О методе нелинейного понижения размерности Laplacian Eigenmaps

Вячеслав Боровицкий

26 февраля 2021 г.

Структура презентации

1. О методах понижения размерности
2. Общая схема Laplacian eigenmaps, а также Isomap и LLE
3. Постановка задачи Laplacian Eigenmaps.
4. Алгоритм Laplacian Eigenmaps.
5. Про оператор Лапласа на графе.
6. Мотивация на основе римановых многообразий.
7. Примеры собственных функций оператора Лапласа-Бельтрами.
8. Пример
9. Второй пример

О методах понижения размерности

Снижение размерности — это преобразование данных, состоящее в уменьшении числа переменных путём получения главных переменных.

— ru.wikipedia.org/wiki/Снижение_размерности

Зачем?

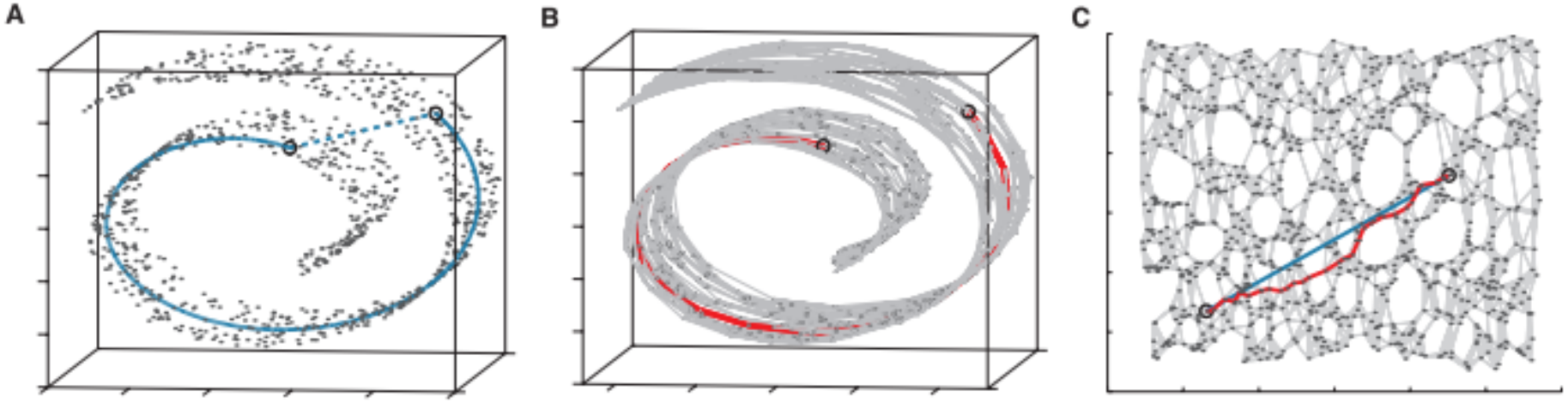
- Визуализация.
- Сжатие.
- Денойзинг, выделение фитч.

Как?

- PCA (проекция на аффинное многообразие).
- Kernel PCA (нелинейное преобразование + PCA).
- Вариации Kernel PCA: Isomap, LLE, Laplacian eigenmaps.
- Нейросети-автоэнкодеры.
- Многое другое, например t-SNE, UMAP, Mapper.

Общая схема Laplacian eigenmaps, Isomap, LLE

Алгоритм Laplacian eigenmaps, а также алгоритмы Isomap и LLE работают по схеме:

