Practical Bioinformatics

Mark Voorhies

5/13/2019

Resources

Course website:

- http://histo.ucsf.edu/BMS270/

Resources on the course website:

- Syllabus
  - Papers and code (for downloading *before* class)
  - Slides and transcripts (available *after* class)
- On-line textbooks (Dive into Python, Numerical Recipes, ...)
- Programs for this course (VirtualBox, JavaTreeView, ...)

Homework

- E-mail Mark your python sessions (.ipynb files) after class
- E-mail Mark any homework code/results before tomorrow's class

Homework

- E-mail Mark your python sessions (.ipynb files) after class
- E-mail Mark any homework code/results before tomorrow's class
- It is fine to work together and to consult books, the web, etc. (but let me know if you do)
- It is fine to e-mail Mark questions
- Don't blindly copy-paste other people's code (you won't learn)

Homework

- E-mail Mark your python sessions (.ipynb files) after class
- E-mail Mark any homework code/results before tomorrow's class
- It is fine to work together and to consult books, the web, etc. (but let me know if you do)
- It is fine to e-mail Mark questions
- Don't blindly copy-paste other people's code (you won't learn)
- If you get stuck, try working things out on paper first.

Goals

At the end of this class, you should have the confidence to take on the day to day tasks of “bioinformatics”.

Goals

At the end of this class, you should have the confidence to take on the day to day tasks of “bioinformatics”.

- Analyzing data.

Goals

At the end of this class, you should have the confidence to take on the day to day tasks of "bioinformatics".

- Analyzing data.
- Writing standalone scripts.

Goals

At the end of this class, you should have the confidence to take on the day to day tasks of “bioinformatics”.

- Analyzing data.
- Writing standalone scripts.
- Shepherding data between analysis tools.

Goals

At the end of this class, you should have the confidence to take on the day to day tasks of “bioinformatics”.

- Analyzing data.
- Writing standalone scripts.
- Shepherding data between analysis tools.
- Aggregating data from multiple sources.

Goals

At the end of this class, you should have the confidence to take on the day to day tasks of “bioinformatics”.

- Analyzing data.
- Writing standalone scripts.
- Shepherding data between analysis tools.
- Aggregating data from multiple sources.
- Implementing new methods from the literature.

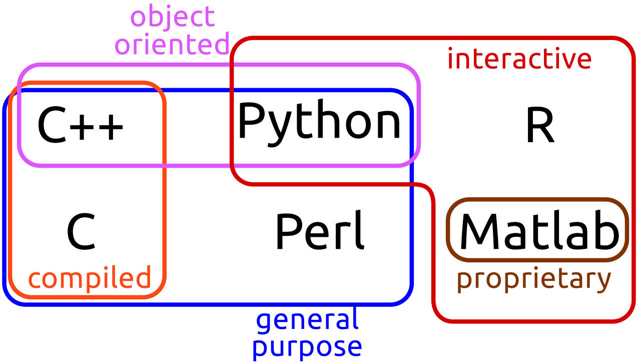
Goals

At the end of this class, you should have the confidence to take on the day to day tasks of “bioinformatics”.

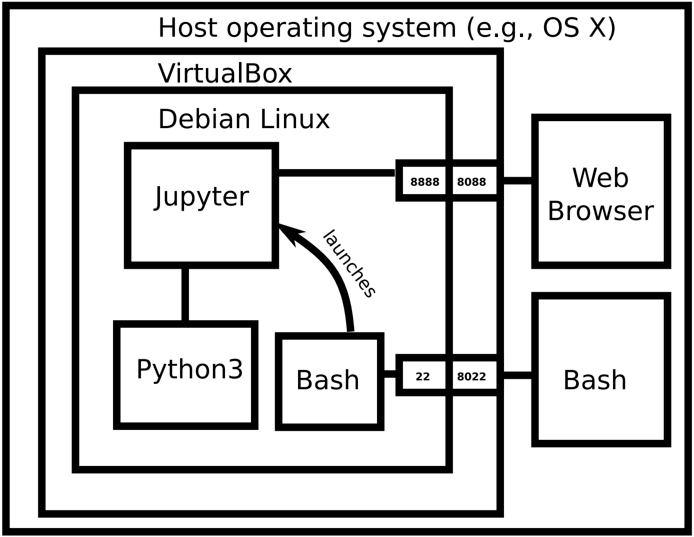
- Analyzing data.
- Writing standalone scripts.
- Shepherding data between analysis tools.
- Aggregating data from multiple sources.
- Implementing new methods from the literature.

