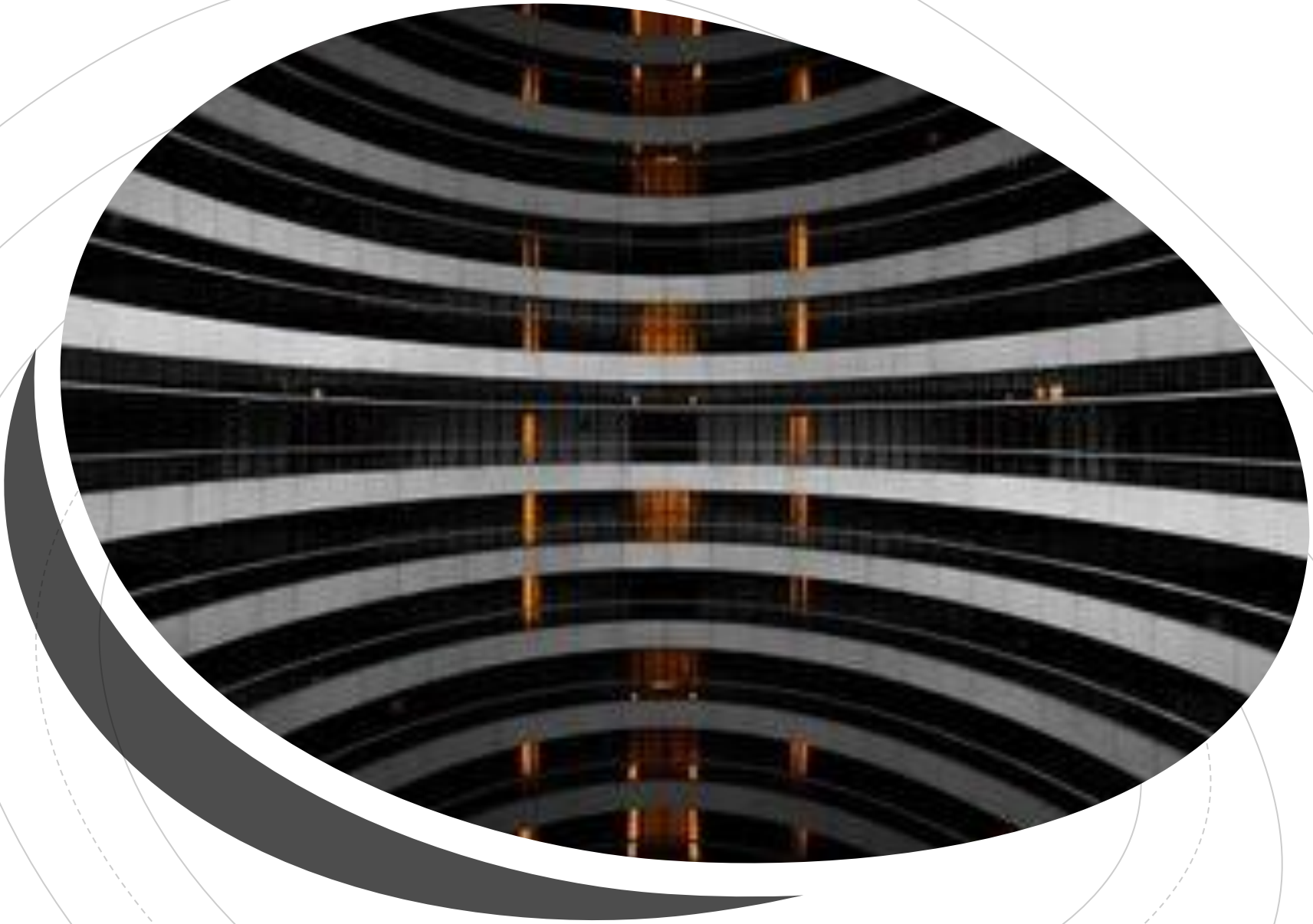
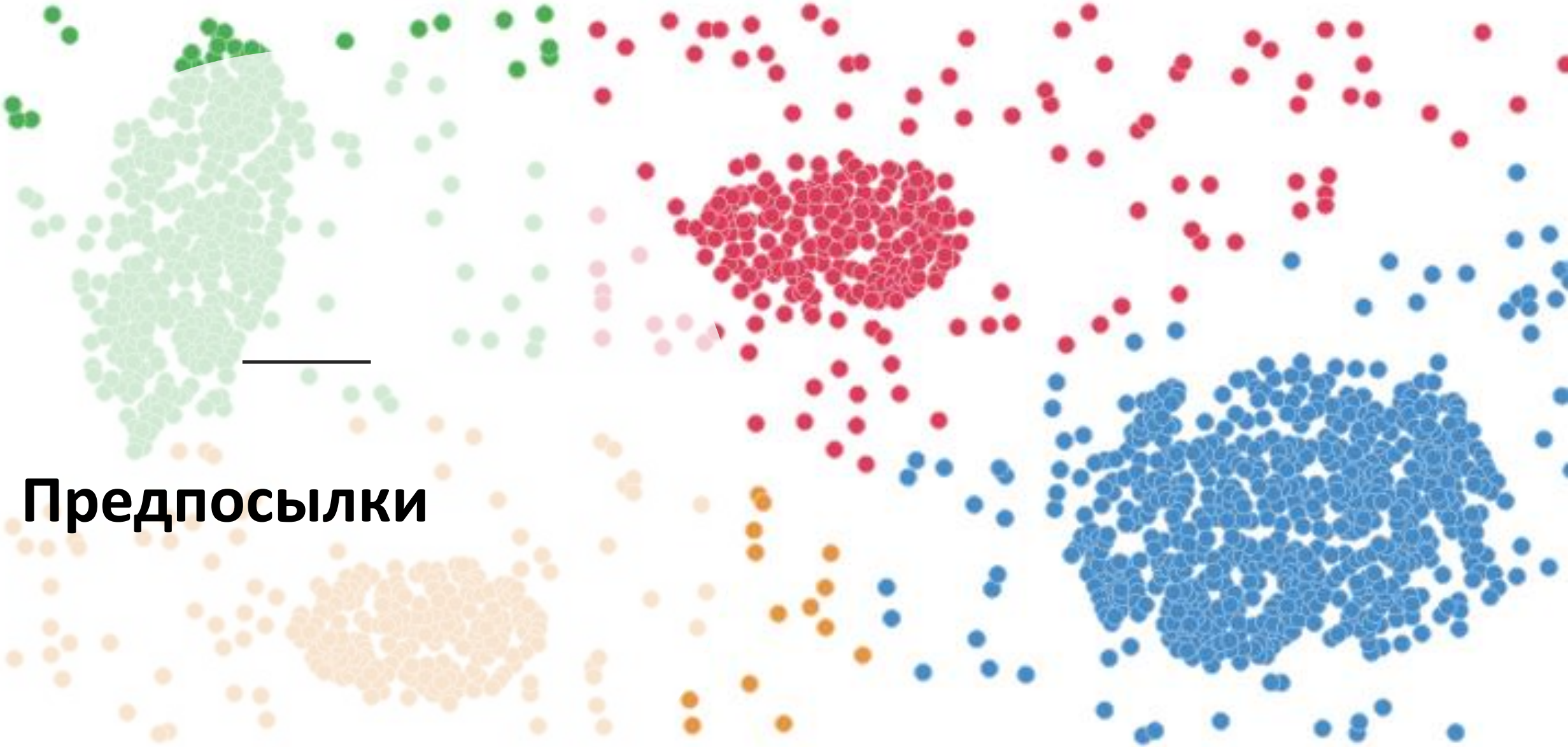


Machine Learning

Лекция #3

Снижение размерности



Предпосылки

Пусть данные – это матрица $\mathbf{D}$ размера $n \times d$ (n точек с d признаками):

$$\mathbf{D} = \left(\begin{array}{c|cccc} X & X_1 & X_2 & \cdots & X_d \\ \mathbf{x}_1 & x_{11} & x_{12} & \cdots & x_{1d} \\ \mathbf{x}_2 & x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_n & x_{n1} & x_{n2} & \cdots & x_{nd} \end{array}\right)$$

Каждая точка $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{id})^T$ —вектор в d-мерном пространстве с базисом $\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_d$, в котором компонент $\mathbf{e}_i$ соотносится с i-тым признаком X_i

