lab01

October 14, 2023

1

1.1

skrlearn.datasets. *sl, sw, pl, pw* - , *target* -

.

[ ]:

```
from sklearn.datasets import load_iris
import numpy  as np
import pandas as pd

iris = load_iris()
iris_pd=pd.DataFrame(data=np.c_[iris['data'], iris['target']],␣
 ↪columns=iris['feature_names'] + ['target'])
iris_pd = iris_pd.rename(columns={'sepal length (cm)': 'sl', 'sepal width (cm)':
 ↪ 'sw', 'petal length (cm)': 'pl', 'petal width (cm)': 'pw'})

print(iris_pd)
print(iris_pd.describe())
```

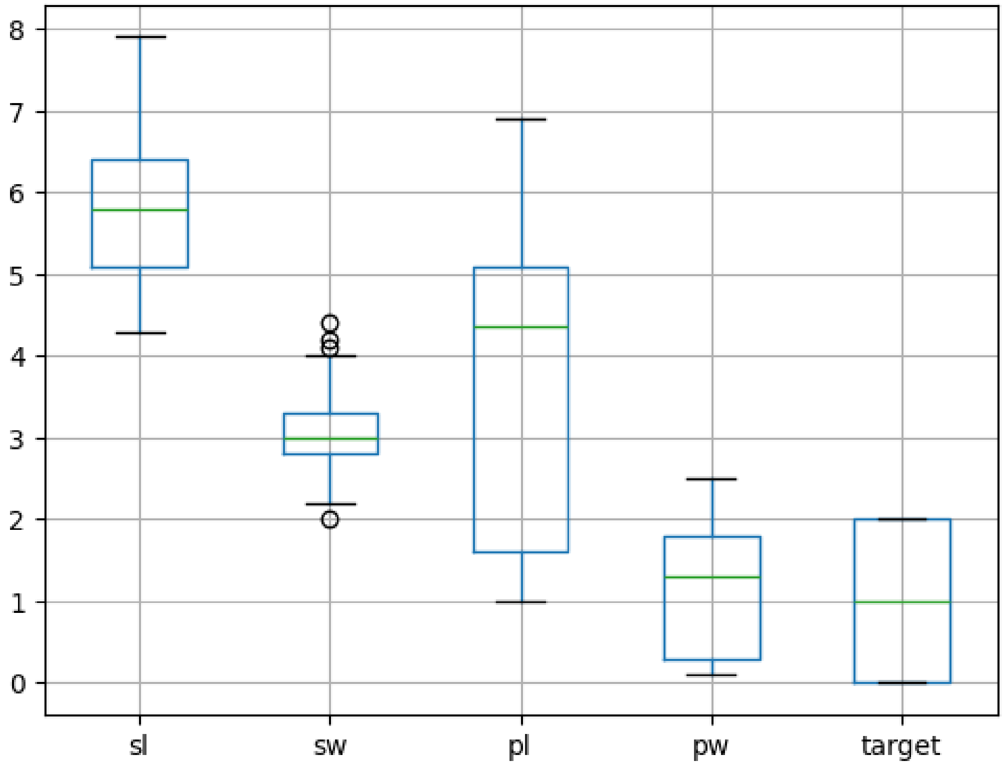
```
      sl   sw   pl   pw  target
0    5.1  3.5  1.4  0.2     0.0
1    4.9  3.0  1.4  0.2     0.0
2    4.7  3.2  1.3  0.2     0.0
3    4.6  3.1  1.5  0.2     0.0
4    5.0  3.6  1.4  0.2     0.0
..   …  …  …  …     …
145  6.7  3.0  5.2  2.3     2.0
146  6.3  2.5  5.0  1.9     2.0
147  6.5  3.0  5.2  2.0     2.0
148  6.2  3.4  5.4  2.3     2.0
149  5.9  3.0  5.1  1.8     2.0

[150 rows x 5 columns]
               sl          sw          pl          pw      target
count  150.000000  150.000000  150.000000  150.000000  150.000000
mean     5.843333    3.057333    3.758000    1.199333    1.000000
std      0.828066    0.435866    1.765298    0.762238    0.819232
min      4.300000    2.000000    1.000000    0.100000    0.000000
```

```
25%       5.100000    2.800000    1.600000    0.300000    0.000000
50%       5.800000    3.000000    4.350000    1.300000    1.000000
75%       6.400000    3.300000    5.100000    1.800000    2.000000
max       7.900000    4.400000    6.900000    2.500000    2.000000
```

2

2.0.1 , .

