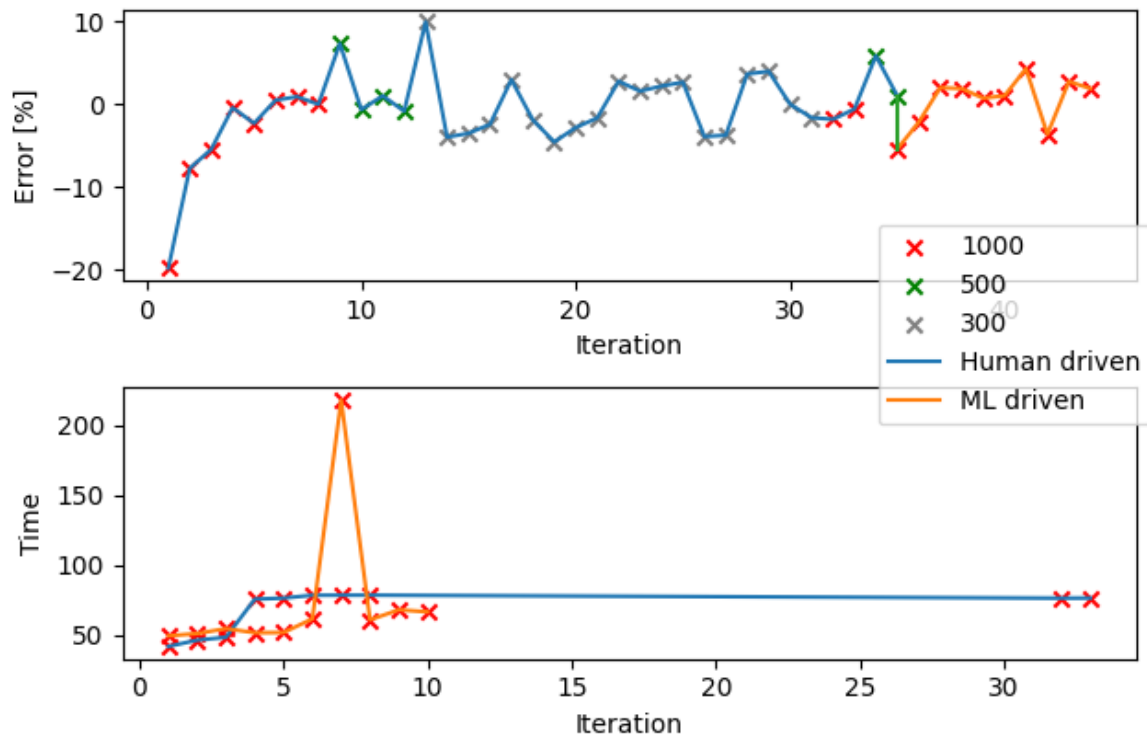
Gaussian Process Regression model with slow transfer penalization,  
trained with full data set



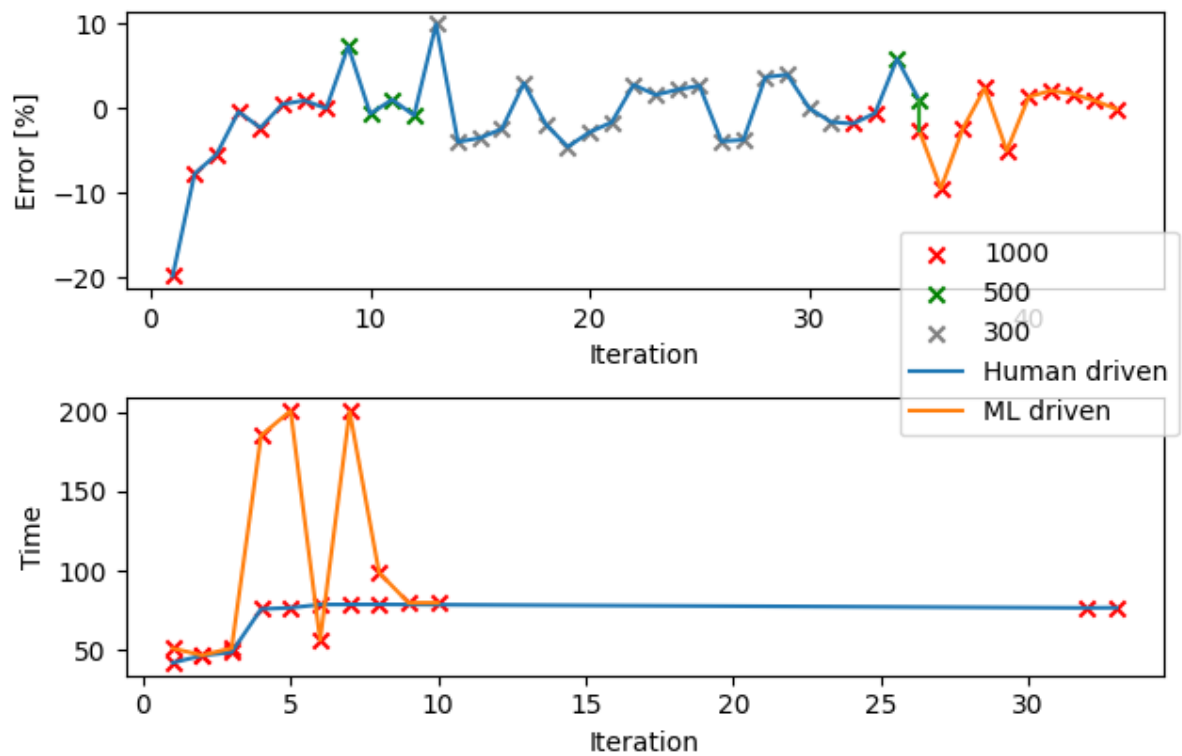
Gaussian Process Regression model with no penalization,  
trained with full data set



Linear Regression model with slow transfer penalization,  
trained with full data set



Linear Regression model with no penalization,  
trained with full data set



## Set: 1

### Observation of trends

Observation 1: LIN - scaling: Without and LIN - scaling: Multiplication

Once an 'optimal' set of parameters are attained, it will double check once or twice to confirm, then continue to explore parameters. The 'confirmation' stage usually occurs in trial 5 or 6 of the 10 trials and it can continue to suggest parameters that deviate quite far from the 'optimal' set of parameters.

The error percentage fluctuates slightly between -2% to 2%, which is within tolerance of 1000uL. The sets of parameters are the most random out of the 3 observations (but does not have very big percentage error)

Observation 2: GPR - scaling: Division and LIN - scaling: Division

The sets of parameters generated are all prioritising faster transfer duration, neglecting the percentage error generated. On average, the percentage errors are all around -11 to -12% for both LIN and GPR models. The parameters do change but very slightly, all targeting at faster transfer rate.

Observation 3: GPR - scaling: Without and GPR - scaling: Multiplication

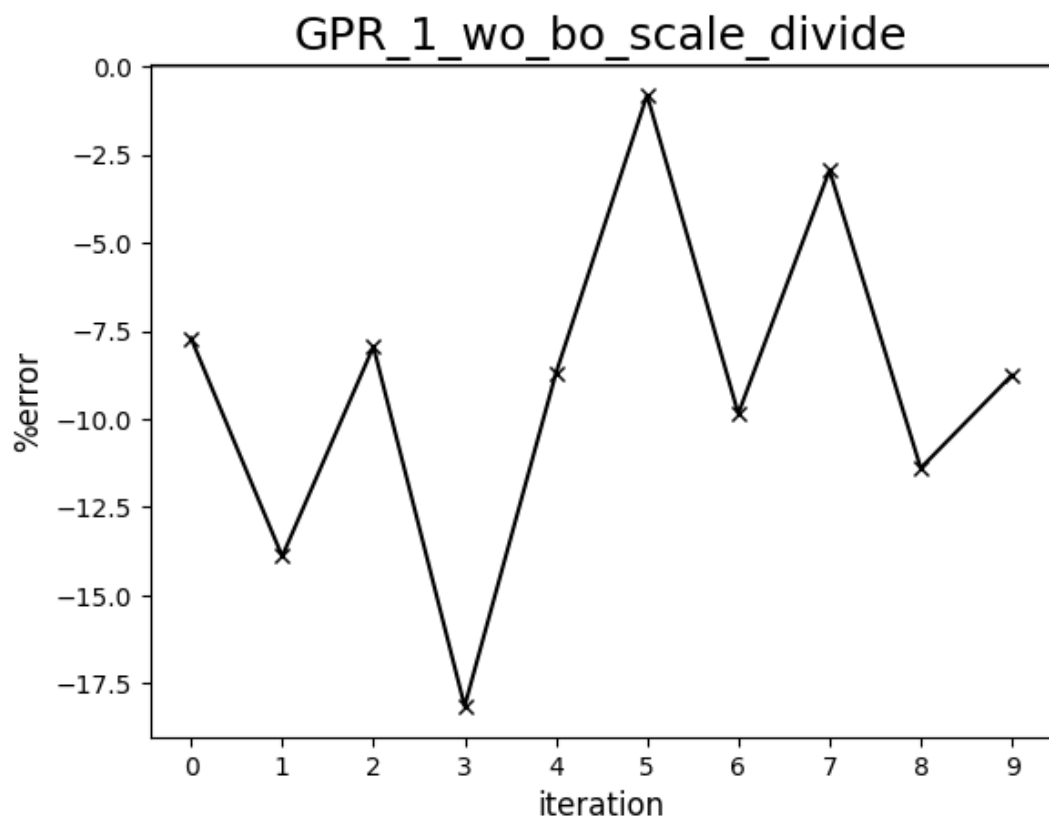
Both iterations show a lot of repetition in the parameters suggested. For scaling: without, the parameters are the same from trial 1 to trial 2 and also the same from trial 3 to trial 10. For scaling: without, the percentage error is an average of -3% while scaling: multiplication has percentage error of -2%. The sets of parameters suggested for GPR scaling: without and multiplication can be said to be almost identical.

Repetition is commonly observed for trials that use the GPR model, this pattern is consistent for both set: 1 and set: full.

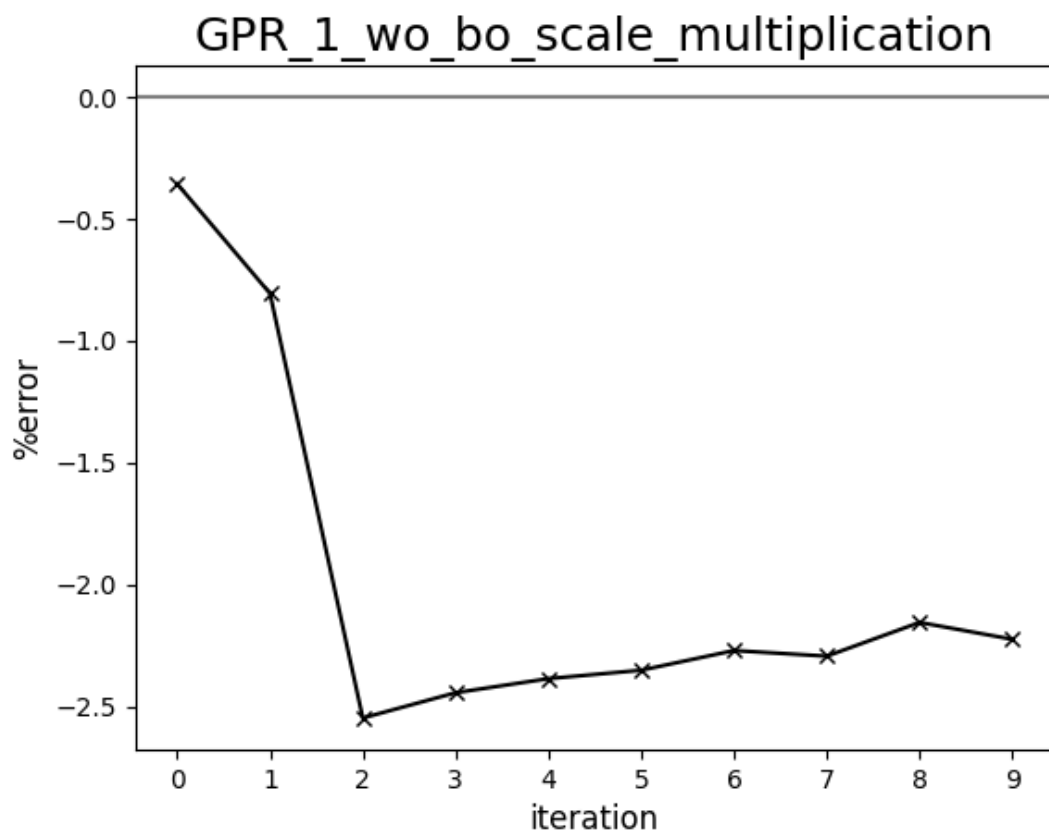
It is worth noting that there are some sets of parameters that are repeated for many different iterations. It might help in determining the optimal set of data in the future. A few of such set of parameters are: (AR,DR,AD,DD)

