

# Debugging for Scientists



Nipun Batra  
July 27, 2017

# Asking Questions the Bad Way!

---

## Plotting a rectangular area on the surface of a sphere

I am trying to trace out a rectangular area on the surface of a sphere this is the code I have for the sphere

```
import numpy as np
import random as rand
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

fig = plt.figure()

ax = fig.gca(projection='3d')

ax.set_aspect("equal")

theta, phi = np.mgrid[0:2*np.pi : 20j , 0:np.pi : 20j]

r = 6.3

x = r * np.cos(phi)*np.sin(theta)
y = r * np.sin(phi)*np.sin(theta)
```

- 2 "I am trying" seems to be a euphemism. Have you actually tried anything? What does "trace out" mean in this context? Which are the coordinates to plot? How would the expected plot look like? – [ImportanceOfBeingErnest](#) 15 hours ago

the points to plot the rectangle will be converted from lat/long to cart coords.

# Minimal, Complete, and Verifiable example

from StackOverflow's How to create a Minimal, Complete, and Verifiable example

---

Minimal – Use as little code as possible that still produces the same problem

Complete – Provide all parts needed to reproduce the problem

Verifiable – Test the code you're about to provide to make sure it reproduces the problem

# Minimal

---

**Restart from scratch.** Create a new program, adding in only what is needed to see the problem. This can be faster for vast systems where you think you already know the source of the problem. Also useful if you can't post the original code publicly for legal or ethical reasons.

**Divide and conquer.** When you have a small amount of code, but the source of the problem is entirely unclear, start removing code a bit at a time until the problem disappears – then add the last part back.

# Not Minimal Example

---

## Why is my graph a crazy flickering monster?



I am trying to create a graph showing the correlation between mini batch accuracy and validation accuracy of a neural net. But instead, I have a crazy graph that is flickering at a super high frequency and is zoomed in on a very small portion of the graph.

Here is my code:

```
num_nodes=1024
batch_size = 128
beta = 0.01

def animate(i):
    graph_data = open('NeuralNetData.txt','r').read()
    lines = graph_data.split('\n')
    xs = []
    ys = []
    for line in lines:
        if len(line) > 1:
            x, y = line.split(',')
            xs.append(x)
            ys.append(y)
    ax1.clear()
    ax1.plot(xs, ys, label='validation accuracy')
    ax1.legend(loc='lower right')
    ax1.set_ylabel("Accuracy(%)", fontsize=15)
    ax1.set_xlabel("Images Seen", fontsize=15)
    ax1.set_title("Neural Network Accuracy Data\nStochastic Gradient Descent", fontsize=15)
    plt.show()

def animate2(i):
    graph_data = open('NeuralNetData2.txt','r').read()
    lines = graph_data.split('\n')
    xs = []
    ys = []
    for line in lines:
        if len(line) > 1:
            x, y = line.split(',')
            xs.append(x)
            ys.append(y)
    ax1.plot(xs, ys, label='mini-batch accuracy')
    ax1.legend(loc='lower right')
    plt.tight_layout()
```

# Complete

---

Make sure all information necessary to reproduce the problem is included:

- Some people might be prepared to load the parts up, and actually try them to test the answer they're about to post.
- The problem might not be in the part you suspect it is, but another part entirely.



# Incomplete Example

---

## Matplotlib scatter color by categorical factors

▲ I have a basic scatter where the x and y are float. But I want to change the color of the marker based on a third categorical variable. The categorical variable is in a string form. This seems to cause an issue.

4

▼ To use the iris dataset- here is the code I think I would use:



```
#Scatter of Petal
x=df['Petal Length']
y=df['Petal Width']
z=df['Species']
plt.scatter(x, y, c=z, s=15, cmap='hot')
plt.xlabel('Petal Width')
plt.ylabel('Petal Length')
plt.title('Petal Width vs Length')
```

But I get an error that: could not convert string to float: iris-setosa

Do I have to change the categorical variable to a numeric one before I run, or is there something I can do with the data in its current format?

