**Case Study on Scikit-learn**

Scikit-learn is an open-source machine learning application. It has simple and efficient tools for data mining and data analysis. It’s easy accessibility and reusability in various contexts makes Scikit a popular tool to use.

**Technology & Platform used for development**

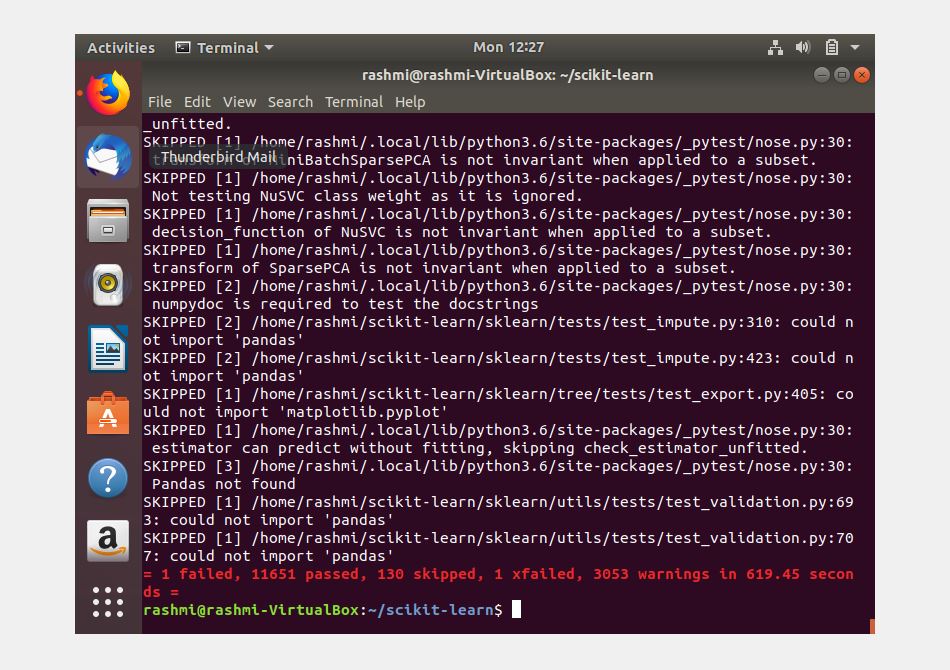
Scikit-learn is built in Python. The latest version of scikit requires Python 3.5 or above. The choice of Python is appropriate due to its rich math and scientific libraries like Numpy, Scipy and matplotlib. Scikit uses Numpy 1.11.0 or above and Scipy 0.17.0 or above.

**Build System used**

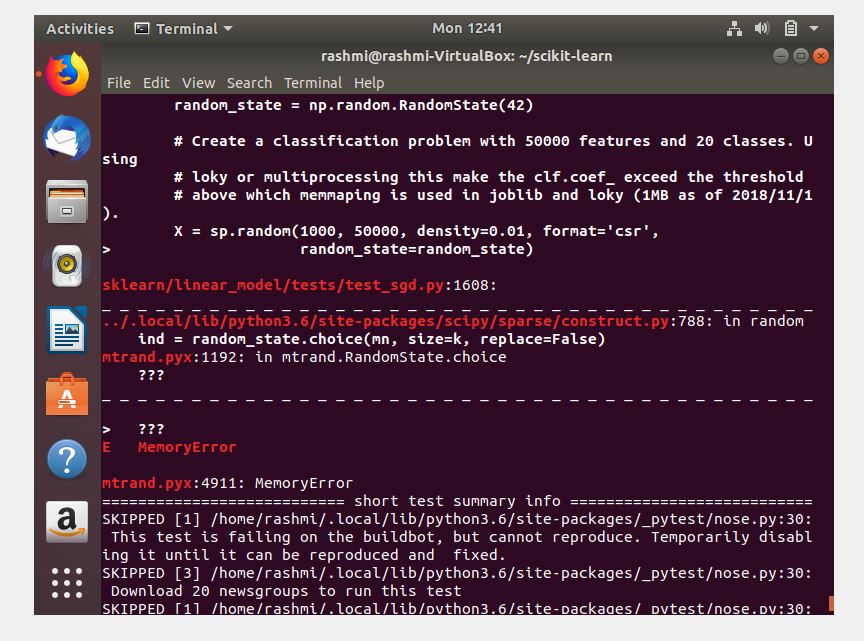
It uses cmake. Building Scikit-learn requires Cython 0.23 or above.

**Tests**

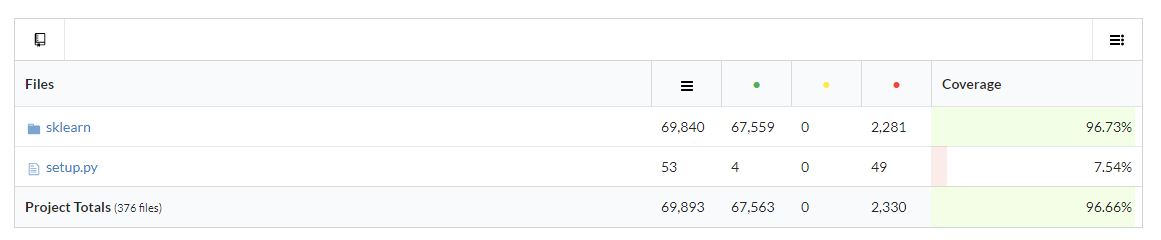
Running tests requires Pytest 3.3.0 or above. Some tests also require pandas. They have Travis CI platform integrated in their github. They test windows, linux and MAC computing platforms. Screenshot of test results is as follows:

