Notebook

January 29, 2019

Statement I $\frac{\sum_{i=1}^{n} a_i x_i}{\sum_{i=1}^{n} a_i} = \sum_{i=1}^{n} x_i$

This is false. If we let $n = 3$, and $a_1 = 1, a_2 = 2, a_3 = 3$, and $x_1 = -1, x_2 = -2, x_3 = -3$... The LHS evaluates to... $\frac{-1 + -4 + -9}{1 + 2 + 3}$ = $\frac{-14}{6}$ = $\frac{-7}{3}$ The RHS evaluates to... -1 + -2 + -3 = -6 As we can see, the LHS $\neq$ RHS.

Statement II $\sum_{i=1}^{n} x_1 = nx_1$

From above, we can see that... $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \sum_{i=1}^{n} x_1 = n * \bar{x} \sum_{i=1}^{n} x_1 = n * \frac{1}{n}(x_1 + x_1 + ... + x_n)$ $\sum_{i=1}^{n} x_1 = nx_1$

Statement III $\sum_{i=1}^{n} a_3 x_i = n a_3 \bar{x}$

From above, we can see that... $\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$ $n a_3 \bar{x} = n \frac{1}{n} a_3 \left(x_1 + x_2 + ... + x_n\right)$ $n a_3 \bar{x} = a_3 \sum_{i=1}^{n} x_i$ $n a_3 \bar{x} = \sum_{i=1}^{n} a_3 x_i$

Statement IV $\sum_{i=1}^{n} a_i x_i = n\bar{a}\bar{x}$

This is false. If we let $n = 3$, and $a_1 = 1, a_2 = 2, a_3 = 3$, and $x_1 = -1, x_2 = -2, x_3 = -3$... The LHS evaluates to... $-1 + -4 + -9 = -14$ The RHS evaluates to... $3\frac{(1+2+3)}{3}\frac{(-1+-2+-3)}{3} = (3)(2)(-2) = -12$ As we can see, the LHS $\neq$ RHS.

Question 4a Suppose we have the following scalar-valued function on x and y:

$$f(x,y) = x^2 + 4xy + 2y^3 + e^{-3y} + \ln(2y)$$

Compute the partial derivative of $f(x,y)$ with respect to x.

$\frac{\partial}{\partial x} = 2x + 4y$

Now compute the partial derivative of $f(x, y)$ with respect to y:
$\frac{\partial}{\partial y} = 4x + 6y^2 - 3e^{-3y} + \frac{1}{x}$

Finally, using your answers to the above two parts, compute $\nabla f(x,y)$ (the gradient of $f(x,y)$) and evaluate the gradient at the point $(x = 2, y = -1)$.

$\nabla f(x,y) = \frac{df}{dx}(x,y) + \frac{df}{dy}(x,y)$ $\nabla f(x,y) = 2(2) + 4(-1) + 4(2) + 6(-1)^2 - 3e^3 + 1/2$ $\nabla f(x,y) = 4 - 4 + 8 + 6 - 3e^3 + 1/2$ $\nabla f(x,y) = 14.5 - 3e^3$

Question 4b Find the value(s) of x which minimizes the expression below. Justify why it is the minimum.

$\sum_{i=1}^{10}(i-x)^2$

Let $\sigma(x) = \sum_{i=1}^{10}(i-x)^2$. $\frac{d}{dx}\sigma(x) = -2\sum_{i=1}^{10}(i-x) = 0\ \sum_{i=1}^{10}(i-x) = 0\ \sum_{i=1}^{10} i - \sum_{i=1}^{10} x = 0\ \sum_{i=1}^{10} i = \sum_{i=1}^{10} x\ 55 = 10x\ x = 5.5$

Question 4c Let $\sigma(x) = \frac{1}{1+e^{-x}}$. Show that $\sigma(-x) = 1 - \sigma(x)$.

Left Hand Side (LHS) $\sigma(-x) = \frac{1}{1+e^{x}}$ $\sigma(-x) = \frac{1}{1+e^{x}} * \frac{1-e^{-x}}{1-e^{-x}} = \frac{1-e^{-x}}{1-e^{-x}+e^{x}-e^{0}} = \frac{1-e^{-x}}{e^{x}-e^{-x}} * \frac{-1}{-1} = \frac{e^{-x}-1}{e^{-x}-e^{x}}$ Right Hand Side (RHS) $1 - \sigma(x) = 1 - \frac{1}{1+e^{-x}}$ $1 - \sigma(x) = \frac{1+e^{-x}-1}{1+e^{-x}} = \frac{e^{-x}}{1+e^{-x}} * \frac{1-e^{x}}{1-e^{x}} = \frac{e^{-x}-e^{0}}{1-e^{x}+e^{-x}-1} = \frac{e^{-x}-1}{e^{-x}-e^{x}}$ Since the LHS = RHS, we can see that

$$\sigma(-x) = 1 - \sigma(x)$$

.