Thanks

update: the entire traceback is:

```
-----
ValueError                                Traceback (most recent call last)
<ipython-input-47-d67ee3bffc3b> in <module>()
      3 y=df['Petal Width']
      4 z=df['Species']
----> 5 plt.scatter(x, y, c=z, s=15, cmap='hot')
      6 plt.xlabel('Petal Width')
      7 plt.ylabel('Petal Length')
```

# Complete Example

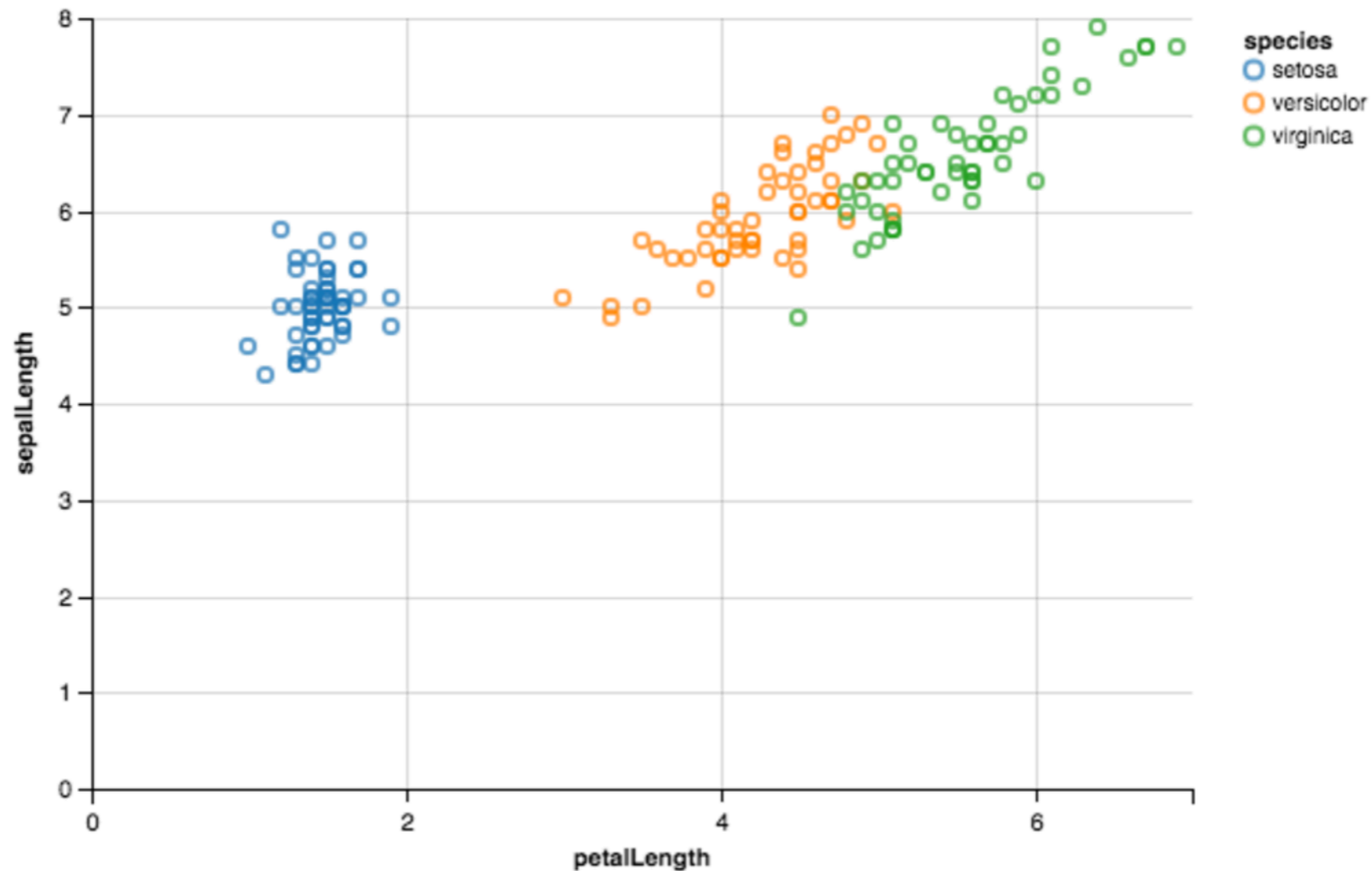
---

▲ Altair should be a breeze here.

0

```
from altair import *
import pandas as pd

df = datasets.load_dataset('iris')
Chart(df).mark_point().encode(x='petalLength', y='sepalLength', color='species')
```





# Verifiable

---

- Ensure that the example actually reproduces the problem!
- Eliminate any issues that aren't relevant to the problem.

# Exercise #1

---

Open StackOverflow. Find questions in your language that do not meet the requirements of Minimal, Verifiable and Complete.