- 52.3, 47.5, 3.1, 2.8
- 66.0, 66.0, 0.0, 6.0

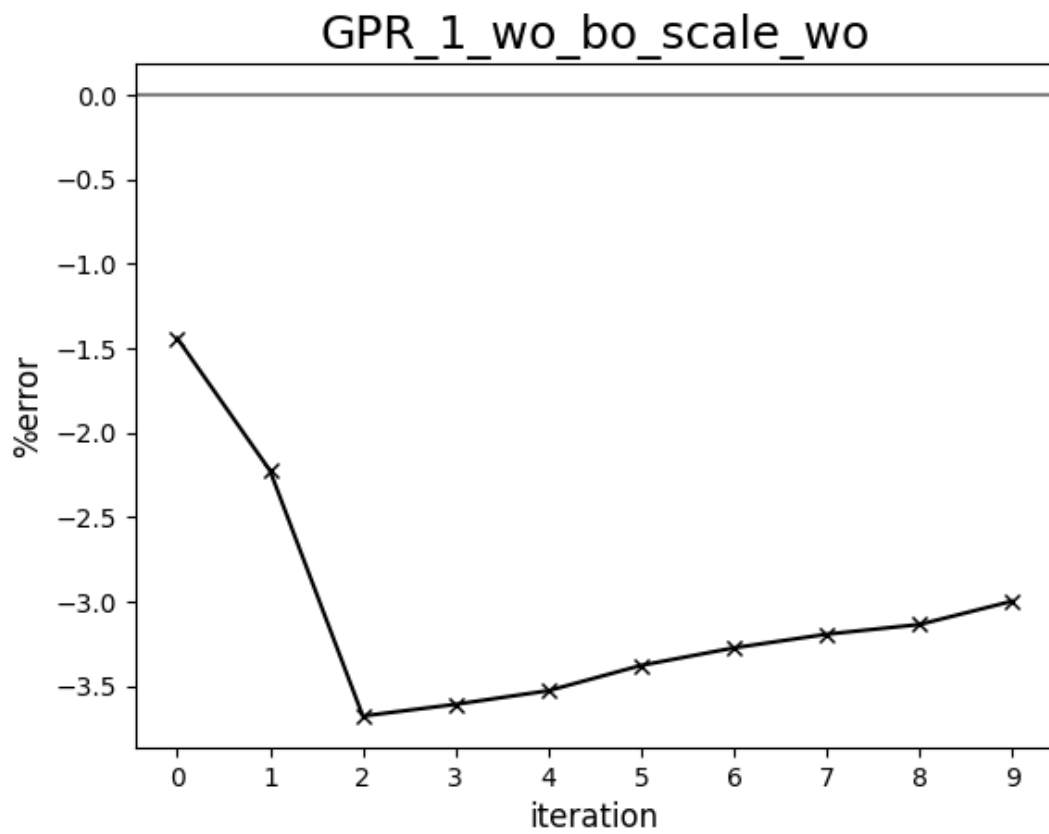
GPR scaling: divide



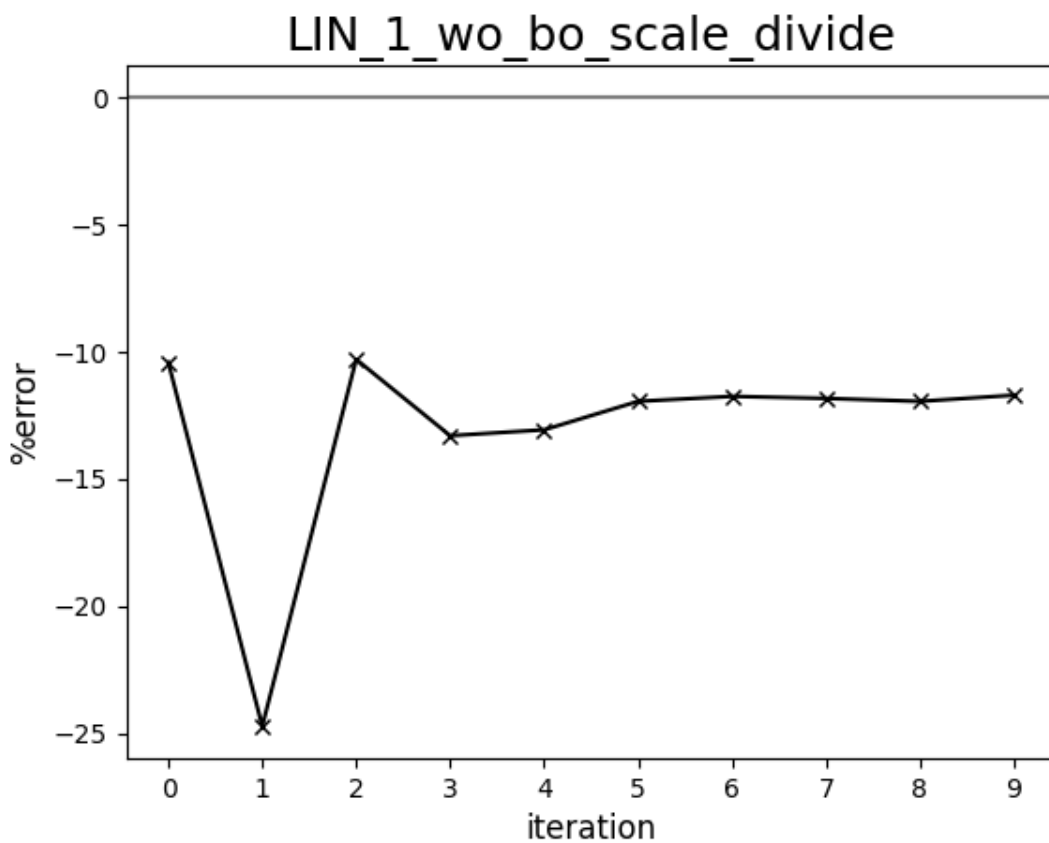
GPR scaling: multiplication



GPR scaling: without

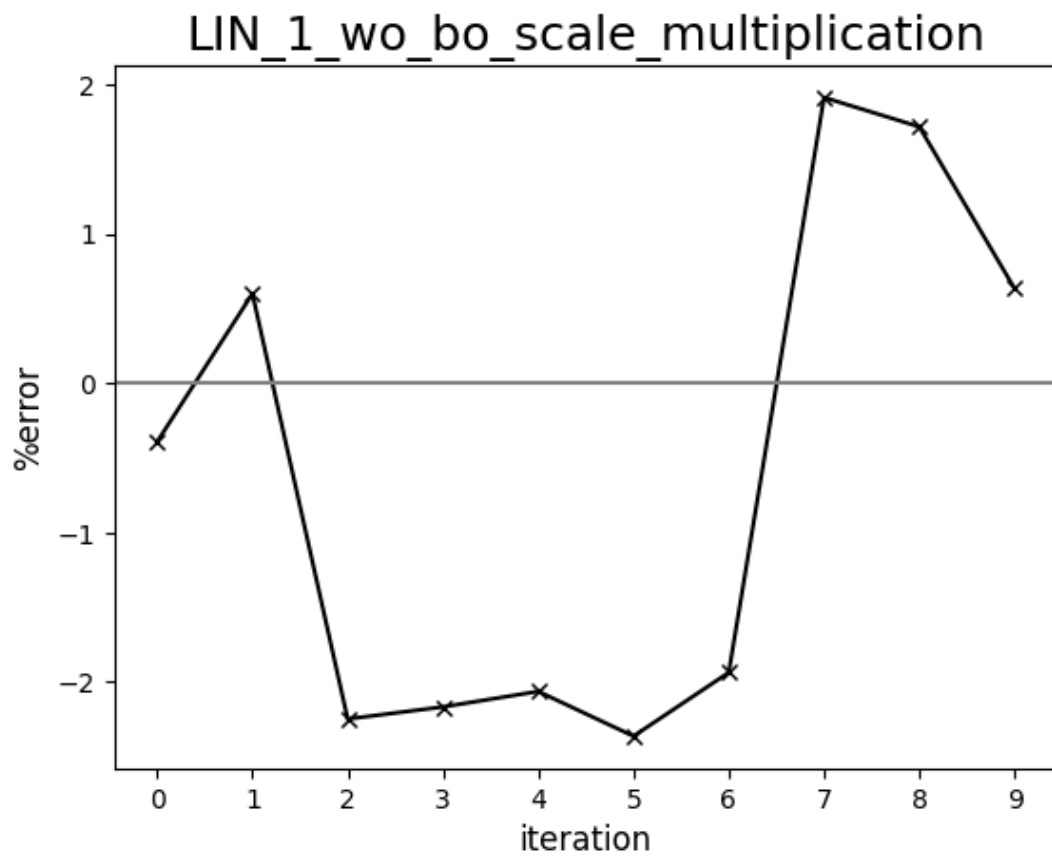


LIN scaling: divide





LIN scaling: multiplication



LIN scaling: without

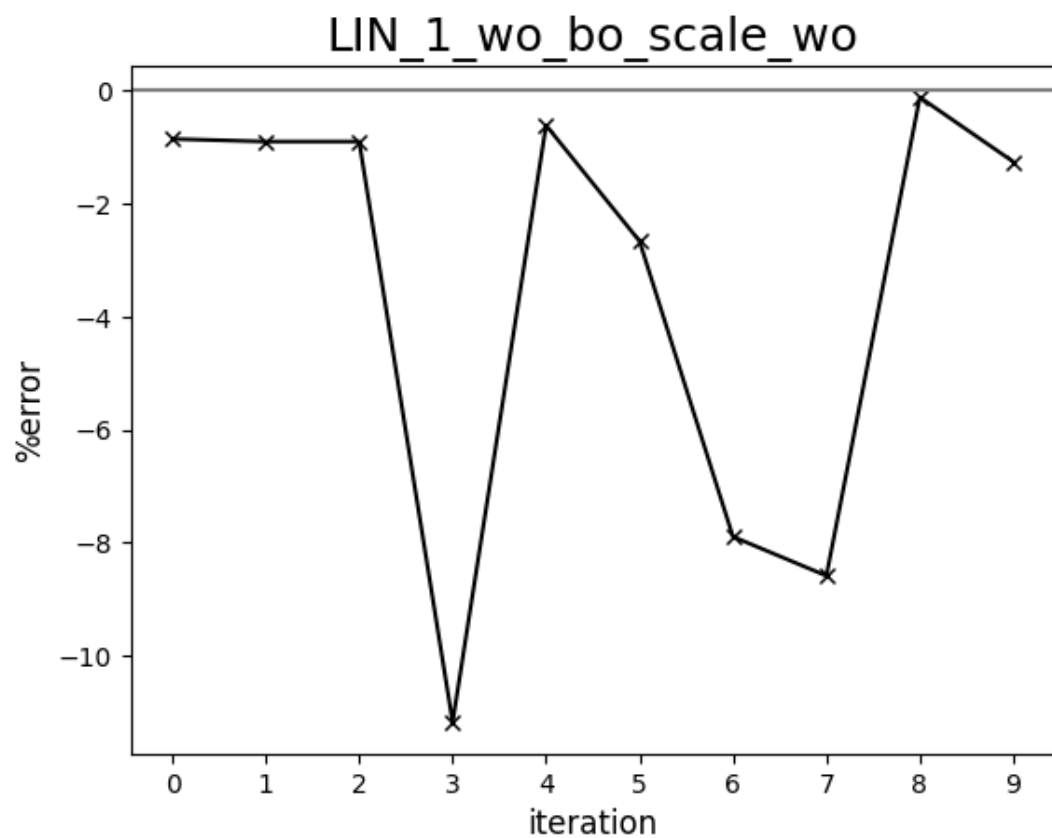
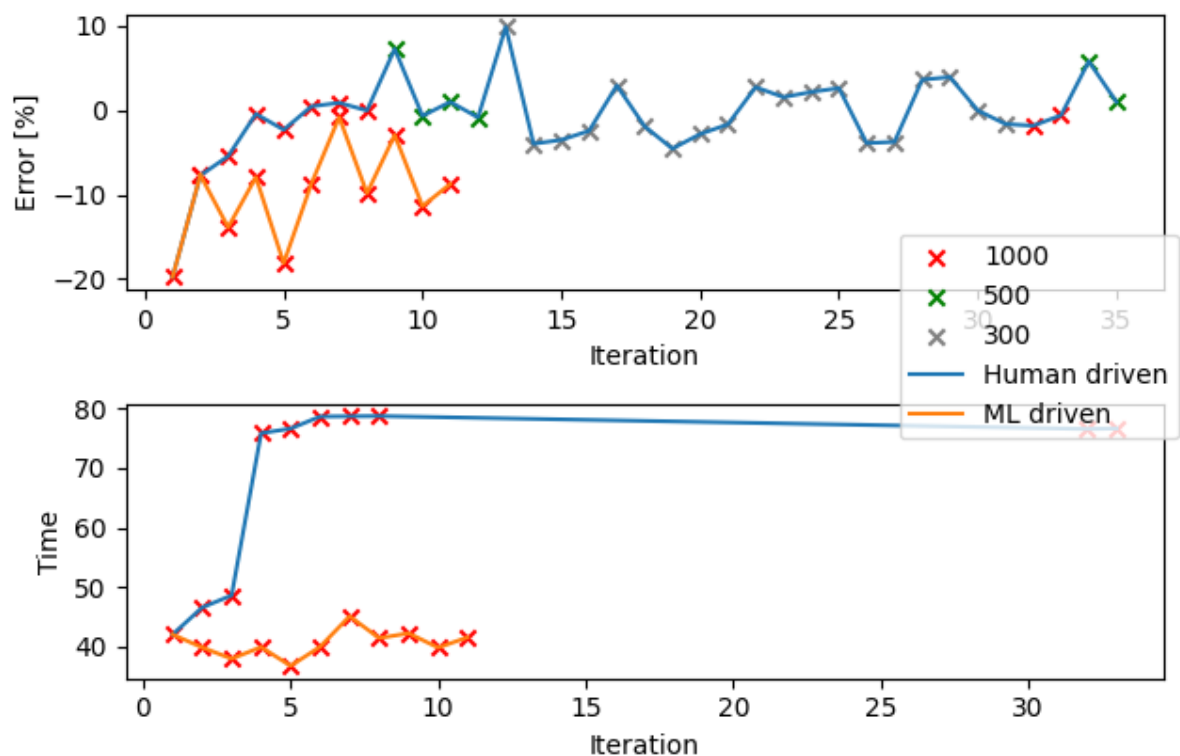
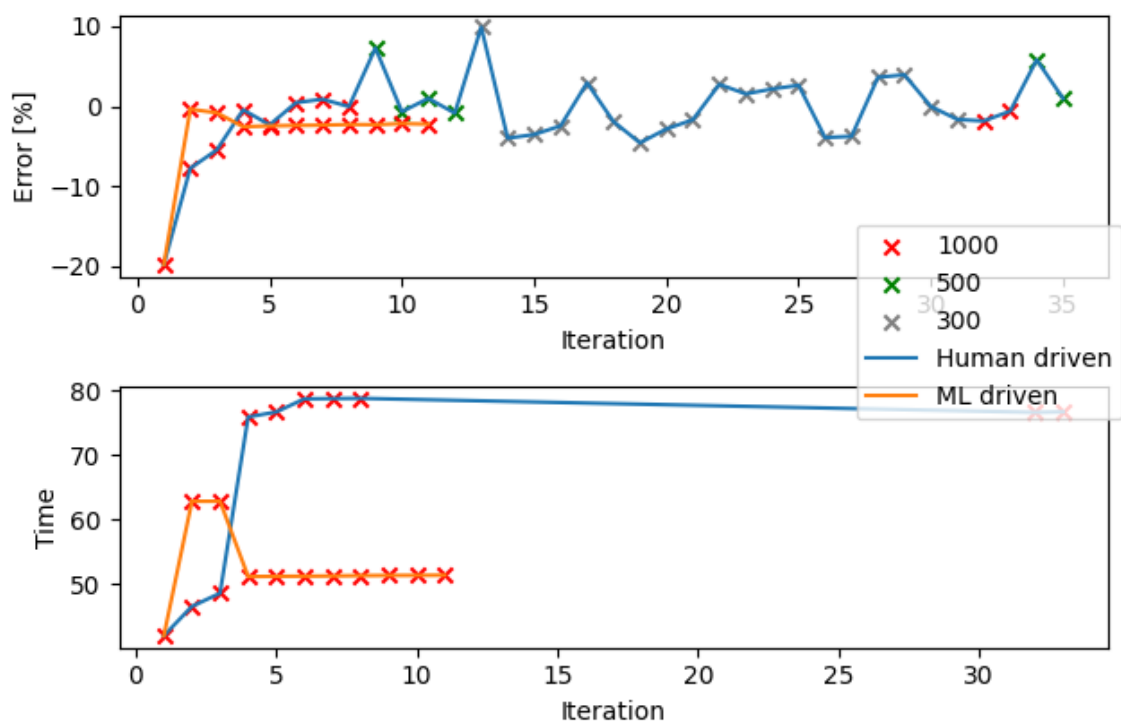


Diagram comparing human-driven and ML test trials (398 cP, set:1)

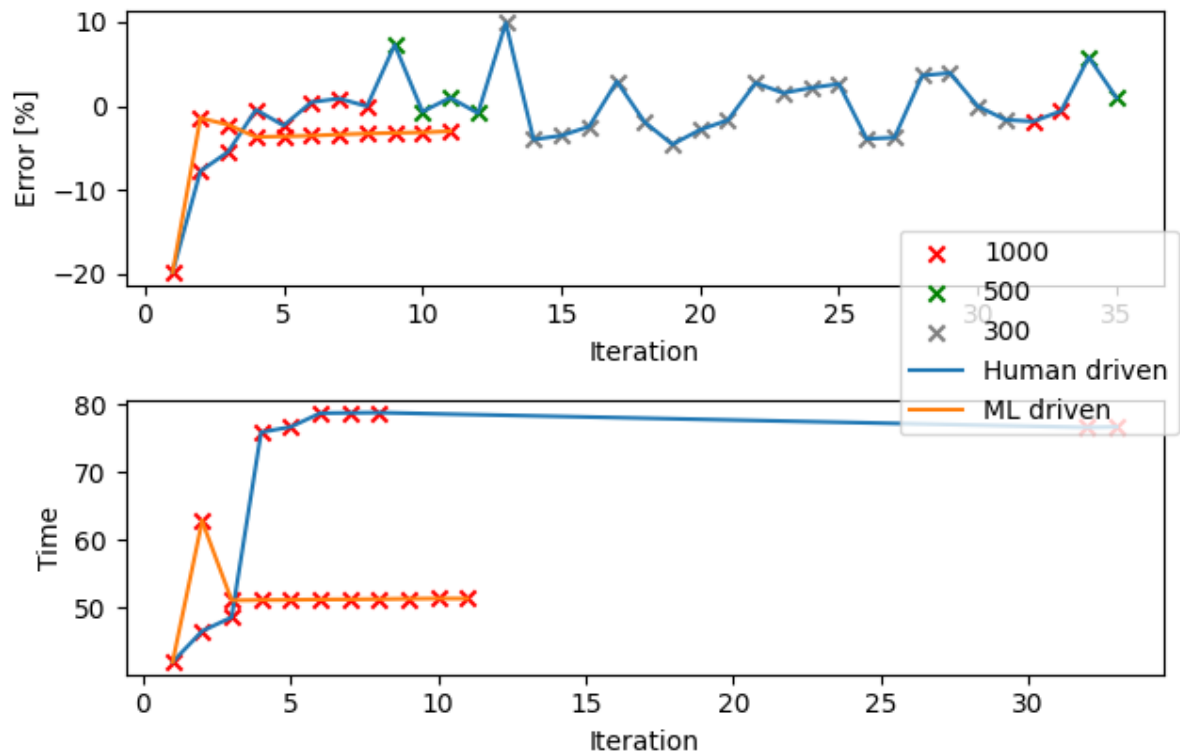
Gaussian Process Regression model with fast transfer penalization,  
trained with 1 initialization data



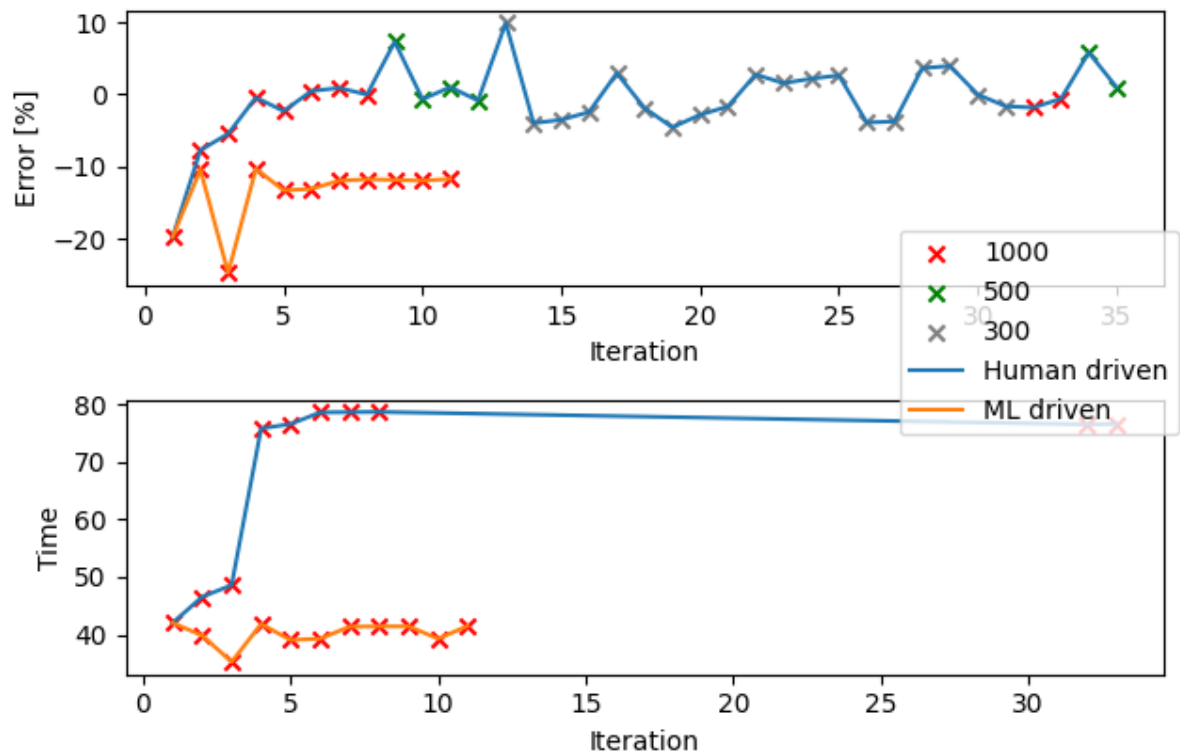
Gaussian Process Regression model with slow transfer penalization,  
trained with 1 initialization data



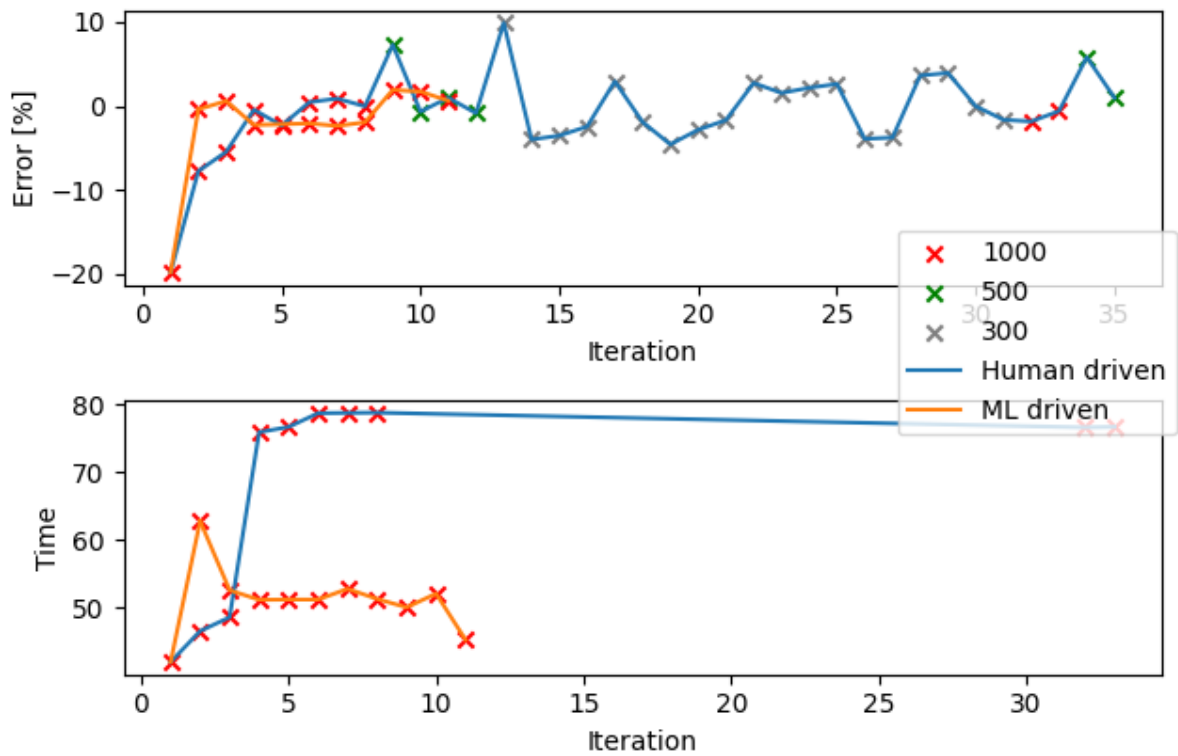
Gaussian Process Regression model with no penalization,  
trained with 1 initialization data



