Problem Set 3

Due by 3/14, 11:59pm

(0) In univariate time-series modeling, one class of models is known as autoregressive (AR) models. For example, an AR(2) process can be written as

$$y_t = \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + u_t$$

for $t = 1, 2, \ldots, T$. The subscript t denotes time periods and T is the last period. Now, suppose $\alpha_1 = 0.8$, $\alpha_2 = 0.15$ and u_t's are independent and identically distributed standard normal random variables. Consider the code snippet below that simulates the AR(2) for 200 periods and keeps the last 100 periods.

```
import numpy as np
np.random.seed(37)
import matplotlib.pyplot as plt
plt.style.use('ggplot')

T = 200
alpha1 = 0.8
alpha2 = 0.15
u = np.random.randn(T)
y = np.zeros(T)

for t in range(1, T):
    y[t] = alpha1*y[t-1] + alpha2*y[t-2] + u[t]

y = y[100:]  # keep the last 100 periods
```

Why do you think the for loop started from index 1? Write a `while` loop which generates 100 observations from the AR(2) process defined above, and then generates a line plot of the simulated AR(2) process as in Figure 1.

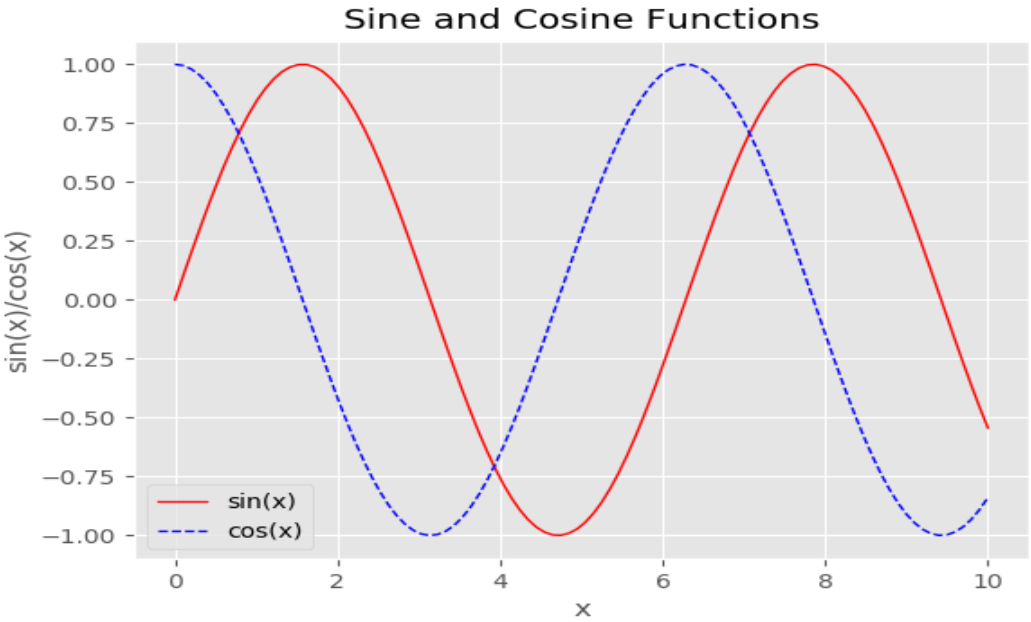
Figure 1. AR(2) example

(1) Consider the following code snippet.

```
x = np.arange(15)
odd = [] # empty list for odd numbers
even = [] # empty list for even numbers
```

Write a control structure that adds odd numbers in `x` to the empty list `odd`, and even numbers in `x` to the empty list `even`.

FIGURE 2. *sin* vs *cos*



(2) Consider the following code snippet.

```
from string import ascii_lowercase
letters = set(ascii_lowercase)
vowel = set()
consonant = set()
```

The object `letters` is a set containing lower case letters. Write a control structure that adds vowels to the set `vowel`, and consonants to the set `consonant`. Note that the vowels are {a,e,i,o,u}, and the rest of the letters are consonants.

(3) Take a look at Figure 2 above. Complete the code snippet below to generate Figure 2.

```
x = np.linspace(0, 10, 100)
sinx = np.sin(x)
cosx = np.cos(x)
```

(4) Modify your code snippet for question 3 to generate Figure 3 below. Note that the markers in the plot have following properties: `markeredgecolor = "black"`, `markersize = 5`.

