


KI - Übersicht über Aufgaben und Methoden

Karl Lockhoff

Chemnitzer Linuxtage 2019



Über mich

- Softwareentwickler TDM/PLM
 - Interessen: *BSD
 - Hobbys: Wandern, Fotographieren, Musik
- 

Agenda

KI – Übersicht über Aufgaben und Methoden

- Suchen
- Klassifizierung
- Wissensbasis
- Regression
- Optimierung

Was ist künstliche Intelligenz?

- Artificial Intelligence is the science of making machines do things that would require intelligence if done by men.

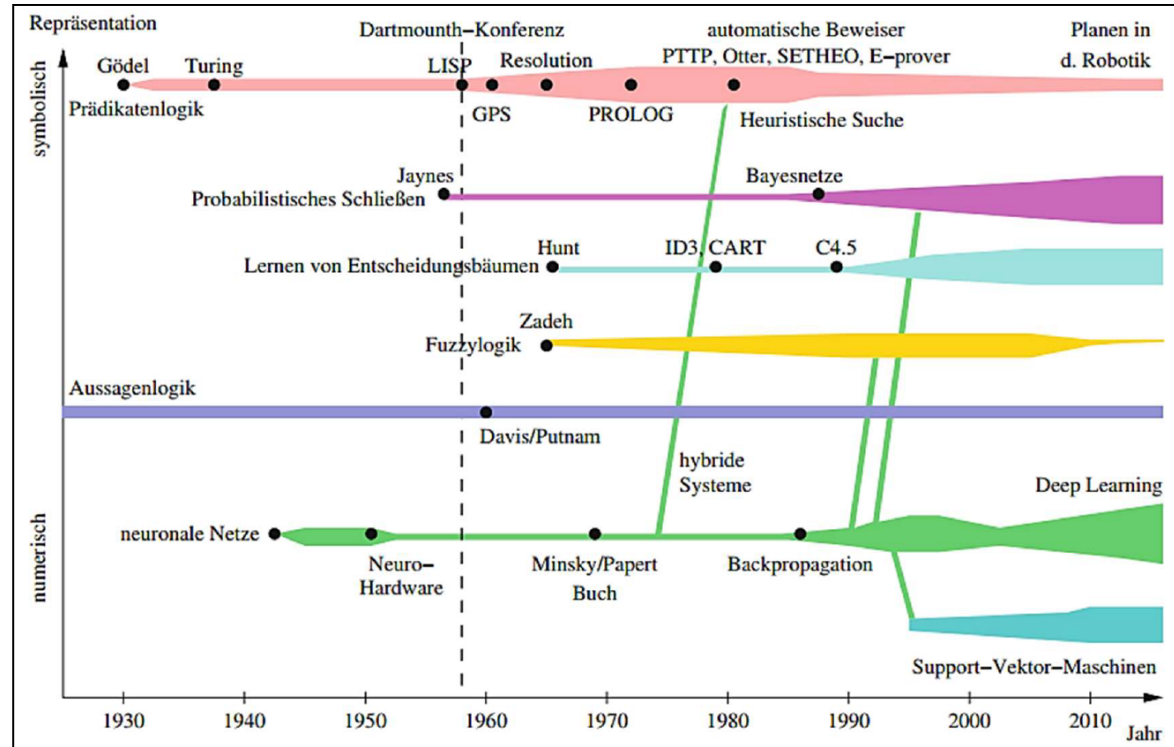
Marvin Minsky

Der kluge Hans



Quelle: Wikipedia

KI - Zeitstrahl



Quelle: Ertel, W. (2013), S. 8.