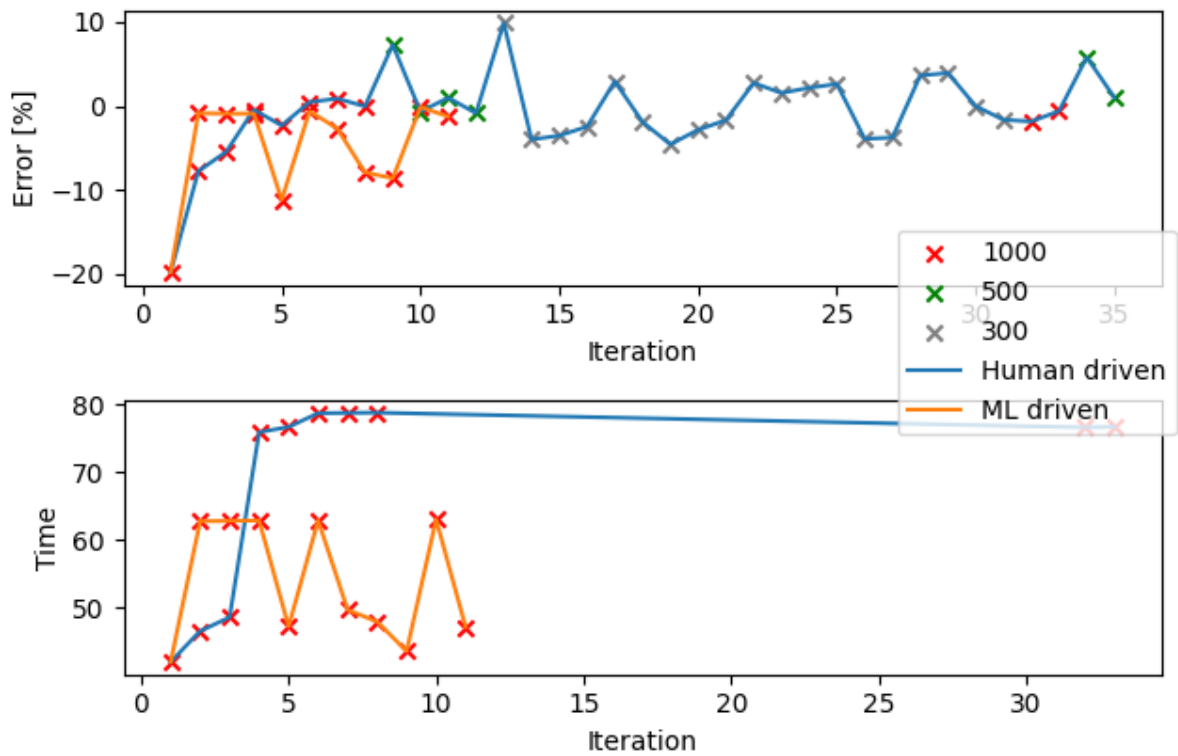
Linear Regression model with fast transfer penalization,  
trained with 1 initialization data



Linear Regression model with slow transfer penalization,  
trained with 1 initialization data



Linear Regression model with no penalization,  
trained with 1 initialization data



## Set: 1 (unorderedT)

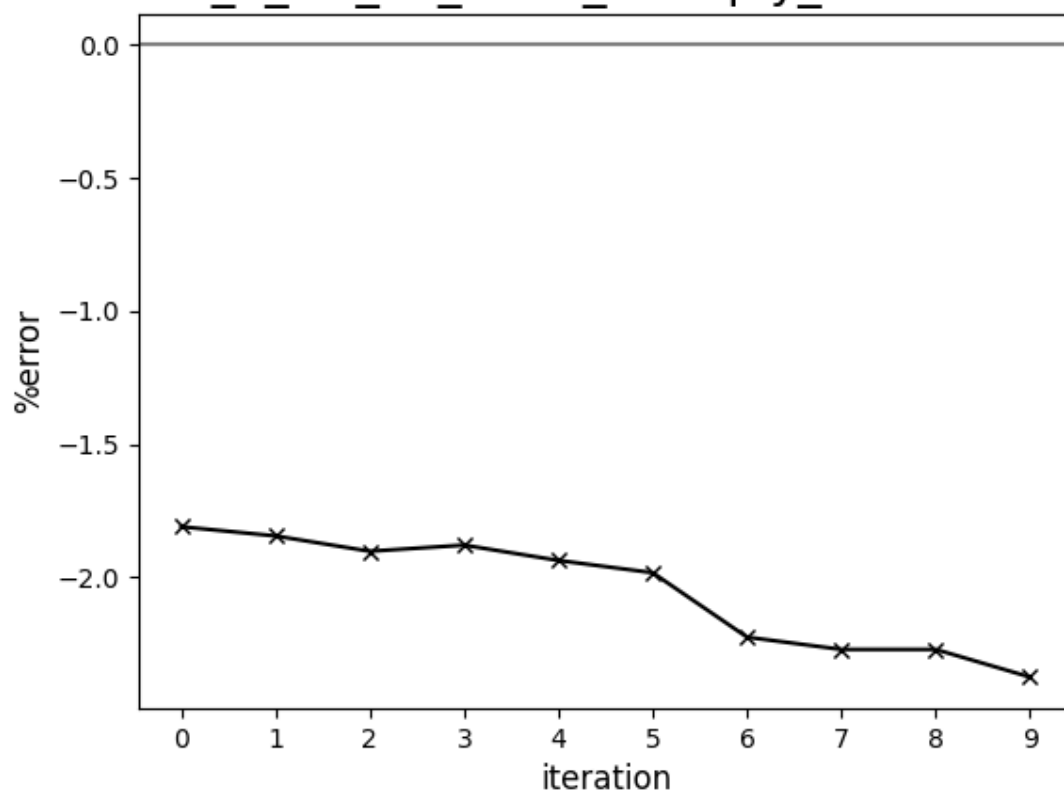
### Observation of trends

All iterations

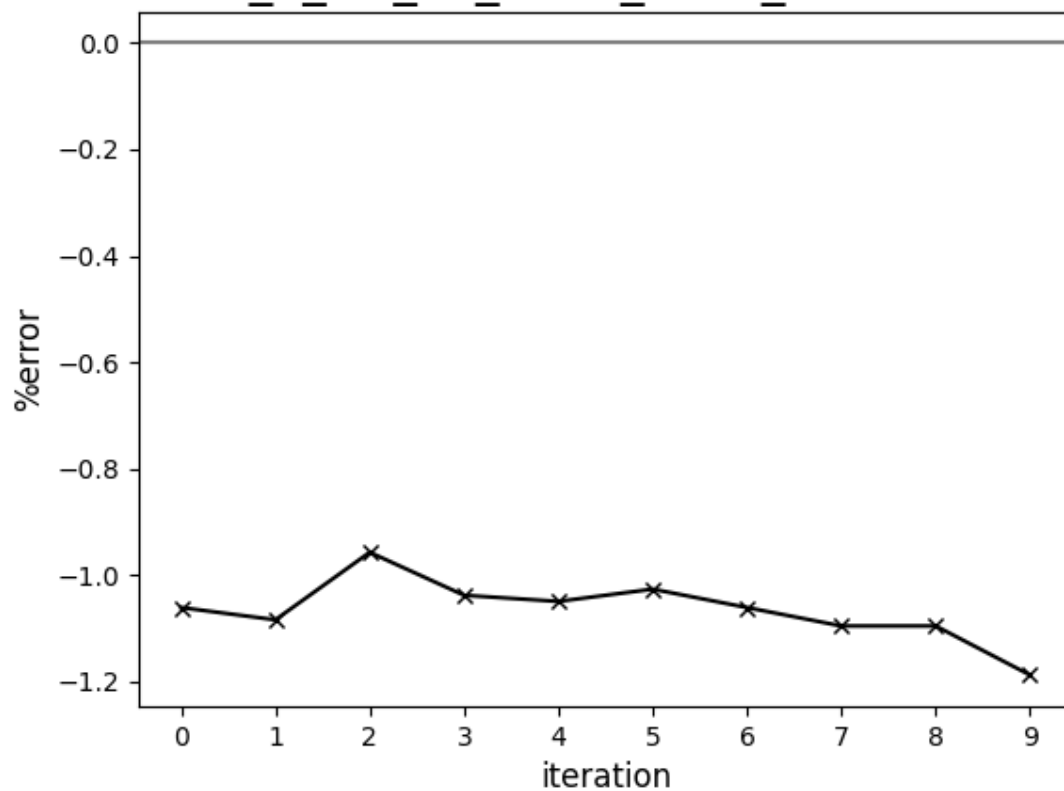
All 4 iterations have sets of parameters that are exactly the same. Basically, the only set of parameters used is: (AR, DR, DA, DD)  
34.6, 47.3, 4.3, 2.9

Despite the fact that all 40 trials are identical, the transfers are quite accurate. The percentage error falls between -1 to -2%. All trials took around 62 to 63 seconds, as compared to 80s derived from the standard calibration.

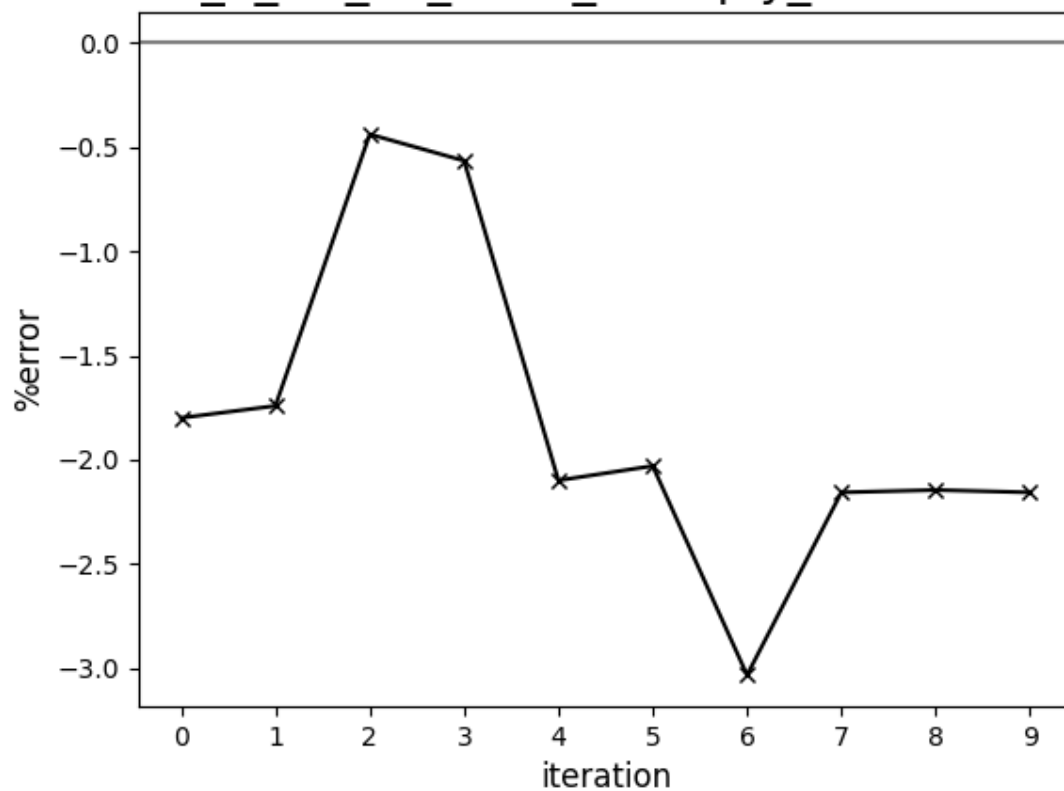
GPR\_1\_wo\_bo\_scale\_multiply\_unorderedT



GPR\_1\_wo\_bo\_scale\_none\_unorderedT



LIN\_1\_wo\_bo\_scale\_multiply\_unorderedT



LIN\_1\_wo\_bo\_scale\_none\_unorderedT

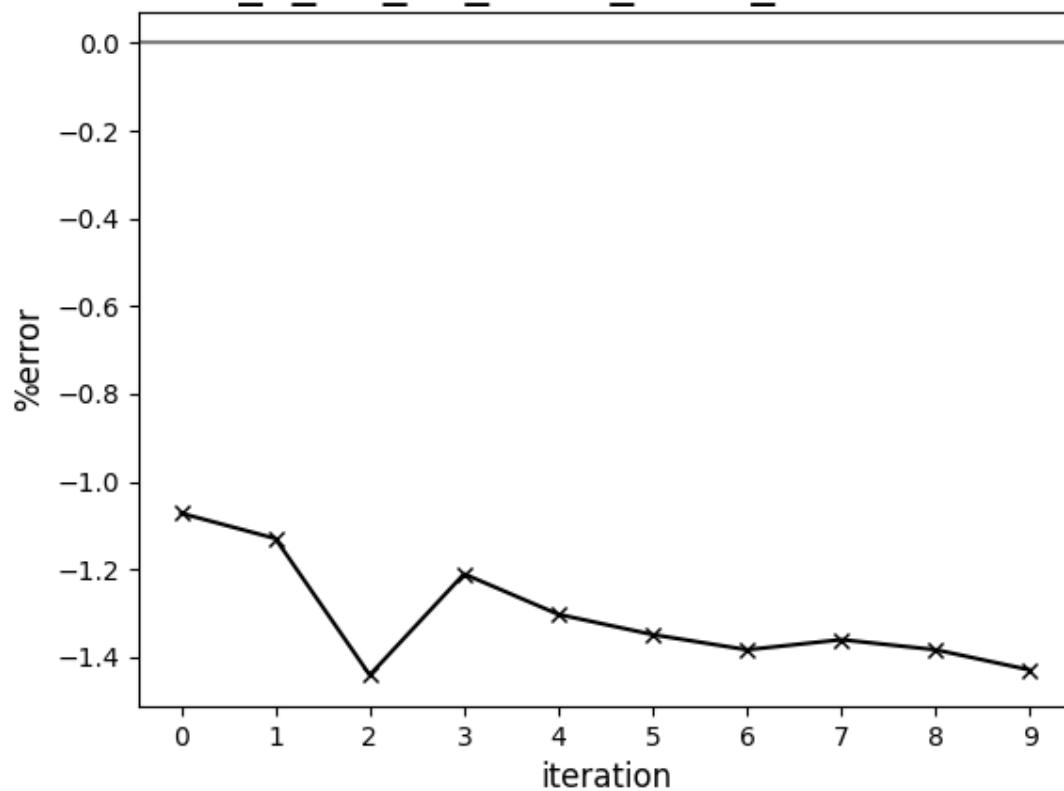
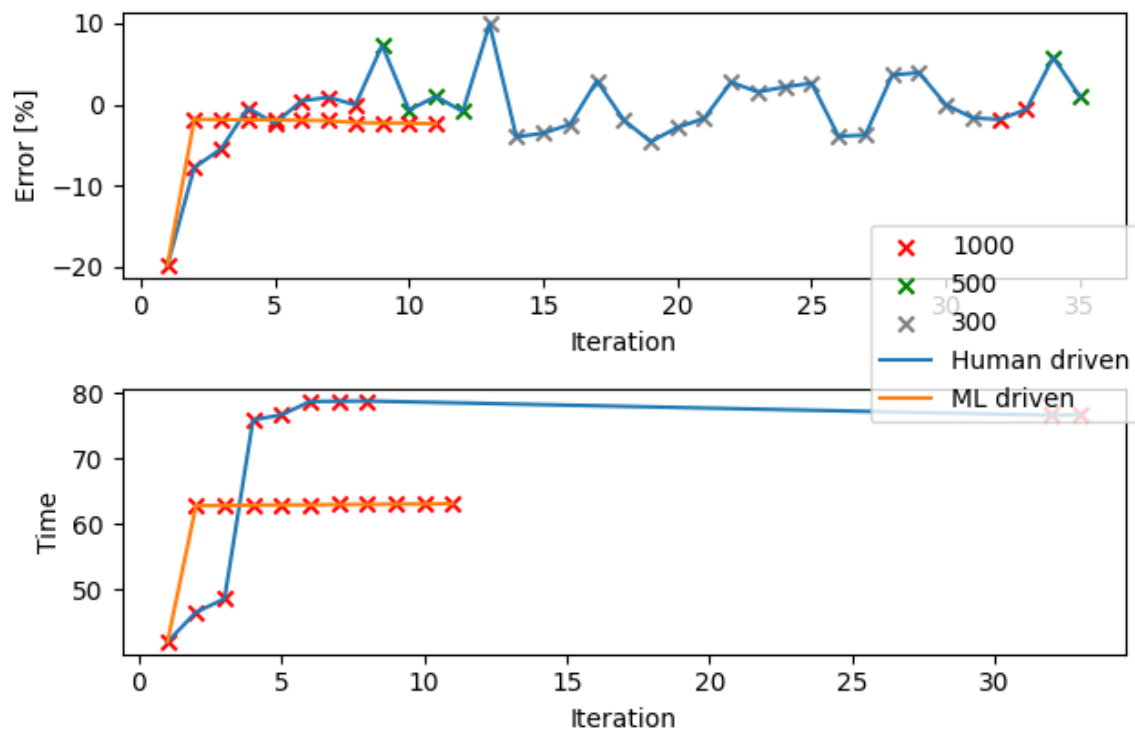
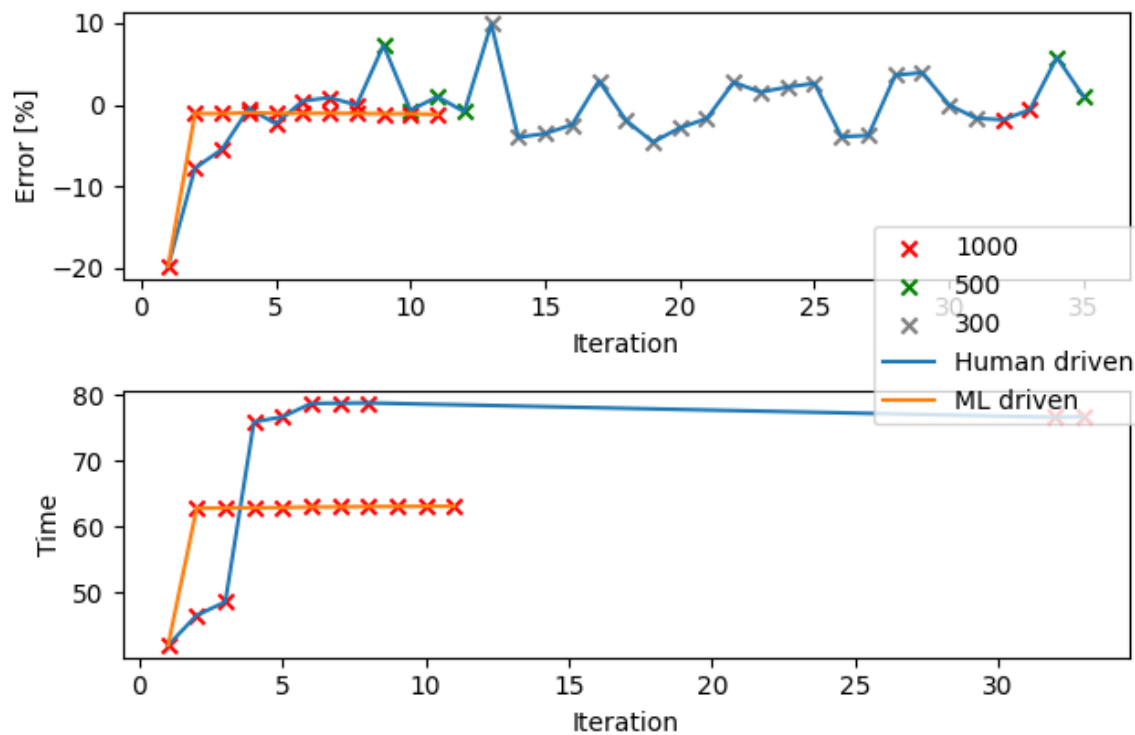


