p10

March 30, 2024

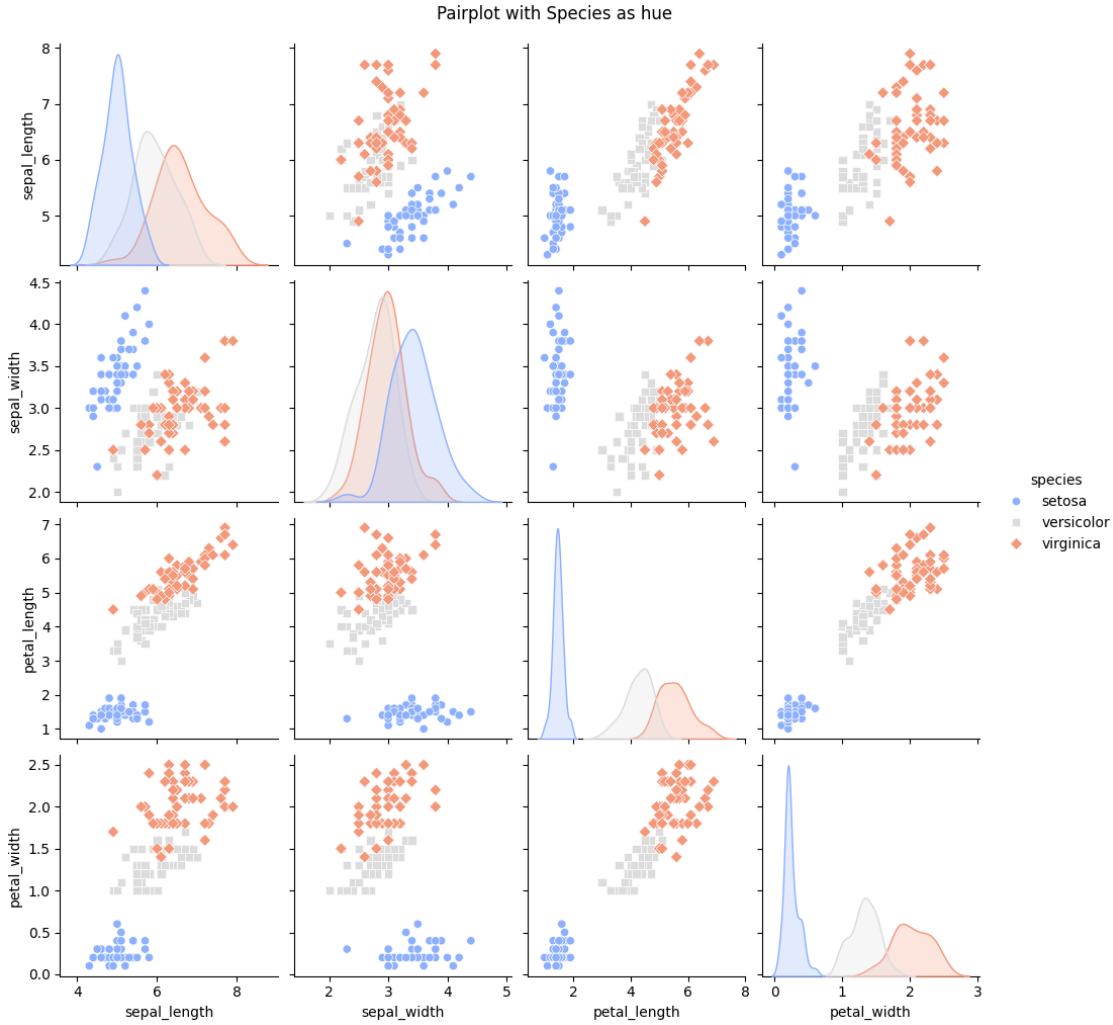
[2]:
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns

df =sns.load_dataset('iris')
print(df)
br = "\n\n"
print(br, df.info())
print(br, df.dtypes)
```

```
     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 x 5 columns]
<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
```

```
None
```

```
sepal_length    float64
sepal_width     float64
petal_length    float64
petal_width     float64
species          object
dtype: object
```

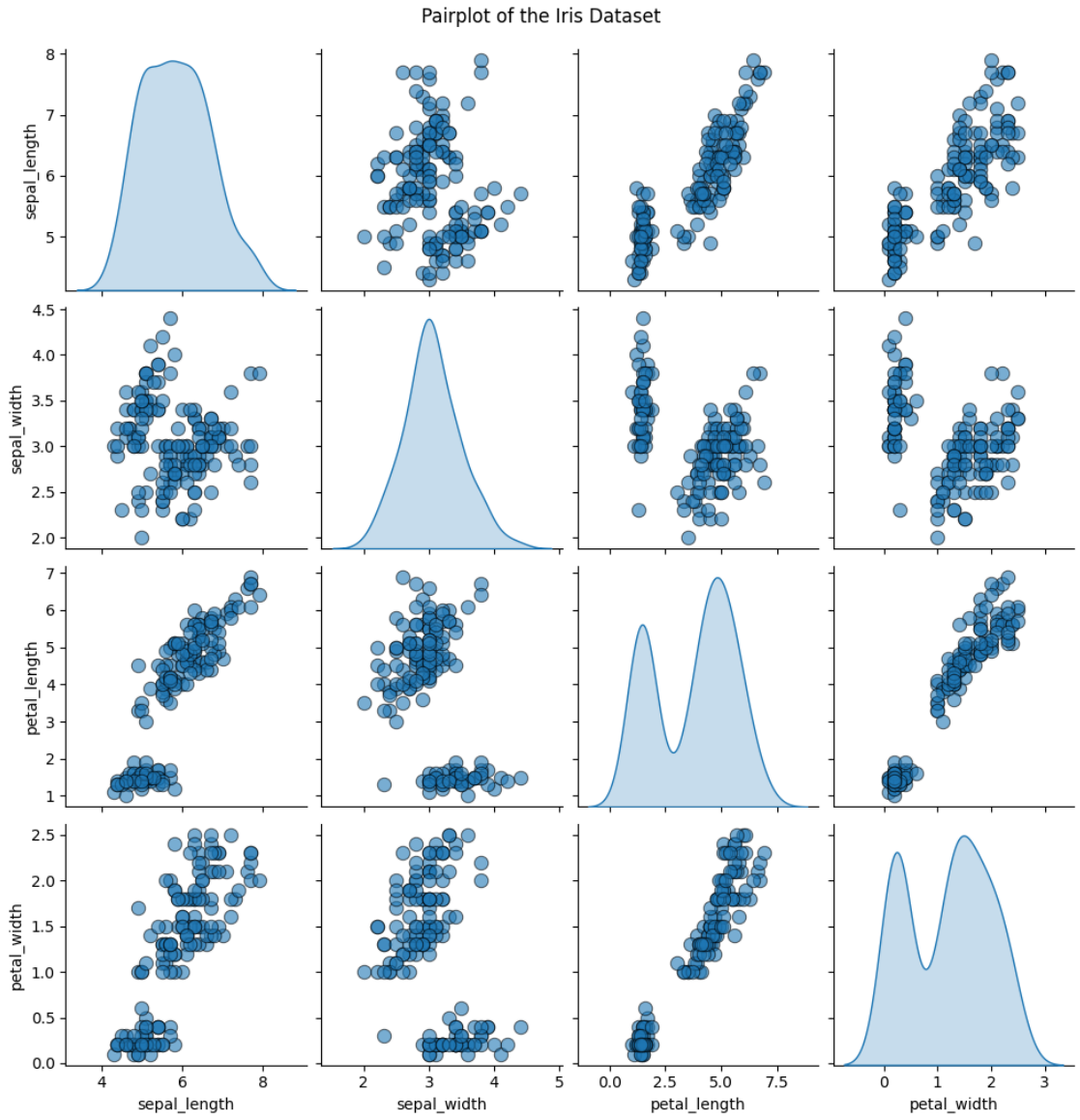
[3]:
```
# Create a pairplot with hue as 'species' and a custom color palette
g = sns.pairplot(df, hue='species', palette='coolwarm', markers=["o", "s", "D"])

