

Seminararbeit

Analyse und Konzeption von Clusteralgorithmen für Entscheidungen im Produktionsnetzwerk

Nikolai Jarre

Matrikelnummer: 3277381

1. Prüfer: Prof. Dr. rer. nat. Phillip Rohde
2. Prüfer: Tino X. Schlosser, M.Sc. RWTH

Studiengang angewandte Mathematik und Informatik

Aachen, den 15.12.2022

Inhalt und Ergebnis dieser Arbeit sind ausschließlich zum internen Gebrauch bestimmt. Ohne ausdrückliche Genehmigung des betreuenden Lehrstuhls ist es nicht gestattet, diese Arbeit oder Teile daraus an Dritte weiterzugeben.

Eidesstattliche Erklärung

Hiermit versichere ich, dass ich die Seminararbeit mit dem Thema

*Analyse und Konzeption von Clusteralgorithmen für Entscheidungen im
Produktionsnetzwerk*

selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, alle Ausführungen, die anderen Schriften wörtlich oder sinngemäß entnommen wurden, kenntlich gemacht sind und die Arbeit in gleicher oder ähnlicher Fassung noch nicht Bestandteil einer Studien- oder Prüfungsleistung war.

Ich verpflichte mich, ein Exemplar der Seminararbeit fünf Jahre aufzubewahren und auf Verlangen dem Prüfungsamt auszuhändigen.

Name: Nikolai Jarre

Aachen, den 15. Dezember 2022

A handwritten signature in grey ink that reads "Nikolai Jarre". The signature is written in a cursive, flowing style.

Unterschrift der Studentin / des Studenten

Abstract

Das aktuelle Marktumfeld ist aufgrund von Krisen und raschen Technologieentwicklungen immer mehr von Volatilität gekennzeichnet. Ein Faktor, um als global produzierendes Unternehmen in diesem volatilen Marktumfeld dem Wettbewerbsdruck standzuhalten und langfristig erfolgreich zu sein, ist die Entwicklung eines gezielten Netzwerkmanagements.

Produktionsnetzwerke bestehen aus verschiedenen Standorten, bei denen Entscheidungen oft dezentral auf Standortebene getroffen werden. Die zunehmende Digitalisierung ermöglicht es, Entscheidungen auch übergeordnet für mehrere Standorte zu treffen. Dadurch lassen sich Skalen- sowie Lerneffekte nutzen und somit bessere Entscheidungen für das gesamte Unternehmen treffen.

Dabei unterscheiden sich sowohl die Entscheidungen in ihren verschiedenen Charakteristiken als auch die Produktionsstandorte in ihren Eigenschaften und Fähigkeiten. Für die optimale Verteilung der Entscheidungsbefugnisse innerhalb des Produktionsnetzwerkes müssen die Standorte unter Rücksicht ihrer Eigenschaften und Fähigkeiten gruppiert bzw. geclustert werden.

Im Zuge dieser Arbeit soll untersucht werden, welche Heuristik sich für eine Clusterbildung hinsichtlich dieser Thematik eignet, um größtmögliche Eigenschaftscluster über die Standorte hinweg zu identifizieren.

Inhaltsverzeichnis

1. Einleitung.....	5
1.1 Ausgangssituation und relevanter Praxiszusammenhang	5
1.2 Zielsetzung.....	5
1.3 Aufbau der Arbeit	5
2. Theoretische Grundlagen	7
2.1 Produktionsnetzwerke	7
2.2 Entscheidungen in Produktionsnetzwerken.....	7
2.3 Herausforderungen an Clusterungen von Entscheidungen	8
3. Clusteralgorithmen.....	10
3.1 Arten der Clusteralgorithmen	10
3.2 Analyse und Bewertung hierarchischer Clusteralgorithmen	15
4 Konzept.....	17
5 Zusammenfassung und Ausblick	23
5.1 Bewertung der Methodik	23
5.2 Kritische Reflexion und Ausblick.....	24
Literaturverzeichnis.....	26

Abbildungsverzeichnis

Abbildung 1: Clusterung mittels K-Means Algorithmus	11
Abbildung 2: Clusterung mittels dichte-basiertem DBSCAN-Algorithmus	12
Abbildung 3: Hierarchische Genom-Clusterung	14
Abbildung 4: Clusterung mittels Single Linkage Algorithmus.....	15
Abbildung 5: Auffassung i. von Ähnlichkeit	19
Abbildung 6: Auffassung ii. von Ähnlichkeit	19
Abbildung 7: Verschiedene Distanzmaße.....	20
Abbildung 8: Laufzeitkomplexität verschiedener Clusteralgorithmen.....	23
Abbildung 9: Laufzeitkomplexität verschiedener Clusteralgorithmen.....	23
Abbildung 10: Laufzeitkomplexität verschiedener Clusteralgorithmen.....	23

1. Einleitung

1.1 Ausgangssituation und relevanter Praxiszusammenhang

In der Produktion werden Produkte oft nicht nur an einem Standort gefertigt, sondern an mehreren Standorten in Teilen produziert, bearbeitet, montiert. Bei einer solchen Menge verschiedener Standorte spricht man von einem Produktionsnetzwerk. Die Standorte agieren aufgrund ihrer lokalen und standortkulturellen Abgrenzung in ihren Entscheidungsfindungen überwiegend dezentral auf Standortebene.¹ Da sie aber Teil der gleichen Wertschöpfungskette sind, kann es von Vorteil sein, bestimmte Entscheidungen zentral für mehrere Standorte zu treffen. Dadurch lassen sich Skalen- und Lerneffekte nutzen, wodurch die Entscheidungen besser getroffen werden können.² Eine solche entscheidungsorientierte Struktur bezeichnet man als Cluster: eine als einheitliches Ganzes zu betrachtende Menge bestehend aus Standorten. Die Umsetzung einer solchen Clusterbildung wird vor allem durch die Digitalisierung und der damit einhergehenden Transparenz ermöglicht.³

Eine grundlegende Problemstellung für die Clusterbildung ist das Herausfinden, wann welche Standorte ein für eine Clusterbildung ausreichendes Zentralisierungspotential aufweisen.⁴ Das Zentralisierungspotential bezieht sich dabei immer nur auf eine Entscheidung, da für eine Entscheidung immer nur bestimmte Faktoren relevant sind. Die relevanten Faktoren hängen dabei vor allem von den Eigenschaften und Fähigkeiten der Standorte ab.

1.2 Zielsetzung

Ziel der Seminararbeit ist das Formulieren einer möglichen Heuristik für die Clusterbildung. Dafür müssen zunächst die für eine Clusterbildung relevanten Faktoren klassifiziert werden. Diese Faktoren sollen anschließend mit den gegebenen Standortinformationen abgeglichen werden um pro Standort das Maß der Eignung für eine Clusterbildung festzustellen zu können.

Nach mehreren durchgeführten Clusterbildungen kann zusätzlich noch eine Bündelung von mehreren Clustern zu einem Cluster, also eine Art übergeordnetes Cluster, vorgenommen werden, um durch die daraus resultierende geringere Anzahl an Clusterstrukturen die Kosten zu senken.

1.3 Aufbau der Arbeit

Zunächst wird auf die theoretischen Grundlagen der Arbeit eingegangen, um für die anschließenden Ausarbeitungen einen verständlichen Kontext zu schaffen. Dabei sollen zuerst Produktionsnetzwerke kurz erläutert werden und anschließend die darin vorkommenden Entscheidungsprozesse geschildert werden. Die Anforderungen und Herausforderungen an diese Entscheidungscluster werden formuliert.

¹ vgl. Schuh et al. 2020 - 2020, S. 849.

² vgl. Schuh et al. 2020 - 2020, S. 850.

³ vgl. Lanza et al. 2019, S. 835.

⁴ vgl. Schuh et al. 2020 - 2020, S. 847–848.

Im dritten Kapitel werden allgemein Clusteralgorithmen vorgestellt und anschließend ausgewählte, relevante Clusteralgorithmen untersucht und bewertet. Die Bewertung bezieht sich dabei auf die formulierten Herausforderungen an Entscheidungscluster.

Anschließend soll mithilfe dieser Untersuchung bereits bestehender Algorithmen zur Clusterung von Daten eine eigene Methodik konzeptioniert werden.

Im letzten Kapitel werden die erarbeiteten Ergebnisse zusammengefasst, bewertet und Herausforderungen angesprochen.

2. Theoretische Grundlagen

2.1 Produktionsnetzwerke

Netzwerke im klassischen Sinne bestehen aus Knoten und Kanten. Die Knoten eines Produktionsnetzwerkes sind dabei die einzelnen Produktionsstandorte und die Kanten die Verbindungen zwischen diesen Produktionsstandorten.⁵ Für unseren Betrachtungsbereich ist ein Produktionsstandort die kleinste Wertschöpfung erbringende Einheit, in der die vorhandenen Prozesse fix gegeben sind. Es ist wie der Name impliziert ein geographischer Ort und kann über die reine Produktion hinaus auch Wertschöpfung schaffen durch Entwicklung, Vertrieb oder Verwaltung. Für uns relevant sind lediglich die Knoten des Produktionsnetzwerkes, also die Produktionsstandorte und ihre Eigenschaften, da diese später als Menge an Datenpunkten untersucht werden sollen.

Ein für Produktionsnetzwerke allumfassendes Ziel ist ihre Wirtschaftlichkeit, also das Maß dafür, wie effizient die Ressourcen für die Erzielung der Wertschöpfung eingesetzt werden.⁶

2.2 Entscheidungen in Produktionsnetzwerken

In der Regel sind Standorte in ihrem Vorgehen unabhängig voneinander, d.h. ein Großteil der Entscheidungen findet auf Standortebene statt, erstreckt die operativen. Entscheidungen können allerdings auch zentraler, beispielsweise auf regionaler Ebene oder im Extremfall komplett netzwerkübergreifend, getroffen werden. Diese Netzwerkebene, auf der sich die Entscheidungsgewalt befindet, definiert den Zentralisierungsgrad der Entscheidung. In Bezug auf das Ziel der Wirtschaftlichkeitssteigerung ist es vorteilhaft, den Zentralisierungsgrad so groß wie möglich zu halten um Synergien und Lerneffekte zu nutzen, beispielsweise durch das Teilen von Best Practices.⁷

Ob eine Clusterung von Entscheidungen auf einer dezentraleren Ebene als auf Standortebene möglich ist, hängt unter anderem von verschiedenen Faktoren ab, von denen nachfolgend einige beispielhaft genannt sind. Dabei ist anzumerken, dass für manche Entscheidungen nur eine bestimmte Anzahl an Faktoren relevant ist.

Der Digitalisierungsgrad der Standorte ermöglicht eine hohe Transparenz und einfachen Datenaustausch innerhalb des Produktionsnetzwerkes.⁸ Innerhalb eines Entscheidungsclusters sollte der Digitalisierungsgrad der Standorte ähnlich sein. Des Weiteren könnte man einen Mindest-Digitalisierungsgrad definieren, da nur durch einen ausreichend hohen Digitalisierungsgrad eine Zentralisierung möglich ist. Der Digitalisierungsgrad lässt sich gut mit einer Kennzahl quantifizieren und kann gut in einem Clusteralgorithmus mit berechnet werden.

Neben dem Digitalisierungsgrad kann auch die IT-Infrastruktur von Bedeutung sein, beispielsweise ob die standortinternen Daten mit einem SAP-System oder mit Excel-Tabellen erfasst werden. Die Umwandlung dieses Faktors in eine Kennzahl ist schwierig, kann aber dennoch für eine Clusterung vorgenommen werden.

⁵ vgl. Merchiers 2008, S. 19.

⁶ vgl. Varandani 2014, S. 4.

⁷ vgl. Fränken 2021, S. 125–126.

⁸ vgl. Gützlaff 2021, S. 29.

Standards können ein weiterer relevanter Faktor für Clusterungen von Entscheidungen sein. Sie sorgen durch ihre ähnlichen Strukturen für erhöhte Transparenz und ermutigen den Austausch von Informationen unter den Standorten.⁹ Ob sich dieser Faktor auch entlang eines 1-dimensionalen Spektrums mit einer Kennzahl bestücken lässt und somit für einen Clusteralgorithmus nutzen lässt hängt von dem betrachteten Standard ab.

Die Kompetenzen der Standorte können relevant sein, da sie Aufschlüsse über den Aufgabenbereich des jeweiligen Standortes geben und eine Bündelung von Standorten mit stark verschiedenen Aufgabenbereichen zu einem Entscheidungscluster sich als schwierig erweisen könnte.¹⁰ Ob sich dieser Faktor auch entlang eines 1-dimensionalen Spektrums mit einer Kennzahl bestücken lässt und somit für einen Clusteralgorithmus nutzen lässt hängt von der betrachteten Kompetenz ab.

Die Art der Kommunikation und Kultur eines Standortes ist für eine potenzielle Clusterung von Entscheidungen immer von Bedeutung.¹¹ Dieser Faktor ist abgekapselt von den bereits erwähnten zu betrachten da er in solchen Fällen die Standorte gleich von der Betrachtung ausschließt. Er muss im Vorhinein diskutiert werden und betrifft somit im Anschluss nicht die Clusterung von Entscheidungen.

2.3 Herausforderungen an Clusterungen von Entscheidungen

Bevor eine Clusterung von Entscheidungen vorgenommen werden kann, muss das vorliegende Produktionsnetzwerk und die einzelnen Produktionsstandorte analysiert und verstanden werden. Dabei sollten die Eigenschaften der Produktionsstandorte soweit möglich mit Kennwerten versehen werden. Einige dieser Eigenschaften wurden im vorigen Kapitel bereits unter dem Begriff ‚Faktoren‘ genannt. Eine solche Quantifizierung der Eigenschaften ermöglicht eine bessere Clusterung von Entscheidungen nach einem mathematischen Vorgehen. Inwiefern sich die Eigenschaften mit einem Kennwert versehen lassen, hängt von den jeweiligen Eigenschaften ab und kann sich als schwierig erweisen.

Ob sich die verschiedenen Entscheidungen gut zu einem Cluster zusammenfassen lassen, hängt davon ab, welche Faktoren für die jeweiligen Entscheidungen von Relevanz sind. Um diese relevanten Faktoren zu identifizieren muss ein gutes Verständnis von Entscheidungen in Produktionsnetzwerken vorliegen.

Wenn die wichtigen Faktoren identifiziert sind, muss weiterhin spezifiziert werden, inwiefern diese Faktoren eine Clusterung begünstigen. Da darauf abgezielt wird, innerhalb eines Clusters Standorte mit bestimmten Gemeinsamkeiten zu haben, ist davon auszugehen, dass die relevanten Faktoren einer Clusterung unter den Standorten ähnlich oder sogar gleich sein sollten.¹² Wie ähnlich diese Faktoren sich genau sein müssen, muss für eine Clusterung also im Vorhinein vorgegeben werden.

Wenn diese Herausforderungen geregelt wurden, können Clusterungen der Standorte nach Entscheidungen vorgenommen werden. Dabei ist noch unklar, **wie** ein solcher Vorgang vorgenommen werden soll. Es existieren zahlreiche datenbasierte Methoden wie beispielsweise Clusteralgorithmen oder Ansätze aus der künstlichen Intelligenz.¹³ Da diese Ansätze jedoch eine hohe Datenmenge, welche

⁹ vgl. Harre 2006, S. 102.

¹⁰ vgl. Harre 2006, S. 129.

¹¹ vgl. Ivanescu 2014, S. 73–81.

¹² vgl. Schuh 2021, S. 546.

¹³ vgl. Fränken 2021, S. 167.

die Anzahl an Standorten innerhalb eines Produktionsstandortes weit übersteigt, voraussetzt, muss ein neuer Ansatz, welcher den Anforderungen einer Clusterung von Entscheidungen entspricht, formuliert werden.

Hierfür werden im nächsten Kapitel die Clusteralgorithmen genauer betrachtet und hinsichtlich der Clusterung von Entscheidungen bewertet.

3. Clusteralgorithmen

Clusteralgorithmen sind analytische Verfahren/Programme, mit denen Ähnlichkeitsstrukturen in Datenbeständen gefunden werden können. Die gefundenen Ähnlichkeitsstrukturen werden dabei als ‚Cluster‘ bezeichnet, daher der Name Clusteralgorithmus.¹⁴ Die zahlreichen Clusteralgorithmen unterscheiden sich dabei trotz ihres gleichen Ziels der Suche nach Ähnlichkeitsstrukturen stark. Das liegt an ihren einhergehenden Einschränkungen, ihrer Definition von Ähnlichkeit und ihren mathematischen Herangehensweisen. Beispielsweise sind vor allem neue Clusteralgorithmen aktuell immer häufiger Machine-Learning-basiert.

3.1 Arten der Clusteralgorithmen

Um einen groben Überblick über die gängigen Clusteralgorithmen zu schaffen werden nun die verschiedenen Arten von Clustering-Algorithmen kurz erläutert und anschließend etwas konkreter einige der bekanntesten Algorithmen analysiert.

Generell lassen sich die Clusteralgorithmen in **5 Kategorien** unterscheiden.

1. Die kombinierten Clusteralgorithmen sind lediglich Verfahren, die sich aus mehreren der folgenden weiteren 4 Clusteralgorithmen zusammensetzen.

Was man sich darunter genauer vorstellen kann wird nach den folgenden Erläuterungen dieser 4 Clusteralgorithmen deutlich.

2. Die partitionierenden Clusteralgorithmen definieren Ähnlichkeitsstrukturen als feste Bereiche innerhalb einer Datenwolke. Dabei ist anzumerken, dass die Anzahl der gesuchten Cluster vorgegeben sein muss und dass die partitionierenden Cluster-Algorithmen iterativ vorgehen.¹⁵ Gutes Beispiele für die Anwendung partitionierender Clusteralgorithmen lassen sich im Marketing finden:¹⁶ Ein Unternehmen will Prospekte mit einigen seiner Produkte an potenzielle Kunden verschicken. Dabei sollen nur 4 verschiedene Typen von Prospekten gedruckt werden. Um den Kunden auch einen Prospekt zu verschicken, welcher die passenden Produkte für den jeweiligen Kunden enthält, kann man aus den bisherigen gesammelten Kundendaten 4 Cluster bilden, um zu identifizieren welches Cluster an Kunden zu welchem Kauf tendiert. So kann das Unternehmen gezielt eines der 4 personalisierten Prospekte an die entsprechende Kundengruppe verschicken.

¹⁴ vgl. Kirchner 2003, S. 2.

¹⁵ vgl. Xu und Tian 2015, S. 168–169.

¹⁶ vgl. Kumar et al. 2021, S. 4–5.

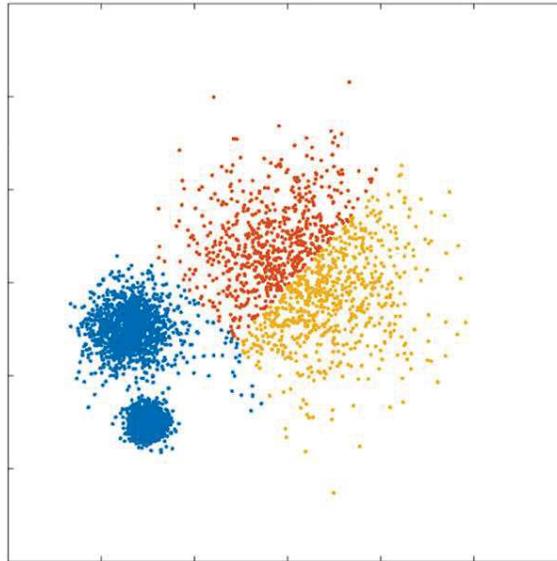


Abbildung 1: Clustering mittels K-Means Algorithmus¹⁷

Durch die Anforderung, die Anzahl an resultierenden Clustern im Vorhinein zu bestimmen, lassen sich die partitionierenden Cluster-Algorithmen kategorisch ausschließen. Durch die feste Vorgabe kann es zu Bildung von Clustern kommen, die zwar das Kriterium der gewünschten Clusterzahl entsprechen, dadurch aber relativ ungleiche Daten enthalten. Der in der Abbildung vorgenommene partitionierende Clusteralgorithmus unterteilt die vorliegenden Punkte beispielsweise in 3 Cluster, verfehlt allerdings die intuitive alternative Clustering in 3 Punkteballungen. Die Partitionierung hat hierbei also Punkteballungen korrekt identifiziert, dabei aber eine große kohärente Punkteballung in ein rotes und ein gelbes Cluster gespalten. Des Weiteren werden in partitionierenden Clusteralgorithmen stets alle Punkte geclustert, was im Kontext der Clustering von Entscheidungen ungünstig ist, da die Standorte, die in ihren Eigenschaften und Fähigkeiten stark verschieden zu allen weiteren Standorten sind, in ihrer Entscheidungsgewalt autonom bleiben müssen. Den untersten gelbe Punkt der Abbildung kann man sich als Repräsentation eines solchen Standortes vorstellen: Er ist von allen anderen Punkten und Punkteballungen weit entfernt und soll deshalb auch nicht in ein Cluster platziert werden.

3. Die dichtebasierten Clusteralgorithmen definieren Ähnlichkeitsstrukturen als Bereiche, in denen jedes Element einen möglichst geringen Abstand zu seinen Nachbarn hat.¹⁸ Anders formuliert werden Bereiche mit einer hohen Dichte gesucht. Dadurch muss im Gegensatz zu den Partitionierenden Algorithmen nicht im Vorhinein die Anzahl der Cluster vorgegeben sein. Dieser Algorithmus findet Anwendung in der Bildbearbeitung. Vor allem in unscharfen Bildern werden mithilfe des dichtebasierten Clusterings Strukturen von ihrer Umgebung abgegrenzt. Ein unscharf fotografiertes roter Tisch in einem grünen Raum beispielsweise hat aufgrund der

¹⁷ P. Raykov 2016, S. 1.

¹⁸ vgl. Xu und Wunsch 2005, S. 663.

Unschärfe keine klaren Grenzen und enthält im Randbereich grüne Pixel, kann aber durch seine rote Pixeldichte dennoch identifiziert und von seiner Umgebung abgegrenzt werden.

Obwohl man hierbei einen Maximalabstand der Datenpunkte zueinander innerhalb eines Clusters definieren kann, eignen sich dichte-basierte Clusteralgorithmen nicht für eine Clustering von Entscheidungen, da sie die ‚Gesamtdifferenz‘ der Datenpunkte innerhalb eines Clusters aus den Augen verlieren. Beispielsweise kann es sein, dass in einem dichte-basierten Cluster alle Punkte dicht beieinanderliegen, sich aber innerhalb des Clusters zwei Punkte an gegenüberliegenden Enden dieses Clusters stark unterscheiden. Im Beispiel des roten Tisches bedeutet dies, dass die Enden zweier Tischbeine demselben dichte-basierten ‚Tisch‘-Clusters angehören, ihre Positionen auf dem Bild aber stark verschieden sind.

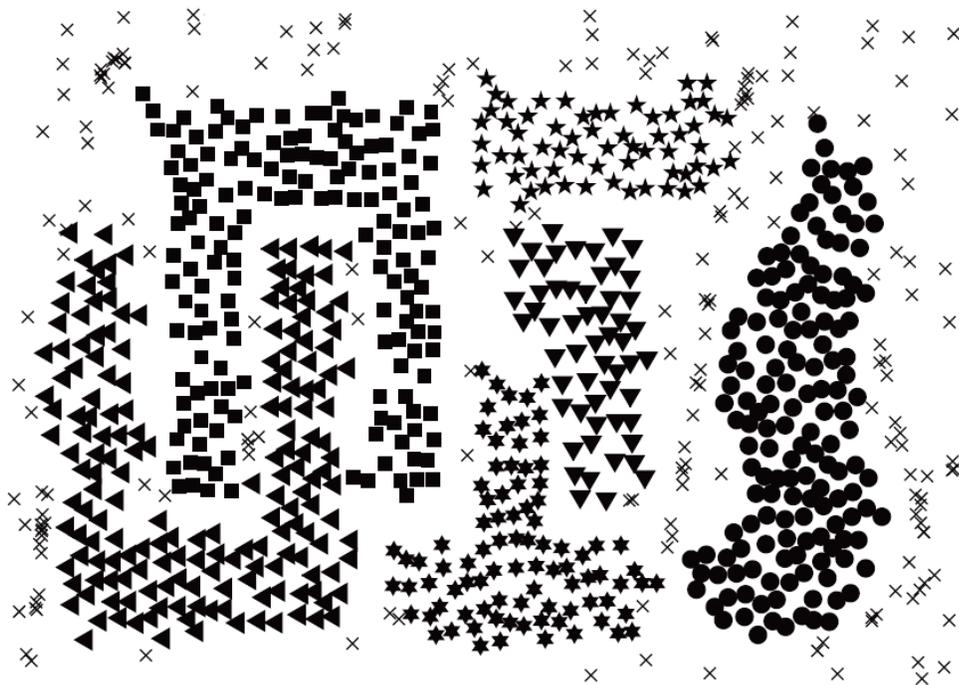


Abbildung 2: Clusterung mittels dichte-basiertem DBSCAN-Algorithmus¹⁹

In diesem abgebildeten Beispiel ist diese Problematik klar ersichtlich. Der angewandte dichte-basierte Algorithmus hat Strukturen hoher Dichte korrekt erkannt und geclustert. Dabei wurden Punkte, die zu weit außerhalb potenzieller Clusterstrukturen liegen im Gegensatz zur partitionierenden Verfahren aus der Wertung genommen (diese Punkte sind mit X gekennzeichnet), was für unseren Kontext auch vorteilhaft ist. Problematisch ist hierbei die längliche Anordnung der Dichtestrukturen. Die linken beiden Tisch-ähnlichen Cluster, welche aus ■ bzw. ▲ bestehen, sind durch den alleinigen Fokus auf Dichte nicht nur in ihrer Komposition unerwünscht länglich, sie sind sogar ineinander verschachtelt.

¹⁹ Tan 2021, S. 531.

4. Die gitterbasierten Clusteralgorithmen unterscheiden sich von den bisher vorgestellten Cluster-Algorithmen stark. Hierbei werden die vorhandenen Daten zunächst in kleinere Datenmengen unterteilt. Zweck dieser Unterteilung ist das effiziente Verarbeiten der Daten und schnelle Prognostizieren von Ähnlichkeitsstrukturen.²⁰ Unter anderem bei der Untersuchung von sozialen Netzwerken, in denen jede Sekunde massive Datenmengen generiert werden, werden aus diesem Grund gitterbasierte Clusteralgorithmen angewandt.

Da für Clusterungen von Entscheidungen keine riesige, hochdimensionale Menge an Daten vorliegen und somit kein Anspruch auf eine Effizienzsteigerung durch Vereinfachungen besteht, sind die gitterbasierten Cluster-Algorithmen für uns uninteressant.

5. Die Hierarchischen Clusteralgorithmen kann man umgangssprachlich als distanzbasierte Cluster-Algorithmen bezeichnen, da sie sich auf den ‚Abstand‘ der einzelnen Datenpunkte zu einem konkreten aber noch nicht im Vorhinein bekannten Punkt fokussiert.²¹ Dabei ist anzumerken, dass sich die Abstände nicht zwangsweise auf geographische Distanzen beziehen sondern auf den Abstand innerhalb eines Raumes mit beliebig ausgewählten Einheiten. Dieses Verfahren verfeinert oder vergrößert, je nach angewandtem Algorithmus schrittweise die vorhandene Cluster-Hierarchie. Hierarchische Clusteralgorithmen werden aufgrund dieser iterativen Vorgehensweise unter anderem in der Evolutionsbiologie am Phylogenetischen Baum angewandt.²² Der Phylogenetische Baum wurde durch eine hierarchische Clusteranalyse der DNA-Sequenzierung aller Tierarten aufgestellt und widerspiegelt somit den evolutionären Verlauf aller Lebewesen.²³ Somit konnte beispielsweise die langjährige Frage, ob Pandabären den Bären oder den Waschbären ähnlicher sind, beantwortet werden.²⁴

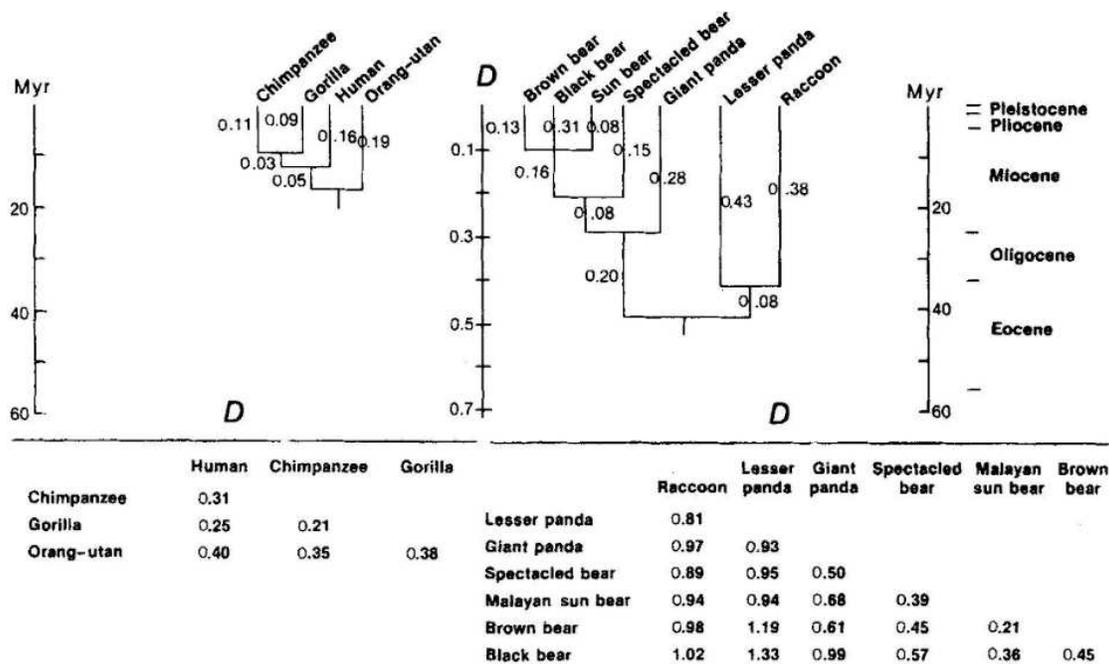
²⁰ vgl. Xu und Tian 2015, S. 174.

²¹ vgl. Xu und Wunsch 2005, S. 648–650.

²² vgl. Heymans und Singh 2003, 139.

²³ vgl. Blanchette et al. 2012, S. 301.

²⁴ vgl. O'Brien 1985, S. 140–144.

Abbildung 3: Hierarchische Genom-Clusterung²⁵

In der Abbildung gut sichtbar ist die hierarchische Struktur: In dem rechts abgebildeten phylogenetischen Baum der Bären beispielsweise liegt auf der untersten Ebene der Hierarchie nur ein einziges Cluster mit allen Spezies vor. Im Kontext der Evolution widerspiegelt diese Ebene den Zeitpunkt, an dem alle Bärenspezies noch denselben Vorfahren besitzen.

Die iterative Herangehensweise hierarchischer Clusteralgorithmen ermöglicht eine facetierte Betrachtung der vorliegenden Daten. Je nachdem welche Iterationsebene der Clusterhierarchie man betrachtet erhält man verschiedene Clusterbildungen und kann sich für die gewünschte Clusterbildung entscheiden. Des Weiteren lassen sich hierarchische Clusteralgorithmen gut mit anderen Algorithmen kombinieren. Diese Anwendung weiterer Algorithmen kann auch zwischen Iterationsschritten vorgenommen werden.

Von den vorgestellten Kategorien der Clusteralgorithmen erweisen sich die hierarchischen Clusteralgorithmen damit als besonders vorteilhaft in Bezug auf eine Clusterung für Entscheidungen im Produktionsnetzwerk. Im folgenden Unterkapitel werden deshalb verschiedene hierarchische Clusteralgorithmen genauer analysiert und bewertet, inwiefern sie sich für unsere Aufgabe eignen.

²⁵ O'Brien 1985, S. 142.

3.2 Analyse und Bewertung hierarchischer Clusteralgorithmen

Hierarchische Clusteralgorithmen fallen in zwei Kategorien²⁶:

1. Agglomerative Clusteralgorithmen oder auch „Bottom-up-Verfahren“ bilden zunächst aus jedem Punkt ein Cluster und fassen diese schrittweise zu größeren Clustern zusammen
2. Divisive Clusteralgorithmen oder auch „Top-down-Verfahren“ bilden zunächst aus allen Punkten ein Cluster und teilt schrittweise die vorhandenen Cluster in immer kleinere Cluster auf

Die verschiedenen hierarchischen Clusteralgorithmen innerhalb dieser beiden Kategorien unterscheiden sich dabei nur in ihren verwendeten Distanzmaßen und Berechnungsvorschriften. Deshalb werden in den folgenden zwei Abschnitten jeweils ein agglomerativer und ein divisiver Clusteralgorithmus betrachtet und auf Verwendbarkeit in Bezug auf Clustering von Entscheidungen untersucht:

Im **Single Linkage Verfahren** wird zunächst jeder Punkt als Cluster definiert und anschließend schrittweise die jeweils (euklidisch) nächstgelegenen Cluster zu einem Cluster fusioniert, bis man das Verfahren abbricht.²⁷ Diese Eigeninitiative, den Abbruch bestimmen zu müssen, kann als Nachteil gewertet werden, ermöglicht aber auch eine Implementation eigener Abbruchkriterien, beispielsweise der Überschreitung einer festgelegten Maximaldistanz oder der Erreichung einer bestimmten Clusterzahl. Ein großes Problem des Single Linkage Verfahrens stellt die potenzielle Bildung länglicher Cluster dar. Es ist derselbe Grund weshalb sich die dichtebasierten Clusteralgorithmen nicht für eine Entscheidungsclustering eignen. Diese Problematik soll am folgenden Beispiel verdeutlicht werden.

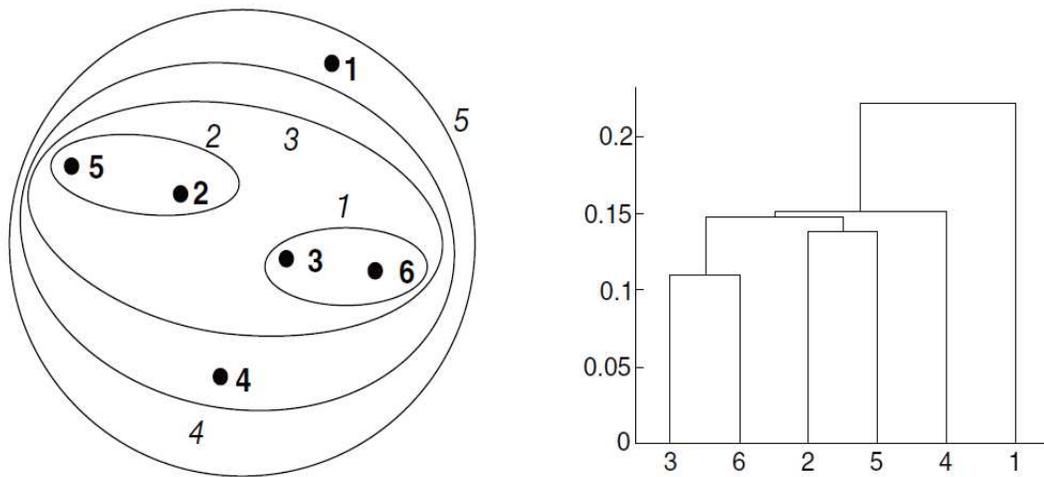


Abbildung 4: Clusterung mittels Single Linkage Algorithmus²⁸

Im hier abgebildeten Single Linkage Verfahren wurden die Punkte **1** bis **6** über die euklidische Distanz geclustert. Die Distanzgrenze, ab der zwei Cluster miteinander verschmolzen werden,

²⁶ vgl. Xu und Wunsch 2005, S. 650.

²⁷ vgl. Rohlf 1982, S. 2–4.

²⁸ Tan 2021, S. 520.

wurde iterativ erhöht. Im Iterationsschritt, bei dem alle Cluster mit Abstand 0.149 miteinander verschmolzen werden, besteht das größte Cluster aus den Punkten **2**, **3**, **5** und **6**. Dabei ist allerdings ersichtlich, dass der Abstand der Punkte **3** und **4** zueinander geringer ist als der Abstand zwischen den Punkten **5** und **6**, welche im Gegensatz zu **3** und **4** aber innerhalb desselben Clusters liegen.

Das **Divisive Analysis Verfahren** kann man sich in etwa wie die Umkehrung des Single Linkage Verfahrens vorstellen. Es werden zunächst alle Punkte als ein großes Cluster definiert und anschließend so lange ein Cluster daraus ‚abgeschnitten‘, bis man abbricht.²⁹ Allerdings muss im Gegensatz zum Single Linkage Verfahren, welches lediglich in jedem Iterationsschritt den Abstand der Cluster zueinander betrachtet, eine eigene Heuristik formuliert werden, nach der ein Cluster abgespaltet wird. Das Definieren einer solchen passenden Heuristik erweist sich bereits bei einfachen, klar definierten Anwendungsfällen als schwierig. Ebenso ist für den Anwendungsfall der Entscheidungsclusterung keine passende ‚Abschneidungsheuristik‘ ersichtlich.

Wie wir gesehen haben, eignen sich aus den 5 Kategorien von Clusteralgorithmen die Clusteralgorithmen aus der Kategorie der hierarchischen Clusteralgorithmen am meisten für eine Clusterung von Entscheidungen. Die hierarchischen Clusteralgorithmen lassen sich wiederum in agglomerative Clusteralgorithmen und in divisive Clusteralgorithmen unterteilen. Das Vorgehen dieser beiden Arten hierarchischer Clusteralgorithmen wurde an jeweils einem konkreten Verfahren untersucht hinsichtlich einer möglichen Anwendung zur Clusterung von Entscheidungen. Da sich sowohl das agglomerative als auch das divisive Vorgehen diesbezüglich nicht gut eignet, soll im nächsten Kapitel ein geeigneter Algorithmus konzeptioniert werden.

²⁹ vgl. Pedro Rodrigues et al. 2008, S. 3–4.

4 Konzept

Um ein Konzept für die Clusterung von Produktionsstandorten zu entwickeln sollten zunächst die Anforderungen an solche Clusterung klargemacht werden. Jede Clusterung hat ihr eigenes Set an relevanten **Faktoren**. Diese Faktoren sind nicht eindeutig bestimmbar sondern subjektiv. Es ist also erforderlich, sie im Vorhinein unternehmensindividuell quantifizieren zu lassen.

Diese Quantifizierung sollte feste, ganze und überschaubare große Zahlenwerte (z.B. 1, 2, 3, 4, 5) ergeben, wobei nah aneinander liegende Kennzahlen auf ähnliche Eigenschaften in dem entsprechenden Faktor hindeuten. Zusätzlich ist anzumerken, dass es nach einer Clusterung immer noch clusterlose Standorte geben kann und dass bei einer Clusterung mehrere Cluster resultieren können, welche sich allerdings nicht überlappen sollen. Im Kontext der Clusterung von Entscheidungen bedeutet diese Vermeidung von Überlappungen, dass ein Produktionsstandort in einer Entscheidungskategorie auch nicht von mehr als einem dieser Entscheidungscluster die Entscheidungsgewalt erfahren soll.

Für die Konzeptionierung einer mathematische Herangehensweise soll an dieser Stelle der Aufbau von Produktionsnetzwerken kurz wiederholt werden und in eine mathematische Schreibweise überführt werden:

Ein Produktionsnetzwerk ist eine Menge von Produktionsstandorten:

$$P = \{s^1, s^2 \dots s^n\}$$

Hierbei stellt P das Produktionsnetzwerk dar, die verschiedenen s^i die verschiedenen Standorte des Produktionsnetzwerkes, und n die Anzahl an Standorten.

Die Produktionsstandorte werden charakterisiert durch eine bestimmte Anzahl an Faktoren:

$$f : P \rightarrow \mathbb{R}^d$$

f ist eine Abbildung, bei der jeder Punkt s aus P auf einen Vektor der Dimension d abgebildet wird. d ist dabei die Anzahl an Faktoren, welche die Produktionsstandorte charakterisieren. Wir sagen auch kurz jeder Punkt s aus d Komponenten.

Die Faktoren sind durch je einen festen, ganzzahligen Wert quantifiziert:

$$s_j \in \{x \in \mathbb{N} \mid x \leq w\} \quad \forall s_j \text{ mit } 1 \leq j \leq d$$

Hierbei stellt s_j eine Komponente eines Punktes $s = s^i$ dar. Diese Komponente nimmt den Wert einer natürlichen Zahl zwischen 1 und w an.

Für eine Clusterung von Entscheidungen können verschiedene Anforderungen an die gewünschten Cluster gegeben sein. Diese Anforderungen müssen sich auf einen Teil der charakterisierenden Faktoren beziehen. Diese Anforderungen an die relevanten Faktoren können in Form von 2 **Anforderungskategorien** vorliegen:

1. Punkte innerhalb eines Clusters müssen in diesem Faktor den *gleichen* Wert besitzen
2. Punkte innerhalb eines Clusters müssen in den Faktoren *ähnlich* sein

Die erste Anforderungskategorie, dass alle Punkte innerhalb eines Clusters denselben Wert besitzen müssen, sollte als erstes abgehandelt werden und stellt eine Art Filterfunktion dar. Die gegebene Menge an Punkten kann man für jeden Faktor mit dieser Anforderung in kleinere Mengen unterteilen, also pro möglichen Wert bildet man eine Menge an Punkten, für die eine Clusterbildung in Frage kommt.

Falls in einem Faktor nicht nur der gleiche Wert verlangt ist, sondern konkret ein genauer Wert, wird die Menge an Punkten logischerweise nur in die Menge mit dem entsprechenden Wert in diesem Faktor unterteilt.

Im folgenden Beispiel wird angenommen, dass die Werte aller Faktoren ganzzahlig zwischen 1 und 5 liegen und genau 3 Faktoren vorliegen, von denen alle für die durchzuführende Clusterung relevant sind. Damit ist $w = 5$ und $d = 3$

Bsp: Gegebene Anforderung an eine Clusterung:

Ein Cluster enthält alle $s \in P$ mit

$s_1 = 3$ (*genau den Wert 3 im ersten Faktor*)

$s_2 \equiv \text{const1}$ für eine feste Konstante const1 (*gleicher Wert im zweiten Faktor*)

$s_3 \equiv \text{const2}$ für eine feste Konstante const2 (*gleicher Wert im dritten Faktor*)

Mit der ersten Anforderung fallen alle Punkte, die im ersten Faktor nicht den Wert 3 haben, aus der Wertung. Nach dieser ersten Filterung werden durch die zweite Anforderung aus der verbleibenden Menge $w = 5$ Mengen gebildet, also eine mit $\{s \in P \mid s_1 = 3 \wedge s_2 = 1\}$ bis hin zu der Menge $\{s \in P \mid s_1 = 3 \wedge s_2 = 5\}$. Aus diesen 5 Mengen werden durch die dritte Anforderung jeweils wiederum auf analoge Art und Weise 5 weitere Mengen gebildet, womit man am Ende mit $1 \cdot 5 \cdot 5 = 25$ Mengen an Punkten verbleibt. Jede dieser Mengen kann mit den vorgegebenen Anforderungen zu einem Cluster zusammengefasst werden.

Im Anschluss an diese Filterung gemäß der ersten Anforderungskategorie können die resultierenden Mengen auf Ähnlichkeiten in den restlichen relevanten Faktoren untersucht werden. Dafür sollte zuerst Klarheit geschaffen werden, wie man den Begriff der **Ähnlichkeit** zu verstehen hat. Im Folgenden werden zwei mögliche Auffassungen von Ähnlichkeit bewertet werden.

- i. In einem relevanten Faktor sollen die Werte innerhalb einer bestimmten Spanne liegen kommen für eine Clusterbildung in Frage.
- ii. Bei allen relevanten Faktoren sollen die Punkte nicht zu weit von einem Punkt entfernt liegen. Dies unterscheidet sich insofern vom vorherigen Begriff der Ähnlichkeit, als dass die Faktoren nicht vereinzelt betrachtet werden sondern in ihrer Gesamtheit untersucht werden.

Der Unterschied dieser beiden Auffassungen von Ähnlichkeit soll an folgendem Beispiel verdeutlicht werden:

- (i.) Es sollen alle Städte geclustert werden, die sich zum einen nicht weiter als 50km südlich oder nördlich von Aachen befinden, und zum anderen nicht weiter als 50km

westlich oder östlich von Aachen befinden. In diesem rechteckigen vorgegebenen Raum ist Düsseldorf enthalten (48km östlich, 49km nördlich).

- (ii.) Es sollen alle Städte geclustert werden, die nicht weiter als 50km von Aachen entfernt sind. In diesem runden vorgegebenen Raum ist Düsseldorf nicht enthalten, da es mit seiner Lage 48km östlich und 49km nördlich von Aachen etwa $\sqrt{(48km)^2 + (49km)^2} = 68,6km$ entfernt liegt.

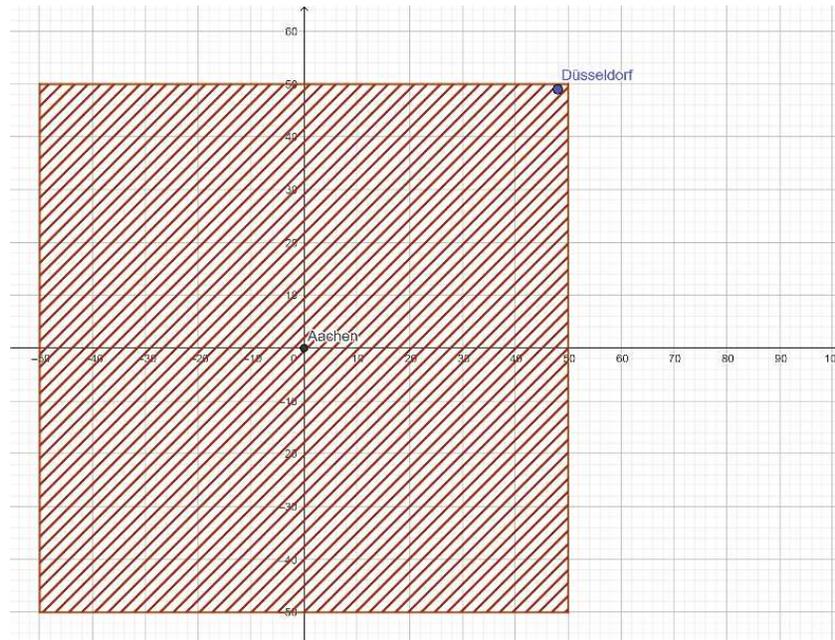


Abbildung 5: Auffassung i. von Ähnlichkeit

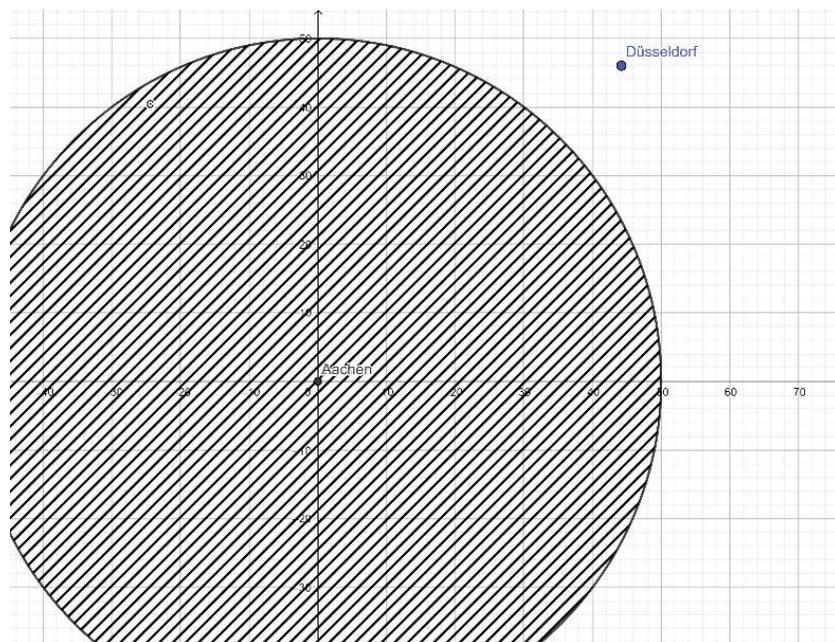


Abbildung 6: Auffassung ii. von Ähnlichkeit

Wie man am Beispiel sieht, können für den ersten Fall(i.) die Faktoren vereinzelt betrachtet werden, wohingegen im zweiten Fall(ii.) die Faktoren in ihrer Gesamtheit betrachtet werden, da sie beide zu dem vorgegebenen Abstand beitragen. Nun stellt sich die Frage, welche der beiden Fälle für den Kontext der Clusterung von Entscheidungen angebrachter ist.

Für die Clusterung von Entscheidungen trifft die erste Betrachtungsweise i. eher zu, da man die Eigenschaften von Produktionsstandorten voneinander unabhängig betrachten kann. Es soll also beispielsweise nicht vorgegeben werden, dass sich innerhalb eines Clusters die Produktionsstandorte in den Faktoren Digitalisierungsgrad und Kompetenzen innerhalb von 1 zum Mittelpunkt befinden(runder vorgegebener Einschränkungraum), sondern dass die Produktionsstandorte im Faktor Digitalisierungsgrad nicht weiter als 2 auseinanderliegen und im Faktor Kompetenzen nicht weiter als 2 auseinanderliegen(eckiger vorgegebener Einschränkungraum).

Neben der genannten Auffassung von Ähnlichkeit nach euklidischem Distanzmaß existieren noch viele weitere Auffassungen, von denen in der folgenden Abbildung die gängigsten aufgelistet sind:

Measures	Forms	Comments	Examples and Applications
Minkowski distance	$D_y = \left(\sum_{i=1}^d x_{yi} - x_{yi} ^n \right)^{1/n}$	Metric. Invariant to any translation and rotation only for $n=2$ (Euclidean distance). Features with large values and variances tend to dominate over other features.	Fuzzy c -means with measures based on Minkowski family [130].
Euclidean distance	$D_y = \left(\sum_{i=1}^d x_{yi} - x_{yi} ^2 \right)^{1/2}$	The most commonly used metric. Special case of Minkowski metric at $n=2$. Tend to form hyperspherical clusters.	K -means algorithm [191]
City-block distance	$D_y = \sum_{i=1}^d x_{yi} - x_{yi} $	Special case of Minkowski metric at $n=1$. Tend to form hyperrectangular clusters.	Fuzzy ART [57]
Sup distance	$D_y = \max_{1 \leq i \leq d} x_{yi} - x_{yi} $	Special case of Minkowski metric at $n \rightarrow \infty$.	Fuzzy c -means with sup norm [39].
Mahalanobis distance	$D_y = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{S}^{-1} (\mathbf{x}_i - \mathbf{x}_j)$, where \mathbf{S} is the within-group covariance matrix.	Invariant to any nonsingular linear transformation. \mathbf{S} is calculated based on all objects. Tend to form hyperellipsoidal clusters. When features are not correlated, squared Mahalanobis distance is equivalent to squared Euclidean distance. May cause some computational burden.	Ellipsoidal ART [13], Hyperellipsoidal clustering algorithm [194].
Pearson correlation	$D_y = (1 - r_y)/2$, where $r_y = \frac{\sum_{i=1}^d (x_{yi} - \bar{x}_i)(x_{yi} - \bar{x}_i)}{\sqrt{\sum_{i=1}^d (x_{yi} - \bar{x}_i)^2 \sum_{i=1}^d (x_{yi} - \bar{x}_i)^2}}$	Not a metric. Derived from correlation coefficient. Unable to detect the magnitude of differences of two variables.	Widely used as the measure for analyzing gene expression data [80].
Point symmetry distance	$D_{ps} = \min_{\substack{j \in 1, \dots, N \\ j \neq i}} \left(\frac{\ (\mathbf{x}_i - \mathbf{x}_r) + (\mathbf{x}_j - \mathbf{x}_r)\ }{\ (\mathbf{x}_i - \mathbf{x}_r)\ + \ (\mathbf{x}_j - \mathbf{x}_r)\ } \right)$	Not a metric. Compute the distance between an object \mathbf{x}_i and a reference point \mathbf{x}_r . D_{ps} is minimized when a symmetric pattern exists.	SBKM (Symmetry-based K -means) [264].
Cosine similarity	$S_y = \cos \alpha = \frac{\mathbf{x}_i^T \mathbf{x}_j}{\ \mathbf{x}_i\ \ \mathbf{x}_j\ }$	Independent of vector length. Invariant to rotation, but not to linear transformations.	The most commonly used measure in document clustering [261].

Abbildung 7: Verschiedene Distanzmaße³⁰

³⁰ Xu und Wunsch 2005, S. 648.

Von den abgebildeten weiteren Auffassungen von Ähnlichkeit wurde mit ii. nur die Ähnlichkeitsauffassung nach euklidischem Distanzmaß betrachtet. Trotz dieser alleinigen Berücksichtigung von nur dem euklidischen Distanzmaß können die weiteren Auffassungen von Distanzmaßen für unseren Kontext beiseitegelassen werden, da sie, genau wie bei dem euklidischen Distanzmaß, die jeweiligen Faktoren nicht unabhängig voneinander betrachten. Diese fehlende Unabhängigkeit zwischen Faktoren ist an den verschiedenen x_i und x_j unter der Spalte ‚Forms‘ ersichtlich.

Bei einer Clusterung ist zu beachten, dass sich die Bereiche überlappen können. Dadurch können Punkte in mehr als einem Cluster liegen. Für eine Clusterung mit den relevanten Faktoren A und B und der Vorgabe $\Delta A \leq 1$ und $\Delta B \leq 1$ wäre beispielsweise der Punkt (A=3, B=4) in bis zu 5 Clustern anzusiedeln. Diese **Überlappungen** sind für unseren Kontext unerwünscht und müssen in unserem Clusteralgorithmus abgefangen werden.

Unseren vorgegebenen Einschränkungsräum kann man sich als ‚eckigen‘, quaderförmigen **Körper** vorstellen. In einem zweidimensionalen Raum, welcher durch Achsen mit den Werten der Faktoren A und B aufgespannt ist, resultieren die Clustervorgaben $\Delta A \leq 1$ und $\Delta B \leq 1$ in einen rechteckigen Körper der Größe 2×2 . Das bedeutet: Ein Cluster mit diesen Clustervogaben besteht stets aus Punkten, welche sich innerhalb eines Körpers der Größe 2×2 befinden. Dieser zweidimensionale Körper kann auch weit über 2 Dimensionen besitzen falls dementsprechend viele Clustervogaben gemacht werden. In diesem Fall kann man sich den Körper zwar nicht mehr räumlich vorstellen, allerdings immer noch mit ihm clustern. Hierbei ist auch anzumerken, dass mit dem Konzept eines Körpers auch die erste Anforderungskategorie an Faktoren („Die Cluster müssen in diesem Faktor den *gleichen* Wert besitzen“) abhandeln kann, indem man dem Körper in diesen Faktoren die Dimension 0 gibt.

Mit den bisher beschriebenen Überlegungen kann bereits ein **Algorithmus** beschrieben werden:

1. Mit den vorliegenden Clustervorgaben definiert man einen Körper der entsprechenden Größe und Dimension
2. Den Körper verschiebt man innerhalb des durch die relevanten Faktoren aufgespannten Raumes und vermerkt dabei auf die Menge an Punkten innerhalb des Körpers
3. Wenn dabei die Position erreicht wird, bei der die größtmögliche Menge an Punkten eingeschlossen wird, bildet man mit dieser Menge an Punkten ein Cluster
4. Die Punkte des Clusters aus dem Raum entfernen und die Schritte ab 2. wiederholen
5. **Abbruchbedingung:** Wenn keine Punkte mehr im Raum vorhanden sind, hat man alle Punkte geclustert und kann abbrechen

Den letzten Schritt kann man optional auch noch modifizieren zu einer **Abbruchbedingung**, welche nicht erst in Kraft tritt, wenn keine Punkte mehr vorhanden sind, sondern schon in Kraft tritt, wenn sich kein Cluster mehr mit mindestens x (beliebiger Wert) Punkten finden lässt. Dies ergibt in unserem Kontext insofern Sinn, als dass sich bei einer sehr geringen Anzahl an Standorten innerhalb eines Clusters, beispielsweise einem einzelnen Standort, gar keine Clusterung mehr lohnt.

Ein **Sonderfall** in dem beschriebenen Algorithmus muss noch abgefangen werden: Liegen mehrere Positionen des Körpers mit maximaler Punkteinschließung vor, muss, wenn sie sich überschneiden, einer der Cluster darunter ausgewählt werden, wodurch die anderen überlappenden Cluster ‚zerstört‘ werden. Um zu entscheiden, welches dieser überlappenden Cluster man priorisiert, kann man eine Fallunterscheidung machen und am Ende des Algorithmus die Clusterbildung wählen, bei der beispielsweise die meisten Cluster gebildet wurden.

5 Zusammenfassung und Ausblick

5.1 Bewertung der Methodik

Der Algorithmus clustert Produktionsstandorte gemäß den gegebenen Anforderungen und zeigt somit, wo Ähnlichkeiten innerhalb des Produktionsnetzwerkes vorliegen. Die Laufzeit des Algorithmus ist mit dem Testdatensatz sehr gering und sollte auch bei größeren Produktionsnetzwerken mit ausgeprägteren Eigenschaftsvektoren der Produktionsstandorte gut laufen.

Ein Vorteil des Algorithmus bietet die anpassbare Abbruchbedingung. Hierbei können flexibel weitere Anforderungen an eine Clusterung implementiert werden, beispielsweise eine Maximalanzahl an Clustern oder eine Mindestanzahl an Produktionsstandorten innerhalb eines Clusters.

Die **Laufzeitkomplexität** des konzeptionierten Algorithmus stellt im Gegensatz zu anderen Clusteralgorithmen kein Problem dar, da er unabhängig von der Anzahl vorliegender Punkte n ist. Um dies zu zeigen sind in der folgenden Abbildung die Laufzeitkomplexitäten der am häufigsten angewandten Clusteralgorithmen aufgelistet.

K-means	K-medoids	PAM	CLARA	CLARANS
$O(knt)$	$O(k(n-k)^2)$	$O(k^3 \cdot n^2)$	$O(ks^2 + k(n-k))$	$O(n^2)$
Low	High	High	Middle	High

Abbildung 8: Laufzeitkomplexität verschiedener Clusteralgorithmen³¹

BIRCH	CURE	ROCK	Chameleon
$O(n)$	$O(s^2 \cdot s)$	$O(n^2 \cdot \log n)$	$O(n^2)$
Low	Low	High	High

Abbildung 9: Laufzeitkomplexität verschiedener Clusteralgorithmen³²

DBCLASD	GMM
$O(n \cdot \log n)$	$O(n^2 \cdot kt)$
Middle	High

Abbildung 10: Laufzeitkomplexität verschiedener Clusteralgorithmen³³

³¹ Xu und Tian 2015, S. 171.

