Data Science Tools, or what's in my e-backpack?

One of the infuriating (and at the same time, strangely cool) things about development data science is that you quite often find yourself in the middle of nowhere with a job to do, and no access to the Internet (although this doesn't happen as often as many Westerners think: there is real internet in most of the world's cities, honest!).

Which means you get to do your job with exactly what you remembered to pack on your laptop: tools, code, help files, datasets and academic papers. This is where we talk about the tools.

The Ds4B toolset

This is the tools list for the DS4B course (if you're following the course, don't panic: install notes are here).

Offline toolset:

- Already on the machine:
    - Terminal window
    - Calculator
    -
- Anaconda:
    - Jupyter notebooks
    - Python (version 3)
    - R (need to add this to Anaconda)
    -
- Rstudio
- OpenRefine
- D3 libraries
- Tabula
- Excel or LibreOffice (opensource equivalent)
- QGIS (Mac users: note the separate instructions on this!)
- GDAL toolset
    - Ogr2Ogr
    - GDALwarp
    - gdal_polygonize
    -
- Already in your bag:
    - Pen and paper (never underestimate this; maybe add post-its)
    -
-

Online toolset (ie. is useful, but not available when you're offline):

- Cometdocs
- Google spreadsheets
- Tableau Public
- CartoDb
-

Python and R both come with a bunch of useful libraries in Anaconda (here's the Python and R lists). The (Anaconda 4.0) pre-installed libraries used in DS4B are:

- Python: Basemap, BeautifulSoup, Dask, Matplotlib, NetworkX, Nltk, Numba, Numpy, Pandas, Requests, Scikit-image, Scikit-learn, SeaBorn, Shapely, Sqlite3
- R: ggplot2, frame, lm
-

The course also uses some Python libraries that don't come with the Anaconda standard install. These are:

- Csv, DateTime, Facepy, Fiona, Gdal, Gdalconst, Geopy, Googlemaps, Json, Ogr2ogr, Osgeo, Pyspark, Re, Twitter
-

How to get each of these is listed in the DS4B course instructions, but usually isn't more onerous than typing "pip install libraryname" in the terminal window. Fiona might give you some trouble ("cannot find the gdal.h library"): if this happens, there are notes in the install instructions.

Choosing the course toolset

For the course, we deliberately biased towards tools that were:

- Free (because they need to be accessible for everyone, not just people who can pay),
- Open source with good communities (because you can ask for help, go in and see how and why things work when you need to, and things get fixed a lot faster when you're not limited by a company's resources/ will).
- Accessible on most platforms (e.g. Windows, Mac and Linux of varying versions of operating system)
- Easy to install. Because nothing puts you off a tool quite as much as watching a build fail again and again, and having to learn all about the technologies its built on and their variants before you can do even the simplest thing with it .
- Stable. It's cruel to tell people without tech support to install a tool that regularly crashes on them. Even if you're a techie, it's still very annoying...

-

I do a lot of basic numbers work in Microsoft Excel (add things up, do sorts and filters of data, click on numerical columns to averages etc at the bottom of the screen) and the calculator. Some data scientists (although a dwindling number of them) only use Excel to process data, but in development data science we're often handling very unstructured or messy data (e.g. 'who knows how many columns this thing really has?' type data) and need to produce results that we can both repeat, and trace back step-by-step to the raw data (and sometimes we get lucky and have a dataset bigger than Excel's limits of 1,048,576 rows by 16,384 columns: NB Excel will silently fail and only give you the first rows/columns if you do this).

That means using a coding language (yes, yes, SPSS, SAS, Matlab, Pentaho etc, but a) those aren't free, and b) did I mention the really messy inputs?).

R and Python are coding languages that turn up a lot in data science (sql is used a lot too, for database data). R is a beautiful language for doing statistical things: it was built by statisticians, has packages that aren't in other languages yet, and has some lovely lovely visualisation libraries. But we needed to pick one language for the course to minimise the amount of code needed to illustrate each step in data science without causing cognitive dissonance in students' heads (the previous version of the course taught both Python and R, but was much more code-focussed), and chose Python.

Why Python? Basically three things:

- its ability to deal with really nastily messy data,
- its great libraries for things outside statistical analysis (e.g. natural language processing, web scraping, content management systems etc)
- not having to rewrite code when you start working with developers building applications and websites (although it's possible to call both Python and R from each other using e.g. the rpy2 package, adding extra languages makes the code that much harder to maintain).
- it's much easier to learn (and teach) than people think
-

There have been many debates on R vs Python for data science. "R vs Python" isn't really the right question to be asking; the question is "what tools will work for me", and the answer for many data scientists is "both R and Python, and sometimes neither": you use what works for you, and you use what's appropriate for the task that you have at hand.

The other tools included are less controversial (!), and more specialised.

- OpenRefine is a great tool for summarising and cleaning unruly row-column data without coding (and has provenance tracing built in)

- Tabula is the most popular open-source pdf processing tool (CometDocs works on messier documents, but is online-only)
- QGIS is a popular open-source GIS (maps etc) visualisation tool; CartoDb (now called Carto) is a beautiful online GIS visualisation tool
- The GDAL toolkit is great for command-line processing of GIS data
- Python includes visualisation libraries, but if you really want something special, D3 is a good offline tool (Tableau Public is good online, but as the name implies, your visualisations will be public).
- Pen and paper are useful for doing mockups and calculations whilst saving your machine's battery, and is flexible and portable to boot.
-

That should be enough to get most people started, and has already been field-tested by both myself and various ex-students (guys: I love it when you send me notes about what you're doing with data, from the field!).

What's in my own backpack?

It's only fair for me to open up my Mac and show you what I've got hiding on my own machine. That's my first Launchpad screen above. It's the usual suspects: Excel, R, OpenRefine, Tableau, Calculator, with a few other things:

- SQL tools: MySQL workbench, Postgres.
- Readers for some common proprietary tool formats: SPSS Smartreader, Pentaho Kettle ("Data Integration").
- More data tools: Weka, Gephi, Dato, Trifecta. Weka is the granddaddy of data mining tools, and is still worth having in your ebackpack. Gephi is great for visualising and playing around with graph data.
- Disk cleaners. OMG can I frag a disk on deployment, which is why I have Ccleaner and Disk Inventory X installed. Because sometimes you just need that extra 2Gb of space.
- Tools for keeping my code contained: Github desktop, VirtualBox.
- A Java development environment (IntelliJ) because, contrary to popular opinion, I don't just write code in Python.
-

The things you can't see here include:

- Mapping tools: OpenStreetMap, LeafletJS, MapBox
- Visualisation tools: Highcharts
- Machine learning tools: specialist tools, as and when I need them
- Ideation tools: freemind
- Text tools: Acrobat reader, sublime text
- Losing the minimum amount of stuff if my laptop dies in the humidity tools: Dropbox, Evernote, Google Drive, biggest portable drive I can find (2 of)
- Giving me textbooks to read on the road tools: iBooks, Safari Books
-

That's my ebackpack. I'd be interested to see what other people pack, and if there's anything useful that I've missed from the lists above.