# Set the title of the pairplot
g.fig.suptitle('Pairplot with Species as hue', y=1.02)

# Display the pairplot
plt.show()
```



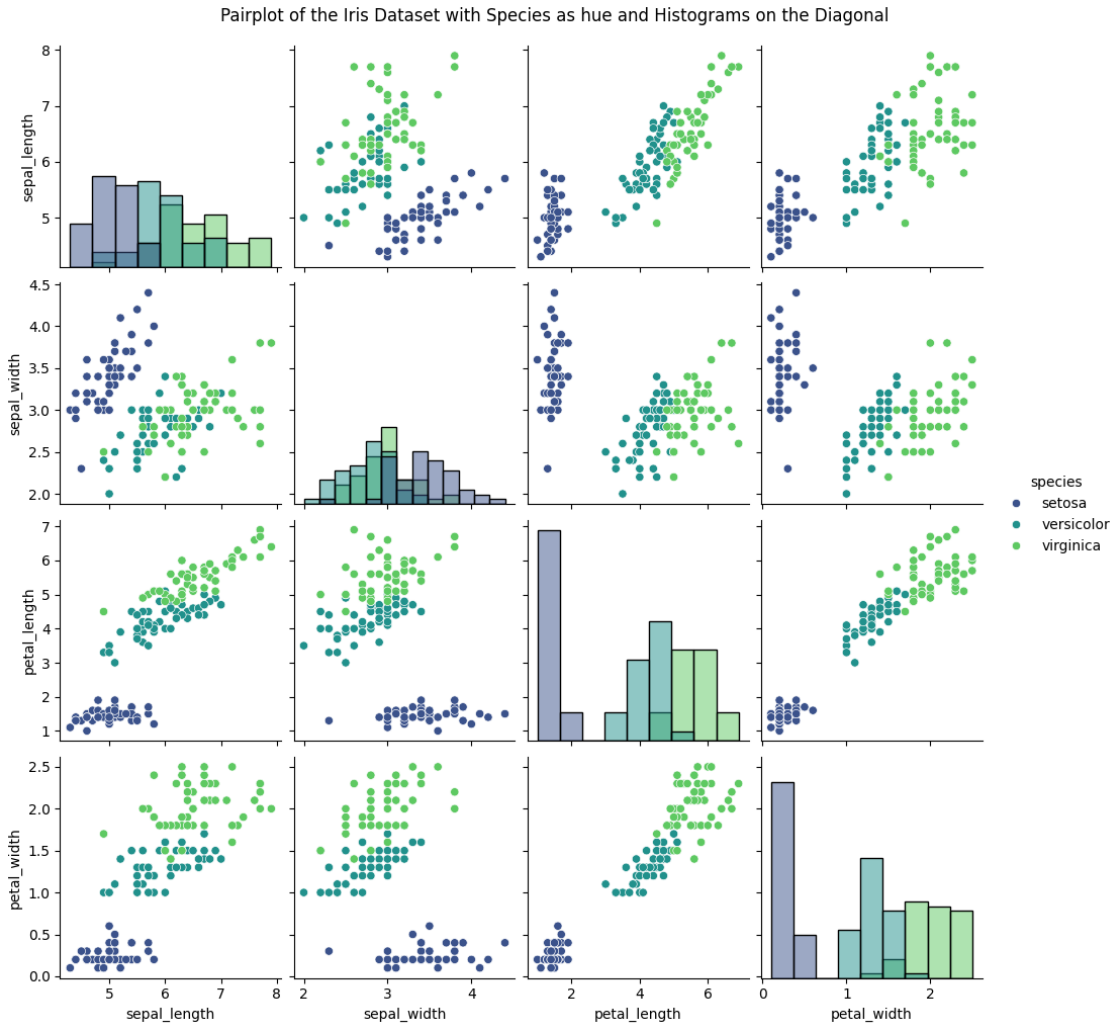
[8]:
```
# Create a pairplot with a custom color palette
g = sns.pairplot(df, palette='coolwarm', diag_kind='kde', plot_kws={'alpha': 0.
  ↪6, 's': 80, 'edgecolor': 'k'})

# Set the title of the pairplot
g.fig.suptitle('Pairplot of the Iris Dataset', y=1.02)

# Display the pairplot
plt.show()
```

```
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1513: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=vector, **plot_kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1513: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=vector, **plot_kwargs)
```