This is also good preparation for communicating with computational collaborators.

Course tool: Python



Course platform: VirtualBox



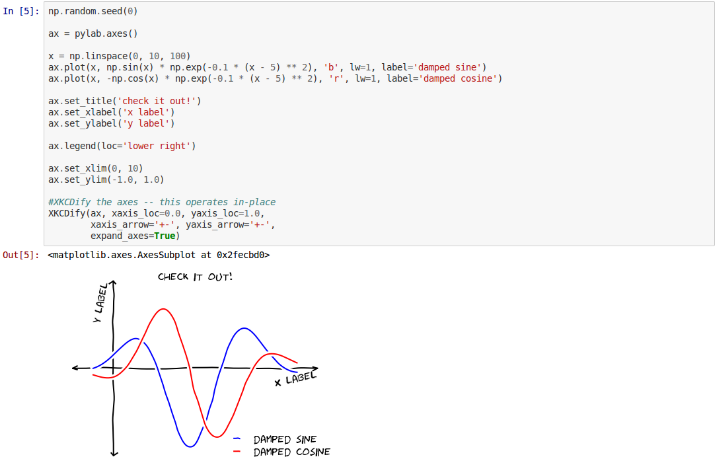
Host side bash commands

```
# Unlock your RSA private key
ssh-add ~/.ssh/VM_rsa
# Copy a file to the VM
scp -P 8022 myfile.txt explorer@localhost:
# Log into the VM
ssh -p 8022 explorer@localhost
# Get help on a command
man ssh
```

Guest side bash commands

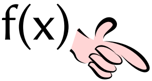
```
# Reboot the VM
su
shutdown -r now
# Shut down the VM
su
shutdown -hP now
# Start a screen session
screen
# Start Jupyter
jupyter notebook
```

Python shell: ipython (jupyter) notebook



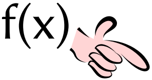
```
In [5]: np.random.seed(0)

        ax = pylab.axes()

        x = np.linspace(0, 10, 100)
        ax.plot(x, np.sin(x) * np.exp(-0.1 * (x - 5) ** 2), 'b', lw=1, label='damped sine')
        ax.plot(x, -np.cos(x) * np.exp(-0.1 * (x - 5) ** 2), 'r', lw=1, label='damped cosine')

        ax.set_title('check it out!')
        ax.set_xlabel('x label')
        ax.set_ylabel('y label')

        ax.legend(loc='lower right')

        ax.set_xlim(0, 10)
        ax.set_ylim(-1.0, 1.0)

        #XKCDify the axes -- this operates in-place
        XKCDify(ax, xaxis_loc=0.0, yaxis_loc=1.0,
                xaxis_arrow='+-', yaxis_arrow='+-',
                expand_axes=True)

Out[5]: <matplotlib.axes.AxesSubplot at 0x2fecbd0>
```

Anatomy of a Programming Language

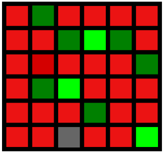


functions

Anatomy of a Programming Language

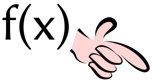


functions

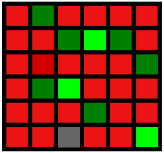


data structures

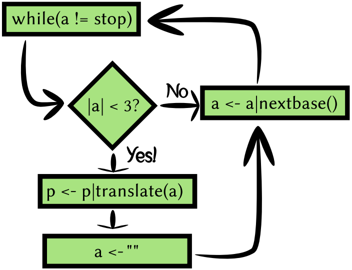
Anatomy of a Programming Language



functions



data structures



while(a != stop)

|a| < 3?

