TECHNICAL APPLICATIONS AND DATA MANAGEMENT. WS 2022.

VORLESUNG 5

18.10.2022

MÜNCHEN

STUDIENGANG SUSTAINABILITY MANAGEMENT & LEADERSHIP SOWIE MEDIEN & KOMMUNIKATION.

AGENDA

1. Wahl Projektarbeit Data Science
2. Einführung Maschinelles Lernen
3. Unüberwachtes Lernen

GEPLANTE ROADMAP VORLESUNG.

ROADMAP	WAS HABEN WIR VOR?
Vorlesung 1	Workflow Data Management, Datentypen und Datenqualität
Vorlesung 2	Einführung Data Science und Data Science Workflow, Grundlagen Data Management
Vorlesung 3 und Vorlesung 4	Deskriptive und explorative Datenanalyse und Vertiefung anhand Case Study
	Vertiefung Datenanalyse anhand Case Study
Vorlesung 5	Aufgabenstellung Data Science, Übersicht und Einführung Machine Learning, unüberwachtes Lernen
Vorlesung 6	Überwachtes Lernen
Vorlesung 7	Vertiefung überwachtes Lernen anhand Case Study
Vorlesung 8	Neuronale Netze und Convolutional Neural Networks (CNN)
Vorlesung 9	Vertiefung CNN anhand Case Study, Aufgabenstellung AI
Vorlesung 10	Schulterblick 1
Vorlesung 11	Übersicht Rekurrente Neuronale Netze
Vorlesung 12	Schulterblick 2
Vorlesung 13	Ausblick zukünftige AI-Themen, „Fragestunde“
Vorlesung 14	Präsentation Ergebnisse

Zusammenlegen!

Zusammenlegen!

WAS MACHEN WIR HEUTE?

- Wahl Projektarbeit Data Science
- Einführung Künstliche Intelligenz: was ist künstliche Intelligenz?
- Unsupervised Learning/ Unüberwachtes Lernen
  - Entdecken von Anomalien
  - ~~Reduktion Dimensionalität von Daten (am Beispiel der Principal Component Analysis)~~
  - Segmentierung/ Clustering
  - Assoziationsanalyse: Market Basket Analyse und Recommender

1. WAHL PROJEKTARBEIT DATA SCIENCE

WAHL PROJEKTARBEIT: FOLGENDE THEMEN STEHEN FÜR DEN DATA SCIENCE-ANTEIL ZUR AUSWAHL.

- Stroke Prediction (https://www.kaggle.com/fedesoriano/stroke-prediction-dataset)
- Diabetes Data Set (https://www.kaggle.com/datasets/brandao/diabetes?select=diabetic_data.csv)
- Boston Housing pricing (https://www.kaggle.com/altavish/boston-housing-dataset)
- Kepler Exoplaneten Suche (https://www.kaggle.com/nasa/kepler-exoplanet-search-results)

Jeder wählt eine der Aufgaben

WAS IST IM RAHMEN PROJEKTARBEIT ZU TUN? SCHULTERBLICK 1 IN VORLESUNG AM 08.11.2022.

- Zu erstellen ist eine Powerpoint oder ein Word-Dokument (max. 4 Seiten) mit:
  - Problem statement: „Welches Problem will ich lösen? Wieso ist es ein Problem? Was ist der Nutzen einer Lösung?“
  - Metriken zur Evaluation Ergebnisse.
  - Vorgehensweise Lösungsansatz anhand Data Science Workflow: „Wie gehe ich es an?“
  - Aktueller Status: gibt’s Probleme? Wie kommen Sie voran?

- Kurze Vorstellung durch Bearbeiter in der Vorlesung.

- Gemeinsame Diskussion.

Templates finden Sie auf der Homepage des Kurses unter Materialien

WAS IST IM RAHMEN PROJEKTARBEIT ZU TUN? 2. TEIL: ABGABEUMFANG ENDE DES SEMESTERS

- Schriftliche Ausarbeitung je Teilnehmer mit maximal 12 Seiten:
  - Projektübersicht
  - Vorgehensweise anhand Data Science Workflow (Get Data, Explore the Data, Model the Data, Visualise Results):
  - eingesetzte Verfahren
  - Implementierung Datenaufbereitung, Datenmodellierung und Datenvisualisierung
  - Ergebnisse: Visualisierung Ergebnisse und Bewertung anhand Metriken
  - Reflektion: Was lief gut? Was lief schlecht?
  - Future Work: „Was wären nächste Schritte?“
  - Literaturverzeichnis
- Notebook

Templates finden Sie auf der Homepage des Kurses unter Materialien

2. KÜNSTLICHE INTELLIGENZ

KÜNSTLICHE INTELLIGENZ: WAS IST DAS?

- Definitionen für Künstliche Intelligenz:
  - Encyclopedia Britannica: “ability … to perform tasks commonly associated with intelligent beings”.
  - Russel, Norvig: “AI is concerned with not just understanding but also building intelligent entities—machines that can compute how to act effectively and safely in a wide variety of novel situations”.

- Wir unterscheiden:
  - schwache AI: AI beherrscht genau ein Thema, das Wissen ist somit nicht/ eingeschränkt auf andere Themen übertragbar.
  - Starke AI: universell einsetzbare AI. Existiert heute noch nicht (Zukunft?).