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

     iris_pd.boxplot()
     plt.show()
```



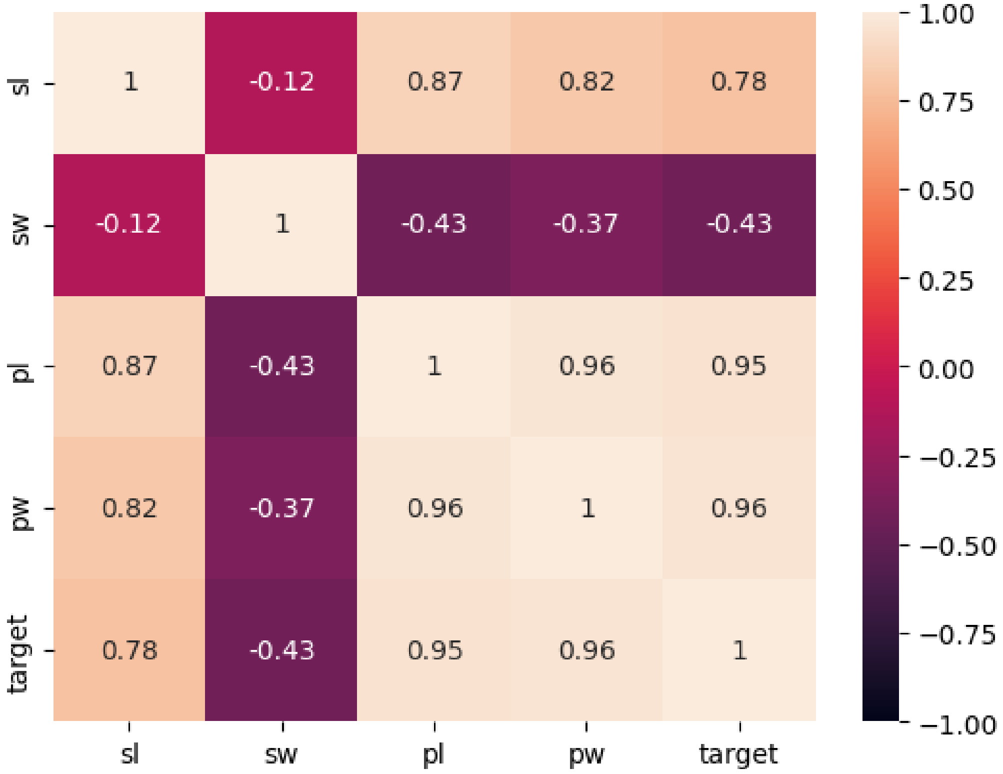
, . ,
sw .

2.0.2 . .

```
[ ]: import seaborn as sns

     corr_matr = iris_pd.corr()
```

```python
print(corr_matr)

sns.heatmap(corr_matr, annot= True, vmin=-1, vmax=1)
plt.show()
```

```
              sl        sw        pl        pw    target
sl      1.000000 -0.117570  0.871754  0.817941  0.782561
sw     -0.117570  1.000000 -0.428440 -0.366126 -0.426658
pl      0.871754 -0.428440  1.000000  0.962865  0.949035
pw      0.817941 -0.366126  0.962865  1.000000  0.956547
target  0.782561 -0.426658  0.949035  0.956547  1.000000
```



, sw. ,
.

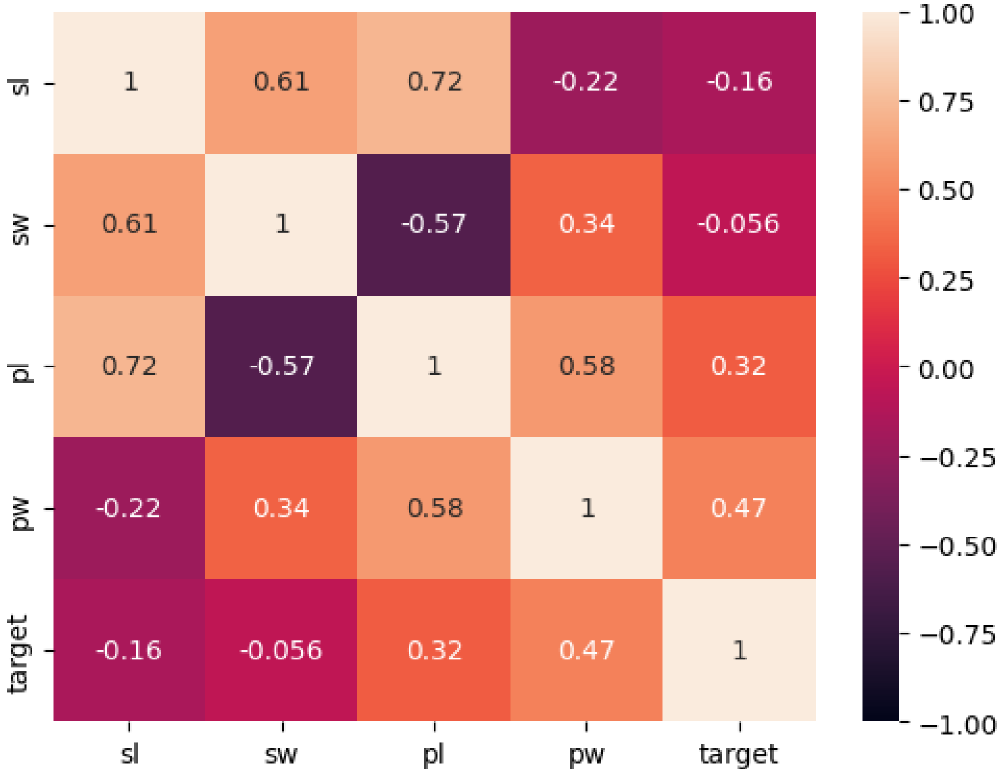
2.0.3 , .
.

[ ]:

