Data Language insights

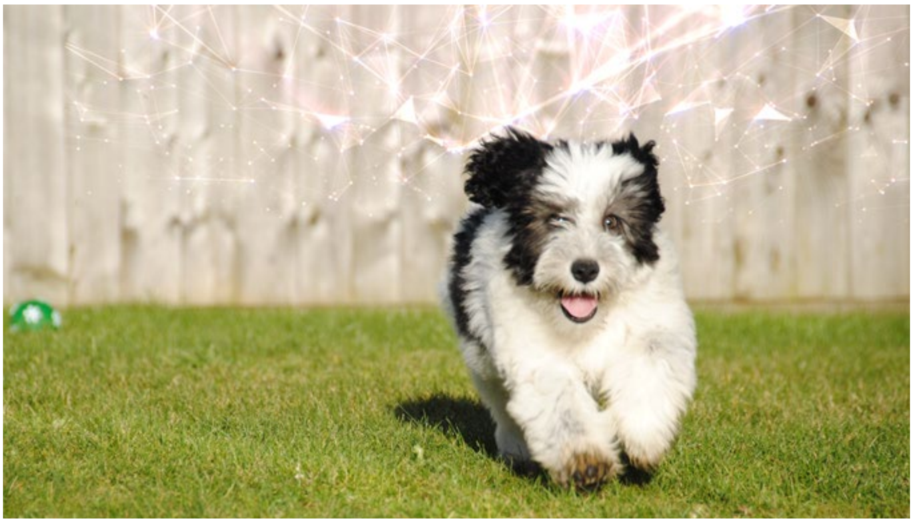
Why you really need to rethink data science in your company

Data science • Machine learning • Artificial intelligence • Software development

If data science is the new family puppy, then he really wants to play run around in the garden. If you bring him indoors then everyone can play. You just need to tame him a bit first.

Fabio Colasanti
Head of Data

Whether the context is academia or enterprise, research or commercial, the adoption of Software Development best practices, tools and methodologies is the key for efficient delivery of outcomes even though more traditional Data Scientists don’t associate this with their profession.



The Problem

There’s a problem with data science in most companies. A caustic, but not unfair definition would be that “Data Science is that thing you get when someone with a PhD plays with a Jupyter Notebook, and a technical architect refuses to let it anywhere near production because it doesn’t have a single unit test.”

This diagram from a notorious paper *Hidden Technical Debt in Machine Learning Systems* by D. Scully et al. illustrates the assumption that Data Science and Machine Learning belong into a hazy limbo somewhere between data analysis, analytics, and R&D.

In data science the following inefficient practices are often accepted as the norm:

- manual mining and manipulation of data
- one-off transformations virtually impossible to replicate
- casual scripting of code with total disregard of any development standard
- artefacts laying around
- code running here or there

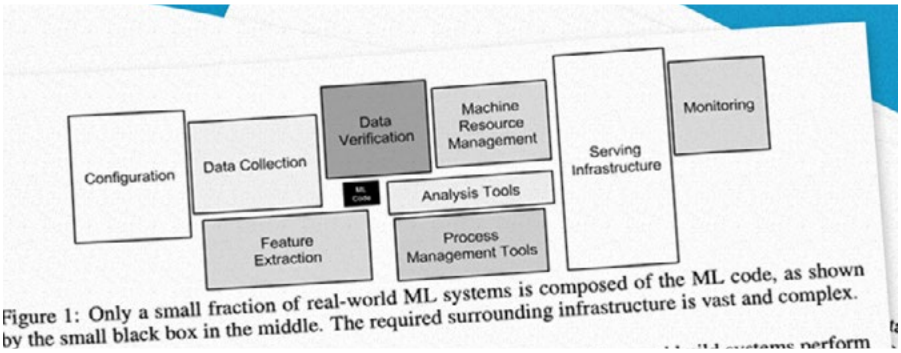


Figure 1: Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small black box in the middle. The required surrounding infrastructure is vast and complex.

