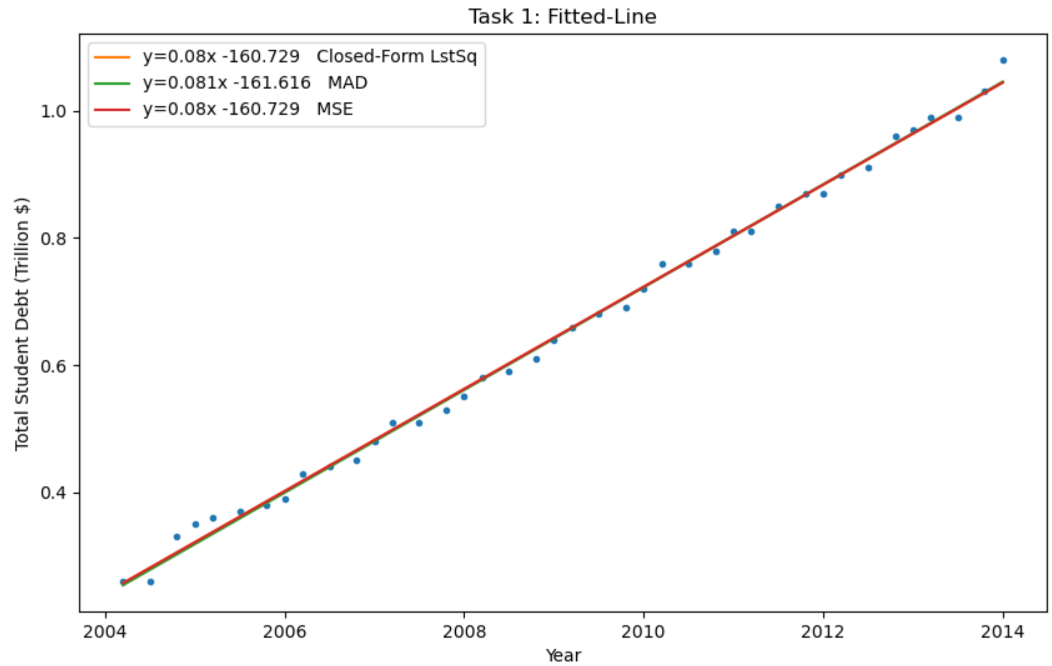
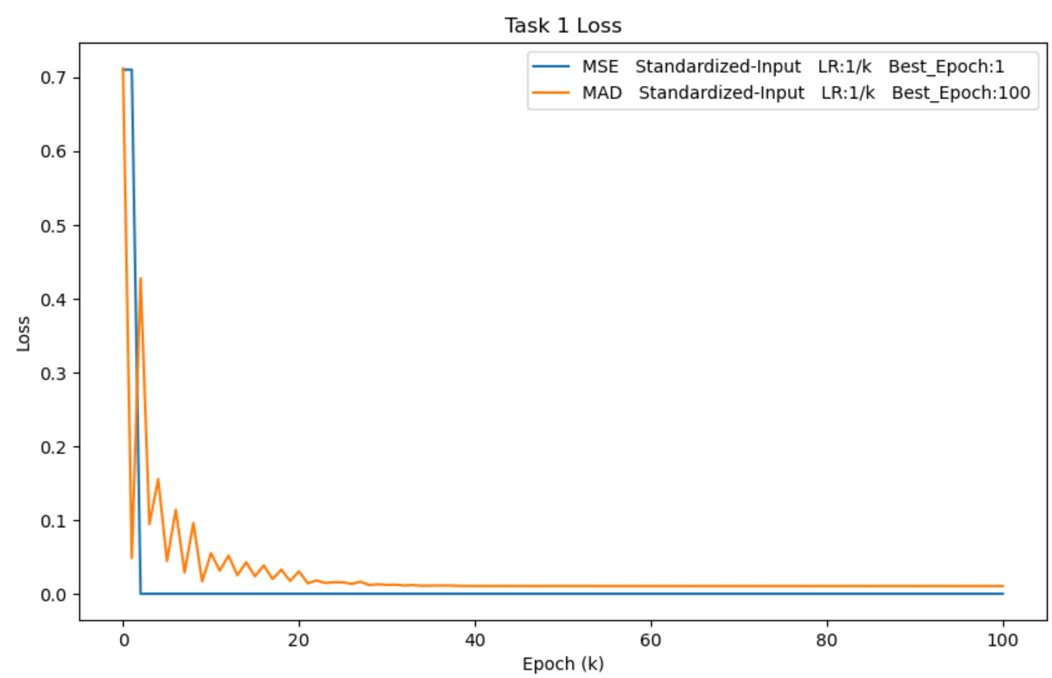
Andrew Hall
CS589 HW3 Report

