

Einführung in Machine Learning

Jan Popko

Python Advanced

Machine Learning

- ist eine Sammlung von mathematischen Methoden der Mustererkennung
- erkennen von Mustern in Datenbeständen

Probleme:

- wenn zu wenig Daten vorhanden sind
- wenn zu viele Dimensionen vorhanden sind
 - Entscheidungsbäume werden zu komplex
 - Vektor basierte Algorithmen werden zu rechenaufwendig

Machine Learning

In Python hilft die sklearn library mit vielen Nützlichen Funktionen.

Installieren: `pip install sklearn`

Allgemein nützliches:

train_test_split(X, y) Funktion:

Import: `from sklearn.model_selection import train_test_split`

Funktion: Spaltet die Daten in Trainings- und Testdatensätze auf

metrics Modul:

Import: `from sklearn import metrics`

Funktion: zum Berechnen verschiedenster Metriken für die Auswertung des Models

<https://scikit-learn.org/stable/modules/classes.html#sklearn-metrics-metrics>

Überwachtes Lernen

Jan Popko

Python Advanced

Klassifikation via K-Nearest Neighbour

Durch das Betrachten der Klassen der k-nächsten Nachbarn soll auf die eigene zurückgeschlossen werden.

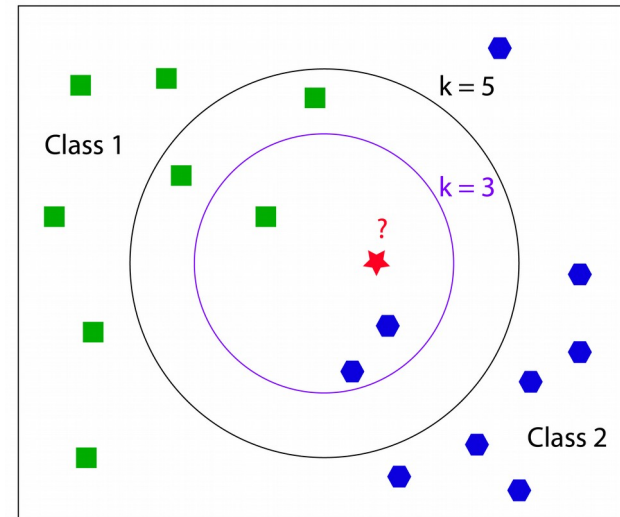
Berechnung der Entfernungen über den euklidischen Abstand:

$$\sqrt{x_1^2 + x_2^2 + \dots + x_n^2}$$

x_i = Dimension/Attribut

Die am häufigsten vorkommende Klasse der Nachbarn wird für sich selbst gewählt.

In SKLearn kann der Algorithmus mit Hilfe des KNeighborsClassifier modelliert werden.



Jan Popko

Python Advanced

K-Nearest Neighbour

```
from sklearn.neighbors import KNeighborsClassifier  
  
knn = KNeighborsClassifier(n_neighbors=5)  
  
knn.fit(data, target)  
  
predicted_target = knn.predict(test_data)
```

Lineare Regression

- modelliert Beziehung einer abhängigen und mehreren unabhängigen Variablen
- Daten zusammenführen und bereinigen
- Datenkorrelation betrachten
- Vorbereitung Regressionsanalyse
 - Spalten müssen intervall- oder ratioskaliert sein
- Lineare Regression mit Scikit-Learn
 - Methode der kleinsten Fehlerquadrate
- bestimmen der MSE (Mittleren Quadratischen Abweichung)
 - (je kleiner, desto besser)
- Bestimmtheitsmaß R^2
 - (zwischen 0 und 1 – je näher an 1, desto besser)

Jan Popko

Python Advanced

Lineare Regression

```
from sklearn.linear_model import LinearRegression
```

```
lr = LinearRegression()  
lr.fit(data, target)
```

```
print(lr.predict(test_data))
```

```
Intercept: lr.intercept_[0]  
Koeffizienten: lr.coef_[0]  
R2 Score: lr.score(X, y)
```

Jan Popko

Python Advanced

Polinomiale Regression

```
from sklearn.linear_model import LinearRegression  
from sklearn.preprocessing import PolynomialFeatures
```

```
quad = PolynomialFeatures(degree=2)
```

```
data_quad = quadratic.fit_transform(data)
```

```
lr = LinearRegression()  
lr.fit(data_quad, target)
```

```
print(lr.predict(test_data_quad))
```

```
Intercept: lr.intercept_[0]  
Koeffizienten: lr.coef_[0]  
R2 Score: lr.score(X, y)
```

Jan Popko

Python Advanced

Support Vector Machines

Stützvektoren:

- meist zur binären Klassifizierung verwendet
- Grenze mittels der nächst liegenden Objekte definiert

Ebenen:

- im n dimensionalen Raum hat die Ebene $n-1$ Dimensionen
- meist sind Cluster nicht linear Trennbar → Hyperebene

Darstellung der Vektorräume in Matplotlib mit Hilfe eines Meshgrids

Support Vector Machines

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.datasets import load_iris
from sklearn.svm import SVC

data = load_iris()
# print(data.DESCR)
features = np.array(data.data)
labels = np.array(data.target)

xx, yy = np.meshgrid(np.arange(features[:, 0].min()-1,
                                features[:, 0].max()+1, 0.05),
                    np.arange(features[:, 1].min()-1,
                                features[:, 1].max()+1, 0.01))

svm = SVC()
svm.fit(features[:, :2], labels)
Z = svm.predict(np.c_[xx.ravel(), yy.ravel()])

fig = plt.figure()
ax = fig.add_subplot(111)
Z = Z.reshape(xx.shape)
ax.contourf(xx, yy, Z, cmap=plt.cm.summer)
ax.scatter(features[:, 0],
            features[:, 1],
            c=labels,
            cmap=plt.cm.summer,
            edgecolors='black')

plt.show()
```

Jan Popko

Python Advanced