Day 1

December 2, 2021

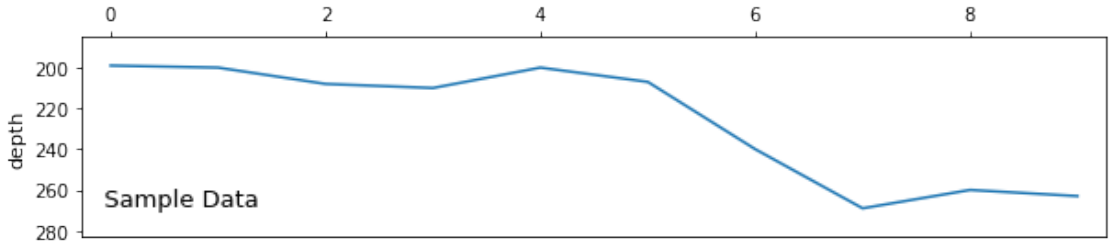
1 Advent of Code 2021 - Day 1

1.1 Setup

Parse and style.

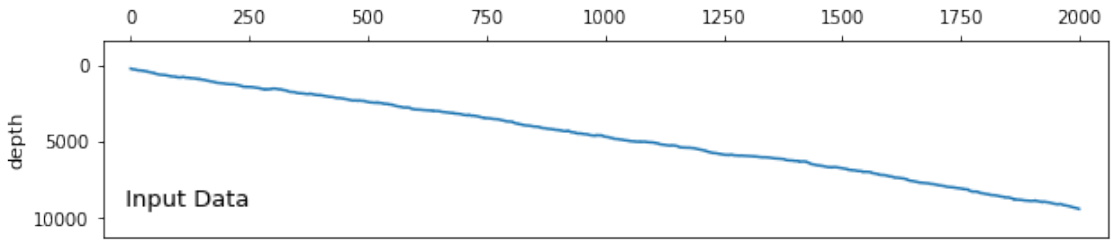
[1]:
```
from libaoc.styles import *
plotStyle()

def plotSetup():
    plotInvertY(.1)
    plt.ylabel("depth")

_inputText = open('day1.input.txt').read()
_inputData = [int(x) for x in re.findall('\d+', _inputText)]

_sampleData = [
    199,
    200,
    208,
    210,
    200,
    207,
    240,
    269,
    260,
    263]

plotSetup(); plt.title('   Sample Data'); plt.plot(_sampleData)
```

[1]:
```
[<matplotlib.lines.Line2D at 0x2933f28bee0>]
```



[2]:
```
plotSetup(); plt.title('   Input Data'); plt.plot(_inputData)
```

[2]:
```
[<matplotlib.lines.Line2D at 0x2934139b5b0>]
```



1.2 Solver

Part 1 and 2 can use the same solver:

- Walk list of ints
- Sum a moving window of given size
- Count instances where a sum is greater than the previous sum

[3]:
```
def solve(depths, window):
    last, count = 0, 0
    for depth in range(len(depths) - window):
        s = sum(depths[depth:depth+window])
        if s > last:
            count += 1
        last = s
    return count
```

1.3 Part 1

Window size = 1.

[4]:
```
def solve1(depths):
    return solve(depths, 1)
```

```
assert solve1(_sampleData) == 7

assert (s1 := solve1(_inputData)) == 1681
print(f"result = {s1}")
```

```
result = 1681
```

1.4 Part 2

Window size = 3.

[5]:
```
def solve2(depths):
    return solve(depths, 3)

assert solve2(_sampleData) == 5

assert (s2 := solve2(_inputData)) == 1704
print(f"result = {s2}")
```

```
result = 1704
```