Git

Don't push datasets

It's simply not designed to fit.

Don't push secrets

And rotate keys quickly if you do.

Don't push notebook outputs

So your commits don't look like trash.

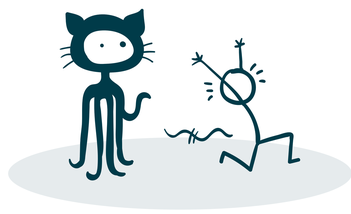
Don't use the --force

It'll take you to the dark side.

Do small commits with clear descriptions

Otherwise, your coworkers will get lost.

Don't be afraid of branching & pull requests



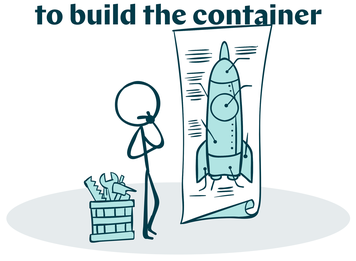
They aren't as scary as they sound.

Docker

It's like a space suit – for your code

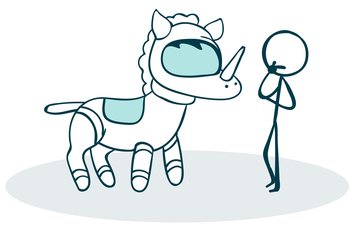
It'll keep it safe from outside factors.

Dockerfile tells you how to build the container



This is slightly different from requirements.

Be safe, avoid `latest`



It can cause some unexpected outcomes.

Don't bake your secrets

Everyone sees your cake and can eat it too.

Use venvs in development, containers in production

It's just way more comfortable that way.

Python Dependencies

Dependencies are like the sewers

Invisible when they work –
but when they don't you'll be knee deep.

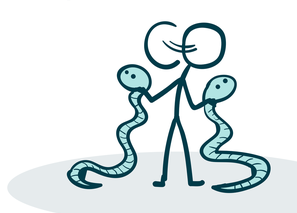
Virtual environments are a safe space

They keep your dependencies in check
and others safe from your mess.

Pinning versions reduces risk so use it

You don't want an updated package to
blow up your project – without asking.

Packages change a lot, Python not so much



It's not as necessary to pin a Python version
because it doesn't change on whim.

Command Line

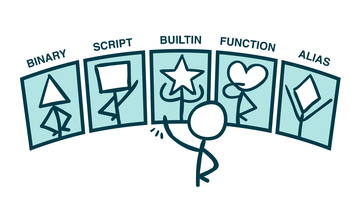
Don't be afraid of the command line

Command line is like "popping the hood"
of an operating system.

Recognize the prompt and the tilde

The command lines can look different but
these two characters should be there.

Combine commands for much more power



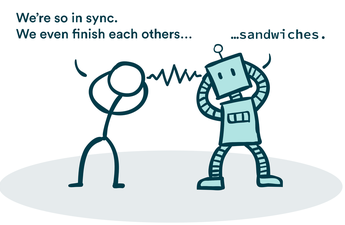
There are five different types of commands:
binary, script, builtin, function, and alias.

Manage data science projects like a boss

Makefiles make complex command line
operations easy for others to execute.

Programming Tools

Don't sleep on code completion

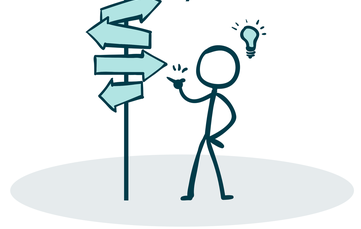


Writing code is becoming less about
remembering the syntax.

Use IDEs to refactor your code

Unlike notebooks, IDEs are much more
context-aware and thus safer.

Check how things actually work

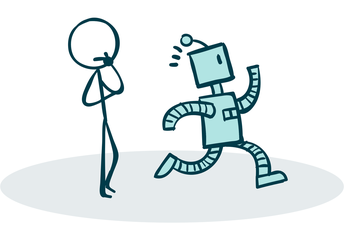


IDEs make it easy to dive into
the source code of 3rd party packages.

Use a debugger to unblock yourself

Visual debuggers are powerful and can
save you a lot of staring time.

Use a profiler to understand slowness



Optimizing performance is hard but
a profiler will at least tell you what is slow.

Use the right tools

More complex projects require more
capable tools. So don't ignore them.

Engineering Practices for Data Scientists

CHEATSHEET

This cheatsheet is companion material to the "Engineering Practices for Data Scientists" eBook, which contains helpful tips and tricks on common engineering topics such as Git, Docker and Python dependencies.

The full eBook is available for free:

https://valoh.ai/engineering-ebook