```
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1513: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=vector, **plot_kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1513: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=vector, **plot_kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1615: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=x, y=y, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1615: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=x, y=y, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1615: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=x, y=y, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1615: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=x, y=y, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1615: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=x, y=y, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1615: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=x, y=y, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1615: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=x, y=y, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1615: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=x, y=y, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1615: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=x, y=y, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1615: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=x, y=y, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1615: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=x, y=y, **kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:1615: UserWarning:
Ignoring `palette` because no `hue` variable has been assigned.
  func(x=x, y=y, **kwargs)
```

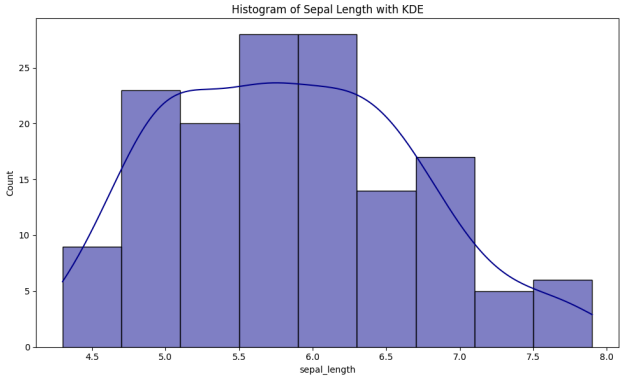


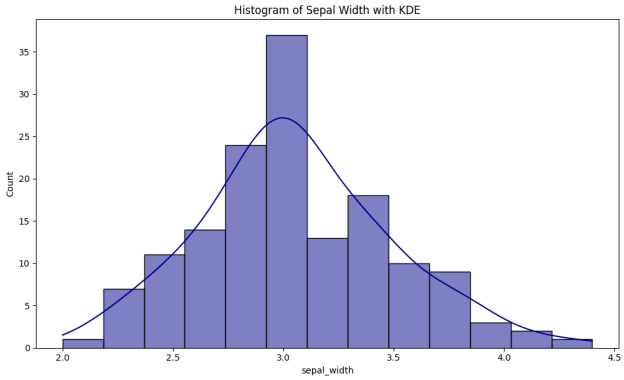
[9]:
```
# Create a pairplot with hue as 'species', diag_kind as 'hist', and a custom␣
 ↪color palette
g = sns.pairplot(df, hue='species', palette='viridis', diag_kind='hist')

# Set the title of the pairplot
g.fig.suptitle('Pairplot of the Iris Dataset with Species as hue and Histograms␣
 ↪on the Diagonal', y=1.02)

# Display the pairplot
plt.show()
```

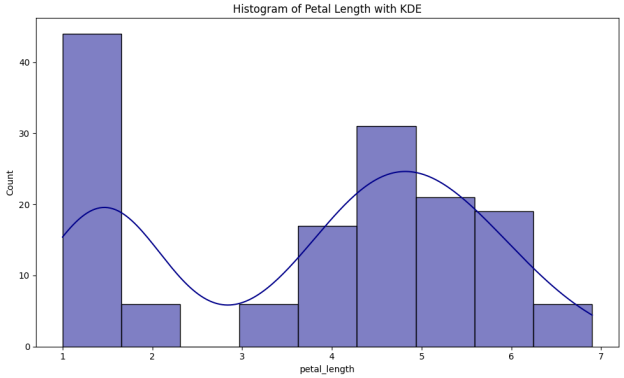


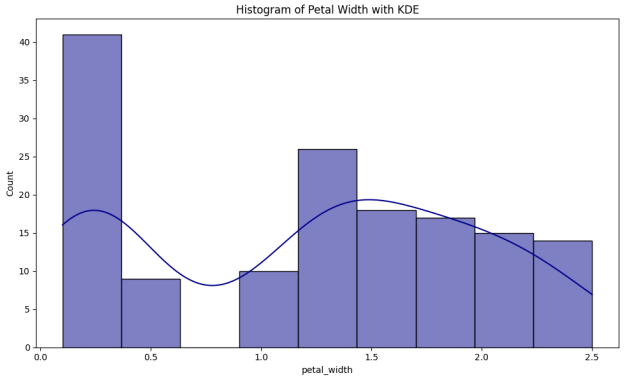
[10]:
```
# Create a new figure and a set of subplots
fig, axs = plt.subplots(5, 1, figsize=(10, 30))

# Create a histogram for 'sepal_length' with additional kde plot
sns.histplot(data=df, x='sepal_length', kde=True, color='darkblue', ax=axs[0])
axs[0].set_title('Histogram of Sepal Length with KDE')

# Create a histogram for 'sepal_width' with additional kde plot
sns.histplot(data=df, x='sepal_width', kde=True, color='darkblue', ax=axs[1])
axs[1].set_title('Histogram of Sepal Width with KDE')

