```
# IRIS FLOWER CLASSIFICATION PROJECT
## PART 3 — Exploratory Data Analysis and Hypothesis Testing
```

```
!pip install seaborn scipy
```

```
Requirement already satisfied: seaborn in /usr/local/lib/python3.12/dist-packages (0.13.2)
Requirement already satisfied: scipy in /usr/local/lib/python3.12/dist-packages (1.16.3)
Requirement already satisfied: numpy!=1.24.0,>=1.20 in /usr/local/lib/python3.12/dist-packages (from seaborn) (2.0.2)
Requirement already satisfied: pandas>=1.2 in /usr/local/lib/python3.12/dist-packages (from seaborn) (2.2.2)
Requirement already satisfied: matplotlib!=3.6.1,>=3.4 in /usr/local/lib/python3.12/dist-packages (from seaborn) (3.10.0)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.12/dist-packages (from matplotlib!=3.6.1,>=3.4->seabor
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.12/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.12/dist-packages (from matplotlib!=3.6.1,>=3.4->seabo
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.12/dist-packages (from matplotlib!=3.6.1,>=3.4->seabo
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.12/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn
Requirement already satisfied: pillow>=8 in /usr/local/lib/python3.12/dist-packages (from matplotlib!=3.6.1,>=3.4->seaborn) (11.
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.12/dist-packages (from matplotlib!=3.6.1,>=3.4->seabor
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.12/dist-packages (from matplotlib!=3.6.1,>=3.4->se
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.12/dist-packages (from pandas>=1.2->seaborn) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.12/dist-packages (from pandas>=1.2->seaborn) (2025.3)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.12/dist-packages (from python-dateutil>=2.7->matplotlib!=3.6.1
```

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

from sklearn.datasets import load_iris
```

```
iris = load_iris()

df = pd.DataFrame(
iris.data,
columns=iris.feature_names
)

df["species"]=iris.target

df["species"]=df["species"].map({
0:"setosa",
1:"versicolor",
2:"virginica"
})

df.head()
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	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

Next steps: Generate code with df New interactive sheet

The Iris dataset is loaded using the machine learning library :contentReference[oaicite:0]{index=0}. The dataset contains four measurement features and three flower species categories.

```
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 (cm)  150 non-null    float64
 1   sepal width (cm)   150 non-null    float64
 2   petal length (cm)  150 non-null    float64
 3   petal width (cm)   150 non-null    float64
```

```
 4   species            150 non-null    object
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

```
df.describe()
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
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

```
df.isnull().sum()
```

	0
sepal length (cm)	0
sepal width (cm)	0
petal length (cm)	0
petal width (cm)	0
species	0

dtype: int64

The dataset contains 150 records with no missing values. All features are numerical measurements suitable for analysis.

```
df.groupby("species").mean()
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
species				
setosa	5.006	3.428	1.462	0.246
versicolor	5.936	2.770	4.260	1.326
virginica	6.588	2.974	5.552	2.026

```
df.groupby("species").median()
```

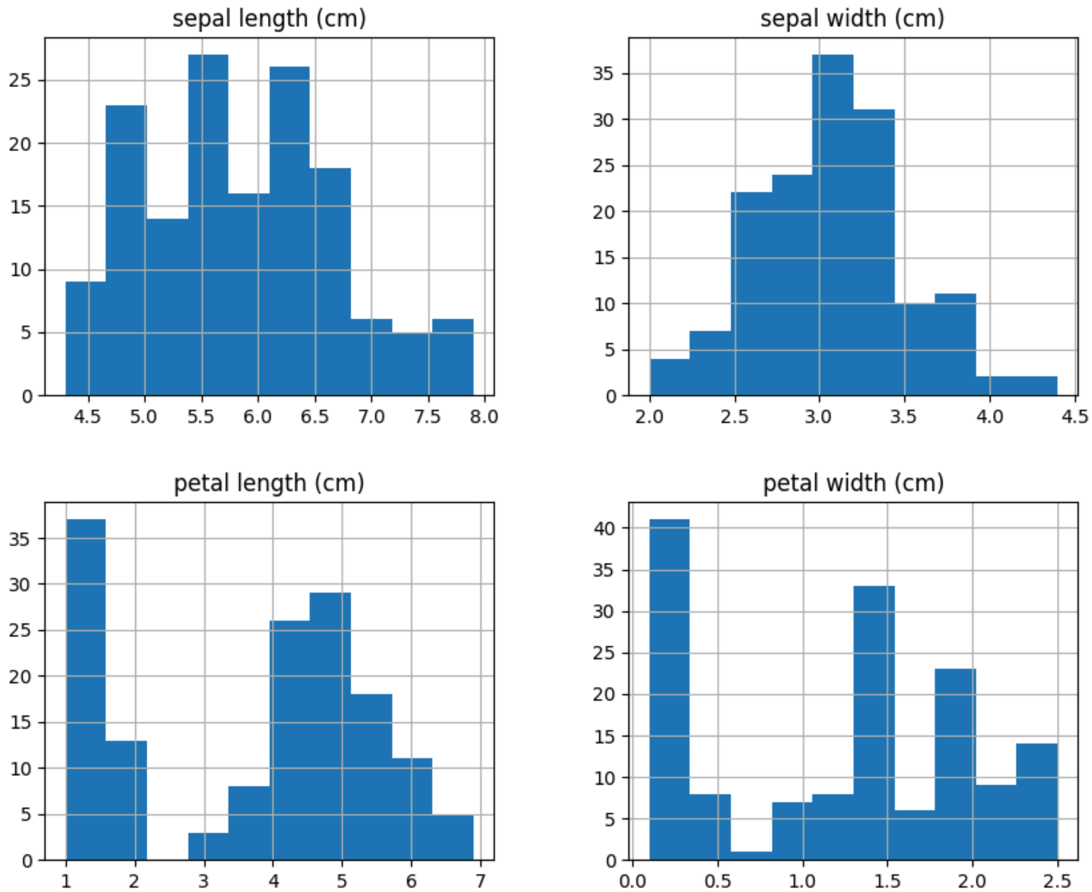
	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
species				
setosa	5.0	3.4	1.50	0.2
versicolor	5.9	2.8	4.35	1.3
virginica	6.5	3.0	5.55	2.0

