Practical 37

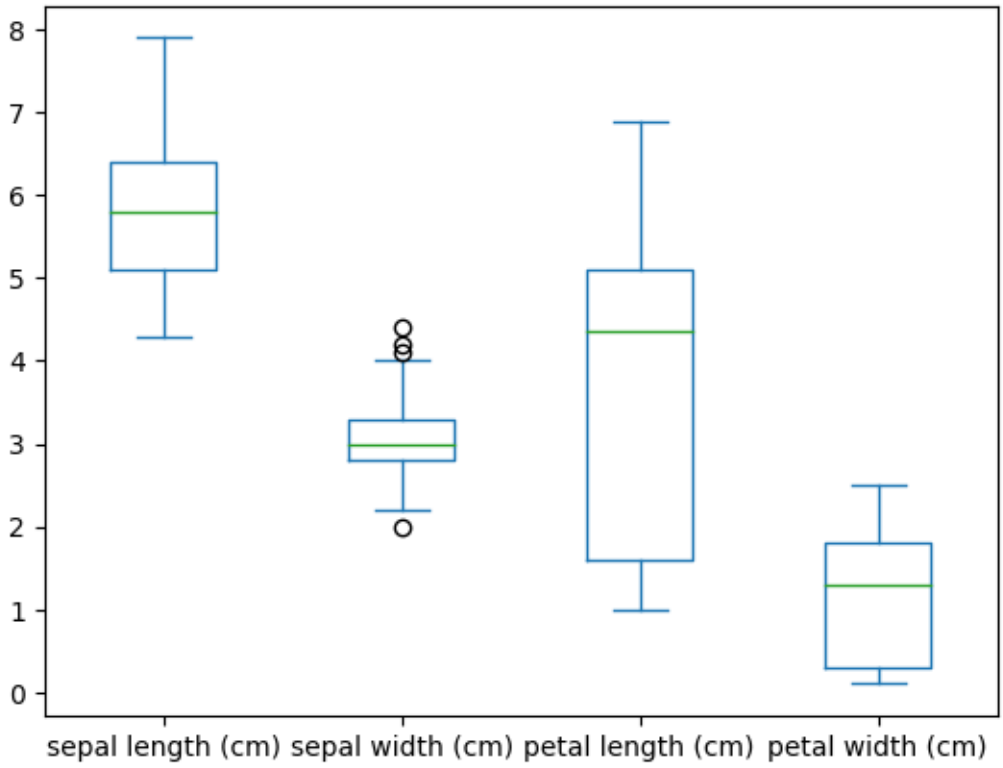
Write a python program to create applied visualization for EDA using boxplots and perform t-tests.

[2]:
```
from sklearn.datasets import load_iris
import pandas as pd

iris = load_iris()
df = pd.DataFrame(iris.data, columns=iris.feature_names)
df['Alien'] = pd.Series([iris.target_names[k] for k in iris.target],␣
 ↪dtype='category')  # Corrected dtype

df.plot(kind='box')
Alien1 = df['Alien'] == 'setosa'
Alien2 = df['Alien'] == 'versicolor'
print(Alien1)
```

```
0       True
1       True
2       True
3       True
4       True
       …
145    False
146    False
147    False
148    False
149    False
Name: Alien, Length: 150, dtype: bool
```



[3]:
```
from scipy.stats import ttest_ind

t, pvalue = ttest_ind(df['petal length (cm)'][Alien1], df['petal length␣
↪(cm)'][Alien2])  # Fixed the second argument

print("P-value:", pvalue)
```

```
P-value: 5.404910513441677e-62
```

[4]:
```
import numpy as np

# Create an array 'ages' with length 30 and fill with random integer values
ages = np.random.randint(0, 100, size=30)  # Generates random integers between␣
↪0 and 99 (inclusive)

# Print the array to check its contents
print(len(ages))
```

```
30
```

[5]:
```
mean = np.mean(ages)
mean
```

[5]:
```
46.766666666666666
```

[6]:
```
sample_size=10
age_sample=np.random.choice(ages,sample_size)
age_sample
```

[6]:
```
array([70, 12, 19, 24, 84, 19, 57, 52, 59, 10])
```

[7]:
```
from scipy.stats import ttest_1samp
ttest,p_value=ttest_1samp(age_sample,30)
print(p_value)
```

```
0.24203093538542797
```

[9]:
```
if p_value < 0.05: # alpha value is 0.05 or 5%
    print(" we are rejecting null hypothesis")
else:
    print("we are accepting null hypothesis")
```

```
we are accepting null hypothesis
```