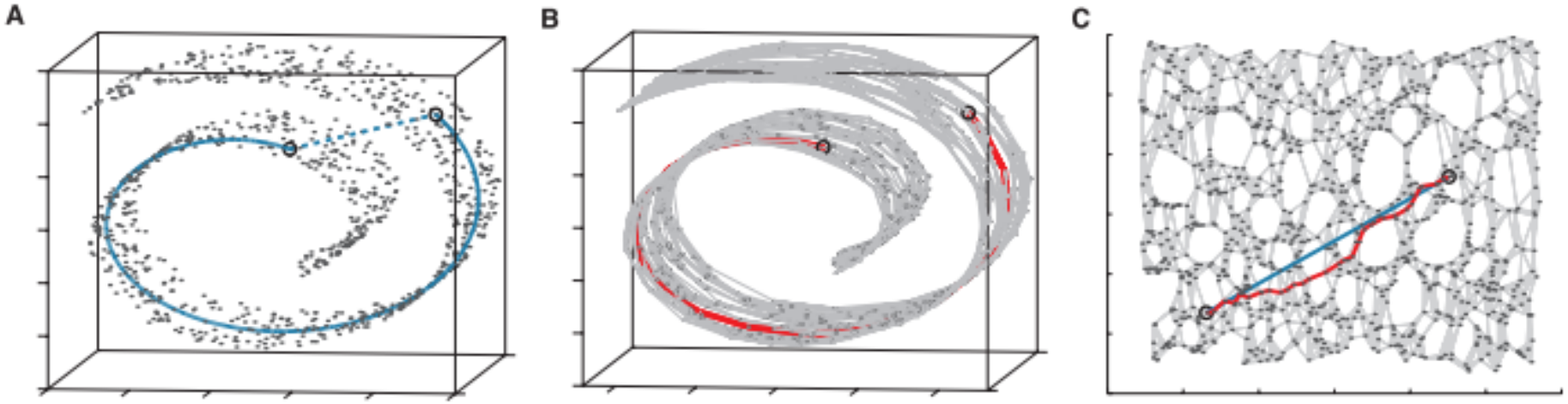


Облако Граф Вложение

- На первом этапе каждого из алгоритмов строится граф соседей. Чаще всего граф k ближайших соседей или граф ε-близких соседей.

Общая схема Laplacian eigenmaps, Isomap, LLE

Алгоритм Laplacian eigenmaps, а также алгоритмы Isomap и LLE работают по схеме:

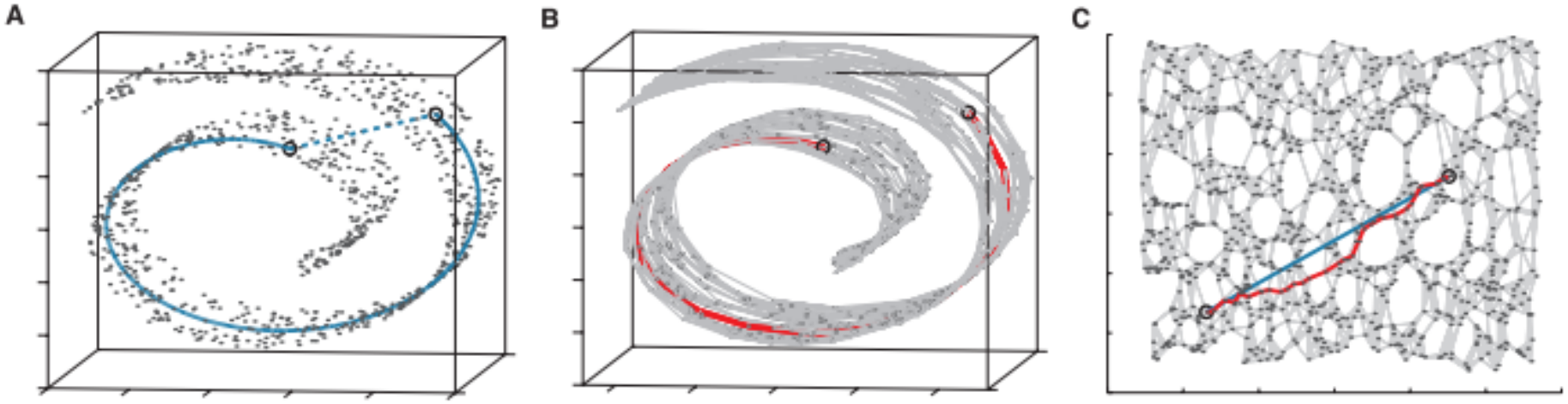


Облако Граф Вложение

- Считается, что граф приближает структуру многообразия:
  - Laplacian Eigenmaps и LLE считают, что граф хорошо кодирует близость точек (соседи на графе — близкие точки на многообразии).
  - Isomap считает, что расстояния по графу хорошо приближают геодезические расстояния.

