**SIP Interview**

**February 20, 2020**

**Gael Varoquaux– scikit.learn**

GV: scikit.learn arose from needs in the field of brain imaging. To be sustainable, we needed a wide reach of applications to gather as large as possible of a community. Avoid the need for too much narrow expertise and narrow market.

For OSS, I like to think in terms of markets. Who are my customers? Need to justify that we are serving a lot of people to get funding. I think about consolidation – what are the basic technologies I need and what is already there? Focus on the limited number of problems I am trying to solve.

CD: How do you deal with a project that is by nature more specialized?

GV: We do have nilearn, which is healthy and viable in terms on community dynamics and funding because our market is much smaller. We get funding from French research agencies. But, it is too dependent on a single source of funding – risky. Also, if I can only convince a single agency I’m useful, am I really useful?

Need to bring obvious value to a community that is big enough.

CD: How did you reach this point? Model of initial development?

GV: Initial funding was internal. A few us of got together and decided what we wanted to do. Needed basic machine learning in Python. First surveyed the state of the art and determined current technologies were insufficient. This is an important step that is often overlooked. From the start, sustainability goal was to build community. Including licensing, branding, governance. Also focused on the technical quality of the project. Small number of high-quality features was the goal. Then invited others to join. People saw it was the right thing at the right time. Tiny steps to bigger steps.

CD: Doesn’t this contradict broad appeal, if it’s just a small number of features?

GV: Scikit.learn is somewhat sustainable due to the broad market. Hard to measure users because it does not phone home. Est. about 1M users.

Challenge with “not invented here” syndrome. In brain imaging, there are silos due to this attitude. Power game are disastrous. Funding gets derailed. Need people working together with common goals. Focus should be on solving challenging problems and competing in a healthy way. Identify bricks that you want to solve together.

CD: Who sustains the software now? Who pays for it?

GV: We have several academic actors, via grants or via “sheltering someone”, dedicating some percent of someone’s time. Private foundations are very helpful. Have a CZI grant as well as private partners who give money to the foundation. Multiple interests, including branding, risk mitigation, influencing roadmap.

CD: More about the foundation?

GV: Set up a bit more than a year ago, so pretty new. Setting up these things is a bit of work. First need to understand the market. Wanted a small amount of people who would give a large amount of money. Also didn’t want to grow the project too fast. Throw too much money at it, then you kill it. People get pipe dreams and become disconnected from the science. Only want as much money as we need. Slowly identified people who have common interests.

Meet users at technical conferences and here we found our partners. Once we created our foundation, we then found a few new partners.

CD: Who are your major funders?

GV: Two types: users and technology providers. Users are people who run data science gigs, who don’t want a huge internal tool. Consulting companies, insurers, banks.

Technology providers: Software and hardware. HW is tricky because they want the software to run best on their hardware and ideally lock-in, which is a no-no. We act as a trusted third party among them.

CD: Do commercial entities use scikit.learn? Have they adopted in a major way?

GV: I don’t think the world is ready to put scikit.learn on a scanner. See a lot of old mentalities in the lab and OSS is disruptive to this. The stakes are so big when they move to a scanner, would need a lot of money. This can happen in the R&D lab but need good vertical communication to go farther.

CD: Users?

GV: The finance industry uses this, but they aren’t really interested in helping other domains.

CD: Has the user community contributed back with Software?

GV: Absolutely. We have cases where the user becomes the developer. We purposely tried to keep it simple so people could contribute. We also have the “drive-by” contributors, e.g., fixing a little annoying detail that can be very useful.

JK: Development governance?

GV: We do have a core team. Best for it not be one person. Team reviews pull requests. Need two approvals to merge a pull requests. Worried because we take a lot time to do this. Core team is mostly funded. Try to tell them they should spend dedicated time reviewing pull requests. Have technical committee of “old timers” for important, big decisions. Also have an extension proposal process. It is bureaucratic but it is a tradeoff. It’s important to understand the rationale. Technical committee can vote if consensus fails, though this has been rare.

GV: Using BSD-3 clause license and the code has been forked. Have not used copyleft – this kicks in when you distribute software. Also, we don’t want to work against the community. Philosphopy: anything that is generic, needs to be fully open source. Where we will win, keep as little as possible in the private aspects.

GS: Agree with community working together to solve the problem and healthy competition. Do you worry about companies benefitting from the R&D and not contributing back.

GV: There are certainly plenty of “leeches”. Sometimes companies ask for specific features but this costs a great deal and they usually walk away. It works better when groups contribute the features. It’s crucial to identify the common features and to focus on these. We have an advisory technical committee of the foundation (technical reps for contributors) and this group jointly identifies the common needs. This is a very interesting exercise.

We don’t work with companies when it’s clear the management doesn’t understand the technical decisions. We chose the BSD license to limit the legal noise.

GS: What if a company request a specific feature and you have a specific contract?

GV: If it’s not seen as useful by many people, we may not do this. We never say no, we just put a very high price. If they still say yes, it is really interesting because it means it has more value than you thought it had (this has never happened). More usual is that companies have shared interest. I will never take money to work on non-OS projects. Because then it becomes a conflict of interest.

CD: How do you compete with market forces in this very hot area? E.g. developers can make much more money outside of academics.

GV: We can’t compete. People are more interested in what they are doing than on high salaries. Important to be a distributed team. I try to make this a good place to work, I try to be a no-BS manager. I protect them from useless things. Make sure there is a lot of technical expertise on the team and that they like to work together. Make sure their quality of life is good. When I hire, I try to understand their motivation.

CD: Long term sustainability of software == Long term sustainability of enthusiasm.

If people lose their enthusiasm, then you need to pay them more.

Tradeoff between features and technical vision. Our team needs to be with the users and understand their users. Best people like to go out of the lab, sit down with users, work with them for a few hours. We send our team to lots of conferences so that they know what to build.

Cell Profiler – Ann Carpenter. This is an interesting project. She has built a project dedicated to her problem. She has a nice understanding of how an ecosystem works. Helps her keep talented people on her team.