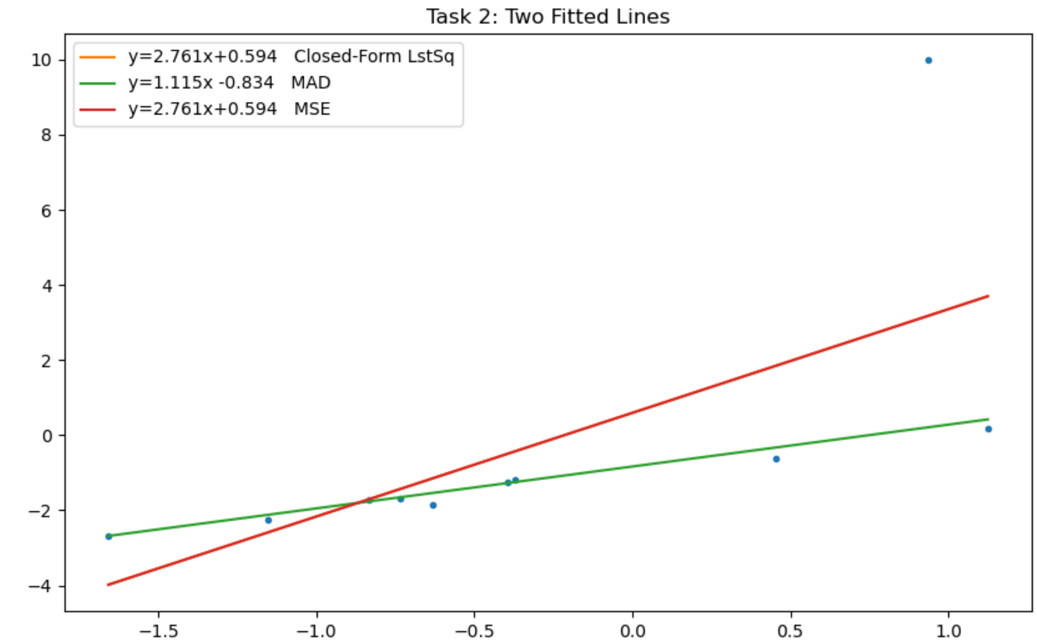


Student debt is projected under this model to reach **2.33 Trillion by 2030.**

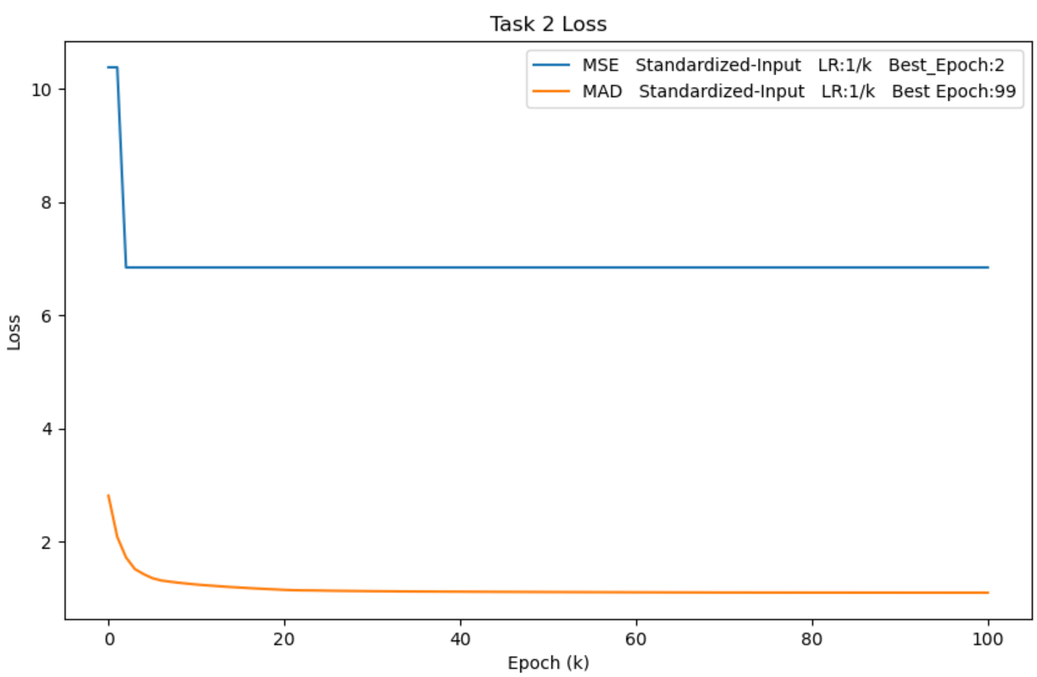
My Linear Regression Model standardizes the training input to smooth the optimization landscape and reduce hyperparameter sensitivity, it also used diminishing learning rate of 1/k, and initialized all weight values to one. With these conditions, gradient descent converged using **Mean-Squared-Error** as a loss function in only one iteration whereas using **Mean-Absolute-Deviation** loss resulted in slower convergence, reaching the shown solution on the 100^{th} iteration. The optimal solution is what NumPy.linalg closed form least squares produces for the input and output on which the model is trained.



Without standardizing the input, the MSE gradient descent almost always diverged unless the learning rate was miniscule (1e-6) and even then, it exhibited explosive divergence until learning rate diminished sufficiently for convergence to occur. MAD also commonly diverged for learning rates starting greater than ~1e-5.



Given this dataset with a lone outlier, modeling with **Mean-Absolute-Deviation** loss captures the main trend of the data more effectively than with **Mean-Squared-Error** loss. While MSE prefers minimizing its largest outliers, MAD treats all deviations proportional to their magnitude. Observe the slight upward shift in the MAD line's bias, as if the line has been gently pulled in the direction of the outlier. Yet, its slope remains consistent with the obvious trend.



The error plot shows that gradient descent using MAD loss converges to a better solution than MSE can achieve when such 'noise' is present.