Quelle: Russel, Norvig:“Artificial Intelligence – A modern approach“ (4. Auflage)

GESCHICHTE KÜNSTLICHE INTELLIGENZ. VON DEN ANFÄNGEN…

- **Initialphase (1943 – 1956):**
  - **McCulloch & Pitts (1943):** Modell für künstliche Neuronen, das berechenbare sowie Logikfunktionen abbilden konnte.
  - **Hebb (1949):** Definition Update-Regel für Neuronale Netze.
  - **Minsky & Edmonds (1950):** Erster Neural Network Computer.
  - **Turing (1950):** Turing Test, ob eine Maschine ein dem Menschen gleichwertiges Denkvermögen hat.
  - **Strachey, Samuel (1952):** Computerprogramm, das selbstständig Mühle spielt.

- **Grosse Erwartungen (1952–1969):**
  - **Newell, Simon (ab 1957):** General Problem Solver.
  - **Dartmouth Summer Research Project (1956)**: "…2-month, 10-man study … to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it. …We think that a significant advance can be made in one or more of these problems if a carefully selected group of scientists work on it together for a summer.
  - **Samuel (1956):** Erweiterung Mühle-Programm um Reinforcement Learning.
  - **Gelernter (1959):** Geometry Theorem Prover für Beweis mathematischer Theoreme.

Quelle: aus Russel, Norvig:"Artificial Intelligence – A modern approach (4. Auflage)

GESCHICHTE KÜNSTLICHE INTELLIGENZ. ÜBER UMWEGE…

- **“A dose of reality (1966–1973)”:**
  - **Lighthill Report (1973):** Rechen-Power nicht ausreichend für komplexere Themen aufgrund kombinatorischer Explosion.

- **Expertensysteme (1969–1986):**
  - **Buchanan (1969):** Schlußfolgerung Molekularstruktur anhand Info aus Massenspektrometer.
  - **Buchanan (1972):** Mycin für Blutinfektionen, basierend auf Experteninterviews inkl. Unsicherheitsfaktoren (Heuristiken).

- **Probabilistische Inferenz und Machine Learning (1987– present):**
  - **Pearl (1988):** Probabilistic Reasoning in Intelligent System, Bayes Networks.
  - **Sutton (1988):** Temporal Difference, Kombination aus Reinforcement Learning und statistischen Markovprozessen.
  - **LeCun (1995):** Einsatz Neuronaler Netzwerke für Handschrifterkennung.
  - **Hochreiter, Schmidhuber (1997):** Long Short-Term Memory → bis heute Standard für Spracherkennung und -auswerten

Quelle: aus Russel, Norvig:“Artificial Intelligence – A modern approach (4. Auflage)

GESCHICHTE KÜNSTLICHE INTELLIGENZ. …ZUR HEUTIGEN WICHTIGKEIT.

- **Big Data (2001 – heute):**
  - **Yarowsky (1995):** Mehrdeutigkeit Wörter kann für sehr große Datensätzen mit Genauigkeit von 96% aufgelöst werden.
  - **Banko, Brill (2001):** Vergrößern Datenmenge um Faktor 2 oder 3 ist besser als jedes Tunen der Parameter Algorithmus.
  - **Deng (2009):** ImageNet-Database mit >> 10 Mio. Bildern.

- **Deep Learning (2011):**
  - **Ciresan (2011), Krizhevsky (2013):** Bildererkennung per Deep Learning massiv besser als vorherige, teils manuelle, Verfahren und vor allem mit Menschenähnlicher Genauigkeit.
  - **Silver et al (2013):** Playing Atari with Deep Reinforcement Learning – selbstständiges Spielen von Atari-Spielen.
  - **Silver et al. (2016, 2017, 2018):** AlphaGo - Kombi aus Deep Learning und Reinforcement Learning schlägt Go-Großmeister
  - **Silver et al. (2017, 2018)**: AlphaGoZero - Weiterentwicklung AlphaGo. Programm kennt nur Regeln und lernt durch Spielen gegen sich selber.
  - **Senior et al, Jumper et al (2019, 2020)**: AlphaFold – Prädiktion 3D-Proteinstruktur basierend auf Aminosäuresequenz
  - Und vieles, vieles mehr…..

Quelle: aus Russel, Norvig:"Artificial Intelligence – A modern approach (4. Auflage)

MACHINE LEARNING.

- **Machine Learning ist ein Teilgebiet der künstlichen Intelligenz:**
  - **Samuel (1959):** Field of study that gives computers the ability to learn without being explicitly programmed
  - **Mitchell (1998):** a computer program is said to *learn* from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.

Quelle: aus Russel, Norvig:"Artificial Intelligence – A modern approach (4. Auflage)

UNTERSCHEIDUNG MACHINE LEARNING

Unsupervised Learning

Lernen ohne vorher definierte Zielwerte oder Belohnung

Supervised Learning

Agent/ Algorithmus lernt eine Funktion, die Eingabegrößen auf vorher definierte Outputs mappt.

Reinforcement Learning

Agent/ Algorithmus lernt selbständig mit dem Ziel, eine Belohnung zu maximieren.

Nur kurzer Einblick am Ende des Semesters

3. UNÜBERWACHTES LERNEN

ÜBERSICHT UNSUPERVISED LEARNING.

Wofür brauchen wir das?

- Verbessern der Datenqualität
- Reduktion sehr großer Datenmengen
- Entdecken unbekannter Zusammenhänge, bspw. bei Online-Kunden

Was schauen wir uns an?

- Entdecken von Anomalien/ ungewöhnlichen Daten am Beispiel IsolationForest.
- Clustering/ Segmentierung am Beispiel kMeans.
- ~~Reduzierung Features (PCA).~~
- Clustering/ Segmentierung am Beispiel kMeans.
- Association Rules am Beispiel Market Basket Analysis und Empfehlungssysteme.

3.1 ENTDECKEN VON ANOMALIEN

OUTLIER DETECTION: WAS IST DAS? WIE GEHT DAS?

Ziel: „process of finding data objects with behaviors that are very different from expectation”[1]

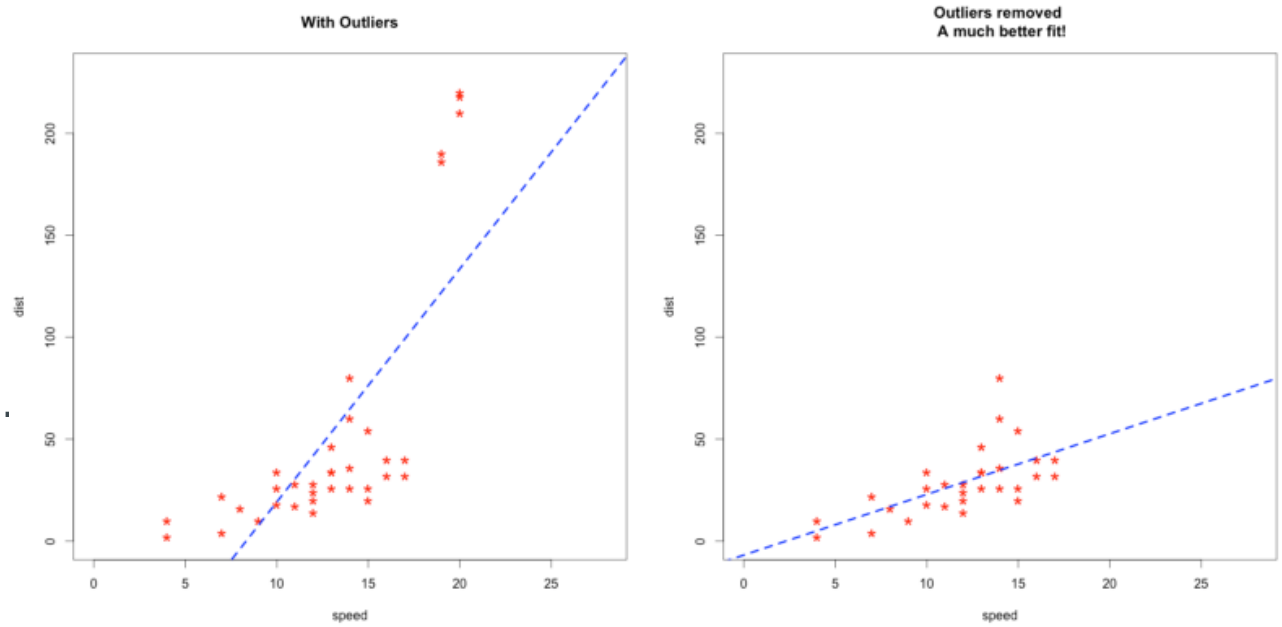
Ermöglicht:

- Verbesserung Datenqualität.
- Optimierung vieler Machine Learning Algorithmen (besonders linearer Verfahren).

Aber: “Outlier” können auch reale, vernünftige Werte sein.

Anwendungsbeispiele:

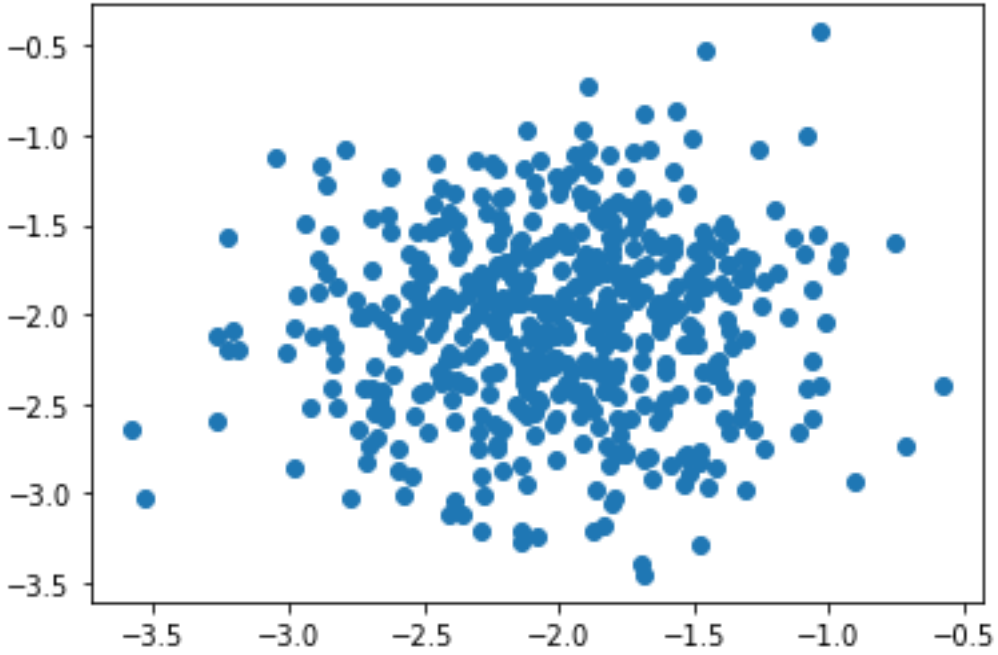
- Fraud detection bei Kreditkarten.
- Plausibilisieren Sensorenwerte.



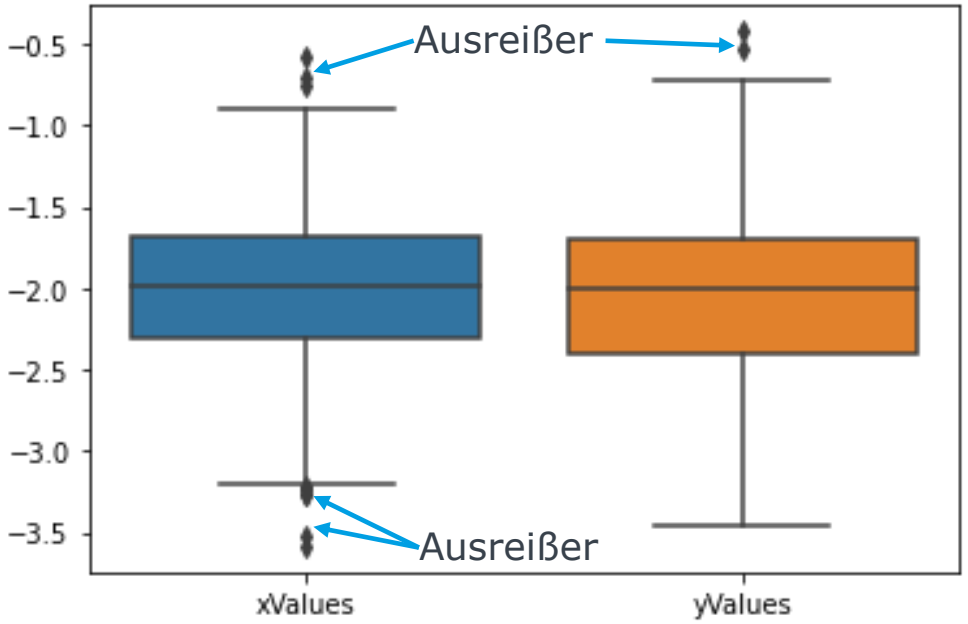
Quellen: 1 Han et al., 2011: Data Mining. Concepts and Techniques
Bild: Link, abgerufen am 10.10.2021

ENTDECKEN VON ANOMALIEN: PER DESKRIPTIVE ANALYSEN.

Graphische Darstellung Verteilung:



Outlier Analyse mit Box-Plot:

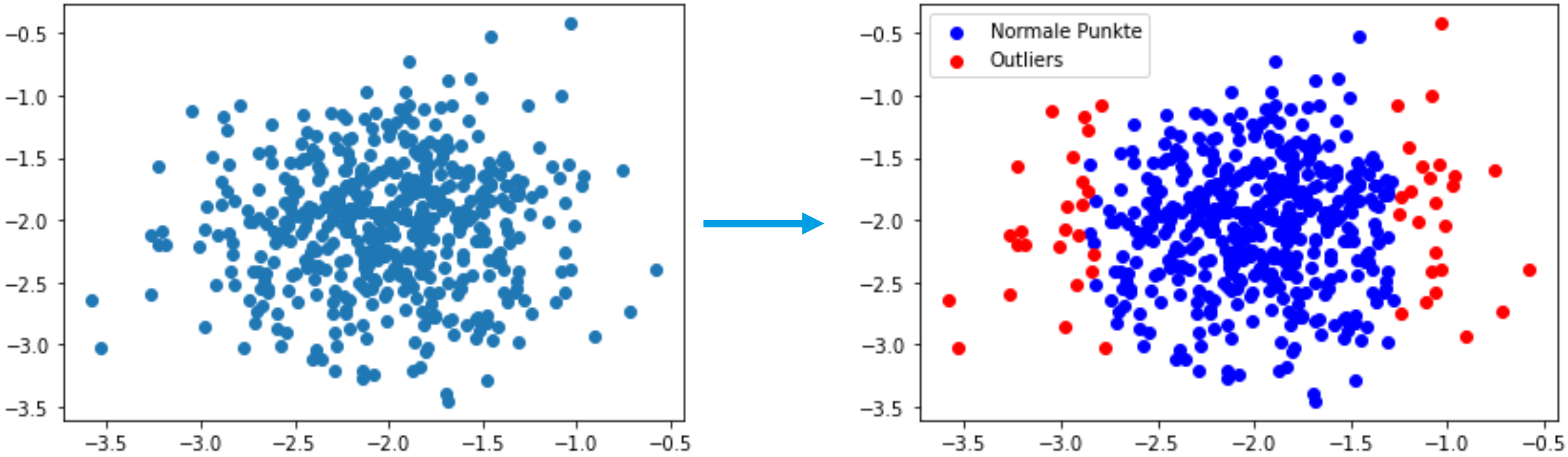


Haben wir letzte Woche schon gemacht….

Quelle Bilder: Visualisierungen eigenes Python-Skript

ENTDECKEN VON ANOMALIEN: ISOLATION FOREST ALGORITHM.

Für Entdecken von Outliern gibt es einige Algorithmen, PyOD (Link) bündelt die verschiedenen Verfahren.




```
from sklearn.ensemble import IsolationForest
isolation_forest = IsolationForest(n_estimators=100, contamination=0.1)
y_hat = isolation_forest.fit_predict(x.reshape(-1, 1))
```

Quelle Bilder: Visualisierungen eigenes Python-Skript

3.2 CLUSTERING/ SEGMENTIERUNG

CLUSTERING.

Ziel: Einteilen eines Datensatzes in k verschiedene Gruppen durch einen Algorithmus (Unterschied zur Klassifizierung).

Ermöglicht:

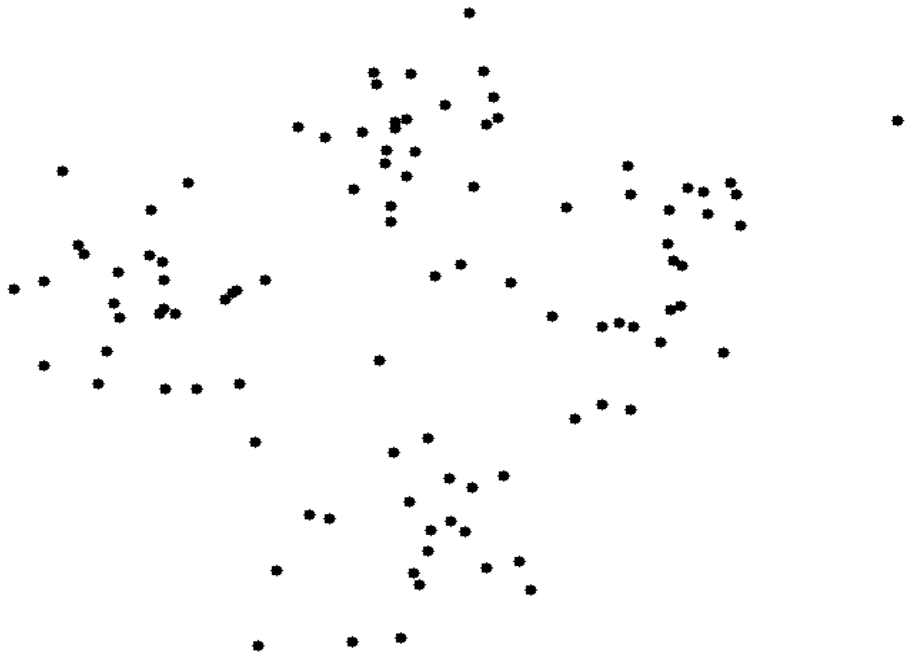
- Erkennen von Strukturen in einem Datensatz.
- Einordnen eines Datensatzes in kleinere Gruppen mit ähnlichen Daten/ Verhalten.

Aber: Ergebnis muß von Experten beurteilt werden, denn der Algorithmus liefert keine Begründung für die Einteilung!

Anwendungsgebiete:

- Generalisieren von Daten.
- Kundenanalyse.
- Sequenzanalyse Biologie (Entdecken von charakteristischen Teilen in der DNA).
- Image segmentation.

CLUSTERING: K-MEANS ALGORITHMUS.



Algorithmus für Clusterung eines Datensatzes X in k Cluster:

1. Wähle k zufällige Punkte als Cluster-Zentrum (Centroid).
2. Berechne für jeden Datenpunkt x_i Distanz zu einem der Centroiden. Weise x_i dem Centroid mit geringstem Abstand zu.
3. Finde für jeden Cluster sein neues Zentrum durch Bilden Durchschnitt aller seiner Punkte (Cluster ist arithmetisches Mittel).
4. Wiederhole Schritte 2 und 3 bis sich keine Cluster-Zuweisung mehr ändert.

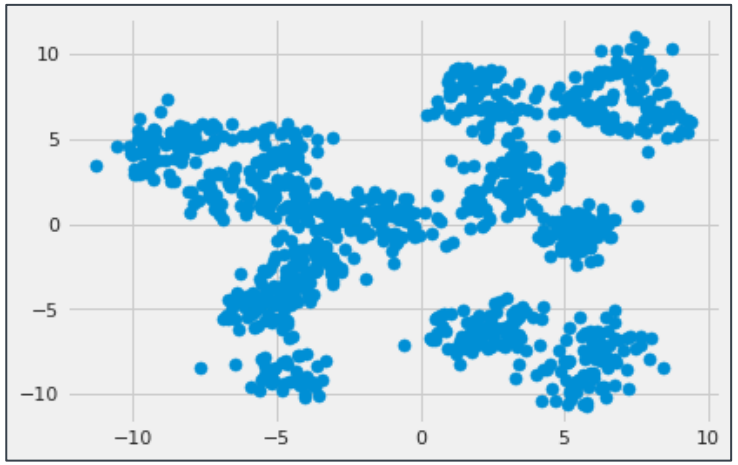
```
from sklearn.cluster import KMeans
kmeans = KMeans(n_clusters=k)
kmeans.fit(X)
y_kmeans = kmeans.predict(X)
```

ABER: wie viele Cluster wählt man?

Quellen:
Animation: Link
kMeans zuerst definiert in S. P. Lloyd: *Least square quantization in PCM, Bell Telephone Laboratories Paper*, 1957.

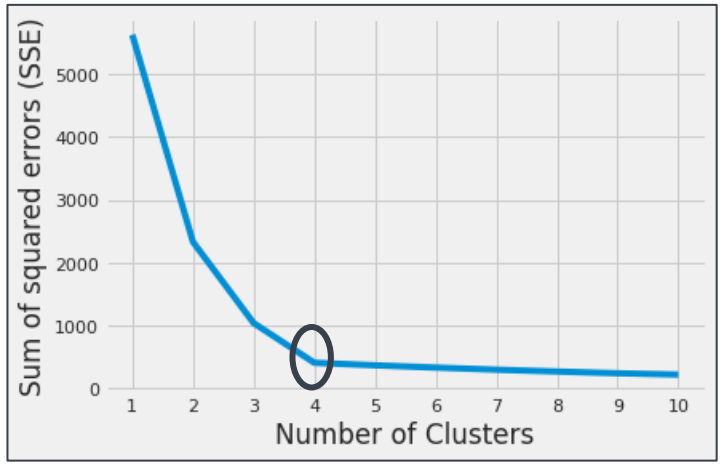
IDENTIFIKATION „IDEALER“ ANZAHL CLUSTER AM BEISPIEL „ELLBOW“-METHODE

Gesucht: „ideale“ Anzahl Cluster

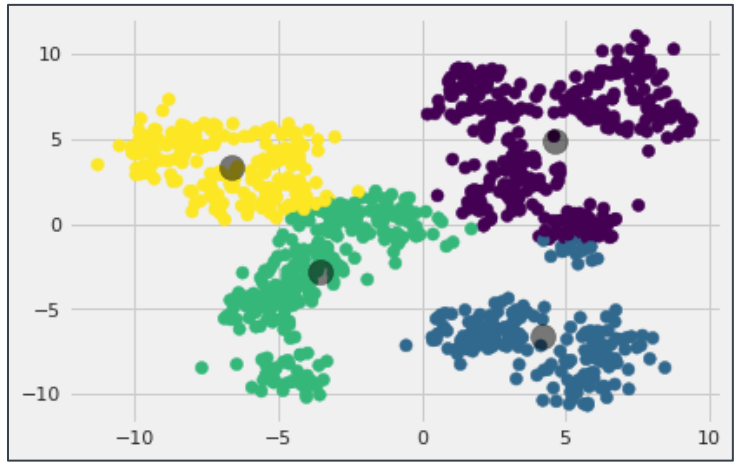


- Je weniger Cluster, desto grober die Einteilung und desto höher Fehler.
- Je mehr Cluster, desto genauer eine Einteilung, aber desto weniger eine Generalisierung.

„Ellbow“-Trick:



- Durchführen kMeans für großes k
- kMeans Metrik Sum of Squared Errors (SSE) plotten.
- Ellbogen finden: im Fallbeispiel 4.



kMeans mit dem durch Ellbogen gefundenen Wert 4 durchführen.

Quellen:
Bilder: Visualisierungen eigenes Python-Skript
Ellbow-Trick: R. L. Thorndike, "Who Belongs in the Family?". Psychometrika 18 (4): 267–276, 1953

3.4 ASSOZIATIONSREGELN

ASSOZIATIONSREGELN[1].

Ziel: automatisiertes Erkennen von Zusammenhängen/ Abhängigkeiten innerhalb eines Datensatz

Ermöglicht:

- Definition einfacher, leicht verständlicher Regeln: „Wenn A gekauft wurde, dann auch gleichzeitig B".
- Aber: Correlation does not imply causation!

Anwendungsgebiete:

- Market Basket Analyse („Frequently bought together"):

- Recommender System:

Quellen: 1 Agrawal et al. (1993). "Mining association rules between sets of items in large databases". *Proceedings of the 1993 ACM SIGMOD international conference on Management of data*.
Bilder: Screenshots Amazon-Homepage.

ASSOZIATIONSREGELN: FALLBEISPIEL „NEULICH IM SUPERMARKT".

Datenset Transaktionsliste

	0	1	2	3	4	5	6
0	Bread	Wine	Eggs	Meat	Cheese	Pencil	Diaper
1	Bread	Cheese	Meat	Diaper	Wine	Milk	Pencil
2	Cheese	Meat	Eggs	Milk	Wine	NaN	NaN
3	Cheese	Meat	Eggs	Milk	Wine	NaN	NaN
4	Meat	Pencil	Wine	NaN	NaN	NaN	NaN
5	Eggs	Bread	Wine	Pencil	Milk	Diaper	Bagel
6	Wine	Pencil	Eggs	Cheese	NaN	NaN	NaN
7	Bagel	Bread	Milk	Pencil	Diaper	NaN	NaN
8	Bread	Diaper	Cheese	Milk	Wine	Eggs	NaN
9	Bagel	Wine	Diaper	Meat	Pencil	Eggs	Cheese

Gegeben:
- Dataset mit Transaktionen
- Schranken für Minimum Support und Minimum Confidence

Gesucht: belastbare Regeln
$\{X_1 \ldots X_n\} \rightarrow \{Y_1 \ldots Y_n\}$.

