# Improving Scikit-learn usability and automation.

## Roles and responsibilities

Principle Investigator: Dr. Andreas Mueller
PI Mueller will oversee the data management, and ensure digital preservation of all produced data. Dr. Mueller will also review the continuous use of version control, as well as the documentation of all experimental procedures and the documentation of all software artifacts.
Research Engineer:
The Research Engineer will use the github version control platform to continuously and incrementaly keep track of software development. While some part of the created software will immediately be integrated in to the scikit-learn project, other parts of the software will exist in a seperate github repository. The Research Engineer will also document all software and experimental protocols within the github projects using the sphinx documentation generator. The Research Engineer will implement autmatic compilation of the documentation into a website that will be publicly available.
Large scale machine learning experiments will be performed on public datasets hosted on the OpenML platform.
The research engineer will retrieve the data via the OpenML python API, perform the experiments, and upload the experimental results to the OpenML platform.
The research engineer will also upload the collective results of all experiments periodically to the figshare platform.

## Types of data

The main output of this project will be python code, which will be archived and shared via github. Documentation of experimental procedures and the source code will be contained in the same github repository as the code.
The project will make use of the public machine learning datasets hosted on the OpenML platform. We will use hundereds of the datasets hosted there, which will be downloaded via the Python API of OpenML. The datasets are stored in ARFF files, with the metadata stored as JSON. The Python API represents these together as Python objects.
This data will be processed using a variety of algorithms from the scikit-learn machine learning library, as well as methods that we will implement as part of the project. The result of the processing will be predictions made by machine learning algorithms and their accuracy and other evaluation metrics. The metadata of these results consists of the original dataset, machine learning task, machine learning model that was used, and the parameters of the model.
These results and the meta-data will be captured and uploaded by the OpenML Python API to the OpenML platform, which will store it as JSON. We will also separately store the collected results across many datasets and algorithms on figshare as JSON file. This will provide an simple access mechanism as alternative to the OpenML interface for accessing the results. OpenML has programming interfaces in Java, R, Python, C# and other languages that will allow processing of the results.

**Policies for access and sharing and appropriate protection and privacy** 

Source code for all software will be made available as an ongoing process during development. All code and accompanying documentation will be licensed under a BSD license.
Results will be pushed to OpenML immediately as part of the evaluation, whenever possible. The results will be archived and published to figshare after inital analysis confirmed them to be correct.
All data will be made available under the CC-0 license.

**Data storage and preservation of access**

The code will be part of an open source project and thereby handed to the open source community. The initial archive will be github, but the community is expected to curate and archive the project beyond the livetime of github.
The results will be stored in the OpenML platform and on figshare, both of which have long-term preservation guarantees.

**Additional possible data management requirements**