

# Sentiment Analysis

This is part 3 of the Python portion of a Software Carpentry workshop that is built around some light handling of Twitter data. There are four parts. At the end of this section participants will have played with a functioning sentiment analysis and visualization program. Skills covered include:

- pip and loading outside libraries.
- Reading and commenting code.
- More visualization work.

## pip

---

Before going further with this portion of the workshop we need to see how to use `pip`. So far we've *mostly* been using code that we can see and edit directly or with libraries that are pre-installed. Sooner or later you will need to make use of libraries that go beyond this basic functionality and you will need to know how to install them. For python this is done on the commandline. For packages that are part of the Anaconda distribution you can often use the `conda` tool but there are many more packages available than what you can get with this. `pip` is usually the package manager of choice and it is almost always installed with Python. Let's go to the commandline now and run:

```
$ pip

Usage:
  pip <command> [options]

Commands:
  install           Install packages.
  download          Download packages.
  uninstall         Uninstall packages.
  freeze            Output installed packages in requirements format.
  list              List installed packages.
  show              Show information about installed packages.
  check             Verify installed packages have compatible dependencies.
  search            Search PyPI for packages.
  ...
```

With no command `pip` is not sure what we want it to do and so it gives us the full list of commands that we could issue. Typically we are interested in doing/knowing or doing one of three things:

1. What is currently installed on the system.
2. What is available to be installed on the system.
3. Actually installing a library on the system.

What software is currently installed?

What happens when you search for 'textblob'?

Making an installation with `pip` can be done with the install command. To install `textblob` we run:

```
pip install textblob
```

[Note: Windows users can run pip commands at the command prompt.]

## Reading and commenting code

---

With `textblob` installed we can now open the third notebook.

This notebook comes with two fully functional sentiment analysis examples. The first simply provides the sentiment polarity for a sample tweet. The second takes what is done in the first example and applies it to a collection of tweets featuring the hashtag #trump that were pulled over approximately 8 hours overnight in late April 2016. Workshop participants should focus on doing the following:

1. Confirming that both pieces of code work. The second should fail until and unless they have unpacked `trumpStream.json.zip`. They can be challenged at how to look at this file given that it is so big (the fact that each line is its own encapsulated json element makes this pretty easy). Note that the second piece of code will not be done instantaneously.
2. Ask them to look at the comments currently in the files and decide on what types of comments are there.
3. Ask them to add their own comments to explain what is going on and to address any questions that they may have.