No

a <- a|nextbase()

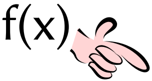
Yes!

p <- p|translate(a)

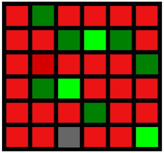
a <- ""

control statements

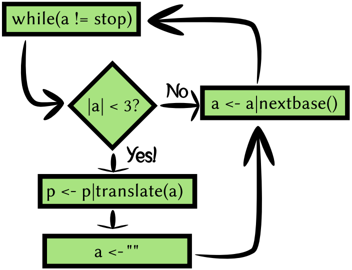
Anatomy of a Programming Language



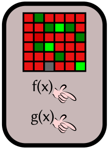
functions



data structures



control statements



objects

Binary files are like genomic DNA

hexdump -C computers.png

```
00000000  89 50 4e 47 0d 0a 1a 0a  00 00 00 0d 49 48 44 52  |.PNG........IHDR|
00000010  00 00 03 5f 00 00 02 cc  08 06 00 00 00 1b c3 08  |..._............|
00000020  30 00 00 00 04 73 42 49  54 08 08 08 08 7c 08 64  |0....sBIT....|.d|
00000030  88 00 00 00 09 70 48 59  73 00 00 2e 23 00 00 2e  |.....pHYs...#...|
00000040  23 01 78 a5 3f 76 00 00  00 19 74 45 58 74 53 6f  |#.x.?v....tEXtSo|
00000050  66 74 77 61 72 65 00 77  77 77 2e 69 6e 6b 73 63  |ftware.www.inksc|
00000060  61 70 65 2e 6f 72 67 9b  ee 3c 1a 00 00 20 00 49  |ape.org..<... .I|
00000070  44 41 54 78 9c ec 9d 79  9c 25 57 59 fe bf cf 39  |DATx...y.%WY...9|
00000080  75 6f 2f 33 93 cc 92 c9  1e 48 42 08 01 45 92 a0  |uo/3.....HB..E..|
00000090  04 c2 26 88 08 8a 80 0a  b2 28 18 14 54 14 45 04  |..&......(..T.E.|
000000a0  7f 02 a2 2c b2 aa 2c 0a  28 22 3b ca 26 20 8b b2  |...,..,.(";.& ..|
000000b0  08 c8 26 9b 61 4d 08 6b  08 d9 c8 be cd 4c 4f 77  |..&.aM.k.....LOw|
000000c0  df 7b eb 9c f7 f7 c7 7b  aa fb e6 ce bd 3d dd 93  |.{.....{.....=..|
000000d0  59 32 a4 9e fe d4 e7 76  55 9d 53 75 ea d4 a9 aa  |Y2.....vU.Su....|
000000e0  77 7d 8e cc 8c 16 2d 5a  b4 68 d1 a2 c5 8f 27 24  |w}....-Z.h....'$|
000000f0  75 81 00 f4 cc cc 24 45  a0 03 d4 66 56 8f 94 ed  |u.....$E...fV...|
00000100  00 b1 ac ee b2 7f 42 d9  15 cb ed 2f 48 da 0a 9c  |......B..../H...|
00000110  08 1c 0d 5c 0f 5c 05 9c  6f 66 fd 03 da b0 9b 29  |...\.\..of.....)|
00000120  24 4d 03 66 66 bd b2 5e  01 15 30 30 b3 b4 86 e3  |$M.ff..^..00....|
00000130  3c 1c 78 2a f0 25 33 7b  f2 1a db b0 01 f8 58 59  |<.x*.%3{......XY|
00000140  7d a0 99 5d bf 96 fa 2d  f6 0c 92 8e 01 9e 08 dc  |}..]...-........|
00000150  01 38 0a 10 f0 7b 66 f6  8d 03 d4 9e 67 01 0f 02  |.8...{f.....g...|
00000160  de 69 66 2f 3f 10 6d d8  9f 08 07 ba 01 2d 5a b4  |.if/?.m......-Z.|
00000170  68 d1 a2 45 8b 7d 8a af  01 0b c0 ed cb fa 6f 97  |h..E.}........o.|
```

```
fp = open("computers.png")
fp.read(50)
fp.close()
```