(5) The web page `http://pypl.github.io/PYPL.html` provides an index showing how often programming languages are searched on Google. Consider the first 15 programming languages in the index. Write a Python program that displays a bar chart of the popularity of these 15 languages (just copy and paste names and associated values to your script file). Your bar chart should display the share of each programming language. Your code snippet should generate (approximately) Figure 4. Note that the plot command `plt.barh` is used to generate horizontal bar charts.

(6) Consider again the index mentioned in question 5. Write a Python program that instead displays a pie chart of the popularity of these programming languages. Your code snippet should generate (approximately) Figure 5.

FIGURE 3. *sin* vs *cos*

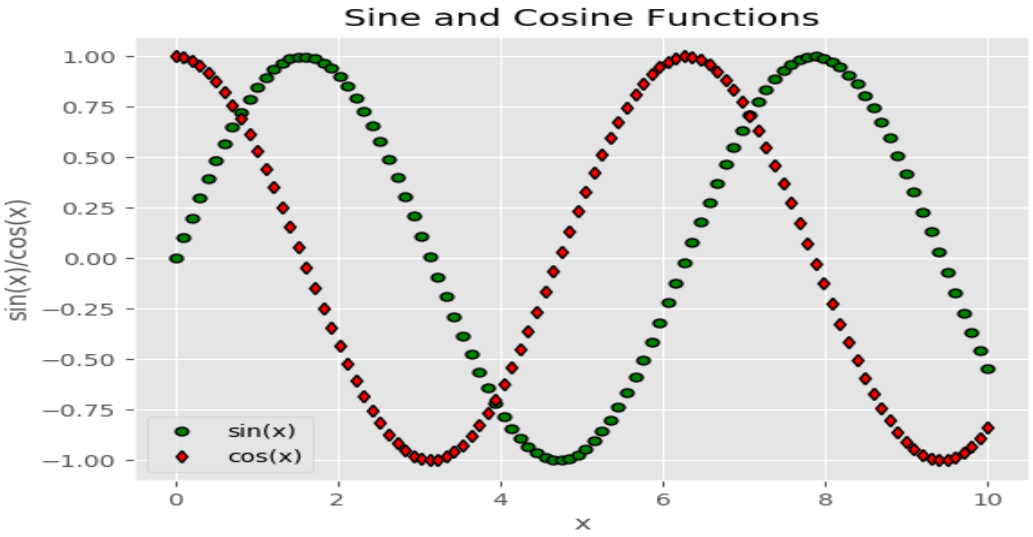


FIGURE 4. Popular programming languages

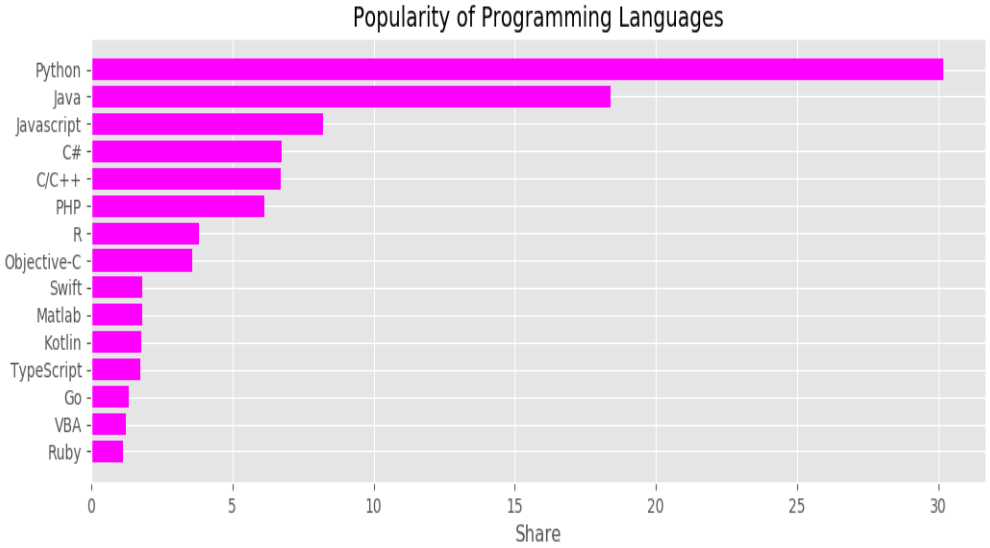


FIGURE 5. Popular programming languages