Kapitel 4
K-Means Clustering

Die Methode *K-Means Clustering* ist ein Beispiel für unüberwachtes Lernen. Wir haben eine Menge an Datenpunkten, die durch Vektoren repräsentiert werden. Jeder Eintrag im Vektor entspricht einem Merkmal. Aber diesen Datenpunkten ist keine Markierung, Bezeichnung oder Klasse *a priori* zugeordnet. Unser Ziel ist es, diese Datenpunkte auf eine vernünftige Art und Weise zu gruppieren. Jede Gruppe wird dabei mit einem Massenschwerpunkt assoziiert. Aber wie viele Gruppen gibt es, und wie finden wir deren Massenschwerpunkte?

4.1 Wofür können wir die Methode verwenden?

K-Means Clustering wird eingesetzt, um unmarkierte Datenpunkte anhand von Merkmalsähnlichkeiten zu gruppieren. Beispiele sind:

- Klassifizierung von Kunden nach ihrer Einkaufshistorie. Jedes Merkmal kann dabei unterschiedliche Warentypen repräsentieren.
- optimale Platzierung von Parkplätzen bzw. Parkhäusern in Städten
- Optimierung der Kragenweite und Armlänge von Hemden
- Gruppierung ähnlicher Bilder, ohne sie vorher klassifiziert zu haben

K-Means Clustering (KMC) ist eine ziemlich einfache Methode, um individuelle Datenpunkte zu einer Sammlung von Gruppen oder Clustern zuzuordnen. Welchem Cluster ein Datenpunkt zugeordnet wird, entscheidet im Wesentlichen die Distanz zum Massenschwerpunkt. Da aber die Maschine entscheidet, zu welcher Gruppe ein Datenpunkt gehört, ist das ein Beispiel für unüberwachtes Lernen. KMC ist eine sehr bekannte und populäre Clustering-Methode. Sie ist höchst intuitiv und schön zu visualisieren, außerdem ist sie sehr leicht zu programmieren. Diese Technik kann in verschiedenen Situationen eingesetzt werden:

- **Suche nach Strukturen in einem Datensatz.**
  Sie haben unklassifizierte Daten, aber Sie vermuten, dass die Daten auf natürliche Art und Weise in verschiedene Kategorien fallen.

Möglicherweise haben Sie Daten zu Automodellen, wie Preis, Treibstoffeffizienz, Radgröße, Lautsprecherleistung etc. Sie meinen, dass es da zwei natürliche Cluster geben müsste: Autos, die geschmacklosen Menschen gefallen, und Autos, die jedem anderen gefallen. Die Daten könnten so aussehen wie in Abbildung 4.1 dargestellt. Dort sehen Sie zwei offensichtliche Gruppen.

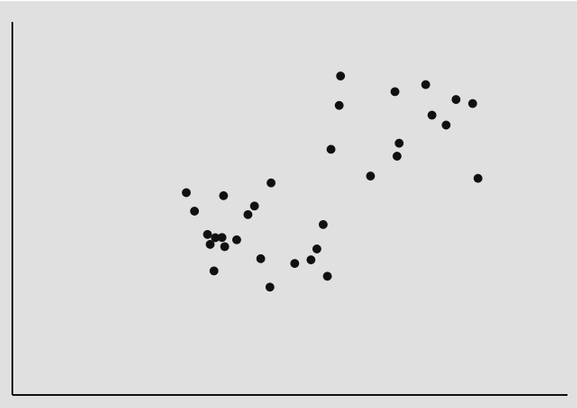

Abbildung 4.1 Zwei offensichtliche Gruppen

- **Unterteilung von Daten mit nicht offensichtlicher Gruppierung.**
  Die Daten in Abbildung 4.2 mögen auf der horizontalen Achse Familieneinkommen darstellen, während die vertikale Achse die Anzahl der Personen im Haushalt zeigt.

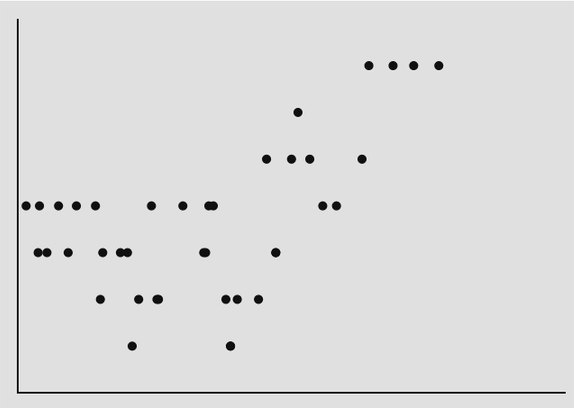

Abbildung 4.2 Wie würde man diese Daten gruppieren?

Ein Hersteller von Besteck möchte gerne wissen, wie viele Messer und Gabeln er in die Auslage geben soll und wie nobel sie aussehen dürfen/sollen. Es mag sein, dass es keine offensichtliche Gruppierung gibt, aber das ist nicht unbedingt sehr wichtig.

4.2 Was macht K-Means Clustering?

Gegeben sei ein Datensatz mit N Datenpunkten, wobei jeder mit einem M-dimensionalen Vektor mit M Merkmalen assoziiert ist, also $\mathbf{x}^{(n)}$ für $n = 1$ bis N. Jeder Eintrag im Vektor steht für eine andere numerische Größe. Jeder Vektor könnte zum Beispiel einen individuellen Haushalt repräsentieren, mit dem Einkommen als erstem Eintrag, der Anzahl der Autos als zweitem, ..., der Anzahl der Waffen als M-tem. Wir werden diese Datenpunkte in K Cluster gruppieren. Wir wählen eine Zahl für K, sagen wir 3. Also werden wir drei Cluster generieren. Jeder dieser drei Cluster wird einen Massenschwerpunkt haben, also einen Punkt im M-dimensionalen Merkmalsraum. Und jeder dieser N Datenpunkte wird dem nächsten Massenschwerpunkt zugeordnet. (Wie Häuser und Postkästen. Der Postkasten ist eine Art Massenschwerpunkt für das Cluster. Und jedes Haus wird mit einem Postkasten assoziiert.) Das Ziel dieser Methode ist es, die besten Positionen für die Massenschwerpunkte der Cluster zu finden. (Und daher könnte man KMC auch nutzen, um die besten Postkastenpositionen zu eruieren.)

In Abbildung 4.1 gibt es ganz klar zwei unterscheidbare Gruppen. Würde ich Sie bitten, jene Punkte einzuzeichnen, die die Massenschwerpunkte bilden, wo würden Sie diese Punkte hinzeichnen? In diesem Beispiel ist das relativ einfach und könnte manuell ausgeführt werden. Wie ist das aber in zwei, drei oder mehr Dimensionen? Nicht so einfach. Da kommt dann die Maschine ins Spiel.

Mathematisch gesehen wollen wir die *Intra-Cluster-* (oder *In-Cluster-*)Varianz minimieren. Diese Intra-Cluster-Varianz ist nur ein Maß dafür, wie weit jeder Datenpunkt vom nächsten Schwerpunkt entfernt ist. (Wie weit ist das Haus vom nächsten Postkasten entfernt?) Typischerweise variiert man K und sieht, welchen Effekt das auf die Distanzen hat. Der Algorithmus selbst ist wirklich einfach. Zuerst müssen wir zufällig Schwerpunkte wählen, und dann nähern wir uns schrittweise einer Konvergenz.

Schritt für Schritt: Der K-Means-Algorithmus

0 *Skalierung*

Wie bereits in Kapitel 2 erklärt, skalieren wir zuerst unsere Daten, da wir Distanzen messen müssen. Jetzt starten wir mit dem iterativen Teil unseres Algorithmus.

1 *Wähle Massenschwerpunkt*

Wir müssen den Algorithmus mit Massenschwerpunkten für die K Cluster versehen. Entweder startet man mit beliebigen K der N Vektoren, oder man erzeugt K zufällige Vektoren.

Im letzteren Fall sollten diese zufälligen Vektoren die gleichen Größeneigenschaften haben wie die skalierten Datenpunkte, entweder in Form von Mittelwert und Standardabweichung oder von Minimum und Maximum der Merkmale. Wir nennen diese Schwerpunkte oder Centroide $\mathbf{c}^{(k)}$ für $k = 1$ bis K. In Abbildung 4.3 werden sie als Rauten dargestellt.

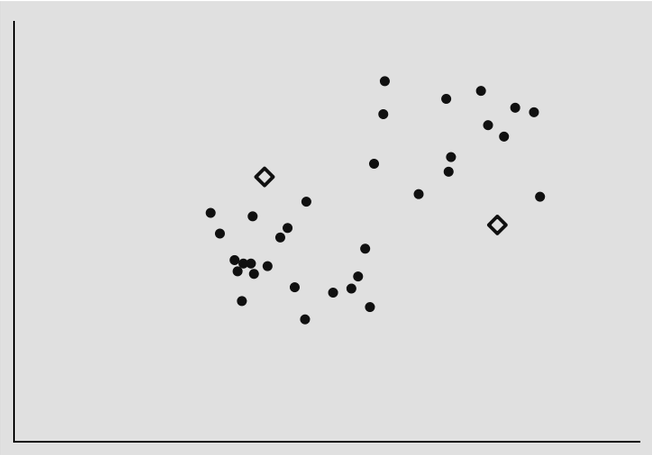

Abbildung 4.3 Initiale Schwerpunktwahl

2 *Berechne die Distanzen von jedem Datenpunkt zu den Schwerpunkten*

Für jeden Datenpunkt (Vektor $\mathbf{x}^{(n)}$) miss die Distanz von den Schwerpunkten der K Cluster. In Kapitel 2 haben wir das bereits ausführlich diskutiert. Das verwendete Distanzmaß ist natürlich problemabhängig.

Für unser Haus-/Postkastenproblem würde sich die Manhatten-Distanz anbieten (außer Sie erwarten, dass Sie des Nachbarn Hintergärten schamlos durchkreuzen). Meist wird aber die natürlichere euklidische Distanz verwendet:

$$\text{Distanz}^{(n,k)} = \sqrt{\sum_{m=1}^{M}\left(x_m^{(n)} - c_m^{(k)}\right)^2} \quad \text{für } k = 1 \text{ bis } K$$

Jeder Datenpunkt, also jedes n, wird dann dem nächsten Cluster/Schwerpunkt/Centroid zugeordnet:

$$\underset{k}{\text{argmin}}\, \text{Distanz}^{(n,k)}$$

Das kann ganz leicht folgendermaßen illustriert werden. Angenommen, K ist gleich 2, es gibt also zwei Cluster und zwei dazugehörige Schwerpunkte. Nennen wir sie das rote und das blaue Cluster. Wir nehmen den ersten Datenpunkt und messen zu beiden Schwerpunkten die Distanz, erhalten also zwei Distanzen. Die geringere der beiden ist in diesem Fall die Distanz zum blauen Schwerpunkt.

Also färben wir unseren Datenpunkt blau. Wir wiederholen diesen Vorgang für jeden Datenpunkt, sodass jeder Datenpunkt eingefärbt wird. In Abbildung 4.4 habe ich eine Linie eingezeichnet, die zwei Punktgruppen trennt.

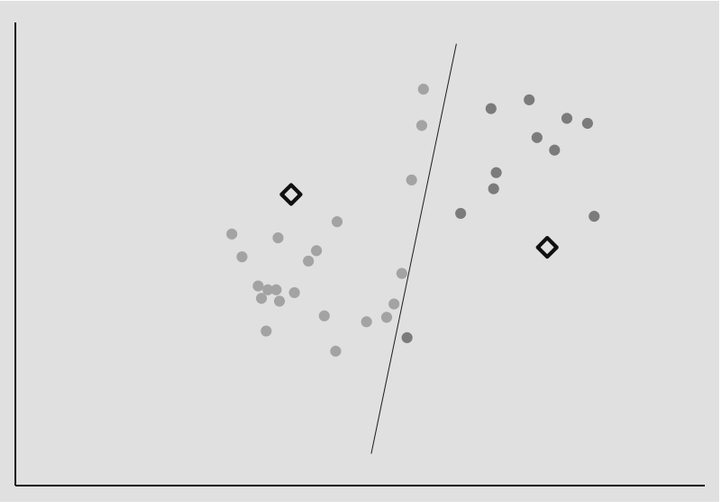

Abbildung 4.4 Zuweisung jedes Punktes zum nächsten Schwerpunkt

3 *Finde die K Schwerpunkte/Centroide*

Nun nehmen wir alle gleich markierten Datenpunkte und berechnen einen neuen Centroid, einen neuen Massenschwerpunkt. In der Farbversion berechnet man den Schwerpunkt aller roten Datenpunkte. Und das Gleiche passiert mit den blauen Punkten. Somit bestimmt man K Schwerpunkte, die zugleich die Clusterschwerpunkte für die nächste Iteration darstellen.

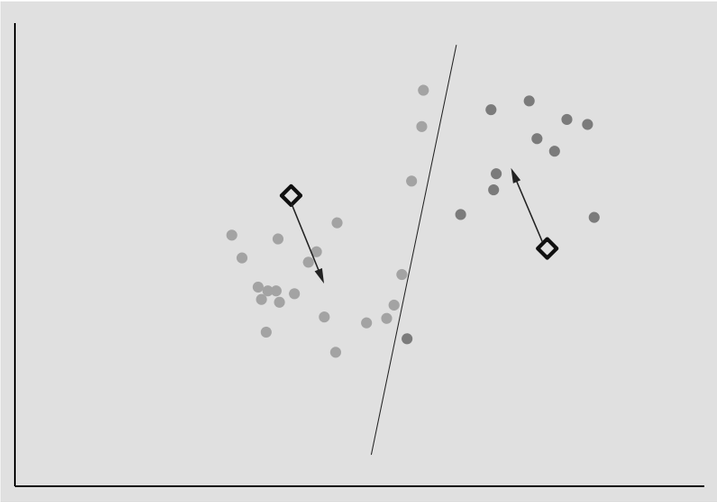

Abbildung 4.5 Aktualisiere den Schwerpunkt.

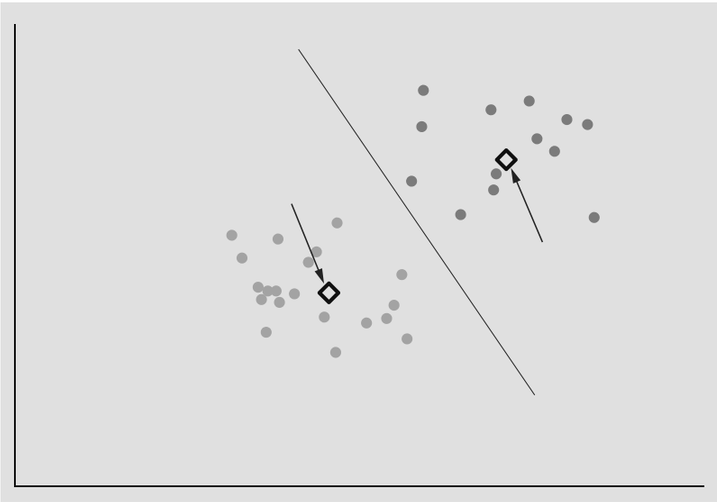

Abbildung 4.6 Wiederhole bis Konvergenz.

Zurück zu Schritt 2 und wiederhole, bis die Konvergenz erreicht ist (siehe Abbildung 4.5 und Abbildung 4.6). ■

4.3 Scree-Plots

Summiert man nun alle quadrierten Distanzen zum nächsten Cluster, dann erhalten wir ein Fehlermaß. Dieser Fehler ist eine abnehmende Funktion der Anzahl der Cluster, K. Im Extremfall mit $K = N$ erhalten wir ein Cluster pro Datenpunkt, und der Fehler ist klarerweise null.

Wenn man den Fehler gegen K aufzeichnet, dann erhält man den sogenannten *Scree-Plot*. Er wird das Aussehen von einem von zwei Typen dieses Plots annehmen, wie in Abbildung 4.7 gezeigt. Erhalten Sie die Zeichnung mit Krümmung, wo der Fehler zuerst dramatisch abfällt und plötzlich abflacht, dann lassen sich die Daten höchstwahrscheinlich schön gruppieren (siehe Dreiecke in Abbildung 4.7). In diesem Beispiel gibt es einen starken Fehlerabfall von $K = 2$ bis $K = 3$, der Abfall von $K = 3$ bis $K = 4$ ist weit weniger stark. Die passende Anzahl an Clustern ist offensichtlich; laut dieser Zeichnung ist $K = 3$.

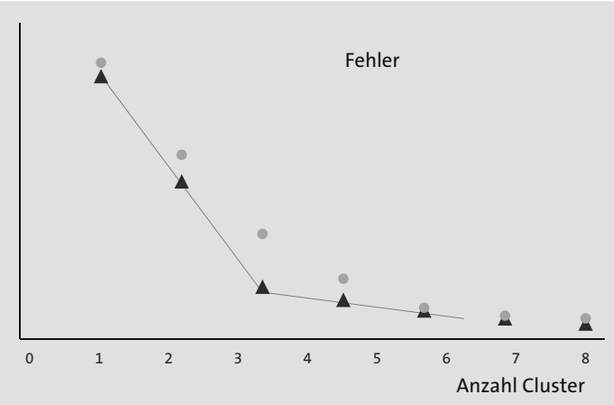

Abbildung 4.7 Zwei Wege für den Fehlerabfall. Die Dreiecke zeigen das Beispiel mit Krümmung.

Wenn der Fehler nur allmählich abnimmt (dargestellt mit Kreisen in Abbildung 4.7), dann gibt es kein offensichtliches bestes K. Hier sehen wir keinen starken Fehlerabfall mit anschließender Abflachung. Das bedeutet natürlich nicht, dass es keine natürliche Gruppierung gibt (wie zum Beispiel Daten wie in Abbildung 4.8), aber es gestaltet sich wesentlich schwieriger, die beste Gruppierung zu finden.

Konvergenz wird üblicherweise recht schnell erreicht, aber es kann natürlich passieren, dass der Algorithmus in einem lokalen Minimum landet. Um das zu verhindern, wiederholt man den Algorithmus mit verschiedenen Initialschwerpunkten.

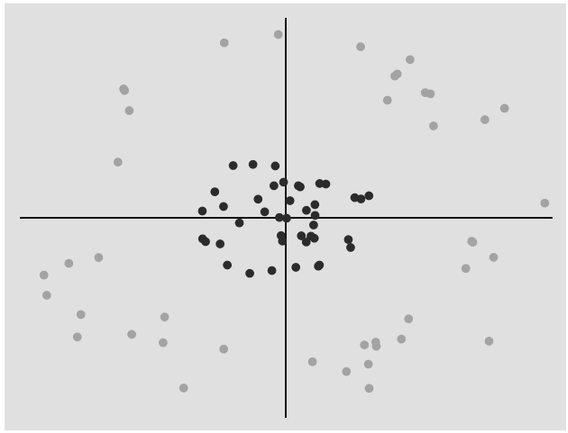

Abbildung 4.8 Es gibt ein offensichtliches Muster, aber – zum Beispiel mit einer Transformation des Koordinatensystems – schwierig zu bestimmen.

4.4 Beispiel: Kriminalität in England, 13 Dimensionen

Für unser erstes echtes Beispiel wähle ich Kriminalitätsdaten aus England. Die verwendeten Daten beziehen sich auf ein Dutzend verschiedener Verbrechenskategorien aus jeder der 342 lokalen Behörden, zusammen mit Bevölkerungszahl und Bevölkerungsdichte. Die Bevölkerungszahlen werden hier nur für die Skalierung der Anzahl an kriminellen Taten verwenden, also arbeiten wir mit 13 Dimensionen. Erwarten Sie nicht allzu viele 13-dimensionale Abbildungen. Die Rohdaten sind in Abbildung 4.9 dargestellt. Die vollständige Liste der Straftaten lautet:

- Einbruch in Betriebsgebäude
- Einbruch in Wohnobjekt
- Sachbeschädigung
- Drogenvergehen
- Betrugs- und Fälschungsdelikte
- Vergehen gegen Fahrzeuge
- andere Vergehen
- andere Diebstahlsdelikte
- Raub
- Sexualdelikt

- Gewalt gegen Person – mit Verletzung
- Gewalt gegen Person – ohne Verletzung

Distrikt	Einbruch in Betriebsgebäude	Einbruch in Wohnobjekt	Sachbeschädigung	Drogenvergehen	Betrugs- und Fälschungsdelikte	Vergehen gegen Fahrzeuge	...	Bevölkerungsanzahl	Bevölkerung pro Quadratmeile
Adur	280	120	708	158	68	382	...	58.500	3.610
Allerdale	323	126	1.356	392	79	394	...	96.100	198
Alnwick	94	33	215	25	11	71	...	31.400	75
Amber Valley	498	367	1.296	241	195	716	...	116.600	1.140
Arun	590	299	1.806	471	194	819	...	140.800	1.651
Ashfield	784	504	1.977	352	157	823	...	107.900	2.543
Ashford	414	226	1.144	196	162	608	...	99.900	446
Aylesbury Vale	696	377	1.490	502	315	833	...	157.900	453
Babergh	398	179	991	137	152	448	...	79.500	346
Barking & Dagenham	639	1.622	2.353	1.071	1.194	3.038	...	155.600	11.862
Barnet	1.342	3.550	2.665	1.198	1.504	4.104	...	331.500	9.654
Barnsley	1.332	860	3.450	1.220	322	1.661	...	228.100	1.803
Barrow-in-Furness	190	134	1.158	179	59	227	...	70.400	2.339
Basildon	756	1.028	1.906	680	281	1.615	...	164.400	3.874
Basingstoke & Deane	1.728	598	426	930	182	1.159	...	147.900	605

Abbildung 4.9 Ein Ausschnitt der Kriminalitätsdaten

Mit einer ersten, schnellen Anwendung des KCM erhalten wir folgenden Scree-Plot (siehe Abbildung 4.10).

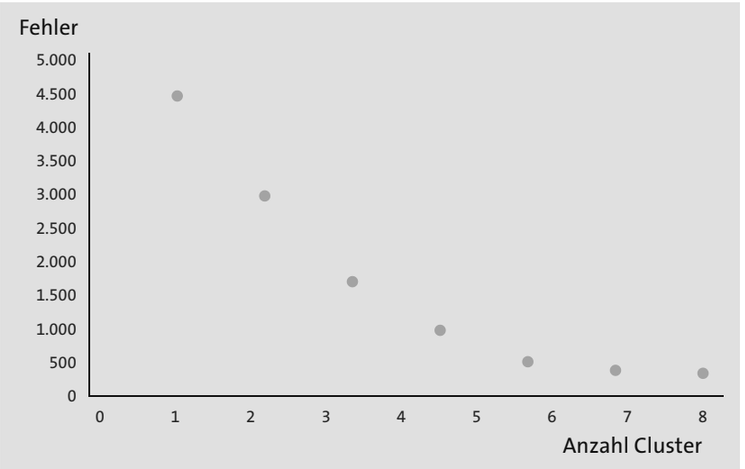

Abbildung 4.10 Scree-Plot für Kriminalität in England

Man findet eine halbwegs überzeugende Krümmung um die drei Cluster. Diese drei Cluster sind hochinteressant. Das Resultat ist in Tabelle 4.1 mit den originalen Variablen dargestellt, die Anzahl der Straftaten wurde jedoch pro Kopf der Bevölkerung skaliert.

	Cluster 1	Cluster 2	Cluster 3
Anzahl in Cluster	1	68	273
Einbruch in Betriebsgebäude	0,0433	0,0059	0,0046
Einbruch in Wohnobjekte	0,0077	0,0079	0,0030
Sachbeschädigung	0,0398	0,0156	0,0114
Drogenvergehen	0,1446	0,0070	0,0029
Betrugs- und Fälschungsdelikte	0,1037	0,0042	0,0020
Vergehen gegen Fahrzeuge	0,0552	0,0125	0,0060
andere Vergehen	0,0198	0,0018	0,0009
andere Diebstahlsdelikte	0,6962	0,0313	0,0154
Raub	0,0094	0,0033	0,0004
Sexualdelikt	0,0071	0,0015	0,0008
Gewalt gegen Person – mit Verletzung	0,0560	0,0098	0,0053
Gewalt gegen Person – ohne Verletzung	0,0796	0,0128	0,0063
Bevölkerung pro Quadratmeile	4493	10952	1907

Tabelle 4.1 Tabelle der Clusterresultate

Cluster 1 hat genau einen Punkt, und das ist der Distrikt City of London. In diesen Daten wirkt das wie ein statistischer Ausreißer, denn die Bevölkerungszahlen der einzelnen lokalen Distrikte beziehen sich auf Menschen, die dort *leben*. Und nicht viele Leute leben im Distrikt City of London. Wir können also aus dieser Analyse nicht herauslesen, wie sicher dieser Distrikt ist, da viele Straftaten wahrscheinlich Menschen betreffen, die nicht dort leben. Einbruch in Wohnobjekte wiederum ist in der City of London und in Cluster 2 recht ähnlich.

Die anderen beiden Cluster repräsentieren gefährlichere (Cluster 2) und sichere (Cluster 3) Distrikte. Und was mir besonders auffällt, als jemand, der meist im ländlichen Gebiet wohnt, ist, dass die sichersten Plätze jene mit geringer Bevölkerungsdichte sind. Puh, Glück gehabt!

Dieses Beispiel betont einen wichtigen Punkt, nämlich den Effekt der Skalierung. Obwohl grundsätzlich nichts falsch daran ist, Cluster mit geringer Anzahl an dazugehörigen Datenpunkten zu haben, könnte das hier möglicherweise problematisch sein.

Ausreißer können sich fatal auf die Skalierung auswirken. Ein großer Merkmalswert in einer kleinen Anzahl an Datenpunkten führt üblicherweise dazu, dass die Merkmale der übrigen Datenpunkte verschwindend klein werden. Das hängt natürlich auch von der Art der Skalierung ab. Wenn wir dann Distanzen messen, verliert dieses Merkmal seinen Einfluss. Für eine wissenschaftliche Arbeit würde ich in diesem Fall den Ausreißer eleminieren und die Berechnungen erneut durchführen.

Obiges Beispiel können wir durchaus als hochdimensionales Problem betrachten: 13 Merkmale heißt 13 Dimensionen im Vergleich zu relativ wenigen Trainingsdaten, nämlich 342. Man könnte also erwarten, dass wir dem Fluch der Dimensionalität, wie in Kapitel 2 erwähnt, erliegen. Die Resultate zeigen aber, dass dem glücklicherweise nicht so ist. Der Grund mag darin liegen, dass die Merkmale nicht so unabhängig voneinander sind. Diebstahl, Raub, Einbruch sind gewissermaßen recht ähnlich, während sich Betrug- und Fälschungsdelikte sehr von Sexualdelikten unterscheiden.

Wollten wir nun eine Dimensionsreduktion erreichen, bevor wir den K-Means-Algorithmus anwerfen, könnten wir die Hauptkomponentenanalyse verwenden. Oder hier auch einfach jene Merkmale weglassen, die uns nach Hausverstand recht ähnlich erscheinen.

Wechseln wir nun zu Finanz- und Wirtschaftsbeispielen. Es gibt eine Fülle solcher Daten, für die unterschiedlichsten finanziellen und wirtschaftlichen Größen und in riesigen Mengen. Für manche muss man zahlen, wie die sogenannten Tick-Daten, die für Hochfrequenz-Handeln verwendet werden, aber meine Beispieldaten sind alle gratis zu haben.

Warnung

Ich werde jetzt Zeitreihendaten verwenden, für die der K-Means-Algorithmus nicht unbedingt die erste Wahl ist, denn die Zeitkomponente wird nicht verwendet. Trotzdem werden wir in den nun folgenden Beispielen zu einigen interessanten Ergebnissen kommen.

4.5 Beispiel: Volatiliät

Natürlich können wir uns Problemen beliebiger Dimensionalität widmen. Also beginnen wir mit einem eindimensionalen Beispiel aus dem Finanzwesen.

Dieses Problem verwendet Finanzdaten, wie viele Beispiele in diesem Buch. Finanzdaten sind besonders leicht zu beschaffen. Sie finden beispielsweise viele Aktien- und Indexkurse unter *finance.yahoo.com*. Zunächst verwenden wir den Index S&P 500 (Standard & Poor's 500). Ich lade ihn zunächst herunter, und zwar bis ins Jahr 1950 zurückreichend. Daraus berechnen wir eine 30-Tages-Volatilität mittels Excel. Wenn Sie aus dem Finanzbereich kommen, dann wissen sie genau, wovon ich spreche. Wenn nicht, dann ist das hier nicht der richtige Ort, um zu sehr ins Detail zu gehen. (Ich könnte Ihnen natürlich die Titel einiger exzellenter Bücher zu diesem Thema nennen.) Hier reicht es, dass man aus der Spalte der täglichen Schlusskurse des S&P 500 eine Spalte mit den täglichen Erträgen berechnet und dann die Standardabweichung aller Tageserträge über ein gewisses Zeitfenster. Volatilität ist dann nur eine skalierte Version dieser Zahl. In Abbildung 4.11 sehen Sie eine bildhafte Darstellung der Volatilität.

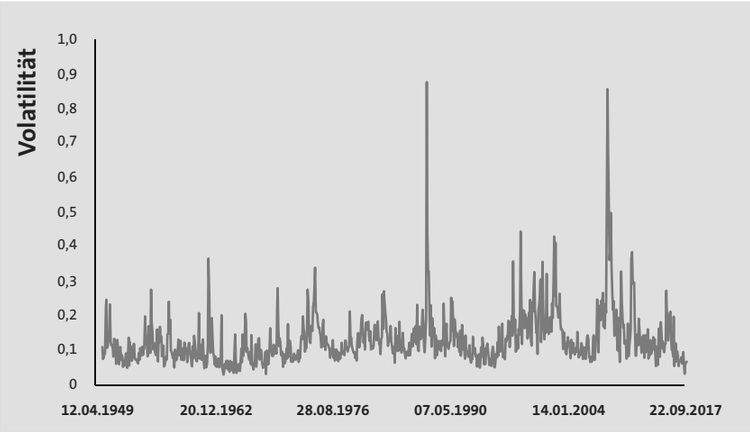

Abbildung 4.11 30-Tage SPX Volatilität

Sie erinnern sich, dass die KMC-Analyse die Zeitabhängigkeit des Volatilitätsverhaltens ignoriert. Und doch gibt es durchaus eine Motivation, K-Means-Clustering auf diese Daten anzuwenden. Und zwar gibt es in der Finanzwelt ein Modell, nach dem die Volatilität von einer Ebene zur anderen springt, von Regime zu Regime. Die Volatilität in unserer Abbildung scheint so ein Verhalten zu zeigen. Sie sehen, dass die Volatilität sehr oft niedrig, manchmal im mittleren Bereich und vereinzelt ziemlich hoch ist.

