# Различные подходы к решению задачи Sparse Principal Components Analisys

*Выступающий:* Кузнецов Кирилл Николаевич
*Руководитель:* к.ф.-м.н. Панов Максим Евгеньевич

Высшая Школа Экономики
Математические Методы Оптимизации и Стохастики

Апрель, 2016

# Цели и задачи

- **Предмет исследования:** Применение методов кластеризации при решении задач Sparse Principal Component Analysis
- **Цель исследования:** Рассмотрение возможности достижения хороших результатов в задачах SPCA
- **Проблемы:**
  - Низкая скорость работы исходного алгоритма
  - Неточность алгоритмов кластеризации

# План работ

1. **Введение**
   - Необходимые определения
   - Метод Главных Компонент
2. **Sparse PCA**
   - Рассмотрение метода
   - Известные решения задачи SPCA
   - Рассмотрение возможности применения кластеризации в решении задачи SPCA
3. **Численные результаты**

- $trA$ - след матрицы
- Норма матрицы

$$\|A\|^2 = \sum_{i=1}^{n}\sum_{j=1}^{m} a_{ij} = trAA^T = trA^TA$$

- **Сингулярное разложение:**
  Произвольная $l \times n$ матрица представима в виде сингулярного разложения:

$$F = VDU^T$$

Свойства сингулярного разложения:

- $n \times n$ матрица $D$ - диагональна.

$$D = diag(\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_n})$$

$\lambda_i$ - общие ненулевые собственные значения матриц $F^T F$ и $FF^T$

- $l \times n$ матрица $V = (v_1, \ldots, v_n)$ ортогональна, $V^T V = I_n$, где столбцы $v_j$ являются собственными векторами матрицы $FF^T$, соответствующими $\lambda_1, \ldots, \lambda_n$;
- $l \times n$ матрица $U = (u_1, \ldots, u_n)$ ортогональна, $U^T U = I_n$, где столбцы $u_j$ являются собственными векторами матрицы $F^T F$, соответствующими $\lambda_1, \ldots, \lambda_n$;

# Метод Главных Компонент

- **Необходимость метода:**
  - возникновение мультиколлинеарности
  - уменьшение количества признаков
- **Идея метода:** в PCA строится минимальное число новых признаков, по которым исходные признаки могут быть восстановлены линейным преобразованием с минимальными погрешностями.

- **Постановка задачи:** Пусть имеется $n$ числовых признаков $f_j(x), j = 1, \ldots, n$. Тогда признаковое описание объекта обучающей выборки представим в виде: $x_i = (f_1(x_i), \ldots, f_n(x_i)), i = 1, \ldots, l$.
  Исходная матрица:

$$F_{l \times n} = \begin{pmatrix} f_1(x_1) & \ldots & f_n(x_1) \\ \ldots & \ldots & \ldots \\ f_1(x_l) & \ldots & f_n(x_l) \end{pmatrix} = \begin{pmatrix} x_1 \\ \ldots \\ x_m \end{pmatrix}$$

  Обозначим через $z_i = (g_1(x_i), \ldots, g_m(x_i))$ признаковые описания тех же объектов в новом просранстве $Z = R^m, m < n$

$$G_{l \times m} = \begin{pmatrix} g_1(x_1) & \ldots & g_m(x_1) \\ \ldots & \ldots & \ldots \\ g_1(x_l) & \ldots & g_m(x_l) \end{pmatrix} = \begin{pmatrix} z_1 \\ \ldots \\ z_m \end{pmatrix}$$

Потребуем, чтобы исходные признаковые описания можно было восстановить по новым описаниям с помощью некоторого линейного преобразования, определяющемого матрицей $U = (u_{js})_{n \times m}$:

$$\hat{f}_j(x) = \sum_{s=1}^{m} g_s(x) u_{js}, \qquad j = 1, \ldots, n$$

В векторной записи: $\hat{x} = zU^T$. При этом, отличие $x$ от $\hat{x}$ должно быть как можно меньше при выбранной размерности $m$.