Общая схема Laplacian eigenmaps, Isomap, LLE

Алгоритм Laplacian eigenmaps, а также алгоритмы Isomap и LLE работают по схеме:



Облако Граф Вложение

- Само построение графа — черная магия: часто построенный на практике граф плохо приближает структуру многообразия.
  Мы от этого абстрагируемся и верим в лучшее.

Помогают в нашей вере работы типа

A graph discretization of the Laplace-Beltrami operator

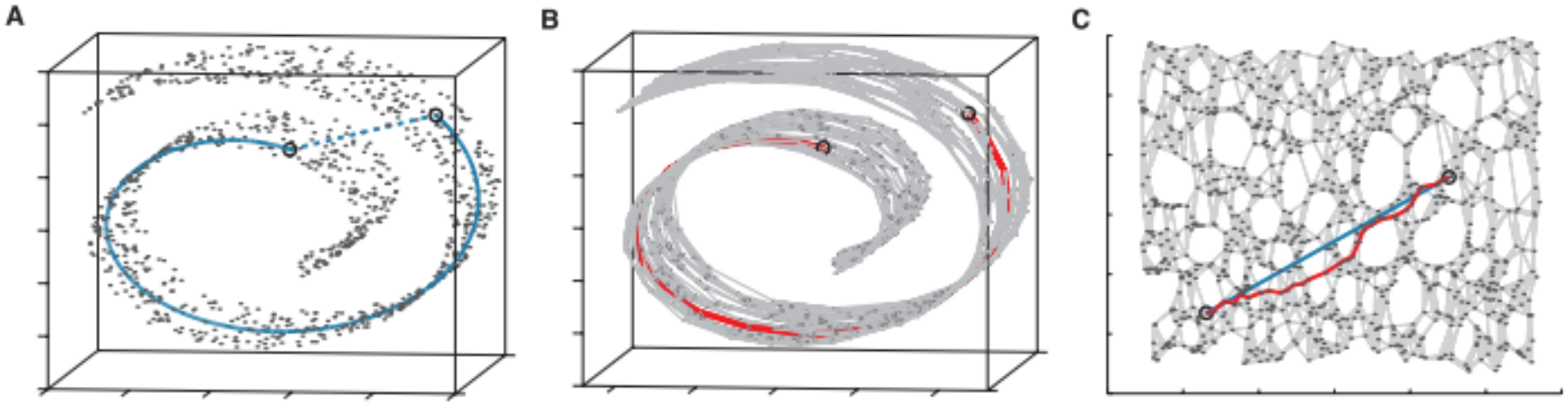
D Burago, S Ivanov, Y Kurylev - arXiv preprint arXiv:1301.2222, 2013 - arxiv.org

Traditionally, **discretization** in Riemannian geometry is associated with triangu- lations and other polyhedral approximations. This approach works perfectly well in dimension two but meets a number of obstacles in higher dimensions. It is now clear, due to works of Cheeger, Petrunin, Panov and others (see [7, 14, 13, 12]) that in dimensions beyond three polyhedral structures are too rigid to serve as discrete models of Riemannian spaces with curvature bounds. In some applications, we get a Riemannian manifold as a cloud of points with approximate distances …

которые устанавливают теоретическую сходимость в подходящем для нас смысле (станет яснее потом).

