PythonNotebook6_solution_2023

December 20, 2023

0.1 Exercise 6.1.1 Plot the RICO input data

- split the matrix into separate variables for each column

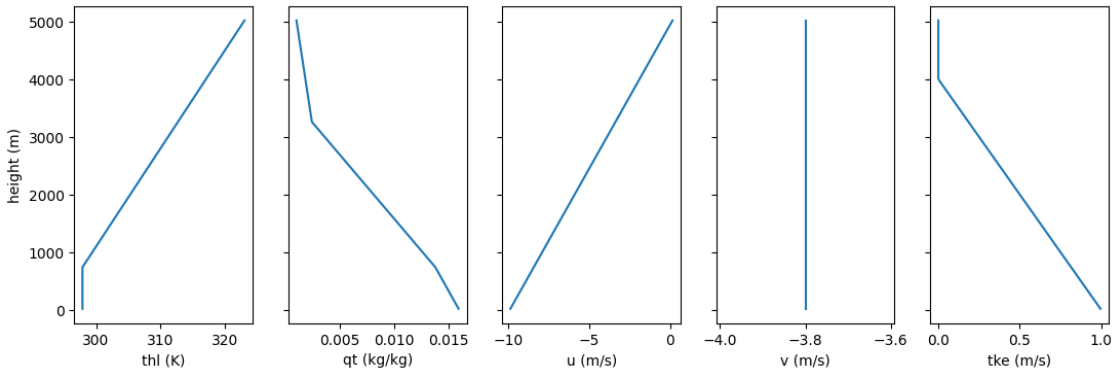
- use subplots to plot several quantities side by side: `fig, axes = plt.subplots(nrows=1, ncols=5, sharey=True, figsize=(13,4))`
- to plot in the first subplot, use `axes[0].plot(...)`
- to set x and y labels, use `axes[0].set_ylabel(...)`
- use the y axis of the plot for the hight above ground
- `sharey=True` makes the plots share the y axis
- figsize=(13,4) sets the figure size in inches (width,height) at some assumed dots-per-inch value. It's included here to make the plot a bit wider, so that the labels don't overlap.

[1]:
```
import matplotlib.pyplot as plt
import numpy as np

rico = np.loadtxt('../rico.txt') # in case the dataset is one folder up

# rico = np.loadtxt('rico.txt') # in case the dataset is in the same folder
```

[2]:
```
fig, axes = plt.subplots(nrows=1, ncols=5, sharey=True, figsize=(13,4))

z   = rico[:,0]
thl = rico[:,1]
qt  = rico[:,2]
u   = rico[:,3]
v   = rico[:,4]
tke = rico[:,5]

axes[0].plot(thl, z)
axes[1].plot(qt, z)
axes[2].plot(u, z)
axes[3].plot(v, z)
axes[4].plot(tke, z)
```

```
axes[0].set_ylabel('height (m)')

axes[0].set_xlabel('thl (K)')
axes[1].set_xlabel('qt (kg/kg)')
axes[2].set_xlabel('u (m/s)')
axes[3].set_xlabel('v (m/s)')
axes[4].set_xlabel('tke (m/s)')
```

[2]: `Text(0.5, 0, 'tke (m/s)')`



0.2 Exercise 6.2.1 More surface plots

- make surface plots of a few other function of your choice
- you can use the same meshgrid as above or choose a different one if you want other x and y ranges.

