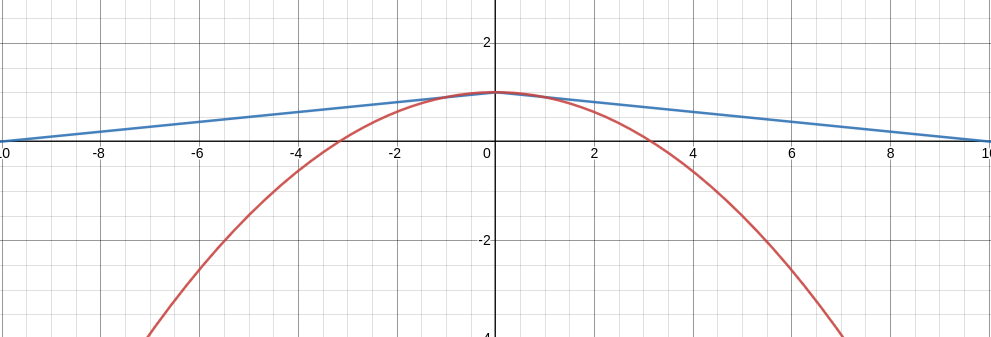
**Friday, October 20, 2023**

I got reinforcement learning using PPO working with the simulator. I create a reward function which penalizes cross-track error (straying from the centerline of the road) and also penalizes collisions. If there is no collision/terminated episode, then the reward is equal to the forward velocity of the car (It's clipped to 0 if going backwards). After 5000 iterations the car was able to make it about 1/3 of the way through the track before crashing.

I tried adjusting the reward function so that it more harshly penalizes distance. Here is an example:



The blue is the old reward function for cross-track error. If the upper bound on cross-track error is 10, it will linearly scale. However, quadratically scaling drastically reduces the reward once you get farther from the centerline. This value is multiplied by the forward velocity of the car to obtain the final reward.

**Monday, October 23, 2023**

On Friday, I was training for around 100 – 5000 timesteps at a time before terminating the session. Today, I decided to train the RL for longer (over most of the class period) to see if it would converge to a decent policy. It is important to note that right now, I’m just using a normal CNN as the policy. Once I obtain a working result using the CNN, I will implement the autoencoder in its stead. Then, the autoencoder’s encode and decode stages will both be trianed simulatenously during the reinforcement learning. Finally, I will manually record driving data to “jumpstart” the encoder model. This should significantly reduce the amount of iterations it takes to create a working driving policy.

**Wednesday, October 25, 2023**

I was sick (not at school)

**Thursday, October 26, 2023**

I ran the reinforcement learning for a few hours to see the results. I will hopefully check my results tomorrow to see how well it does. Also, the proper camera has arrived.