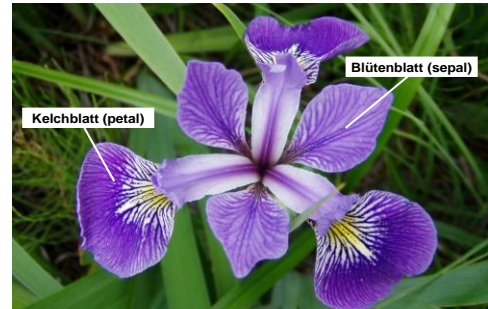


FISHER's Iris Datensatz

FISHER's Iris Datensatz ist ein experimenteller Datensatz, der Messwerte für $N = 4$ Eigenschaften von insgesamt $M = 150$ Schwertlilien (engl. Iris) enthält und zuerst im Jahr 1936 von dem Biologen Ronald FISHER veröffentlicht wurde.

Die untersuchten Eigenschaften sind:

- x_1 – Länge der Kelchblätter (engl. sepal length)
- x_2 – Breite der Kelchblätter (engl. sepal width)
- x_3 – Länge der Blütenblätter (engl. petal length)
- x_4 – Breite der Blütenblätter (engl. petal width)



Wir haben also ein Set bestehend aus $M = 150$ Datenpunkten:

$$\{\underline{x}^{(1)}, \dots, \underline{x}^{(150)}\}$$

Jeder Datenpunkt besteht aus einem Vektor mit den $N = 4$ Eigenschaften:

$$\underline{x}^{(i)} = [x_1^{(i)}, x_2^{(i)}, x_3^{(i)}, x_4^{(i)}]$$

Wir können alle Datenpunkte in der Matrix \underline{X} zusammenfassen:

$$\underline{X} = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & x_3^{(1)} & x_4^{(1)} \\ \vdots & \vdots & \vdots & \vdots \\ x_1^{(150)} & \dots & \dots & x_4^{(150)} \end{bmatrix}$$

Die untersuchten Schwertlilien lassen sich jeweils einer von drei Arten zuordnen. Wir können jedem Datenpunkt also eine von drei Klassen $y \in \{1, 2, 3\}$ zuordnen:

- $y = 1$ – *Iris setosa*
- $y = 2$ – *Iris virginica*
- $y = 3$ – *Iris versicolor*

Maschinelles Lernen in der Verfahrenstechnik

Grundlegende statistische Kenngrößen des Datensatzes

```
import pandas as pd #import pandas
from sklearn import datasets #import datasets from scikitlearn
iris = datasets.load_iris() #Load data set
pd.DataFrame(iris.data, columns=iris.feature_names).describe()
```

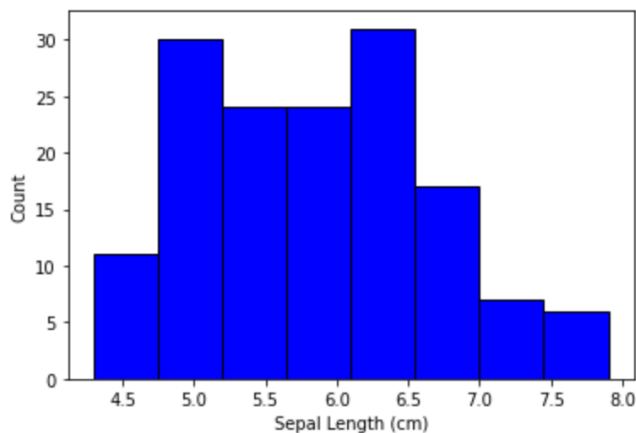
	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333
std	0.828066	0.435866	1.765298	0.762238
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

Histogramm

```
import matplotlib.pyplot as plt # import matplotlib
import numpy as np # import numpy

X = iris.data # Extract features

# Plot histogram of first feature set bin width according to
# Freedman Diaconis Estimator ('fd')
plt.hist(X[:,0], bins='fd', facecolor='b', edgecolor="black")
plt.xlabel('Sepal Length (cm)') # label x axis
plt.ylabel('Count') # label y axis
```



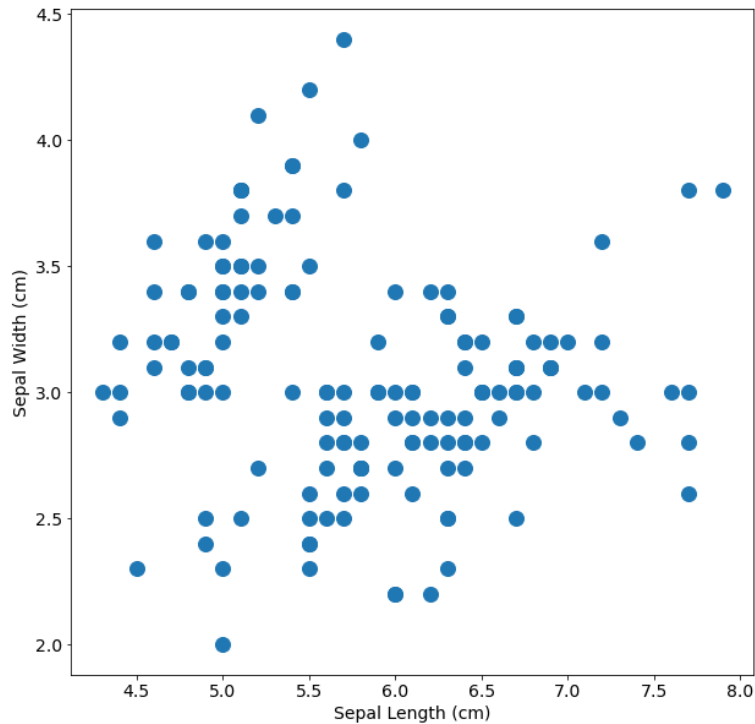
Freedman Diaconis Estimator: $\text{Binbreite} = 2 \frac{IQR}{\sqrt[3]{M}}$, wobei IQR den Interquartilabstand und M die Anzahl an Datenpunkten beschreibt.

Maschinelles Lernen in der Verfahrenstechnik

Scatterplot

Scatterplots stellen den Zusammenhang zwischen jeweils zwei Eigenschaften dar.

```
plt.scatter(X[:,0],X[:,1])           # Plot second over first feature
plt.xlabel('Sepal Length (cm)')      # Label x axis
plt.ylabel('Sepal Width (cm)')       # Label y axis
```



Maschinelles Lernen in der Verfahrenstechnik

Scattermatrix

Eine Scattermatrix stellt die Zusammenhänge zwischen allen möglichen Paaren von Eigenschaften dar.

```
# Import the required python library
import seaborn as sns                                # Import seaborn library

Y = iris.target                                      # Assign targets
df=pd.DataFrame(X,columns=Feature_Names)             # Define dataframe
df["Classes"]=Y+1

# Create pairplot and indicate class labels by different colors
sns.pairplot(df,hue="Classes", palette=['blue','orange','green'],diag_kind="hist")
plt.show()
```