Text files are like ORFs

hexdump -C 3_4_2010.txt

```
00000000  4d 65 65 74 20 77 2f 20  4a 6f 65 20 72 65 3a 20  |Meet w/ Joe re: |
00000010  77 69 72 65 6c 65 73 73  20 74 68 65 72 6d 6f 73  |wireless thermos|
00000020  74 61 74 73 0a 20 20 20  2d 2d 3e 20 64 6f 6e 65  |tats.   --> done|
00000030  0a 20 20 20 20 20 20 42  75 79 20 74 68 65 72 6d  |.      Buy therm|
00000040  6f 73 74 61 74 73 20 66  72 6f 6d 20 68 74 74 70  |ostats from http|
00000050  3a 2f 2f 77 77 77 2e 6f  6d 65 67 61 2e 63 6f 6d  |://www.omega.com|
00000060  0a 20 20 20 20 20 20 20  20 20 20 53 74 61 72 74  |.          Start|
00000070  20 77 69 74 68 3a 0a 20  20 20 20 20 20 20 20 20  | with:.         |
00000080  20 20 20 20 52 6f 75 74  65 72 20 55 57 54 43 52  |    Router UWTCR|
00000090  45 43 33 20 28 61 62 6f  75 74 20 24 31 32 30 29  |EC3 (about $120)|
000000a0  0a 20 20 20 20 20 20 20  20 20 20 20 20 20 20 20  |.               |
000000b0  20 43 61 6e 20 72 65 63  65 69 76 65 20 66 72 6f  | Can receive fro|
000000c0  6d 20 31 32 20 74 72 61  6e 73 6d 69 74 74 65 72  |m 12 transmitter|
000000d0  73 0a 20 20 20 20 20 20  20 20 20 20 20 20 20 20  |s.              |
000000e0  20 20 43 61 6e 20 70 75  73 68 20 63 6f 6e 66 69  |  Can push confi|
000000f0  67 75 72 61 74 69 6f 6e  20 74 6f 20 74 72 61 6e  |guration to tran|
00000100  73 6d 69 74 74 65 72 73  0a 20 20 20 20 20 20 20  |smitters.       |
00000110  20 20 20 20 20 20 20 20  20 43 6f 6d 6d 75 6e 69  |         Communi|
00000120  63 61 74 65 20 76 69 61  20 65 74 68 65 72 6e 65  |cate via etherne|
00000130  74 20 70 6f 72 74 20 61  6e 64 20 65 6d 62 65 64  |t port and embed|
00000140  64 65 64 20 77 65 62 20  73 65 72 76 65 72 0a 20  |ded web server. |
00000150  20 20 20 20 20 20 20 20  20 20 20 20 20 20 20 41  |               A|
00000160  73 73 69 67 6e 20 73 74  61 74 69 63 20 49 50 20  |ssign static IP |
00000170  61 64 64 72 65 73 73 20  61 6e 64 20 63 6f 6e 6e  |address and conn|
```

OS X *sometimes* uses CR newlines

hexdump -C macfile.txt

```
00000000  4d 65 65 74 20 77 2f 20  4a 6f 65 20 72 65 3a 20  |Meet w/ Joe re: |
00000010  77 69 72 65 6c 65 73 73  20 74 68 65 72 6d 6f 73  |wireless thermos|
00000020  74 61 74 73 0d 20 20 20  2d 2d 3e 20 64 6f 6e 65  |tats.   --> done|
00000030  0d 20 20 20 20 20 20 42  75 79 20 74 68 65 72 6d  |.      Buy therm|
00000040  6f 73 74 61 74 73 20 66  72 6f 6d 20 68 74 74 70  |ostats from http|
00000050  3a 2f 2f 77 77 77 2e 6f  6d 65 67 61 2e 63 6f 6d  |://www.omega.com|
00000060  0d 20 20 20 20 20 20 20  20 20 20 53 74 61 72 74  |.          Start|
00000070  20 77 69 74 68 3a 0d 20  20 20 20 20 20 20 20 20  | with:.         |
00000080  20 20 20 20 52 6f 75 74  65 72 20 55 57 54 43 52  |    Router UWTCR|
00000090  45 43 33 20 28 61 62 6f  75 74 20 24 31 32 30 29  |EC3 (about $120)|
000000a0  0d 20 20 20 20 20 20 20  20 20 20 20 20 20 20 20  |.               |
000000b0  20 43 61 6e 20 72 65 63  65 69 76 65 20 66 72 6f  | Can receive fro|
000000c0  6d 20 31 32 20 74 72 61  6e 73 6d 69 74 74 65 72  |m 12 transmitter|
000000d0  73 0d 20 20 20 20 20 20  20 20 20 20 20 20 20 20  |s.              |
000000e0  20 20 43 61 6e 20 70 75  73 68 20 63 6f 6e 66 69  |  Can push confi|
000000f0  67 75 72 61 74 69 6f 6e  20 74 6f 20 74 72 61 6e  |guration to tran|
00000100  73 6d 69 74 74 65 72 73  0d 20 20 20 20 20 20 20  |smitters.       |
00000110  20 20 20 20 20 20 20 20  20 43 6f 6d 6d 75 6e 69  |         Communi|
00000120  63 61 74 65 20 76 69 61  20 65 74 68 65 72 6e 65  |cate via etherne|
00000130  74 20 70 6f 72 74 20 61  6e 64 20 65 6d 62 65 64  |t port and embed|
00000140  64 65 64 20 77 65 62 20  73 65 72 76 65 72 0d 20  |ded web server. |
00000150  20 20 20 20 20 20 20 20  20 20 20 20 20 20 20 41  |               A|
00000160  73 73 69 67 6e 20 73 74  61 74 69 63 20 49 50 20  |ssign static IP |
00000170  61 64 64 72 65 73 73 20  61 6e 64 20 63 6f 6e 6e  |address and conn|
```