```
df.groupby("species").std()
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
species				
setosa	0.352490	0.379064	0.173664	0.105386
versicolor	0.516171	0.313798	0.469911	0.197753
virginica	0.635880	0.322497	0.551895	0.274650

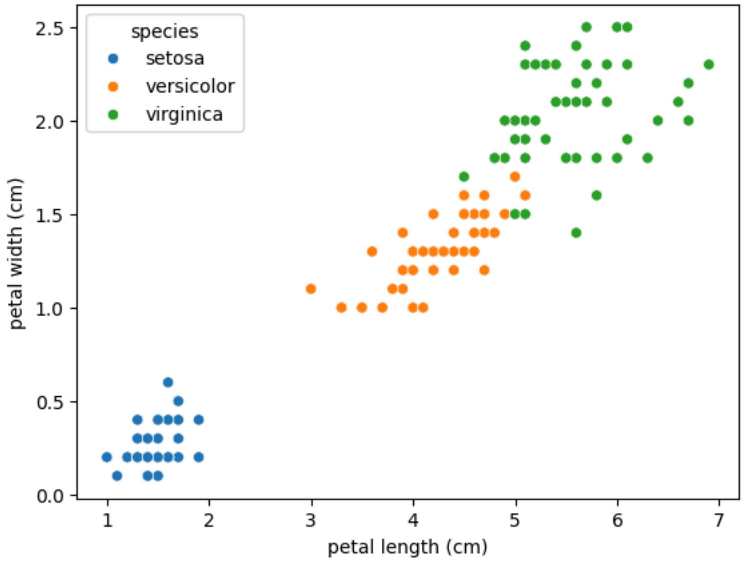
Petal measurements show major differences among species compared to sepal measurements.

```
df.hist(figsize=(10,8))

plt.show()
```



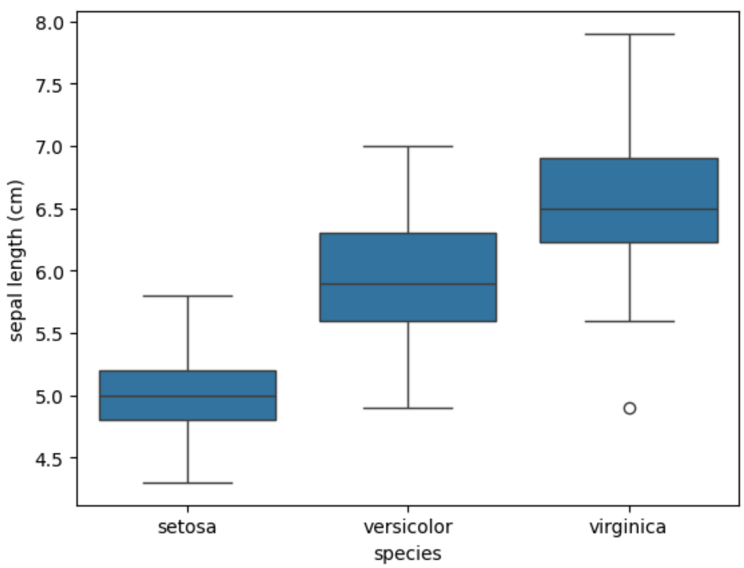
Histograms show distribution of flower measurements. Setosa species has smaller petal sizes compared to others.

```
sns.scatterplot(
data=df,
x="petal length (cm)",
y="petal width (cm)",
hue="species"
)

plt.show()
```



Petal length and petal width provide clear separation among Iris species.

```
sns.boxplot(
x="species",
y="sepal length (cm)",
data=df
)

plt.show()
```



Box plots show variation and median differences among species.

```
from scipy.stats import ttest_ind

setosa=df[df["species"]=="setosa"]

versicolor=df[df["species"]=="versicolor"]

ttest_ind(
setosa["sepal length (cm)"],
versicolor["sepal length (cm)"]
)
```

```
TtestResult(statistic=np.float64(-10.52098626754911), pvalue=np.float64(8.985235037487079e-18), df=np.float64(98.0))
```

Since the p-value is less than 0.05, the null hypothesis is rejected. Sepal length differs significantly between Setosa and Versicolor species.

```
from scipy.stats import f_oneway

virginica=df[df["species"]=="virginica"]

f_oneway(
setosa["petal width (cm)"],
versicolor["petal width (cm)"],
virginica["petal width (cm)"]
)
```

```
F_onewayResult(statistic=np.float64(960.0071468018067), pvalue=np.float64(4.169445839443833e-85))
```

The ANOVA test shows significant differences in petal width among the three species.

Key Insights:

1. Petal measurements strongly distinguish Iris species.
2. Setosa species forms a separate cluster.
3. Versicolor and Virginica partially overlap.
4. Statistical tests confirm significant measurement differences.