# Create a histogram for 'petal_length' with additional kde plot
sns.histplot(data=df, x='petal_length', kde=True, color='darkblue', ax=axs[2])
axs[2].set_title('Histogram of Petal Length with KDE')
```

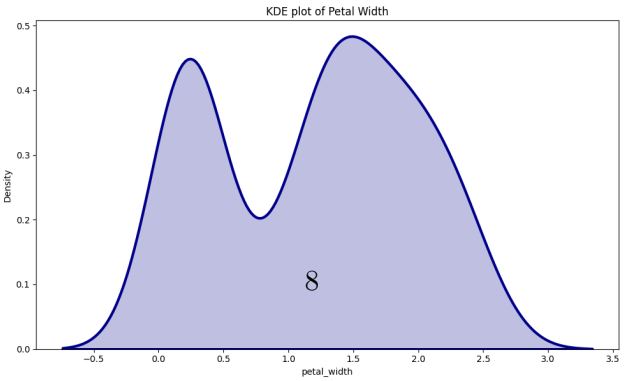
```
# Create a histogram for 'petal_width' with additional kde plot
sns.histplot(data=df, x='petal_width', kde=True, color='darkblue', ax=axs[3])
axs[3].set_title('Histogram of Petal Width with KDE')

# Create a KDE plot for 'petal_width' with a custom color
sns.kdeplot(data=df, x='petal_width', fill=True, color='darkblue', linewidth=3,␣
 ↪ax=axs[4])
axs[4].set_title('KDE plot of Petal Width')

