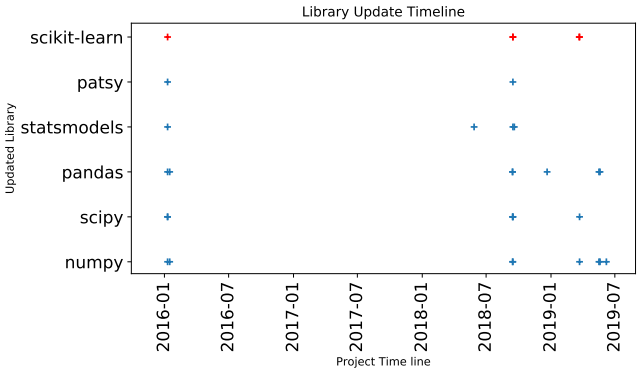
1 APPENDIX

1.1 Email to Group A: Developers who updated ML library versions

Hello Person-A,
We are researchers from Oregon State University, trying to understand paining points in machine learning library dependency update process. Typically, a library update might require to do software changes that would consume a lot of development effort. Our research focus on understanding these time consuming (or intractable) tasks and determining the feasibility of automating them.
Out of the **18,000** highest-rated projects that we analyzed, we observed *Project-A* to have the 63^{rd} highest usage density of **Scikit-learn**. We also observed that *Project-A* is amongst the **4%** projects that keep their machine learning libraries up to date.
[Figure below shows the entire library update history of *Project-A* based on *requirement.txt* analysis]



Scikit-learn [('>=', '0.17.1') to ('>=', '0.20.2')] 22, Mar 2019 Commit1

Q1. Was the above update difficult to perform? If so, what made it difficult? If not, what changes you had to perform as a result of the update?

Q2. How long it took to perform this update?

Q3. Was the update performed manually or using automated tools (or both)? What tasks would you like to get automated for *Project-A*?

Your response will be anonymized in our study and will be used only for research purposes.

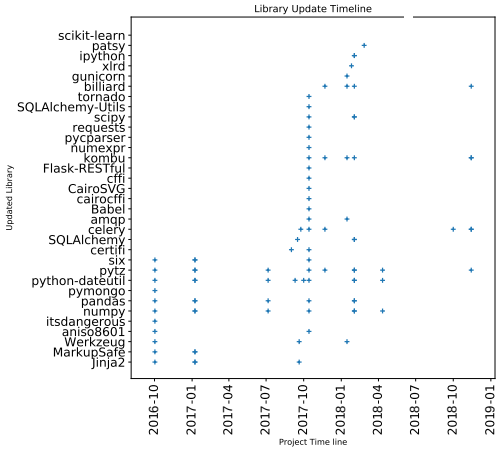
Thank you very much for your help and participation in our study. If you are interested in the results of this study, please let us know and we will contact you once the study is concluded.

If you have any other thoughts and insights into library update process, that you would like to share, please feel free to share it.
Regards,
XXX

1.2 Email to Group B: Developers who did not update ML library versions

Hello Person-B,
We are researchers from Oregon State University, trying to understand difficulties in updating machine learning library dependencies in open source projects

Out of the 18,000 top-rated projects that we analyzed, we observed *Project-B* to have the 150^{th} highest usage density of Scikit-learn. The figure below shows the entire library update history of *Project-B* and we noticed the machine learning libraries (Scikit-image) have not been updated while other libraries are updated. Probably, you may have tried to update Scikit-image and faced some challenges.



Below are the commit links for library upgrades done by you:
pytz [('==', '2.7.3') to ('==', '2.7.2')] 08, Nov 2018 Commit1
clery [('==', '0.23') to ('==', '0.19')] 08, Nov 2018 Commit2
kimpu [('==', '1.9.0') to ('===', '1.9.0')] 08, Nov 2018 Commit3
We would like to ask you the following questions regarding the above observation on Scikit-learn.

Q1. What prevents *Project-B* from updating Scikit-learn?

Q2. What would help *Project-B* to perform updates?(eg:tool support)

Q3. Would you desire to have tools to perform these update?

Your response will be anonymized in our study and will be used only for research purposes. Since the anonymized results will be presented to the library builders in the future, you answers will help them to solve the problems.

Thank you very much for your help and participation in our study. If you are interested in the results of this study, please let us know and we will share our conclusions once the study is completed.

If you have any other thoughts and insights into ML library update process, that you would like to share, please feel free to share it with us.
Regards,
XXX

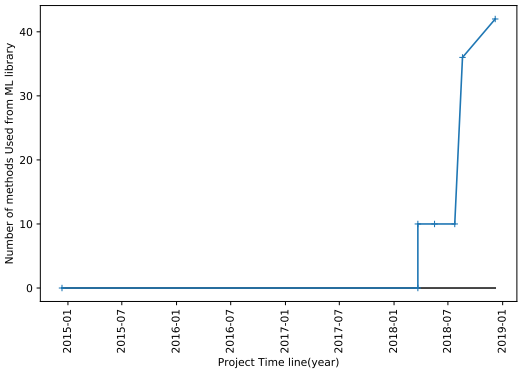
1.3 Email to Group C: Developers who retrofitted ML libraries

Hello Person-C,
We are a group of researchers from Oregon State University, investigating on why projects are adopting machine learning and we are trying to understand the adoption process. We are contacting

you because you have introduced a learning library *Tensorflow* for the project *Project-C* on *9-June-2018* for the first time in the project history.

After studying *18,000* top rated open source python repositories in GitHub, we identified 156 projects which introduced machine learning into the project significantly later in its lifecycle. *Project-C* is one such project, which introduced machine learning at the $1,094^{th}$ Commit1 after its creation. We have ranked your project as the 54^{th} project based on the library usage density and 53^{rd} project based on the time period between project creation date and ML retrofitting date among the 156 projects. After introducing the library, we noticed *Project-C* has continued to use Tensorflow capabilities as given in the plot below.

As a developer, you have introduced 31 ML library call sites to the project which makes you as one of the rock star ML developer in the project *Project-C*. Considering the strong facts mentioned above, we consider you as a resourceful person for helping us to understand the ML adoption process.



We have the following questions for you:

Q1. Do you accept that your project was a matured project before starting to use machine learning?

Q2. Why did you introduce the machine learning library for the first time in the project?

Q3. What challenges did you face?

Your response will be anonymized in our study and will be used only for research purposes.
Thank you very much for your time and participation in our study. Feel free to ask any questions you have about our research.
Regards,
XXX