Question 4d Show that the derivative can be written as:

$$\frac{d}{dx}\sigma(x) = \sigma(x)(1-\sigma(x))$$

Left Hand Side (LHS) $\frac{d}{dx}\sigma(x) = \frac{d}{dx}(1+e^{-x})^{-1}$ $\frac{d}{dx}\sigma(x) = -(1+e^{-x})^{-2}(-e^{-x})$ $\frac{d}{dx}\sigma(x) = \frac{e^{-x}}{(1+e^{-x})^2}$
Right Hand Side (RHS) $\sigma(x)(1-\sigma(x)) = \frac{1}{1+e^{-x}}(1-\frac{1}{1+e^{-x}}) = \frac{1}{1+e^{-x}} - \frac{1}{(1+e^{-x})^2} = \frac{1+e^{-x}-1}{(1+e^{-x})^2}$
$= \frac{e^{-x}}{(1+e^{-x})^2}$ Therefore, since LHS = RHS, we see that

$$\frac{d}{dx}\sigma(x) = \sigma(x)(1-\sigma(x))$$

.

Question 4e Write code to plot the function $f(x) = x^2$, the equation of the tangent line passing through $x = 8$, and the equation of the tangent line passing through $x = 0$.

Set the range of the x-axis to (-15, 15) and the range of the y axis to (-100, 300) and the figure size to (4,4).

Your resulting plot should look like this:

You should use the `plt.plot` function to plot lines. You may find the following functions useful:

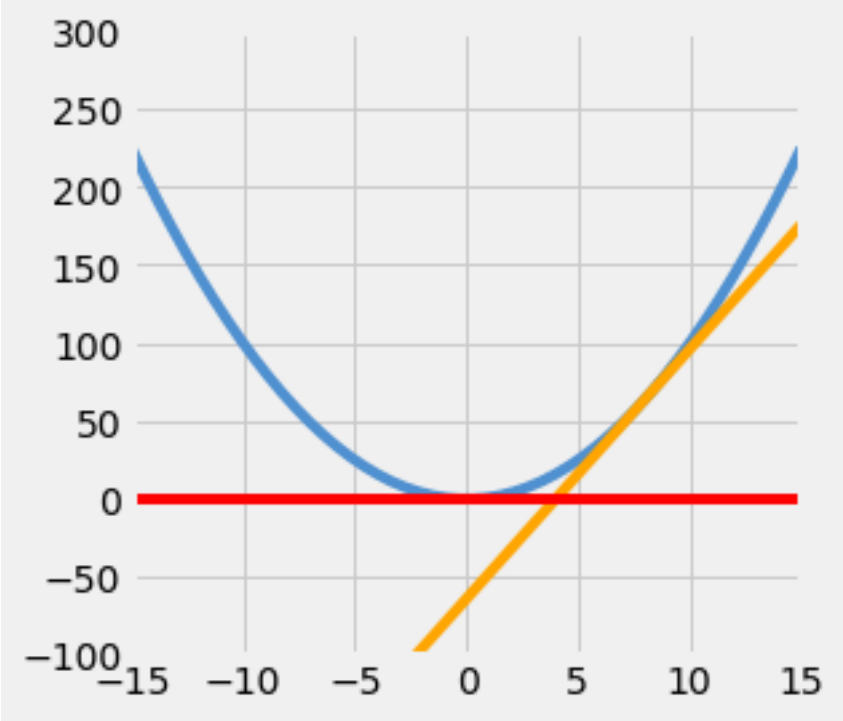
- `plt.plot(..)`
- `plt.figure(figsize=..)`
- `plt.ylim(..)`
- `plt.axhline(..)`

```
In [82]: def f(x):
             return x**2

         def df(x):
             return 16*x - 64

         def plot(f, df):
             plt.plot(xRange, f(xRange), color='#4f90d0')
             plt.plot(xRange, df(xRange), color='orange')
             plt.plot(xRange, xRange*0, color='red')

         xRange = np.linspace(-15, 15)
         plt.figure(figsize=(4,4))
         plt.axis([0, 15, -100, 300])
         plt.xticks(np.linspace(-15, 15, num=7))
         plot(f, df)
```



0.0.1 Question 5

Consider the following scenario:

Only 1% of 40-year-old women who participate in a routine mammography test have breast cancer. 80% of women who have breast cancer will test positive, but 9.6% of women who don't have breast cancer will also get positive tests.

