Метод главных компонент

Дисперсия (variance)

•

Ковариация (covariance)

•

Ковариационная матрица

-

Корреляция (correlation)

•

Метод главных компонент

- Principal Components Analysis (PCA)
- Один из основных практических способов уменьшить размерность данных
- Дана матрица $X_{m \times n}$
  - матрица «объекты –признаки»
- Реализация метода:
  - вычисление собственных векторов и собственных значений ковариационной матрицы исходных данных
  - сингулярное разложение центрированной матрицы исходных данных
  - алгоритм NIPALS (для первых k компонент)

Формализация

- $$X = TP^T$$
- T – матрица счетов (*score matrix*)
  - ортогональная матрица
  - столбцы t_i – главные компоненты
- P – матрица нагрузок (*loadings matrix*)
  - ортогональная матрица
- Сокращение размерности
  - Возьмем первые k столбцов T и P:

$$X = T_k P_k^T + E$$

Классическая реализация

- Строим матрицу ковариаций столбцов матрицы X
  - $cov(X) = C = [\mathrm{c}_{ij}]$
  - $\mathrm{c}_{ij} = \frac{1}{m-1}\sum_{k=1}^{m}(x_{ki} - \overline{X}_i)(x_{kj} - \overline{X}_j)$
  - variance-covariance matrix
- Находим собственные векторы (t_i) и собственные числа (λ_i) матрицы C
- Матрица T формируется из столбцов t_i, отсортированных по убыванию значений соответствующих λ_i
  - $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0$

Реализация на основе SVD

- Предварительно необходимо центрирование исходной матрицы
  - $x_{ij} = x_{ij} - \overline{x_j}, j = 1, \dots n$
- $X = U\Sigma V^T$

$$\mathrm{T} = U\Sigma$$
$$\mathrm{P} = V$$

- Матрица $X^T X$ пропорциональна матрице ковариаций
  - сингулярное разложение X равнозначно нахождению собственных векторов $X^T X$

PCA – NIPALS
(Nonlinear Iterative Partial Least Squares)

- Начало
  - $i = 1,\ X_1 = X$
- Итерация i
  1. *Вектор* t_i - *произвольный столбец* X_i
  2. $p_i = \frac{X_i^T t_i}{\|X_i^T t_i\|}$ (веса, ищем направление в пространстве X, дающее максимальную ковариацию)
  3. $\hat{t_i} = X_i p_i$ (score vector, линейная комбинация X_i с весами p_i)
  4. *if* $(t_i \approx \hat{t_i})$ *goto step 5* (проверка сходимости)
     else $\{t_i = \hat{t_i};$ *goto step 2;* $\}$
  5. $X_{i+1} = X_i - t_i p_i^T$ (вычисление остатков)
- Stop $if\ (i = k)$

PCA – NIPALS (пояснение)

- Покажем, что алгоритм находит собственные числа и векторы матрицы $X^T X$
  - Пусть $\|X^T t_i\| = \lambda_i$
  - Шаг 2: $X^T t_i = \lambda_i p_i$
  - Подставим $t_i = X_i p_i$ (шаг 3):

$$X^T X p_i = \lambda_i p_i$$

  - Следовательно:
    - λ_i - собственное число $X^T X$,
    - p_i - собственный вектор $X^T X$

PCA – NIPALS (пояснение)

- Покажем, что t_1 и $X_2 = X - t_1 p_1^T$ ортогональны:

$$t_i^T t_i = p_i^T X^T X p_i = \lambda_i p_i^T p_i = \lambda_i$$

  - последний шаг: т.к. p_i - единичный вектор

- После шагов 1-5, $i = 1$:

$$X = t_1 p_1^T + X_2$$

- Тогда

$$(X - t_1 p_1^T)^T t_1 = X^T t_1 - p_1 t_1^T t_1 = X^T X p_1 - p_1 \lambda_1 = 0$$

- После шагов 1-5, $i = 2$:

$$X = t_1 p_1^T + t_2 p_2^T + X_3$$

- После k итераций ($i = \mathrm{k}$):

$$X = t_1 p_1^T + t_2 p_2^T + \cdots + t_k\, p_k^T + X_{k+1}$$

  - В случае $k = r$:

$$X_{k+1} = 0$$