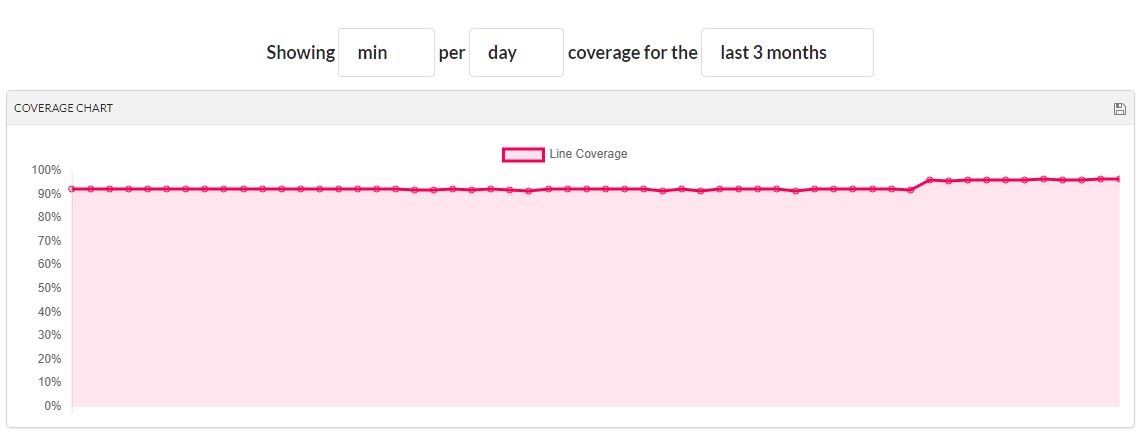


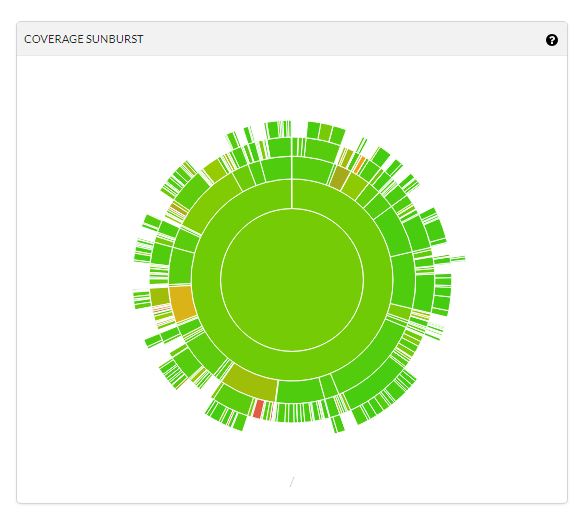
Investigating further into the failed test case, I found this:



Code coverage gives an analytical idea on how well the code or an application has been tested. Tests might return positive results across the board, but if they cover only 30-40% of the code, it’s really hard to gain confidence about the application. Hence, higher the value of code coverage, the better is the end product. Higher code coverage value indicates a more thorough testing of the application and less show-stopper bugs. Their github also has code coverage metrics. Screen shots of the code coverage are as follows:







Software Architecture

Scikit is basically an objected oriented implementation of various functional components. Six major functional components include classification, clustering, regression, dimensionality reduction, model selection and preprocessing. A high-level system diagram is as follows:



**Defects**

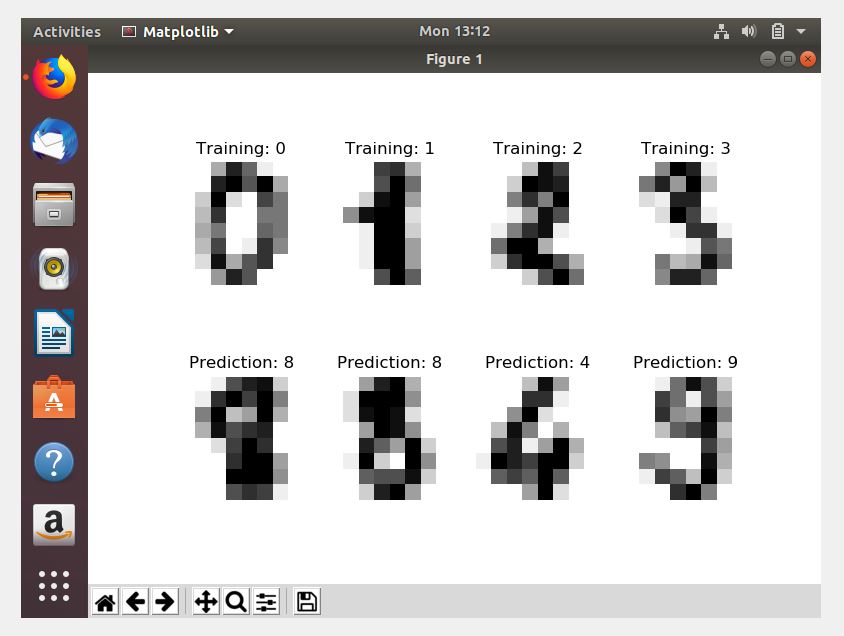
Scikit-learn recommends opening an issue in the issue tracker. This enables contributors to get some feedback from core developers. Advantage of this approach is that it avoids work duplication and avoids adding non-trivial features to the existing system. They also suggest going through the issue list or the pull request list before opening an issue.

Most of the issues in Scikit are related to feature update or adding a new functionality. I couldn’t spot an issue which requires an architecture change.

**Demonstration:**

After code installation and setup, I build the project and was able to run examples. Two sample demonstrations are as follow:

1. Classification example (to predict mnist numbers correctly):



1. Clustering example (K-means clustering with n=6):