Diagram comparing human-driven and ML test trials (398 cP, set:1, unorderedT)

Gaussian Process Regression model with slow transfer penalization,  
trained with 1 initialization data

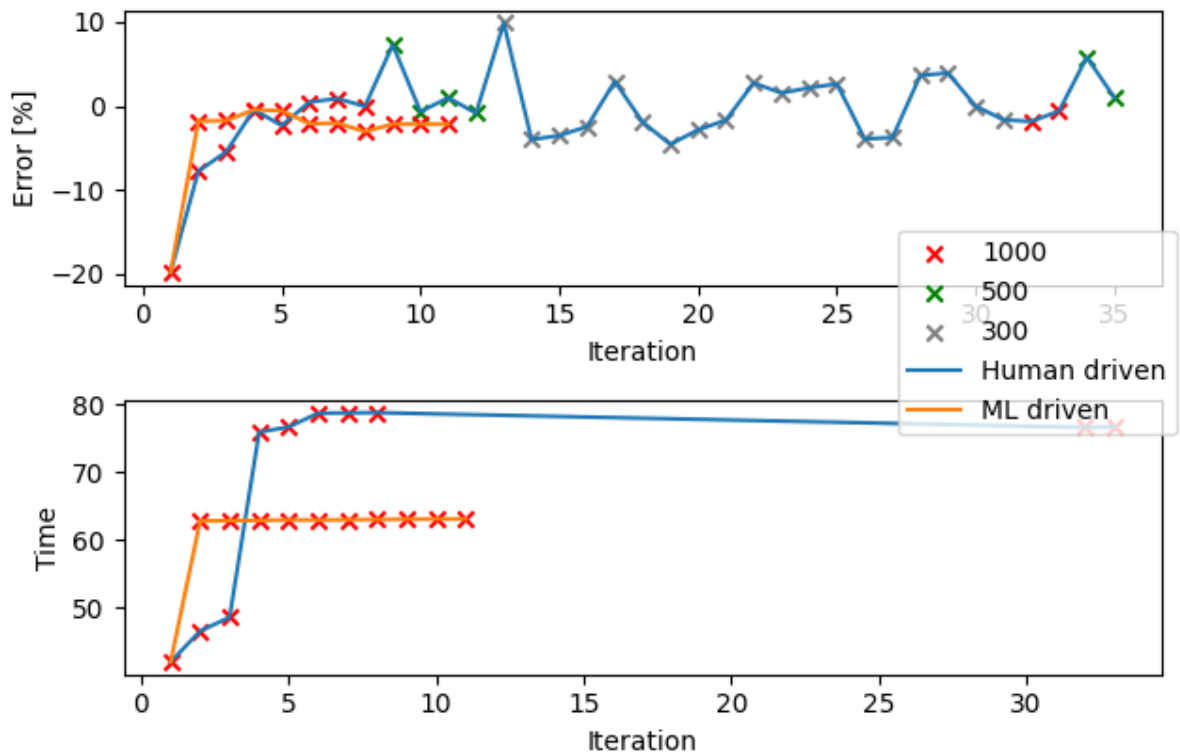


Gaussian Process Regression model with no penalization,  
trained with 1 initialization data

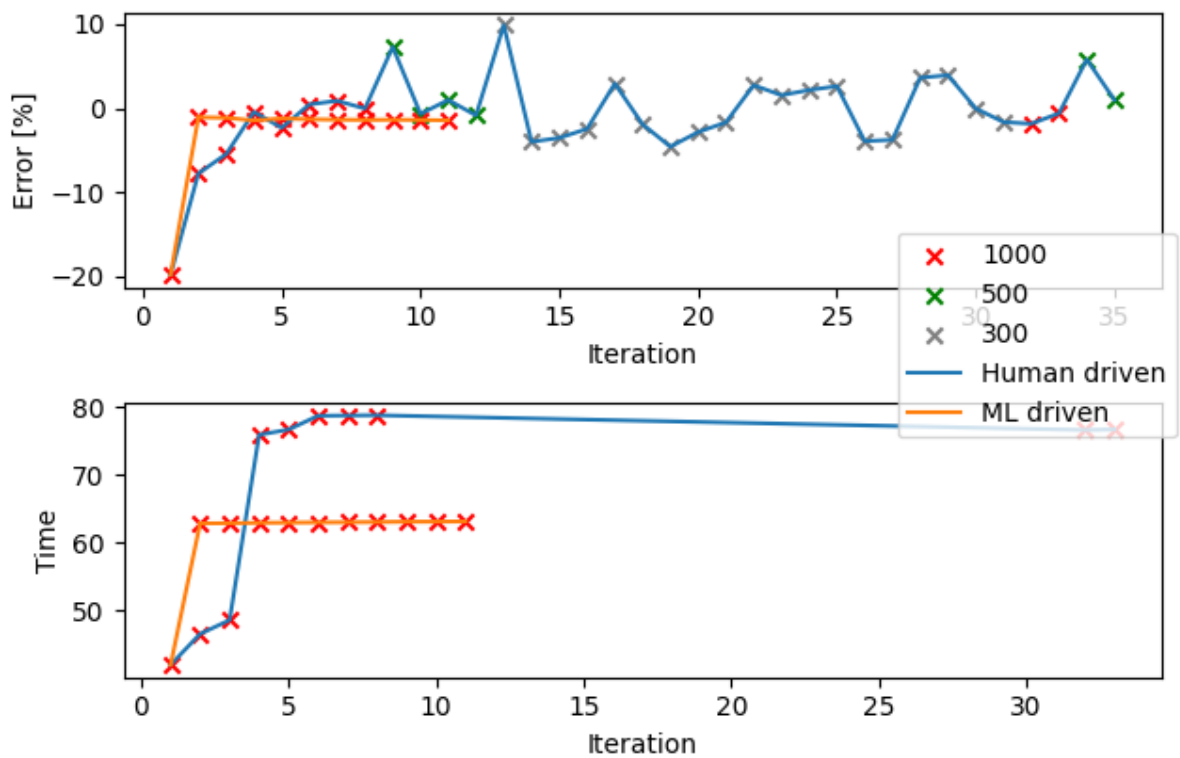




Linear Regression model with slow transfer penalization,  
trained with 1 initialization data



Linear Regression model with no penalization,  
trained with 1 initialization data



# Standard 1275 cP

## Machine learning segment

---

### Set: Full

#### Observation of trends

Observation: GPR - scaling: Multiplication

The percentage error for this iteration is relatively consistent, ranging from -7 to -8%. A lot of repetitions in the sets of parameters generated are observed. The first 2 trials have the same sets of parameters and the rest of the trials share another set of parameters.

The most common set of parameters is: (AR, DR, DA, DD)  
15.3, 7.0, 2.6, 4.5

Observation: GPR - scaling: Without

Repetitions of sets of parameters are also observed. The percentage error ranges from -3 to -7%, thus not as consistent as the trials with GPR - scaling: Multiplication.

The most common set of parameters is: (AR, DR, DA, DD)  
11.1, 13.7, 4.1, 5.5

Observation: LIN - scaling: Multiplication

For this iteration, not as many repetitions are seen in the sets of parameters generated. Thus, the percentage errors are more distributed, ranging from -3 to -17%, with an average of -7%.

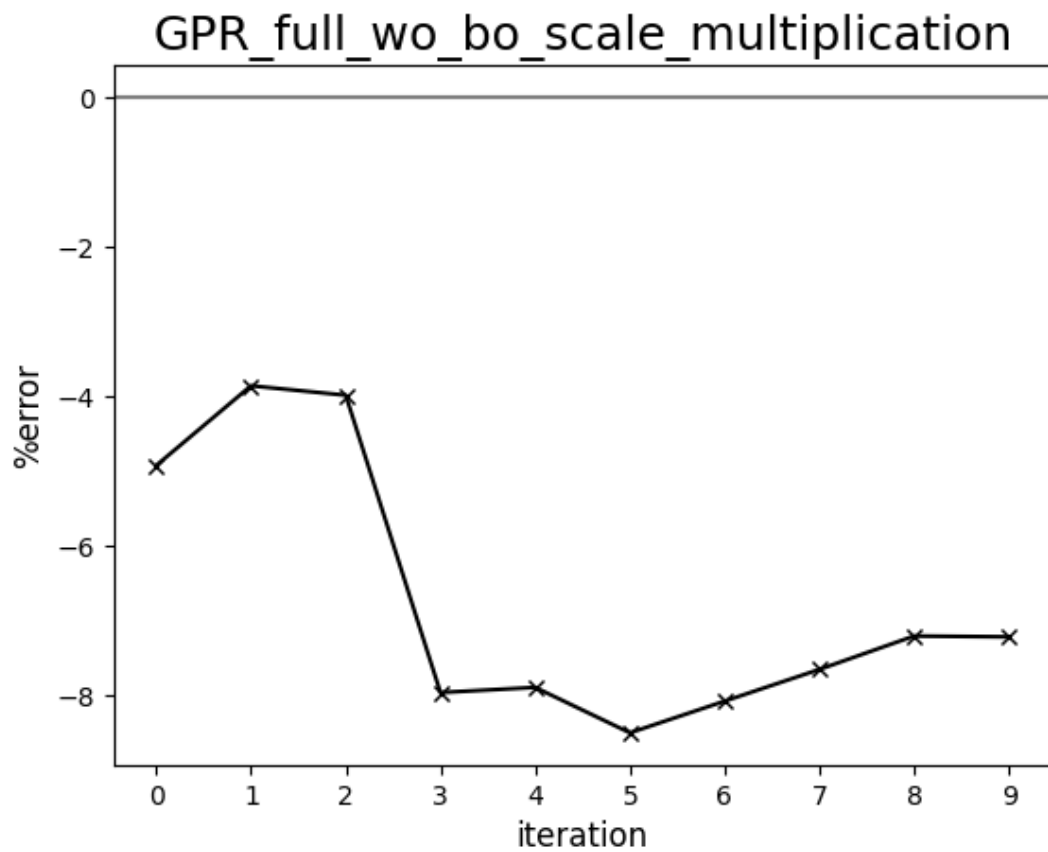
The most common set of parameters is: (AR, DR, DA, DD)  
11.5, 7.3, 4.2, 3.6

Observation: LIN - scaling: Without

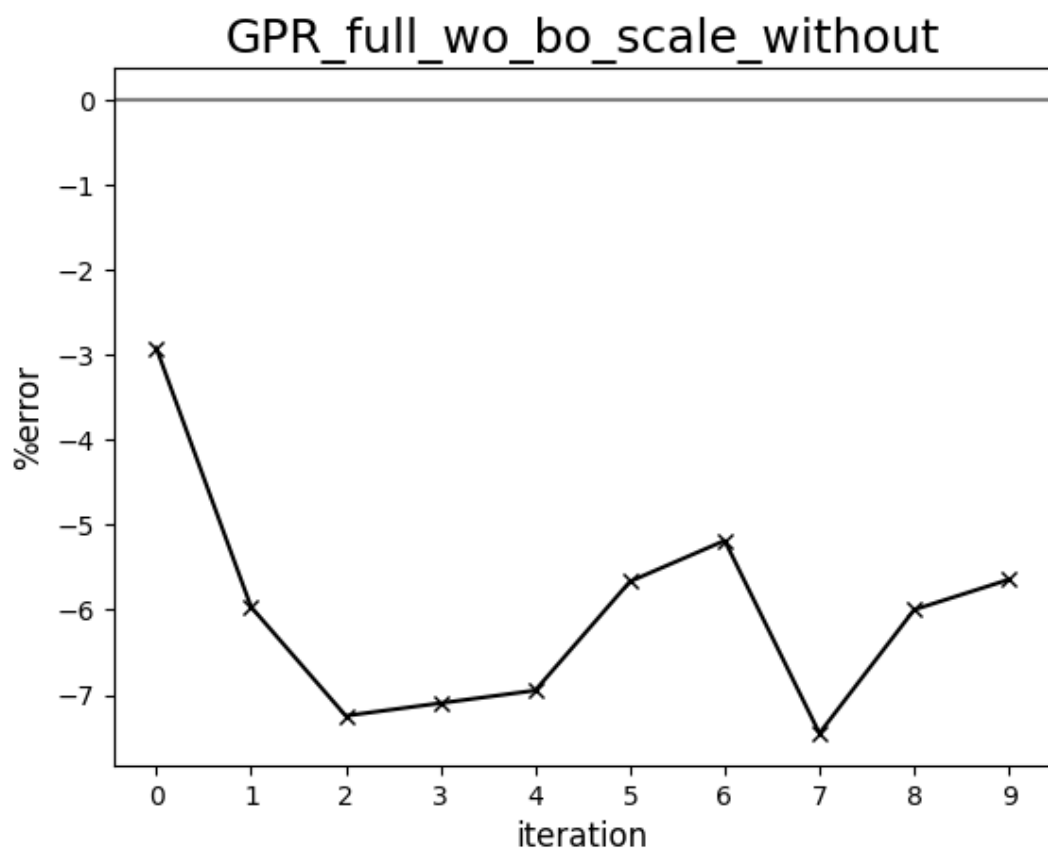
There are a lot of repetitions in the parameters generated, the parameters remained the same from the fourth trial onwards. The average is around -8%.