$$\Delta^2(G, U) = \sum_{i=1}^{l} \|\hat{x}_i - x_i\|^2 = \sum_{i=1}^{l} \|z_i U^T - x_i\|^2 = \|GU^T - F\|^2 \to \min_{G,U}$$

**Теорема**
Если $m \leq rkF$, то минимум $\Delta^2(G, U)$ достигается, когда столбцы матрицы $U$ есть собственные векторы $F^T F$, соответствующие $m$ максимальным собственным значениям. При этом $G = FU$, матрицы $U$ и $G$ ортогональны

$$\Delta^2(G, U) = \sum_{j=m+1}^{n} \lambda_j,$$

где $\lambda_1, \ldots, \lambda_n$ - все собственные значения матрицы $F^T F$. Собственные векторы $u_1, \ldots, u_m$, отвечающие максимальным собственным значениям называют **главными компонентами**

**Связь с сингулярным разложением**

Если $m = n$, то $\Delta^2(G, U) = 0$. Тогда представление $F = GU^T$ является точным и совпадает с сингулярным разложением: $F = GU^T = VDU^T$, если положить, что $G = VD$ и $\Lambda = D^2$. При этом $V^T V = I_M$

Если $m < n$, то представление $F \approx GU^T$ является приближенным. Сингулярное разложение матрицы $GU^T$ получается из сингулярного разложения матрицы $F$ путем обнуления $n - m$ минимальных собственных значений.

**Эффективная размерность:**
Главные компоненты содержат основную информацию о матрице $F$. Число главных компонент $m$ называют эффективной размерностью задачи. Все собственные значения матрицы $F^T F$ упорядочивают по убыванию: $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_n \geq 0$.

$$E(m) = \frac{\|GU^T - F\|^2}{\|F\|^2} = \frac{\lambda_{m+1} + \cdots + \lambda_n}{\lambda_1 + \cdots + \lambda_n} \leq \epsilon$$

## Разреженный Метод Главных Компонент

Разреженный метод главных компонент добавляет к обычному методу главных компонент разреженность на входные переменные.

Pitprops Data: Loadings of the First Six Principal Components

| *Variable* | *PC1* | *PC2* | *PC3* | *PC4* | *PC5* | *PC6* |
|---|---|---|---|---|---|---|
| topdiam | −0.404 | 0.218 | −0.207 | 0.091 | −0.083 | 0.120 |
| length | −0.406 | 0.186 | −0.235 | 0.103 | −0.113 | 0.163 |
| moist | −0.124 | 0.541 | 0.141 | −0.078 | 0.350 | −0.276 |
| testsg | −0.173 | 0.456 | 0.352 | −0.055 | 0.356 | −0.054 |
| ovensg | −0.057 | −0.170 | 0.481 | −0.049 | 0.176 | 0.626 |
| ringtop | −0.284 | −0.014 | 0.475 | 0.063 | −0.316 | 0.052 |
| ringbut | −0.400 | −0.190 | 0.253 | 0.065 | −0.215 | 0.003 |
| bowmax | −0.294 | −0.189 | −0.243 | −0.286 | 0.185 | −0.055 |
| bowdist | −0.357 | 0.017 | −0.208 | −0.097 | −0.106 | 0.034 |
| whorls | −0.379 | −0.248 | −0.119 | 0.205 | 0.156 | −0.173 |
| clear | 0.011 | 0.205 | −0.070 | −0.804 | −0.343 | 0.175 |
| knots | 0.115 | 0.343 | 0.092 | 0.301 | −0.600 | −0.170 |
| diaknot | 0.113 | 0.309 | −0.326 | 0.303 | 0.080 | 0.626 |
| Variance (%) | 32.4 | 18.3 | 14.4 | 8.5 | 7.0 | 6.3 |
| Cumulative variance (%) | 32.4 | 50.7 | 65.1 | 73.6 | 80.6 | 86.9 |

Pitprops Data: Loadings of the First Six Sparse PCs by SPCA. Empty cells have zero loadings.