Finde häufige Mengen

	support	itemsets
0	0.501587	(Milk)
1	0.425397	(Bagel)
2	0.501587	(Cheese)
3	0.438095	(Wine)
4	0.476190	(Meat)
5	0.438095	(Eggs)
6	0.361905	(Pencil)
7	0.406349	(Diaper)
8	0.504762	(Bread)
9	0.225397	(Milk, Bagel)

Finde alle 1…n Teilmengen mit höherem Support-Wert/ relativer Häufigkeit als Minimum Support.

Erzeuge Assoziationsregeln

14 Regeln gefunden

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(Milk)	(Cheese)	0.501587	0.501587	0.304762	0.607595	1.211344	0.053172	1.270148
1	(Cheese)	(Milk)	0.501587	0.501587	0.304762	0.607595	1.211344	0.053172	1.270148
2	(Bagel)	(Bread)	0.425397	0.504762	0.279365	0.656716	1.301042	0.064641	1.442650
3	(Wine)	(Cheese)	0.438095	0.501587	0.269841	0.615942	1.227986	0.050098	1.297754
4	(Cheese)	(Meat)	0.501587	0.476190	0.323810	0.645570	1.355696	0.084958	1.477891
5	(Meat)	(Cheese)	0.476190	0.501587	0.323810	0.680000	1.355696	0.084958	1.557540
6	(Eggs)	(Cheese)	0.438095	0.501587	0.298413	0.681159	1.358008	0.078670	1.563203
7	(Eggs)	(Meat)	0.438095	0.476190	0.266667	0.608696	1.278261	0.058050	1.338624
8	(Milk, Meat)	(Cheese)	0.244444	0.501587	0.203175	0.831169	1.657077	0.080564	2.952137
9	(Milk, Cheese)	(Meat)	0.304762	0.476190	0.203175	0.666667	1.400000	0.058050	1.571429
10	(Cheese, Meat)	(Milk)	0.323810	0.501587	0.203175	0.627451	1.250931	0.040756	1.337845
11	(Cheese, Eggs)	(Meat)	0.298413	0.476190	0.215873	0.723404	1.519149	0.073772	1.893773
12	(Cheese, Meat)	(Eggs)	0.323810	0.438095	0.215873	0.666667	1.521739	0.074014	1.685714
13	(Eggs, Meat)	(Cheese)	0.266667	0.501587	0.215873	0.809524	1.613924	0.082116	2.616667

1. Nimm Regeln aus vorigem Schritt, bilde Regeln und berechne deren Konfidenz.
2. Lösche Regel falls Konfidenz ≤ Min. Confidence.
3. Sortiere Regeln nach absteigendem Lift-Wert.
4. Darstellung Regeln mit Support und Konfidenz.

$$\text{Support (Item I)} = \frac{Anzahl\ Transaktionen\ mit\ Item\ I}{Gesamtzahl\ Transaktionen}$$

$$\text{Confidence (Item I1} \rightarrow \text{Item I2)} = \frac{Anzahl\ Transaktionen\ mit\ I_1 und\ I_2}{Anzahl\ Transaktionen\ mit\ I_1}$$

$$\text{Lift (Item I1} \rightarrow \text{Item I2)} = \frac{Confidence(ItemI_1 \rightarrow I_2)}{Support(ItemI_2)}$$

Quelle: Agrawal et al. (1993). "Mining association rules between sets of items in large databases". Proceedings of the 1993 ACM SIGMOD international conference on Management of data.

RECOMMENDER

- Einfacher Recommender: Top 20 Bücher/ Filme/ ….
- Collaborative Filtering:
  - finde ähnliche Nutzer mit gleichem Verhalten und nutze deren Verhalten für Vorhersage.
  - Amazon-Ansatz: erstelle Ähnlichkeitsmatrix zwischen Produkten und leite daraus Vorlieben einzelner User ab.
- Content-based filtering: empfehle ähnliche Produkte (gleicher Hersteller/ Schauspieler/ Genre/…).

