

Name :sandip gadadare.
Roll no: CO3A09. ASSIGNMENT NO:-10

Div:A
BATCH:B1

Q.Data Visualization III

Download the Iris flower dataset or any other dataset into a DataFrame. (e.g., <https://archive.ics.uci.edu/ml/datasets/Iris>). Scan the dataset and give the inference as:

1. List down the features and their types (e.g., numeric, nominal) available in the dataset.
2. Create a histogram for each feature in the dataset to illustrate the feature distributions.
3. Create a boxplot for each feature in the dataset.
4. Compare distributions and identify outliers.

CODE:-

[1]:-

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
```

[2]:-

```
iris = sns.load_dataset("iris")
```

iris

out[2]:-

	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
...
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

150 rows × 5 columns

```
[3]:-
iris.describe()
out[3]:-
```

	sepal_length	sepal_width	petal_length	petal_width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333
std	0.828066	0.435866	1.765298	0.762238
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

```
[4]:-
iris.info()
out[4]:-
<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
```

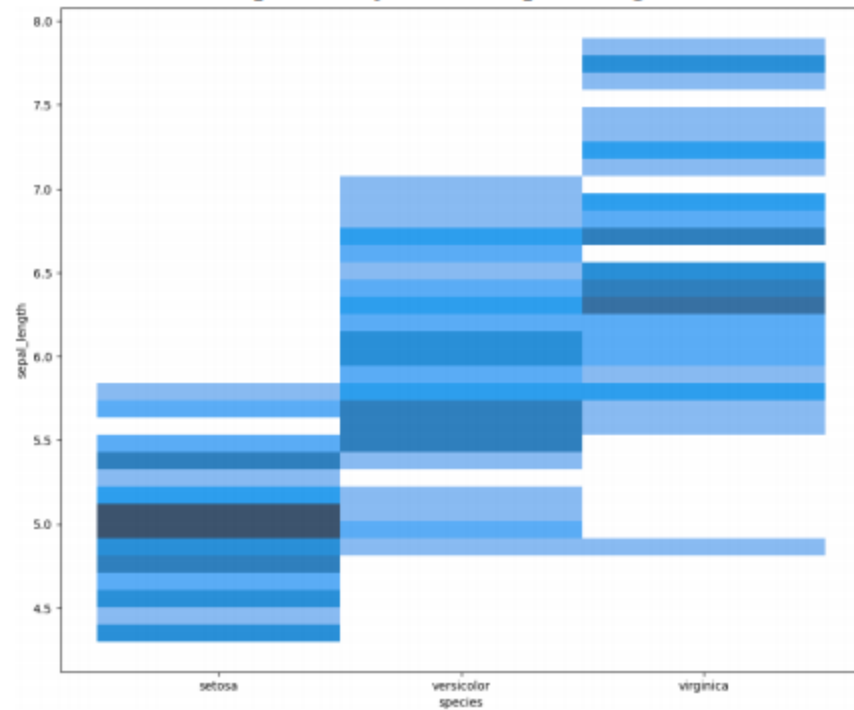
```
[5]:-
iris.isnull().sum()
```

```
out[5]:-
sepal_length    0
sepal_width     0
petal_length    0
petal_width     0
species         0
dtype: int64
```

