Multivariate Feature Selection, as it currently exists, occurs in the form of wrapper methods or embedded methods. Wrapper methods specifically utilize a classifier to externally estimate relative to combinations of features in order to select an ideal feature subset relative to a particular method's conditions. Current primary wrapper methods in Scikit-learn include the Sequential Feature Selector, which sequentially builds a subset by adding the next best feature or deletes the worst feature at each step in order to obtain a feature subset of the selected number of features by the user; and Recursive Feature Elimination, in which an external estimator is applied on a feature subset at each step and the feature with the lowest gini importance(decision tree-related metric) calculated relative to the particular classifier/estimator is eliminated. This process continues recursively in Recursive Feature Elimination until the user selected number of features is obtained.

Our novel development is the MultivariateFeatureSelector. The MultivariateFeatureSelector is a fusion of sequential forward selection with ksample testing via multivariate independence testing. Mechanically, the user-specified multivariate independence test acts as a wrapper object that builds the feature subset by determining the next best feature. The test is performed at each iteration on the already obtained features with each additional feature not yet selected. The additional feature associated with the subset with the best test statistic is chosen as the next best feature to add and this process continues until the feature subset of the amount of features specified by the user is obtained.

MultivariateFeatureSelector has shown baseline feature selection method functionality on scikit-learn simulations and has shown comparatively better performance as a transformer in the classification pipeline relative to univariate feature selection in certain situations. For instance when there are many classes and/or highly chaotic data per class, MultivariateFeatureSelector has shown better performance. Additionally, when feature sizes are very high as in genetics datasets, MultivariateFeatureSelector often captures multivariate dependencies that univariate-based filtering misses. MultivariateFeatureSelector has also demonstrated its ability to capture multivariate dependencies on simulations designed from Neuro Data Design Lab in which our method is able to build a subset including redundant features that link features that have a high capability of determining class when combined together. Directly in comparison to univariate selection which statically filters to generate a subset and performs worse.

Currently, the only multivariate independence test in Scipy is Multiscale Graph Correlation (MGC) test and so that was the test used in our use cases. Neuro Data Design Lab is currently pull requesting Distance Correlation(Dcorr) Test into Scipy and will have a successful pull request within months. When Dcorr is pull requested, and there becomes multiple user options for a multivariate independence test, MultivariateFeatureSelector will be pull requested into scikit-learn in the feature selection module.