Folgefolie

Natural Language Processing, nicht Fokus Vorlesung

COLLABORATIVE FILTERING RECOMMENDER. FALLBEISPIEL FILME

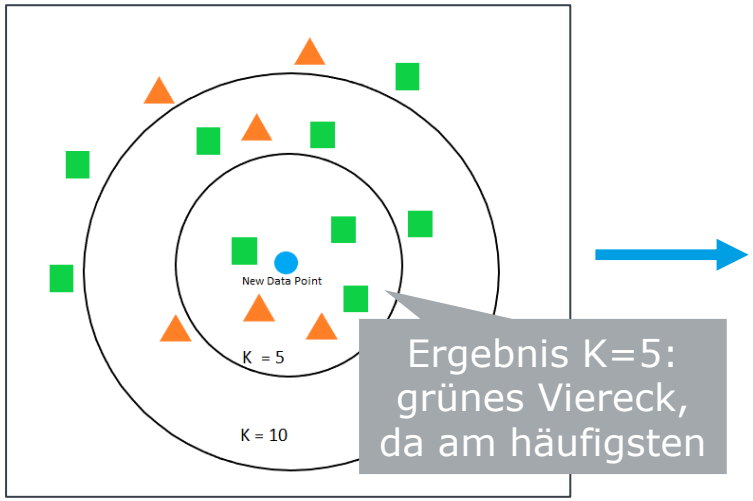
Datenset Filme

movieId	rating	title	genres
296	50	Pulp Fiction (1994)	Comedy|Crime|Drama|Thriller
306	35	Trois couleurs: Rouge (1994)	Drama
307	50	Trois couleurs: Bleu (1993)	Drama
665	50	Underground (1995)	Comedy|Drama|War
899	35	Singin' in the Rain (1952)	Comedy|Musical|Romance
1088	40	Dirty Dancing (1987)	Drama|Musical|Romance
1175	35	Delicatessen (1991)	Comedy|Drama|Romance
1217	35	Ran (1985)	Drama|War
1237	50	Sjunde inseglet, Det (1957)	Drama

Gegeben: Datenset mit Filmen, Genre und User-Bewertung.

Gesucht: Filme, die Nutzern mit ähnlichen Vorlieben gefielen.

K-Nearest Neighbours Algorithm[1]



Trainiere K-Nearest Neighbours auf Datensatz

```
# By specifying the metric = cosine,
# the model will measure similarity
# bectween artist vectors by using
# cosine similarity.
model_knn = NearestNeighbors(metric='cosine',
                             algorithm='brute',
                             n_neighbors=20,
                             n_jobs=-1)

model_knn.fit(movie_user_mat_sparse)
```

Recommender Modell

```
make_recommendation(
    model_knn=model_knn,
    data=movie_user_mat_sparse,
    fav_movie="Wolf of Wall Street",
    mapper=movie_to_idx,
    n_recommendations=10)
```

```
You have input movie: Wolf of Wall Street
Found possible matches in our database: ['Wolf of Wall Street, The (2013)']

Recommendation system start to make inference
......

Recommendations for Wolf of Wall Street:
1: Inglourious Basterds (2009), with distance of 0.47006362676620483
2: Ex Machina (2015), with distance of 0.46493279933929443
3: The Imitation Game (2014), with distance of 0.45827585458755493
4: The Martian (2015), with distance of 0.447864294052124
5: Shutter Island (2010), with distance of 0.4431135058403015
6: Dark Knight Rises, The (2012), with distance of 0.4414052963256836
7: Gone Girl (2014), with distance of 0.43579208850860596
8: Inception (2010), with distance of 0.42389780282974243
9: Interstellar (2014), with distance of 0.3947324752807617
10: Django Unchained (2012), with distance of 0.363944947719574
```

Algorithmus gibt zu einem Film die 10 Filme an, die aufgrund Ratings und Genre am nächsten liegen.

Quelle: 1 Fix, E. & Hodges, J. Discriminatory Analysis. Nonparametric Discrimination: Consistency Properties, 1951

4. CASE STUDY

CASE STUDY IN HAUSARBEIT (FREIWILLIG).

1. Erstellen Sie ein Notebook in Google Colab.
2. Laden Sie die Standard-Bibliotheken (analog Beispiel-Notebook „VL 5 Unsupervised Learning-Update“).
3. Laden Sie den Datensatz per Pandas-Funktion read_csv(), der Datensatz ist verfügbar unter: Link.
4. Wenden Sie die gelernten Verfahren KMeans und K-Nearest-Neighbours auf den Datensatz an.
   Welche Ergebnisse können Sie daraus generieren?
5. Nutzen Sie die Ellbow-Methode um die „ideale“ Anzahl von Klassen für KMeans zu finden.
6. Nutzen Sie die gelernten Verfahren der explorativen Datenanalyse, um selber Muster zu entdecken.
7. Vergleichen Sie die Ergebnisse der eingesetzten Lernverfahren mit Ihren Ergebnissen der Datenanalyse.
8. Stellen Sie Ihre Ergebnisse, Hypothesen und Plots vor.

~60 Minuten

ZUSAMMENFASSUNG UNSUPERVISED LEARNING.

Vorteile:

- Einfach anwendbar.
- Kein zeitaufwendiges Labeln notwendig.
- Entdecken unbekannter Zusammenhänge.

Nachteile:

- Bewertung Ergebnisse durch Experten notwendig.
- Kein eigenständiges Ableiten von Handlungen.

LITERATUR UND WEITERE QUELLEN (AUSZUG).

Künstliche Intelligenz:

- Burkov: The Hundred-Page Machine Learning Book, online verfügbar unter Link
- Nielsen: Neural Networks and Deep Learning, online verfügbar unter Link
- Russel, Norvig: Artificial Intelligence – a modern approach
- Bishop: Pattern Recognition and Machine Learning
- Geron: Hands-on Machine Learning with SciKit-Learn, Keras and TensorFlow
- Produktentwicklung mit AI:
  - Ameisen: Building Machine Learning Powered Applications: Going from Idea to Product
  - Ng: Machine Learning Yearning, online verfügbar unter Link

Kostenfreie Online-Kurse (bei Interesse):

- Python-Kurse
  - Python for Everybody (Link)
  - Udacity Python Course (Link)
  - **Coursera Course Deep Learning** (Link)
  - FAST AI (Link)