```
[6]:-
plt.figure(figsize=(12,10))
sns.histplot(data=iris, x="species", y="sepal_length", bins=35)
```

out[6]:-

<Axes: xlabel='species', ylabel='sepal_length'>



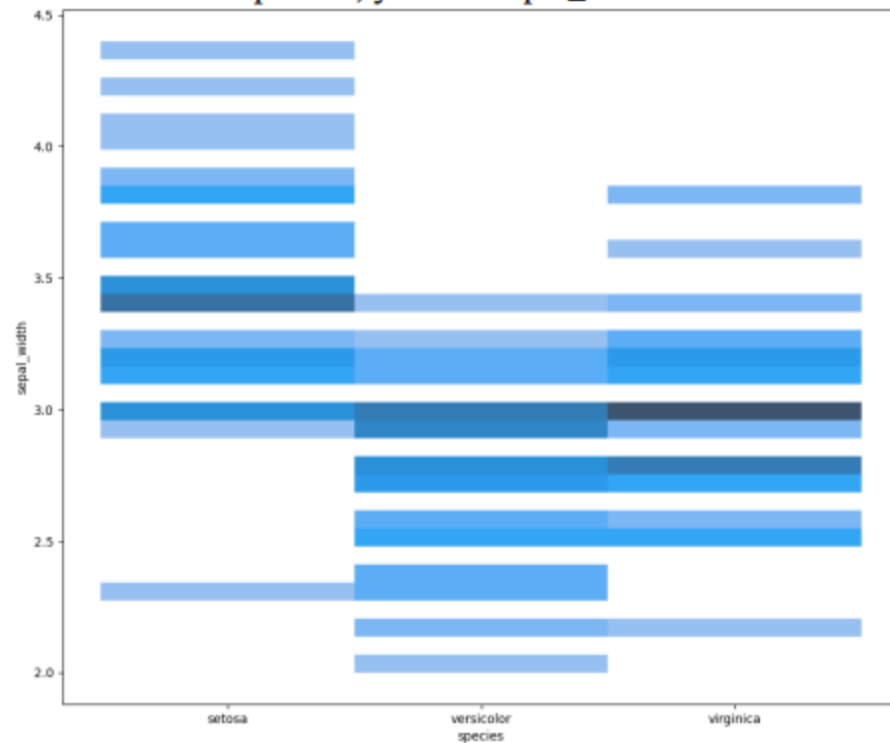
[7]:-

plt.figure(figsize=(12,10))

sns.histplot(data=iris, x="species", y="sepal_width", bins=35)

out[7]:-

<Axes: xlabel='species', ylabel='sepal_width'>



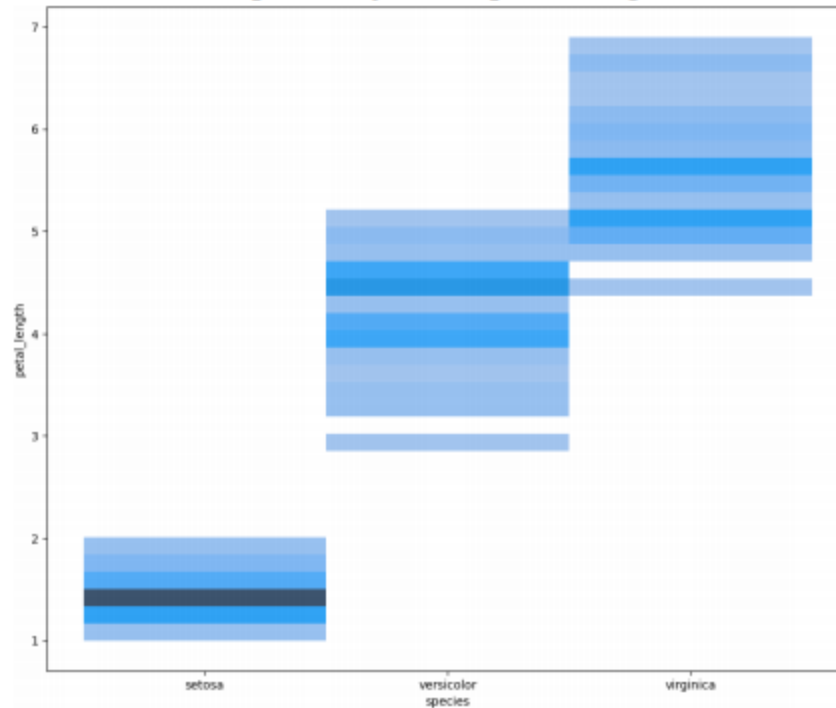
[8]:-