Общая схема Laplacian eigenmaps, Isomap, LLE

Алгоритм Laplacian eigenmaps, а также алгоритмы Isomap и LLE работают по схеме:

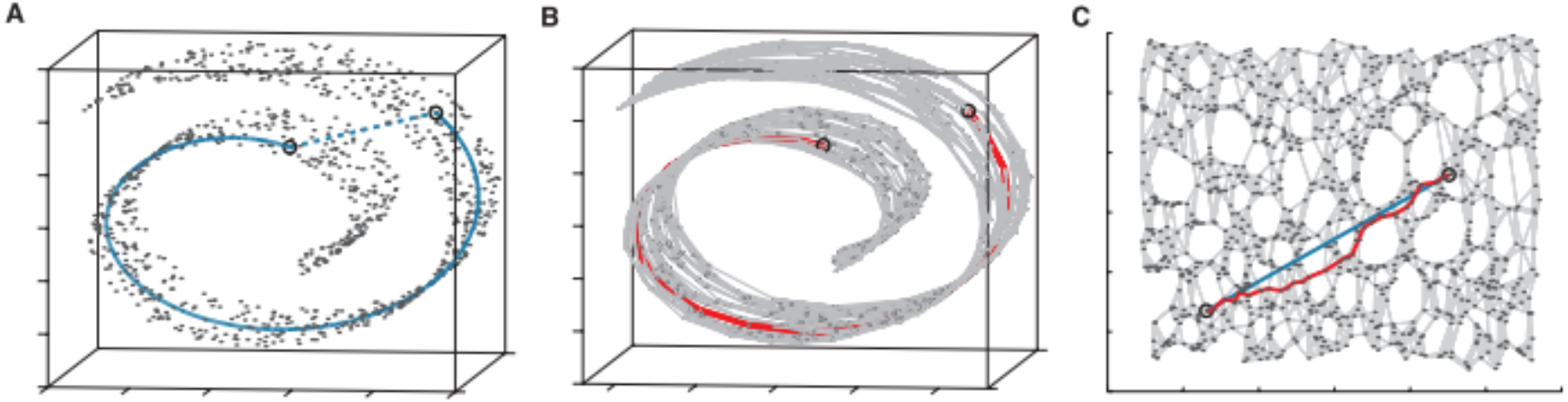


Облако Граф Вложение

- Построение вложения по графу
  - в Laplacian Eigenmaps и LLE строится так, чтобы образы близких на графе точек оставались близкими.
  - в Isomap происходит с помощью MultiDimensional Scaling (MDS) так, чтобы расстояния образов точек из облака были близки к расстояниям между ними на графе.

Общая схема Laplacian eigenmaps, Isomap, LLE

Алгоритм Laplacian eigenmaps, а также алгоритмы Isomap и LLE работают по схеме:

