Description

The Iris dataset was used in R.A. Fisher's classic 1936 paper, The Use of Multiple Measurements in Taxonomic Problems, and can also be found on the UCI Machine Learning Repository.

It includes three iris species with 50 samples each as well as some properties about each flower. One flower species is linearly separable from the other two, but the other two are not linearly separable from each other.

The columns in this dataset are:

Id

1.SepalLengthCm

2.SepalWidthCm

3.PetalLengthCm

4.PetalWidthCm

5.Species

In [56]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [16]:
```
df = pd.read_csv('iris.csv')
df.head()
```

Out[16]:

	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 [17]:
```
df[df['sepal_width']>4]
```

Out[17]:

	sepal_length	sepal_width	petal_length	petal_width	species
15	5.7	4.4	1.5	0.4	setosa
32	5.2	4.1	1.5	0.1	setosa
33	5.5	4.2	1.4	0.2	setosa

In [18]:
```
df[df['petal_width']>1]
```

Out[18]:

	sepal_length	sepal_width	petal_length	petal_width	species
50	7.0	3.2	4.7	1.4	versicolor
51	6.4	3.2	4.5	1.5	versicolor
52	6.9	3.1	4.9	1.5	versicolor
53	5.5	2.3	4.0	1.3	versicolor
54	6.5	2.8	4.6	1.5	versicolor
...	...	...	...	...	...
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

93 rows × 5 columns

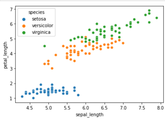
In [19]:
```
df[df['petal_width']>2]
```

Out[19]:

	sepal_length	sepal_width	petal_length	petal_width	species
100	6.3	3.3	6.0	2.5	virginica
102	7.1	3.0	5.9	2.1	virginica
104	6.5	3.0	5.8	2.2	virginica
105	7.6	3.0	6.6	2.1	virginica
109	7.2	3.6	6.1	2.5	virginica
112	6.8	3.0	5.5	2.1	virginica
114	5.8	2.8	5.1	2.4	virginica
115	6.4	3.2	5.3	2.3	virginica
117	7.7	3.8	6.7	2.2	virginica
118	7.7	2.6	6.9	2.3	virginica
120	6.9	3.2	5.7	2.3	virginica
124	6.7	3.3	5.7	2.1	virginica
128	6.4	2.8	5.6	2.1	virginica
132	6.4	2.8	5.6	2.2	virginica
135	7.7	3.0	6.1	2.3	virginica
136	6.3	3.4	5.6	2.4	virginica
139	6.9	3.1	5.4	2.1	virginica
140	6.7	3.1	5.6	2.4	virginica
141	6.9	3.1	5.1	2.3	virginica
143	6.8	3.2	5.9	2.3	virginica
144	6.7	3.3	5.7	2.5	virginica
145	6.7	3.0	5.2	2.3	virginica
148	6.2	3.4	5.4	2.3	virginica

Data visualization

In [20]:
```
sns.scatterplot(x = 'sepal_length',y = 'petal_length',data=df,hue='species')
plt.show()
```



In [21]:
```
x = df[['sepal_width']]
```

In [22]:
```
y = df[['sepal_length']]
```

In [23]:
```
from sklearn.model_selection import train_test_split
```

In [24]:
```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.3)
```

In [25]:
```
x_train.head()
```

Out[25]:

	sepal_width
58	2.9
51	3.2
55	2.8
98	2.5
50	3.2

In [26]:
```
x_test.head()
```

Out[26]:

	sepal_width
16	3.9
80	2.4
43	3.5
97	2.9
37	3.6

In [27]:
```
y_train.head()
```

Out[27]:

	sepal_length
58	6.6
51	6.4
55	5.7
98	5.1
50	7.0

In [28]:
```
y_test.head()
```

Out[28]:

	sepal_length
16	5.4
80	5.5
43	5.0
97	6.2
37	4.9

In [29]:
```
from sklearn.linear_model import LinearRegression
```

In [31]:
```
model = LinearRegression()
model
```

Out[31]:
```
LinearRegression()
```

In [32]:
```
model.fit(x_train,y_train)
```

Out[32]:
```
LinearRegression()
```

In [33]:
```
y_pred=model.predict(x_test)
```

In [38]:
```
y_pred[0:5]
```

Out[38]:
```
array([[5.67100063],
       [5.9810156 ],
       [5.75307129],
       [5.87767727],
       [5.73390362]])
```

In [39]:
```
y_test.head()
```

Out[39]:

	sepal_length
16	5.4
80	5.5
43	5.0
97	6.2
37	4.9

In [40]:
```
from sklearn.metrics import mean_squared_error
```

In [41]:
```
mean_squared_error(y_test,y_pred)
```

Out[41]:
```
0.5114452067067081
```

In [42]:
```
# Model-2
```

In [43]:
```
y = df[['sepal_length']]
```

In [44]:
```
x = df[['sepal_width','petal_length','petal_width']]
```

In [46]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
```

In [48]:
```
x_train.head()
```

Out[48]:

	sepal_width	petal_length	petal_width
39	3.4	1.5	0.2
123	2.7	4.9	1.8
33	4.2	1.4	0.2
21	3.7	1.5	0.4
132	2.8	5.6	2.2

In [49]:
```
y_train.head()
```

Out[49]:

	sepal_length
39	5.1
123	6.3
33	5.5
21	5.1
132	6.4

In [51]:
```
model2=LinearRegression()
model2
```

Out[51]:
```
LinearRegression()
```

In [52]:
```
model2.fit(x_train,y_train)
```

Out[52]:
```
LinearRegression()
```

In [53]:
```
y_pred=model2.predict(x_test)
```

In [54]:
```
mean_squared_error(y_pred,y_test)
```

Out[54]:
```
0.09104824875895582
```

In [ ]:
```

```