```python
import pingouin

pcorr_matr = iris_pd.pcorr()
print(pcorr_matr)
```

```
sns.heatmap(pcorr_matr, annot= True, vmin=-1, vmax=1)
plt.show()
```

```
              sl        sw        pl        pw    target
sl      1.000000  0.610735  0.723756 -0.220409 -0.159156
sw      0.610735  1.000000 -0.565057  0.336649 -0.055676
pl      0.723756 -0.565057  1.000000  0.578470  0.316796
pw     -0.220409  0.336649  0.578470  1.000000  0.472174
target -0.159156 -0.055676  0.316796  0.472174  1.000000
```



,

. , , , *pl* *pw*.

2.0.4 - .

.

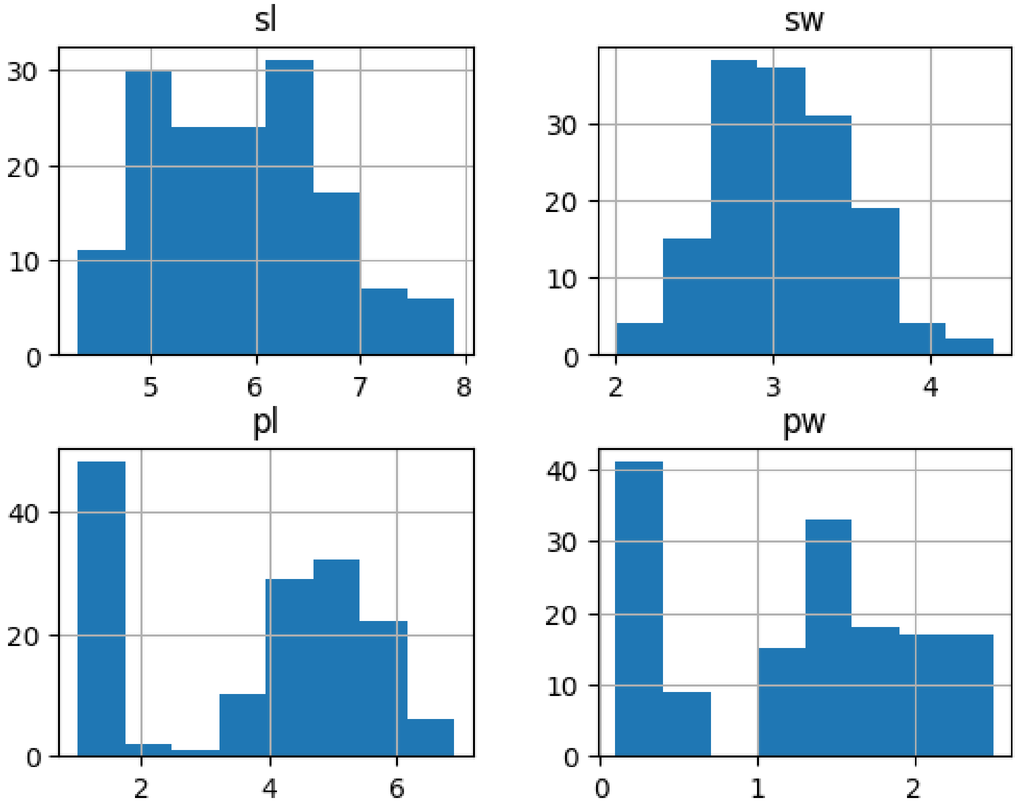
```
[ ]: import scipy.stats

kstest_res = [ scipy.stats.kstest(iris_pd.iloc[:,i], 'norm') for i in␣
 ↪range(len(iris_pd.iloc[0]))]
for elem in kstest_res:
    print(f'           : {elem.statistic}, pvalue: {elem.pvalue}')
```

```
: 0.999991460094529, pvalue: 0.0
: 0.9794298858198347, pvalue: 1.9343513094431716e-253
: 0.8765328487477231, pvalue: 1.4044248603466388e-136
: 0.5459263761057697, pvalue: 1.876499271371568e-42
: 0.5080114127352096, pvalue: 2.0970191237053004e-36
```

- ,
pvalue .

2.0.5 , , , .