```
plt.figure(figsize=(12,10))
```

```
sns.histplot(data=iris, x="species", y="petal_length", bins=35)
```

out[8]:-

<Axes: xlabel='species', ylabel='petal_length'>



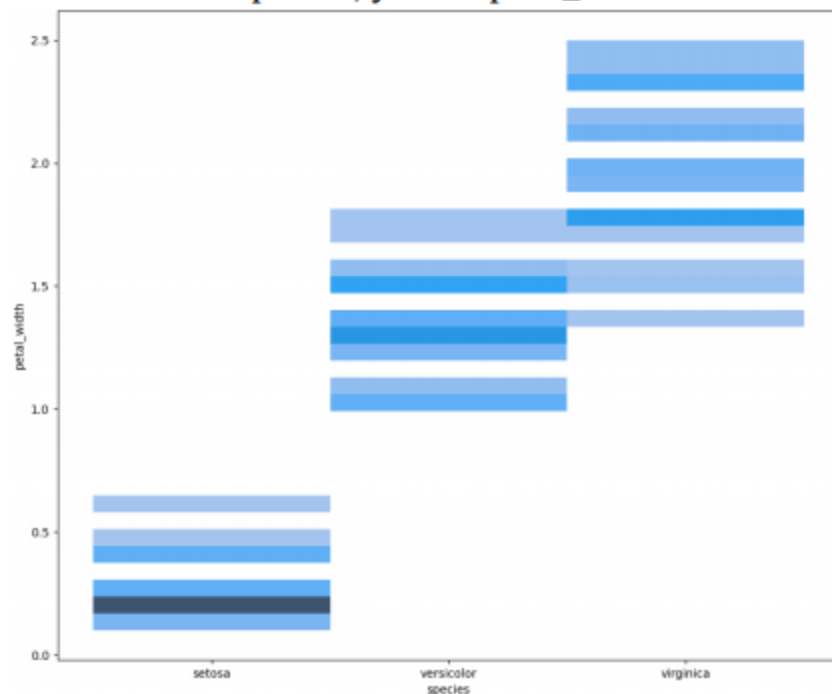
[9]:-

```
plt.figure(figsize=(12,10))
```

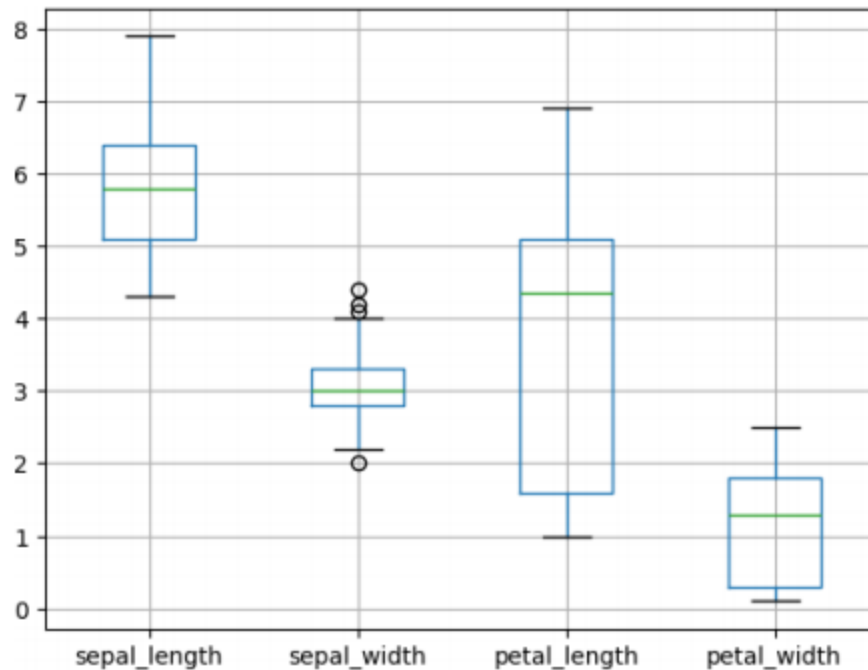
```
sns.histplot(data=iris, x="species", y="petal_width", bins=35)
```

out[9]:-

<Axes: xlabel='species', ylabel='petal_width'>



```
[10]:-
numerical_col = ['sepal_length','sepal_width','petal_length','petal_width']
categorical_col = ['species']
iris.boxplot(numerical_col)
out[10]:-
<Axes: >
```



```
[11]:-
for x in ['sepal_width']:
q75,q25 = np.percentile(iris.loc[:,x],[75,25])
intr_qr = q75-q25
max = q75+(1.5*intr_qr)
min = q25-(1.5*intr_qr)
iris.loc[iris[x] < min,x] = np.nan
iris.loc[iris[x] > max,x] = np.nan
[12]:-
print("Sum of count of NULL values/outliers in each column of the dataset:")
iris.isnull().sum()
out[12]:-
Sum of count of NULL values/outliers in each column of the dataset:
sepal_length    0
sepal_width     4
petal_length    0
petal_width     0
species         0
dtype: int64
[13]:-
iris['sepal_width'].describe()
out[13]:-
count    146.000000
```

```
mean    3.040411
std     0.397853
min     2.200000
25%     2.800000
50%     3.000000
75%     3.300000
max     4.000000
Name: sepal_width, dtype: float64
```

[14]:-

```
iris.boxplot(numerical_col)
```

out[14]:-