Таким образом, для любого набора d ортонормированных векторов $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_d$ такого, что $\mathbf{u}_i^T \mathbf{u}_j = 0$ и $\|\mathbf{u}_i\| = 1$ (или $\mathbf{u}_i^T \mathbf{u}_i = 1$), возможно представление каждой точки $\mathbf{x}$ как линейной комбинации:

$$\mathbf{x} = a_1 \mathbf{u}_1 + a_2 \mathbf{u}_2 + \cdots + a_d \mathbf{u}_d, (7.1)$$

где вектор $\mathbf{a} = (a_1, a_2, \ldots, a_d)^T$ — это координаты $\mathbf{x}$ в новом базисе. Или в матричной форме: $\mathbf{x} = \mathbf{U}\mathbf{a}$, (7.2)

где $\mathbf{U}$ — матрица размера $d \times d$, i-ый столбец которой содержит i-тый базисный вектор:

$$\mathbf{U} = \begin{pmatrix} | & | & & | \\ \mathbf{u}_1 & \mathbf{u}_2 & \cdots & \mathbf{u}_d \\ | & | & & | \end{pmatrix}$$

$\mathbf{U}$ =Матрица $\mathbf{U}$ является *ортогональной*, все её столбцы, будучи базисными векторами, *ортогональны*, то есть, они попарно ортогональны и имеют единичную длину:

$$\mathbf{u}_i^T \mathbf{u}_j = \begin{cases} 1, \text{если } i = j \\ 0, \text{если } i \neq j \end{cases}.$$

Поскольку $\mathbf{U}$ ортогональна, её обратная матрица равна её транспонированной:

$$\mathbf{U}^{-1} = \mathbf{U}^T$$

следовательно: $\mathbf{U}^T\mathbf{U} = \mathbf{I}$, где $\mathbf{I}$ — единичная матрица размера $d \times d$.

Умножив ур. (7.2) слева на $\mathbf{U}^T$, получим выражение для вычисления координат $\mathbf{x}$ в новом базисе:

$$\mathbf{U}^T\mathbf{x} = \mathbf{U}^T\mathbf{U}\mathbf{a}$$
$$\mathbf{a} = \mathbf{U}^T\mathbf{x}. \quad (7.3)$$

Пусть $\mathbf{x}$ – произвольная точка и пусть базисные векторы отсортированы в порядке уменьшения значимости – тогда выражение 7.1 может быть усечено до r членов:

$$\mathbf{x}' = a_1\mathbf{u}_1 + a_2\mathbf{u}_2 + \cdots + a_r\mathbf{u}_r = \sum_{i=1}^{r} a_i\mathbf{u}_i \,. (7.4)$$

Здесь $\mathbf{x}'$ — это проекция $\mathbf{x}$ на первые r базисных векторов, или в терминах матричных операций:

$$\mathbf{x}' = \begin{pmatrix} | & | & | & | \\ \mathbf{u}_1 & \mathbf{u}_2 & \cdots & \mathbf{u}_r \\ | & | & | & | \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_r \end{pmatrix} = \mathbf{U}_r\mathbf{a}_r, (7.5)$$

где $\mathbf{U}_r$ — матрица, состоящая из r первых базисных векторов, а $\mathbf{a}_r$ — вектор, содержащий первые r координат. Далее, ограничив ур-е 7.3 первыми r членами, получим:

$$\mathbf{a}_\mathrm{r} = \mathbf{U}_r^T\mathbf{x}. (7.6)$$

Подставив 7.6 в 7.5, можно получить окончательное выражение для проекции $\mathbf{x}$ на первые r базисных векторов:

$$\mathbf{x}' = \mathbf{U}_r\mathbf{U}_r^T\mathbf{x} = \mathbf{P}_r\mathbf{x}, (7.7)$$

где $\mathbf{P}_r = \mathbf{U}_r\mathbf{U}_r^T$ — *ортогональная проекционная матрица* для подпространства над первыми r базисными векторами.

То есть, $\mathbf{P}_r$ симметрична, и $\mathbf{P}_r^2 = \mathbf{P}_r$. Поскольку: $\mathbf{P}_r^T = (\mathbf{U}_r\mathbf{U}_r^T)^T = \mathbf{U}_r\mathbf{U}_r^T = \mathbf{P}_r$ и $\mathbf{P}_r^2 = (\mathbf{U}_r\mathbf{U}_r^T)(\mathbf{U}_r\mathbf{U}_r^T) = \mathbf{U}_r\mathbf{U}_r^T = \mathbf{P}_r$. Используя факт, что $\mathbf{U}_r^T\mathbf{U}_r = \mathbf{I}_{r\times r}$, единичная матрица размера $r \times r$. Проекционную матрицу $\mathbf{P}_r$ можно записать как декомпозицию:

$$\mathbf{P}_r = \mathbf{U}_r\mathbf{U}_r^T = \sum_{i=1}^{r} \mathbf{u}_i\mathbf{u}_i^T \,. (7.8)$$

Из выражений 7.1 и 7.4 следует, что проекция $\mathbf{x}$ на оставшиеся размерности составляет *вектор ошибки*:

$$\epsilon = \sum_{i=r+1}^{d} a_i\mathbf{u}_i = \mathbf{x} - \mathbf{x}'$$

Следует заметить, что $\mathbf{x}'$ и ϵ — ортогональные векторы:

$$\mathbf{x}'^T \epsilon = \sum_{i=1}^{r} \sum_{j=r+1}^{d} a_i a_j \, \mathbf{u}_i^T \mathbf{u}_j = 0$$

что следует из ортонормированности базиса. Более того, возможно сделать утверждение: подпространство над первыми r базисными векторами:

$$S_r = span(\mathbf{u}_1, \dots, \mathbf{u}_r)$$

и подпространство над оставшимися базисными векторами:

$$S_{d-r} = span(\mathbf{u}_{r+1}, \dots, \mathbf{u}_d)$$

являются *ортогональными подпространствами*, то есть, любая пара векторов $\mathbf{x} \in S_r$ и $\mathbf{y} \in S_{d-r}$ должна быть ортогональной. Подпространство S_{d-r} называют также *ортогональным дополнением* S_r.

Цель понижения размерности *— найти такой* r*-мерный базис, который даст наилучшее приближение* $\boldsymbol{x}_i'$ *для всех точек* $\boldsymbol{x}_i \in \boldsymbol{D}$*, либо уменьшить ошибку* $\epsilon = \boldsymbol{x} - \boldsymbol{x}'$ *для всех точек.*

Метод главных компонент

Наилучшая линейная аппроксимация

Метод главных компонент (PCA) *– это метод, который ищет r-мерный базис, наилучшим образом описывающий дисперсию данных. Направление с наибольшей проецируемой дисперсией называется первым главным компонентом. Ортогональное направление, которое захватывает вторую по величине проецируемую дисперсию, называется вторым главным компонентом и т.д. Как мы увидим, направление, которое максимизирует дисперсию, также минимизирует среднеквадратичную ошибку.*

Наилучшая линейная аппроксимация

Начнем с $r = 1$, т.е. с одномерного подпространства или прямой $\boldsymbol{u}$, которая наилучшим образом аппроксимирует $\boldsymbol{D}$, с точки зрения дисперсии проецируемых точек. Это приведет к общей методике PCA для наилучшего $1 \leq r \leq d$ размерного базиса для $\boldsymbol{D}$.

Предполагаем, что $\boldsymbol{u}$ имеет величину $\|\boldsymbol{u}\|^2 = \boldsymbol{u}^T\boldsymbol{u} = 1$; в иначе можно продолжать увеличивать проецируемую дисперсию, просто увеличивая величину $\boldsymbol{u}$. Также предполагаем, что данные центрированы и среднее значение $\boldsymbol{\mu} = \mathbf{0}$.