Extract from Hidden Technical Debt in Machine Learning Systems *by D. Scully et al.*

> “Data Science is about building software”

Scully talks about the problems of “glue code” and “plumbing” but underplays the issue because, to put it bluntly, Data Science is about building software. And it should be. Data Science is about building software.

How did we get to this place?

Academics moved into software, experts in their field, but not used to working as part of a wider business. They rarely fully consider the implications of their coding decisions and have little experience working with peers from engineering even less so those from a business perspective whose priority is making a commercially viable product .

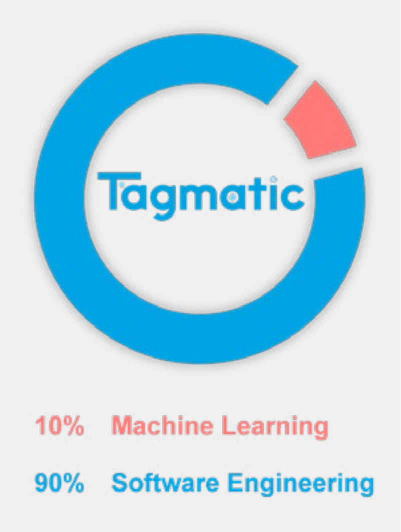
So what's the solution?

The adoption of established best practices in Software Development will boost productivity, shorten the feedback cycles, and produce more robust code that is easier to change, maintain and ship.

Data Language have worked with a number of clients at different levels of "enlightenment", and witnessed first hand how injecting software craftsmanship into Data Science can lead to profoundly transformational effects.

The Tagmatic experience

The Data Language team developed Tagmatic to generate high quality tags for content automatically using Machine Learning & AI.



Ratio of Software Engineering to Machine Learning in the development of Data Language's Tagmatic tool.

The powerful model behind the tool could never have been so successful without employing containerisation, high availability, high throughput, load balancing, scaling, automation, security, governance, monitoring and the rest of the software engineering.

Data Language recommendations for Data Science in business

1. **No Jupyter notebooks**
   These editors foster bad habits

2. **Excellent version control strategy**
   This help real collaboration aiding, for example, branching and merging.

3. **Use Test Driven Development and Continuous Integration**
   These are universally recognised to contribute the most to faster, safer, better code. This applies in Data Science too. To move faster with your code enables quicker feedback and shorter iterations. The faster that experimentation and proving ideas can happen, the greater the boost to productivity. Additionally the resulting code is better designed and is easier to transition into a production environment.

4. **Infrastructure as code**
   Write code that is portable and can be deployed wherever is faster to run it to take advantage of the ongoing democratisation of computing power. Typically it is preferable to deploy and execute code in the cloud, collect the outputs then dispose of the remote resources all in a fully-automated way. Using containerisation and learning it well is an excellent way to achieve this.

5. **Excellent data management**
   As with all science, experiments must be repeatable under the same conditions so the immutability of inputs must be preserved through transformations. Versioning and artefacts management tools play an invaluable role in the development of Data Science solutions.

6. **Excellent metadata management**
   This is at the core of many activities that are: routine (e.g. comparing performance of models); required for production (e.g. backout strategy for a failing data pipeline); or reach into ethics and accountability (e.g. quantitative justification of Machine Learning prediction). Retaining key information from a process permits useful analysis, interpretation and enhancement of outputs. The insight gleaned relates closely to the quality of the metadata recorded.

A daunting prospect

This seems like a big change, but remember life before DevOps? It's only relatively recently that network and infrastructure specialists became the DevOps that design and build infrastructure solutions through code. They are fully embedded in the fabric of development teams from the onset, sharing tools and methodologies. Before then there needed to be a heavily managed pipeline of stages before release. The change needed for integrating Data Science properly is nearly identical.

The key for enabling the evolution of Data Science from R&D practice to fully fledged enterprise function, lies in bringing data scientists and developers closer. Before long the distinction will blur, and specialist knowledge will thrive on a common substrate of working practices.

Remember **Data Science is about building software.**

Go and play with the puppy.
And get in touch if you need some help.