The most common set of parameters is: (AR, DR, DA, DD)  
15.3, 7.0, 2.6, 4.5

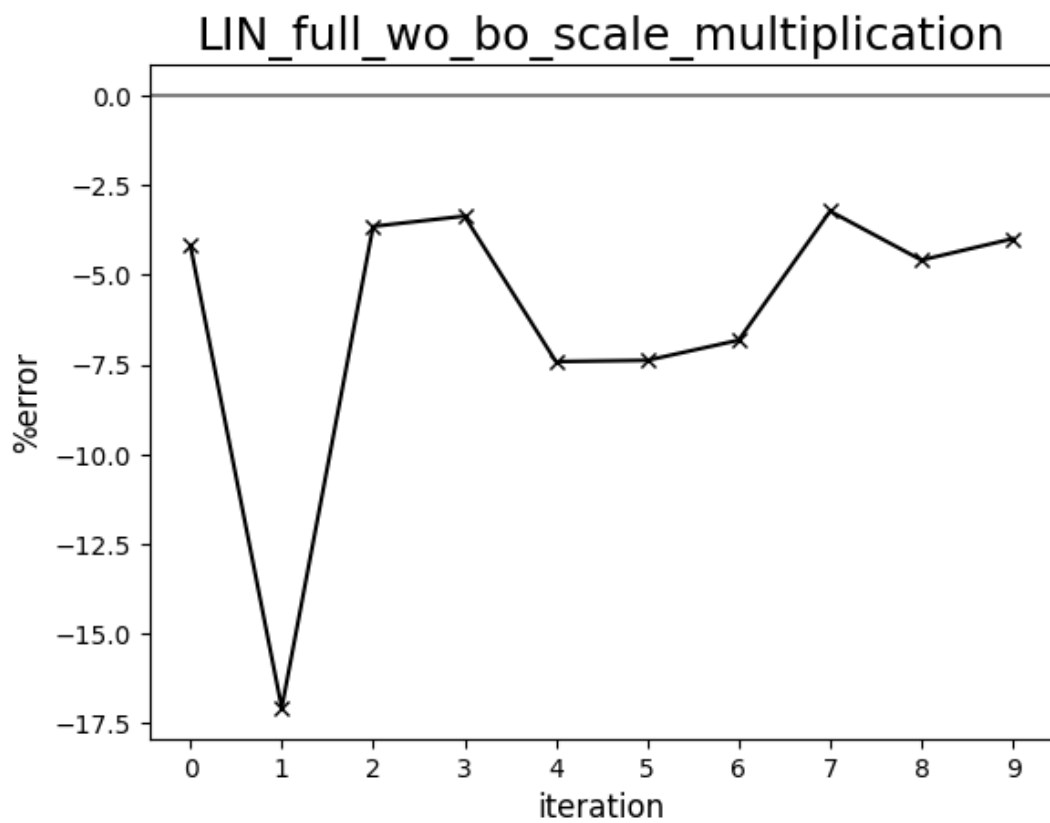
GPR scaling: multiplication



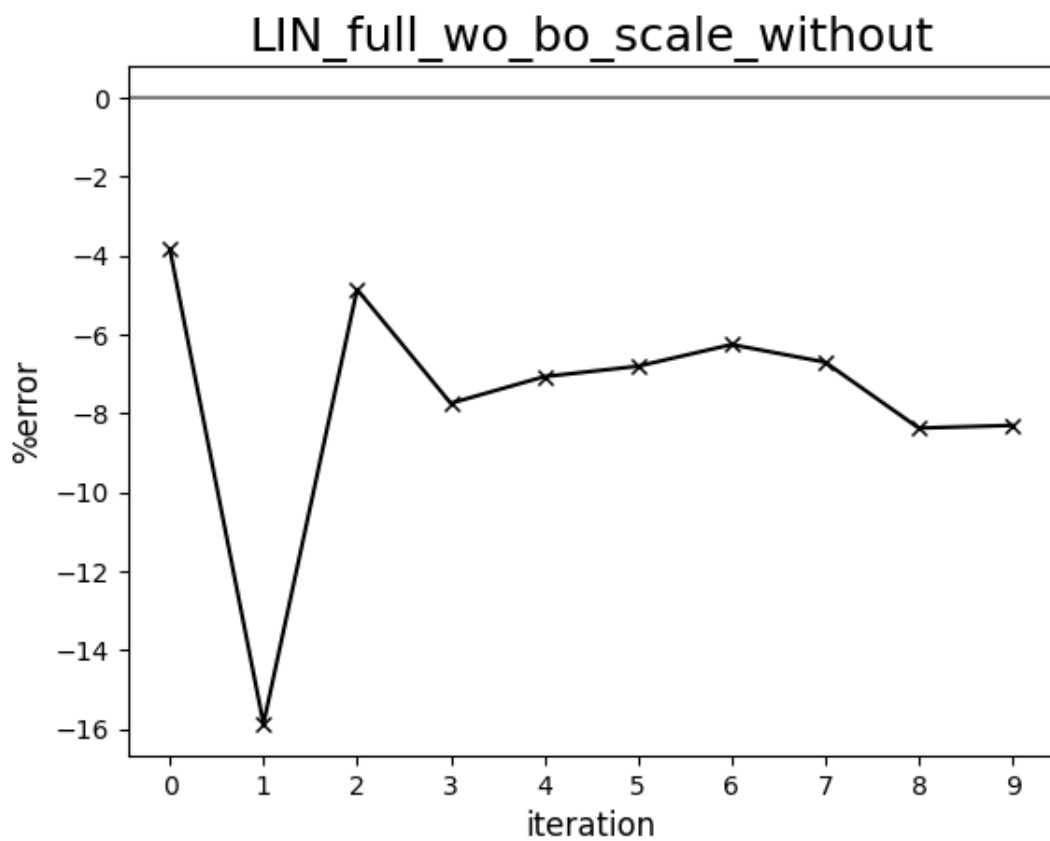
GPR scaling: without



LIN scaling: multiplication

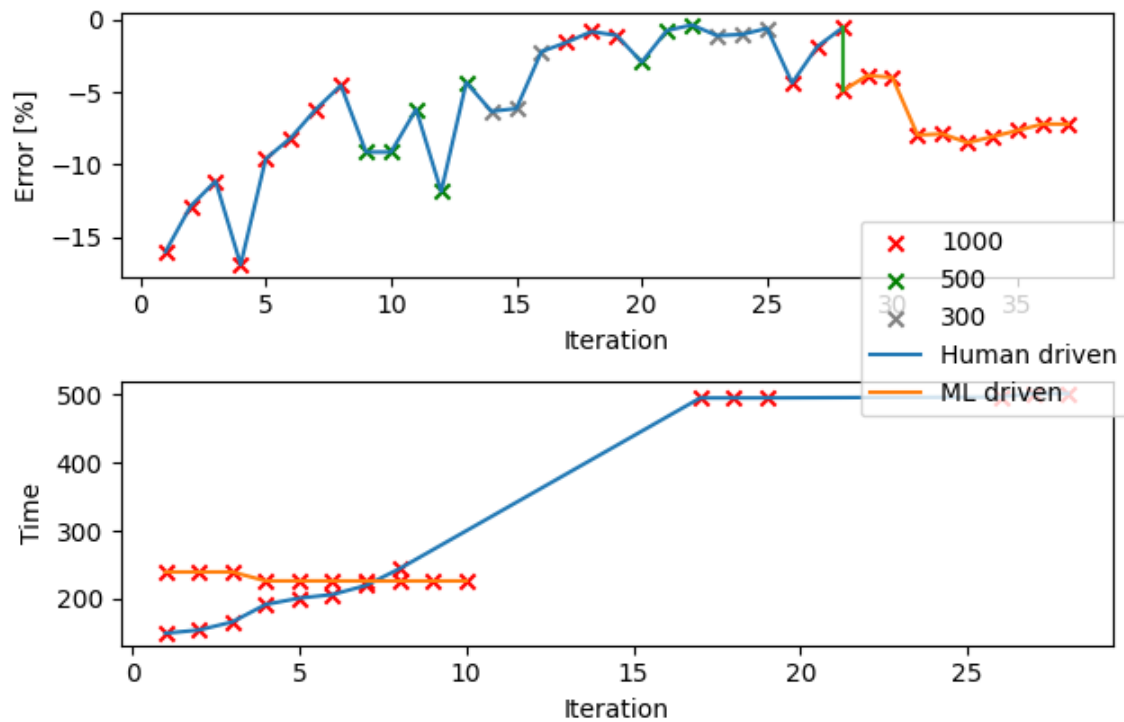


LIN scaling: without

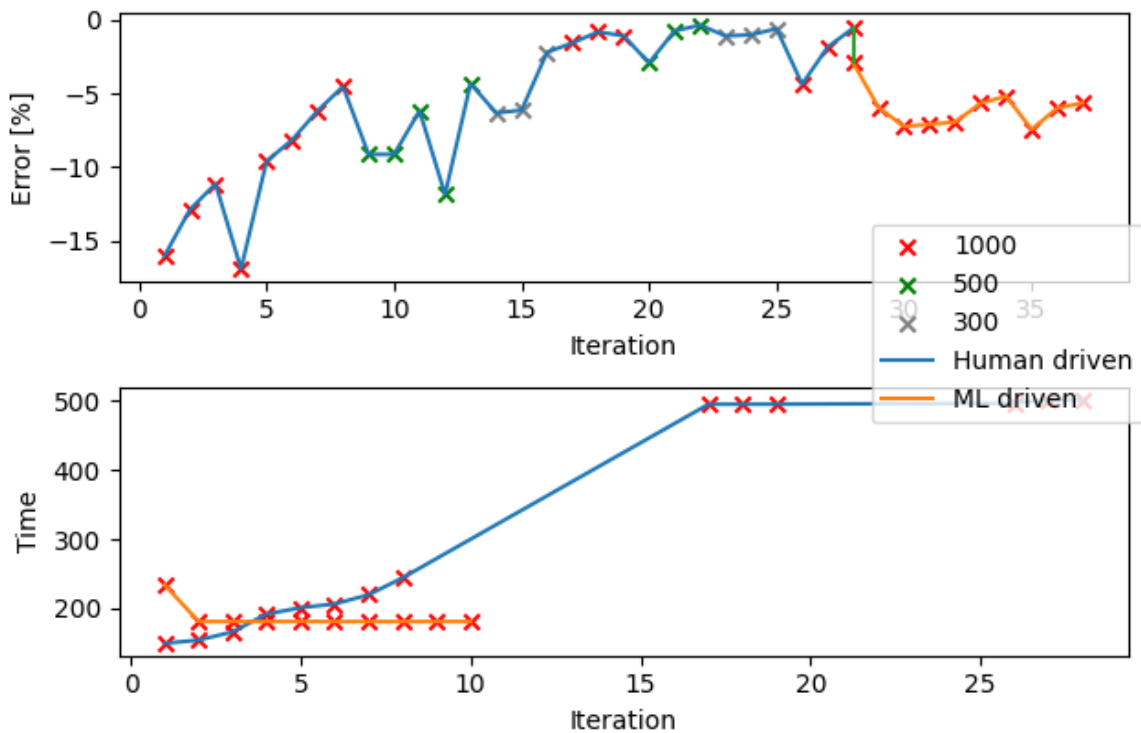


## Diagram comparing human-driven and ML test trials (1275 cP, set:full)

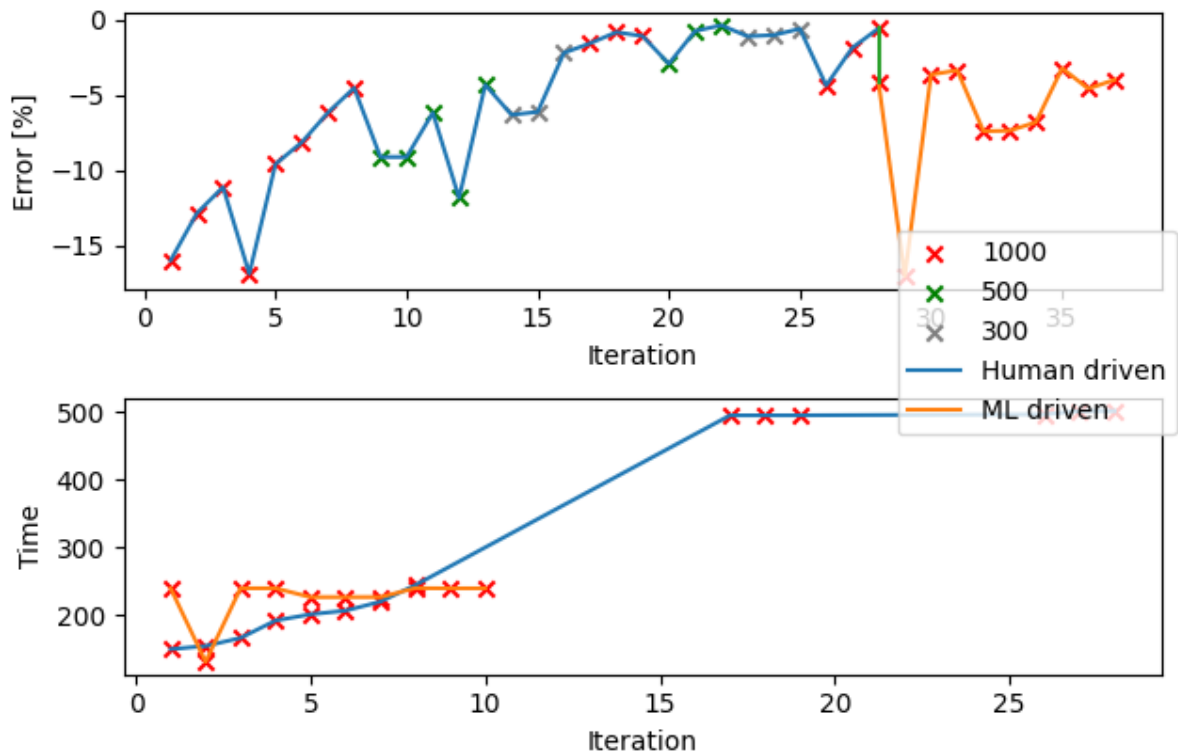
Gaussian Process Regression model with slow transfer penalization,  
trained with full data set



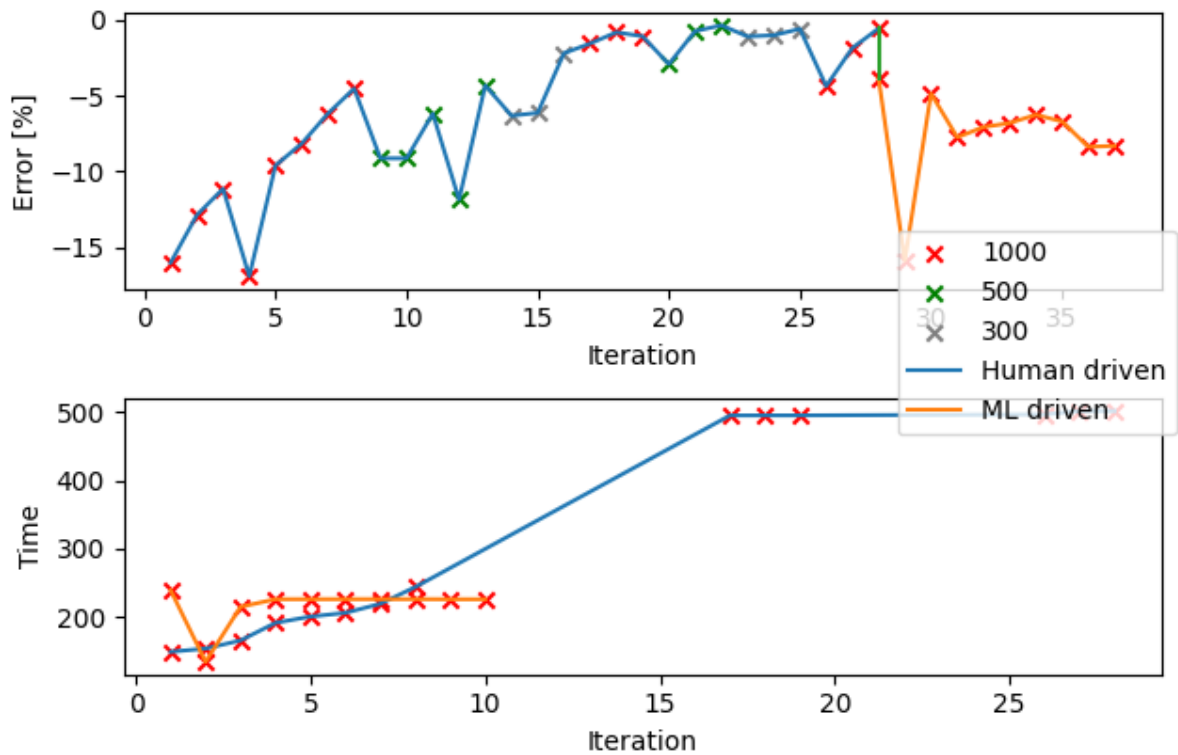
Gaussian Process Regression model with no penalization,  
trained with full data set



Linear Regression model with slow transfer penalization,  
trained with full data set



Linear Regression model with no penalization,  
trained with full data set



## Set: Full (unorderedT)

### Observation of trends

Observation for LIN - scaling: none

Some variations in the parameters generated but most trials share the same set of parameters: 15.3, 7.0, 2.6, 4.5, with a percentage error of -6 to -7%.

Observation for LIN - scaling: multiply

Some variations between the 2 sets of parameters: 11.5, 7.3, 4.2, 3.6 (percentage error around -3 to -4%) and 15.3, 7.0, 2.6, 4.5.

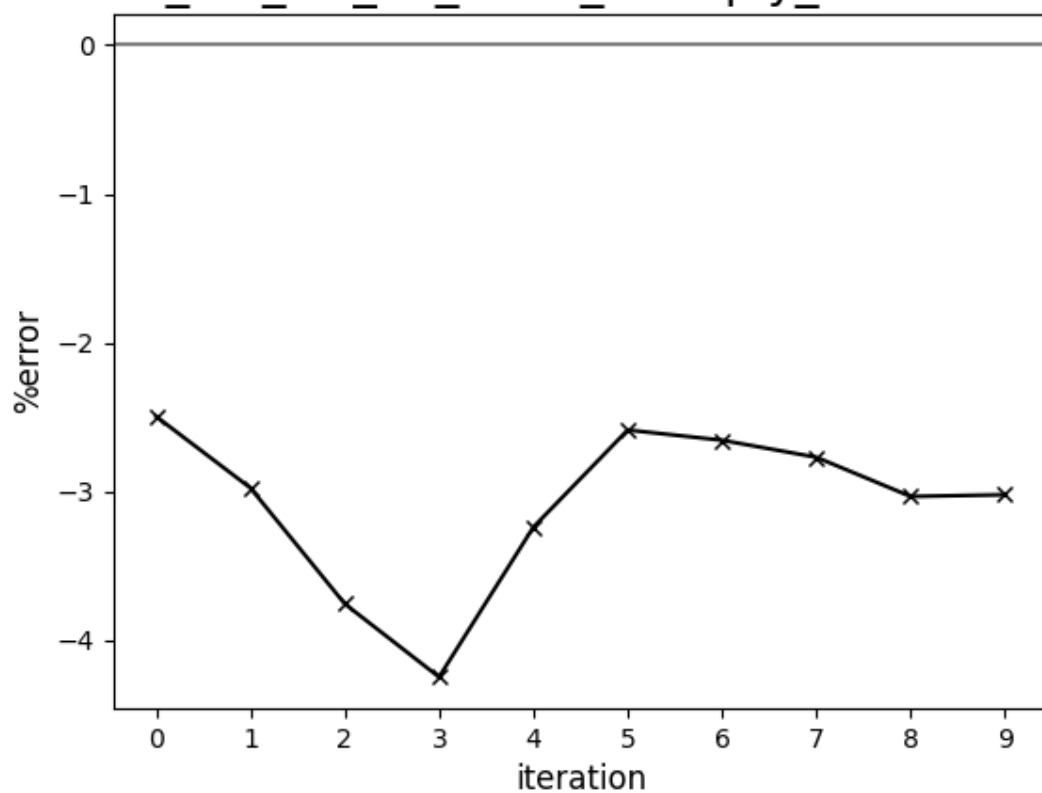
Observations for GPR - scaling: none

Very slight variations in parameters, most test trials share the same parameters: 11.1, 13.7, 4.1, 5.5.

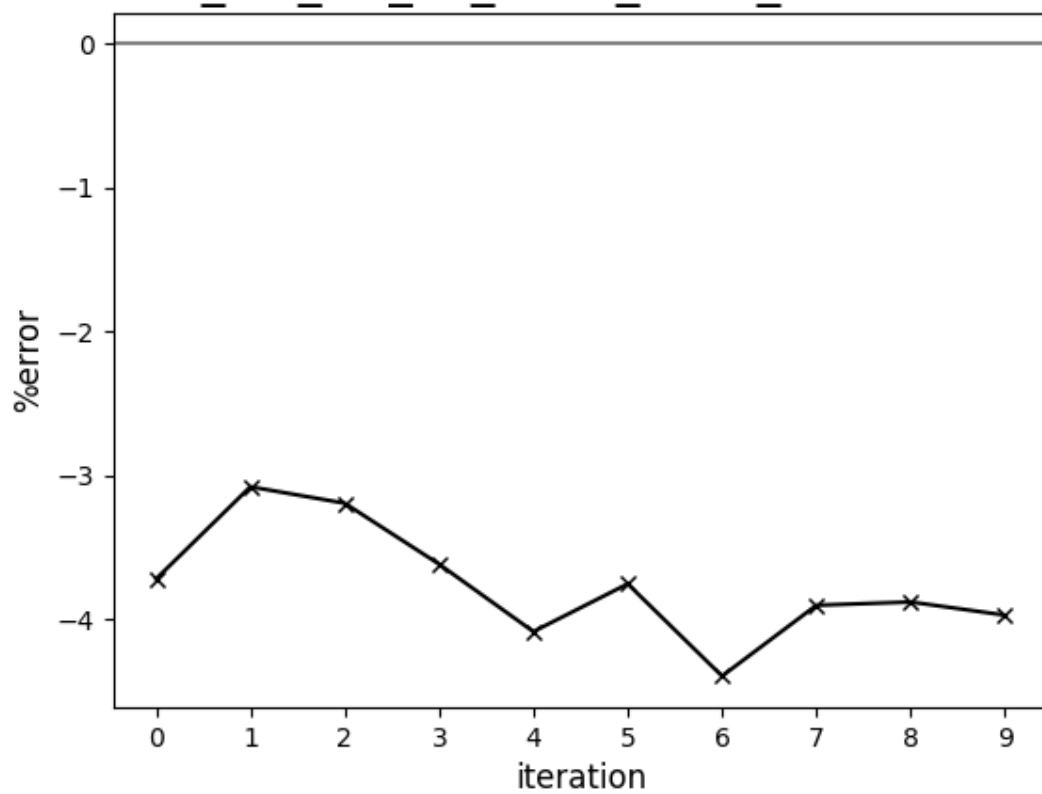
Observations for GPR - scaling: multiply

Some variations between the 2 sets of parameters: 11.5, 7.3, 4.2, 3.6 (percentage error around -3 to -4%) and 15.3, 7.0, 2.6, 4.5.