Suchen

Suchen - Aufgabe

Den besten, kürzesten oder günstigsten Weg in einem gerichteten Graphen suchen

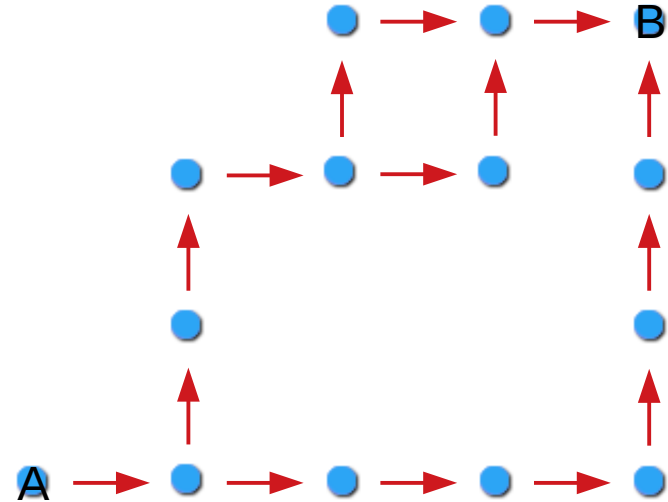
- Puzzles: N-Damen-Problem, Flußüberquerung
- Spiele: Tic-Tac-Toe, Schach, Backgammon
- Handelsreisende / Navigationssysteme

Suchen - Lösungen

- Tiefensuche (deep search)
- Breitensuche
- A*-Algorithmus
- Simuliertes Erstarren (simulated annealing)
- Alpha-Beta Suche
- Evolutionäre (genetische) Algorithmen
- ...

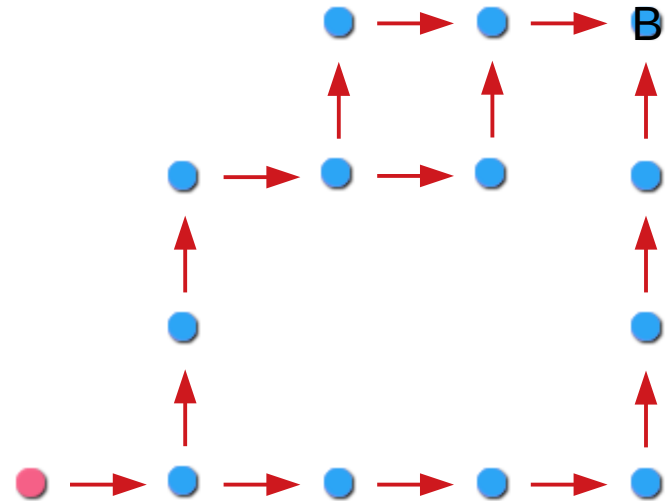
Suchen – Beispiel A*-Algorithmus

- Suche des kürzesten Weges von A nach B auf dem Stadtplan.
- Die Knoten des Graphen sind die Kreuzungen.
- Die Straßen sind die Kanten.
- Die Kosten entsprechen der Weglänge.
- Der Weg zwischen zwei Knoten sei 1
- Als Schätzfunktion kann die Rechtecknorm $|y_2 - y_1| + |x_2 - x_1|$



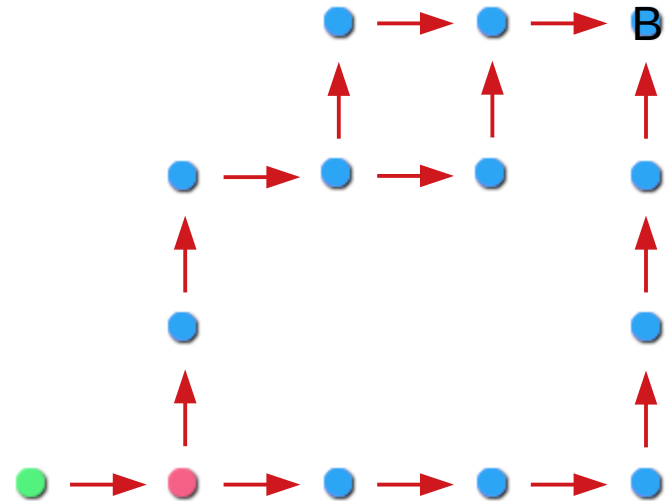
Suchen – Beispiel A*-Algorithmus

- Starte im Startpunkt A
- **Kosten bisher: 0**
- Wiederhole
 - Wähle neuen Knoten mit minimalen geschätzten Kosten
 - Bestimme alle Nachfolger
 - Bestimme die Nachfolger mit den geringsten bisherigen Kosten



