HW3: Regularization

Marko Ivanovski

Data preparation

The first step in this task was to prepare the data. Since we are using linear regression and penalizing big parameter values I had to make sure that all the features were on the same scale. Otherwise, features that have big values will have bigger parameter values and will get more penalized. For this step, I standardized all the variables such that their mean was 0 and variance 1.

Parameter tuning

The main issue with this task was the fact that we had a lot of features and not a lot of data. In such a scenario, our model is prone to overfitting hence using regularization (L1 and/or L2) is beneficial. However, the task of finding good λ (regularization weight) which would generalize well on a new unseen data was not easy as well. When I performed train validation split and chose λ I noticed lambda varied a lot based on the randomness of the split. Consequently, I decided to implement my own idea which would return a more stable lambda output.

I started off by dividing the data into the first 200 instances for training and parameter validation, and the last 100 instances only for the final testing. Next, I obtained a distribution of optimal λs by repeating a cross-validation evaluation technique 100 times. On every iteration, the data was split into 5 chunks and every chunk was used as a validation set on which I found the best λ. The lambda was chosen from a predefined list:

$$[0.1, 0.5, 0.7, 0.9, 1, 3, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 60, 70]$$

	L1 regularization	L2 regularization
RMSE	18.33±2.61	16.50±2.39
λ	30.48±17.06	11.43±19.04

Table 1. RMSE results obtained on the testing dataset using model trained on λ

In the end, I had a distribution of 500 λs and the final one was chosen using average. Using this distribution I also provided the standard deviation.

The standard deviation on the RMSE score was calculated using the bootstrap technique on the testing data with 100 repetitions.

Results

With the method described above, I managed to get the result in Table 1 on the testing dataset.

When fitting the Lasso regularization, I noticed the solution was also not quite stable and depended on the starting point. Consequently, as a starting point, I used a vector of zeros since it is as close as we can get to the final solution and it converged the fastest.