tr '\r' '\n' < macfile.txt > unixfile.txt

Windows uses CRLF newlines

hexdump -C dosfile.txt

```
00000000  4d 65 65 74 20 77 2f 20  4a 6f 65 20 72 65 3a 20  |Meet w/ Joe re: |
00000010  77 69 72 65 6c 65 73 73  20 74 68 65 72 6d 6f 73  |wireless thermos|
00000020  74 61 74 73 0d 0a 20 20  20 2d 2d 3e 20 64 6f 6e  |tats..   --> don|
00000030  65 0d 0a 20 20 20 20 20  20 42 75 79 20 74 68 65  |e..      Buy the|
00000040  72 6d 6f 73 74 61 74 73  20 66 72 6f 6d 20 68 74  |rmostats from ht|
00000050  74 70 3a 2f 2f 77 77 77  2e 6f 6d 65 67 61 2e 63  |tp://www.omega.c|
00000060  6f 6d 0d 0a 20 20 20 20  20 20 20 20 20 20 53 74  |om..          St|
00000070  61 72 74 20 77 69 74 68  3a 0d 0a 20 20 20 20 20  |art with:..     |
00000080  20 20 20 20 20 20 20 20  52 6f 75 74 65 72 20 55  |        Router U|
00000090  57 54 43 52 45 43 33 20  28 61 62 6f 75 74 20 24  |WTCREC3 (about $|
000000a0  31 32 30 29 0d 0a 20 20  20 20 20 20 20 20 20 20  |120)..          |
000000b0  20 20 20 20 20 20 43 61  6e 20 72 65 63 65 69 76  |      Can receiv|
000000c0  65 20 66 72 6f 6d 20 31  32 20 74 72 61 6e 73 6d  |e from 12 transm|
000000d0  69 74 74 65 72 73 0d 0a  20 20 20 20 20 20 20 20  |itters..        |
000000e0  20 20 20 20 20 20 20 20  43 61 6e 20 70 75 73 68  |        Can push|
000000f0  20 63 6f 6e 66 69 67 75  72 61 74 69 6f 6e 20 74  | configuration t|
00000100  6f 20 74 72 61 6e 73 6d  69 74 74 65 72 73 0d 0a  |o transmitters..|
00000110  20 20 20 20 20 20 20 20  20 20 20 20 20 20 20 20  |                |
00000120  43 6f 6d 6d 75 6e 69 63  61 74 65 20 76 69 61 20  |Communicate via |
00000130  65 74 68 65 72 6e 65 74  20 70 6f 72 74 20 61 6e  |ethernet port an|
00000140  64 20 65 6d 62 65 64 64  65 64 20 77 65 62 20 73  |d embedded web s|
00000150  65 72 76 65 72 0d 0a 20  20 20 20 20 20 20 20 20  |erver..         |
00000160  20 20 20 20 20 20 20 41  73 73 69 67 6e 20 73 74  |       Assign st|
00000170  61 74 69 63 20 49 50 20  61 64 64 72 65 73 73 20  |atic IP address |
```

Talking to Python: Nouns

```
# This is a comment
# This is an int (integer)
42
# This is a float (rational number)
4.2
# These are all strings (sequences of characters)
'ATGC'

"Mendel's Laws"

""">CAA36839.1 Calmodulin
MADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTEAEL
QDMINEVDADDLPGNGTIDFPEFLTMMARKMKDTDSEEEIREAFRVFDK
DGNGYISAAELRHVMTNLGEKLTDEEVDEMIREADIDGDGQVNYEEFVQ
MMTAK"""
```

Python as a Calculator

```
# Addition
1+1
# Subtraction
2-3
# Multiplication
3*5
# Division
5/3
# Exponentiation
2**3
# Order of operations
2*3-(3+4)**2
```