Suppose we know that a woman of this age tested positive in a routine screening. What is the probability that she actually has breast cancer?

Hint: Use Bayes' rule.

$$\text{Pr(cancer\_when\_positive)} = \frac{\Pr(positive_when_cancer) * \Pr(cancer)}{\Pr(positive)} = \frac{0.01 * 0.8}{(0.01 * 0.8) + (0.99 * 0.096)} = \frac{0.008}{0.10304} = 7.76\%$$

0.0.2 Question 6

Generate a 2 by 2 plot that plots the function $g(t)$ as a line plot with values $f = 2, 8$ and $a = 2, 8$. Since there are 2 values of f and 2 values of a there are a total of 4 combinations, hence a 2 by 2 plot. The rows should vary in a and the columns should vary in f.

Set the x limit of all figures to $[0, \pi]$ and the y limit to $[-10, 10]$. The figure size should be 8 by 8. Make sure to label your x and y axes with the appropriate value of f or a. Additionally, make sure the x ticks are labeled $[0, \frac{\pi}{2}, \pi]$. Your overall plot should look something like the one above.

Hint 1: Modularize your code and use loops.

Hint 2: Are your plots too close together such that the labels are overlapping with other plots? Look at the `plt.subplots_adjust` function.

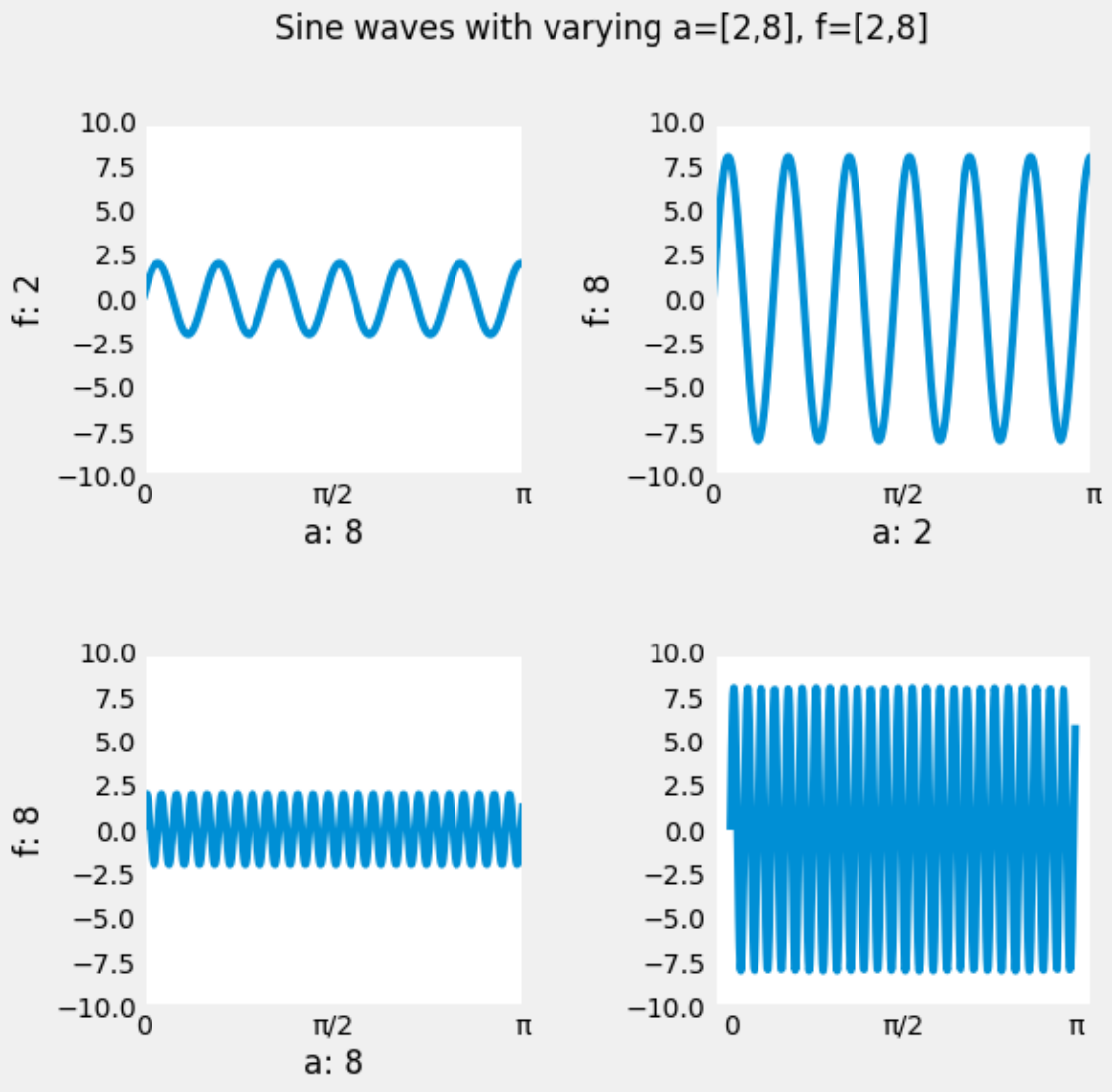
Hint 3: Having trouble setting the x-axis ticks and ticklabels? Look at the `plt.xticks` function.