GPR\_full\_wo\_bo\_scale\_multiply\_unorderedT

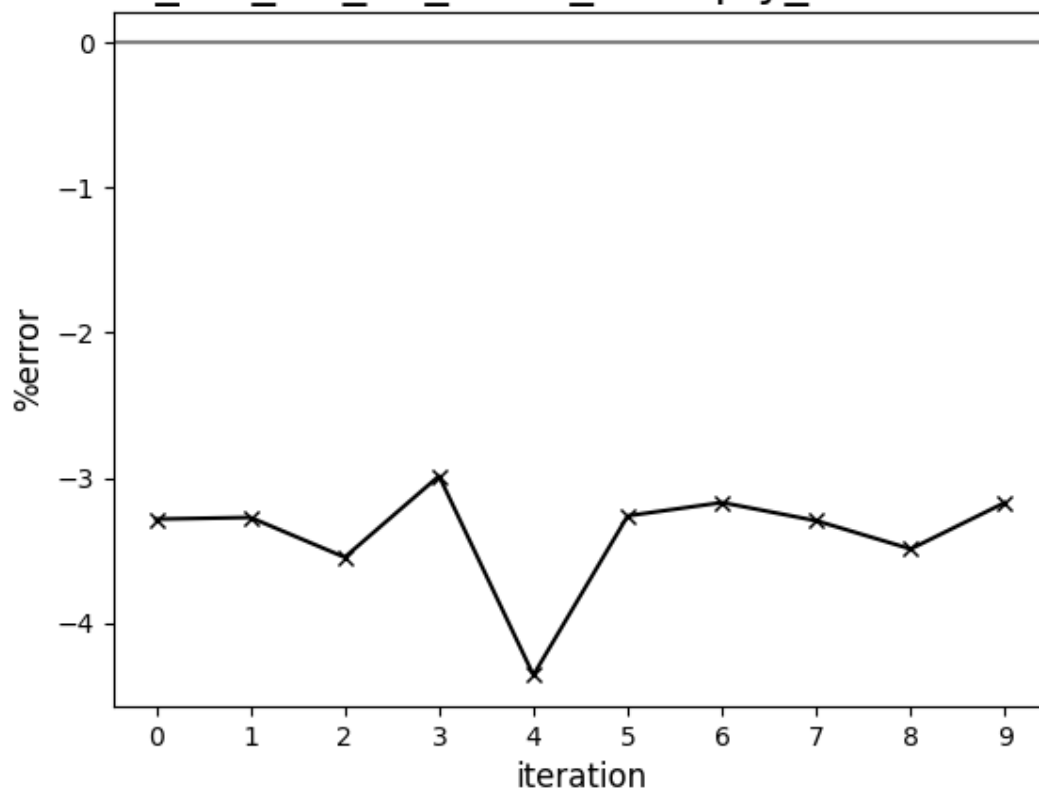


GPR\_full\_wo\_bo\_scale\_none\_unorderedT





LIN\_full\_wo\_bo\_scale\_multiply\_unorderedT



LIN\_full\_wo\_bo\_scale\_none\_unorderedT

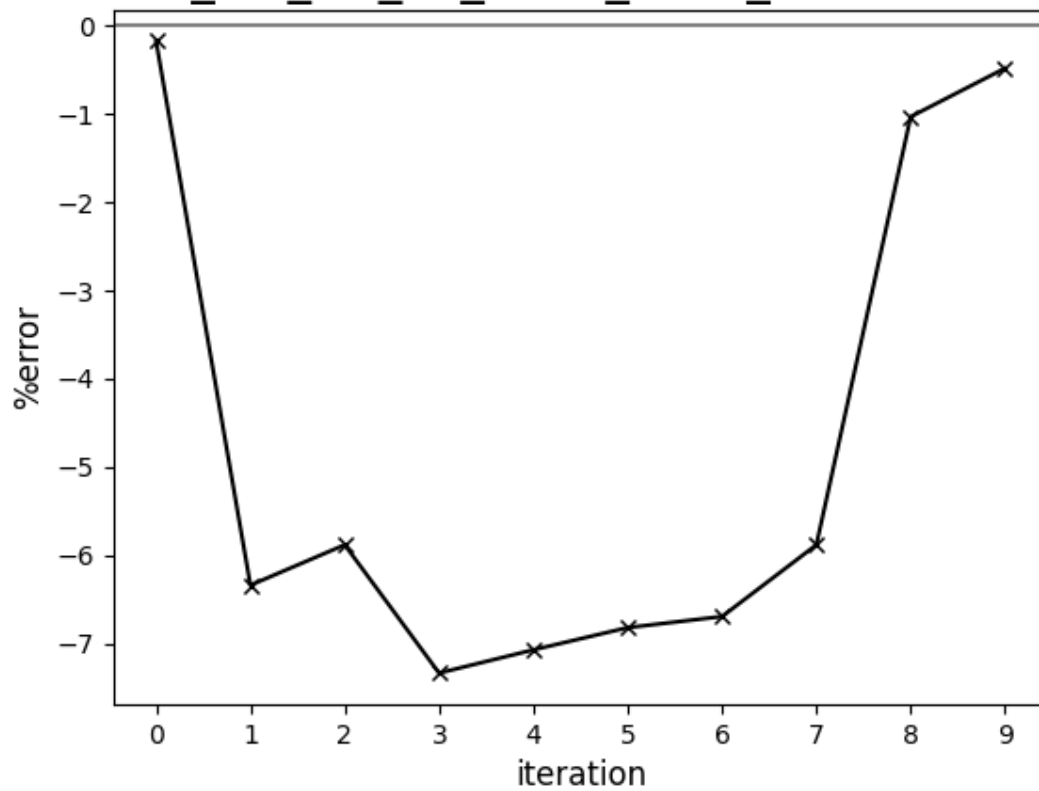
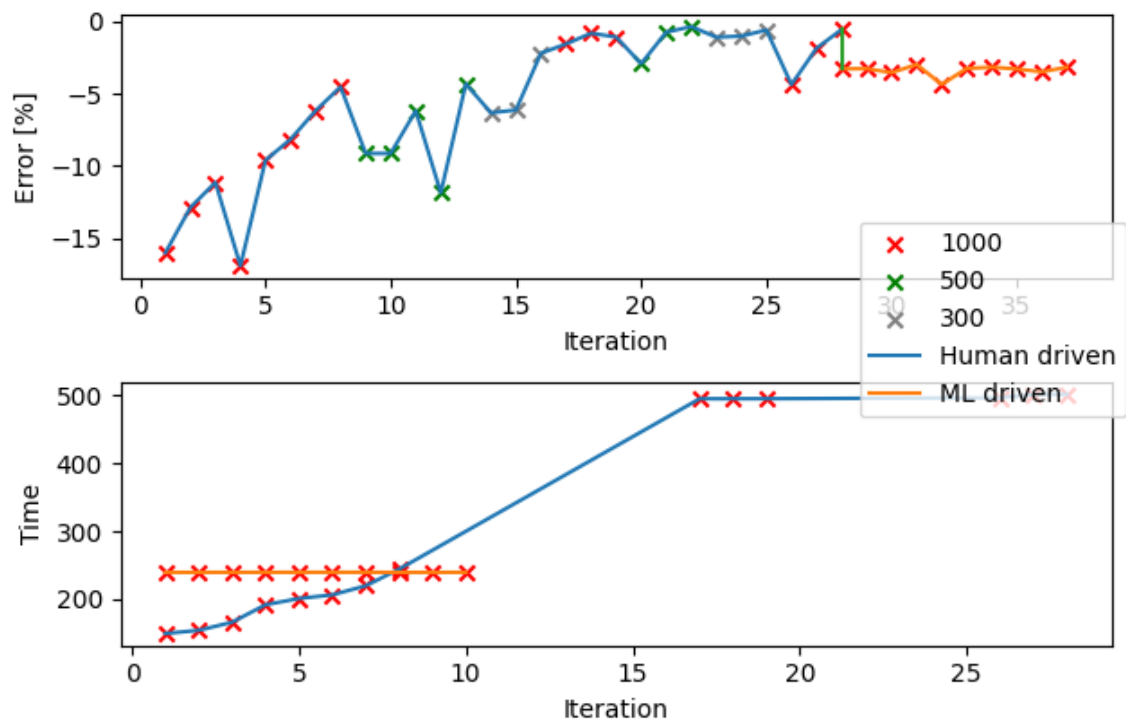
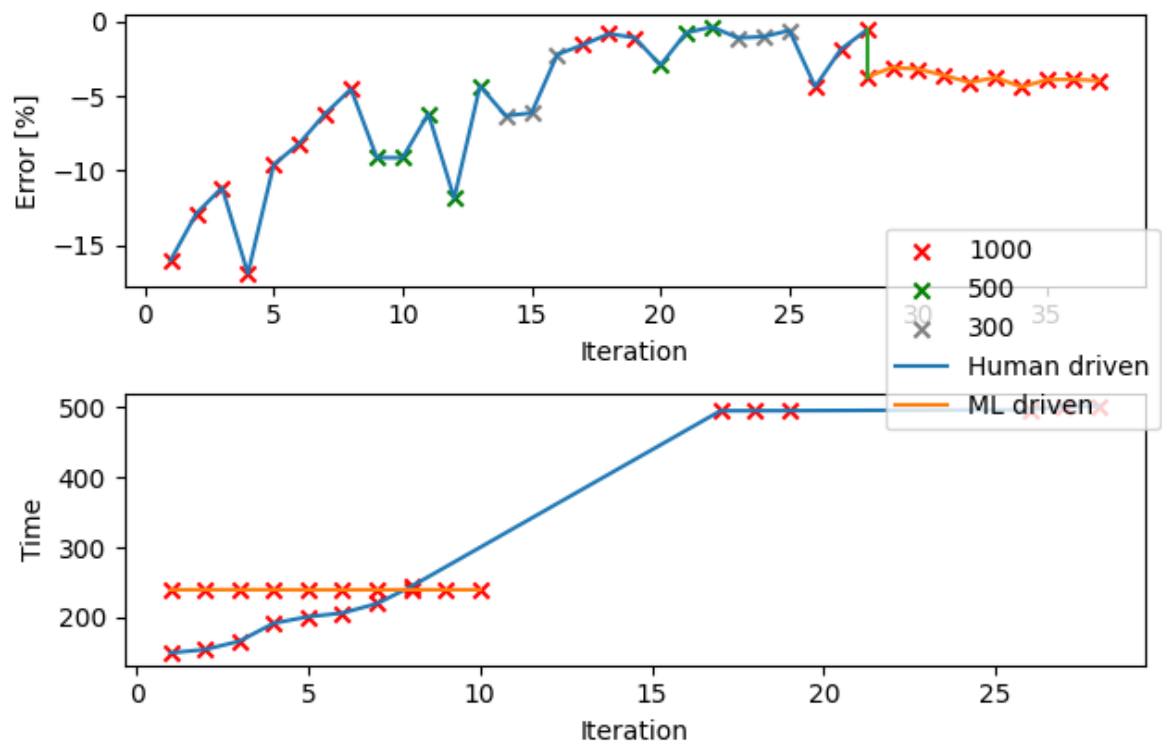


Diagram comparing human-driven and ML test trials (1275 cP, set:full, unorderedT)

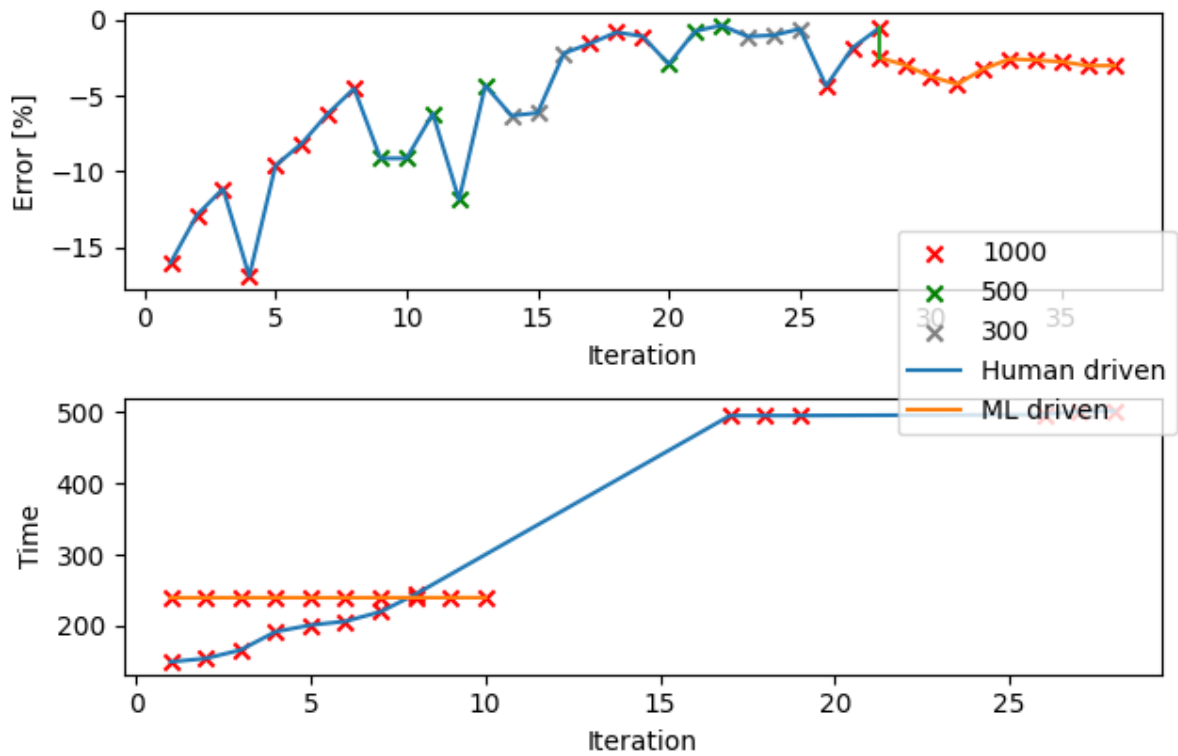
Linear Regression model with slow transfer penalization,  
trained with full data set



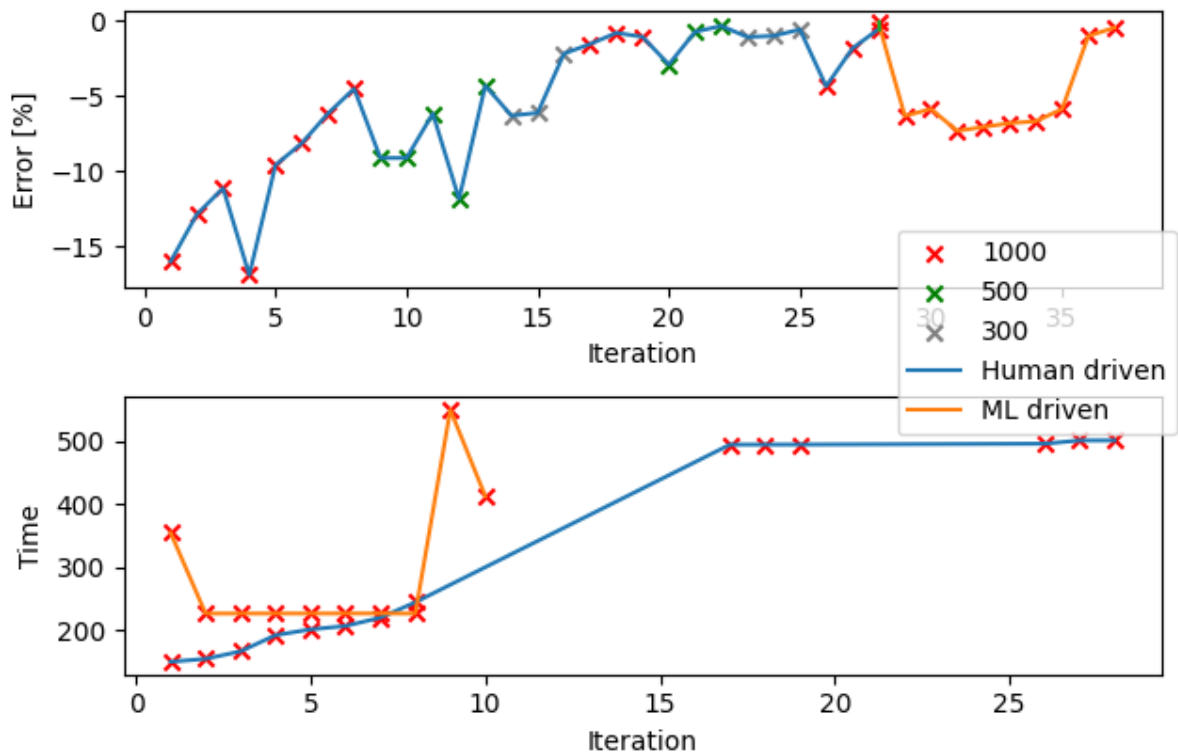
Gaussian Process Regression model with no penalization,  
trained with full data set



Gaussian Process Regression model with slow transfer penalization,  
trained with full data set



Linear Regression model with no penalization,  
trained with full data set



## Set: 1

### Observation of trends

Observation for all iterations:

The parameters generated are more focused on quick transfer times than percentage error. Even though the transfer process will take around 160 seconds as derived from the standard calibration documents, most transfers only took between 90 to 140 seconds, with an average of 100 seconds.

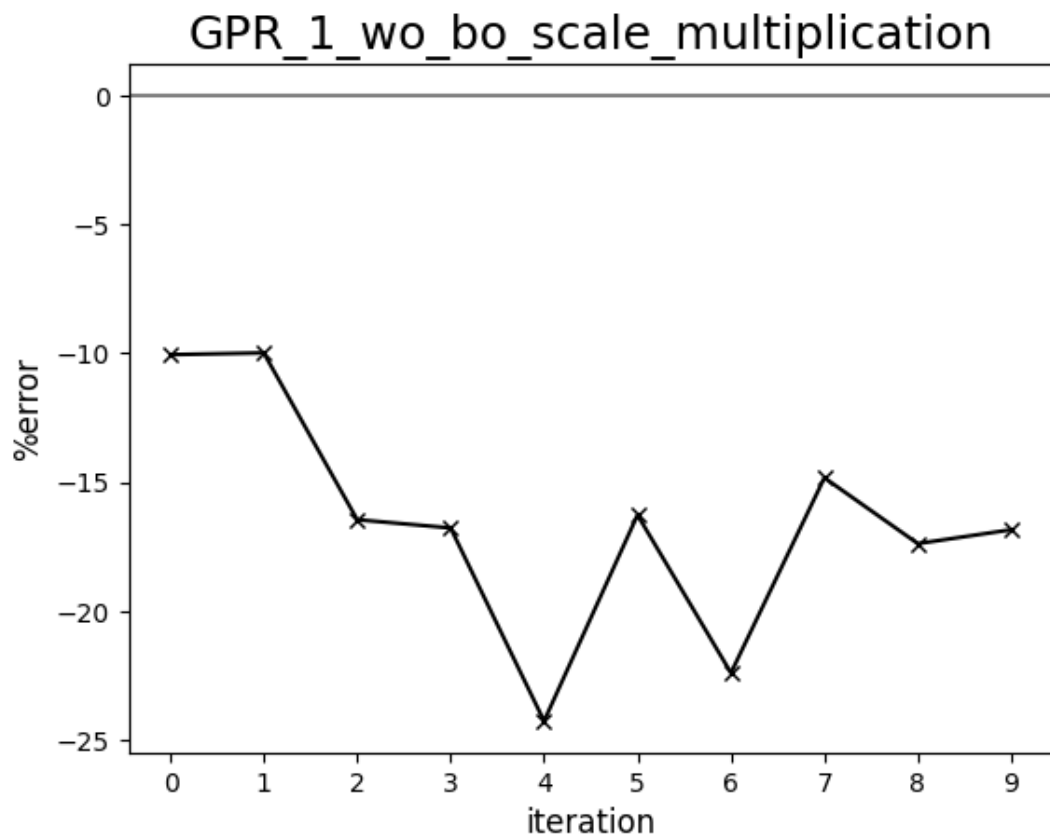
### Suggestion for improvement:

When I was doing the test trials, it was discovered that the first reading acquired will not be accurate. This is because the pipette tip has not been used and it is harder to aspirate and dispense the liquid as compared to one that is already “rinsed” with the standard. Since this is only discovered in the later part of this project, I did not manage to make amendments to the csv when I am still finding the standard calibrations. Thus, by using the first reading as the only point of reference, it might feed the system with the wrong information.

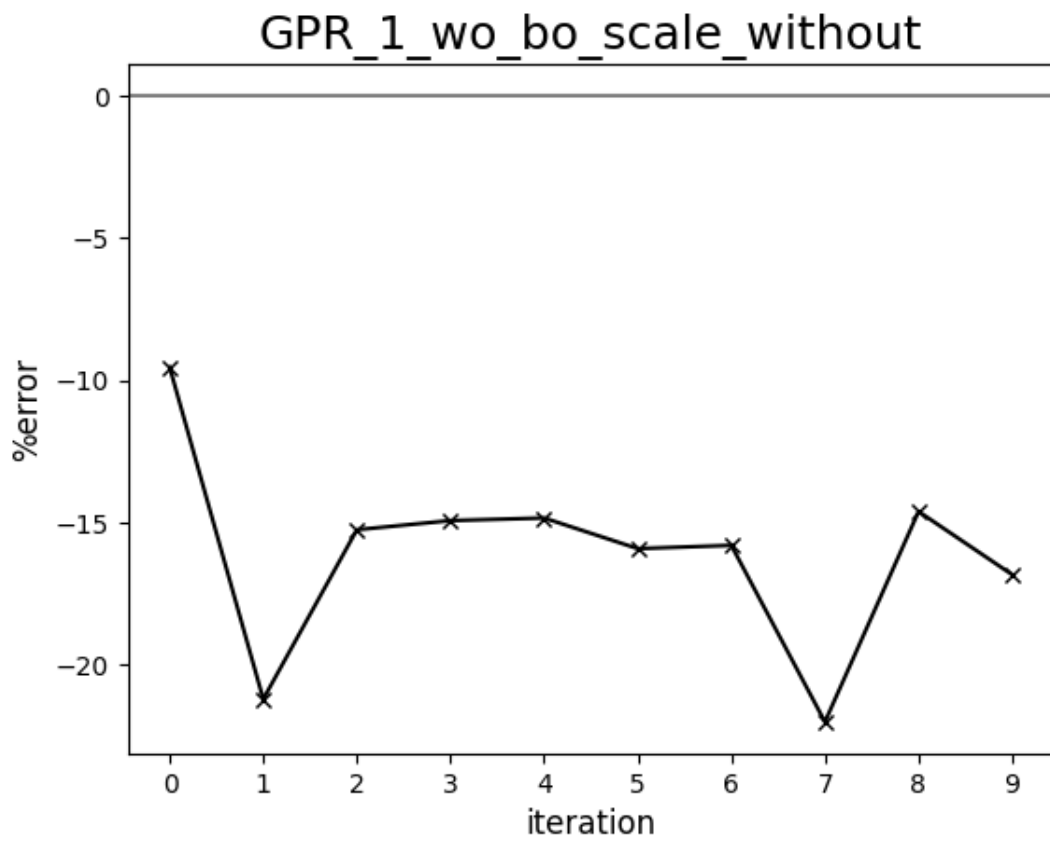
It is also worth noting that most trials have a percentage error of around -8% which is around the same error percentage that is caused by the “clean pipette error”.