| *Variable* | *PC1* | *PC2* | *PC3* | *PC4* | *PC5* | *PC6* |
|---|---|---|---|---|---|---|
| topdiam | −0.477 | | | | | |
| length | −0.476 | | | | | |
| moist | | 0.785 | | | | |
| testsg | | 0.620 | | | | |
| ovensg | 0.177 | | 0.640 | | | |
| ringtop | | | 0.589 | | | |
| ringbut | −0.250 | | 0.492 | | | |
| bowmax | −0.344 | −0.021 | | | | |
| bowdist | −0.416 | | | | | |
| whorls | −0.400 | | | | | |
| clear | | | | −1 | | |
| knots | | 0.013 | | | −1 | |
| diaknot | | | −0.015 | | | 1 |
| Number of nonzero loadings | 7 | 4 | 4 | 1 | 1 | 1 |
| Variance (%) | 28.0 | 14.4 | 15.0 | 7.7 | 7.7 | 7.7 |
| Adjusted variance (%) | 28.0 | 14.0 | 13.3 | 7.4 | 6.8 | 6.2 |
| Cumulative adjusted variance (%) | 28.0 | 42.0 | 55.3 | 62.7 | 69.5 | 75.8 |

# Разреженный Метод Главных Компонент

PCA:

$$\max v^T \Sigma v$$

$$s.t. \|v\|_2 = 1$$

SPCA:

$$\max v^T \Sigma v$$

$$s.t. \|v\|_2 = 1$$

$$\|v\|_0 \leq k$$

# Разреженный Метод Главных Компонент

После нахождения оптимального вектора $v$, вычисляем новую матрицу $\Sigma_1$

$$\Sigma_1 = \Sigma - (v^T \Sigma v) v v^T$$

Повторяя данную процедуру, мы найдем последующие главные компоненты.

- **Но данная задача является NP-сложной!**

# Методы решения задачи SparsePCA

- a regression framework[1]
- a convex relaxation/semidefinite programming framework[2]
- a generalized power method framework[3]
- an alternating maximization framework[4]
- forward/backward greedy search and exact methods using branch-and-bound techniques[5]
- Bayesian formulation framework[6]
- ...

[1] - H. Zou; T. Hastie; R. Tibshirani (2006). "Sparse principal component analysis"
[2] -Alexandre d'Aspremont; Laurent El Ghaoui; Michael I. Jordan; Gert R. G. Lanckriet (2007). "A Direct Formulation for Sparse PCA Using Semidefinite Programming"
[3] - Michel Journee; Yurii Nesterov; Peter Richtarik; Rodolphe Sepulchre (2010). "Generalized Power Method for Sparse Principal Component Analysis"
[4] - Peter Richtarik; Martin Takac; S. Damla Ahipasaoglu (2012). "Alternating Maximization: Unifying Framework for 8 Sparse PCA Formulations and Efficient Parallel Codes"
[5] - Baback Moghaddam; Yair Weiss; Shai Avidan (2005). "Spectral Bounds for Sparse PCA: Exact and Greedy Algorithms"
[6] - Yue Guan; Jennifer Dy (2009). "Sparse Probabilistic Principal Component Analysis

**Алгоритм:**

1. Let A start at $U[, 1 : k]$, the loadings of the first k ordinary principal components.
2. Given a fixed $A = [\alpha_1, ..., \alpha_k]$, solve the following elastic net problem for $j = 1, 2, \ldots, k$

$$\beta_j = arg \min_{\beta} (\alpha_j - \beta)^T X^T X (\alpha_j - \beta) + \lambda \|\beta\|^2 + \lambda_{1,j} \|\beta\|_1$$

3. For a fixed $B = [\beta_1, \ldots, \beta_k]$, compute the SVD of $X^T XB = VDU^T$ , then update $A = VU^T$
4. Repeat Steps 2–3, until convergence
5. Normalization $\hat{U}_j = \frac{\beta_j}{\|\beta_j\|}, \quad j = 1, \ldots, k$

**Алгоритм:**

**input**: Data matrix $X \in R^{p \times n}$, Sparsity controlling parameter $\gamma \geq 0$, initial iterate $\beta \in S^p$

