robuste Hauptkomponentenanalyse (robuste 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 robuste Hauptkomponentenanalyse (robuste PCA) zerlegt eine Merkmalsmatrix in eine niedrig-rank Matrix, deren Zeilen in einem Unterraum kleiner Dimension liegen, und eine dünnbesetzte Matrix, die einige fehlerhafte Merkmal-Werte enthält. Anders als die Hauptkomponentenanalyse (PCA) wird die niedrig-rank Matrix nicht aus den aufgezeichneten Werten berechnet, sodass ein großer Fehler in der dünnbesetzten Matrix landet, anstatt den Unterraum zu verdrehen. Ein Ansatz formuliert die robuste PCA als ein konvex Optimierungsproblem, das als principal component pursuit bezeichnet wird. Konvex-optimization-Verfahren lösen dieses Problem, darunter der alternating direction method of multipliers (ADMM). Das resultierende Verfahren liefert Schätzungen für die Hauptkomponentenrichtungen, die den Unterraum aufspannen, sowie für die fehlerhaften Einträge in den Merkmale jedes Datenpunkt.

Definition

Die Wetterstation Krems an der Donau zeichnet die Lufttemperatur stündlich auf, sodass zehn Tage des Jahres 2024 zehn Zeilen mit jeweils 24 Werten ergeben (Abb. 1). Das Stapeln der 24 Werte eines Tages ergibt die Merkmalsvektor $\mathbf{x} \in \mathbb{R}^d$ mit $d = 24$ des zugehörigen Datenpunkt. Das zeilenweise Stapeln der $m = 10$ Merkmalsvektoren ergibt die $m \times d$ Merkmalsmatrix $\mathbf{X}$. Jede Merkmalsvektor wird zentriert, indem der Mittelwert der m Merkmalsvektoren abgezogen wird, sodass dieser Mittelwert der Null- Vektor ist.

Die Hauptkomponentenanalyse (PCA) verwendet die Merkmalsmatrix, um eine vorgegebene Anzahl d' von orthonormal Vektoren $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(d')}$, die Hauptkomponentenrichtungen von $\mathbf{X}$, zu lernen. Diese spannen einen d'-dimensionalen Unterraum $\mathcal{U}$ von $\mathbb{R}^d$ auf, sodass jede Merkmalsvektor $\mathbf{x}^{(r)}$ für $r = 1, \ldots, m$ gut durch ihre Projektion

$$\mathbf{P}\mathbf{x}^{(r)} = \sum_{j=1}^{d'} \langle \mathbf{u}^{(j)}, \mathbf{x}^{(r)} \rangle \mathbf{u}^{(j)}.$$

approximiert werden kann. Hierbei bezeichnet $\mathbf{P} \in \mathbb{R}^{d\times d}$ die orthogonale Projektion-Matrix auf den Unterraum $\mathcal{U}$. Sie ist explizit gegeben durch $\mathbf{P} = \sum_{j=1}^{d'} \mathbf{u}^{(j)}\big(\mathbf{u}^{(j)}\big)^\top$ und liefert eine vollständig äquivalente Darstellung des von den Hauptkomponentenrichtungen aufgespannten Subraums $\mathcal{U}$.

Die von der PCA gelieferte Projektion-Matrix $\mathbf{P}$ zerlegt jede Merkmalsvektor als $\mathbf{x}^{(r)} = \mathbf{P}\mathbf{x}^{(r)} + \mathbf{e}^{(r)}$, wobei $\mathbf{e}^{(r)}$ den Approximationsfehler bezeichnet. Diese Zerlegungen (jeweils für $r = 1, \ldots, m$) lassen sich kompakter mit der Merkmalsmatrix schreiben als

$$\mathbf{X} = \mathbf{L} + \mathbf{E}, \text{ wobei } \mathbf{L} = \mathbf{X}\mathbf{P} \text{ und } \mathbf{E} = \mathbf{X} - \mathbf{L}. \tag{1}$$

Der rank von $\mathbf{L}$ ist höchstens d', da jede Zeile von $\mathbf{L}$ im Unterraum $\mathcal{U}$ liegt. Die Fehler-Matrix $\mathbf{E}$ enthält die $\mathbf{e}^{(r)}$ als Zeilen, sodass $\|\mathbf{E}\|_F^2 = \sum_{r=1}^{m} \big\|\mathbf{e}^{(r)}\big\|_2^2$ gilt. Die PCA wählt $\mathbf{P}$ so, dass diese Größe möglichst klein wird. In dieser Lesart zerlegt die PCA die Merkmalsmatrix in einen niedrig-rank-Anteil $\mathbf{L}$ und einen kleinen Rest $\mathbf{E}$.

Die Zerlegung (1) legt nahe, die Komponente $\mathbf{P}\mathbf{x}^{(r)}$ als neue Merkmale zu verwenden. Eine solche Darstellung ist der Vektor $\mathbf{z}^{(r)} = \big(\langle \mathbf{u}^{(1)}, \mathbf{x}^{(r)}\rangle, \ldots, \langle \mathbf{u}^{(d')}, \mathbf{x}^{(r)}\rangle\big)^\top$, dessen Einträge die Skalarprodukte zwischen $\mathbf{x}^{(r)}$ und den Hauptkomponentenrichtungen der Merkmalsmatrix $\mathbf{X}$ sind. Für $d' = 2$ können die neuen Merkmale z_1, z_2 als Koordinaten in einem scatterplot verwendet werden, wie Panel (a) von Abb. 2 zeigt.

Manchmal ist ein Merkmal-Wert fehlerhaft, etwa weil ein Temperatursensor ausfällt. Das rote Quadrat in Abb. 1 markiert einen solchen Ausfall: Der Mittagswert des fünften Tages ist um 22 Grad erhöht, was zur gestörten Merkmalsvektor $\widetilde{\mathbf{x}}^{(5)}$ und damit zur gestörten Merkmalsmatrix $\widetilde{\mathbf{X}}$ führt. Panel (b) von Abb. 2 zeigt das scatterplot, das die PCA aus $\widetilde{\mathbf{X}}$ liefert. Die neuen Merkmale $z_1^{(5)}, z_2^{(5)}$, die für $\widetilde{\mathbf{x}}^{(5)}$ erhalten werden, liegen weit entfernt von denen der anderen Tage, und die zweite gelernte Hauptkomponentenrichtung dreht sich in Richtung der Achse des gestörten Merkmal. Ein isolierter, aber großer Fehler kann also die Koordinaten jedes Datenpunkt erheblich verzerren.

Die Zerlegung (1) erklärt, warum das so ist. Angewandt auf $\widetilde{\mathbf{X}}$ ergibt sich der niedrig-rank-Anteil $\mathbf{L} = \widetilde{\mathbf{X}}\mathbf{P}$, der aus den aufgezeichneten Werten berechnet wird, und ein Rest $\mathbf{E}$, den die PCA in der Frobenius-Norm klein hält. Die 22 Grad können daher nicht in $\mathbf{E}$ verbleiben. Der Großteil davon muss in $\mathbf{L}$ erscheinen, und die einzige Möglichkeit, wie eine Matrix mit rank höchstens d' einen großen Wert an einem einzelnen Eintrag tragen kann, ist, dass sich der Unterraum $\mathcal{U}$ in Richtung des fehlerhaften Merkmal dreht.

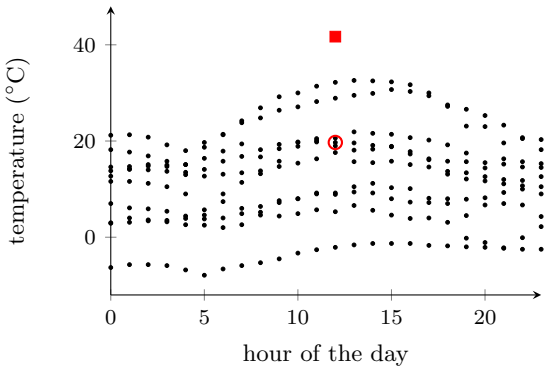

Abbildung 1: Stündliche Temperaturaufzeichnungen für zehn Tage des Jahres 2024 in Krems an der Donau, dargestellt als kleine schwarze Punkte. Jeder Tag ist ein Datenpunkt mit einem 24-dimensionalen Merkmalsvektor $\mathbf{x}$. Der rote Kreis markiert den korrekten Mittagswert von Tag fünf, und das große rote Quadrat die Störung, die ihn ersetzt, 22 Grad höher

Die robuste PCA lässt die Nebenbedingung $\mathbf{L} = \mathbf{XP}$ weg und fügt einen dritten Summanden für die fehlerhaften Einträge hinzu,

$$\mathbf{X} = \mathbf{L} + \mathbf{S} + \mathbf{E}. \tag{2}$$

Die niedrig-rank Matrix $\mathbf{L}$ ist nun eine eigene Unbekannte, nur durch $\operatorname{rank}(\mathbf{L}) \leq d'$ beschränkt. Die dünnbesetzte Matrix $\mathbf{S}$ enthält die fehlerhaften Merkmal-Werte und ist in allen anderen Einträgen null. Der Rest $\mathbf{E}$ bleibt wie in der PCA-Zerlegung (1) klein. Die Zeile r der Zerlegung (2) teilt die Merkmalsvektor $\mathbf{x}^{(r)}$ in diese drei Anteile auf.

Der Unterraum und die Hauptkomponentenrichtungen werden anschließend aus $\mathbf{L}$ berechnet. Die Zeilen von $\mathbf{L}$ spannen einen Unterraum der Dimension höchstens d' auf, und die Hauptkomponentenrichtungen $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(d')}$ sind die führenden rechten Singulär-Vektoren von $\mathbf{L}$, geliefert durch deren Singulärwertzerlegung (SVD). Die Projektion-Matrix $\mathbf{P}$ ergibt sich daraus, und $\mathbf{L} = \mathbf{LP}$ gilt, da jede Zeile von $\mathbf{L}$ bereits im Unterraum liegt, auf den $\mathbf{P}$ projiziert. Die robuste PCA liefert also die gleichen Objekte wie die PCA, jedoch aus einer Matrix $\mathbf{L}$, die frei von den fehlerhaften Merkmal-Werten ist, da $\mathbf{S}$ diese trägt.

Das Ziel, $\mathbf{E}$ klein zu halten, während $\mathbf{S}$ nur wenige von Null verschiedene Einträge enthält, lässt sich als Optimierungsproblem

$$\min_{\mathbf{L},\mathbf{S}\in\mathbb{R}^{m\times d}} \|\mathbf{X} - \mathbf{L} - \mathbf{S}\|_F \quad \text{unter der Nebenbedingung } \operatorname{rank}(\mathbf{L}) \leq d',\ \|\mathbf{S}\|_0 \leq s. \tag{3}$$

Die optimization-Variablen sind die beiden Matrizen $\mathbf{L}$ und $\mathbf{S}$, und die Zielfunktion ist die Frobenius-Norm des Rests $\mathbf{E} = \mathbf{X} - \mathbf{L} - \mathbf{S}$. Die Pseudonorm $\|\mathbf{S}\|_0$ bezeichnet die Anzahl der von Null verschiedenen Einträge von $\mathbf{S}$, und s ist eine vorgegebene Schranke für die Anzahl fehlerhafter Merkmal- Werte.

Das Optimierungsproblem (3) ist nicht-konvex, da weder der rank noch die Anzahl der von Null verschiedenen Einträge eine konvex Funktion ist. Principal component pursuit ersetzt es durch das konvex Optimierungsproblem

$$\min_{\mathbf{L},\mathbf{S}\in\mathbb{R}^{m\times d}} \|\mathbf{L}\|_* + \alpha\|\mathbf{S}\|_1 \quad \text{unter der Nebenbedingung } \mathbf{X} = \mathbf{L} + \mathbf{S}, \tag{4}$$

mit einem Regularisierung-Parameter $\alpha > 0$ (Candès et al., 2011). Hierbei ist $\|\mathbf{L}\|_*$ die Summe der Singulärwerte von $\mathbf{L}$ und $\|\mathbf{S}\|_1$ die Summe der Beträge der Einträge von $\mathbf{S}$. Diese beiden Normen sind konvex, wie jede Norm (Boyd and Vandenberghe, 2004, Sektionen 3.1.5 und A.1.6), und ersetzen den rank von $\mathbf{L}$ sowie die Anzahl der von Null verschiedenen Einträge von $\mathbf{S}$. Ein großer Regularisierung- Parameter α macht jeden von Null verschiedenen Eintrag von $\mathbf{S}$ teuer, sodass die Lösung $\mathbf{S}$ dünnbesetzter hält und mehr von $\mathbf{X}$ auf $\mathbf{L}$ überträgt. Der rank von $\mathbf{L}$ und damit die Dimension d' des Unterraum fällt dann größer aus, während ein kleiner α den gegenteiligen Effekt hat.
PCA ist ein Spezialfall von empirische Risikominimierung (ERM) mit der squared error loss. Principal Component Pursuit (4) ändert ihren Hypothesenraum, ihr Kriterium und ihre Optimierungsverfahren. Der Hypothesenraum ist größer: Ein Kandidat der PCA ist eine Projektion-Matrix $\mathbf{P}$, die bereits $\mathbf{L} = \mathbf{XP}$ festlegt, während ein Kandidat der robusten PCA ein Paar $(\mathbf{L}, \mathbf{S})$ ist, sodass die dünnbesetzte Matrix eine zweite Unbekannte ist. Die squared error loss entfällt, da die Nebenbedingung $\mathbf{X} = \mathbf{L} + \mathbf{S}$ keinen Rest zum Messen übriglässt. Was Principal Component Pursuit minimiert, ist daher keine Verlustfunktion. Es handelt sich um eine konvexe Relaxierung der rank- und Sparsamkeitsnebenbedingungen des nicht-konvexen Optimierungsproblems (3), und diese Relaxierung macht das Optimierungsproblem lösbar (Candès et al., 2011, Sect. 1.2). Die Optimierungsverfahren ändert sich mit dem Kriterium: Eine SVD der Merkmalsmatrix genügt für die PCA, da deren führende rechte singuläre Vektoren die Hauptkomponentenrichtungen sind (Hastie et al., 2009, Sect. 14.5.1), während Algorithmus 1 pro Iteration eine SVD berechnet (Candès et al., 2011, Sect. 5).

Methoden der robusten PCA unterscheiden sich in der verwendeten Optimierungsverfahren zur Lösung der Principal Component Pursuit (4). Algorithmus 1 fasst eine solche Methode zusammen, die durch Anwendung des alternating direction method of multipliers (ADMM) gewonnen wird. Panel (c) von Abb. 2 zeigt das scatterplot, das die PCA aus der niedrig-rank-Matrix $\mathbf{L}$ liefert, die Algorithmus 1 aus der gestörten Merkmalsmatrix $\widetilde{\mathbf{X}}$ rekonstruiert. Die zehn Tage liegen wieder an den Koordinaten von Panel (a). Die von Null verschiedenen Einträge von $\mathbf{S}$ zeigen auf die fehlerhaften Messwerte selbst, weshalb die robuste PCA auch zur anomaly detection eingesetzt wird.

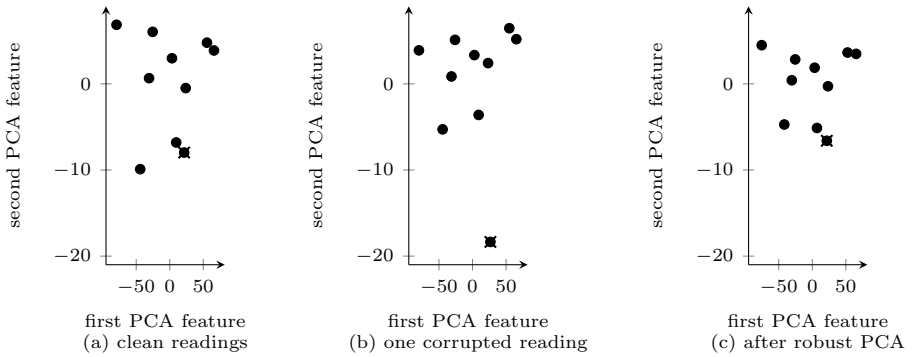

Abbildung 2: Koordinaten derselben zehn Tage, gegeben durch die zwei Merkmale, die (a) die PCA auf den unverfälschten Messwerten, (b) die PCA nach Erhöhung eines Mittagswerts von Tag fünf um 22 Grad und (c) die PCA auf der niedrig-rank-Matrix $\mathbf{L}$, die die robuste PCA aus den verfälschten Messwerten rekonstruiert, liefern. Ein Kreuz markiert Tag fünf in jedem Panel, ein ausgefüllter Kreis jeweils einen der anderen neun Tage. Alle drei Panels teilen sich einen Achsenbereich, sodass die Punktwolken direkt verglichen werden können

Algorithmus 1 ist eine fixed-point iteration

$$\big(\mathbf{L}^{(t+1)}, \mathbf{S}^{(t+1)}, \mathbf{W}^{(t+1)}\big) = T\big(\mathbf{L}^{(t)}, \mathbf{S}^{(t)}, \mathbf{W}^{(t)}\big),$$

deren Operator T zwei proximale Operatoren mit dem Update der Matrix $\mathbf{W}$ verknüpft. Das Schrumpfen der Singulärwerte von $\mathbf{B}$ um $1/\rho$ ist die proximaler Operator der nuklearen Norm (Candès et al., 2011, Eq. (5.3)), weshalb dieser Schritt die SVD benötigt. Das Schrumpfen jedes Eintrags von $\mathbf{C}$ um α/ρ ist die proximaler Operator von $\|\cdot\|_1$ (Candès et al., 2011, Eq. (5.2)), also diejenige, die auch Löser für das least absolute shrinkage and selection operator (Lasso) verwenden und der einzige Schritt, in dem α in die Iteration eingeht. An einem Fixpunkt von T verändern die drei Updates das Iterat nicht mehr. Ein Fixpunkt ist genau dann erreicht, wenn $\mathbf{X} = \mathbf{L} + \mathbf{S}$ gilt, $(\mathbf{L}, \mathbf{S})$ Principal Component Pursuit (4) löst und $\rho\mathbf{W}$ ein Lagrange-Multiplikator der Nebenbedingung ist. Das Optimierungsproblem (4) ist konvex, daher konvergiert die Iteration für jedes $\rho > 0$: Das Residuum $\mathbf{X} - \mathbf{L} - \mathbf{S}$ strebt gegen Null und die Zielfunktion gegen ihr Minimum (Boyd et al., 2011, Sect. 3.2). Für die Mittelwerte der Iterierten über die ersten T Iterationen gilt diese Konvergenz mit einer expliziten Schranke (Beck, 2017, Thm. 15.4).

Algorithm 1 Principal Component Pursuit mittels ADMM (Candès et al., 2011, Sect. 5, Algorithm 1)

Input: Merkmalsmatrix $\mathbf{X} \in \mathbb{R}^{m \times d}$, Regularisierung Parameter $\alpha > 0$, penalty Parameter $\rho > 0$, tolerance $\varepsilon^{(\text{tol})} > 0$

$\mathbf{L}^{(0)} := \mathbf{0}$, $\mathbf{S}^{(0)} := \mathbf{0}$, $\mathbf{W}^{(0)} := \mathbf{0}$, $t := 0$

repeat

  $\mathbf{B} := \mathbf{X} - \mathbf{S}^{(t)} + \mathbf{W}^{(t)}$

  compute the SVD $\mathbf{B} = \mathbf{V}\mathbf{\Lambda}\mathbf{U}^\top$, whose Singulärwerte are the diagonal entries $\Lambda_{j,j} = \lambda_j \geq 0$, for $j = 1, \ldots, \min\{m, d\}$

  $\mathbf{L}^{(t+1)} := \mathbf{V}\widetilde{\mathbf{\Lambda}}\mathbf{U}^\top$, where $\widetilde{\mathbf{\Lambda}} \in \mathbb{R}^{m \times d}$ has the diagonal entries $\max\left\{\lambda_j - 1/\rho,\, 0\right\}$ and is zero off the diagonal

  $\mathbf{C} := \mathbf{X} - \mathbf{L}^{(t+1)} + \mathbf{W}^{(t)}$

  $S_{r,j}^{(t+1)} := \operatorname{sign}\left(C_{r,j}\right) \max\left\{\left|C_{r,j}\right| - \alpha/\rho,\, 0\right\}$, for $r = 1, \ldots, m$ and $j = 1, \ldots, d$

  $\mathbf{W}^{(t+1)} := \mathbf{W}^{(t)} + \mathbf{X} - \mathbf{L}^{(t+1)} - \mathbf{S}^{(t+1)}$

  $t := t + 1$

until $\left\|\mathbf{X} - \mathbf{L}^{(t)} - \mathbf{S}^{(t)}\right\|_F \leq \varepsilon^{(\text{tol})} \|\mathbf{X}\|_F$

Output: low-rank Matrix $\mathbf{L}^{(t)}$ and sparse Matrix $\mathbf{S}^{(t)}$

Siehe auch: PCA, sparse principal component analysis (sparse PCA), probabilistic principal component analysis (probabilistic PCA), SVD, rank, ADMM, proximaler Operator, fixed-point iteration, Lasso, Regularisierung, Ausreißer, anomaly detection, robustness, ERM, Hypothesenraum, Verlustfunktion, squared error loss.

Querverweise

Literatur

1. Beck (2017). First-Order Methods in Optimization. SIAM-Society for Industrial and Applied Mathematics, Philadelphia, PA, USA.

2. Boyd and Vandenberghe (2004). Convex Optimization. Cambridge Univ. Press, Cambridge, U.K.

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

4. Boyd et al. (2011). Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. Found. Trends Mach. Learn., 3(1), pp. 1–122.

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