Hint 4: You can add title to overall plot with `plt.suptitle`.

```
In [83]: def g(t, a, f):
             return a * np.sin(2 * np.pi * f * t)

         xRange = np.linspace(0, np.pi, num=500)
         plt.figure(figsize=(8,8))
         plt.suptitle("Sine waves with varying a=[2,8], f=[2,8]")
         combinations = [(2, 2), (8, 2), (2, 8), (8, 8)]
         i = 0
         subInd = 1

         for tuple in combinations:
             subplot = g(xRange, combinations[i][0], combinations[i][1])
             plt.xlabel("a: " + str(combinations[i][0]))
             plt.ylabel("f: " + str(combinations[i][1]))
             plt.axis([0, np.pi, -10, 10])
             plt.subplot(2, 2, subInd, facecolor='white')
             plt.subplots_adjust(wspace=0.5, hspace=0.5)
             plt.grid(b=None)
             plt.plot(xRange, subplot)
             plt.xticks([0, np.pi/2, np.pi], ['0', '/2', ''])
             plt.yticks(np.linspace(-10, 10, 9))
             subInd = subInd + 1
             i = i + 1
```


Sine waves with varying a=[2,8], f=[2,8]
f: 2
a: 8
f: 8
a: 2
f: 8
a: 8
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
0
π/2
π

0.0.3 Question 7

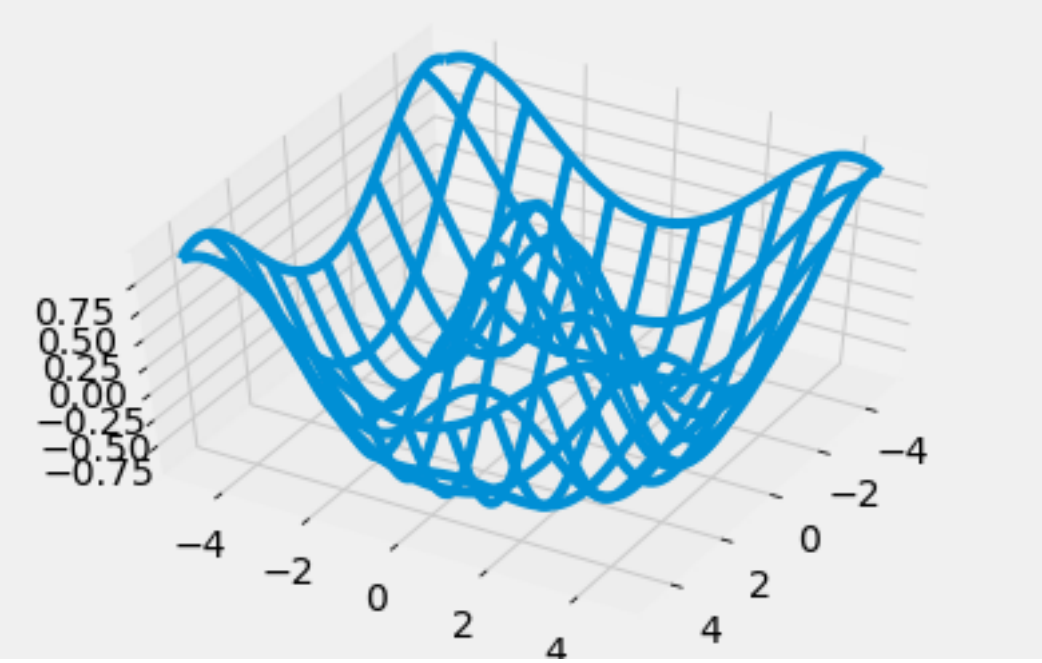
We should also familiarize ourselves with looking up documentation and learning how to read it. Below is a section of code that plots a basic wireframe. Replace each `# Your answer here` with a description of what the line above does, what the arguments being passed in are, and how the arguments are used in the function. For example,

```
np.arange(2, 5, 0.2)
# This returns an array of numbers from 2 to 5 with an interval size of 0.2
```