Remembering objects

```
# Use a single = for assignment:
TLC = "GATACA"
YFG = "CTATGT"
MFG = "CTATGT"

# A name can occur on both sides of an assignment:
codon_position = 1857
codon_position = codon_position + 3

# Short-hand for common updates:
codon += 3
weight -= 10
expression *= 2
CFU /= 10.0
```

Displaying values with print

```
# Use print to show the value of an object
message = "Hello, world"
print(message)
# Or several objects:
print(1,2,3,4)
# Older versions of Python use a
# different print syntax
print "Hello, world"
```

Collections of objects

```
# A list is a mutable sequence of objects
mylist = [1, 3.1415926535, "GATACA", 4, 5]
# Indexing
mylist[0] == 1
mylist[-1] == 5
# Assigning by index
mylist[0] = "ATG"
# Slicing
mylist[1:3] == [3.1415926535, "GATACA"]
mylist[:2] == [1, 3.1415926535]
mylist[3:] == [4,5]
# Assigning a second name to a list
also_mylist = mylist
# Assigning to a copy of a list
my_other_list = mylist[:]
```

Repeating yourself: iteration

```
# A for loop iterates through a list one element
# at a time:
for i in [1,2,3,4,5]:
    print(i, i**2)

# A while loop iterates for as long as a condition
# is true:
population = 1
while(population < 1e5):
    print(population)
    population *= 2
```

Verb that noun!

return_value = function(parameter, ...)
“Python, do *function* to *parameter*”

```
# Built-in functions
#  Generate a list from 0 to n-1
a = range(5)
#  Sum over an iterable object
sum(a)
#  Find the length of an object
len(a)
```

Verb that noun!

return_value = function(parameter, ...)
"Python, do *function* to *parameter*"

```
# Importing functions from modules
import numpy
numpy.sqrt(9)

import matplotlib.pyplot as plt
fig = plt.figure()
plt.plot([1,2,3,4,5],
         [0,1,0,1,0])

from IPython.core.display import display
display(fig)
```

New verbs

```python
def function(parameter1, parameter2):
    """Do this!"""
    # Code to do this
    return return_value
```

Summary

- Python is a general purpose programming language.

Summary

- Python is a general purpose programming language.
- We can extend Python's built-in functions by defining our own functions (or by importing third party modules).

Summary

- Python is a general purpose programming language.
- We can extend Python's built-in functions by defining our own functions (or by importing third party modules).
- We can define complex behaviors through control statements like "for", "while", and "if"

Summary

- Python is a general purpose programming language.
- We can extend Python's built-in functions by defining our own functions (or by importing third party modules).
- We can define complex behaviors through control statements like "for", "while", and "if"
- We can use an interactive Python session to experiment with new ideas and to explore data.

Summary

- Python is a general purpose programming language.
- We can extend Python's built-in functions by defining our own functions (or by importing third party modules).
- We can define complex behaviors through control statements like "for", "while", and "if"
- We can use an interactive Python session to experiment with new ideas and to explore data.
- Saving interactive sessions is a good way to document our computer "experiments".

Summary

- Python is a general purpose programming language.
- We can extend Python's built-in functions by defining our own functions (or by importing third party modules).
- We can define complex behaviors through control statements like "for", "while", and "if"
- We can use an interactive Python session to experiment with new ideas and to explore data.
- Saving interactive sessions is a good way to document our computer "experiments".
- Likewise, we can use modules and scripts to document our computer "protocols".

Summary

- Python is a general purpose programming language.
- We can extend Python's built-in functions by defining our own functions (or by importing third party modules).
- We can define complex behaviors through control statements like "for", "while", and "if"
- We can use an interactive Python session to experiment with new ideas and to explore data.
- Saving interactive sessions is a good way to document our computer "experiments".
- Likewise, we can use modules and scripts to document our computer "protocols".
- Most of these statements are applicable to any programming language (Perl, R, Bash, Java, C/C++, FORTRAN, ...)

Homework: Exploring Files

1. Try reading the first few bytes of different files on your computer. Can you distinguish binary files from text files?
2. Create a simple data table in your favorite spreadsheet program and save it in a text format (*e.g.*, save as CSV or tab-delimited text from Excel[1]). Practice reading the data from Python.

[1] Note for Mac users: Excel will offer you Macintosh and DOS/Windows text formats. Choose *DOS/Windows*