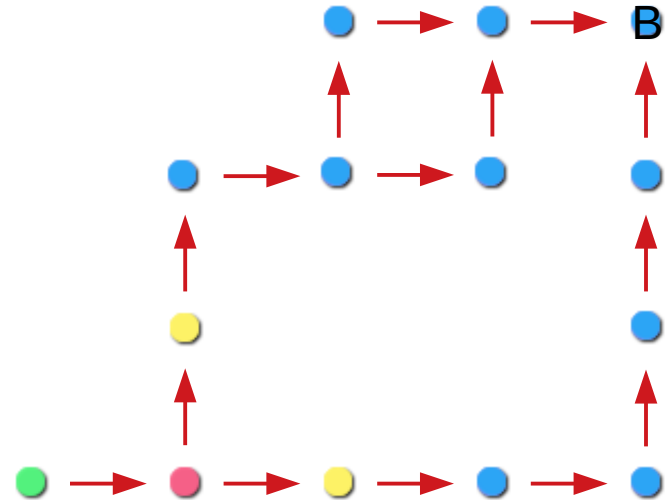
Suchen – Beispiel A*-Algorithmus

- Starte im Startpunkt A
- Kosten bisher: 0
- Wiederhole
 - **Wähle neuen Knoten mit minimalen geschätzten Kosten**
 - Bestimme alle Nachfolger
 - Bestimme die Nachfolger mit den geringsten bisherigen Kosten



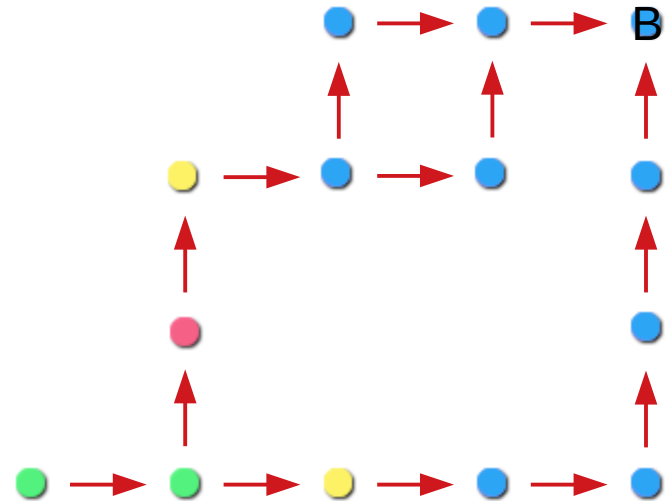
Suchen – Beispiel A*-Algorithmus

- Kosten bisher: 1
- Wiederhole
 - Wähle neuen Knoten mit minimalen geschätzten Kosten
 - **Bestimme alle Nachfolger**
 - **Bestimme die Nachfolger mit den geringsten bisherigen Kosten**



Suchen – Beispiel A*-Algorithmus

- Kosten bisher: 2
- Wiederhole
 - Wähle neuen Knoten mit minimalen geschätzten Kosten
 - Bestimme alle Nachfolger
 - Bestimme die Nachfolger mit den geringsten bisherigen Kosten



Klassifikation

Klassifikation - Aufgabe

Die Aufgabe ist es, Objekte auf Grund der Eigenschaften zu klassifizieren (oder ein zukünftiges Ereignis vorherzusagen)

- Zu welcher Kundengruppe gehört eine Person?
- Diese Nachricht ist für uns wichtig oder unwichtig?
- Texterkennung, Objekterkennung
- Textmining
- Mustererkennung

- Neue Klassifizierungen finden

Klassifikation - Lösungen

Objekte nach bekannten Klassifikationen einordnen

- Neuronale Netze
 - Tensorflow, Torch, Scikit-learn, ...
- Entscheidungsbäume
 - GNU R, Scikit-learn
- Ensemble Verfahren
 - Random Forest
- In einigen Anwendungsfälle Expertensysteme

Klassifikationen erstellen/finden (Clustering)

