In order to find an optimum model for the given training data set that generalizes to test data sets different combinations of cross validation with repeated k-fold were used. The base model used no k-fold repetition at all and achieved a public score (PS) of ~9.3. Implementing k-fold with 10 repetitions led to a PS=~14.3, whereas 1000 repetitions yielded PS=~13.7. Therefore, the computational burden of k-fold repetitions is not worth it taking these results into consideration.

The above results were all achieved by setting the Ridge() function parameter fit\_intercept=True, which is the default setting. However, the same tests as above were then run setting fit\_intercept=False which is the kind of optimization problem we want to solve in this project. This implementation led to much better results when applied together with repeated k-fold. Using 10 repetitions led to PS=~2.5 (vs. 14.3) and 1000 repetitions yielded PS=~0.14 (vs. 13.7). Hence, the latter improved the result by a factor of 100.

Given the above experiments and results it is possible to draw the conclusion that setting fit\_intercept to False, which corresponds to the “vanilla” version of ridge regression, yields indeed much better results. Furthermore, it can be concluded that a repeated k-fold approach can boost performance. The 1000 iterations applied in the hand-in code improved the public score from ~9 to ~0.14.

<https://stackoverflow.com/questions/46510242/how-fit-intercept-parameter-impacts-linear-regression-with-scikit-learn>