Goals of this week:

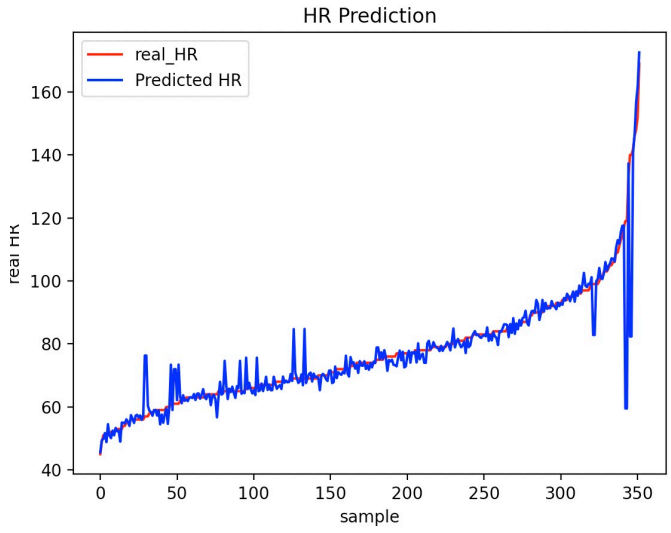
1. (Add Gaussian noise)Data augmentation
2. Create more data with sliding window
3. Learn more about app

First, I think if I reverse the 1d signal, we can label it as the same heart rate as the original sequence. So I try to reverse it add add it to our data set: the result is as following

```
test set
mse:     199.831426
rmse:   14.136174
mae: 10.870764
train set
mse:     5.698946
rmse:   2.387246
mae: 1.777699
```

So we can see it the performance of test set is a little bit better than before.
But later I thought that this method will break causality. So, I removed that part

Then I started to add noise to data set to do data augmentation
At the first, I add noise to data and stack them together **before train_test_split()**, and it gives me a very good performance




```
test set
mse:     50.351006
rmse:   7.095844
mae: 2.573241
train set
mse:     5.111172
rmse:   2.260790
mae: 1.638261
```

This could be due to something like signal after adding noise in test set and original signal in the training set, so it's not correct I think

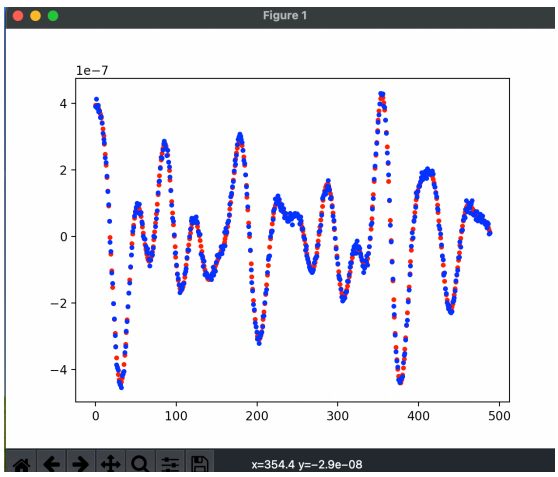
But I thought that we should **only do this data augmentation with training set**

So I move this part to **after train_test_split()** and **before normalization**

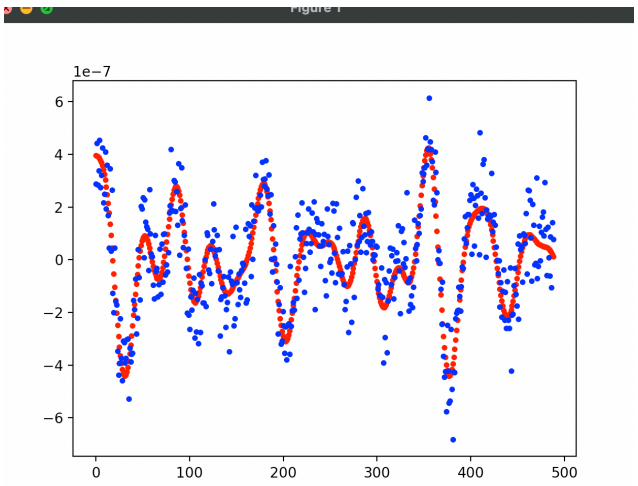
And performance goes back to

```
 [ 83.01005 ]]
test set
mse:      271.064522
rmse:    16.464037
mae: 12.201445
train set
mse:      5.933924
rmse:    2.435965
mae: 1.810722
```

The one signal before adding noise(red) and after adding noise(blue)
sigma = 1e−8 (we use this one)



sigma= 1e−7



The analysis:
The main task of our algorithm is to find the peaks of waveform
Adding noise to waveform signal will hurt the smooth of the signal. It's not so good because our preprocessing method is trying to smooth the coarse signal

After sliding window