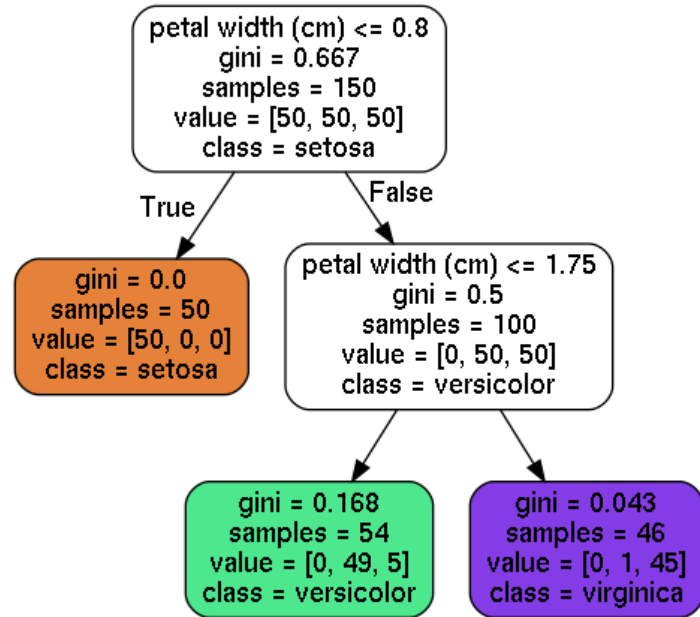
- k-Means
 - OpenCV, Scikit-image, ELKI, GNU R
- Supportvektormaschine (Stützvektor)
 - Libsvm, SVMlight, Scikit-learn, Matlab, GNU R

Klassifikation – Beispiel Entscheidungsbaum

```
from sklearn.datasets import load_iris
from sklearn.tree import
DecisionTreeClassifier
from sklearn.tree import export_graphviz
iris = load_iris()
```

```
X = iris.data[:, 2:]
y = iris.target
```

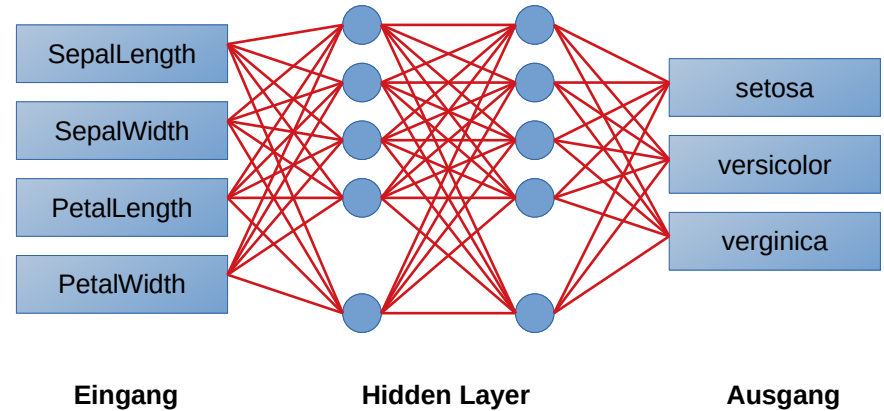
```
tree_clf =
DecisionTreeClassifier(max_depth=2)
tree_clf.fit(X,y)
```



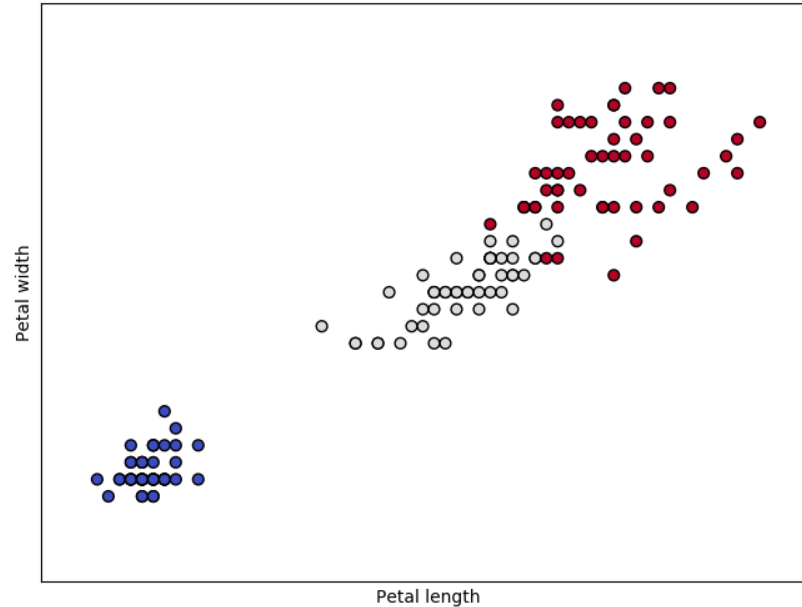
Klassifikation – Beispiel Neuronales Netz

...

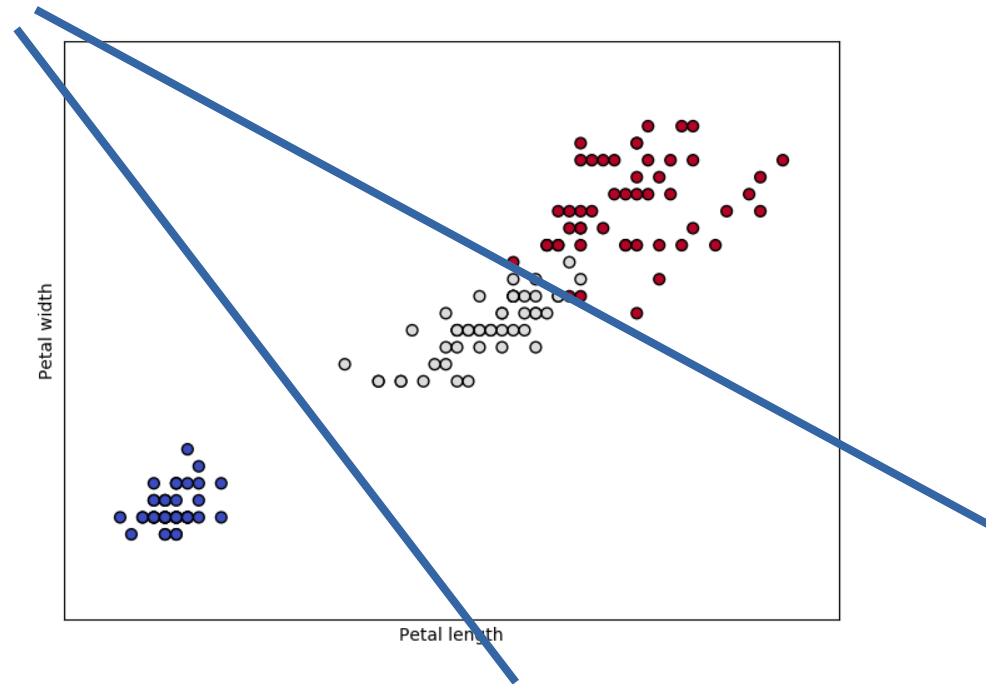
```
classifier = tf.estimator.DNNClassifier(  
    feature_columns=my_feature_columns,  
    hidden_units=[10, 10],  
    n_classes=3)  
  
classifier.train(  
    input_fn=lambda:iris_data.train_input_fn(train_x,  
    train_y, args.batch_size),  
    steps=args.train_steps)
```



Klassifikation – Beispiel SVM



Klassifikation – Beispiel SVM



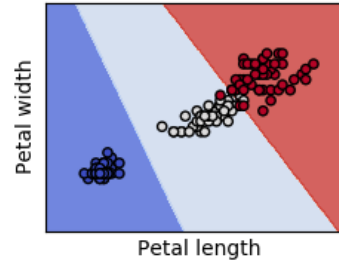
Klassifikation – Beispiel SVM

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

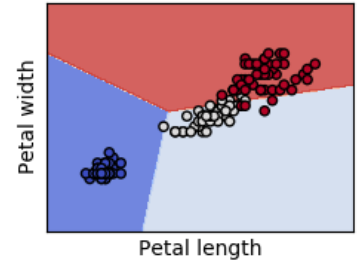
```
iris = datasets.load_iris()
X = iris.data[:, 2:]
y = iris.target
```

```
C = 1.0 # SVM regularization parameter
models = (svm.SVC(kernel='linear', C=C),
          svm.LinearSVC(C=C),
          svm.SVC(kernel='rbf', gamma=0.7,
                  C=C),
          svm.SVC(kernel='poly', degree=3,
                  C=C))
models = (clf.fit(X, y) for clf in models)
```

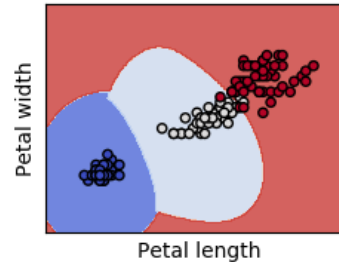
SVC with linear kernel



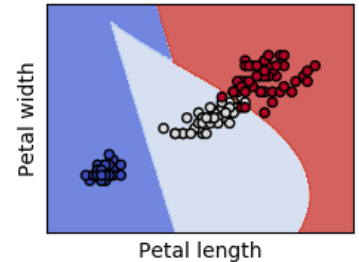
LinearSVC (linear kernel)



SVC with RBF kernel



SVC with polynomial (degree 3) kernel



Wissensbasis

KI – Übersicht über Aufgaben und Methoden

Wissensbasis - Aufgabe

Erstellung der Wissenspräsentation um diese für automatisiertes logisches Schließen zu nutzen.

- Auto springt nicht an, was nun?
- Welche Pflanze/Tier ist das?
- Exploration von Ölquellen (PROSPEKTOR)
- Anwendungen wie digitale Assistenten
- Maschinelles Übersetzen
- Zusammenstellen von Rechnersysteme (XCON für Vax von Dec)

- Darstellung von Wissen

- Induktion (Finden von Regeln an Hand von Beispielen)

Wissensbasis - Lösungen

- Expertensystem / Interferenzmaschine
 - Vorwärtsverkettete Systeme
 - Rückwärtsverkettete Systeme
 - Ops5, clips
- Semantische Netze/Semantische Systeme
 - Cyc, Watson, ...
 - Wolfram Alpha
 - OWL (Semantische Beschreibungssprache für das Web)
- Induktion
 - Symbolische Systeme
 - Subsymbolische Systeme (Neuronale Netze)

Wissensbasis – Beispiel CLIPS

```
(deffacts Wahrheiten
  (ist Mensch sterblich)
  (ist Sokrates Mensch)
  (ist Hund sterblich)
)
```

```
1- (ist Mensch sterblich)
2- (ist Sokrates Mensch)
3- (ist Hund sterblich)
```

```
(defrule translation
  (ist ?a ?b)
  (ist ?b ?c)
  =>
  (assert (ist ?a ?c))
)
```

```
1- (ist Mensch sterblich)
2- (ist Sokrates Mensch)
3- (ist Hund sterblich)
4- (ist sokrates sterblich)
```

Regression

Regression – Probleme

Ziel ist es Beziehungen zwischen einer abhängiger und mehreren unabhängigen Variablen zu modellieren.

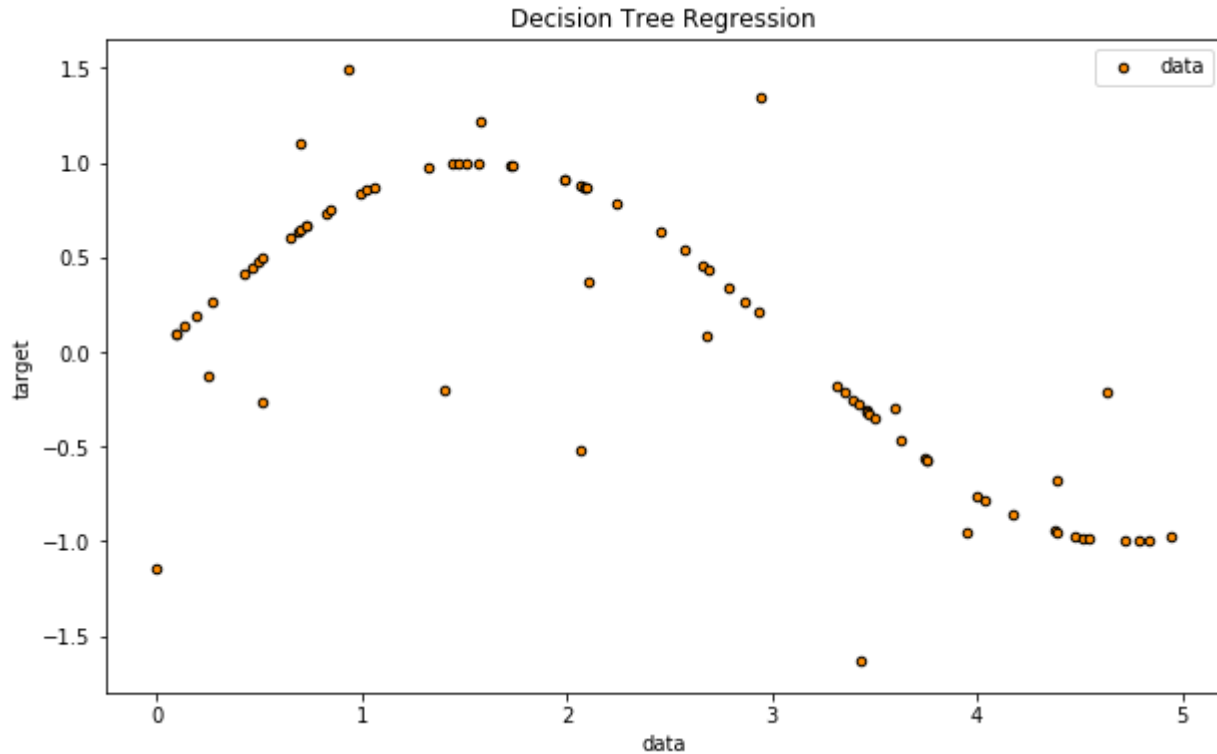
- Steuerungs- und Messtechnik
 - Trennung von Signal und Rauschen
- Analyse des Kaufverhaltens
 - Beziehungen zwischen Kaufkraft und Umsatz
 - Beziehungen zwischen Einkommen, Kaufpreis und Lage der Immobilie

Regression - Lösungen

- Neuronale Netzwerke
- Entscheidungsbäume
- Random Forest
- Andere Klassifikationsverfahren

- Andere mathematische Methoden stehen in Konkurrenz

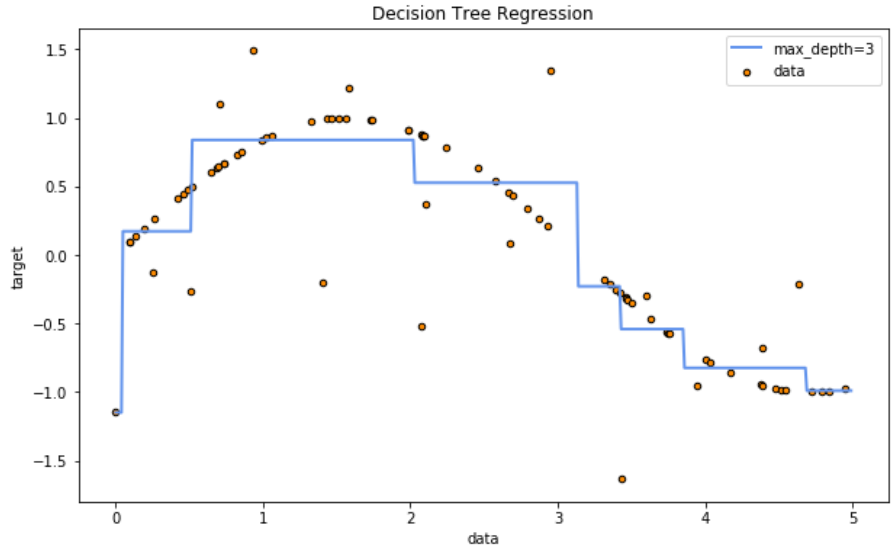
Regression - Beispiel



Regression - Beispiel

```
import numpy as np
from sklearn.tree import
DecisionTreeRegressor
import matplotlib.pyplot as plt

rng = np.random.RandomState(1)
X = np.sort(5 * rng.rand(80, 1), axis=0)
y = np.sin(X).ravel()
y[::5] += 3 * (0.5 - rng.rand(16))
regr = DecisionTreeRegressor(max_depth=3)
regr.fit(X, y)
```



Optimierung

KI – Übersicht über Aufgaben und Methoden

Optimierung – Aufgabe

Bestpassende Parameter zu einem komplexen System zu finden, so das eine Zielfunktion minimiert oder maximiert wird.

- KI, z.B: SVM
- Handelsreisender / Navigationsystem

Optimierung – Lösungen

- Neuronale Netze
 - Anwendung zur Optimierung
 - Anlernen der Netze
- Heuristische Suchverfahren
- Simulated Annealing
- Evolutionäre Algorithmen

Software:

<https://scikit-learn.org>

<https://www.tensorflow.org>

<http://www.clipsrules.net>

<https://opencv.org>

<https://www.r-project.org/>

<https://www.csie.ntu.edu.tw/~cjlin/libsvm>

Datensammlungen:

<http://archive.ics.uci.edu/ml/index.php>

<https://www.kaggle.com/datasets>

**ICH DANKE
FÜR IHRE AUFMERKSAMKEIT**