Live 2025-10-13a

October 14, 2025

1 (A little) more advanced steps in Python

[1]:
```
import numpy as np
import pandas as pd
```

[2]:
```
from sklearn import datasets
import seaborn as sns
```

[3]:
```
X = sns.load_dataset("iris")
```

[4]:
```
X.head()
```

[4]:

	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

[5]:
```
X.describe()
```

[5]:

	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

[7]:
```
X['species'][50:55]
```

[7]:
```
50    versicolor
51    versicolor
52    versicolor
53    versicolor
```

```
54    versicolor
Name: species, dtype: object
```

[9]:
```
X.species.unique()
```

[9]:
```
array(['setosa', 'versicolor', 'virginica'], dtype=object)
```

[10]:
```
X.petal_length.unique()
```

[10]:
```
array([1.4, 1.3, 1.5, 1.7, 1.6, 1.1, 1.2, 1. , 1.9, 4.7, 4.5, 4.9, 4. ,
       4.6, 3.3, 3.9, 3.5, 4.2, 3.6, 4.4, 4.1, 4.8, 4.3, 5. , 3.8, 3.7,
       5.1, 3. , 6. , 5.9, 5.6, 5.8, 6.6, 6.3, 6.1, 5.3, 5.5, 6.7, 6.9,
       5.7, 6.4, 5.4, 5.2])
```

[11]:
```
X.species == 'versicolor'
```

[11]:
```
0      False
1      False
2      False
3      False
4      False
       …
145    False
146    False
147    False
148    False
149    False
Name: species, Length: 150, dtype: bool
```

[12]:
```
X[X.species == 'versicolor']
```

[12]:

	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
55	5.7	2.8	4.5	1.3	versicolor
56	6.3	3.3	4.7	1.6	versicolor
57	4.9	2.4	3.3	1.0	versicolor
58	6.6	2.9	4.6	1.3	versicolor
59	5.2	2.7	3.9	1.4	versicolor
60	5.0	2.0	3.5	1.0	versicolor
61	5.9	3.0	4.2	1.5	versicolor
62	6.0	2.2	4.0	1.0	versicolor
63	6.1	2.9	4.7	1.4	versicolor
64	5.6	2.9	3.6	1.3	versicolor

65	6.7	3.1	4.4	1.4	versicolor
66	5.6	3.0	4.5	1.5	versicolor
67	5.8	2.7	4.1	1.0	versicolor
68	6.2	2.2	4.5	1.5	versicolor
69	5.6	2.5	3.9	1.1	versicolor
70	5.9	3.2	4.8	1.8	versicolor
71	6.1	2.8	4.0	1.3	versicolor
72	6.3	2.5	4.9	1.5	versicolor
73	6.1	2.8	4.7	1.2	versicolor
74	6.4	2.9	4.3	1.3	versicolor
75	6.6	3.0	4.4	1.4	versicolor
76	6.8	2.8	4.8	1.4	versicolor
77	6.7	3.0	5.0	1.7	versicolor
78	6.0	2.9	4.5	1.5	versicolor
79	5.7	2.6	3.5	1.0	versicolor
80	5.5	2.4	3.8	1.1	versicolor
81	5.5	2.4	3.7	1.0	versicolor
82	5.8	2.7	3.9	1.2	versicolor
83	6.0	2.7	5.1	1.6	versicolor
84	5.4	3.0	4.5	1.5	versicolor
85	6.0	3.4	4.5	1.6	versicolor
86	6.7	3.1	4.7	1.5	versicolor
87	6.3	2.3	4.4	1.3	versicolor
88	5.6	3.0	4.1	1.3	versicolor
89	5.5	2.5	4.0	1.3	versicolor
90	5.5	2.6	4.4	1.2	versicolor
91	6.1	3.0	4.6	1.4	versicolor
92	5.8	2.6	4.0	1.2	versicolor
93	5.0	2.3	3.3	1.0	versicolor
94	5.6	2.7	4.2	1.3	versicolor
95	5.7	3.0	4.2	1.2	versicolor
96	5.7	2.9	4.2	1.3	versicolor
97	6.2	2.9	4.3	1.3	versicolor
98	5.1	2.5	3.0	1.1	versicolor
99	5.7	2.8	4.1	1.3	versicolor

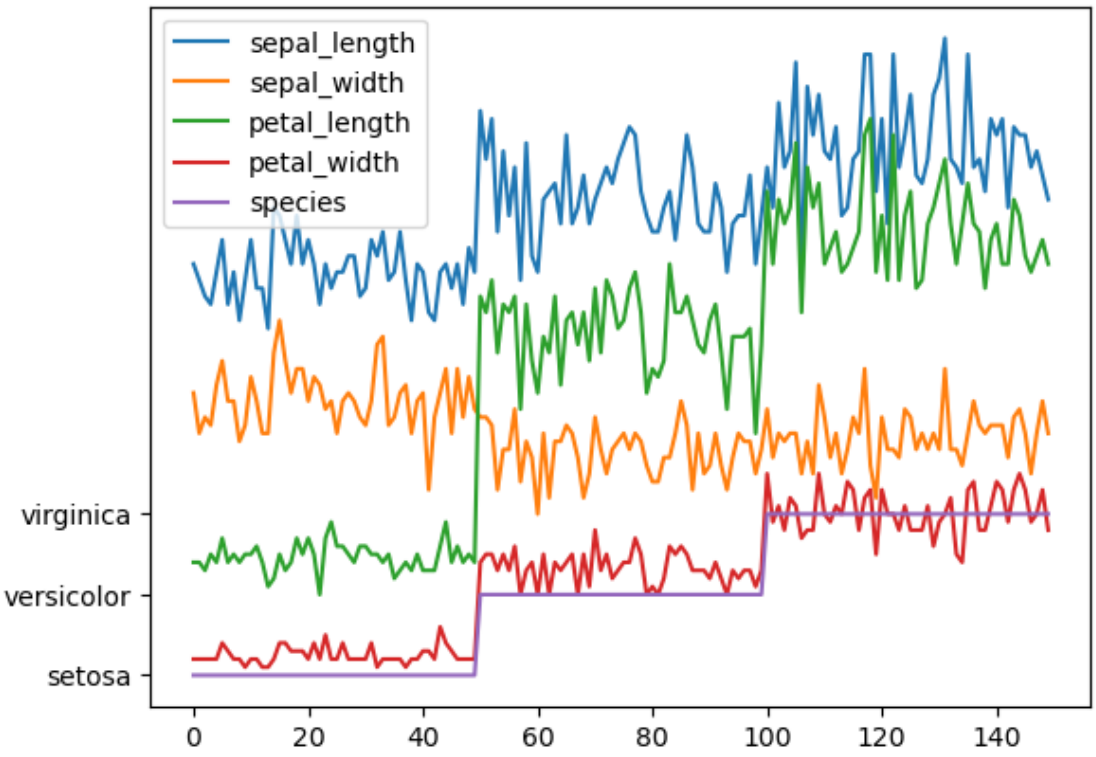
1.1 Now for some plotting

[13]:
```
import matplotlib.pyplot as plt
```

[15]:
```
for f in X.columns:
    plt.plot(X[f],label=f)

plt.legend()
```

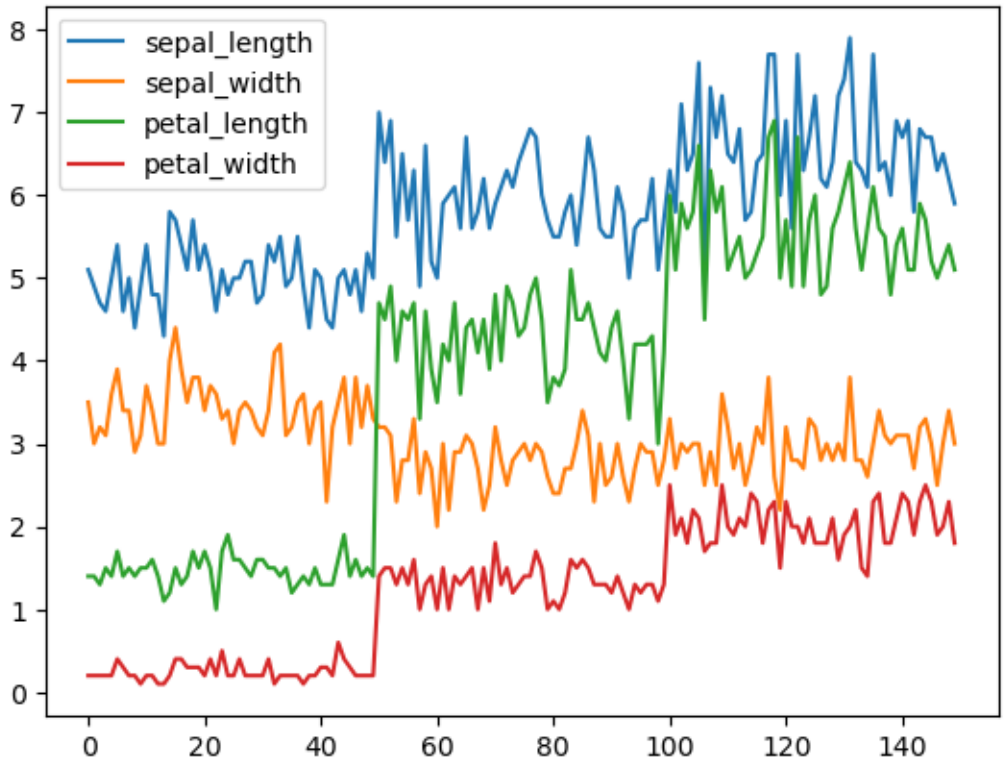
[15]:
```
<matplotlib.legend.Legend at 0x7f7770bf94e0>
```