Hint: The `Shift + Tab` tip from earlier in the notebook may help here. Remember that objects must be defined in order for the documentation shortcut to work; for example, all of the documentation will show for method calls from `np` since we've already executed `import numpy as np`. However, since `z` is not yet defined in the kernel, `z.reshape()` will not show documentation until you run the line `z = np.cos(squared)`.

In [84]:
```
from mpl_toolkits.mplot3d import axes3d

u = np.linspace(1.5*np.pi, -1.5*np.pi, 100)
# This sets the value of u to be 100 evenly spaced points, calculated over the interval [1.5*n
[x,y] = np.meshgrid(u, u)
# Sets [x,y] to be coordinate matrices from coordinate vectors (u, u).
squared = np.sqrt(x.flatten()**2 + y.flatten()**2)
z = np.cos(squared)
# Takes the cosine of each element in @param: squared.
z = z.reshape(x.shape)
# Returns an array containing the same data with a new x.shape.
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
# Adds a 3-dimensional axis to the figure as part of a subplot arrangement,
# where pos is a three digit integer, such that the first digit (1) is the number of rows,
# the second digit (1) is the number of columns, and the third digit (1) is the index of the s
ax.plot_wireframe(x, y, z, rstride=10, cstride=10)
# Plots a 3-D wireframe using the x, y, z axes. The rstride (array row step size) and cstride
# arguments will determine at most how many evenly spaced samples (in this case 10) will be ta
ax.view_init(elev=50., azim=30)
# Set the elevation to 50 and azimuth to 30 of the axes.
plt.savefig("figure1.png")
# Save the current figure, in this case figure1.png.
```


0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−4
−2
0
2
4
−4
−2
0
2
4

0.0.4 Question 8

For a data-driven question of your choice, describe your approach and thought process in addressing this question. Outline what a sensible workflow might look like, including framing the question and identifying relevant data. Also consider transversal issues such as ethics and governance with respect to your question.

This question is about data-driven reasoning; you should focus more on *what* to do, than on *how* do to it exactly. You may use any of the questions presented in the first lecture, excluding the real estate, crowd size, and COMPAS questions. A complete response should contain about 250 words.

The data-driven question I would like to discuss is the Coordinated Entry System (CES) in Los Angeles, which attempts to find housing for the homeless. The problem with this system is that it collects fairly personal information about these homeless people (such as immigration and residency status, domestic violence history, health and mental health histories, substance use, sexuality, etc.). Afterward, a digital registry stores this data and runs an algorithm to determine a score which, in turn, is used to determine whether a homeless person can receive housing. The problem with this system is that even after divulging such personal information, there is no guarantee that a homeless person will receive housing. Furthermore, there is no guarantee that the CES will wipe away all the data stored on a particular individual. So, if a homeless person doesn't receive housing, he or she would have divulged most, if not all, of his or her personally identifiable information for absolutely nothing in return. I believe that a sensible workflow should hold the registry accountable for withholding homeless people's data. A reasonable question to ask is "Of all the homeless people who submit their personal information, how many actually receive housing offers and for the proportion that don't, are they concerned that they revealed all their information for nothing in return?" A good starting point would be to conduct a survey on the people who did and didn't receive housing and see whether this result was positively or negatively correlated with whether or not they were bothered by providing their personal information.