Exercise Nr. 4

a) ESN implementation

We implemented trainESN and the differential evolution algorithmin DELearningESN. However we could not achieve an error below 1.5, which we assume is still much higher than a correct implementation. We implemented biases, output feedback scaling and played around with different hyperparameter settings: reservoirsize (10,20,30), size of dataset between 500 and 10000 datapoints with different train/test splits, OUTPUT-FEEDBACK-SCALE (10e-8, 10e-10, 10e-12). Despite all this, we could not bring the error down and also the trajectories remained very zig-zaggy and weird. We conclude that there must an additional bug in our code that completely breaks the training but we could not find it despite hours of searching.

b) Data Acquisition and Training

I think we recorded the data accurately using the relative position but as mentioned above we were unable to achieve a low RMSE.

c) Motion Prediction

We integrated the ESN module in the simulation step. We took care to use the enemyesncopy for predictions to not interfere with the internal state of enemyesn. We also reset the enemyesncopy in each timestep after predicting. We also added the origin to the predictions to convert the relative predictions into absolute positions. However, as mentioned above, the trajectories still looked all over the place. Because we were unable to achieve a low RMSE, we also could not be sure whether this part is correct or if we missed something because in any case the trajectory wouldn't be correct.