```
[ ]: iris_pd.iloc[:,:-1].hist(bins = 8)
     plt.show()
```



. , , - . , *pl* *pw* .

, , , . . , , *pl* *pw*, .

2.1

2.1.1 . (sklearn).

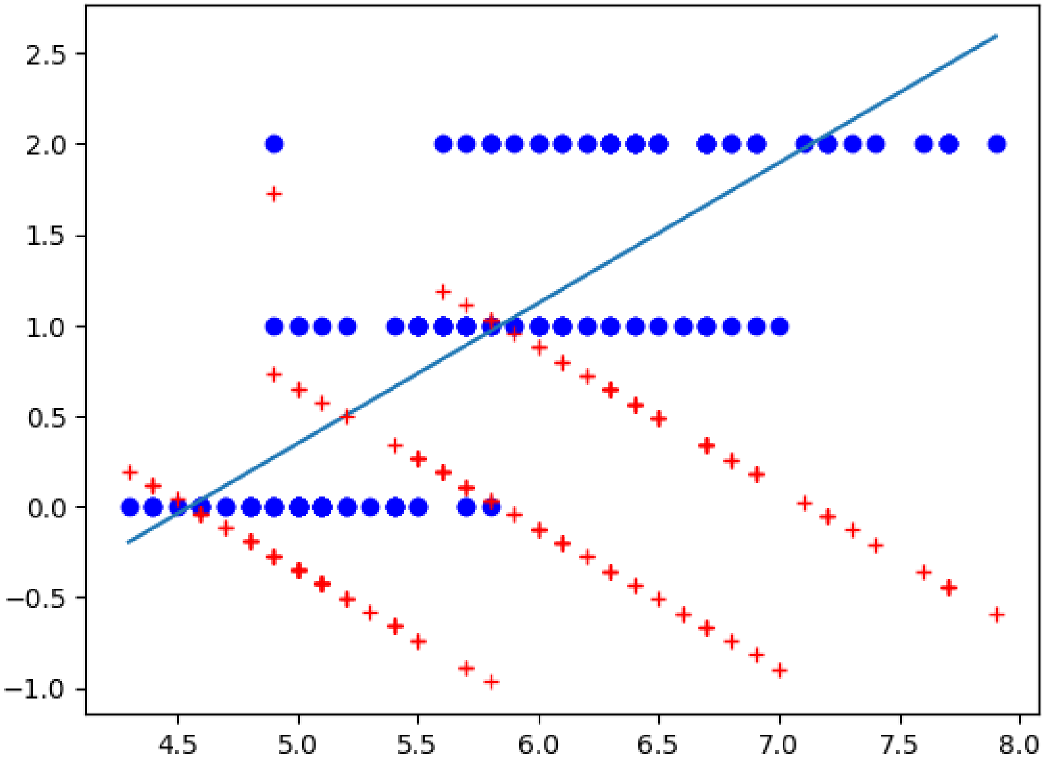
. , ,

sl, *sw*, *pl*, *pw*, .

[ ]:

```
from sklearn import linear_model


'''
x = []
for i in range(len(iris_pd.iloc[0]) - 1):
    x.append(iris_pd.iloc[:,i].tolist())
'''

inp = [ iris_pd.iloc[:,0].tolist()]
output = iris_pd['target'].tolist()

data = [ [inp[j][i] for j in range(len(inp))] for i in range(len(inp[0])) ]

lin_pred = lambda b, x: b[0] + x * b[1]

reg = linear_model.LinearRegression()
reg.fit(data, output)
b = [reg.intercept_, reg.coef_[0]]

x = [min(inp[0]), max(inp[0])]
out_pred = [ lin_pred(b, x[i]) for i in range(len(x))]
residues = [ output[i] - lin_pred(b, inp[0][i])  for i in range(len(output))]

print(out_pred)
print(reg.coef_)

plt.plot(inp[0], output, 'bo')
plt.plot(x, out_pred)
plt.plot(inp[0], residues, 'r+')
plt.show()
```

```
[-0.19486794668928775, 2.592297026149656]
[0.77421249]
```



2.1.2 . ( ),
( ), , ,
. ( ).