One can try to use the second set of data collected instead. Although I’m not sure if it will work because even the trials with the full set of data as the reference points did not give very good parameters, but maybe the first data is prioritised (?) by the algorithm.

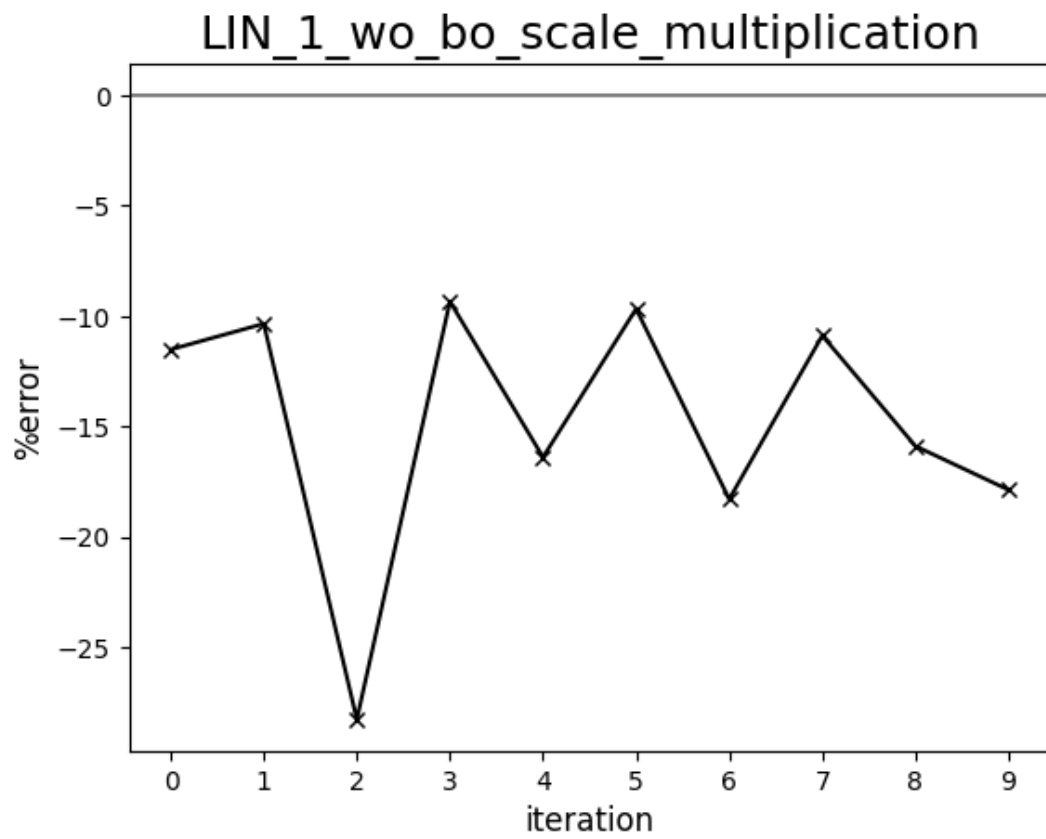
GPR scaling: multiplication



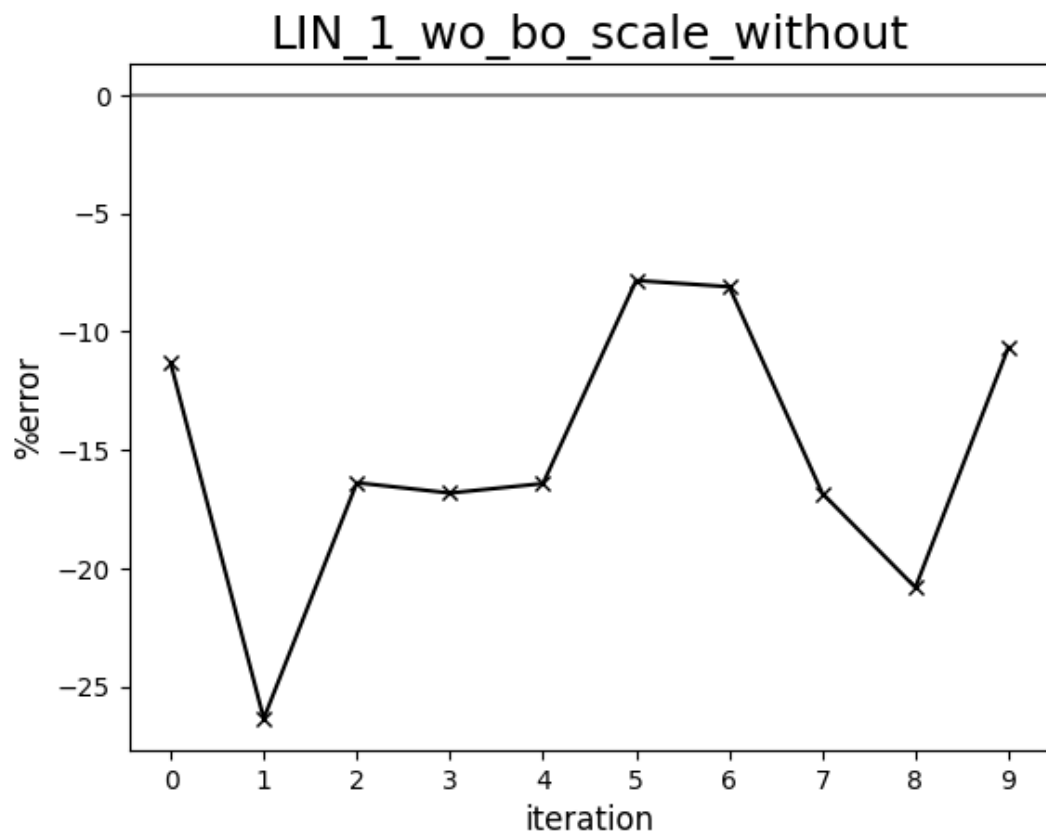
GPR scaling: without



LIN scaling: multiplication

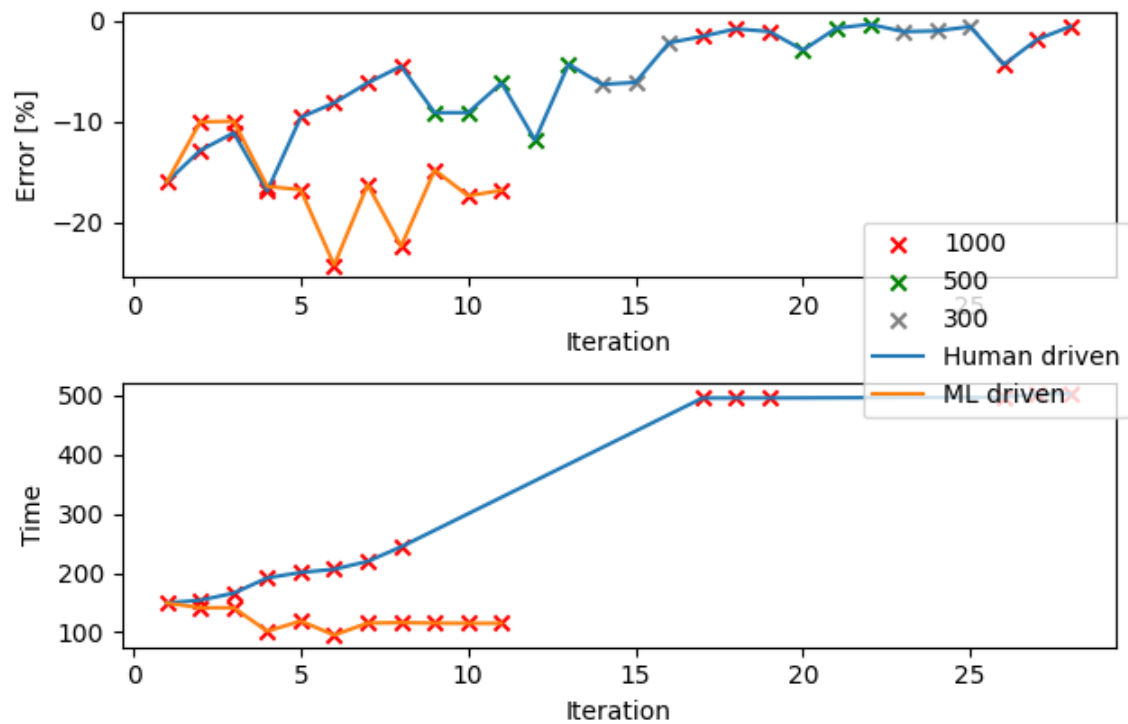


LIN scaling: without

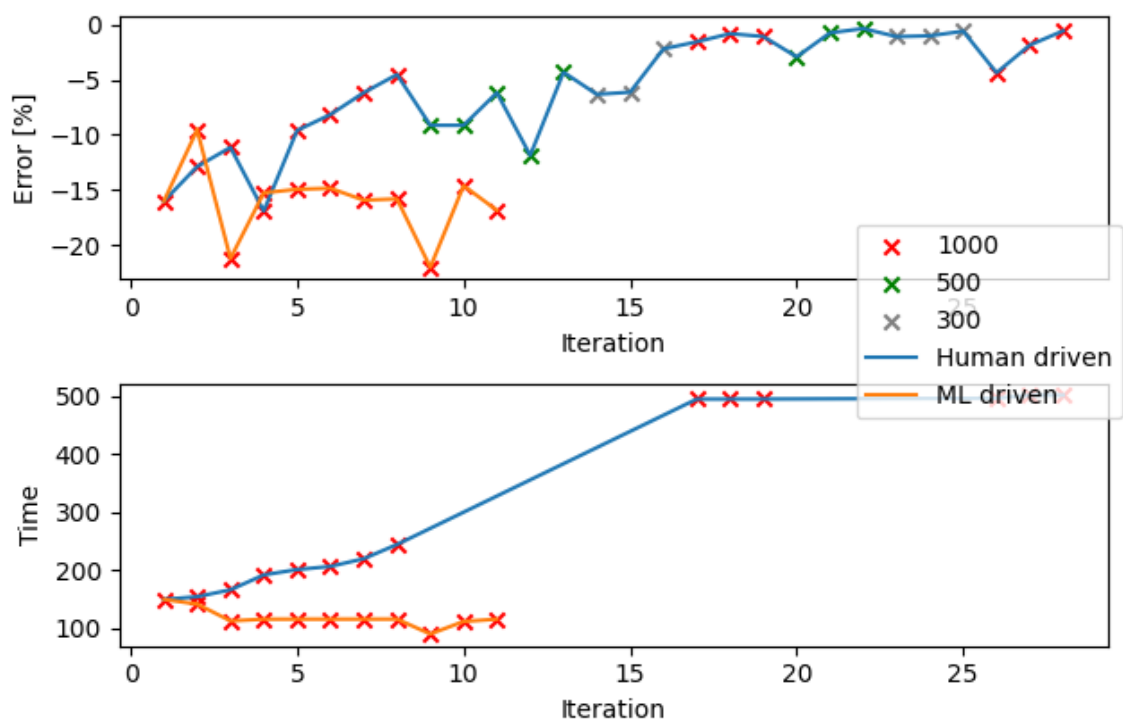


## Diagram comparing human-driven and ML test trials (1275 cP, set:1)

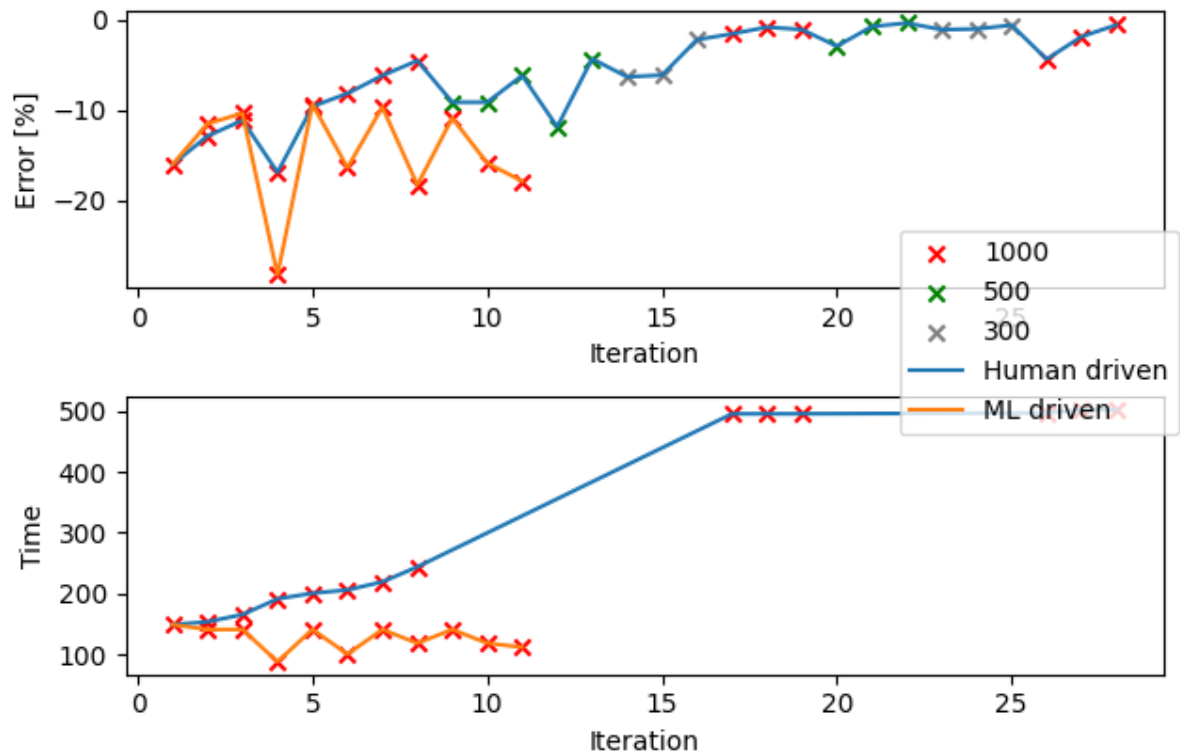
Gaussian Process Regression model with slow transfer penalization,  
trained with 1 initialization data



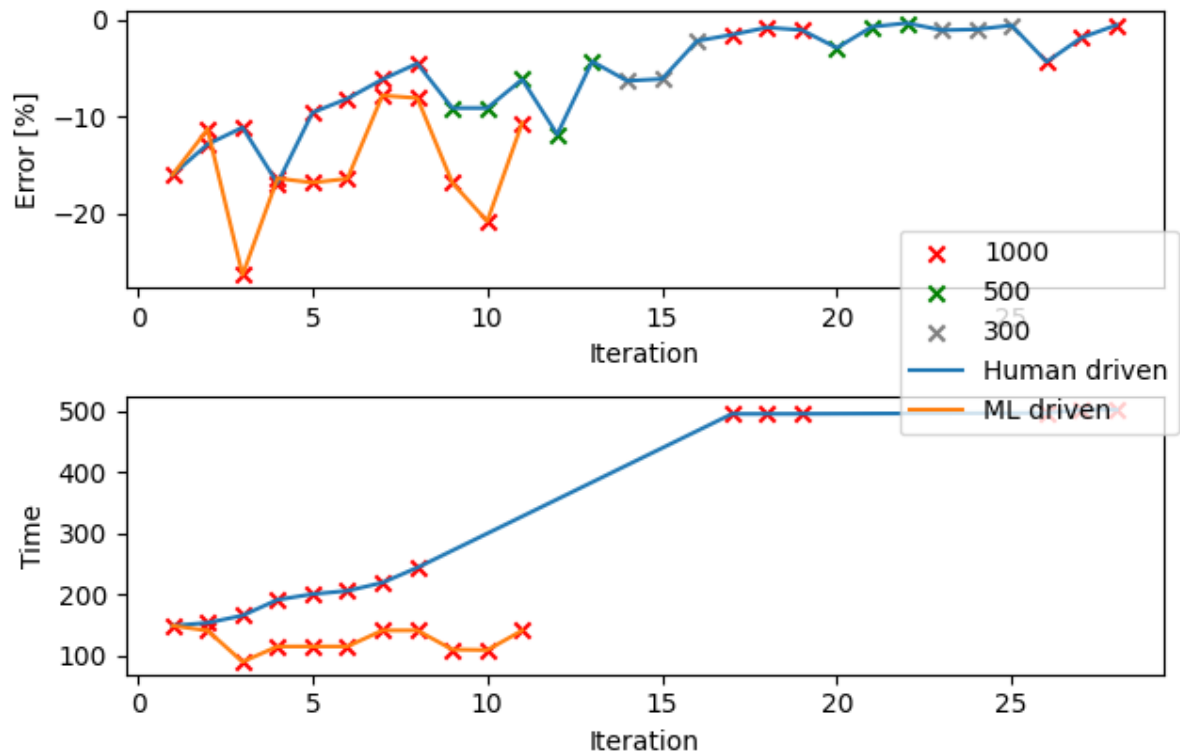
Gaussian Process Regression model with no penalization,  
trained with 1 initialization data



Linear Regression model with slow transfer penalization,  
trained with 1 initialization data



Linear Regression model with no penalization,  
trained with 1 initialization data





## Set: 1 (unorderedT)

### Observation of trends

Observation for LIN - scaling: none

Mostly between these 2 sets of parameters: 13.8, 18.6, 4.3, 2.9 (with a percentage error of around -8 to -11%) and 20.9, 18.6, 3.1, 2.8 (with a percentage error of around -18 to -20%)

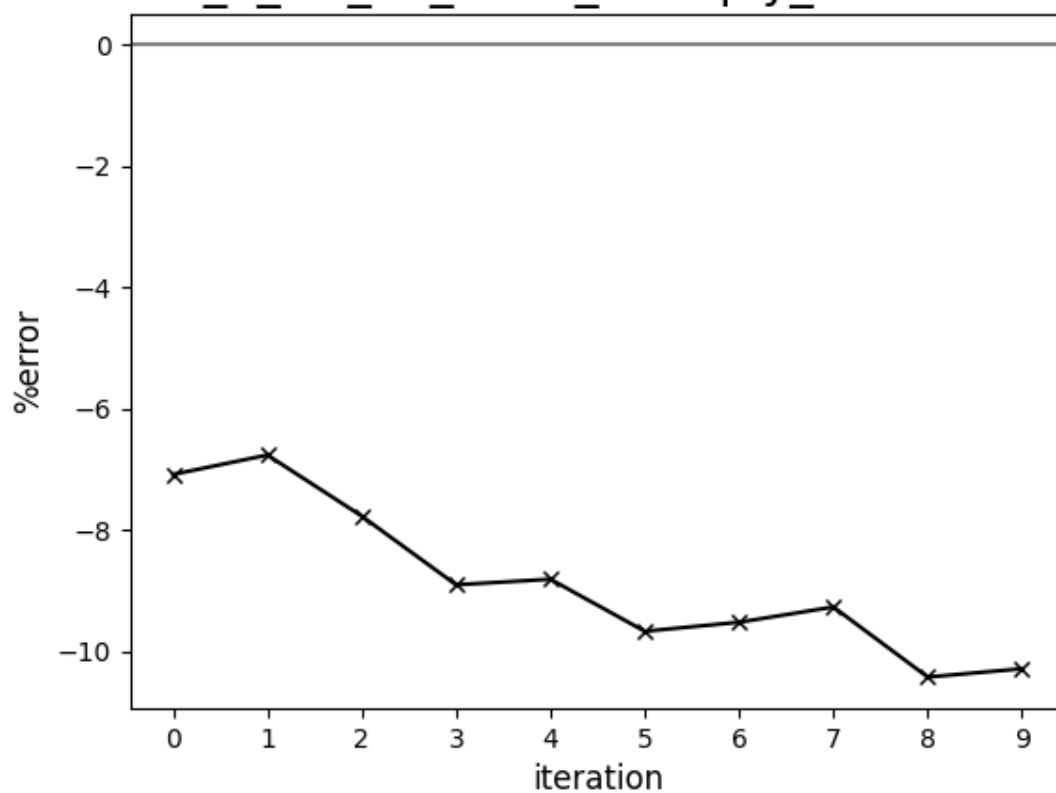
Observation for LIN - scaling: multiply and GPR - scaling: none

Most test trials use this set of parameters: 20.9, 18.6, 3.1, 2.8 and 26.4, 24.3, 0.1, 2.9, very inaccurate sets of parameters.