Проекция $\boldsymbol{x}_i$ на вектор $\boldsymbol{u}$ берется как

$$\boldsymbol{x}_i' = \left(\frac{\boldsymbol{u}^T\boldsymbol{x}_i}{\boldsymbol{u}^T\boldsymbol{u}}\right)\boldsymbol{u} = (\boldsymbol{u}^T\boldsymbol{x}_i)\boldsymbol{u} = a_i\boldsymbol{u}$$

где скаляр

$$a_i = \boldsymbol{u}^T\boldsymbol{x}_i$$

дает координату $\boldsymbol{x_i'}$ вдоль прямой $\boldsymbol{u}$. Поскольку средняя точка $\boldsymbol{\mu = 0}$, то ее координата вдоль $\boldsymbol{u}$ равна $\mu_{\boldsymbol{u}} = 0$. Необходимо выбрать направление $\boldsymbol{u}$ так, чтобы дисперсия проецируемых точек была максимальной. Проецируемая дисперсия вдоль $\boldsymbol{u}$: где $\boldsymbol{\Sigma}$ – это ковариационная матрица для центрированных данных $\boldsymbol{D}$.

Задача ограниченной оптимизации (максимизировать $\sigma_{\boldsymbol{u}}^2$ с учетом ограничения $\boldsymbol{u^T u} = 1$). Для решения вводим множитель Лагранжа α для ограничения, чтобы получить задачу безусловной максимизации.

$$\max_{\boldsymbol{u}} J(\boldsymbol{u}) = \boldsymbol{u^T \Sigma u} - \alpha(\boldsymbol{u^T u} - 1) \quad (7.10)$$

Приравнивая производную $J(\boldsymbol{u})$ по $\boldsymbol{u}$ к нулевому вектору, получаем:

$$\sigma_{\boldsymbol{u}}^2 = \frac{1}{n}\sum_{i=1}^{n}(a_i - \mu_{\boldsymbol{u}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{u}^T \boldsymbol{x}_i)^2$$

$$= \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{u}^T(\boldsymbol{x}_i \boldsymbol{x}_i{}^T)\boldsymbol{u}$$

$$= \boldsymbol{u}^T\left(\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i{}^T\right)\boldsymbol{u} = \boldsymbol{u^T \Sigma u} \quad (7.9)$$

$$\frac{\partial}{\partial \boldsymbol{u}} J(\boldsymbol{u}) = \boldsymbol{0}$$

$$\frac{\partial}{\partial \boldsymbol{u}}\left(\boldsymbol{u^T \Sigma u} - \alpha(\boldsymbol{u^T u} - 1)\right) = \boldsymbol{0}$$

$$2\boldsymbol{\Sigma u} - \boldsymbol{2}\alpha\boldsymbol{u} = \boldsymbol{0}$$

$$\boldsymbol{\Sigma u} = \alpha \boldsymbol{u} \quad (7.11)$$

Следовательно α - собственное значение $\boldsymbol{\Sigma}$, с соответствующим собственным вектором $\boldsymbol{u}$. Далее, взяв скалярное произведение с $\boldsymbol{u}$ по обе стороны от уравнения (7.11) получаем:

$$\boldsymbol{u}^T\boldsymbol{\Sigma}\boldsymbol{u} = \boldsymbol{u}^T\alpha\boldsymbol{u}$$

Из уравнения (7.9) имеем:

$$\sigma_{\boldsymbol{u}}^2 = \alpha\boldsymbol{u}^T\boldsymbol{u}$$

$$\text{или } \sigma_{\boldsymbol{u}}^2 = \alpha \ (7.12)$$

Таким образом, чтобы максимизировать проецируемую дисперсию σ_u^2*, необходимо выбрать наибольшее собственное значение матрицы* Σ *. Другими словами, доминирующий собственный вектор* u_1 *задает направление наибольшей дисперсии, также называемое первым главным компонентом, т.е.* $u = u_1$ *. Кроме того, наибольшее собственное значение* λ_1 *определяет проецируемую дисперсию, т.е.* $\sigma_u^2 = \alpha = \lambda_1$.

Минимальная квадратичная ошибка

Теперь покажем, что направление, которое максимизирует проецируемую дисперсию, также минимизирует среднеквадратичную ошибку.

Предположим, что набор данных $\boldsymbol{D}$, центрирован путем вычитания среднего значения из каждой точки. Для точки $\boldsymbol{x}_i \in \boldsymbol{D}$ пусть $\boldsymbol{x}_i'$ обозначает ее проекцию вдоль направления $\boldsymbol{u}$, и пусть $\boldsymbol{\epsilon}_i = \boldsymbol{x}_i - \boldsymbol{x}_i'$ обозначает вектор ошибки. Среднеквадратичная ошибка (MSE) - условия оптимизации определяется как

$$MSE(\boldsymbol{u}) = \frac{1}{n}\sum_{i=1}^{n} \|\boldsymbol{\epsilon}_i\|^2 \quad (7.13)$$

$$= \frac{1}{n}\sum_{i=1}^{n} \|\boldsymbol{x}_i - \boldsymbol{x}_i'\|^2$$

$$= \frac{1}{n}\sum_{i=1}^{n} (\boldsymbol{x}_i - \boldsymbol{x}_i')^T(\boldsymbol{x}_i - \boldsymbol{x}_i') = \frac{1}{n}\sum_{i=1}^{n} \left(\|\boldsymbol{x}_i\|^2 - 2\boldsymbol{x}_i^T\boldsymbol{x}_i' + (\boldsymbol{x}_i')^T\boldsymbol{x}_i'\right) \quad (7.14)$$

Заметив, что $\boldsymbol{x}_i' = (\boldsymbol{u}^T\boldsymbol{x}_i)\boldsymbol{u}$, имеем

$$= \frac{1}{n}\sum_{i=1}^{n}\left(\|\boldsymbol{x}_i\|^2 - 2\boldsymbol{x}_i^T(\boldsymbol{u}^T\boldsymbol{x}_i)\boldsymbol{u} + ((\boldsymbol{u}^T\boldsymbol{x}_i)\boldsymbol{u})^T(\boldsymbol{u}^T\boldsymbol{x}_i)\boldsymbol{u}\right)$$

$$= \frac{1}{n}\sum_{i=1}^{n}\left(\|\boldsymbol{x}_i\|^2 - 2(\boldsymbol{u}^T\boldsymbol{x}_i)\left(\boldsymbol{x}_i^T\boldsymbol{u}\right) + (\boldsymbol{u}^T\boldsymbol{x}_i)\left(\boldsymbol{x}_i^T\boldsymbol{u}\right)\boldsymbol{u}^T\boldsymbol{u}\right) = \frac{1}{n}\sum_{i=1}^{n}\left(\|\boldsymbol{x}_i\|^2 - (\boldsymbol{u}^T\boldsymbol{x}_i)\left(\boldsymbol{x}_i^T\boldsymbol{u}\right)\right)$$

$$= \frac{1}{n}\sum_{i=1}^{n}\|\boldsymbol{x}_i\|^2 - \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{u}^T\left(\boldsymbol{x}_i\boldsymbol{x}_i^T\right)\boldsymbol{u} = \frac{1}{n}\sum_{i=1}^{n}\|\boldsymbol{x}_i\|^2 - \boldsymbol{u}^T\left(\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i\boldsymbol{x}_i^T\right)\boldsymbol{u}$$

$$= \sum_{i=1}^{n}\frac{\|\boldsymbol{x}_i\|^2}{\boldsymbol{n}} - \boldsymbol{u}^T\boldsymbol{\Sigma}\boldsymbol{u} \quad (7.15)$$

Заметим, что общая дисперсия центрированных данных (т.е. $\boldsymbol{\mu} = \mathbf{0}$) получается как

$$\mathrm{v}ar(\boldsymbol{D}) = \frac{1}{n}\sum_{i=1}^{n}\|\boldsymbol{x}_i - \mathbf{0}\|^2 = \frac{1}{n}\sum_{i=1}^{n}\|\boldsymbol{x}_i\|^2$$

Далее, из уравнения (2.28) имеем *(см. основной учебник курса)*

$$\mathrm{v}ar(\boldsymbol{D}) = tr(\boldsymbol{\Sigma}) = \sum_{i=1}^{d}\sigma_i^2$$

Таким образом, можно переписать уравнение (7.15) как

$$MSE(\boldsymbol{u}) = \mathrm{v}ar(\boldsymbol{D}) - \boldsymbol{u}^T\boldsymbol{\Sigma}\boldsymbol{u} = \sum_{i=1}^{d}\sigma_i^2 - \boldsymbol{u}^T\boldsymbol{\Sigma}\boldsymbol{u}$$

Поскольку первый член $\mathrm{v}ar(\boldsymbol{D})$ является константой для данного набора данных $\boldsymbol{D}$, вектор $\boldsymbol{u}$, который минимизирует $MSE(\boldsymbol{u})$, максимизирует второй член, проецируемую дисперсию $\boldsymbol{u}^T\boldsymbol{\Sigma}\boldsymbol{u}$. Поскольку мы знаем, что $\boldsymbol{u}_1$, доминирующий собственный вектор матрицы $\boldsymbol{\Sigma}$, максимизирует проецируемую дисперсию, имеем

$$MSE(\boldsymbol{u}_1) = \mathrm{v}ar(\boldsymbol{D}) - \boldsymbol{u}_1{}^T\boldsymbol{\Sigma}\boldsymbol{u}_1 = \mathrm{v}ar(\boldsymbol{D}) - \boldsymbol{u}_1{}^T\lambda_1\boldsymbol{u}_1 = \mathrm{v}ar(\boldsymbol{D}) - \lambda_1 \quad (7.16)$$

Таким образом, главный компонент u_1, являющийся направлением, которое максимизирует проецируемую дисперсию, также является направлением, которое минимизирует среднеквадратичную ошибку MSE.

Метод главных компонент

Наилучшая двумерная аппроксимация

Найдем наилучшее двумерное приближение к $\boldsymbol{D}$. Требуется найти направление $\mathbf{v}$, которое также максимизирует проецируемую дисперсию, но ортогонально $\boldsymbol{u}_1$. Согласно (7.9) проецируемая на $\mathbf{v}$ дисперсия определяется как

$$\sigma_{\mathbf{v}}^2 = \mathbf{v}^T \boldsymbol{\Sigma} \mathbf{v}$$

Требуется, чтобы $\mathbf{v}$ был единичным вектором, ортогональным $\boldsymbol{u}_1$, т.е.

$$\mathbf{v}^T \boldsymbol{u}_1 = 0$$
$$\mathbf{v}^T \mathbf{v} = 1$$

Тогда условие оптимизации определяется как

$$\max_{\mathbf{v}} J(\mathbf{v}) = \mathbf{v}^T \boldsymbol{\Sigma} \mathbf{v} - \alpha(\mathbf{v}^T \mathbf{v} - 1) - \beta(\mathbf{v}^T \boldsymbol{u}_1 - 0) \quad (7.17)$$

Взяв производную $J(\mathbf{v})$ по $\mathbf{v}$ и приравняв ее к нулевому вектору, получаем

$$2\boldsymbol{\Sigma}\mathbf{v} - 2\alpha\mathbf{v} - \beta\boldsymbol{u}_1 = \mathbf{0} \quad (7.18)$$

Если умножить левую часть на $\boldsymbol{u}_1^T$, то получим

$$2\boldsymbol{u}_1^T\boldsymbol{\Sigma}\mathbf{v} - 2\alpha\boldsymbol{u}_1^T\mathbf{v} - \beta\boldsymbol{u}_1^T\boldsymbol{u}_1 = 0$$

$2\mathbf{v}^T\boldsymbol{\Sigma}\boldsymbol{u}_1 - \beta = 0$, откуда следует, что

$$\beta = 2\mathbf{v}^T\lambda_1\boldsymbol{u}_1 = 2\lambda_1\mathbf{v}^T\boldsymbol{u}_1 = 0$$

При выводе выше мы использовали тот факт, что $\boldsymbol{u}_1^T\boldsymbol{\Sigma}\mathbf{v}$ $= \mathbf{v}^T\boldsymbol{\Sigma}\boldsymbol{u}_1$, и что $\mathbf{v}$ ортогонален $\boldsymbol{u}_1$. Подставляя $\beta = 0$ в уравнение (7.18), получаем

$$2\boldsymbol{\Sigma}\mathbf{v} - 2\alpha\mathbf{v} = \mathbf{0}$$

$$\boldsymbol{\Sigma}\mathbf{v} = \alpha\mathbf{v}$$

Это означает, что $\mathbf{v}$ – это еще один собственный вектор $\boldsymbol{\Sigma}$. Так же, как в уравнении (7.12), имеем $\sigma_{\mathbf{v}}^2 = \alpha$. Чтобы максимизировать дисперсию по $\mathbf{v}$, выбираем $\alpha = \lambda_2$, второе по величине собственное значение $\boldsymbol{\Sigma}$, второй главный компонент задается соответствующим собственным вектором, т.е. $\mathbf{v} = \boldsymbol{u}_2$.

Общая проецируемая дисперсия

Пусть $\boldsymbol{U}_2$ – матрица, столбцы которой соответствуют двум главным компонентам

$$\boldsymbol{U}_2 = \begin{pmatrix} | & | \\ \boldsymbol{u}_1 & \boldsymbol{u}_2 \\ | & | \end{pmatrix}$$

Для точки $\boldsymbol{x}_i \in \boldsymbol{D}$ координаты в двумерном подпространстве, натянутом на $\boldsymbol{u}_1$ и $\boldsymbol{u}_2$, могут быть вычислены по формуле (7.6) следующим образом

$$\boldsymbol{a}_i = \boldsymbol{U}_2^T \boldsymbol{x}_i$$

Предположим, что каждая точка $\boldsymbol{x}_i \in \mathbb{R}^d$ в $\boldsymbol{D}$ была спроецирована для получения ее координат $\boldsymbol{a}_i \in \mathbb{R}^d$, что дало новый набор данных $\boldsymbol{A}$. Так как предполагается, что $\boldsymbol{D}$ центрирован ($\boldsymbol{\mu} = \boldsymbol{0}$), координаты спроецированного среднего также равны нулю, т.к. $\boldsymbol{U}_2\boldsymbol{\mu} = \boldsymbol{U}_2^T\boldsymbol{0} = \boldsymbol{0}$.

Общая дисперсия для $\boldsymbol{A}$ определяется как

$$var(\boldsymbol{A}) = \frac{1}{n}\sum_{i=1}^{n} \|\boldsymbol{a}_i - \boldsymbol{0}\|^2 = \frac{1}{n}\sum_{i=1}^{n} (\boldsymbol{U}_2^T\boldsymbol{x}_i)^T\,(\boldsymbol{U}_2^T\boldsymbol{x}_i)$$

$$= \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i^T(\boldsymbol{U}_2\boldsymbol{U}_2^T)\boldsymbol{x}_i = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i^T\boldsymbol{P}_2\boldsymbol{x}_i \quad (7.19)$$

Где $\boldsymbol{P}_2$ – матрица ортогональной проекции [уравнение (7.8)], которая определяется как

$$\boldsymbol{P}_2 = \boldsymbol{U}_2\boldsymbol{U}_2^T = \boldsymbol{u}_1\boldsymbol{u}_1^T + \boldsymbol{u}_2\boldsymbol{u}_2^T$$

Подставляя это в (7.19), проецируемая общая дисперсия определяется как

$$var(\boldsymbol{A}) = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i^T\boldsymbol{P}_2\boldsymbol{x}_i \quad (7.20)$$

$$= \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i^T(\boldsymbol{u}_1\boldsymbol{u}_1^T + \boldsymbol{u}_2\boldsymbol{u}_2^T)\boldsymbol{x}_i = \frac{1}{n}\sum_{i=1}^{n} (\boldsymbol{u}_1^T\boldsymbol{x}_i)(\boldsymbol{x}_i^T\boldsymbol{u}_1) + \frac{1}{n}\sum_{i=1}^{n} (\boldsymbol{u}_2^T\boldsymbol{x}_i)(\boldsymbol{x}_i^T\boldsymbol{u}_2)$$

$$= \boldsymbol{u}_1^T\boldsymbol{\Sigma}\boldsymbol{u}_1 + \boldsymbol{u}_2^T\boldsymbol{\Sigma}\boldsymbol{u}_2 \quad (7.21)$$

Поскольку $\boldsymbol{u}_1$ и $\boldsymbol{u}_2$ – собственные вектора $\boldsymbol{\Sigma}$, имеем $\boldsymbol{\Sigma u}_1 = \lambda_1 \boldsymbol{u}_1$ и $\boldsymbol{\Sigma u}_2 = \lambda_2 \boldsymbol{u}_2$, т.е.

$$var(\boldsymbol{A}) = \boldsymbol{u}_1^T \boldsymbol{\Sigma u}_1 + \boldsymbol{u}_2^T \boldsymbol{\Sigma u}_2 = \boldsymbol{u}_1^T \lambda_1 \boldsymbol{u}_1 + \boldsymbol{u}_2^T \lambda_2 \boldsymbol{u}_2 = \lambda_1 + \lambda_2 \ (7.22)$$

Таким образом, сумма собственных значений представляет собой общую дисперсию проецируемых точек, а первые два главных компонента максимизируют эту дисперсию.

Среднеквадратичная ошибка

Покажем, что первые два главных компонента минимизируют среднеквадратичную ошибку. Среднеквадратичная ошибка задается как

$$\mathrm{M}SE = \frac{1}{n}\sum_{i=1}^{n} \|\boldsymbol{x}_i - \boldsymbol{x}_i'\|^2$$

$$= \frac{1}{n}\sum_{i=1}^{n} \left(\|\boldsymbol{x}_i\|^2 - 2\boldsymbol{x}_i^T\boldsymbol{x}_i' + (\boldsymbol{x}_i')^T\boldsymbol{x}_i'\right), \text{используя (7.14)}$$

$$= var(\boldsymbol{D}) + \frac{1}{n}\sum_{i=1}^{n} (-2\boldsymbol{x}_i^T\boldsymbol{P}_2\boldsymbol{x}_i + (\boldsymbol{P}_2\boldsymbol{x}_i)^T\boldsymbol{P}_2\boldsymbol{x}_i), \text{используя (7.7), что } \boldsymbol{x}_i' = \boldsymbol{P}_2\boldsymbol{x}_i$$

$$= var(\boldsymbol{D}) - \frac{1}{n}\sum_{i=1}^{n} (\boldsymbol{x}_i^T\boldsymbol{P}_2\boldsymbol{x}_i)$$

$$= var(\boldsymbol{D}) - var(\boldsymbol{A}), \text{используя (7.20) (7.23)}$$

Таким образом, MSE минимизируется, когда общая проецируемая дисперсия $var(\boldsymbol{A})$ максимальна. Из (7.22) имеем $MSE = var(\boldsymbol{D}) - \lambda_1 - \lambda_2$

Метод главных компонент

Наилучшая r-мерная аппроксимация

Найдем наилучшую r-мерную аппроксимация к $\boldsymbol{D}$, где $2 < r \leq d$. Предположим, что мы уже вычислили первые $j-1$ главных компонент или собственных векторов, $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_{j-1}$, соответствующих $j-1$ наибольшим собственным значениям матрицы $\boldsymbol{\Sigma}$, для $1 \leq j \leq r$. Чтобы вычислить j-й новый базисный вектор $\mathbf{v}$, мы должны убедиться, что он нормализован до единичной длины, т.е. $\mathbf{v}^T\mathbf{v} = 1$, и ортогонален всем предыдущим компонентам $\boldsymbol{u}_i$, т.е. $\boldsymbol{u}_i^T\mathbf{v} = 0$, для $1 \leq i < j$. Как и до этого, проецируемая дисперсия по $\mathbf{v}$ задается как

$$\sigma_{\mathbf{v}}^2 = \mathbf{v}^T\boldsymbol{\Sigma}\mathbf{v}$$

В сочетании с ограничениями на $\mathbf{v}$ это приводит к следующей задаче максимизации с множителями Лагранжа:

$$\max_{\mathbf{v}} J(\mathbf{v}) = \mathbf{v}^T\boldsymbol{\Sigma}\mathbf{v} - \alpha(\mathbf{v}^T\mathbf{v} - 1) - \sum_{i=1}^{j-1} \beta_i(\boldsymbol{u}_i{}^T\mathbf{v} - 0)$$

Взяв производную $J(\mathbf{v})$ по $\mathbf{v}$ и приравняв ее к нулевому вектору, получаем

$$2\boldsymbol{\Sigma}\mathbf{v} - 2\alpha\mathbf{v} - \sum_{i=1}^{j-1} \beta_i \boldsymbol{u}_i = \mathbf{0} \quad (7.24)$$

Если умножить левую часть на $\boldsymbol{u}_k^T$ для $1 \leq k < j$, то получим

$$2\boldsymbol{u}_k^T\boldsymbol{\Sigma}\mathbf{v} - 2\alpha\boldsymbol{u}_k^T\mathbf{v} - \beta_k\boldsymbol{u}_k^T\boldsymbol{u}_k - \sum_{\substack{i=1 \\ i\neq k}}^{j-1} \beta_i \boldsymbol{u}_k^T\boldsymbol{u}_i = 0$$

$$2\mathbf{v}^T\boldsymbol{\Sigma}\boldsymbol{u}_k - \beta_k = 0$$

$$\beta_k = 2\mathbf{v}^T\lambda_k\boldsymbol{u}_k = 2\lambda_k\mathbf{v}^T\boldsymbol{u}_k = 0$$

где использован тот факт, что $\boldsymbol{\Sigma}\boldsymbol{u}_k = \lambda_k\boldsymbol{u}_k$, т.к. $\boldsymbol{u}_k$ является собственным вектором, которому соответствует k-е наибольшее собственное значение λ_k матрицы $\boldsymbol{\Sigma}$. Таким образом, мы находим, что $\beta_i = 0$ для всех $i < j$ в уравнении (7.24), откуда следует

$$\boldsymbol{\Sigma}\mathbf{v} = \alpha\mathbf{v}$$

Чтобы максимизировать дисперсию по $\mathbf{v}$, устанавливаем $\alpha = \lambda_j$, j-е наибольшее собственной значение матрицы $\mathbf{\Sigma}$, где $\mathbf{v} = \boldsymbol{u}_j$ задает j-ю главную компоненту.

Таким образом, чтобы найти наилучше r-мерное приближение к D, вычисляем собственные значения матрицы Σ. Т.к. Σ является положительно полуопределенной, все ее собственные числа должны быть неотрицательными, и мы можем отсортировать их в порядке убывания следующим образом:

$$\lambda_1 \geq \lambda_2 \geq \cdots \lambda_r \geq \lambda_{r+1} \ldots \geq \lambda_d \geq 0$$

Затем выбираем r наибольших собственных значений и соответствующие им собственные векторы, чтобы сформировать лучшее r-мерное приближение.

Общая проецируемая дисперсия

Пусть $\boldsymbol{U_r}$ – r-мерная матрица базисных векторов $\boldsymbol{U_r} = \begin{pmatrix} | & | & . & | \\ \boldsymbol{u}_1 & \boldsymbol{u}_2 & \ldots & \boldsymbol{u}_r \\ | & | & . & | \end{pmatrix}$

с матрицей проекции, заданной как

$$\boldsymbol{P_r} = \boldsymbol{U_r U_r^T} = \sum_{i=1}^{r} \boldsymbol{u_i u_i^T}$$

Пусть $\boldsymbol{A}$ - набор данных, сформированный координатами проецируемых точек в r-мерном подпространстве, т.е. $\boldsymbol{a_i} = \boldsymbol{U_r^T x_i}$, и пусть $\boldsymbol{x_i'} = \boldsymbol{P_r x_i}$ обозначает спроецированную точку в исходном d-мерном пространстве. Из (7.19), (7.21) и (7.22) проецируемая дисперсия определяется как

$$var(\boldsymbol{A}) = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x_i^T P_r x_i} = \sum_{i=1}^{r} \boldsymbol{u_i^T \Sigma u_i} = \sum_{i=1}^{r} \lambda_i$$

Таким образом, общая проецируемая дисперсия – это просто сумма r наибольших собственных значений матрицы $\boldsymbol{\Sigma}$.

Среднеквадратичная ошибка

Из (7.23), среднеквадратичная ошибка в r-измерениях может быть записана как

$$\mathrm{M}SE = \frac{1}{n}\sum_{i=1}^{n}\|\boldsymbol{x}_i - \boldsymbol{x}_i'\|^2 = var(\boldsymbol{D}) - var(\boldsymbol{A}) = var(\boldsymbol{D}) - \sum_{i=1}^{r}\boldsymbol{u}_i^T\boldsymbol{\Sigma}\boldsymbol{u}_i = var(\boldsymbol{D}) - \sum_{i=1}^{r}\lambda_i$$

Первые r главных компонент максимизируют проецируемую дисперсию $var(\boldsymbol{A})$ и, таким образом, они также минимизируют MSE.

Общая дисперсия

Общая дисперсия $\boldsymbol{D}$ инвариантна к изменению базисных векторов. Следовательно, имеем следующее тождество:

$$var(\boldsymbol{D}) = \sum_{i=1}^{d}\sigma_i^2 = \sum_{i=1}^{d}\lambda_i$$

Выбор размерности

Часто мы можем не знать, сколько измерений r использовать для хорошей аппроксимации. Одним из критериев выбора r является вычисление доли общей дисперсии, охваченной первыми r главными компонентами

$$f(r) = \frac{\lambda_1 + \lambda_2 + \cdots + \lambda_r}{\lambda_1 + \lambda_2 + \cdots + \lambda_d} = \frac{\sum_{i=1}^{r} \lambda_i}{\sum_{i=1}^{d} \lambda_i} = \frac{\sum_{i=1}^{r} \lambda_i}{var(\boldsymbol{D})} \quad (7.25)$$

Алгоритм 7.1 демонстрирует псевдокод для алгоритма метода главных компонент PCA.

АЛГОРИТМ 7.1. Метод главных компонент PCA

$\boldsymbol{PCA}(\boldsymbol{D}, \alpha)$:

1. $\boldsymbol{\mu} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i$ // вычисление среднего
2. $\boldsymbol{Z} = \boldsymbol{D} - \boldsymbol{1} \cdot \boldsymbol{\mu}^T$ // центрирование данных
3. $\boldsymbol{\Sigma} = \frac{1}{n}(\boldsymbol{Z}^T\boldsymbol{Z})$ // вычисление ковариационной матрицы
4. $(\lambda_1, \lambda_2, \ldots, \lambda_d) = eigenvalues(\boldsymbol{\Sigma})$ // вычисление собственных значений
5. $\boldsymbol{U} = (\boldsymbol{u}_1 \quad \boldsymbol{u}_2 \quad \ldots \quad \boldsymbol{u}_d) = eigenvectors(\boldsymbol{\Sigma})$ // собственных векторов
6. $f(r) = \frac{\sum_{i=1}^{r} \lambda_i}{\sum_{i=1}^{d} \lambda_i}$, for all r = 1, 2, …, d // доля от общей дисперсии
7. $Choose\ smallest\ r\ so\ that\ f(r) \geq \alpha$ // выбор размерности
8. $\boldsymbol{U}_r = (\boldsymbol{u}_1 \quad \boldsymbol{u}_2 \quad \ldots \quad \boldsymbol{u}_r)$ // уменьшенный базис
9. $\boldsymbol{A} = \{\boldsymbol{a}_i | \boldsymbol{a}_i = \boldsymbol{U}_r^T \boldsymbol{x}_i, for\ i = 1, \ldots, n\}$ // данные уменьшенной размерности

Ядерный анализ главных компонент

Анализ главных компонент может быть расширен, чтобы искать нелинейные "направления" в данных с использованием ядерных методов. То есть, вместо поиска линейных комбинаций во входном пространстве, алгоритм ищет линейные комбинации в пространстве признаков, полученном из нелинейного преобразования входного пространства.

Пусть ϕ – отображение из входного пространства в пространство признаков. Каждой точке в пространстве признаков соответствует образ $\phi(\boldsymbol{x}_i)$ точки $\boldsymbol{x}_i$ во входном пространстве. Во входном пространстве первая главная компонента показывает направление наибольшей дисперсии; её собственный вектор совпадает с наибольшим собственным значением в матрице ковариантности. Таким же образом, через поиск собственного вектора, соответствующий наибольшему собственному значению в матрице ковариантности в пространстве признаков, можно найти первую главную компоненту в пространстве признаков $\boldsymbol{u}_1$ (с $\boldsymbol{u}_1^T \boldsymbol{u}_1 = 1$):

$$\boldsymbol{\Sigma}_\phi = \lambda_1 \boldsymbol{u}_1 \ (7.29)$$

где $\boldsymbol{\Sigma}_\phi$ – матрица ковариантности в пространстве признаков, определенная как

$$\boldsymbol{\Sigma}_\phi = \frac{1}{n} \sum_{i=1}^{n} \phi(\boldsymbol{x}_i) \phi(\boldsymbol{x}_i)^T \ (7.30)$$

Здесь предполагается, что точки центрированы, т.е. $\phi(\boldsymbol{x}_i) = \phi(\boldsymbol{x}_i) - \boldsymbol{\mu}_\phi$, где $\boldsymbol{\mu}_\phi$ – среднее в пространстве признаков.

Подставляя (7.30) в (7.29) получаем:

$$\left(\frac{1}{n}\sum_{i=1}^{n}\phi(\boldsymbol{x}_i)\phi(\boldsymbol{x}_i)^T\right)\boldsymbol{u}_1 = \lambda_1\boldsymbol{u}_1$$

$$\frac{1}{n}\sum_{i=1}^{n}\phi(\boldsymbol{x}_i)(\phi(\boldsymbol{x}_i)^t\boldsymbol{u}_1) = \lambda_1\boldsymbol{u}_1 \quad (7.31)$$

$$\sum_{i=1}^{n}\left(\frac{\phi(\boldsymbol{x}_i)^T\boldsymbol{u}_1}{n\lambda_1}\right)\phi(\boldsymbol{x}_i) = \boldsymbol{u}_1$$

$$\sum_{i=1}^{n} c_i\phi(\boldsymbol{x}_i) = \boldsymbol{u}_1 \quad (7.32)$$

где $c_i = \frac{\phi(\boldsymbol{x}_i)^T\boldsymbol{u}_1}{n\lambda_1}$ – скалярная величина. Из (7.32) видно, что лучшее направление в пространстве признаков, $\boldsymbol{u_1}$ есть линейная комбинация преобразованных точек, где скаляры c_i показывают важность каждой точки в направлении наибольшей дисперсии.

Теперь можно подставить (7.32) назад в (7.31) и получить

$$\left(\frac{1}{n}\sum_{i=1}^{n}\phi(\boldsymbol{x}_i)\phi(\boldsymbol{x}_i)^T\right)\left(\sum_{i=1}^{n}c_j\phi(\boldsymbol{x}_i)\right)=\lambda_1\sum_{i=1}^{n}c_i\phi(\boldsymbol{x}_i)$$

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n}c_j\phi(\boldsymbol{x}_i)\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)=\lambda_1\sum_{i=1}^{n}c_i\phi(\boldsymbol{x}_i)$$

$$\sum_{i=1}^{n}\left(\phi(\boldsymbol{x}_i)\sum_{j=1}^{n}c_j\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)\right)=n\lambda_1\sum_{i=1}^{n}c_i\phi(\boldsymbol{x}_i)$$

Заменяем скалярное произведение в пространстве признаков, $(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)$, соответствующей ядерной функцией во входном пространстве, а именно $K(\boldsymbol{x}_i,\boldsymbol{x}_j)$

$$\sum_{i=1}^{n}\left(\phi(\boldsymbol{x}_i)\sum_{j=1}^{n}c_jK(\boldsymbol{x}_i,\boldsymbol{x}_j)\right)=n\lambda_1\sum_{i=1}^{n}c_i\phi(\boldsymbol{x}_i)\ (7.33)$$

Здесь предполагается, что точки в пространстве признаков уже центрированы, т.е. ядерная матрица была центрирована через уравнение (5.14):

$$\boldsymbol{K} = \left(\boldsymbol{I} - \frac{1}{n}\mathbf{1}_{n\times n}\right)\boldsymbol{K}\left(\boldsymbol{I} - \frac{1}{n}\mathbf{1}_{n\times n}\right)$$

где $\boldsymbol{I}$ – единичная матрица, у которой элементы главной диагонали 1, а остальные 0, и $\mathbf{1}_{n\times n}$ – матрица $n \times n$, у которой все элементы – 1.

Мы заменили одно скалярное произведение ядерной функцией. Чтобы убедится, что все вычисления в пространстве признаков выполняются только в терминах скалярного произведение, возьмем любую точку $\phi(\boldsymbol{x}_k)$ и умножим (7.33) на неё с обеих сторон чтобы получить

$$\begin{gathered}\sum_{i=1}^{n}\left(\phi(\boldsymbol{x}_k)\phi(\boldsymbol{x}_i)\sum_{j=1}^{n}c_j K(\boldsymbol{x}_i,\boldsymbol{x}_j)\right) = n\lambda_1\sum_{i=1}^{n}c_i\phi(\boldsymbol{x}_i)\phi(\boldsymbol{x}_k) \\ \sum_{i=1}^{n}\left(K(\boldsymbol{x}_k,\boldsymbol{x}_i)\sum_{j=1}^{n}c_j K(\boldsymbol{x}_i,\boldsymbol{x}_j)\right) = n\lambda_1\sum_{i=1}^{n}c_i K(\boldsymbol{x}_k,\boldsymbol{x}_i)\end{gathered} \tag{7.34}$$

Затем, обозначим $\boldsymbol{K}_i$ строку i центрированной ядерной матрицы, записанную как вектор-столбец:

$$\boldsymbol{K}_i = (K(\boldsymbol{x}_i, \boldsymbol{x}_1) \quad K(\boldsymbol{x}_i, \boldsymbol{x}_2) \quad \cdots \quad K(\boldsymbol{x}_i, \boldsymbol{x}_n))^T$$

Пусть $\boldsymbol{c}$ обозначает вектор-столбец весов

$$\boldsymbol{c} = (c_1 \quad c_2 \quad \cdots \quad c_n)^T$$

Теперь можно записать (7.34) как:

$$\sum_{i=1}^{n} K(\boldsymbol{x}_k, \boldsymbol{x}_i)\boldsymbol{K}_i^T\boldsymbol{c} = n\lambda_1\boldsymbol{K}_k^T\boldsymbol{c}$$

На самом деле, поскольку можно выбрать любые из n точек $\phi(\boldsymbol{x}_k)$ в пространстве признаков, чтобы получить (7.34), имеется набор уравнений:

$$\begin{aligned}
\sum_{i=1}^{n} K(\boldsymbol{x}_1, \boldsymbol{x}_i)\boldsymbol{K}_i^T\boldsymbol{c} &= n\lambda_1\boldsymbol{K}_1^T\boldsymbol{c} \\
\sum_{i=1}^{n} K(\boldsymbol{x}_2, \boldsymbol{x}_i)\boldsymbol{K}_i^T\boldsymbol{c} &= n\lambda_1\boldsymbol{K}_2^T\boldsymbol{c} \\
\vdots &= \vdots \\
\sum_{i=1}^{n} K(\boldsymbol{x}_2, \boldsymbol{x}_i)\boldsymbol{K}_i^T\boldsymbol{c} &= \sum_{i=1}^{n} K(\boldsymbol{x}_k, \boldsymbol{x}_i)\boldsymbol{K}_n^T\boldsymbol{c}
\end{aligned}$$

Можно представить эти n уравнений как: $\boldsymbol{K}^2\boldsymbol{c} = n\lambda_1\boldsymbol{K}\boldsymbol{c}$

где $\boldsymbol{K}$ – центрированная ядерная матрица. Умножением на $\boldsymbol{K}^{-1}$ получаем:

$$\begin{aligned}
\boldsymbol{K}^{-1}\boldsymbol{K}^2\boldsymbol{c} &= n\lambda_1\boldsymbol{K}^{-1}\boldsymbol{K}\boldsymbol{c} \\
\boldsymbol{K}\boldsymbol{c} &= n\lambda_1\boldsymbol{c} \qquad (7.35) \\
\boldsymbol{K}\boldsymbol{c} &= \eta_1\boldsymbol{c}
\end{aligned}$$

где $\eta_1 = n\lambda_1$. Таким образом, вектор весов $\boldsymbol{c}$ есть собственный вектор, соответствующий наибольшему собственному значению η_1 ядерной матрицы $\boldsymbol{K}$.

Как только найдено $\boldsymbol{c}$, её можно подставить обратно в (7.32), чтобы получить первую ядерную основную компоненту $\boldsymbol{u}_1$. Единственное ограничение – надо нормализовать $\boldsymbol{u}_1$ до единичного вектора следующим образом:

$$\boldsymbol{u}_1^T \boldsymbol{u}_1 = 1$$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} c_i c_j \phi(\boldsymbol{x}_i)^T \phi\left(\boldsymbol{x}_j\right) = 1$$

$$\boldsymbol{c}^T \boldsymbol{K}\boldsymbol{c} = 1$$

Из (7.35) имеем:

$$\boldsymbol{c}^T(\eta_1 \boldsymbol{c}) = 1$$

$$\eta_1 \boldsymbol{c}^T \boldsymbol{c} = 1$$

$$\|\boldsymbol{c}\|^2 = \frac{1}{\eta_1}$$

Однако, поскольку $\boldsymbol{c}$ – собственный вектор $\mathbf{K}$, он будет иметь единичную норму. То есть, чтобы удостоверится, что $\boldsymbol{u}_1$ – единичный вектор, надо масштабировать вектор весов $\boldsymbol{c}$ так, что его норма $\|c\| = \sqrt{\frac{1}{\eta_1}}$, что можно получить, домножив $\boldsymbol{c}$ на $\sqrt{\frac{1}{\eta_1}}$.

В общем случае, поскольку мы не отображаем входные точки в пространство признаков через ϕ, невозможно непосредственно вычислить главное направление в терминах $\phi(\boldsymbol{x}_i)$, как показано в (7.32). Однако, здесь важно, что любую точку $\phi(\boldsymbol{x})$ можно отобразить на главное направление $\boldsymbol{u}_1$ следующим образом:

$$\boldsymbol{u}_1^T \phi(\boldsymbol{x}) = \sum_{i=1}^{n} c_i \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}) = \sum_{i=1}^{n} c_i K(\boldsymbol{x}_i, \boldsymbol{x})$$

что требует только ядерные операции. Когда $\boldsymbol{x} = \boldsymbol{x}_i$ – одна из входных точек, проекция $\phi(\boldsymbol{x}_i)$ на главную компоненту $\boldsymbol{u}_1$ может быть записана как скалярное произведение

$$\boldsymbol{a}_i = \boldsymbol{u}_1^T \phi(\boldsymbol{x}_i) = \boldsymbol{K}_i^T \boldsymbol{c} \quad (7.36)$$

где $\boldsymbol{K}_i$ – вектор-столбец, соответствующий i-й строке в ядерной матрице. Таким образом, мы показали, что все вычисления, или для решения главной компоненты, или для проекции точек, могут быть выполнены с использованием одной только ядерной функции. Затем, мы можем получить дополнительные главные компоненты путём решения 7.35 для других собственных значений и собственных векторов. Другими словами, отсортировав собственные значения $\boldsymbol{K}$ в порядке убывания $\eta_1 \geq \eta_2 \geq \cdots \geq \eta_n \geq 0$, можно получить j-ю главную компоненту как соответствующий собственный вектор $\boldsymbol{c}_j$, который должен быть нормализован по норме $\|\boldsymbol{c}_j\| = \sqrt{\frac{1}{n}}$, при условии $\eta_j > 0$. И поскольку $\eta_j = n\lambda_j$, дисперсия по j-й главной компоненте определяется как $\lambda_j = \frac{\eta_j}{n}$.

Алгоритм 7.2 приводит псевдокод для метода ядерного PCA.

Алгоритм 7.2. Ядерный анализ главных компонент

KernelPCA$(\boldsymbol{D}, K, \alpha)$:

1. $\boldsymbol{K} = \{K(\boldsymbol{x}_i, \boldsymbol{x}_j)\}_{\{i,j=1\ldots n\}}$ // вычислить ядерную матрицу $n \times n$
2. $\boldsymbol{K} = \left(\boldsymbol{I} - \frac{1}{n}\mathbf{1}_{n\times n}\right)\boldsymbol{K}\left(\boldsymbol{I} - \frac{1}{n}\mathbf{1}_{n\times n}\right)$ // центрировать ядерную матрицу
3. $(\eta_1, \eta_2, \ldots, \eta_d) = \text{eigenvalues}(\boldsymbol{K})$ // собственные значения
4. $(\boldsymbol{c}_1 \quad \boldsymbol{c}_2 \quad \cdots \quad \boldsymbol{c}_n) = \text{eigenvectors}(\boldsymbol{K})$ // собственные вектора
5. $\lambda_i = \frac{\eta_i}{n}, \forall i = 1, \ldots, n$ // вычислить дисперсию каждой компоненты
6. $\boldsymbol{c}_i = \sqrt{\frac{1}{\eta_i}} \cdot \boldsymbol{c}_i, \forall i = 1, \ldots, n$ // убедится, что $\boldsymbol{u}_i^T \boldsymbol{u}_i = 1$
7. $f(r) = \frac{\sum_{i=1}^{r} \lambda_i}{\sum_{i=1}^{d} \lambda_i}, \forall r = 1,2, \ldots, d$ // часть общей дисперсии
8. Выбрать наименьшую r такую, что $f(r) \geq \alpha$ // выбор размерности
9. $\boldsymbol{C}_r = (\boldsymbol{c}_1 \quad \boldsymbol{c}_2 \quad \cdots \quad \boldsymbol{c}_r)$ // Уменьшенный базис
10. $\boldsymbol{A} = \{\boldsymbol{a_i} | \boldsymbol{a_i} = \boldsymbol{C}_r^T \boldsymbol{K}_i, \forall i = 1, \ldots, n\}$ // Данные меньшей размерности

Сингулярное разложение

Анализ главных компонентов является частным случаем более общего метода разложения матриц, называемого сингулярным разложением (Singular Value Decomposition - SVD).

SVD обобщает приведенную выше факторизацию для любой матрицы. В частности, для матрицы данных $\boldsymbol{D}$ размера $n \times d$ с n точками и d столбцами SVD факторизует $\boldsymbol{D}$ следующим образом:

$$\boldsymbol{D} = \boldsymbol{L}\Delta\boldsymbol{R}^{\boldsymbol{T}} \quad 7.38$$

где $\boldsymbol{L}$ - ортогональная матрица размера $n \times n$, $\boldsymbol{R}$ - ортогональная матрица размера $d \times d$, а $\boldsymbol{\Delta}$ - «диагональная» матрица $n \times d$. Столбцы $\boldsymbol{L}$ называются левыми сингулярными векторами, а столбцы $\boldsymbol{R}$ (или строки $\boldsymbol{R}^{\boldsymbol{T}}$) называются правыми сингулярными векторами. Матрица $\boldsymbol{\Delta}$ определяется как

$$\boldsymbol{\Delta}(\mathrm{i},\mathrm{j}) = \begin{cases} \delta_i, & \text{если } i = j \\ 0, & \text{если } i \neq j \end{cases}$$

где $i = 1, \ldots, n$ и $j = 1, \ldots, d$. Элементы $\boldsymbol{\Delta}(\mathrm{i},\mathrm{i}) = \delta_i$ вдоль главной диагонали $\boldsymbol{\Delta}$ называются сингулярными числами $\boldsymbol{D}$, и все они неотрицательны.

Если ранг $\boldsymbol{D}$ равен $r \leq \min(n, d)$, то существует только r ненулевых особых значений, которые, как мы предполагаем, упорядочены следующим образом:

$$\delta_1 \geq \delta_2 \geq \cdots \geq \delta_r \geq 0$$

Можно отбросить эти левые и правые сингулярные вектора, которые соответствуют нулевым сингулярным значениям, чтобы получить сокращенный SVD в следующем виде:

$$\boldsymbol{D} = \boldsymbol{L_r \Delta_r R_r^T} \quad 7.39$$

где $\boldsymbol{L_r}$ - матрица левых сингулярных векторов размера $n \times r$, $\boldsymbol{R_r}$ - матрица правых сингулярных векторов размера $d \times r$, а $\boldsymbol{\Delta_r}$ - диагональная матрица размера $r \times r$, содержащая положительные сингулярные вектора. Уменьшенный SVD приводит непосредственно к спектральному разложению $\boldsymbol{D}$, заданному как

$$\boldsymbol{D} = \boldsymbol{L_r \Delta_r R_r^T} = \begin{pmatrix} | & | & & | \\ l_1 & l_2 & \dots & l_r \\ | & | & & | \end{pmatrix} \begin{pmatrix} \delta_1 & 0 & \cdots & 0 \\ 0 & \delta_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \delta_r \end{pmatrix} \begin{pmatrix} - & r_1^T & - \\ - & r_2^T & - \\ - & \vdots & - \\ - & r_r^T & - \end{pmatrix}$$

$$= \delta_1 l_1 r_1^T + \delta_2 l_2 r_2^T + \cdots + \delta_r l_r r_r^T = \sum_{i=1}^{r} \delta_i l_i r_i^T$$

Спектральное разложение $\boldsymbol{D}$ представляет собой сумму матриц первого ранга вида $\delta_i l_i r_i^T$. Выбрав q наибольших сингулярных значений $\delta_1, \delta_2, \dots, \delta_q$ и соответствующие левый и правый сингулярные вектора, мы получаем наилучшее приближение ранга q к исходной матрице $\boldsymbol{D}$.

То есть, если $\boldsymbol{D}_q$ - матрица, определенная как

$$\boldsymbol{D}_q = \sum_{i=1}^{r} \delta_i l_i r_i^T$$

то можно показать, что $\boldsymbol{D}_q$ - это матрица ранга q, которая минимизирует следующее выражение $\left\|\boldsymbol{D} - \boldsymbol{D}_q\right\|_F$

где$\|\boldsymbol{A}\|_F$ - норма Фробениуса матрицы $\boldsymbol{A}$ размера $n \times d$, определяемая как

$$\|\boldsymbol{A}\|_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{d} \boldsymbol{A}(i,j)^2}$$

Связь между SVD и PCA

Предположим, что матрица $\boldsymbol{D}$ центрирована, и, что она разложена с помощью SVD [ур-е. (7.38)] как $\boldsymbol{D} = \boldsymbol{L\Delta R^T}$. Рассмотрим матрицу рассеяния для $\boldsymbol{D}$, заданную как $\boldsymbol{D^T D}$. Имеем

$$\boldsymbol{D^T D} = \left(\boldsymbol{L\Delta R^T}\right)^{\boldsymbol{T}}\left(\boldsymbol{L\Delta R^T}\right) = \boldsymbol{R}\Delta^T\boldsymbol{L^T L\Delta R^T} = \boldsymbol{R}(\Delta^T\Delta)\boldsymbol{R^T} = \boldsymbol{R}\left(\Delta_d^2\right)\boldsymbol{R^T} \quad 7.40$$

где Δ_d^2 - диагональная матрица $d \times d$, определенная как $\Delta_d^2(i,i) = \delta_i^2$ для $i = 1, \ldots, d$. Причем только $r \leq \min(d,n)$ из этих собственных чисел положительны, а остальные - нули.

Поскольку ковариационная матрица центрированного $\boldsymbol{D}$ задается как $\Sigma = \frac{\boldsymbol{1}}{\boldsymbol{n}}\boldsymbol{D}^T\boldsymbol{D}$ и поскольку ее можно разложить как $\Sigma = \boldsymbol{U\Lambda U}^T$ через PCA [ур-е. (7.37)], имеем

$$\boldsymbol{D^T D} = n\Sigma = n\boldsymbol{U\Lambda U}^T = \boldsymbol{U}(n\boldsymbol{\Lambda})\boldsymbol{U}^T \quad 7.41$$

Приравнивая уравнение (7.40) и уравнение (7.41), мы получаем, что правые сингулярные вектора $\boldsymbol{R}$ совпадают с собственными векторами Σ. Кроме того, соответствующие сингулярные значения $\boldsymbol{D}$ связаны с собственными числами Σ через выражение

$$n\lambda_i = \delta_i^2 \text{ или } \lambda_i = \frac{\delta_i^2}{n} \text{ для } i = 1, \ldots, d \quad 7.42$$

Рассмотрим теперь матрицу $\boldsymbol{DD^T}$. Имеем

$$\boldsymbol{DD^T} = \left(\boldsymbol{L\Delta R^T}\right)\left(\boldsymbol{L\Delta R^T}\right)^{\boldsymbol{T}} = \boldsymbol{L\Delta R^T R\Delta^T L^T} = \boldsymbol{L}(\Delta\Delta^T)\boldsymbol{L^T} = \boldsymbol{L\Delta_n^2 L^T}$$

где Δ_n^2 - диагональная матрица $d \times d$, определенная как $\Delta_n^2(i,i) = \delta_i^2$ для $i = 1, \ldots, d$. Только r из этих сингулярных значений положительны, а все остальные - нули. Таким образом, левые сингулярные вектора $\boldsymbol{L}$ являются собственными векторами матрицы $n \times n\,\boldsymbol{DD^T}$, а соответствующие собственные числа задаются как δ_i^2.

Вопросы?