Methodische und Praktische Grundlagen der Informatik 4

Übungsblatt 5

mpgi4@cg.tu-berlin.de

WiSe 2013/2014

Zu Beginn, Diskussion des letzten Aufgabenblatts:

- Grundlegendes Verständnis von Python wurde vorausgesetzt. Dazu gab es die ersten Übungsblätter. Wenn diese bearbeitet wurden, dann war auch der Umfang des Übungsblatts angemessen.
- Wrapper-Funktionen: in der Praxis sehr häufig notwendig. In der Praxis ist es auch üblich, dass sich Spezifikationen mit der Zeit entwickeln.
- Lösungen sind relativ kompakt: entscheidend für diesen Kurs ist das Verständnis des Materials, hier zum Beispiel wie Gauss-Elimination funktioniert.
- Referenzlösungen werden am Montag den 18.11.2013 online gestellt.

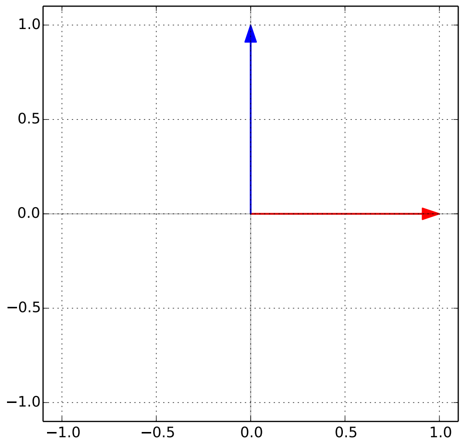
Aufgabe 1: Singulärwertzerlegung einer $2D$ Transformationsmatrix

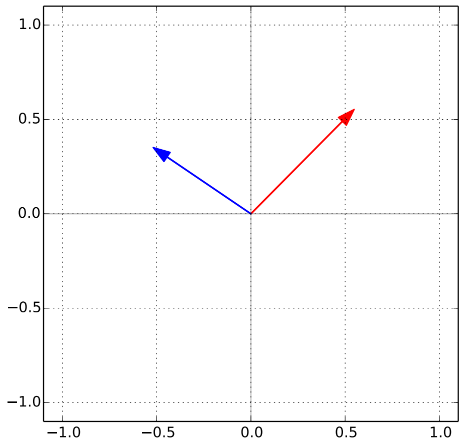
Ziel: Geometrische Bedeutung der Singulärwertzerlegung. Festigung des Verständnisses von Rotation und Skalierung aus der letzten Aufgabe, und Verbindung von SVD mit diesen.

Betrachten wir die 2D Transformationsmatrix:

$$A = \begin{pmatrix} 0.55 & -0.52 \\ 0.55 & 0.35 \end{pmatrix}. \tag{1}$$

Für die Standard-Basis (links) erzeugt diese folgende Transformation (rechts):





Dies kann als eine verallgemeinerte Scherungstransformation des Einheitsquadrates, welches durch die Standard-Basis aufgespannt wird, verstanden werden. Nach der Anwendung von A erhalten wir also ein verkleinertes, verzerrtes und rotiertes Einheitsquadrat.

Betrachten wir nun die Singulärwertzerlegung der Matrix A:

$$A = U S V^T = \begin{pmatrix} 0.87 & -0.50 \\ 0.50 & 0.87 \end{pmatrix} \begin{pmatrix} 0.80 & 0.0 \\ 0.0 & 0.60 \end{pmatrix} \begin{pmatrix} 0.94 & 0.34 \\ -0.34 & 0.94 \end{pmatrix}^T \tag{2}$$

Wir können A also als drei einzelne Transformationen auffassen. Für $y = Ax$ wird zunächst V^T auf x angewendet, dann S, und zuletzt U,

$$y_V = V^T x \tag{3a}$$
$$y_S = S\, y_V \tag{3b}$$
$$y = U\, y_S. \tag{3c}$$

Wir werden nun den Effekt der einzelnen Transformationen im Detail betrachten.

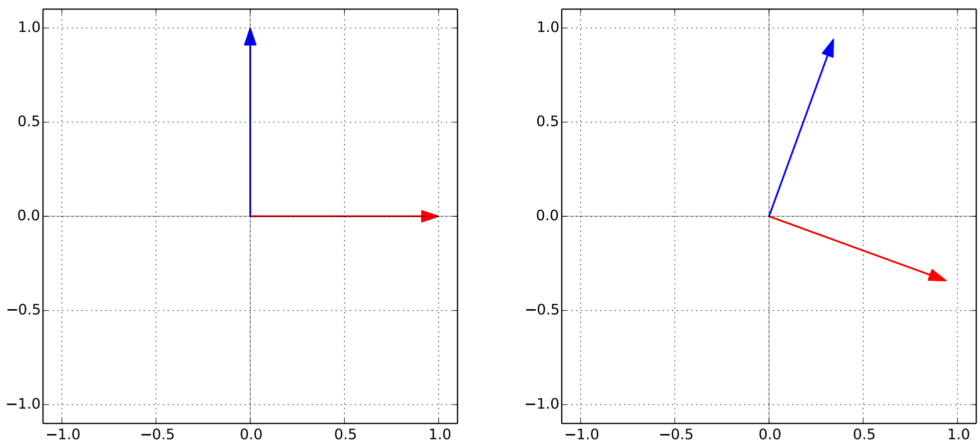
Die Matrix V, und damit auch V^T, ist bei "Konstruktion" der Singulärwertzerlegung eine orthogonale Matrix, d.h.

$$V^T V = \begin{pmatrix} 0.94 & -0.34 \\ 0.34 & 0.94 \end{pmatrix} \begin{pmatrix} 0.94 & 0.34 \\ -0.34 & 0.94 \end{pmatrix} = \begin{pmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{pmatrix} = \mathrm{Id} \tag{4}$$

Wenn wir das Matrixprodukt näher betrachten, dann berechnen wir in Gleichung 4 vier Skalarprodukte, welche die Elemente der Identitätsmatrix formen:

$$\mathrm{Id}_{ij} = \langle v_i, v_j \rangle = \begin{cases} 1 & \text{wenn } i = j \\ 0 & \text{ansonsten} \end{cases} \tag{5}$$

wobei v_i die i-te Spalte von V ist, also $v_1 = (0.94, -0.34)$ und $v_2 = (0.34, 0.94)$. Da $\langle v_1, v_2 \rangle = 0$ sind die Vektoren orthogonal und zwei orthogonale Vektoren formen im $\mathbb{R}^2$ notwendiger Weise eine orthogonale Basis für den Vektorraum. Auf der rechten Seite in der folgenden Abbildung sind die Spalten von V als Vektoren dargestellt:

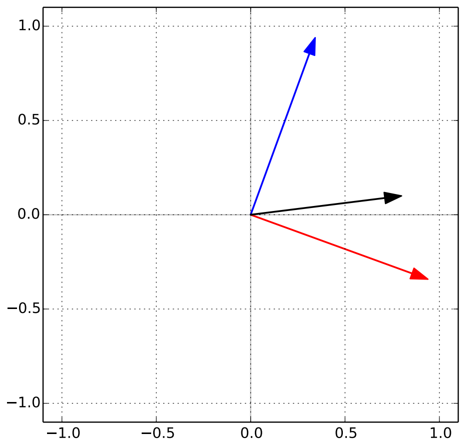


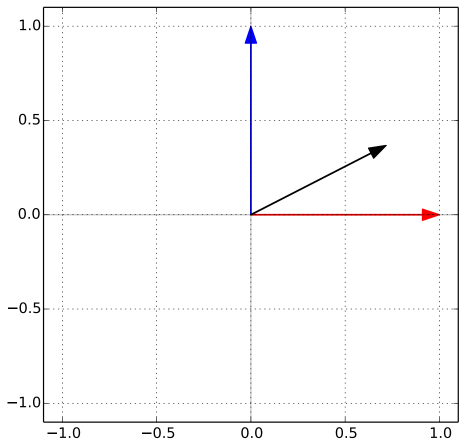
Wir sehen also, dass wir (v_1, v_2) als eine rotierte Version der Standardbasis (e_1, e_2) auffassen können. In der Tat haben wir für unser Beispiel (bei Konstruktion)

$$V^T = \begin{pmatrix} 0.94 & 0.34 \\ -0.34 & 0.94 \end{pmatrix} = \begin{pmatrix} \cos(-20.0°) & -\sin(-20.0°) \\ \sin(-20.0°) & \cos(-20.0°) \end{pmatrix} = R(-20.0°) \tag{6}$$

d.h. V^T ist die Rotationsmatrix $R(-20.0°)$ um $-20.0°$ im Uhrzeigersinn. Die Element einer Rotationsmatrix $R(\theta)$ sind also nicht nur durch $\sin(\theta), \cos(\theta)$ gegeben, sondern sie sind auch die kartesischen Koordinaten der rotierten Basis in der Standardbasis (e_1, e_2) ausgedrückt.

Wir haben bereits in den vorhergehenden Tutorien besprochen, dass Vektoren geometrische Objekte sind, die unabhängig von ihrer Beschreibung in einem Koordinatensystem sind: was sich ändert, wenn man ein anderes Koordinatensystem verwendet sind die Koordinaten, zum Beispiel a_1, a_2, welche den Vektor bezüglich der Basis beschreiben. Für einen gegebenen, fixen Vektor erhalten wir im rotierten Koordinatensystem also neue Koordinaten. Allerdings ist geometrisch auch einfach zu sehen, dass die Beziehung des fixen Vektors zum um den Winkel $-\theta$ rotierten Koordinatensystem die gleiche ist, wie die eines um θ gedrehten Winkels zu einem fixen Koordinatensystem:





Aber wenn die geometrische Beziehung die gleiche ist, dann müssen auch die Koordinaten in beiden Fällen übereinstimmen. Die Elemente des rotierten Vektors, oder des fixen Vektors bezüglich des inversen rotierten Koordinatensystems, sind also durch

$$R(\theta)\vec{a} = \vec{a}_\theta \tag{7}$$

gegeben.

Wir können uns den Zusammenhang zwischen Rotation eines Vektors $\vec{a}$ und den Koordinaten des rotierten Koordinatensystems $(\vec{v}_1, \vec{v}_2)$ auch noch etwas mathematischer vor Augen führen. Bei Definition sind die Komponenten von $\vec{a}$ bezüglich v_i durch das Skalarprodukt gegeben:

a) Warum ist dies der Fall? Wenn wir einen beliebigen Vektor $\vec{a}$ in die Basis $(\vec{v}_1, \vec{v}_2)$, welche von den Spalten von V aufgespannt wird, projezieren, dann müssen wir $\langle \vec{a}, \vec{v}_i \rangle$ berechnen. Berechnung durchführen und zeigen, wie dies zu den Elementen von R führt, wenn $\vec{a}$ und $\vec{v}_i$ in der Standardbasis dargestellt werden.

b) Rotation um θ oder Basiswechsel in ein Koordinatensystem, welches um $-\theta$ gedreht ist. Für die SVD wird oft die zweite Perspektive bevorzugt, insbesondere im Kontext von PCA.

c) S ist eine Skalierungsmatrix, wie man einfach bei einer Anwendung auf die Einheitsvektoren sehen kann.

d) Wir können uns V^T also als die Wahl eines Koordinatensystems vorstellen, in der der Effekt der ursprünglichen Matrix A eine einfache Skalierung ist.

e) Und was ist mit U? U ist wieder eine Rotation des skalierten Vektors. Natürlich können wir das auch wieder als einen Wechsel des Koordinatensystems vorstellen.

f) Qualitative Diskussion des Falls wenn A nicht quadratisch ist: zunächst zeigen, wie eine Projektion auf die X-Achse als Matrix dargestellt werden kann. Dann Struktur der SVD für ein 2×3 Matrix zeigen:

$$C = USV^T = \begin{pmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \end{pmatrix} = \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix} \begin{pmatrix} s_{11} & 0.0 & 0.0 \\ 0.0 & 0.0 & s_{22} \end{pmatrix} \begin{pmatrix} v_{11} & v_{12} & v_{13} \\ v_{21} & v_{22} & v_{23} \\ v_{31} & v_{32} & v_{33} \end{pmatrix} \tag{8}$$

V stellt dann eine orthogonale Basis für die Eingangsdaten dar und U für die Ausgangsdaten.

g) Alle Ideen funktionieren konzeptionell genauso in R^n und Vektoren nicht mehr geometrische Objekte sondern beliebige Daten darstellen.

h) Zusammenfassung: V und U stellen Koordinatensystem dar, welche

Aufgabe 2: PCA für Punktwolken

Ziel: Die Aufstellung einer Datenmatrix und die notwendige Schritte um eine principal component analysis durchzuführen, sollen besprochen werden.

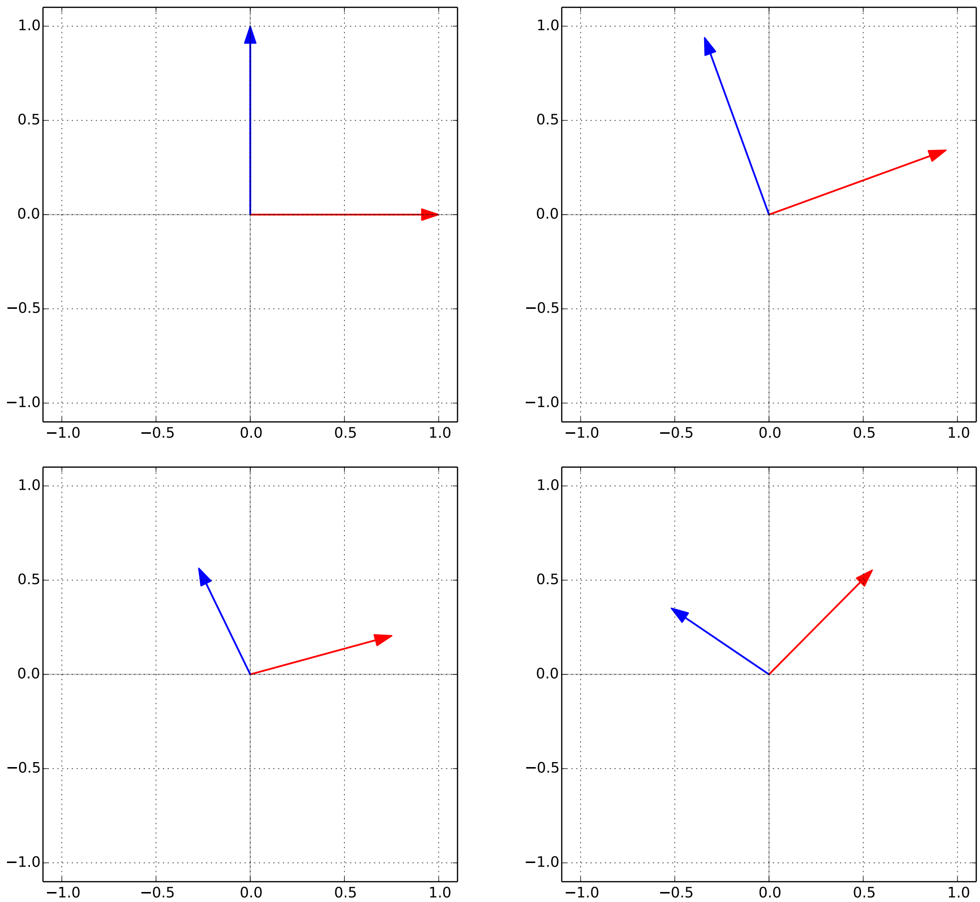


Abbildung 1: Zerlegung einer Transformation der Euklidischen Standardbasis (links oben) durch die SVD: Im ersten Schritt erfolgt eine Rotation durch V^T. Im zweiten Schritt führt S eine Skalierung der rotierten Vektoren durch. Zuletzt erfolgt eine erneute Rotation durch U.

Eine Implementierung liegt in `matching.py` vor. Betonung insbesondere auf die Erstellung der Datenmatrix und die Interpretation Spalten von V (Achtung: numpy.linalg.svd() gibt V^T zurück) als die Richtungen, welche das Datenset aufspannen: jeder Datenpunkt kann im PCA Koordinatensystem dargestellt werden. Singulärwerte geben die Wichtigkeit der verschiedenen Richtungen an. Für viele Anwendungen, wie Eigenfaces, hat man Richtungen mit sehr kleinem Singulärwert. Dann können diese Dimensionen vernachlässigt werden, und dies führt zur Dimensionsreduktion bei der man Speicherplatz sparen kann. Dies ist sehr ähnlich zur Eigenface-Aufgabe auf dem Aufgabenblatt. Keine größere Betonung auf das Alignment der Achsen. Wenn man die principal components für beide Datensätze hat, dann kann man die Länge der Vektoren vergleichen, welche den (anisotropischen) Varianzen entsprechen (eine intuitive Diskussion ist hier ausreichend).