DataCraft

3.5 KI & Machine Learning

DataCraft

5
Clustering: K-means

Clustering: Der K-Means Algorithmus

DataCraft

- Der **K-Means Algorithmus** sortiert Datenpunkte in k Gruppen ein, ist also ein Clusteringalgorithmus und gehört zum unüberwachtem Lernen

- Die Idee ist, die Einteilung so zu wählen, dass die Summe der quadrierten Abweichungen vom Clusterschwerpunkt minimal ist (analog zur linearen Regression)

- Am häufigsten wird der iterative Lloyd-Algorithmus für die approximative Lösung verwendet
  1. Wähle zufällig k Werte als Schwerpunkte
  2. Ordne jeden Datenpunkt dem Cluster zu, bei dem sich die Streuung am wenigsten erhöht
  3. Berechne die Schwerpunkte neu und wiederhole Schritte 2-3

- Voraussetzungen
  - Es können nur numerische Attribute verwendet werden, da der Mittelwert berechnet wird
  - Die Anzahl Cluster k muss vorher bekannt sein. Das optimale k kann aber auch durch Vergleich von verschiedenen Werten bestimmt werden
  - Die Cluster sollten ungefähr gleich viele Punkte enthalten
  - Der Algorithmus ist sensitiv gegenüber Ausreißern (da Mittelwertberechnung)

Scikit-Learn hat ein Untermodul cluster, das mehrere Clustering-Algorithmen implementiert, natürlich auch den K-Means Algorithmus.

```
import sklearn.cluster as skcluster

km = skcluster.KMeans(n_clusters=5)
km.fit(df)

# zugeordnete Clusternummer ergänzen
df["cluster"] = km.predict(df)
```

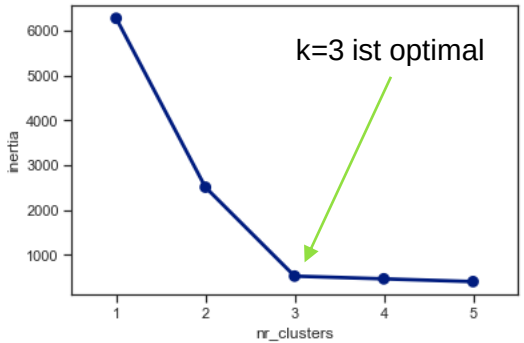
Clustering: Der K-Means Algorithmus – Anzahl Cluster

Die "beste" Anzahl Cluster lässt sich durch eine Grafik bestimmen. Dazu lässt man den Algorithmus mit verschiedenen Anzahl Clustern durchrechnen und bestimmt die sogenannte Inertia (Trägkeit).
Dann zeichnet man ein Diagramm. Das K, bei dem der Knick in der Linie ist, ist das optimale.

```
max_clusters = 5
clusters = pd.DataFrame({"nr_clusters":range(1, max_clusters+1), "inertia":None})

for i, row in clusters.iterrows():
km = skcluster.KMeans(n_clusters=row["nr_clusters"])
km.fit(df[["x","y"]])
clusters.loc[i, "inertia"] = km.inertia_

sns.pointplot(data=clusters, x="nr_clusters", y="inertia")
```



Clustering: Der K-Means Algorithmus – Anzahl Cluster

Inertia misst, wie gut ein Datensatz durch K-Means geclustert wurde. Sie wird berechnet, indem der Abstand zwischen jedem Datenpunkt und seinem Cluster-Schwerpunkt berechnet, dieser Abstand quadriert und diese Quadrate summiert werden.

Ein gutes Modell ist eines mit geringer Inertia und einer geringen Anzahl von Clustern (K). Dies ist jedoch ein Kompromiss, da mit zunehmendem K die Inertia sowieso abnimmt. Daher sucht man nach dem Knick, da ab da die Verbesserung nur minimal ist.