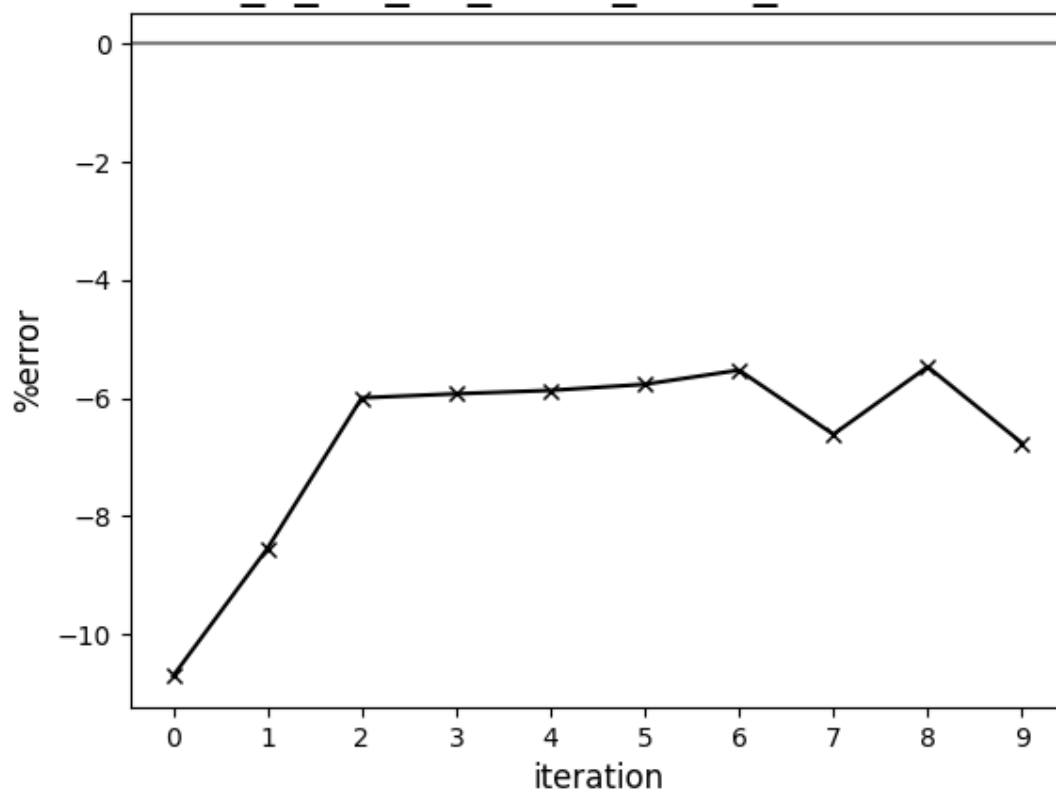
Observation for GPR - scaling: multiply

It has a blend of all 3 sets of parameters that are commonly used in set: 1 test trials.  
13.8, 18.6, 4.3, 2.9 & 20.9, 18.6, 3.1, 2.8 & 26.4, 24.3, 0.1, 2.9

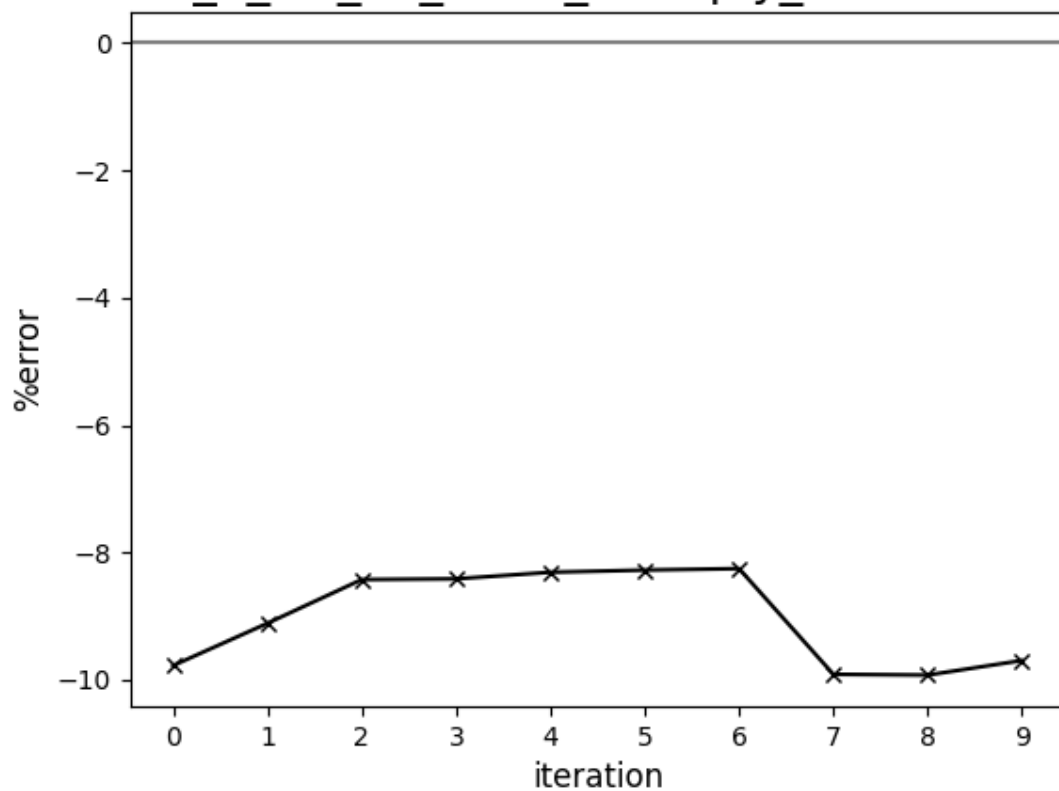
GPR\_1\_wo\_bo\_scale\_multiply\_unorderedT



GPR\_1\_wo\_bo\_scale\_none\_unorderedT



LIN\_1\_wo\_bo\_scale\_multiply\_unorderedT



LIN\_1\_wo\_bo\_scale\_none\_unorderedT

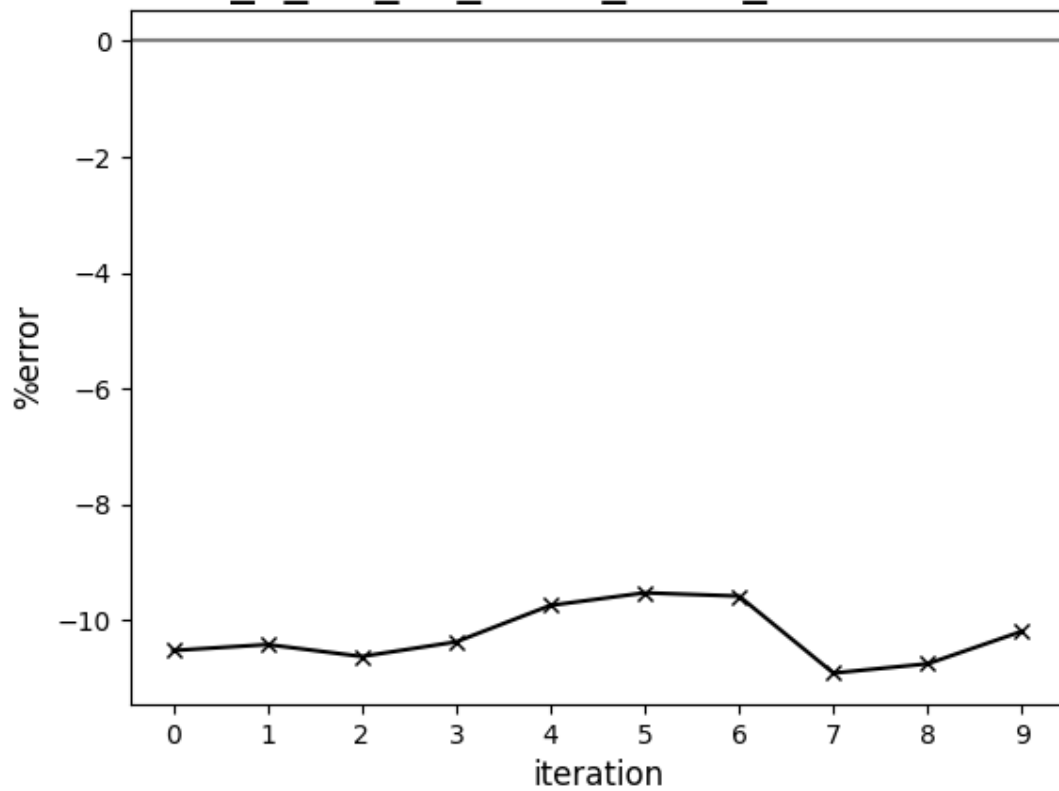
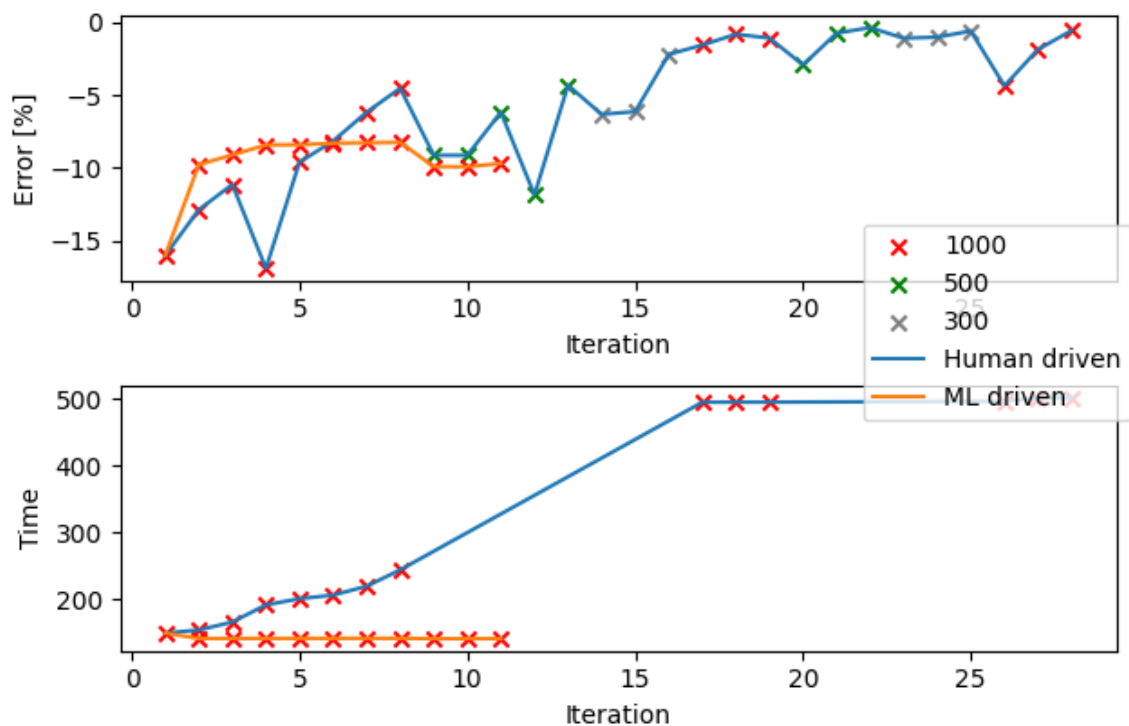
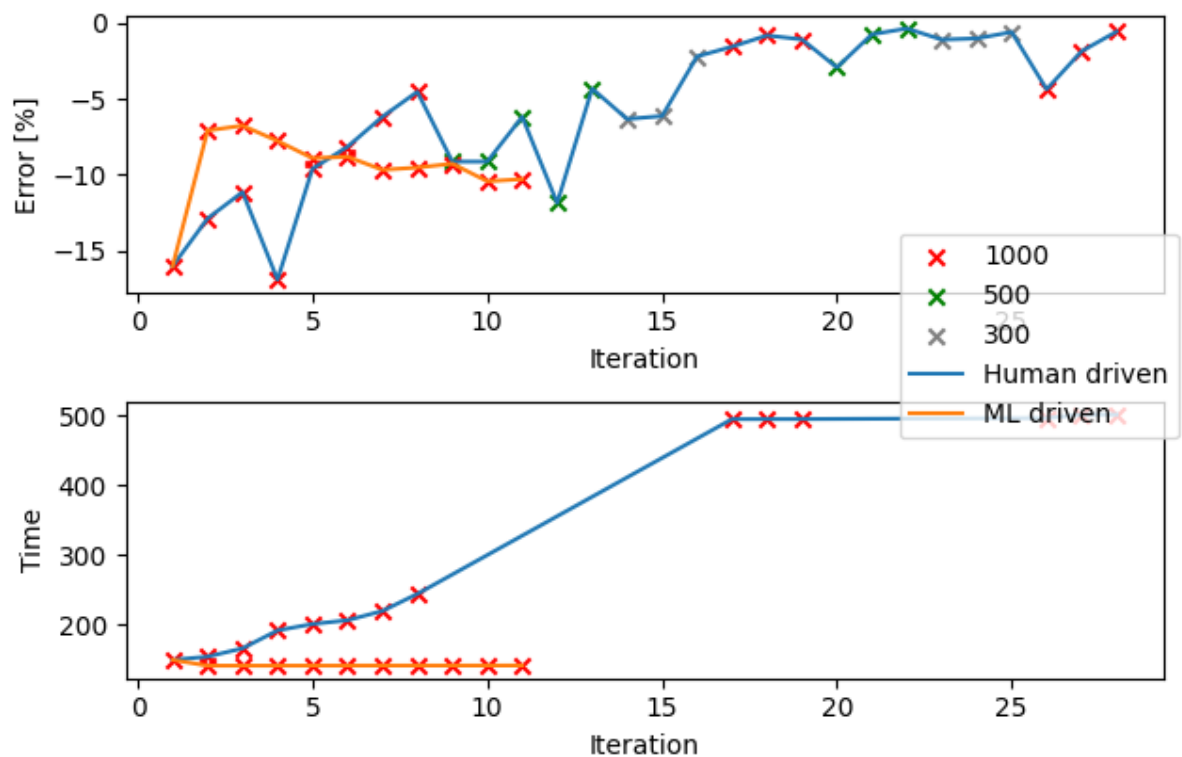


Diagram comparing human-driven and ML test trials (1275 cP, set:1, unorderedT)

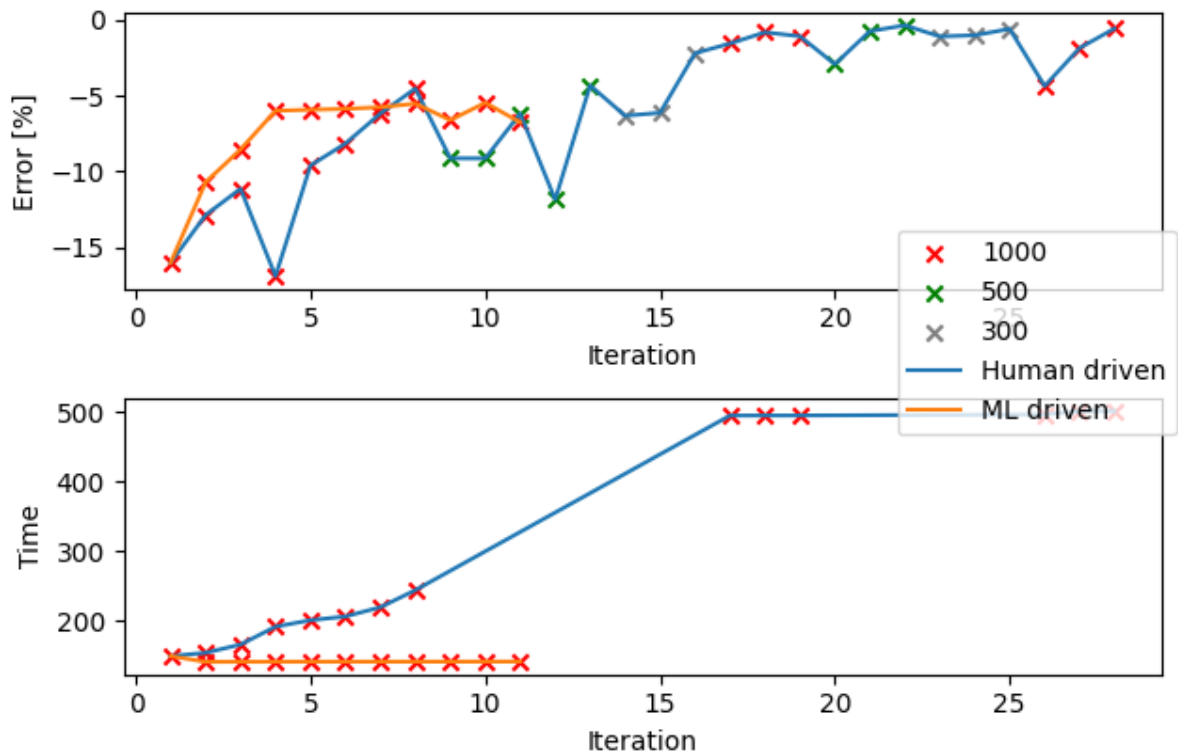
Linear Regression model with slow transfer penalization,  
trained with 1 initialization data



Gaussian Process Regression model with slow transfer penalization,  
trained with 1 initialization data



Gaussian Process Regression model with no penalization,  
trained with 1 initialization data



Linear Regression model with no penalization,  
trained with 1 initialization data