# Automatically adjust subplot params so that the subplot fits into the figure␣
 ↪area
plt.tight_layout()

# Display the figure
plt.show()
```


Histogram of Sepal Length with KDE
Count
sepal_length

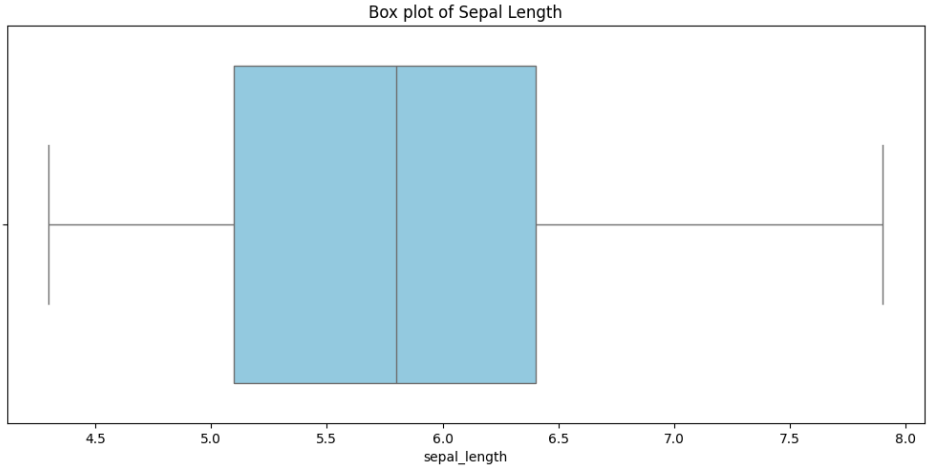

Histogram of Sepal Width with KDE
Count
sepal_width

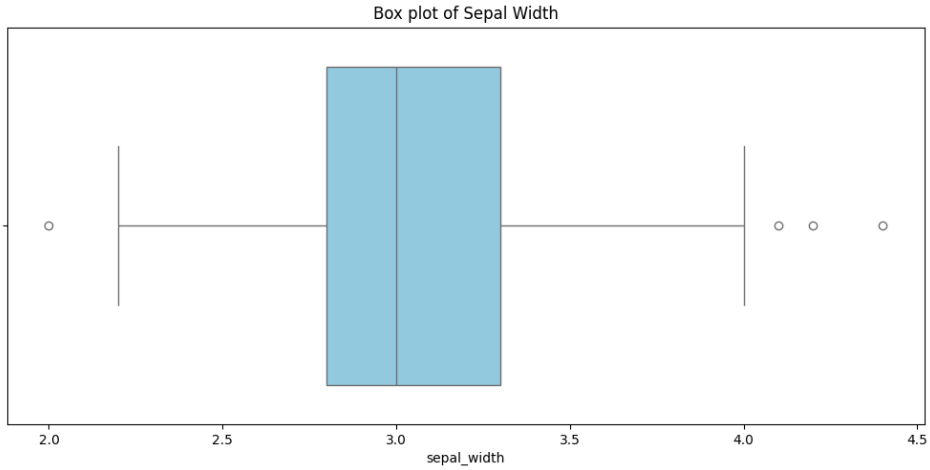

Histogram of Petal Length with KDE
Count
petal_length

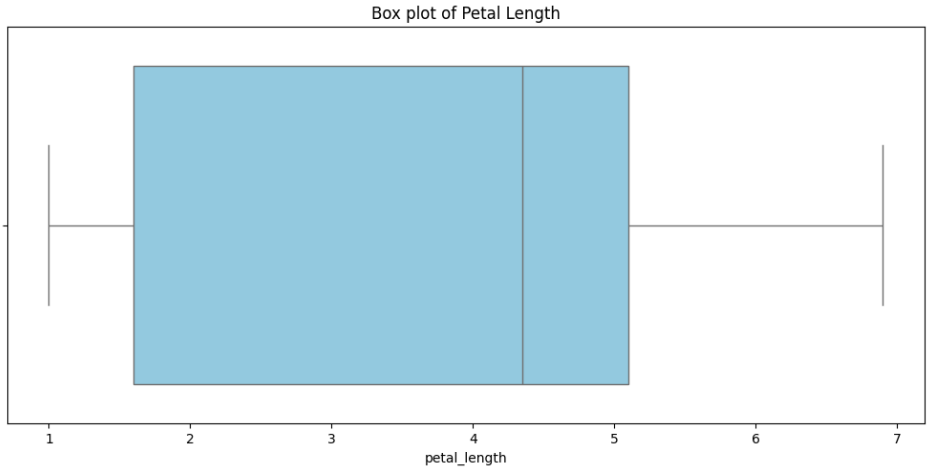

Histogram of Petal Width with KDE
Count
petal_width

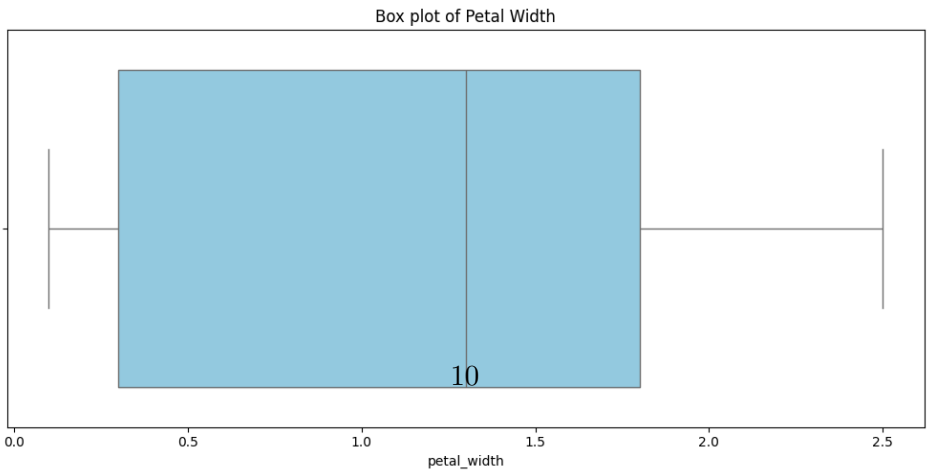

KDE plot of Petal Width
Density
petal_width

[11]:
```
# Create a new figure and a set of subplots
fig, axs = plt.subplots(4, 1, figsize=(10, 20))

# Create a box plot for 'sepal_length' with a custom color
sns.boxplot(x=df['sepal_length'], color='skyblue', ax=axs[0])
axs[0].set_title('Box plot of Sepal Length')

# Create a box plot for 'sepal_width' with a custom color
sns.boxplot(x=df['sepal_width'], color='skyblue', ax=axs[1])
axs[1].set_title('Box plot of Sepal Width')

# Create a box plot for 'petal_length' with a custom color
sns.boxplot(x=df['petal_length'], color='skyblue', ax=axs[2])
axs[2].set_title('Box plot of Petal Length')

# Create a box plot for 'petal_width' with a custom color
sns.boxplot(x=df['petal_width'], color='skyblue', ax=axs[3])
axs[3].set_title('Box plot of Petal Width')

# Automatically adjust subplot params so that the subplot fits into the figure␣
 ↪area
plt.tight_layout()

# Display the figure
plt.show()
```


Box plot of Sepal Length
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
sepal_length


Box plot of Sepal Width
2.0
2.5
3.0
3.5
4.0
4.5
sepal_width


Box plot of Petal Length
1
2
3
4
5
6
7
petal_length


Box plot of Petal Width
0.0
0.5
1.0
1.5
2.0
2.5
petal_width

[ ]: