アルゴリズム 7: 文書ベクトル (DocVec) の計算アルゴリズム.

1: 式(4.94)に従って文書–単語行列 $\mathbf{X}$ を作成する. (疎行列フォーマットで)
2: $\mathbf{U}, \mathbf{S}, \mathbf{V} = \mathrm{svds}(\mathbf{X}, K)$ と K 次元に特異値分解する.
3: 文書ベクトル行列 $\mathbf{D} = \mathbf{U}\mathbf{S}^{1/2}$, 単語ベクトル行列 $\mathbf{W} = \mathbf{V}^T\mathbf{S}^{1/2}$ を計算する.

図 4.31: 特異値分解による文書ベクトル (DocVec) の計算アルゴリズム.

```
   # -R をつけて実行し，回帰行列を事前に計算しておく
   % docvec-search.py model.docvec-R livedoor.txt 映画 東京
⇒ loading model from model.docvec-R.. done.
   keyword: 映画 東京
    0.0158 movie-enter    この夏、東京スカイツリーが映画に 2008 年 7 月の
    0.0152 movie-enter    小栗旬は“使えない若者”、映画『キツツキと雨』の
    0.0152 movie-enter    スパイダーマンが地上 75 メートルの通天閣を『アメ
    0.0149 peachy         【スナップレポート】東京ガールズコレクション 2011
    0.0149 movie-enter    食べて、で、映画を観れる『東京ごはん映画祭』が今
    0.0149 peachy         【スナップレポート】東京ガールズコレクション 2011
    0.0148 movie-enter    基礎から勉強しよう!初心者でもわかる「東京国際映
    0.0144 movie-enter    三池監督が映画『一命』について「満島ひかりがいろ
    0.0141 movie-enter    東京国際映画祭、『最強のふたり』がグランプリを受
```

4.5.2　単語ベクトル/文書ベクトルの解釈

こうして得られた文書ベクトルは高速に計算でき, 数学的に word2vec(Doc2Vec) と同じニューラル文書ベクトルとなっているため, 高い性能を持っています. 唯一の欠点は, K 個の次元が LDA のようにトピックとして解釈ができないということでしょう. たとえば, 上の実験で得られた単語ベクトルを並べた行列 $\mathbf{W}$ について, その 1 次元目, 2 次元目, …の値が大きい単語を求めると表 4.10 のようになり, ここには強い規則性は見出せそうにありません.

考えてみるとこれは当然で, 式(4.95)による行列分解は式(4.97)のように内積だけを問題にしているため, 空間全体を任意に回転しても, 図 4.32 のように 2 つのベクトルの間の内積は同じになるからです. 数学的には, ある直交行列 $\mathbf{R}$ をとって $\widetilde{\mathbf{D}} = \mathbf{D}\mathbf{R}$, $\widetilde{\mathbf{W}} = \mathbf{W}\mathbf{R}$ とベクトル全体を回転したとき, 式(4.95)の右辺は

$$\widetilde{\mathbf{D}}\widetilde{\mathbf{W}}^T = \mathbf{DR}(\mathbf{WR})^T = \mathbf{D}\underbrace{\mathbf{RR}^T}_{=\mathbf{I}}\mathbf{W}^T = \mathbf{DW}^T \tag{4.102}$$

となり, もとと等しくなります.

すなわち, 解釈のためには図 4.32 に示したように, 単語ベクトルや文書ベクトルがちょうど軸の近くに配置されるような回転 $\mathbf{R}$ を見つける必要があります. こうした方法として, 心理学ではバリマックス回転のような方法が知られています[166]. しかし, これは言語のように数百次元以上にもなる高次元の場合にはあまりうまくいかず[167], 最近, 京都大学の下平らにより, **ICA** (独立成分分析) を適用することで解釈が容易な軸が, しかも言語横断的に見つかることが示されました[168]. ICA はデータを線形変換して, 各次元が可能な限り統計的に独立となるような座標軸を発見する方法です [*42]. たとえば図 4.33 では, PCA(主成分分析) ではデータの分散を最大にするように真ん中のような座標軸がとられてしまいますが, ICA ではデータの分布を独立な軸の積で説明する右のような座標軸を計算することができます.

「統計的に独立」とは, 1 章の式(1.21) でみたように, 確率変数 x と y の同時確率が $p(x,y) = p(x)p(y)$ のように周辺確率の積に分解できることでした. われわれの場合, たとえば式(4.95)で得られた, 単語ベクトルを並べた行列 $\mathbf{W}$ を行列

次元 1		次元 2		次元 3		次元 4	
級	0.6090	自身	0.6138	Watch	0.7628	再生	0.7540
節電	0.5832	イベント	0.5650	Sports	0.6780	ホラー	0.5644
全	0.5279	クリスマス	0.5647	婚	0.5346	恐怖	0.5515
電力	0.5185	http	0.5261	活	0.5325	音楽	0.5425
シリーズ	0.4905	作っ	0.5027	答え	0.5069	デビュー	0.4848
AKB	0.4835	城	0.4955	音楽	0.5028	PC	0.4742
母親	0.4638	シーズン	0.4949	佐	0.4505	現象	0.4396
通話	0.4564	テレビ	0.4719	曲	0.4487	娘	0.4229
スマ	0.4509	婦	0.4476	学校	0.4450	ロンドン	0.4222
出展	0.4390	超	0.4430	枝	0.4409	YouTube	0.4201
家族	0.4374	家政	0.4391	調査	0.4366	必ず	0.4186
額	0.4268	www	0.4380	選手	0.4173	韓国	0.4184

表 4.10: Livedoor コーパスから DocVec で計算した単語ベクトルの各次元の値が大きい単語 (一部). もとのままでは, 各次元に明確な意味は見出せそうにありません.

*42 158 ページで説明した白色化はデータを無相関にする方法ですが, 独立とは無相関を含んでおり, それより強い条件になります.

$\mathbf{A}$ を使って $\mathbf{S}=\mathbf{WA}$ と線形変換したとき, $\mathbf{S}$ の各行ベクトル $\mathbf{s}=(s_1, s_2, \cdots, s_K)$ について, 分解

$$p(s_1, s_2, \cdots, s_K) = p(s_1)p(s_2)\cdots p(s_K) \tag{4.103}$$

が可能な限り成り立つような行列 $\mathbf{A}$ を求めることになります [*43].

こうした $\mathbf{A}$ は, 機械学習の分野で ICA を定式化したフィンランドの Hyvärinen によるパッケージ `FastICA`[*44] で計算することができます. Python では,

```
from sklearn.decomposition import FastICA # ICA の計算
from scipy.stats import skew              # 歪度の計算
import numpy as np
def ica (X):
    X = X - np.mean (X, axis=0)
    analyzer = FastICA (whiten="arbitrary-variance")
    S = analyzer.fit_transform (X)
    A = analyzer.components_
    # sort by skewness
    N,D = S.shape
    skews = np.abs (skew (S, axis=0)) # 0=Gaussian
    index = map (lambda x: x[1], sorted (zip(skews, np.arange(D)),
                 key=lambda x: x[0], reverse=True)))
    return S[:,list(index)], A
```

で $\mathbf{S}$ および $\mathbf{A}$ を求めることができます. 多くの信号が混ざると, 中心極限定

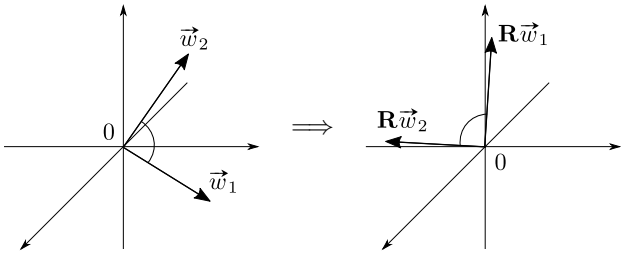

図 4.32: 単語ベクトルの回転. 単語ベクトルを直交行列 $\mathbf{R}$ で回転しても, 二つの単語ベクトルのなす角度は不変です. 適切な回転 $\mathbf{R}$ を求めることで, 単語ベクトルを軸に沿った形で, よりわかりやすく解釈することができます.

---

*43　ICA は実際には, 158 ページで説明したデータの白色化を行ってから $\mathbf{R}$ による回転を行うことと等価で, 白色化行列を $\mathbf{B}$ とおくと $\mathbf{A}=\mathbf{BR}$ になります.

*44　`http://research.ics.aalto.fi/ica/fastica/`

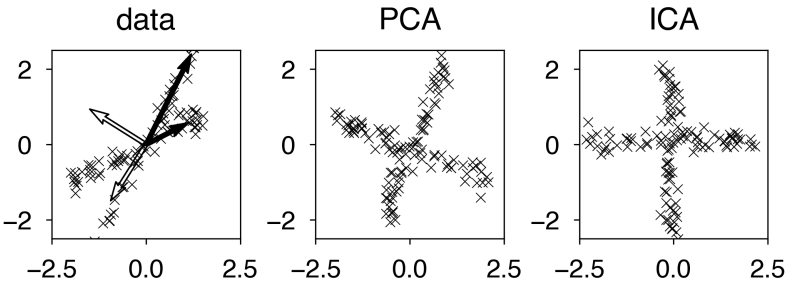

図 4.33: ICA の概要図. PCA(主成分分析) では白矢印のように, データ (×印) の分散を全体的に説明する軸がとられてしまいますが, ICA(独立成分分析) では黒矢印のように, データの分布を独立な軸の積として説明する軸が求められているのがわかります.

理によって分布はガウス分布に近づくことから, 逆に ICA で分解した軸では, 各次元の周辺分布は非ガウス的になります. その度合いは, 期待値 μ, 標準偏差 σ をもつ確率変数 X の**歪度**(わいど)

$$\delta = \mathbb{E}[(X-\mu)^3]/\sigma^3 \tag{4.104}$$

で表すことができます[169]. ガウス分布では δ は 0 で, $\pm$ になるほど左右の裾が広くなることが知られています. ICA は PCA と異なり, 独立性の高い軸から求まるとは限りませんので, 上のコードでは δ の絶対値を用いて, 非ガウス性の高い順に次元を並び換えています. Livedoor コーパスから計算した単語ベクトルについて, ICA で変換した各次元の値の大きい単語を表 4.11 に示しました. ほとんどトピックモデルのような, 解釈性の高い意味的な軸が求められていることがわかります.

ただしトピックモデルと異なり, これはベクトル空間の「軸」ですので, 負の方向も存在します [*45]. 表 4.12 に, 表 4.11 のいくつかの軸について値が負の単語を示しました. 単なる文書–単語の共起行列から得られたにもかかわらず, iPhone ⟷ Android, ビジネス ⟷ 悩み相談のような興味深い対立軸が教師なしで得られていることがわかります.

∗45 空間全体をある軸に関して折り返してもベクトル間の関係は変わりませんので, 符号が正負どちらになるかに大きな意味はありません.

∗45 `https://news.livedoor.com/article/detail/6029862/`

次元 3		次元 5		次元 6		次元 10	
iPhone	0.0224	apps	0.0853	ビジネス	0.0180	等	0.0272
クリック	0.0185	google	0.0851	経営	0.0174	由里子	0.0269
ブック	0.0163	要件	0.0848	成功	0.0172	吉	0.0268
電子	0.0160	store	0.0847	キャリア	0.0164	愛	0.0265
術	0.0160	play	0.0842	オフィスエム	0.0155	サイト	0.0260
アップ	0.0152	ANDROID	0.0821	笑	0.0147	幸せ	0.0246
サイト	0.0150	details	0.0820	話し	0.0144	果たし	0.0243
携帯	0.0142	Play	0.0779	管理	0.0139	公式	0.0242
既報	0.0135	Google	0.0600	スキル	0.0132	務め	0.0234
背面	0.0134	Hisumi	0.0565	戦略	0.0129	篇	0.0233
ライフ	0.0123	以上	0.0421	力	0.0129	リアル	0.0224
iPad	0.0123	Android	0.0420	重要	0.0124	女優	0.0223

次元 15		次元 19		次元 28		次元 39	
高画質	0.0122	サッカー	0.2559	ロードショー	0.0428	スタイル	0.0218
デジタル	0.0105	代表	0.2401	全国	0.0425	オシャレ	0.0197
実現	0.0095	戦	0.2193	女優	0.0237	着	0.0171
操作	0.0093	試合	0.2175	決意	0.0206	ファッション	0.0170
ソニー	0.0084	杯	0.1897	土	0.0200	楽しむ	0.0166
ズーム	0.0082	W	0.1591	実力	0.0199	シンプル	0.0164
進化	0.0081	チーム	0.1532	最強	0.0198	ダイエット	0.0153
保存	0.0080	ゴール	0.1410	黄金	0.0194	味わい	0.0153
シーン	0.0080	日本	0.1280	絆	0.0193	体重	0.0151
新	0.0079	予選	0.1277	祭	0.0187	食事	0.0151
軽量	0.0079	選手	0.1261	ベルセルク	0.0186	誰	0.0147
レス	0.0079	リーグ	0.1239	TOHO	0.0184	運動	0.0145

表 4.11: ICA で変換した単語ベクトルの各軸が大きい単語 (抜粋). 次元は歪度の絶対値の大きい順に並んでいます. 表 4.10 と比べて, ほとんどトピックモデルのような, 意味的な軸が学習されていることがわかります. 次元 10 の意味については, 表 4.12 を参照してください.

次元 3		次元 6		次元 10		次元 19	
apps	−0.2364	お答え	−0.2046	妖	−0.2595	フィギュア	−0.0405
store	−0.2363	辛口	−0.2041	ヶ	−0.2562	スケート	−0.0403
details	−0.2362	悩め	−0.2026	巻	−0.2316	選手権	−0.0364
play	−0.2300	説教	−0.2018	劇場	−0.1994	演技	−0.0312
google	−0.2299	尽き	−0.1938	勅使河原	−0.1992	克也	−0.0268
要件	−0.2225	早	−0.1856	妖怪	−0.1967	浅田	−0.0265
ANDROID	−0.2187	面白く	−0.1820	栄華	−0.1961	真央	−0.0262
id	−0.2119	姉妹	−0.1689	仙人	−0.1900	メダリスト	−0.0262
Play	−0.2006	悩み	−0.1680	お嬢様	−0.1695	野村	−0.0255
com	−0.1913	type	−0.1663	ネコ	−0.1589	ノム	−0.0246
Store	−0.1798	若手	−0.1650	話	−0.1497	バンクーバー	−0.0231
カテゴリ	−0.1682	瞬時	−0.1596	次	−0.1430	野球	−0.0225

表 4.12: ICA で変換した単語ベクトルの各軸が小さい単語 (抜粋). 表 4.11 と見比べると, 次元 3 では iPhone の「反対の概念」として Android が, 次元 6 ではビジネスについてお悩み相談が, 次元 19 ではサッカーについてフィギュアスケートが得られていることがわかります. 次元 10 は元データの Livedoor ブログでの “いちおう妖ヶ劇場” という連載記事 [*46] に関連している軸で, 負の方にだけ意味を持っています.