**output**: A locally optimal pattern P

**begin**

**repeat**

$$\beta = \sum_{i=1}^{n} [|x_i^T \beta - \gamma]_+ sign(x_i^T \beta) x_i$$

$$\beta = \frac{\beta}{\|\beta\|}$$

**until** stopping criterion is satisfied

Construct vector $P \in \{0, 1\}^n$ such that

$$\begin{cases} p_i = 1 & \text{if } |x_i^T \beta| \geq \gamma \\ p_i = 0 & \text{otherwise} \end{cases}$$

**end**

**Алгоритм:**

**input**: Data matrix $X \in R^{p \times n}$, Sparsity controlling parameter $\gamma \geq 0$, initial iterate $\beta \in S^p$

**output**: A locally optimal pattern P

**begin**

**repeat**

$$\beta = \sum_{i=1}^{n} [sign((x_i^T \beta)^2 - \gamma)]_+ x_i^T \beta x_i$$

$$\beta = \frac{\beta}{\|\beta\|}$$

**until** stopping criterion is satisfied

Construct vector $P \in \{0, 1\}^n$ such that

$$\begin{cases} p_i = 1 & \text{if } (x_i^T \beta)^2 \geq \gamma \\ p_i = 0 & \text{otherwise} \end{cases}$$

**end**

**Алгоритм:**

**input**: Data matrix $X \in R^{p \times n}$, Sparsity controlling vector $[\gamma_1, \ldots, \gamma_m]^T \geq 0$, Parameters $\mu_1, \ldots, \mu_m > 0$, initial iterate $B \in S_m^p$

**output**: A locally optimal pattern P

**begin**

**repeat**

**for** $j = 1, \ldots, m$ **do**

$\beta_j = \sum_{i=1}^{n} \mu_j [\mu_j |x_i^T \beta| - \gamma_j]_+ sign(x_i^T \beta) x_i$

$B = Polar(B)$

**until** stopping criterion is satisfied

Construct vector $P \in \{0,1\}^{n \times m}$ such that

$$\begin{cases} p_{ij} = 1 & \text{if } \mu_j |x_i^T \beta| \geq \gamma_j \\ p_{ij} = 0 & \text{otherwise} \end{cases}$$

**end**

**Алгоритм:**

**input**: Data matrix $X \in R^{p\times n}$, Sparsity controlling vector $[\gamma_1, \ldots, \gamma_m]^T \geq 0$, Parameters $\mu_1, \ldots, \mu_m > 0$, initial iterate $B \in S^p_m$

**output**: A locally optimal pattern P

**begin**

**repeat**

**for** $j = 1, \ldots, m$ **do**

$\beta_j = \sum_{i=1}^{n} \mu_j^2 [sign((\mu_j x_i^T \beta_j)^2 - \gamma_j)]_+ x_i^T \beta x_i$

$B = Polar(B)$

**until** stopping criterion is satisfied

Construct vector $P \in \{0,1\}^{n\times m}$ such that

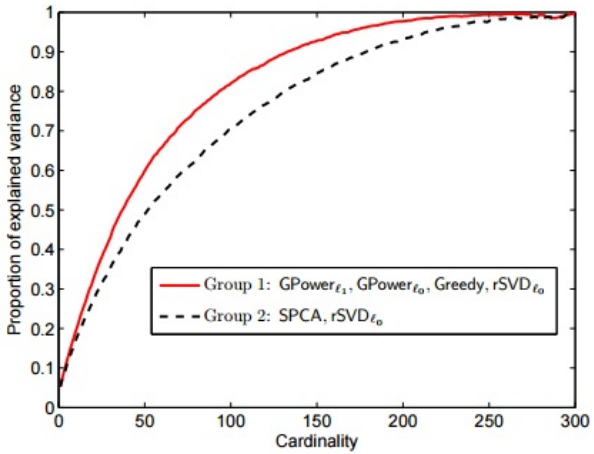
$$\begin{cases} p_{ij} = 1 & \text{if } (\mu_j x_i^T \beta_j)^2 \geq \gamma_j \\ p_{ij} = 0 & \text{otherwise} \end{cases}$$

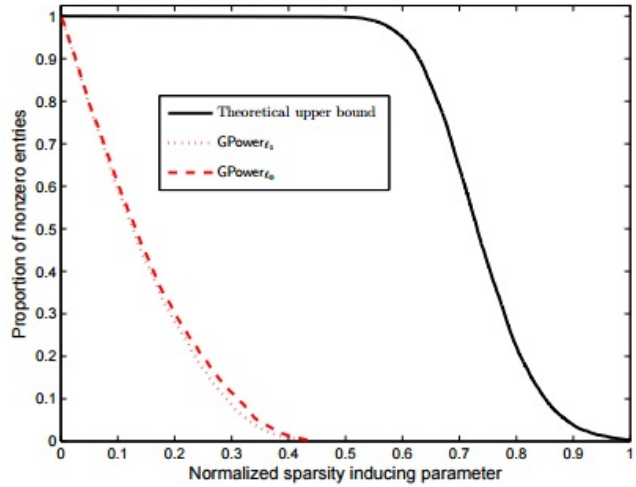
**end**

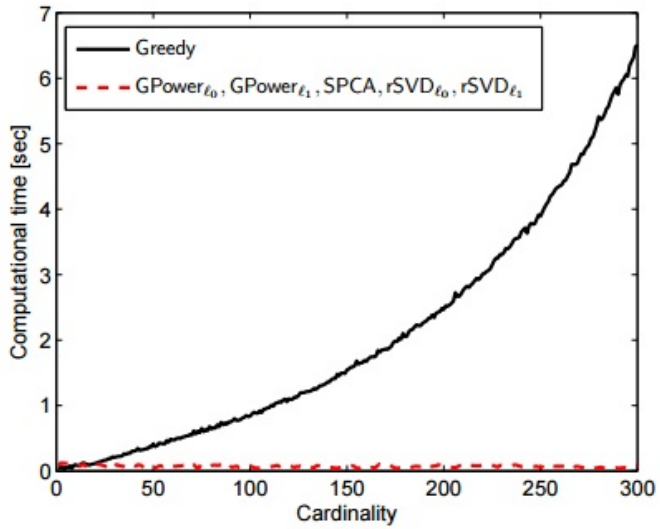
# Сравнение различных методов

- $GPower_{l_1}$ Single-unit sparse PCA via $l_1$-penalty
- $GPower_{l_0}$ Single-unit sparse PCA via $l_0$-penalty
- $GPower_{l_1,m}$ Block sparse PCA via $l_1$-penalty
- $GPower_{l_0,m}$ Block sparse PCA via $l_0$-penalty
- Greedy Greedy method
- SPCA SPCA algorithm
- $rSVD_{l_1}$ sPCA-rSVD algorithm with an $l_1$-penalty ("soft thresholding")
- $rSVD_{l_0}$ sPCA-rSVD algorithm with an $l_0$-penalty ("hard thresholding")

# Сравнение различных методов



# Сравнение различных методов

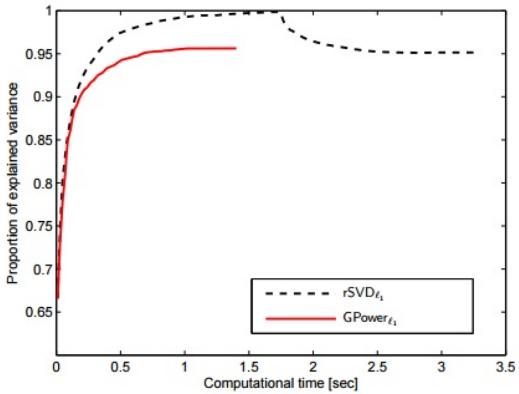
| $p \times n$ | $100 \times 1000$ | $250 \times 2500$ | $500 \times 5000$ | $750 \times 7500$ | $1000 \times 10000$ |
|---|---|---|---|---|---|
| GPower$_{\ell_1}$ | 0.10 | 0.86 | 2.45 | 4.28 | 5.86 |
| GPower$_{\ell_0}$ | 0.03 | 0.42 | 1.21 | 2.07 | 2.85 |
| SPCA | 0.24 | 2.92 | 14.5 | 40.7 | 82.2 |
| rSVD$_{\ell_1}$ | 0.19 | 2.42 | 3.97 | 7.51 | 9.59 |
| rSVD$_{\ell_0}$ | 0.18 | 2.14 | 3.85 | 6.94 | 8.34 |

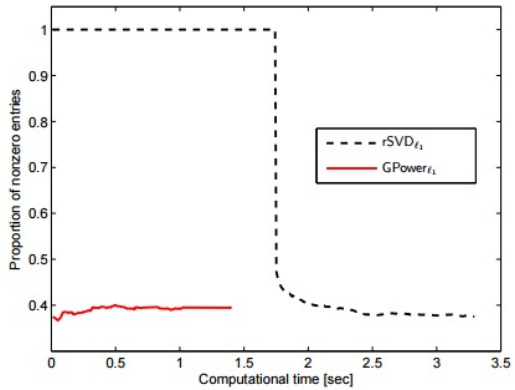
Average computational time for the extraction of one component (in seconds).

| $p \times n$ | $500 \times 1000$ | $500 \times 2000$ | $500 \times 4000$ | $500 \times 8000$ | $500 \times 16000$ |
|---|---|---|---|---|---|
| GPower$_{\ell_1}$ | 0.42 | 0.92 | 2.00 | 4.00 | 8.54 |
| GPower$_{\ell_0}$ | 0.18 | 0.42 | 0.96 | 2.14 | 4.55 |
| SPCA | 5.20 | 7.20 | 12.0 | 22.6 | 44.7 |
| rSVD$_{\ell_1}$ | 1.05 | 2.12 | 3.63 | 7.43 | 14.4 |
| rSVD$_{\ell_0}$ | 1.02 | 1.97 | 3.45 | 6.58 | 13.2 |

Average computational time for the extraction of one component (in seconds).

# Сравнение различных методов




| $p \times n$ | $50 \times 500$ | $100 \times 1000$ | $250 \times 2500$ | $500 \times 5000$ | $750 \times 7500$ |
|---|---|---|---|---|---|
| GPower$_{\ell_1}$ | 0.22 | 0.56 | 4.62 | 12.6 | 20.4 |
| GPower$_{\ell_0}$ | 0.06 | 0.17 | 2.15 | 6.16 | 10.3 |
| GPower$_{\ell_1,m}$ | 0.09 | 0.28 | 3.50 | 12.4 | 23.0 |
| GPower$_{\ell_0,m}$ | 0.05 | 0.14 | 2.39 | 7.7 | 12.4 |
| SPCA | 0.61 | 1.47 | 13.4 | 48.3 | 113.3 |
| rSVD$_{\ell_1}$ | 0.29 | 1.12 | 7.72 | 22.6 | 46.1 |
| rSVD$_{\ell_0}$ | 0.28 | 1.03 | 7.21 | 20.7 | 41.2 |

Average computational time for the extraction of $m = 5$ components (in seconds).

# Сравнение различных методов

| Method | Parameters | Total cardinality | Prop. of explained variance |
|---|---|---|---|
| $\text{rSVD}_{\ell_1}$ | see Shen and Huang (2008) | 25 | 0.7924 |
| SPCA | see Zou et al. (2006) | 18 | 0.7680 |
| Greedy | cardinalities: 6-2-3-1-1-1 | 14 | 0.7150 |
| | cardinalities: 5-2-2-1-1-1 | 12 | 0.5406 |
| $\text{GPower}_{\ell_1}$ | $\gamma_j/\bar{\gamma}_j = 0.22$, for $j = 1,\ldots,6$ | 25 | 0.8083 |
| | $\gamma_j/\bar{\gamma}_j = 0.28$ | 18 | 0.7674 |
| | $\gamma_j/\bar{\gamma}_j = 0.30$ | 15 | 0.7542 |
| | $\gamma_j/\bar{\gamma}_j = 0.40$ | 13 | 0.7172 |
| | $\gamma_j/\bar{\gamma}_j = 0.50$ | 11 | 0.6042 |
| $\text{GPower}_{\ell_1,m}$ | $\gamma_j/\bar{\gamma}_j = 0.17$, for $j = 1,\ldots,6$ | 25 | 0.7733 |
| with $\mu_j = 1$ | $\gamma_j/\bar{\gamma}_j = 0.25$ | 17 | 0.7708 |
| | $\gamma_j/\bar{\gamma}_j = 0.3$ | 14 | 0.7508 |
| | $\gamma_j/\bar{\gamma}_j = 0.4$ | 13 | 0.7076 |
| | $\gamma_j/\bar{\gamma}_j = 0.45$ | 11 | 0.6603 |
| $\text{GPower}_{\ell_1,m}$ | $\gamma_j/\bar{\gamma}_j = 0.18$, for $j = 1,\ldots,6$ | 25 | 0.8111 |
| | $\gamma_j/\bar{\gamma}_j = 0.25$ | 18 | 0.7849 |
| with $\mu_j = \frac{1}{j}$ | $\gamma_j/\bar{\gamma}_j = 0.30$ | 15 | 0.7610 |
| | $\gamma_j/\bar{\gamma}_j = 0.35$ | 13 | 0.7323 |
| | $\gamma_j/\bar{\gamma}_j = 0.40$ | 12 | 0.6656 |

Extraction of 6 components from the pitprops data.

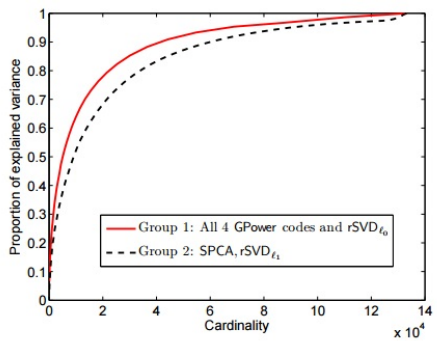
# Сравнение различных методов

| Study | Samples ($p$) | Genes ($n$) | Reference |
|---|---|---|---|
| Vijver | 295 | 13319 | van de Vijver et al. (2002) |
| Wang | 285 | 14913 | Wang et al. (2005) |
| Naderi | 135 | 8278 | Naderi et al. (2007) |
| JRH-2 | 101 | 14223 | Sotiriou et al. (2006) |

Breast cancer cohorts.

| | Vijver | Wang | Naderi | JRH-2 |
|---|---|---|---|---|
| GPower$_{\ell_1}$ | 5.92 | 5.33 | 2.15 | 2.69 |
| GPower$_{\ell_0}$ | 4.86 | 4.93 | 1.33 | 1.73 |
| GPower$_{\ell_1,m}$ | 5.40 | 4.37 | 1.77 | 1.14 |
| GPower$_{\ell_0,m}$ | 5.61 | 7.21 | 2.25 | 1.47 |
| SPCA | 77.7 | 82.1 | 26.7 | 11.2 |
| rSVD$_{\ell_1}$ | 10.19 | 9.97 | 3.96 | 4.43 |
| rSVD$_{\ell_0}$ | 9.51 | 9.23 | 3.46 | 3.61 |

Average computational times (in seconds) for the extraction of $m = 10$ components.



# Сравнение различных методов

| | Vijver | Wang | Naderi | JRH-2 |
|---|---|---|---|---|
| PCA | 0.0728 | 0.0466 | 0.0149 | 0.0690 |
| GPower$_{\ell_1}$ | **0.1493** | 0.1026 | 0.0728 | 0.1250 |
| GPower$_{\ell_1}$ | 0.1250 | 0.1250 | 0.0672 | 0.1026 |
| GPower$_{\ell_1,m}$ | 0.1418 | 0.1250 | **0.1026** | **0.1381** |
| GPower$_{\ell_0,m}$ | 0.1362 | **0.1287** | 0.1007 | 0.1250 |
| SPCA | 0.1362 | 0.1007 | 0.0840 | 0.1007 |
| rSVD$_{\ell_1}$ | 0.1213 | 0.1175 | 0.0914 | 0.0914 |
| rSVD$_{\ell_0}$ | 0.1175 | 0.0970 | 0.0634 | 0.1063 |

PEI-values based on a set of 536 cancer-related pathways.

| | Vijver | Wang | Naderi | JRH-2 |
|---|---|---|---|---|
| PCA | 0.0347 | 0 | 0.0289 | 0.0405 |
| GPower$_{\ell_1}$ | 0.1850 | 0.0867 | 0.0983 | 0.1792 |
| GPower$_{\ell_0}$ | 0.1676 | 0.0809 | 0.0925 | **0.1908** |
| GPower$_{\ell_1,m}$ | **0.1908** | **0.1156** | **0.1329** | 0.1850 |
| GPower$_{\ell_0,m}$ | 0.1850 | 0.1098 | **0.1329** | 0.1734 |
| SPCA | 0.1734 | 0.0925 | 0.0809 | 0.1214 |
| rSVD$_{\ell_1}$ | 0.1387 | 0.0809 | 0.1214 | 0.1503 |
| rSVD$_{\ell_0}$ | 0.1445 | 0.0867 | 0.0867 | 0.1850 |

PEI-values based on a set of 173 motif-regulatory gene sets.

# Другой подход

Будем использовать методы кластерного анализа для решения задачи SparsePCA.

**Алгоритм:**

Строим ковариационную матрицу

Применяем к ковариационной матрице методы кластерного анализа

Для каждого отдельного кластера находим **Главные компоненты** обычным методом главных компонент

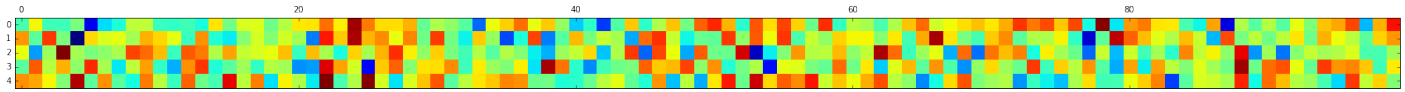



Рис.: Первые 5 компонент с помощью PCA

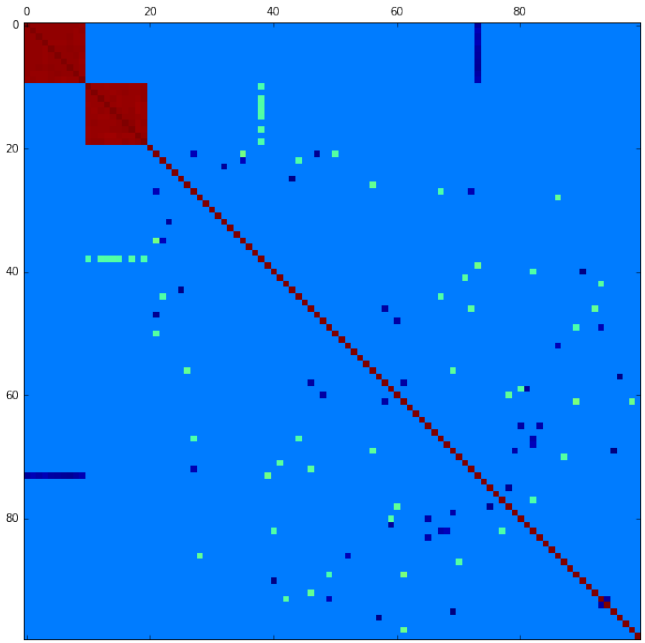



Рис.: Ковариационная матрица

# Другой подход

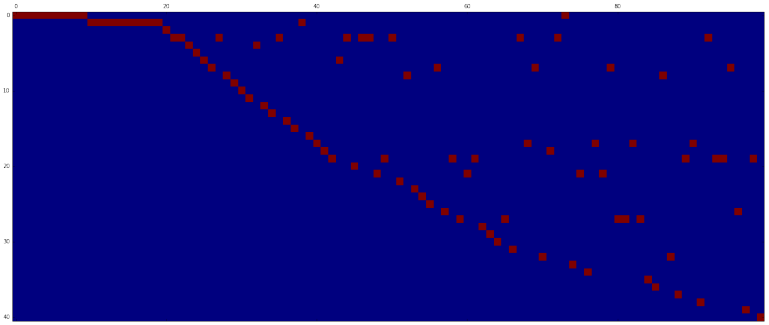


Рис.: Признаки объединенные в кластеры


Рис.: Результат применения PCA к первым кластерам

Спасибо за внимание!