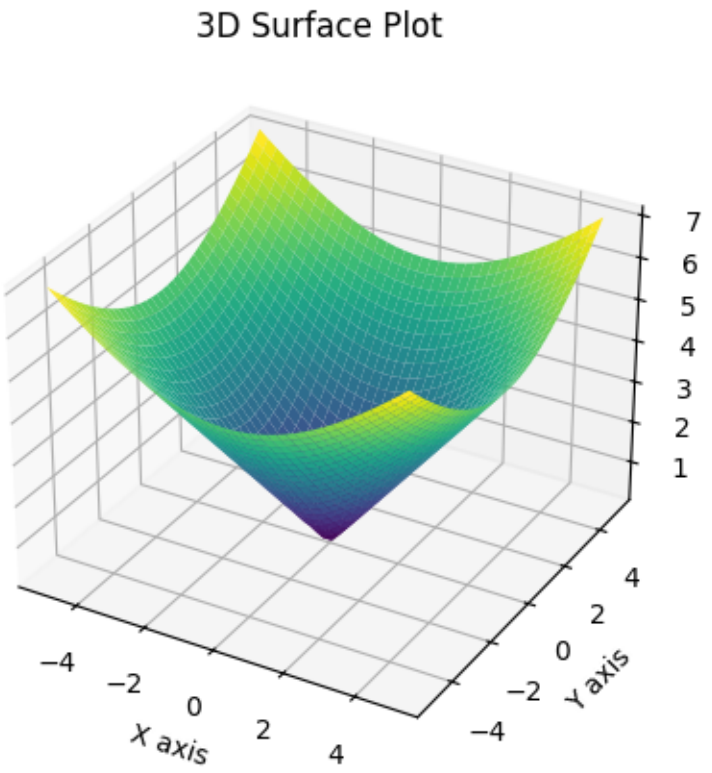
0.2.1 some inspiration

...which you can combine with other things

```
R = np.sqrt(X**2 + Y**2)  # distance to origo
h = np.sin(1.4*X)         # a wave
```

[3]:
```
x = np.linspace(-5, 5, 40) # create new coordinate arrays with more points
y = np.linspace(-5, 5, 40)

X,Y = np.meshgrid(x,y)

R = np.sqrt(X**2 + Y**2)

fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

surf = ax.plot_surface(X, Y, R, cmap='viridis')
```

```
ax.set_xlabel('X axis')
ax.set_ylabel('Y axis')
ax.set_zlabel('R')
ax.set_title('3D Surface Plot')

plt.show()
```

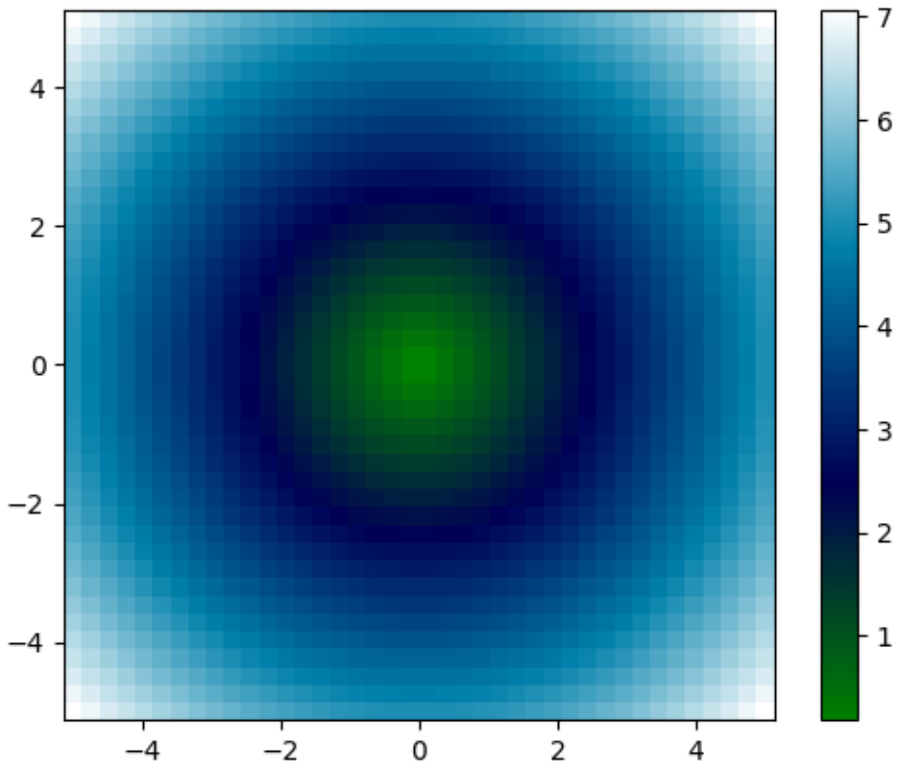


0.3 Exercise 6.2.2 ...and color plots

Plot your functions from above using color maps. Remember to include the color bar. Choose color maps you like from here, and apply them with the `cmap` argument:

`plt.pcolormesh(X,Y,Z,cmap='ocean')`

[4]:
```
plt.gca().set_aspect('equal')

plt.pcolormesh(X,Y,R, cmap='ocean')
plt.colorbar()

plt.show()
```



0.4 6.3 List and loop repetition

0.5 Exercise 6.3.1

- Make a list
- Check it twice (by printing it's elements once from the start and once from the end)

[5]:
```
my_list = [i for i in range(10)]

# printing from the start
print("Printing my list from the start to end.")
for val in my_list:
    print(val, end=' ') # note the `end` argument here to print with spaces␣
 ↪instead of new lines

# printing from the end

print("\nPrinting my list from the end to start.")
for val in my_list[::-1]:
    print(val, end= ' ')
```

```
Printing my list from the start to end.
0 1 2 3 4 5 6 7 8 9
```

```
Printing my list from the end to start.
9 8 7 6 5 4 3 2 1 0
```