Hauptkomponentenanalyse (PCA)

Autor und Herausgeber

Alexander Jung*, Department of Computer Science, Aalto University, Espoo, Finland (ORCID 0000-0001-7538-0990)
Editors: Konstantina Olioumtsevits (ORCID 0000-0003-2697-8659), Ekkehard Schnoor (ORCID 0000-0001-8252-1499), Aalto University, Espoo, Finland
*Corresponding author: alex.jung@aalto.fi

Zusammenfassung

Die Hauptkomponentenanalyse (PCA) ist der lineare Spezialfall des Repräsentationslernens: Sie bildet den Merkmalsvektor eines Datenpunkts durch eine lineare Abbildung auf einen Vektor mit weniger Einträgen ab, die so gewählt wird, dass eine zweite lineare Abbildung das Original so gut wie möglich rekonstruiert. Das Kriterium ist der Rekonstruktionsfehler auf einem Datensatz, dessen Labels, sofern vorhanden, nicht verwendet werden, da der Datenpunkt selbst reproduziert werden muss; dies macht die PCA zu einer Instanz der empirische Risikominimierung (ERM) mit dem quadratischer Verlust. Die lineare Abbildung, die diesen Fehler minimiert, ist durch die Eigenvektoren der Stichproben-Kovarianzmatrix gegeben, die zu ihren größten Eigenwerte gehören, sodass die PCA durch eine Eigenwertzerlegung berechnet wird. Der verbleibende Fehler ist die Summe der verworfenen Eigenwerte, wodurch der Fehler für jede Wahl der Anzahl beibehaltener Richtungen im Voraus feststeht. Jeder Eigenwert ist die Varianz entlang seines Eigenvektors, sodass die erste Richtung diejenige mit der größten Varianz ist. Ein Autoencoder behält dasselbe Kriterium bei, setzt jedoch künstliche neuronale Netze (KNN) an die Stelle der beiden lineare Abbildungen.

Definition

Eine Wetterstation in Krems an der Donau zeichnet für jeden Tag des Jahres 2024 acht Werte auf: die tiefste, höchste und mittlere Lufttemperatur, den Niederschlag, die Sonnenscheindauer, die Luftfeuchtigkeit, den Luftdruck und die Windgeschwindigkeit. Acht Zahlen pro Tag sind mehr, als ein Streudiagramm darstellen kann, und eine davon, die Mitteltemperatur, ist bis auf die Genauigkeit des Archivs der Mittelpunkt zweier anderer. Die PCA ersetzt die acht Zahlen durch weniger, die so gewählt sind, dass eine zweite lineare Abbildung die ursprünglichen Werte aus den wenigen mit dem kleinsten mittleren quadratischen Fehler rekonstruiert.

Die PCA ist der lineare Spezialfall des Repräsentationslernens, und ihr Kriterium ist der Rekonstruktionsfehler. Der Encoder ist eine lineare Abbildung

$$\boldsymbol{\Phi}^{(\mathbf{W})} : \mathbb{R}^{d} \to \mathbb{R}^{d'} : \mathbf{x} \mapsto \mathbf{W}\mathbf{x},$$

gegeben durch eine Matrix $\mathbf{W} \in \mathbb{R}^{d' \times d}$ mit $d' < d$, und der Decoder ist eine lineare Abbildung $\mathbf{R} \in \mathbb{R}^{d \times d'}$, die den ursprünglichen Merkmalsvektor aus dem *Code* rekonstruiert, der niedrigdimensionalen Darstellung $\mathbf{z} = \mathbf{W}\mathbf{x}$ des Datenpunkts. Der Nutzen eines bestimmten $\mathbf{W}$ bemisst sich nach dem Rekonstruktionsfehler im Minimum, den es auf einem Datensatz $\mathcal{D} = \left\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}\right\}$ aus Merkmalsvektoren $\mathbf{x}^{(r)} \in \mathbb{R}^d$ mit dem Stichprobenmittel null hinterlässt, wobei der Decoder gemeinsam mit dem Encoder gelernt wird (Bishop, 2006; Hastie et al., 2009, Sect. 14.5.1; Hotelling, 1933):

$$\min_{\mathbf{R} \in \mathbb{R}^{d \times d'}} \frac{1}{m} \sum_{r=1}^{m} \left\| \mathbf{x}^{(r)} - \mathbf{R}\mathbf{W}\mathbf{x}^{(r)} \right\|_2^2. \tag{1}$$

Eine gemeinsame Minimierung über $\mathbf{R}$ und $\mathbf{W}$ liefert dasselbe Paar; das innere Minimum wird zuerst gebildet, damit der Rekonstruktionsfehler zu einer Verlustfunktion allein von $\mathbf{W}$ wird. Im Rekonstruktionsfehler (1) treten keine Labels auf: Der Decoder muss den Datenpunkt selbst reproduzieren. Die PCA ist daher eine Instanz der empirische Risikominimierung (ERM), deren Verlustfunktion der quadratischer Verlust $L(\mathbf{x}, \mathbf{W}) = \left\| \mathbf{x} - \widehat{\mathbf{R}}\mathbf{W}\mathbf{x} \right\|_2^2$ ist, für den Decoder $\widehat{\mathbf{R}}$, der das Minimum annimmt.

Die Eigenvektoren der Stichproben-Kovarianzmatrix lösen das Rekonstruktionsproblem (1). Die Merkmalsvektoren sind zentriert, d. h., ihr Stichprobenmittel ist null, wodurch die folgende Matrix zur Stichproben-Kovarianzmatrix wird:

$$\widehat{\mathbf{Q}} = \frac{1}{m} \sum_{r=1}^{m} \mathbf{x}^{(r)} \left(\mathbf{x}^{(r)}\right)^\top,$$

und sie ist positiv semidefinit, da $\mathbf{u}^\top \widehat{\mathbf{Q}}\mathbf{u} = \frac{1}{m} \sum_{r=1}^{m} \left(\mathbf{u}^\top \mathbf{x}^{(r)}\right)^2 \geq 0$ für jedes $\mathbf{u} \in \mathbb{R}^d$ gilt. Eine symmetrische Matrix besitzt eine orthonormale Basis aus Eigenvektoren (Horn and Johnson, 2013; Meyer, 2000, p. 549), und eine positiv semidefinite hat nichtnegative Eigenwerte (Meyer, 2000, Sect. 7.6, p. 558), sodass die Eigenwertzerlegung von $\widehat{\mathbf{Q}}$ eine Spektralzerlegung ist, $\widehat{\mathbf{Q}} = \sum_{j=1}^{d} \lambda_j \mathbf{u}^{(j)} \left(\mathbf{u}^{(j)}\right)^\top$ mit den Eigenwerte $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$ und den Eigenvektoren $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(d)}$, die eine orthonormale Basis des $\mathbb{R}^d$ bilden.

Die Einschränkung des Decoders $\mathbf{R}$ auf orthonormale Spalten bedeutet keinen Verlust an Allgemeinheit, da die Produkte $\mathbf{RW}$ ohnehin die Matrizen mit einem Rang von höchstens d' durchlaufen. Für ein solches $\mathbf{R}$ ist der Encoder, der das Minimum annimmt, die Transponierte $\mathbf{W} = \mathbf{R}^\top$ (Goodfellow et al., 2016, Sect. 2.12, Eq. (2.66); Hastie et al., 2009, Sect. 14.5.1, Eq. (14.52)): Der beste Code enthält die Skalarprodukte $\mathbf{R}^\top \mathbf{x}$ von $\mathbf{x}$ mit den orthonormalen Spalten von $\mathbf{R}$. Die Rekonstruktion $\mathbf{R}\mathbf{R}^\top \mathbf{x}$ ist daher die orthogonale Projektion von $\mathbf{x}$ auf den von ihnen aufgespannten Unterraum. Diese Projektion ist orthogonal zum verbleibenden Fehler, und es gilt

$$\left\| \mathbf{x} - \mathbf{R}\mathbf{R}^\top \mathbf{x} \right\|_2^2 = \|\mathbf{x}\|_2^2 - \left\| \mathbf{R}^\top \mathbf{x} \right\|_2^2.$$

Durch Mittelung über alle Datenpunkte in $\mathcal{D}$ geht der Rekonstruktionsfehler (1) in $\operatorname{tr}\left(\widehat{\mathbf{Q}}\right) - \operatorname{tr}\left(\mathbf{R}^\top \widehat{\mathbf{Q}} \mathbf{R}\right)$ über. Die erste Spur hängt nicht von $\mathbf{R}$ ab, sodass die Minimierung des Fehlers der Maximierung der zweiten über Matrizen mit orthonormalen Spalten entspricht. Dieses Maximum beträgt $\sum_{j=1}^{d'} \lambda_j$ und wird von den Eigenvektoren angenommen, die zu den d' größten Eigenwerte gehören (Goodfellow et al., 2016, Sect. 2.12, Eq. (2.84); Horn and Johnson, 2013, Cor. 4.3.39). Eine Matrix $\mathbf{W} = \left(\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(d')}\right)^\top$, deren Zeilen diese Eigenvektoren sind, minimiert daher im Paar mit dem Decoder $\widehat{\mathbf{R}} = \mathbf{W}^\top$ das Kriterium (1). Der Rekonstruktionsfehler im Minimum ist $\sum_{j=d'+1}^{d} \lambda_j$, die Summe der ausgelassenen Eigenwerte, sodass das Spektrum von $\widehat{\mathbf{Q}}$ den Rekonstruktionsfehler für jede Codelänge im Voraus angibt. Algorithmus 1 gibt die vollständige Berechnung an.

Algorithm 1 Hauptkomponentenanalyse (Bishop, 2006, Abschn. 12.1.2)

Input: Merkmalsvektoren $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)} \in \mathbb{R}^d$, number $d' \leq d$ of retained Merkmale

1: center the Merkmalsvektoren: subtract their Stichprobenmittel from each of them
2: form the Stichproben-Kovarianzmatrix $\widehat{\mathbf{Q}}$ of the centered Merkmalsvektoren
3: compute the Eigenwertzerlegung of $\widehat{\mathbf{Q}}$, giving Eigenwerte $\lambda_1 \geq \ldots \geq \lambda_d$ and an orthonormal basis of Eigenvektoren $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(d)}$
4: stack the Eigenvektoren belonging to the d' largest Eigenwerte as the rows of $\mathbf{W} = \left(\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(d')}\right)^\top$

Output: the Encoder $\mathbf{W}$ and the Decoder $\widehat{\mathbf{R}} = \mathbf{W}^\top$; the code of a Merkmalsvektor is $\mathbf{W}\mathbf{x}$ and its reconstruction $\mathbf{W}^\top\mathbf{W}\mathbf{x}$

Die Eigenvektoren $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(d)}$ von $\widehat{\mathbf{Q}}$ sind die *Hauptrichtungen* des Datensatzes, geordnet nach absteigendem Eigenwert. Stapelt man die zentrierten Merkmalsvektoren zeilenweise in die Merkmalsmatrix $\mathbf{X} \in \mathbb{R}^{m \times d}$, so schreibt sich die Stichproben-Kovarianzmatrix als $\widehat{\mathbf{Q}} = (1/m)\mathbf{X}^\top\mathbf{X}$, sodass sie ebenso die Eigenvektoren von $\mathbf{X}^\top\mathbf{X}$ sind. Die ersten d' unter ihnen spannen den Unterraum auf, den die PCA beibehält, und jeder Merkmalsvektor wird durch seine orthogonale Projektion auf diesen ersetzt. Diese Projektion ist die Approximation, deren quadratischer Verlust der Rekonstruktionsfehler (1) misst. Die Skalarprodukte von $\mathbf{x}$ mit den ersten d' Hauptrichtungen sind seine Koordinaten in diesem Unterraum, und sie bilden die neuen Merkmale, welche die PCA liefert.

Die Eigenwerte von $\widehat{\mathbf{Q}}$ sind die Varianzen der Daten entlang ihrer Eigenvektoren, sodass $\mathbf{u}^{(1)}$ die Richtung der größten Varianz ist. Abb. 1 zeigt die beiden Richtungen für die Tage in Krems anhand zweier Merkmale, der tiefsten und der höchsten Tagestemperatur (beide zentriert). Ihre Eigenwerte betragen 130.9 und 7.9 Grad zum Quadrat, deren Quadratwurzeln die Standardabweichungen entlang dieser Richtungen sind.

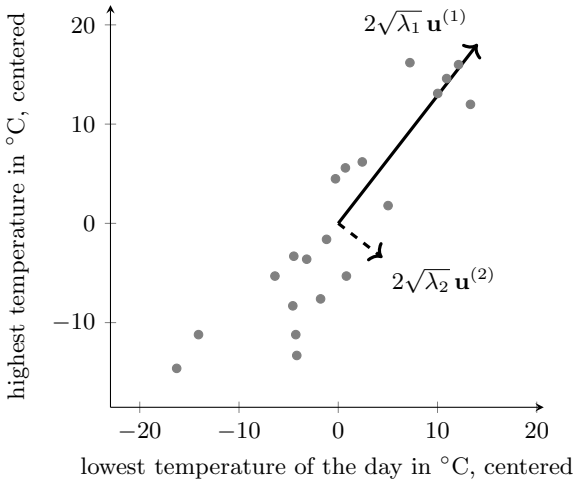


Abbildung 1: Zwanzig der 366 Tage des Jahres 2024 in Krems an der Donau (jeder 19. Tag), jeweils als grauer Punkt eingezeichnet, ein Datenpunkt mit der Tiefst- und der Höchsttemperatur des Tages als seinen beiden Merkmale, zentriert. Die Pfeile sind die Eigenvektoren der Stichproben-Kovarianzmatrix aller 366 Tage ($\mathbf{u}^{(1)}$ durchgezogener und $\mathbf{u}^{(2)}$ gestrichelter Pfeil), jeweils mit der Länge von zwei Standardabweichungen der Daten gezeichnet, sodass der längere Pfeil die Richtung größerer Varianz markiert. Behält man nur $\mathbf{u}^{(1)}$ bei, wird jeder Tag durch seine Position auf diesem Pfeil ersetzt. Daten erzeugt mit `pythondemos/pca.py`

Über alle acht Messgrößen hinweg, die jeweils auf eine Stichprobenvarianz von eins skaliert wurden, da Temperaturen, Niederschlag und Luftdruck in unterschiedlichen Einheiten gemessen werden, tragen die beiden größten Eigenwerte 64% der gesamten Varianz. Ein Code aus zwei Zahlen hinterlässt einen Rekonstruktionsfehler von 2.91, genau die Summe der sechs verworfenen Eigenwerte. Abb. 2 stellt den verbleibenden Fehler für jede Codelänge dar.
Die Codelänge d' ist ein Hyperparameter. Jede zum Code hinzugefügte Zahl zieht genau den nächsten Eigenwert vom Fehler ab, und die zuerst abgezogenen sind die größten (siehe Abb. 2). Keine Wahl ist bezüglich des Kriteriums selbst optimal, da dieses bei $d' = d$, wo nichts komprimiert wird, auf null fällt. Was d' festlegt, ist der Verwendungszweck des Codes: Ein Streudiagramm benötigt zwei, einer nachgelagerten Lernaufgabe kann der Validierungsfehler jeder Länge zur Auswahl gestellt werden, und ein Speicherbudget gibt sie direkt vor.

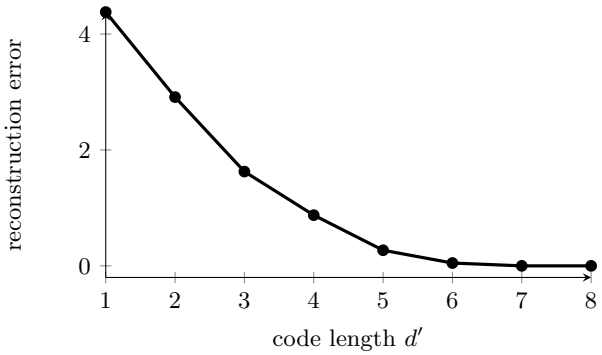
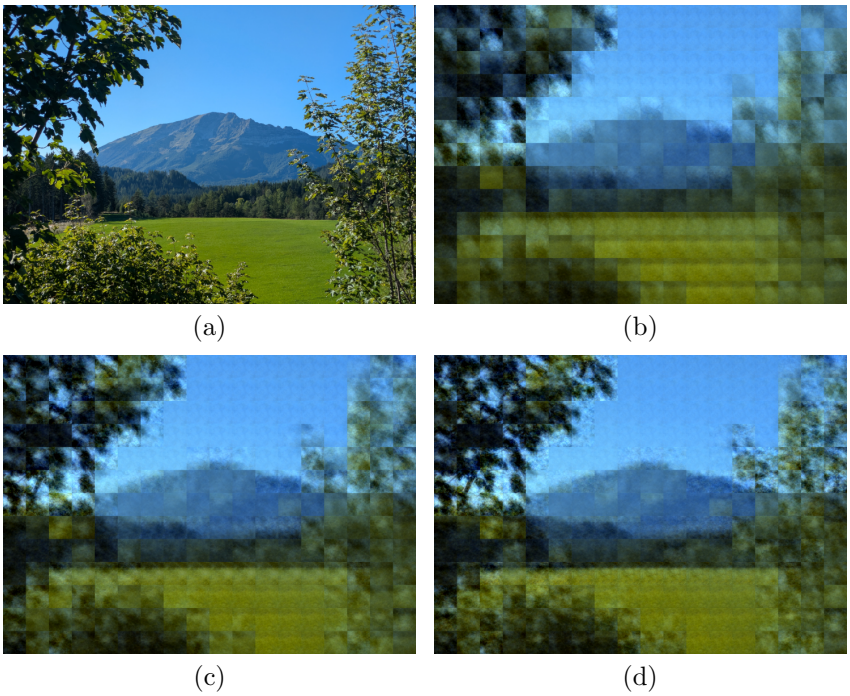


Abbildung 2: Der verbleibende Rekonstruktionsfehler bei einem Code aus d' Zahlen auf den acht auf eine Stichprobenvarianz von eins skalierten Messgrößen aus Krems an der Donau. Er ist die Summe der Eigenwerte, die der Code verwirft, fällt also zu Beginn am schnellsten ab und erreicht null erst, wenn nichts komprimiert wird. Daten erzeugt mit `pythondemos/pca.py`

Insbesondere verrät bei denselben acht Messgrößen der kleinste Eigenwert ebenso viel wie der größte. Er beträgt $6.5{\cdot}10^{-6}$ gegenüber 3.62 beim größten, und sein Eigenvektor lädt allein auf den drei Temperaturen: Die Mitteltemperatur eines Tages ist der Mittelpunkt seiner Tiefst- und seiner Höchsttemperatur, bis auf die 0.05 Grad genau, die das Archiv verzeichnet. Eine Richtung mit fast keiner Varianz ist eine Abhängigkeit zwischen den Merkmale, und die PCA findet sie ohne einen eigenen Test dafür. Das Auffinden solcher Abhängigkeiten macht die PCA zu einer Methode der Dimensionsreduktion und nicht nur des Repräsentationslernens: Die Merkmale, die sie verwirft, sind diejenigen, die durch die anderen bereits bestimmt sind.

Diese Zahlen sind Mittelwerte über den Datensatz, aus dem die Eigenvektoren berechnet wurden. Ob der Unterraum auch Datenpunkte außerhalb davon rekonstruiert, ist die Frage der Generalisierung. Die Antwort hängt davon ab, wie gut $\widehat{\mathbf{Q}}$ die Kovarianzmatrix von Datenpunkte außerhalb des Datensatzes schätzt.

Dasselbe Kriterium komprimiert Bilder. Schneidet man eine Fotografie in Ausschnitte von 64×64 Pixeln, wird jeder Ausschnitt zu einem Datenpunkt mit 12288 Merkmale, drei Farbwerten pro Pixel, und die PCA ersetzt diese durch einen Code aus d' Zahlen (siehe Abb. 3). Bei $d' = 5$ sind der Berg und die Wiese bereits an ihrem Platz, während die Grenzen der Ausschnitte noch sichtbar sind; bei 15 kommt die Rekonstruktion dem Original nahe.



(a) (b) (c) (d)

Abbildung 3: PCA auf den 234 Ausschnitten einer Fotografie des Ötschermassivs, wobei jeder Ausschnitt ein Datenpunkt ist. (a) Das Original. (b), (c), (d) Jeder Ausschnitt durch einen Code aus $d' = 5$, 10 und 15 Zahlen geschickt und rekonstruiert, wobei 29%, 24% und 21% der gesamten Varianz unerklärt bleiben. Daten erzeugt mit `pythondemos/pca.py`

Drei Varianten der PCA ändern jeweils einen Aspekt. Die probabilistische Hauptkomponentenanalyse (probabilistische PCA) behält den Unterraum bei, ersetzt aber das Kriterium durch ein Wahrscheinlichkeitsmodell. Der Code ist ein Gaußsche Zufallsvariable, und der Datenpunkt ist dessen lineares Bild plus isotropes gaußsches Rauschen. Die mittels EM-Algorithmus (EM) berechnete maximale Likelihood gewinnt den Unterraum der PCA zurück, wenn die Varianz des Rauschens gegen null geht (Tipping and Bishop, 1999, Abschn. 3.2, Gl. (7) und Abschn. 3.4).

Die Sparse PCA behält das Kriterium bei, fügt aber einen Strafterm auf die Einträge (Ladungen) eines Eigenvektors hinzu. Jede Richtung umfasst dann nur wenige Merkmale und lässt sich als benannte Kombination aus ihnen interpretieren. Bei den acht Messgrößen erhalten vier Ladungen, die drei Temperaturen und die Sonnenscheindauer, 81% der Varianz von $\mathbf{u}^{(1)}$ (Zou et al., 2006).

Die robuste Hauptkomponentenanalyse (robuste PCA) ersetzt den quadratischer Verlust, den ein einziger stark verfälschter Eintrag dominieren kann, durch eine Zerlegung der Daten-Matrix in einen Anteil mit niedrigem Rang plus einen Anteil mit wenigen von null verschiedenen Einträgen. Eine kleine Anzahl feh-

lerhafter Messungen fällt dann in den zweiten Anteil, anstatt die Richtungen zu kippen (Candès et al., 2011).

Ein Autoencoder behält den Rekonstruktionsfehler (1) als Kriterium bei und ersetzt die beiden lineare Abbildungen durch künstliche neuronale Netze (KNN), was das nichtlineare Gegenstück zur PCA darstellt. Mit einem linearen Decoder und dem quadratischer Verlust rekonstruiert er in denselben Unterraum wie die PCA (Goodfellow et al., 2016, Abschn. 14.1).

Siehe auch: Repräsentationslernen, Merkmalstransformation, Spektrum, Encoder, Decoder, Autoencoder, Dimensionsreduktion, Eigenwertzerlegung, Spektralzerlegung, Stichproben-Kovarianzmatrix, Eigenvektor, Eigenwert, ERM, Unterraum, probabilistische PCA, Sparse PCA, robuste PCA.

Querverweise

Literatur

1. Bishop (2006). Pattern Recognition and Machine Learning. Springer Science+Business Media, New York, NY, USA.

2. Candès et al. (2011). Robust Principal Component Analysis?. Journal of the ACM, 58(3), pp. 11:1–11:37.

3. Goodfellow et al. (2016). Deep Learning. MIT Press, Cambridge, MA, USA.

4. Horn and Johnson (2013). Matrix Analysis. 2nd ed., Cambridge Univ. Press, New York, NY, USA.

5. Hotelling (1933). Analysis of a Complex of Statistical Variables into Principal Components. Journal of Educational Psychology, 24(6), pp. 417–441.

6. Meyer (2000). Matrix Analysis and Applied Linear Algebra. SIAM, Philadelphia, PA, USA.

7. Tipping and Bishop (1999). Probabilistic Principal Component Analysis. J. Roy. Statist. Soc.: Ser. B (Statist. Methodology), 61(3), pp. 611–622.

8. Zou et al. (2006). Sparse Principal Component Analysis. Journal of Computational and Graphical Statistics, 15(2), pp. 265–286.

9. Hastie et al. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed., Springer Science+Business Media, New York, NY, USA.