In [13]:

```
import pandas as pd
import numpy as np
import seaborn as sns

df = sns.load_dataset('iris')
df.head()
```

Out[13]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

In [ ]:

In [14]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   sepal_length  150 non-null    float64
 1   sepal_width   150 non-null    float64
 2   petal_length  150 non-null    float64
 3   petal_width   150 non-null    float64
 4   species       150 non-null    object 
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

In [15]:

```
x = df.iloc[ : , : -1]
y = df.iloc[ : ,-1: ]
```

In [16]:

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(x,y, train_size=0.8, ranc
```

In [18]:

```
from sklearn.ensemble import RandomForestClassifier

rf = RandomForestClassifier(n_estimators=100).fit(X_train, y_train)
rf
```

```
C:\Users\MohHu\AppData\Local\Temp\ipykernel_12040\1216515390.py:3: DataConv
ersionWarning: A column-vector y was passed when a 1d array was expected. P
lease change the shape of y to (n_samples,), for example using ravel().
  rf = RandomForestClassifier(n_estimators=100).fit(X_train, y_train)
```

Out[18]:

```
RandomForestClassifier()
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [21]:

```
model = rf.predict(X_test)
```

In [26]:

```
model.score()
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
c:\Users\MohHu\Downloads\Python_learning\Python_VS\Machine_Learning\Random_
Forest.ipynb Cell 7 in <cell line: 1>()
----> <a href='vscode-notebook-cell:/c%3A/Users/MohHu/Downloads/Python_lear
ning/Python_VS/Machine_Learning/Random_Forest.ipynb#ch0000006?line=0'>1</a>
model.score()

AttributeError: 'numpy.ndarray' object has no attribute 'score'
```

In [24]:

```
from sklearn import metrics
```

In [25]:

```
from sklearn.metrics import accuracy_score


accuracy_score(y_test, model)
```

Out[25]:

```
0.9666666666666667
```

In [27]:

```
metrics.confusion_matrix(y_test, model)
```

Out[27]:

```
array([[10,  0,  0],
       [ 0, 12,  0],
       [ 0,  1,  7]], dtype=int64)
```

In [30]:

```
import matplotlib.pyplot as plt

plt.figure(figsize=(9,9))
sns.heatmap(metrics.confusion_matrix(y_test, model), annot="True", linewidths
plt.ylabel("Actual Output");
plt.xlabel("Predicted Output");
all_sample_title = "Accuracy Score:{0}".format(score)
plt.title(all_sample_title, size=15)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
c:\Users\MohHu\Downloads\Python_learning\Python_VS\Machine_Learning\Random_
Forest.ipynb Cell 11 in <cell line: 4>()
      <a href='vscode-notebook-cell:/c%3A/Users/MohHu/Downloads/Python_lear
ning/Python_VS/Machine_Learning/Random_Forest.ipynb#ch0000010?line=0'>1</a>
import matplotlib.pyplot as plt
      <a href='vscode-notebook-cell:/c%3A/Users/MohHu/Downloads/Python_lear
ning/Python_VS/Machine_Learning/Random_Forest.ipynb#ch0000010?line=2'>3</a>
plt.figure(figsize=(9,9))
----> <a href='vscode-notebook-cell:/c%3A/Users/MohHu/Downloads/Python_lear
ning/Python_VS/Machine_Learning/Random_Forest.ipynb#ch0000010?line=3'>4</a>
sns.heatmap(metrics.confusion_matrix(y_test, model), annot="True", linewidt
hs=.5, square=True, camp='Spectral');
      <a href='vscode-notebook-cell:/c%3A/Users/MohHu/Downloads/Python_lear
ning/Python_VS/Machine_Learning/Random_Forest.ipynb#ch0000010?line=4'>5</a>
plt.ylabel("Actual Output");
      <a href='vscode-notebook-cell:/c%3A/Users/MohHu/Downloads/Python_lear
ning/Python_VS/Machine_Learning/Random_Forest.ipynb#ch0000010?line=5'>6</a>
plt.xlabel("Predicted Output");

File c:\Users\MohHu\AppData\Local\Programs\Python\Python310\lib\site-packag
es\seaborn\_decorators.py:46, in _deprecate_positional_args.<locals>.inner_
f(*args, **kwargs)
     36     warnings.warn(
     37         "Pass the following variable{} as {}keyword arg{}: {}. "
     38         "From version 0.12, the only valid positional argument "
   (...)
     43         FutureWarning
     44     )
     45 kwargs.update({k: arg for k, arg in zip(sig.parameters, args)})
---> 46 return f(**kwargs)

File c:\Users\MohHu\AppData\Local\Programs\Python\Python310\lib\site-packag
es\seaborn\matrix.py:540, in heatmap(data, vmin, vmax, cmap, center, robus
t, annot, fmt, annot_kws, linewidths, linecolor, cbar, cbar_kws, cbar_ax, s
quare, xticklabels, yticklabels, mask, ax, **kwargs)
    362 """Plot rectangular data as a color-encoded matrix.
    363 
    364 This is an Axes-level function and will draw the heatmap into the
   (...)
    537     ...     ax = sns.heatmap(corr, mask=mask, vmax=.3, square=True)
    538 """
    539 # Initialize the plotter object
--> 540 plotter = _HeatMapper(data, vmin, vmax, cmap, center, robust, anno
t, fmt,
    541                       annot_kws, cbar, cbar_kws, xticklabels,
    542                       yticklabels, mask)
    544 # Add the pcolormesh kwargs here
    545 kwargs["linewidths"] = linewidths

File c:\Users\MohHu\AppData\Local\Programs\Python\Python310\lib\site-packag
es\seaborn\matrix.py:173, in _HeatMapper.__init__(self, data, vmin, vmax, c
map, center, robust, annot, fmt, annot_kws, cbar, cbar_kws, xticklabels, yt
icklabels, mask)
    171         if annot_data.shape != plot_data.shape:
    172             err = "`data` and `annot` must have same shape."
--> 173             raise ValueError(err)
    174     annot = True
    176 # Save other attributes to the object

ValueError: `data` and `annot` must have same shape.


<Figure size 648x648 with 0 Axes>
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: