



**LEUPHANA**  
UNIVERSITÄT LÜNEBURG



# BUSINESS ANALYTICS

## 08. Data Mining - Klassifikation



# Agenda

- 1** Einführung
- 2** BI Strategie & Management
- 3** Data Warehouses & OLAP
- 4** ETL-Prozesse und Tools
- 5** Kommunikation und Visualisierung
- 6** Dashboards & Self-Service BI
- 7** Vorgehensmodelle BA
- 8** **Klassifikation I**
- 9** Klassifikation II
- 10** Wirkungsprognosen
- 11** Modellbeurteilung & ML-Pipelines
- 12** Zeitreihenanalyse und –vorhersage
- 13** Nichtüberwachte Verfahren
- 14** Wrap-up und Q&A



# Heutige Agenda



**Inhalte:** Anwendungsfälle und Methoden zur Klassifikation



**Lernziele:**

- a) Grundlagen der Klassifikation
- b) Überblick häufig genutzter Algorithmen
- c) Entscheidungsbäume



# Data Mining - Definition

„Data Mining bezeichnet den Prozess der **Identifikation** und Gewinnung neuer, valider und **nicht-trivialer Muster** oder Informationen. Data Mining wird zur Analyse von **umfangreichen Datenbeständen** verwendet.“

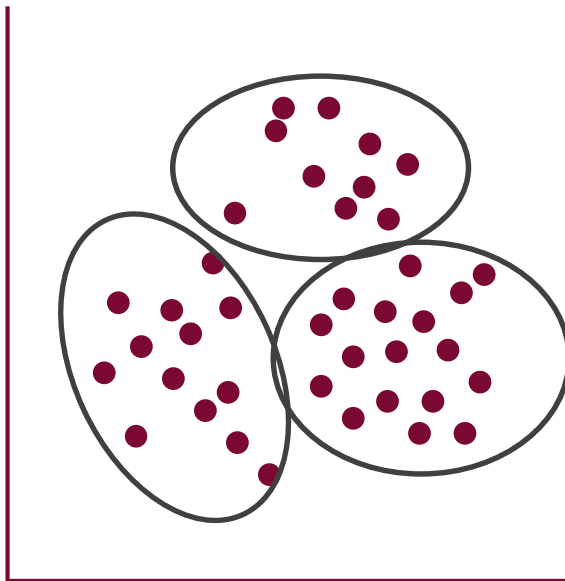
Hans-Georg Kemper



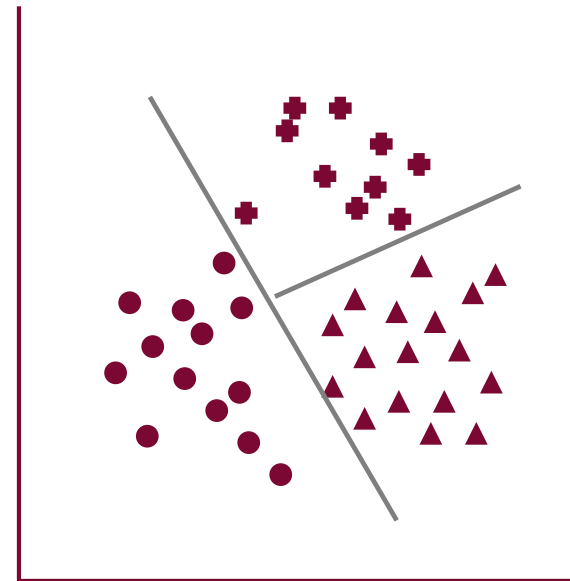


# Machine Learning

## Nicht überwacht

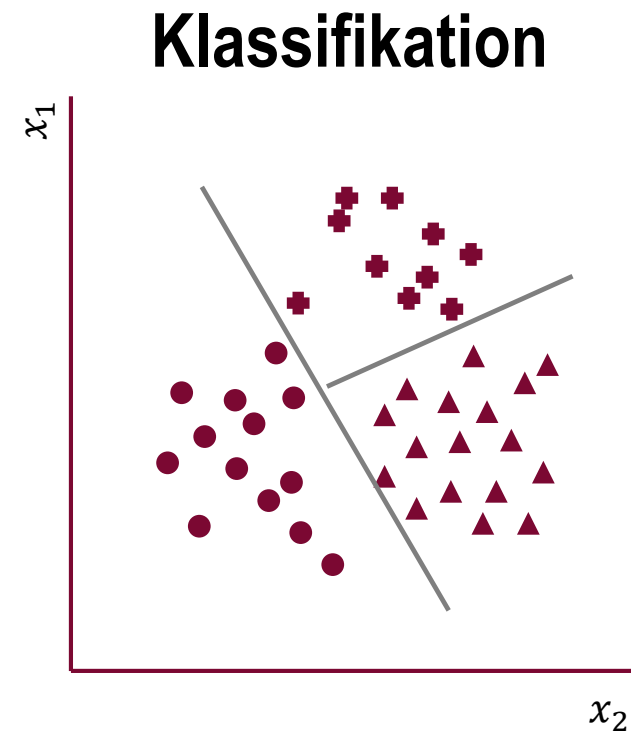
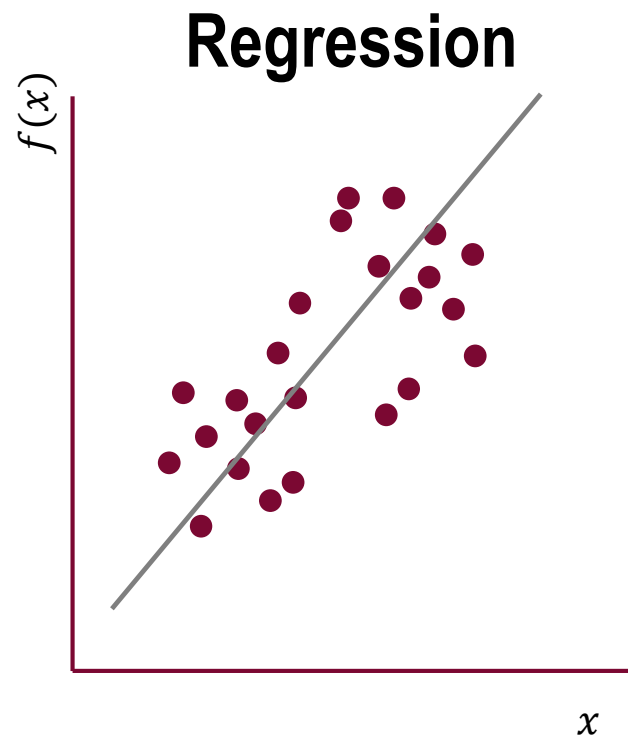


## Überwacht





# Überwachtes Lernen: kontinuierlich vs. kategorial





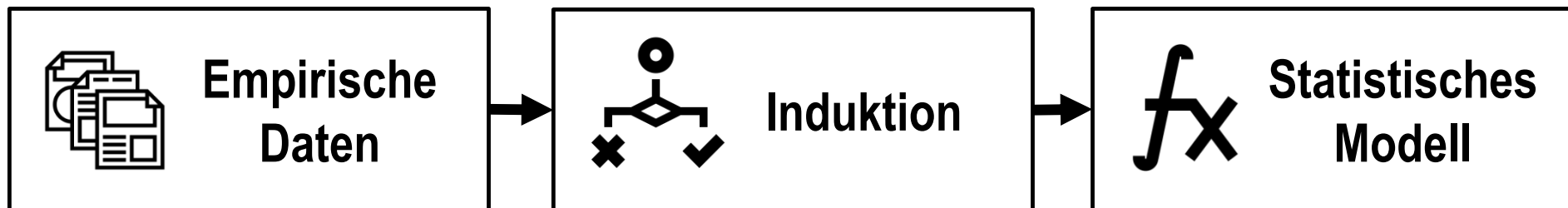
# Wiederholung Begriffe

Label/ Outcome

Beobachtungen

Maschinen_ID	Wartungs_num	Alter_brühventil	Tassen	Preis	...	Brühventil_Event
3425	2	360	7221	2022.72	...	0
11886	1	180	7194	4764.05	...	1
11893	1	94	15171	4867.56	...	1

Features/ Attribute/ Prädiktoren





# Klassische Statistik vs. Maschinelles Lernen

## Inferenz mit klassischer Statistik

### 1. Hypothesen Aufstellen

Hypothese 1: Je älter ein Brühventil, desto eher geht es kaputt

Hypothese 2: Je mehr Tassen, desto eher geht etwas kaputt

### 2. Fit und Signifikanz prüfen = Evaluation

100% der Trainingsdaten

```
Coefficients:
      Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.1675    0.1384  -1.210  0.23281
x1           0.5306    0.1754   3.025  0.00414 **
x2          -0.4115    0.1769  -2.326  0.02470 *
x3           0.1289    0.1673   0.771  0.44510
x4          -0.5884    0.1818  -3.237  0.00230 **
x5          -0.2476    0.1432  -1.728  0.09094 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

### 3. Generelle Insights:

- Brühventile immer nach X Tage austauschen
- Ab einer bestimmten Anzahl Tassen sollte der Mietpreis sich erhöhen

## Vorhersage mit ML

80% Trainingsdaten

20% Test

1. Fit:  $\hat{Y} = \beta_0 + \beta_1 * \text{Alter} + \beta_2 * \text{Tassen}$

2. Evaluation

3. Vorhersage ungesehene Daten

Ma_ID	Alter	Tassen
128323	1080	10566

$$\sigma(\beta_0 + \beta_1 * 1080 + \beta_2 * 10566) = 0.98$$

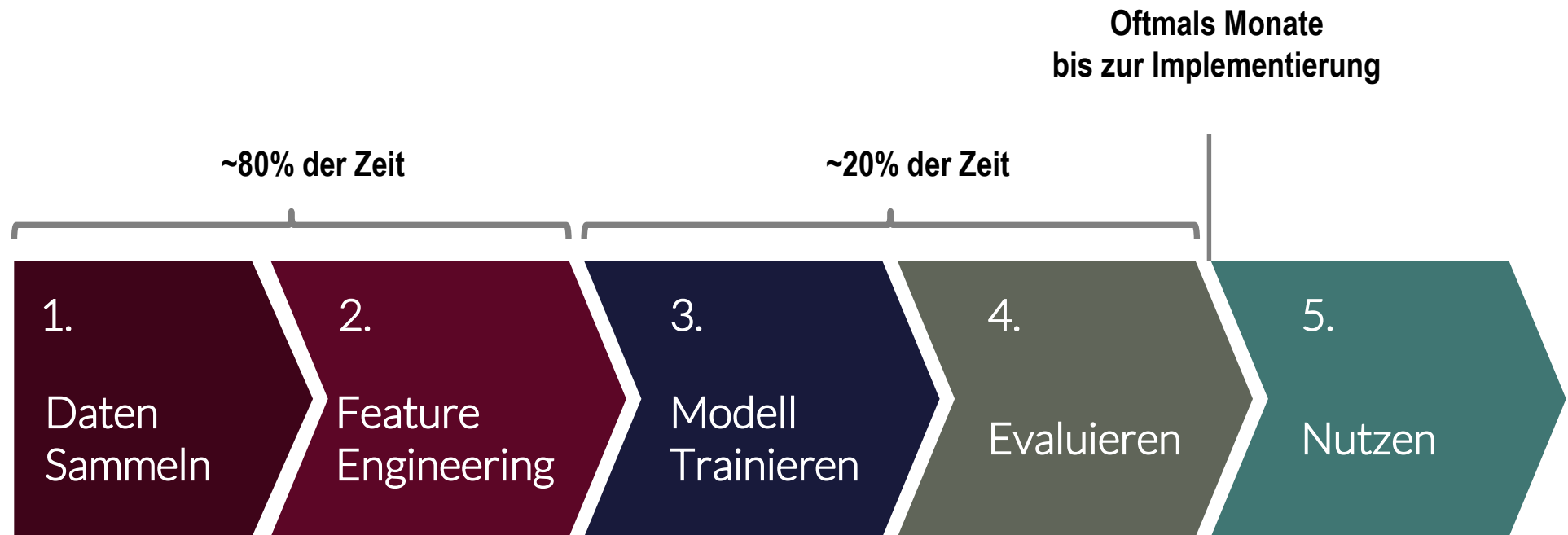
4. Automatische Individuelle Entscheidungen:

- In Wartung 2 sollte das Brühventil der Maschine 128323 repariert werden





# Rückblick – Datenprojektmanagement





# Anwendungsfälle Klassifikation

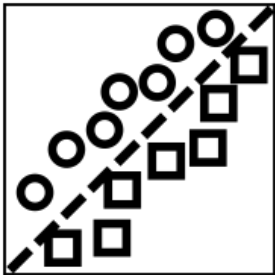
**Welche Anwendungsfälle von  
Klassifikationsverfahren kennen Sie?  
Worin besteht der Nutzen?**



# Algorithmen – Versuch einer Übersicht

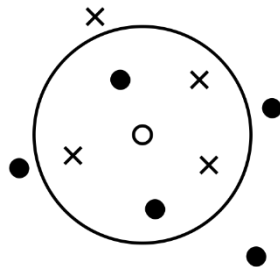
## Linear

Annahme: „lineare“  
Hyperebene trennt  
Gruppen



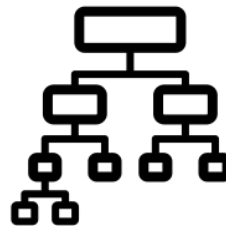
## Instance-Based

Klassifikation erfolgt  
aufgrund der Klassen  
der Nachbarn (Lazy  
Learning)



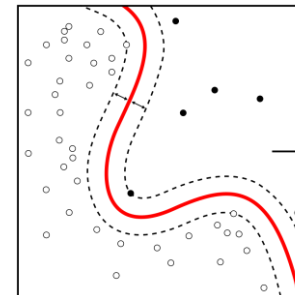
## Tree-Based

Klassifikation anhand  
eines Entscheidungs-  
baums



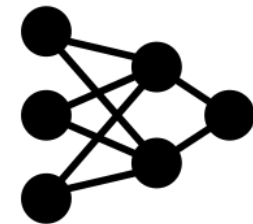
## Kernel-Based

Verwendung nicht-  
linearer  
Transformationen



## Neurale Netze

Verbindung einfacher  
Funktionen zu  
komplexen Netzen

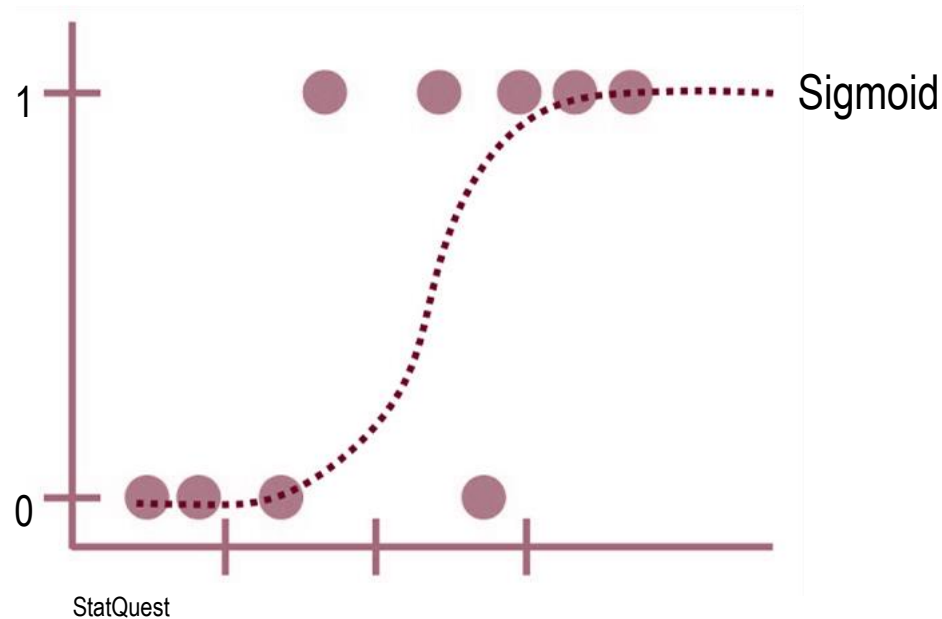




# Logistische Regression

**Linearer Zusammenhang:**  $z = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n$

**Sigmoid Output:**  $y = \sigma(z) = \frac{1}{1+e^{-z}}$



## Algorithmus:

- Schätzung der Parameter  $\beta_i$  anhand der Daten
- **Zur Schätzung Lösung eines Optimierungsproblem erforderlich (mehr dazu in der Einführung KI Vorlesung)**

## Bewertung:

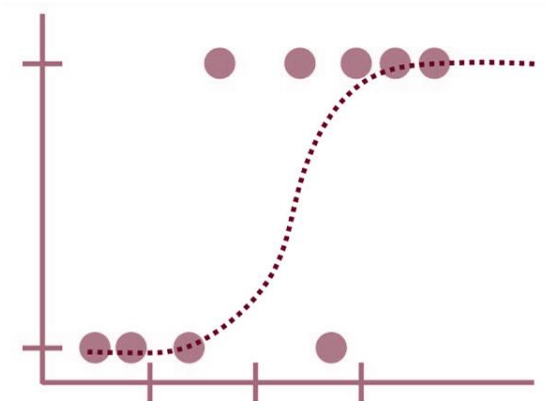
- Sehr einfaches Modell, aber robust und einfach umsetzbar
- Gute Interpretierbarkeit
- Kurze Trainingszeiten, funktioniert auf kleinen und großen Datensätzen



# Logistische Regression – Anwendung Kundenabwanderung



**Linearer Zusammenhang:**  $z = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n$

**Sigmoid Output:**  $y = \sigma(z) = \frac{1}{1+e^{-z}}$





# Umsetzung in Python

 jupyterhub logreg Last Checkpoint: vor 24 Minuten (autosaved)  Logout Control Panel

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

Memory: 183.6 MB / 2.5 GB Memory: 183.6 MB / 2.5 GB

## Logistic Regression

Wir verwenden hier einen bekannten Datensatz von Kaggle (<https://www.kaggle.com/datasets/blastchar/telco-customer-churn/>), um zu demonstrieren, wie die logistische Regression funktioniert.

```
In [ ]: import pandas as pd
import numpy as np
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.linear_model import LogisticRegression
from sklearn.pipeline import Pipeline
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.impute import SimpleImputer

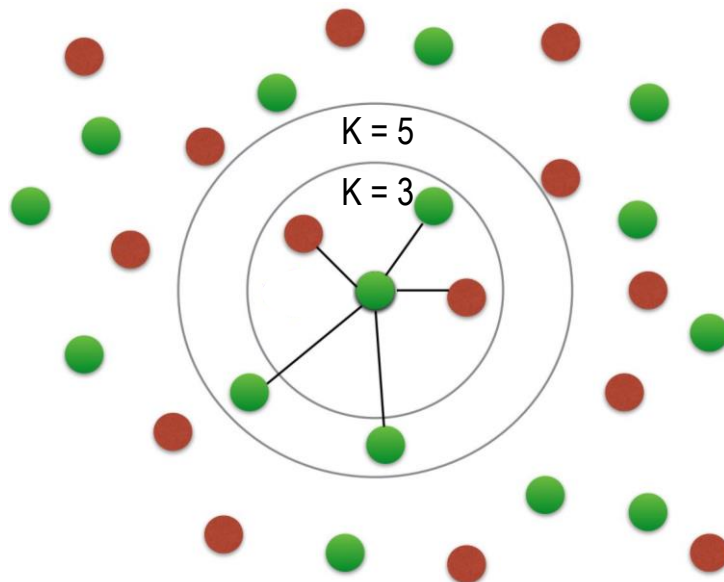
# Load dataset
df = pd.read_csv('WA_Fn-UseC_-Telco-Customer-Churn.csv')

# Convert empty strings in 'TotalCharges' to NaN and then to float
df['TotalCharges'] = pd.to_numeric(df['TotalCharges'].replace(' ', np.nan))

# Preprocessing
X = df.drop(['Churn', 'customerID'], axis=1)
y = df['Churn'].apply(lambda x: 1 if x == 'Yes' else 0)
```



# K nächste Nachbarn



Data Camp

## Algorithmus:

- Vorhersage basiert auf der Klassenzugehörigkeit der  $k$  nächsten Nachbarn (Mehrheitsvotum)
- „Lazy learner“, da nicht trainiert wird, sondern Entscheidung anhand der umliegenden Punkte erfolgt

## Bewertung:

- Kein Trainingsaufwand, Vorhersageaufwand geht mit  $\mathcal{O}(k \log n)$
- Distanzbasierter Ansatz funktioniert bei niedrig- nicht aber hochdimensionalen Problemen („Curse of Dimensionality“)
- Intuitive, modellfreie Methode



# Umsetzung in Python

The screenshot shows a Jupyter Notebook titled 'knn' with a last checkpoint of 'vor ein paar Sekunden (autosaved)'. The interface includes a top bar with 'Logout' and 'Control Panel' buttons, and a menu bar with 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. A toolbar below the menu bar contains icons for saving, adding cells, undo, redo, and running code. The notebook content is titled 'KNN' and contains the following Python code:

```
In [ ]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.neighbors import KNeighborsClassifier
from sklearn.pipeline import Pipeline
from sklearn.metrics import roc_auc_score, confusion_matrix, ConfusionMatrixDisplay
from sklearn.impute import SimpleImputer

# Load dataset
df = pd.read_csv('WA_Fn-UseC_-Telco-Customer-Churn.csv')
df['TotalCharges'] = pd.to_numeric(df['TotalCharges'].replace(' ', np.nan))

# Preprocessing
X = df.drop(['Churn', 'customerID'], axis=1)
y = df['Churn'].apply(lambda x: 1 if x == 'Yes' else 0)

# Define categorical and numeric features
categorical_features = ['gender', 'Partner', 'Dependents', 'PhoneService', 'MultipleLines',
                        'InternetService', 'OnlineSecurity', 'OnlineBackup', 'DeviceProtection',
                        'TechSupport', 'StreamingTV', 'StreamingMovies', 'Contract',
                        'PaperlessBilling', 'PaymentMethod']
numeric_features = ['SeniorCitizen', 'tenure', 'MonthlyCharges', 'TotalCharges']
```





# Zusammenfassung



Klassifikationsmodelle sagen die Zugehörigkeit einzelner Beobachtungen zu definierten Klassen (Labels) „voraus“



Es gibt eine Vielzahl von Klassifikationsalgorithmen, die sehr unterschiedliche Ansätze nutzen



# Gastvortrag



- **Dr. Martin Stange**, Data Scientist bei AboutYou
- Thema: **ML Pipelines in der Marketing Steuerung**
- Dipl. Physiker (Hannover), M.Sc. WI (Wismar), Promotion Dr. rer. nat. (Leuphana & WU)
- Berufserfahrung: Analyst Werum, Data Scientist Dreamlines