My code works like the following for both regression and classification:

They divide the training data into 2, so I can use it for cross validation

Then I test from the range of 1~100 the most optimal multiple of 0.05 so that when multiplied to a minimal normalized variable generates the highest cross validation accuracy.

Chart, line chart

Description automatically generated

For example, when multiplied by 0.05\*21 Hillshade\_Afternoon generated optimal result.

This operation is done around 3 times, and then by using the generated “weights” I use it to create the final preprocessed test\_data which I can use knn algorithm for the classification and regression models.

However, for regression, the lack of test cases made it hard for the cross validation to create optimal models, so I had to try the suggestions made by the cross validation model one by one.

The python file I have posted creates csv files that are approximately the same score as the highest I have submitted; Not exact as the ones I have scored the highest on had some randomness incorporated during the training/test set splitting process.

It takes around 20~30min for the classification, and around 2~3min for the regression.