Machine Learning and Data Analysis with Python

Derek Harter

TEXAS A&M UNIVERSITY COMMERCE

2013-08-18

Outline

Additional Resources

The following are additional resources, all free and available online, that you should use to learn Python.

- Think Python: How to think like a computer scientist http://www.greenteapress.com/thinkpython A free but actually professionally done and published textbook.
- Google Developers Python Class https://developers.google.com/edu/python A short course from Google, but has a good set of videos to cover the basics.
- Software Carpentry Python Lectures http://software-carpentry.org/v4/python/index.html Well done video lectures part of a larger course on scientific software development.

Declaring Variables

- Python is a high-level interpreted language.
- Python does not force you to declare variable types.
- Type is inferred from assigned value.
- Python manages memory for you, will garbage collect unreferenced data.

Variable Declaration

```
x = 1
y = x + 3
print x, y
print type(x)
```

```
1 4
<type 'int'>
```

Operations on Variables

- Python includes all of the arithmetic and boolean operations with same syntax as C, Java, etc.
- Arithmetic operators use standard order of precedence: () ** * / % + -
- Boolean operators: == != < > <= >=

Operators Example

```
x = (3 + 5) * 2 ** 3
print x
print x <= 5
```

```
64
False
```

Functions

- A function is a named sequence of statements that performs a computation.
- Python uses def to define a new function.
- All Python functions return results, if you don't specify result using return, then None is returned as function value.

Function Example

```
def sum_ceiling(x, y, z, ceiling):
    """Return the sum of x+y+z if it is less than
    maximum ceiling.  Otherwise return the ceiling"""
    s = x + y + z
    if s < ceiling:
        return s
    else:
        return ceiling

print sum_ceiling(3, 8, 11, 20)
print sum_ceiling(1, 2, 3, 99)
```

```
20
6
```

Built In Data Structures: Lists

- Lists are sequences of values.
- The list values do not have to be of the same type (unlike a C or Java array).
- Lists are indexed by an integer value, starting at 0.
- Lists can be changed, values added or removed, etc.

List Example

```
states = ['Alaska', 'Alabama', 'Texas', 'Mississippi']
print states[0]  # first item in list
print states[1:3] # items 1 up to but not including 3 of list
print states[-1] # last item in list
states[2] = 'California'
print states
```

```
Alaska
['Alabama', 'Texas']
Mississippi
['Alaska', 'Alabama', 'California', 'Mississippi']
```

Built In Data Structures: Dictionaries

- Dictionaries map an arbitrary key to a value (key->value pair).
- Dictionaries are mutable, values can be changed, added or removed.

Dictionary Example

```
phone_number = {'John': '818-922-2381',
                'Susan': '414-938-1923',
                'Ray': 9034541238}
print phone_number['Ray']
phone_number['Alice'] = 8184531923
print phone_number


 9034541238
 {'John': '818-922-2381', 'Ray': 9034541238, 'Alice': 8184531923, '
```

Built In Data Structures: Tuples

- Tuples are immutable lists, they can't be changed.
- We mention because you will run across them early, for example to return multiple values from a function, Python programmers often return a tuple of values.

Tuples Example

```
def find_min_max(l):
    """Return the minumum and maximum values in the list l"""
    minimum = min(l)
    maximum = max(l)
    return (minimum,maximum)

l,h = find_min_max([9, 8, 2, 11, 42, 10])
print "Minimum was: ", l
print "Maximum was: ", h
```

```
Minimum was:  2
Maximum was:  42
```