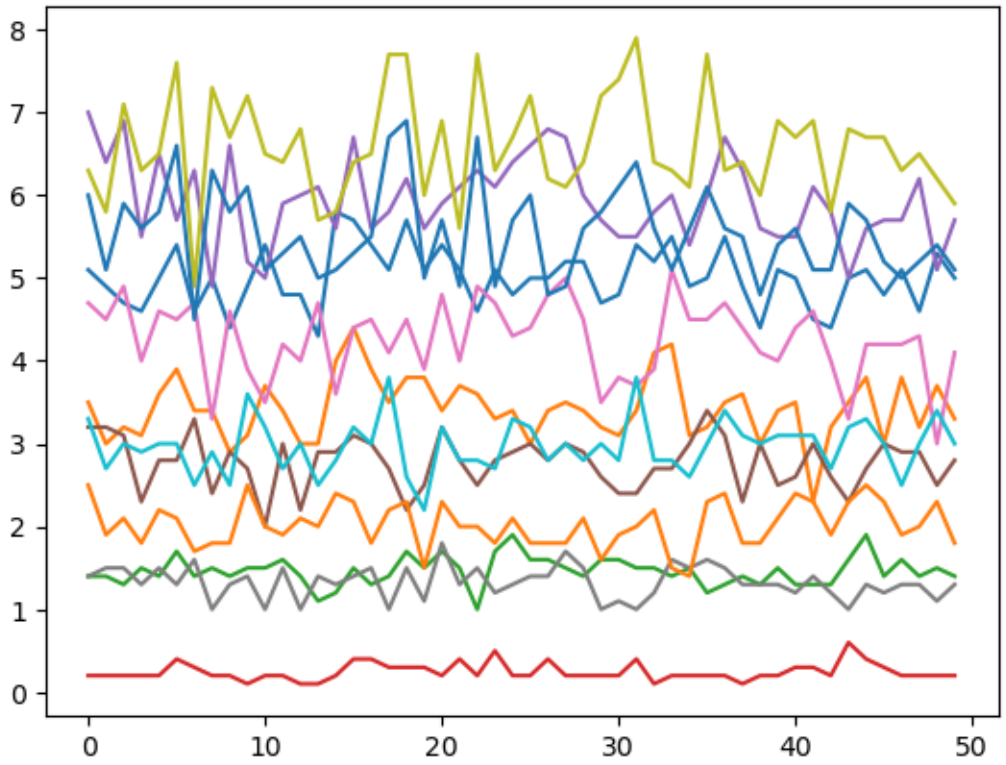
[19]:
```
cols = X.columns
cols = cols.drop("species")
for f in cols:
    plt.plot(X[f],label=f)

plt.legend()
```

[19]: <matplotlib.legend.Legend at 0x7f7770085bd0>



[22]:
```
for sp in X.species.unique():
    xx = X[X.species == sp]
    for f in cols:
        plt.plot(xx.index - xx.index.min(),xx[f],label=f"{sp} - {f}")
#plt.legend()
```


8
7
6
5
4
3
2
1
0
0
10
20
30
40
50

[ ]: