

Titel der Arbeit auf Deutsch

English Title

Bachvarov, Vladislav

Master-Abschlussarbeit

Betreuer: Titel Vorname Name

Ort, Abgabedatum

Vorwort

Ein Vorwort ist nicht unbedingt ntig. Falls Sie ein Vorwort schreiben, so ist dies der Platz, um z.B. die Firma vorzustellen, in der diese Arbeit entstanden ist, oder einigen Leuten zu danken, die in irgendeiner Form positiv zur Entstehung dieser Arbeit beigetragen haben. Auf keinen Fall sollten Sie im Vorwort die Aufgabenstellung nr erlern oder vertieft auf technische Sachverhalte eingehen.

Kurzfassung

In der Kurzfassung soll in kurzer und prägnanter Weise der wesentliche Inhalt der Arbeit beschrieben werden. Dazu sind vor allem eine kurze Aufgabenbeschreibung, der Lösungsansatz sowie die wesentlichen Ergebnisse der Arbeit. Ein häufiger Fehler für die Kurzfassung ist, dass lediglich die Aufgabenbeschreibung (d.h. das Problem) in Kurzform vorgelegt wird. Die Kurzfassung soll aber die gesamte Arbeit widerspiegeln. Deshalb sind vor allem die erzielten Ergebnisse darzustellen. Die Kurzfassung soll etwa eine halbe bis ganze DIN-A4-Seite umfassen.

Hinweis: Schreiben Sie die Kurzfassung am Ende der Arbeit, denn eventuell ist Ihnen beim Schreiben erst vollends klar geworden, was das Wesentliche der Arbeit ist bzw. welche Schwerpunkte Sie bei der Arbeit gesetzt haben. Andernfalls laufen Sie Gefahr, dass die Kurzfassung nicht zum Rest der Arbeit passt.

The same in english.

Inhaltsverzeichnis

1	Einleitung und Problemstellung	1
2	Word2Vec	2
2.1	Word Embedding	2
3	Glove	5
3.1	The GloVe Method.....	5
4	Vektordarstellung	8
4.1	Einleitung	8
4.2	Prozess	8
5	Zusammenfassung und Ausblick	12
	Glossar	13
	Erklng der Kandidatin / des Kandidaten	14

Abbildungsverzeichnis

2.1	Skip-gram Word2Vec Model	3
2.2	CBOW Word2Vec Model	4
4.1	Prinzipiele Linie	10

Tabellenverzeichnis

3.1	Tabelle von dem Vorkommen der beiden Wörter <i>ice</i> und <i>steam</i>	6
-----	---	---

Einleitung und Problemstellung

Begonnen werden soll mit einer Einleitung zum Thema, also Hintergrund und Ziel erlert werden.

Weiterhin wird das vorliegende Problem diskutiert: Was ist zu lsen, warum ist es wichtig, dass man dieses Problem lst und welche Lsungsanse gibt es bereits. Der Bezug auf vorhandene oder eben bisher fehlende Lsungen begrndet auch die Intention und Bedeutung dieser Arbeit. Dies knnen allgemeine Gesichtspunkte sein: Man liefert einen Beitrag fr ein generelles Problem oder man hat eine spezielle Systemumgebung oder ein spezielles Produkt (z.B. in einem Unternehmen), woraus sich dieses noch zu lsende Problem ergibt.

Im weiteren Verlauf wird die Problemstellung konkret dargestellt: Was ist spezifisch zu lsen? Welche Randbedingungen sind gegeben und was ist die Zielsetzung? Letztere soll das beschreiben, was man mit dieser Arbeit (mindestens) erreichen mchte.

Word2Vec

In diesem Kapitel wird das Verfahren "Word2Vec" durch die Nutzung von Neuronalem Netz vorgestellt. Word2Vec ist eine Darstellung von Wörtern mit Vektoren, was auch aus der Abkürzung klar wird - Word ist klar; 2 - to; und Vec - Vector und das ganze "word to vector". Dieses Modell ist am meisten in der Natural Language Processing (NLP) verbreitet und wird in vielen Bereichen der Informatik genutzt, unter anderem in Spamfilterung und Dokumentenanalyse. Jedoch diese Technik besagt nur wie die Wörter eines Textes dargestellt werden können. Das Verfahren, bei dem die möglichst passenden Vektoren in einem ausgewählten Text, auch Corpus genannt, gelernt werden, heißt Word Embeddings. Bei dieser Technik wird ein Neuronales Netz eingesetzt. Die Vorgehensweise und die Idee wird folglich erklärt.

2.1 Word Embedding

Wie es schon in der Einleitung erwähnt wurde, Word Embedding ist der Prozess, bei dem die Wörter eines Textes in mathematischen Vektoren gewandelt werden. Zuerst muss der Corpus vorbereitet werden. Ich stelle hier nur die Theorie und in einem späteren Kapitel (!? WICHTIG WELCHE GENAU!?) gehe ich tiefer in den Programmcode.

!! DAS HIER GEHÖRT IN EIN ANDERES KAPITEL !!! !! DIE KAPITEL FÜR TEXTVORBEREITUNG ODER SOWAS!!!

Als der Text vorbereitet ist, sodass es von Sonderzeichen und allen unnötigen Zeichen bereinigt wird. Wenn der Text vorbereitet ist, werden die Wörter aus dem Corpus bestimmt und jeder erhält einen Index. Üblicherweise werden die Wörter nach ihrer Häufigkeit angeordnet. Das häufigste Wort erhält somit den Index 1. Als nächstes werden die Wörter im Corpus durch ihren Index ersetzt, um alle Trainingspaare fürs Lernen generiert zu werden. Dies erfolgt in dem es durch das Corpus iteriert und in einem bestimmten Fenster, oder in der Literatur auch als Window gezeichnet, alle Contextwörter und den Targetwort ausgelesen werden. Das Targetwort ist das Wort in der Mitte, während die Wörter um das Targetwort entsprechend die Kontextwörter.

!!! BIS HIER MUSS WEG !!!!

In der Literatur werden zwei Arten von Wort2Vec Modelle - SKIP-gram und CBOW (Continuous Bag of Words). Beide Modelle verwenden ein Neuronales Netz mit einem oder zwei versteckten Schichten (siehe !!KAPITEL MIT DEM PROGRAMMCODE!!). Die beiden Methoden unterscheiden sich nach ihren Ein- und Ausgaben.

SKIP-gram Model

Bei dem SKIP-gram-Model fließen die Targetwörter als Eingabe und das Model versucht ein Kontextwort zu raten. Hier ist die Struktur eines Skip-gram Models:

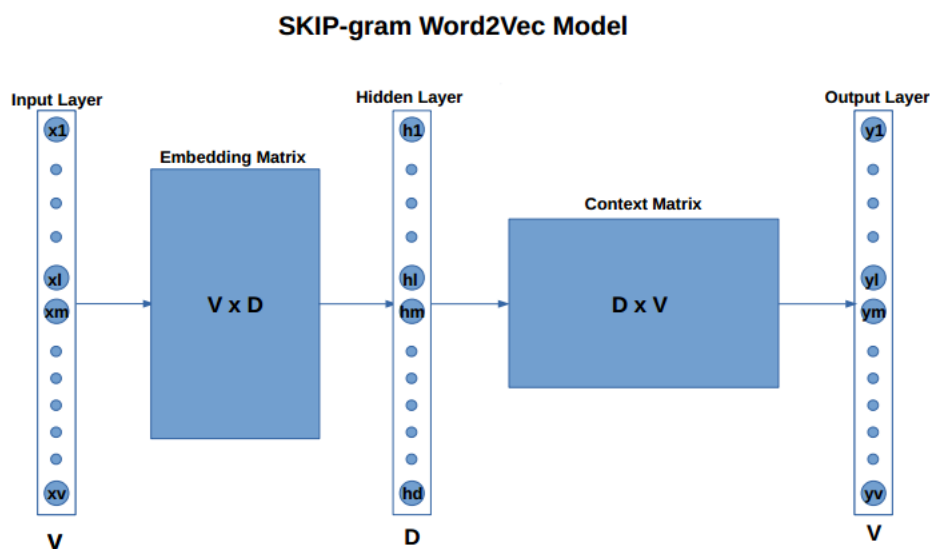


Abb. 2.1: Skip-gram Word2Vec Model

Aus der Abbildung 2.1 ist es zu entnehmen, dass ein Skip-gram Model aus einer hidden Schicht und zwei Eingabeschichten. Die Eingabe sowie die Ausgabe ist ein Vektor, der aus V Komponente besteht. Das entspricht die Größe des Wörterbuchs (engl. Vocabulary). Das versteckte Schicht besteht aus D Variablen und stellt einen Vektor dar. Die Dimensionalität dieses Vektors nimmt üblich einen Wert zwischen 25 und 300. Diese Variablen können auch als Eigenschaften für die Wörter betrachtet werden. Je mehr Kriterien es untersucht werden, desto besser die Beziehung zwischen Wörtern widerspiegelt werden kann. Die Ausgabe ist wieder einen V -dimensionalen Vektor. Jedoch die Ausgabe ist kein One-Hot Vektor mehr, der das Kontextwort wiedergibt, sondern einen Vektor mit Wahrscheinlichkeiten, zwar die Wahrscheinlichkeit für jedes Wort, das gesucht Kontextwort zu sein.

Die zwei Matrizen sind identisch, jedoch die Kontextmatrize ist die transponierte Embeddingmatrize. Diese Matrix beinhaltet unsere Wortvektoren. Deswegen ist sie eine $V \times D$ dimensionale Matrix.

CBOW Model

Die Kontextwörter sind die Eingabe in dem CBOW-Model und das Model ratet der Targetwort.

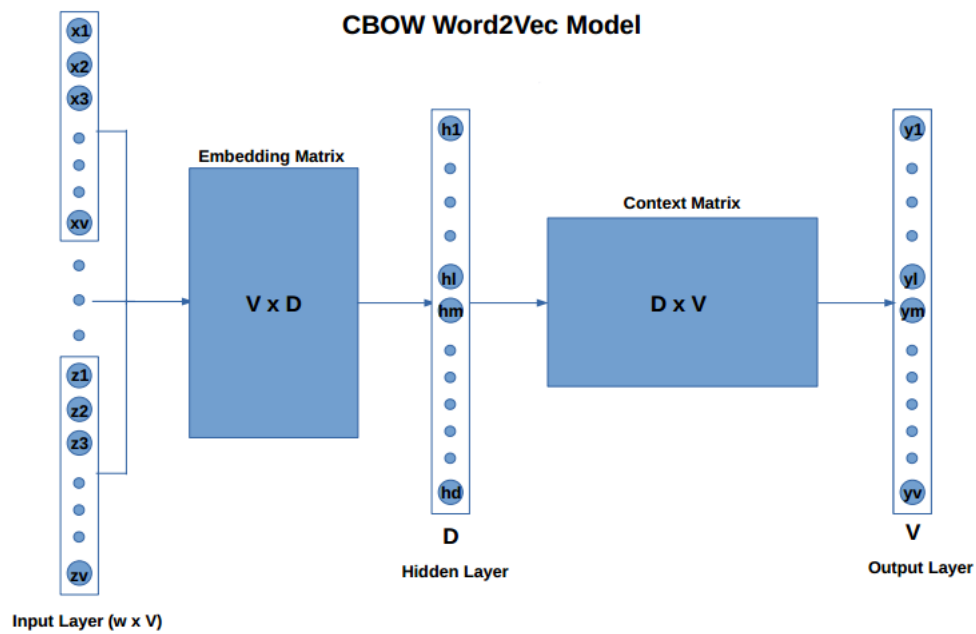


Abb. 2.2: CBOW Word2Vec Model

Glove

3.1 The GloVe Method

Der Vorrat von Wörtern ist die Hauptquelle von Information für unsupervised Learning zum Lernen von Wortrepräsentation. Trotz der zahlreichen Existenz von Algorithmen ist die Hauptfrage, wie man Sinngehalt aus den Statistiken herauszieht und wie die resultierende Vektorrepräsentation von Wörtern diesen Sinn spiegelt.

Wir führen einige Definitionen auf. Die Matrix der Word-Word-co-occurrence notieren wir mit X und einen Eintrag in der Matrix mit X_{ij} . Der Wert X_{ij} stellt dar, wie oft das Wort j in dem Kontext vom Wort i vorkommt. Weiterhin beschreibt die Einheit X_i die Anzahl des Vorkommens eines beliebigen Wortes in dem Kontext vom Wort i und ist in Gleichung 3.1 definiert:

$$X_i = \sum_k X_{ik}. \quad (3.1)$$

Schließlich definieren wir die Einheit P_{ij} , die die Wahrscheinlichkeit beschreibt, dass ein Wort j in dem Kontext vom Wort i vorkommt. Die Formel ist in der Gleichung 3.2 aufgeführt:

$$P_{ij} = P(j|i) = \frac{X_{ij}}{X_i}. \quad (3.2)$$

Ein kleines Beispiel wird angegeben, damit es verstanden werden kann, wie bestimmte Aspekte aus dem gemeinsamen Auftreten gewonnen werden können. Es werden zwei Wörter i und j betrachtet, die zu einer Menge gehören, für das Beispiel ist das der thermodynamische Zustand. Nehmen wir die Wörter $i = ice$ und $j = steam$. Die Beziehung der beiden Wörter kann so untersucht werden, indem ihre Relation mit anderen Probewörtern k berechnet wird. Für Wörter, die in direkter Bezug zu $i = ice$ stehen, erwarten wir, dass das Verhältnis $\frac{P_{ik}}{P_{jk}}$ groß ist. Zum Beispiel wird das Wort $k = solid$ betrachtet. Analog bei Wörtern, die näher zu $j = steam$ sind, erhalten wir einen kleineren Wert des Bruchs $\frac{P_{ik}}{P_{jk}}$ - in diesem Fall nehmen wir das Wort $k = gas$. Selbstverständlich ist das Verhältnis des Bruchs bei Wörtern, die entweder zu beiden Wörtern i, j in Bezug stehen oder solchen zu keinem nah an eins. Die Tabelle stellt das Verhältnis zwischen den Wörtern dar.

Probability and Ration	k = solid	k = gas	k = water	k = fashion
P(k—ice)	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
P(k—steam)	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
P(k—ice)/P(k—steam)	8.9	8.5×10^{-2}	1.36	0.96

Tabelle 3.1: Tabelle von dem Vorkommen der beiden Wörter *ice* und *steam*

In der Tabelle 3.1 wird die Beziehung zwischen die Wörter *ice* und *steam* dargestellt. Die Erwartungen werden durch die Tabelle gerechtfertigt. Die Rate erlaubt uns besser die Verhältnisse zwischen die einzelnen Wörter zu verstehen. Mit Hilfer des Bruches werden besser unterschieden, wie die Wörter zueinander stehen, im Vergleich zu der einfachen Wahrscheinlichkeit.

Die oben genannten Argumente ergeben, dass der Anfang von Word-Vector-Learning mit der Rate des gemeinsamen Auftretens starten soll, anstatt die Wahrscheinlichkeiten selbst. Zu betrachten ist, dass die Kookurenzwahrscheinlichkeit hängt von drei Eingangsgrößen ab - i , j , k . Die Allgemeinform der Funktion ist in der Gleichung 3.3 angegeben:

$$F(w_i, w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}}, \quad (3.3)$$

mit Wortvektoren $w \in \mathbb{R}$ und andere Wortvektoren $\tilde{w} \in \mathbb{R}$. In der Gleichung ergibt sich die Rechte Seite aus dem Korpus. F hängt in diesem Fall von den drei Vektoren w_i, w_j, \tilde{w}_k ab. F wird demnächst wegen der hohen Variation der Formel angepasst. Zuerst ist es gewünscht, die Information aus der Rate in Word-Vector-Raum darzustellen. Da Vektorräume ursprünglich linear sind, kann dies in einem Vektordiferenz erfolgen. Auf dieser wird der Fakus nur auf die Funktionen fallen, die von der Diferenz von den zwei Vektoren abhängen. Die Änderung ergibt sich in Gleichung 3.4.

$$F((w_i - w_j), \tilde{w}_k) = \frac{P_{ik}}{P_{jk}}. \quad (3.4)$$

Aus der Gleichung ist zu entnehmen, dass die Parameter von F Vektoren sind, während die Rechte einen Skalar ist. Währen F als eine komplexere Funktion, die von einem Neuronalen Netzt parametrisiert werden kann, genommen werden kann, würde das die Linearstruktur der Formel verschleiern. Es wird das Skalarprodukt genommen, damit das vermieden wird. Die Formel 3.5 stellt die Änderung dar.

$$F((w_i - w_j)^T \tilde{w}_k) = \frac{P_{ik}}{P_{jk}}. \quad (3.5)$$

In der word-word Kookuranzmatrizen erfolgt der Unterschied zwischen Wort und Kontextwort willkürlich. Die Formel für F muss es erlauben, die zwei Rollen beliebig zu tauschen. Das heißt also nicht nur $w \leftrightarrow \tilde{w}$ auszutauschen, sondern auch $X \leftrightarrow X^T$. Um diese Symmetrie zu verschaffen, sind zwei einfache Schritte erfordert.

Zuerst muss vergewissert werden, dass die Formel homomorphisch zwischen die Gruppen $(\mathbb{R}, +)$ und $(\mathbb{R}_{>0}, \times)$ ist:

$$F((w_i - w_j)^T \tilde{w}_k) = \frac{F(w_i^T \tilde{w}_k)}{F(w_j^T \tilde{w}_k)}, \quad (3.6)$$

was nach Gleichung 3.5 wie folgt gelöst wird:

$$F(w_i^T \tilde{w}_k) = P_{ik} = \frac{X_{ik}}{X_i}. \quad (3.7)$$

Die Lösung von 3.6 ist $F = \exp$, oder auch:

$$w_i^T \tilde{w}_k = \log(P_{ik}) = \log(X_{ik}) - \log(X_i). \quad (3.8)$$

Zunächst wird die Gleichung umgestellt, jedoch werden einige Konstanten, oder Bias, eingeführt. Es könnte der Wert von $\log(X_i)$ in einer Konstante b_i umgewandelt werden. Schließlich wird die Konstante b_k addiert, damit die Symmetrie erhalten wird. Die vereinfachte Formel ist in Gleichung 3.9 gegeben:

$$w_i^T \tilde{w}_k + b_i + \tilde{b}_k = \log(X_{ik}). \quad (3.9)$$

Vektordarstellung

4.1 Einleitung

Meistens bei der Repräsentation und Analyse einer numerischen Datensammlung werden die Daten nach bestimmten Kriterien klassifiziert, meistens nach mehr als zwei oder drei, sodass eine Graphische Darstellung relative schwierig zu bilden ist. In der Datenanalyse existieren entsprechende Methoden zur Darstellung von Daten mit mehreren Komponenten. Eins dieser Methode ist Principal Component Analysis (abgekürzt PCA), oder Prinzipale Komponentenanalyse. Diese Methode ist ideal in der NLP zu verwenden, da die Wörter in mehrdimensionalen Vektoren repräsentiert werden, üblicherweise solche mit mehr als drei Komponente. Die Methode verringert die Anzahl der Komponenten auf eine kleinere Zahl, zwei oder drei Dimensionen für eine Darstellung der Wörter im 2D- oder 3D-Raum entsprechend, jedoch so viele Informationen wie möglich über die einzelnen Wörter zu behalten.

4.2 Prozess

Ekläre wie die Berechnung erfolgt und, dass es eine Matrix verändert. Erkläre über DataFrame

1. Schritt: Standardization

Im ersten Schritt des PCAs handelt es sich um Standardisierung der Komponenten, sodass jeder gleichmäßig zu der Analyse beibringt. Dieser Schritt ist wichtig, da PCA sehr sensibel bezüglich Variation der Werte ist. Variablen mit großen Werten dominieren solche mit niedrigen und so ist das Endergebnis beeinflusst/voreingenommen. Die Standardisierung erfolgt in Formel:

$$z_{ij} = \frac{value - mean}{standard\ deviation}, \quad (4.1)$$

Wo Z_{ij} der normierte Wert mit Zeile i und Spalte j aus der Matrix ist. Der Meanwert ist der Durchschnitt in einem Vektor und der Standard Deviation bezieht sich auf dem selben Vektor. Nach der Normierung haben alle Werte der Matrix den selben Maßstab.

2. Schritt: Berechnung der Kovarianzmatrix

Nachdem die Normierte Matrix berechnet wurde, wird die Kovarianzmatrix erstellt. Ziel der Kovarianzmatrix ist es die Beziehung zwischen die Variablen zu bestimmen, bzw. wie sie wachsen. Die Abhängigkeit wird von dem Vorzeichen der Kovarianzwert bestimmt - bei positivem Wert wachsen die Variablen proportional und bei negativem - antiproporzional. Die Formel für die Kovarianz ist gegeben:

$$Cov(X, Y) = \sum \frac{E((X - \mu)(Y - \nu))}{(n - 1)} \quad (4.2)$$

Die Variable n entspricht die Anzahl der Komponenten in X und in Y . Die Zwei konstanten μ und ν sind die Durchschnittswerte der beiden Variablen X und Y . Mit E ist der Erwartungswert des Produktes gegeben.

Die Kovarianzmatrix ist eine $p \times p$ symmetrische Matrix mit Einträgen als die Corianzwert für alle möglichen Paare, gebildet aus allen Variablen, in diesem Fall normierten Wortvektoren. Zur Darstellung betrachten wir einen Datensatz mit 3 variablen x , y und z . Die Kovarianzmatrix sieht wie folgt aus:

$$\begin{pmatrix} Cov(x, x) & Cov(x, y) & Cov(x, Z) \\ Cov(y, x) & Cov(y, y) & Cov(y, Z) \\ Cov(z, x) & Cov(z, y) & Cov(z, Z) \end{pmatrix} \quad (4.3)$$

In der Diagonale der Matrix stehen die Werte für Kovarianz der Variablen mit sich selbst. Dieser Wert entspricht der Varianz der Variable. Da die Kovarianz Kommutative ist, sind die obere und untere Dreiecksmatrix symmetrisch in bezug auf die Diagonale, beziehungsweise gleich.

3. Berechnung der Eigenvektors und Eigenwerte

Der nächste Schritt erfordert die Berechnung der Eigenvektors und Eigenwerte der Kovarianzmatrix. Auf diesem Weg bestimmen wir die gesuchten prinzipiellen Komponenten. Diese Komponenten werden als lineare Kombination oder Mischung der Ursprungsvariablen erstellt. Die neuen Variablen sind unabhängig von einander. Der Prozess versucht die meiste Informationen aus allen Variablen in die ersten prinzipiellen Komponenten zu beladen. Das erlaubt es die Dimensionen zu verringern, ohne große Mengen an Information zu verlieren. Es ist jedoch wichtig zu erwähnen, dass die prinzipiellen Komponenten nicht interpretierbar sind, da sie aus der Linearkombination der alten Variablen berechnet werden.

Geometrisch angesehen die prinzipiellen Komponenten sind Richtungen die einen maximalen Varianzwert darstellen. Das sind Geraden, die die meisten Punkte in einem n -Dimensionalen Raum beschreiben. Die Beziehung zwischen Varianz und Information ist es, dass je größer die Varianz bezüglich einer gegebenen Linie, desto mehr Punkten, bzw. Variable, entlang dieser Linie verteilt sind, umso mehr Information von dieser Linie getragen wird.

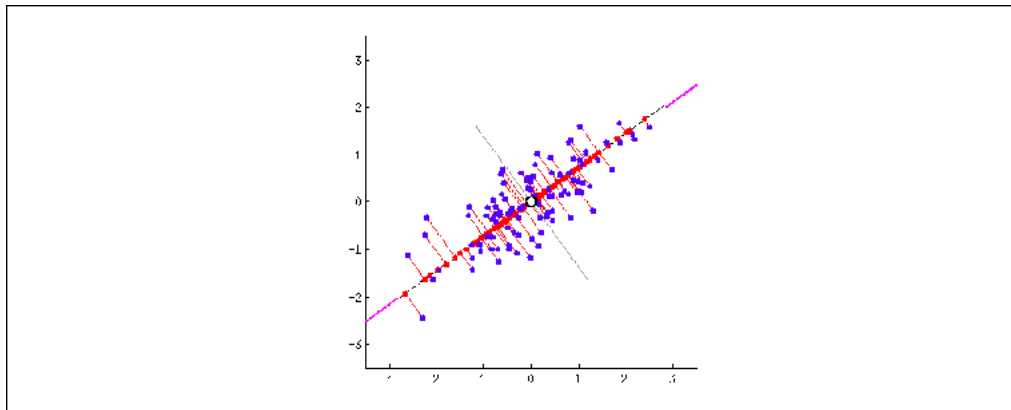


Abb. 4.1: Prinzipiele Linie

In der Abbildung 4.1 ist die Linie mit der größte Variation und so ist die die erste Prinzipielle Komponente (PK), da sie mit sich die meiste Information trägt. Falls die Variablen dann mit Hilfe dieser PK transformiert werden, wird die meiste Information übertragen. Nachdem die erste Komponente gewählt wird, wird die zweite auf den selben Prinzip gewählt, jedoch wird eine andere Linie gesucht, die dann unabhängig von der erste, meistens eine die Orthogonal zu der Erste liegt. Diese zweite besitzt entsprechend die zweitgrößte Varianz zu den Variablen aus der Datensammlung. Dieser Prozess wiederholt sich bis alle prinzipiellen Komponenten bestimmt sind, oder p oft - genau so oft wie wir Variablen in unsere Kovarianzmatrix haben.

Nun zurück zu den Eigenwerten und -vektoren. Zu Jedem Eigenwert gehört ein Vektor und umgekehrt. Sie kommen immer in Paare und ihre Anzahl entspricht die Dimensionalität der Matrix, wie schon oben erwähnt wurden. Ihre Beziehung in mathematischer Form kann wie folgt dargestellt werden:

$$Av = \lambda v, \quad (4.4)$$

wo A ist die Matrix, v der Eigenvektor und λ der zugehörige Eigenwert. Es existiert für eine quadratische Matrix einen Vektor v und einen Faktor λ , sodass bei der Multiplikation der Matrix mit dem Vektor, erhalten wir das gleiche Ergebnis, wie wenn wir den Vektor mit dem Faktor multiplizieren. Es kann die Formel in 4.4 umgeformt werden, um nun die Eigenvektoren und Eigenwerte zu berechnen:

$$(A - \lambda E).v = 0. \quad (4.5)$$

In der Gleichung entspricht E gleich der Einheitsmatrix. Der Eigenvektor ist Lösung der Gleichungssystem. Wir setzen für λ den entsprechenden Wert ein und lösen nach v . Jedoch muss der Eigenwert zuerst berechnet werden. Der wird aus der folgenden Formel errechnet:

$$\det(A - E) = 0 \quad (4.6)$$

Wo wir nach den Nullstellen der Determinante suchen. Diese Nullstellen sind die Eigenwerte der Matrix A .

Die Eigenvektoren bestimmen eigentlich die Richtung der Axen mit der meisten Information, oder auch Prinzipielle Komponenten genannt. Die Eigenwerte sind die Koeffizienten der Komponente. Je höher der Wert, desto mehr Information wird durch ihre Richtung repräsentiert. Falls die Eigenvektoren nach ihren Eigenwerten absteigend geordnet sind, werden die Ordnung der Prinzipiellen Komponenten bestimmt, sodass die erste Komponente entsprechend diese ist, die den größten Eigenwert besitzt. Als alle Vektoren angeordnet sind, wird ein neuer Vektor aus den zusammengesetzt. Jeder Eigenvektor ist eine Spalte im neuen Vektor. Als nächstes muss die Entscheidung getroffen werden, wie viele prinzipiellen Komponenten erhalten werden. Diese Frage hat keine richtige Antwort, in meinem Fall brauche ich nur zwei Komponenten, um die Daten in einem zweidimensionalen Raum darzustellen.

4. Transformation der Daten

Im letzten Schritt wird der Prozess abgeschlossen, indem die Ursprungsdaten nach den gefundenen Richtungen (*Prinzipiellen Komponenten*), transformiert werden. Die folgende Formel beschreibt die Operation:

$$T_L = V_L^T * Z^T, \quad (4.7)$$

wo T_L die transformierten, V enthält die L Prinzipiellen Komponenten und Z beinhaltet den standardisierten Datensatz. Die erhaltene Matrix T hat die gewünschte L Anzahl an Komponenten.

QUELLEN: <https://builtin.com/data-science/step-step-explanation-principal-component-analysis>

<https://royalsocietypublishing.org/doi/10.1098/rsta.2015.0202>

https://en.wikipedia.org/wiki/Principal_component_analysis

https://en.wikipedia.org/wiki/Sample_mean_and_covariance

<https://www.kaggle.com/jeffd23/visualizing-word-vectors-with-t-sne>

<https://towardsdatascience.com/visualizing-word-embedding-with-pca-and-t-sne-961a692509f5>

<https://towardsdatascience.com/visualization-of-word-embedding-vectors-using-gensim-and-pca-8f592a5d3354>

<https://studyflix.de/mathematik/eigenwert-1635>

Zusammenfassung und Ausblick

In diesem Kapitel soll die Arbeit noch einmal kurz zusammengefasst werden. Insbesondere sollen die wesentlichen Ergebnisse Ihrer Arbeit herausgehoben werden. Erfahrungen, die z.B. Benutzer mit der Mensch-Maschine-Schnittstelle gemacht haben oder Ergebnisse von Leistungsmessungen sollen an dieser Stelle prntiert werden. Sie knnen in diesem Kapitel auch die Ergebnisse oder das Arbeitsumfeld Ihrer Arbeit kritisch bewerten. Wnschenswerte Erweiterungen sollen als Hinweise auf weiterfhrende Arbeiten erwt werden.

A

Glossar

DisASter	DisASter (Distributed Algorithms Simulation Terrain), A platform for the Implementation of Distributed Algorithms
DSM	Distributed Shared Memory
AC	Linearisierbarkeit (atomic consistency)
SC	Sequentielle Konsistenz (sequential consistency)
WC	Schwache Konsistenz (weak consistency)
RC	Freigabekonsistenz (release consistency)

B

Erklng der Kandidatin / des Kandidaten

☐ Die Arbeit habe ich selbststtig verfasst und keine anderen als die angegebenen Quellen- und Hilfsmittel verwendet.

☐ Die Arbeit wurde als Gruppenarbeit angefertigt. Meine eigene Leistung ist ...

Diesen Teil habe ich selbststtig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet.

Datum

Unterschrift der Kandidatin / des Kandidaten