In unserem Plot ist das nicht ganz der Fall. Hier haben wir es mit einer kontinuierlichen Veränderung der Volatilitäts-Ebenen zu tun. Aber darum kümmern wir uns mal nicht. Wir starten mit drei Clustern, $K = 3$.

An dieser Stelle folgt wieder der Hinweis, dass das nur ein illustratives Beispiel ist. Lassen Sie uns loslegen. Wir werden diese drei Cluster finden und dann das Modell etwas weiter entwickeln. Der Algorithmus findet sehr schnell die Cluster aus Tabelle 4.2. Wie sich die Volatilität von Cluster zu Cluster bewegt, sieht man in Abbildung 4.12.

	Cluster 1	Cluster 2	Cluster 3
Anzahl in Cluster	586	246	24
SPX-Volatilität	9,5 %	18,8 %	44,3 %

Tabelle 4.2 Volatilitäts-Cluster im Index S&P 500

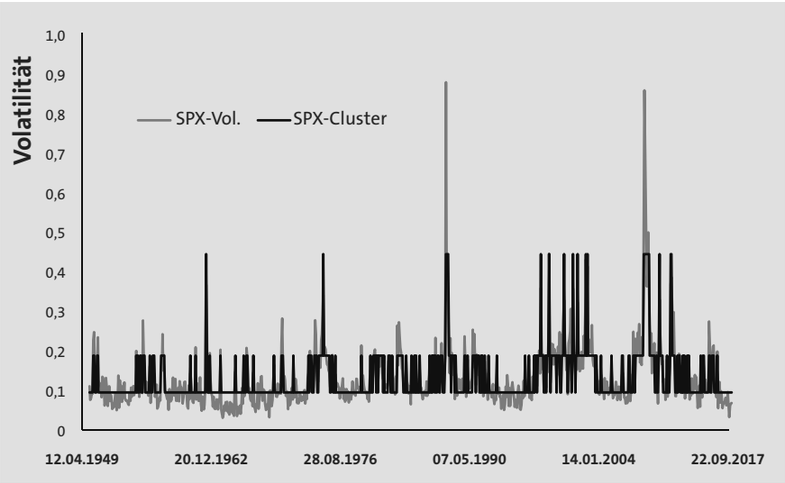

Abbildung 4.12 Die originale Volatilität und ihre Cluster

Wir können noch einen Schritt weitergehen, indem wir eine Vorstellung von der Wahrscheinlichkeit dafür entwickeln, von einem Volatilitätsregime zum nächsten zu springen. Und hier kommt auf raffinierte Weise eine einfache, wenn auch schwache Zeitabhängigkeit wieder mit ins Spiel. Die Wahrscheinlichkeitsmatrix aus Tabelle 4.3 lässt sich sehr leicht bestimmen. Wir interpretieren das so, dass die Sprungwahrscheinlichkeit von Cluster 1 zu Cluster 2 bei 16 % alle 30 Tage liegt.

von/zu	Cluster 1	Cluster 2	Cluster 3
Cluster 1	84 %	16 %	0 %
Cluster 2	38 %	57 %	5 %
Cluster 3	0 %	54 %	46 %

Tabelle 4.3 Übergangswahrscheinlichkeit für ein Sprungvolatiliätsmodell

4.6 Beispiel: Zinssatz und Inflation

Um bei einem finanziellen oder eher wirtschaftlichen Thema zu bleiben, sehen wir uns mal Zinssätze und Inflationsdaten an. Die Motivation dahinter ist, dass Zentralbanken angeblich die Zinssätze benutzen, um die Inflation zu kontrollieren. Aber auch dieses Beispiel dient nur der Illustration. (Irgendwann werde ich aufhören, mich diesbezüglich zu wiederholen.) Selbstverständlich ist die Zeitkomponente für diese beiden Variablen durchaus wichtig, aber ich nehme mir wieder die Freiheit, sie für diese KMC-Analyse zu ignorieren. Ich habe Daten für das Vereinigte Königreich gefunden, die Zinssätze und Inflation bis zurück in das Jahr 1751 enthalten (siehe Abbildung 4.13).

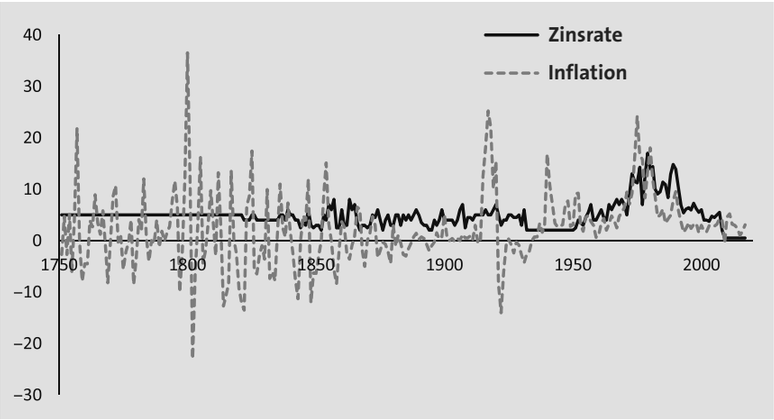

Abbildung 4.13 Inflation und Zinssätze seit dem Jahr 1751

Es ist unwahrscheinlich, dass etwas so Einfaches wie K-Means Clustering recht gut mit all diesen Daten umgeht, deshalb habe ich mich auf die Daten ab 1945 beschränkt. In Abbildung 4.14 habe ich die Inflation gegen die Zinssätze aufgetragen. Die Punkte habe

ich verbunden, um die Entwicklung der Daten hervorzuheben, aber klarerweise wird das beim einfachen K-Means-Algorithmus ignoriert.

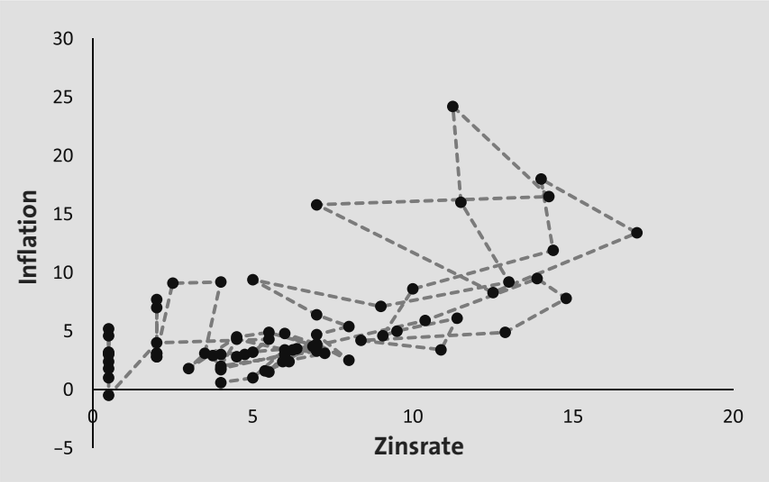

Abbildung 4.14 Inflation versus Zinssatz, zeitlich verbunden

Auch wenn Zins- und Inflationswerte in etwa in der gleichen Größenordnung liegen, habe ich sie dennoch zuerst translatiert und dann skaliert. Mit vier Clustern kommen wir zum Ergebnis, das Tabelle 4.4 zeigt.

	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Anz. im Cluster	25	30	11	7
Beschreibung	sehr niedriger Leitzins, normale Inflation	»Normal-Wirtschaft«	hoher Leitzins, mittlere Inflation	hoher Leitzins, hohe Inflation
Zinssatz	2,05 %	6,15 %	11,65 %	12,77 %
Inflation	3,21 %	3,94 %	6,89 %	16,54 %

Tabelle 4.4 Cluster in Zinssätzen und Inflation (in originaler Skalierung)

Die Clusterschwerpunkte und die Originaldaten sind in Abbildung 4.15 dargestellt. Ich habe die skalierten Größen mit gleich großer Achsenlänge gezeichnet, damit die Cluster-Trennlinien leichter zu sehen (und zu zeichnen) sind.

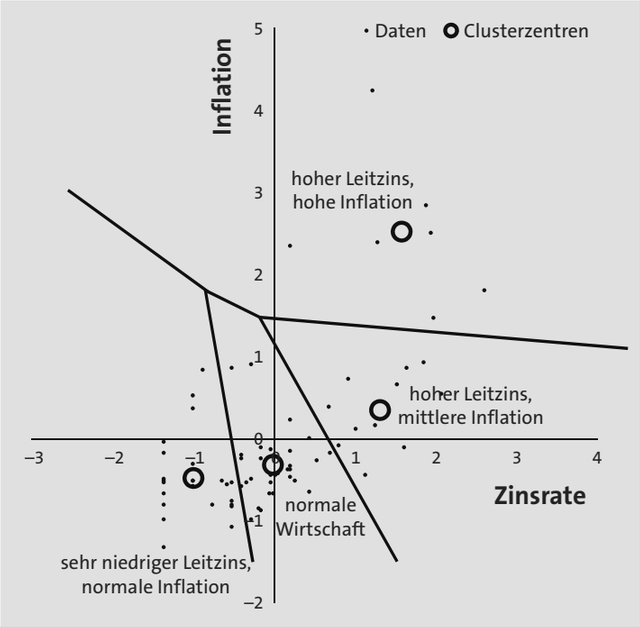

Abbildung 4.15 Inflation versus Zinssatz mit den gefundenen vier Clustern (skalierte Größen). Das ist ein Voronoi-Diagramm (siehe Abschnitt 4.8).

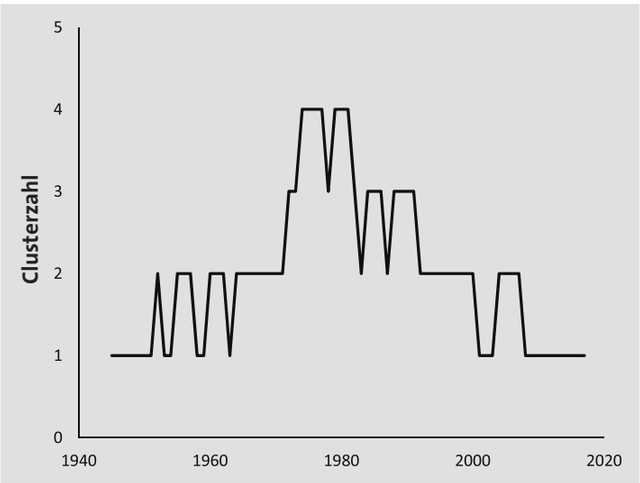

Abbildung 4.16 Evolution der Cluster

Wieder können wir die Sprungwahrscheinlichkeiten zwischen den Clustern berechnen. Die Ergebnisse stehen in Tabelle 4.5.

von/zu	**Cluster 1**	**Cluster 2**	**Cluster 3**	**Cluster 4**
Cluster 1	88,9 %	11,1 %	0,0 %	0,0 %
Cluster 2	5,6 %	86,1 %	8,3 %	0,0 %
Cluster 3	0,0 %	25,0 %	58,3 %	16,7 %
Cluster 4	0,0 %	0,0 %	33,3 %	66,7 %

Tabelle 4.5 Clustersprung-Wahrscheinlichkeiten

Interessant wäre folgendes Experiment, in dem man die Daten zufällig mischt, wobei die Zinssätze gleich bleiben, während die Inflationsdaten zufällig neu sortiert werden. Gäbe es zu Beginn irgendeine Struktur in den Daten, dann wäre sie durch das Mischen definitiv verloren.

4.7 Beispiel: Zinssätze, Inflation und BIP-Wachstum

Nehmen wir die Daten des vorherigen Beispiels und erweitern sie um das BIP-Wachstum (BIP = Bruttoinlandsprodukt). Dann haben wir ein dreidimensionales Problem. (Und ich verwende dafür vierteljährliche Daten.)

Mit sechs Clustern kam ich zu dem Schluss, dass drei der vorherigen Cluster nicht viel Änderung erfahren haben, nur die normale Wirtschaft unterteilte sich weiter in drie zusätzliche Cluster. Sehen Sie sich dazu Abbildung 4.17 an, die das Resultat in zwei Dimensionen darstellt. Die Cluster sind in Tabelle 4.6 zu sehen.

	Cluster 1	**Cluster 2**	**Cluster 3**	**Cluster 4**	**Cluster 5**	**Cluster 6**
Anzahl im Cluster	31	107	23	31	17	38
Zinssatz	0,50 %	5,77 %	5,89 %	7,44 %	12,11 %	12,56 %
Inflation	2,24 %	2,37 %	5,59 %	5,17 %	18,89 %	6,18 %
BIP Wachstum	0,48 %	0,71 %	–0,63 %	2,25 %	–0,51 %	0,33 %

Tabelle 4.6 Cluster für Zinssatz, Inflation und BIP-Wachstum

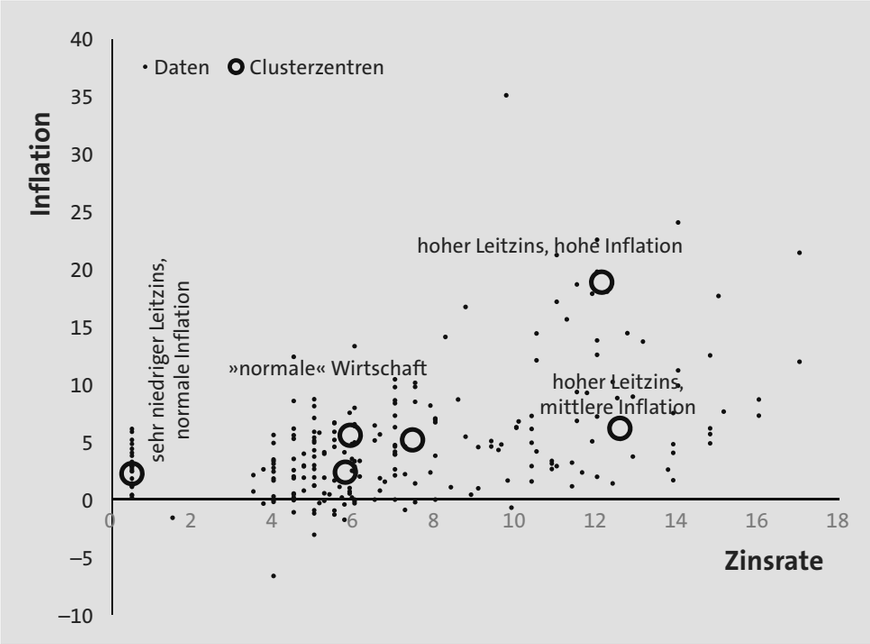

Abbildung 4.17 Inflation versus Zinssatz (die Achse für das BIP-Wachstum würde aus dem Buch heraus zum Leser zeigen; originale Skalierung).

4.8 Ein paar Kommentare

- **Voronoi-Diagramme:** Abbildung 4.15 ist ein Voronoi-Diagramm. Das ist die Aufteilung eines Raumes in Regionen, abhängig von einer Distanz (meist der euklidischen Distanz, aber nicht unbedingt), basierend auf einer gegebenen Punktemenge. Die Verwendung von Voronoi-Diagrammen geht auf Descartes im 17. Jahrhundert zurück. Ein hochinteressanter Anwendungsfall ist von John Snow (nein, nicht der aus »Game of Thrones«) aus dem Jahre 1854 zu berichten. Er konnte damit zeigen, dass die meisten Menschen, die während eines Choleraausbruchs gestorben sind, in der Nähe einer bestimmten Wasserpumpe lebten.
- **Wahl von K:** Mit etwas Glück hilft Ihnen der Scree-Plot bei der Bestimmung der optimalen Zahl an Clustern. Möglicherweise ist die Zahl naheliegend und ergibt sich aus der Natur des Problems. Nicht ganz so überzeugend, aber es nützt der Glaubwürdigkeit, wenn die Cluster einen Namen bekommen, wie im ersten Zinssatzbeispiel.

- Für eines der Beispiele am Beginn des Kapitels müssen Sie eine kleine Änderung vornehmen, speziell in Bezug auf die Kragenweite von Hemden. Erraten Sie, was zu tun ist? (Hinweis: Mit einer Kragenweite von 43 cm tun Sie sich schwer, ein 38-cm-Hemd zu tragen.)

4.9 Weiterführende Literatur

Für eine Implementierung des KMC mit der Programmiersprache R siehe *»Applied Unsupervised Learning with R: Uncover previously unknown patterns and hidden relationships with k-means clustering, heirarchical clustering and PCA«* von Bradford Tuckfield, veröffentlicht durch Packt in 2019.

Es gibt viele Bücher aus der »Für Dummies«-Reihe, die verschiedene Aspekte des maschinellen Lernens abdecken. KMC findet sich in *Predictive Analytics For Dummies, 2nd Edition* von Anasse Bari, veröffentlicht im John Wiley & Sons Verlag in 2016.

Möchten Sie mehr über das Modellieren der Volatilität als Sprungprozess erfahren, dann gehen Sie zu *Ito33.fr* und laden deren technische Publikationen herunter.

Wie versprochen kann ich ein paar exzellente Bücher zur quantitativen Finanzmathematik empfehlen, von ganzem Herzen zum Beispiel *Paul Wilmott Introduces Quantitative Finance*, *Paul Wilmott On Quantitative Finance* (in drei unglaublichen Bänden!) und *Frequently Asked Questions in Quantitative Finance*. Alle von mir, versteht sich, und veröffentlicht durch John Wiley & Sons.