In order to simulate the actual situation, cross-version defect prediction is adopted. That is, software defect prediction models constructed according to one version are used to predict

defects of the next version.

According to the research questions, we need to implement the learning-to-ranking approach, linear regression, ridge regression, lasso regression, and the two multi-objective approaches. The involved objectives included fault-percentile-average(FPA), mean square error (MSE) and number of nonzero (NNZ), which are described in Section III. All methods are implemented in Python. Specifically, the two multi-objective approaches, NSGA-II and the revised NSGAII are implemented based on Geatpy2, which is a genetic and evolutionary algorithm toolbox for Python with high performance. The parameters of the compared methods are simply set according to previous work [12], [14], [15], and the parameters for the multi-objective approach are simply set like the learning-to-rank approach [14]. The feasible solution space is set as Ω =Πd i=1[􀀀20; 20], and the population size and maximal generation are set to 100.

Besides, linear regression, ridge regression and lasso regression models are implemented based on scikit-learn, which is a popular software machine learning library for the python programming language. Default parameters are adopted for linear regression. In ridge regression model, the value of alpha has an impact on the performance of models. Thus, we use RidgeCV with alpha values ranges from 0 to 1000 at 0.1 interval, to build ridge regression models with a relatively suitable alpha. And in lasso regression model, the parameters are optimized by least angle regression. Like Ridge regression, the parameter alpha is also important. Thus, we use LassoLarsCV, which set the alpha parameter by cross-validation to build lasso regression model. In this method, we use 10-fold cross-validation, and no more than 1000 points on the path are used to compute the residuals in the cross-validation.

And the lasso shrinks the ordinary least-squares estimator

towards zero and potentially sets   
i to zero for some i , so

it may build sparse defect prediction models.