³² Xu und Tian 2015, S. 171.

³³ Xu und Tian 2015, S. 172.

In den abgebildeten Laufzeitkomplexitäten O ist, mit einer Ausnahme des partitionierenden CURE-Algorithmus, stets die Anzahl der vorliegenden Punkte n ausschlaggebend für die resultierende Laufzeitkomplexität.

Um die Laufzeitkomplexität des konzeptionierten Algorithmus zu bestimmen muss der Worst-Case betrachtet werden. Dieser liegt vor, wenn der gegebene eckige, quaderförmige Körper so klein wie möglich ist und zugleich die für eine Clusterung der Entscheidung umfassenden relevanten Faktoren nicht nur einen Teil, sondern alle vorhandenen Faktoren beinhalten. In einem solchen Fall beträgt die Laufzeitkomplexität:

$$O(w^d)$$

Die Dimension d des Abbildungsbereiches beträgt in unser Anwendung in etwa 5, ähnlich der annehmbaren Werte w . Die Standorte unseres Produktionsnetzwerkes lassen sich demnach charakterisieren durch 5 Eigenschaften ($d = 5$), welche wiederum jeweils einen von 5 Werten ($w = 5$) annehmen. Mit den Werten 5 ergibt sich somit eine Laufzeitkomplexität von:

$$5^5 = 3125$$

Die Laufzeitkomplexität liegt damit weit unter dem für Rechner kritischen Bereich und ist von der Anzahl an Punkten unabhängig.

Die Möglichkeit, eine eigene **Abbruchbedingung** zu definieren, bietet einen weiteren Vorteil des konzeptionierten Algorithmus. Genau wie die iterative Vorgehensweise der hierarchischen Clusteralgorithmen kann nach jedem Iterationsschritt auf eine eigene Abbruchbedingung geprüft werden.

Der **Sonderfall**, bei dem nach einem Iterationsschritt mehr als eine Position des Einschränkungskörpers mit maximaler Punkteinschließung vorliegt und sich diese Positionen überschneiden, stellt ein Problem dar, kann aber durch gezieltes Auswählen von einem dieser Positionen gelöst werden. Bei Auswahl nach einfachem Zufallsprinzip kann eine Fallunterscheidung umgangen werden und somit die Komplexität des Algorithmus kleingehalten werden.

5.2 Kritische Reflexion und Ausblick

Für eine Clusterung von Produktionsstandorten kann sich das Versehen der relevanten Faktoren mit einer Kennzahl als problematisch erweisen. Ein Faktor wie Maschinenarten kann man beispielsweise mit einer Kennzahl versehen, welche bei Gleichheit zwischen bestimmten Standorten auch ähnliche Maschinenarten dieser Standorte impliziert, allerdings kann es sein, dass sich dieser Faktor nicht als Kennzahl auf einer Skala von niedrig bis hoch verstehen lässt wie es bei dem Erfassen der Digitalisierungsgrade schon eher der Fall ist. In einem solchen Fall bleibt einem noch die Möglichkeit, nur nach genau gleichen Werten in diesen Faktoren zu clustern, oder, trotz der fehlenden Quantifizierbarkeit auf einer Skala, die Maschinenarten die sich für eine gemeinsame Clusterung eignen, mit ähnlichen Kennwerten zu versehen.

Im Rahmen dieser Arbeit wurden Clusteralgorithmen in Bezug auf die Anwendung der Clusterung von Entscheidungen in Produktionsnetzwerken untersucht und bewertet. Im Anschluss wurde ein Clusteralgorithmus für Entscheidungen im Produktionsnetzwerk konzeptioniert und vorgestellt. Dazu wurden auch die nötigen Grundlagen zu Produktionsnetzwerken und Entscheidungen erläutert.

Der konzeptionierte Algorithmus eignet sich für eine Clusterung von Entscheidungen in Produktionsnetzwerken, wenn im Vorhinein die relevanten Faktoren festgelegt, diskutiert und mit einem festen Wert charakterisiert werden. Neben dem Festlegen der Vorgaben an eine Clusterung muss der Anwender auch die durch den Algorithmus entstehenden Cluster selbst bewerten. Der Algorithmus führt in seiner aktuellen Form nur eine einfache Clusterung durch, kann aber durch Vorverarbeitung der Daten oder Festlegung einer Abbruchbedingung mit neuen Heuristiken flexibel erweitert werden. Zusätzlich kann der Anwender den Algorithmus auch dazu nutzen, neue Clusterbildungen bei leichten Anpassungen der eingegeben Clustervorgaben zu untersuchen.

Literaturverzeichnis

Blanchette, Glenn; O'Keefe, Richard; Benuskova, Lubica (2012): Inference of a Phylogenetic Tree: Hierarchical Clustering versus Genetic Algorithm. In: David Hutchison, Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell et al. (Hg.): *AI 2012: Advances in Artificial Intelligence*, Bd. 7691. Berlin, Heidelberg: Springer Berlin Heidelberg (Lecture Notes in Computer Science), S. 300–312.

Fränken, Bastian (2021): Standortrollen in Produktionsnetzwerken.

Gützlaff, Andreas (2021): Entscheidungsprozess zur kontinuierlichen Gestaltung von Produktionsnetzwerken.

Harre, Jan (2006): Strategische Standortstrukturplanung für multinational produzierende Unternehmen.

Heymans, Maureen; Singh, Ambuj K. (2003): Deriving phylogenetic trees from the similarity analysis of metabolic pathways. In: *Bioinformatics (Oxford, England)* 19 Suppl 1, i138-46. DOI: 10.1093/bioinformatics/btg1018.

Ivanescu, Sebastian-Mihail (2014): Standortsspezifische Anpassung von Produktionssystemen unter Berücksichtigung kultureller Einflüsse.

Kirchner, Kathrin (2003): Moderne Clusteralgorithmen-eine vergleichende Analyse auf zweidimensionale Daten.

Kumar, M. Rajesh; Venkatesh, J.; Rahman, A. M. J. Md Zubair (2021): Data mining and machine learning in retail business: developing efficiencies for better customer retention. In: *J Ambient Intell Human Comput.* DOI: 10.1007/s12652-020-02711-7.

Lanza, Gisela; Ferdows, Kasra; Kara, Sami; Mourtzis, Dimitris; Schuh, Günther; Váncza, József et al. (2019): Global production networks: Design and operation. In: *CIRP Annals* 68 (2), S. 823–841. DOI: 10.1016/j.cirp.2019.05.008.

Merchiers, Andreas (2008): Bewertung globaler Standortstrukturalternativen im Maschinenbau.

O'Brien, Stephen J. (1985): A molecular solution to the riddle of the giant panda's phylogeny. Proceedings of the thirteenth ACM SIGKDD international conference on knowledge discovery and data mining, August 12-15, 2007, San Jose, CA, USA. New York NY: Association for Computing Machinery.

P. Raykov, Yordan (2016): What to Do When K-Means Clustering Fails A Simple yet Principled Alternative Algorithm.

Pedro Rodrigues et al. (2008): Hierarchical Time-Series Clustering for Data Streams.

Rohlf, F. James (1982): Single-link Clustering Algorithms.

Schuh, G.; Gützlaff, A.; Thomas, K.; Schlosser, T. X. (2020 - 2020): Determination of an Efficient Degree of Centralization in Global Production Networks. In: 2020 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM). 2020 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM). Singapore, Singapore, 14.12.2020 - 17.12.2020: IEEE, S. 847–851.

Schuh, Gützlaff et al. (2021): Framework for determining the degree.

Tan, Pang-Ning (2021): Introduction to Data Mining (Second Edition).

Varandani, Rawina Mehru (2014): Managementkomplexität als Gestaltungsgröße kostenoptimierter globaler Produktionsnetzwerke.

Xu, Dongkuan; Tian, Yingjie (2015): A Comprehensive Survey of Clustering Algorithms. In: *Ann. Data. Sci.* 2 (2), S. 165–193. DOI: 10.1007/s40745-015-0040-1.

Xu, Rui; Wunsch, Donald (2005): Survey of clustering algorithms. In: *IEEE transactions on neural networks* 16 (3), S. 645–678. DOI: 10.1109/TNN.2005.845141.