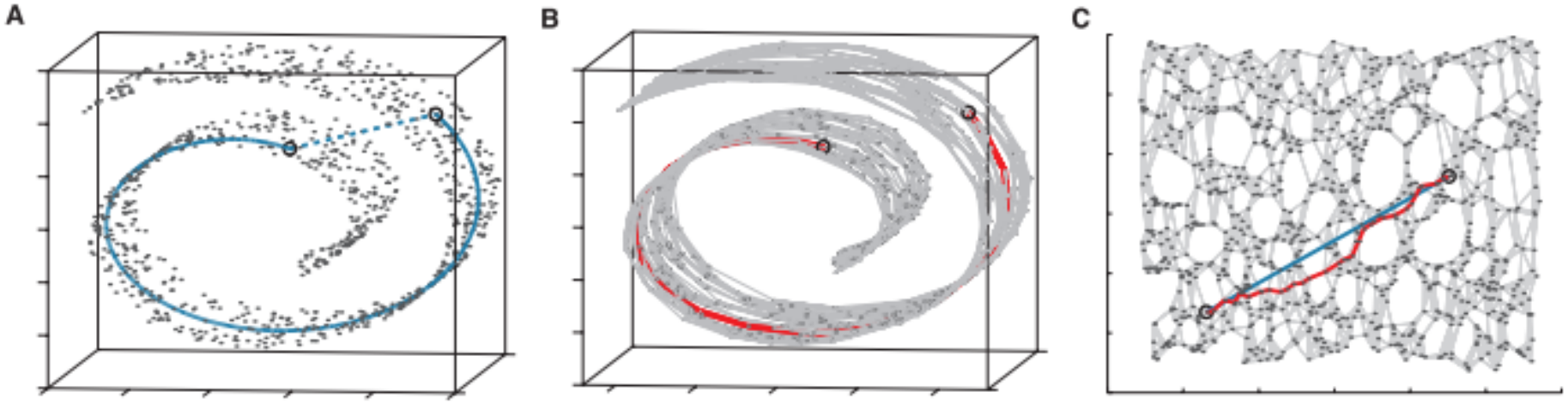


Облако Граф Вложение

- С одной стороны, Isomap решает более правильную задачу, пытаясь не только близкие точки переводить в близкие, но и оставить далекие точки далекими.

Общая схема Laplacian eigenmaps, Isomap, LLE

Алгоритм Laplacian eigenmaps, а также алгоритмы Isomap и LLE работают по схеме:

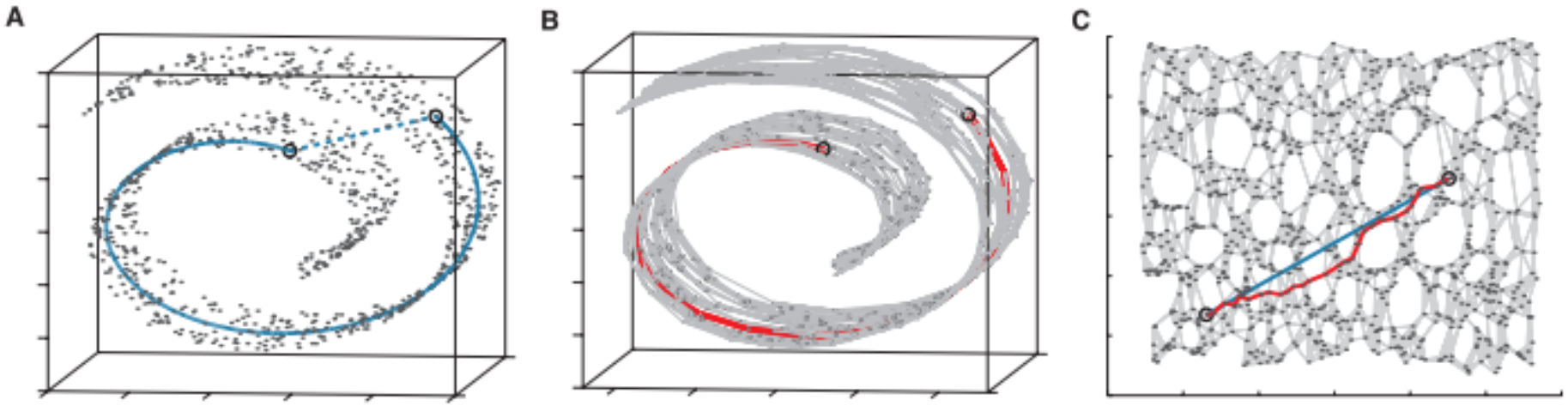


Облако Граф Вложение

- С другой стороны, Laplacian Eigenmaps и LLE не требуют от графа воспроизведения геодезических расстояний (требуют только адекватного воспроизведения близости). Их предположения чаще выполняются на практике.

Общая схема Laplacian eigenmaps, Isomap, LLE

Алгоритм Laplacian eigenmaps, а также алгоритмы Isomap и LLE работают по схеме:



Облако Граф Вложение

- Все эти алгоритмы можно понимать, как разные способы построить ядро, отражающее структуру облака точек, плюс Kernel PCA. См. про это Ham, ..., Schölkopf (2004).

Постановка задачи Laplacian Eigenmaps

Данные: ~~N векторов $x_1, x_2, \ldots, x_N \in \mathbb{R}^D$~~.
Граф, определяющий близость векторов.

Цель: Найти вложение этих векторов в $\mathbb{R}^P$, которое оставляет близкие точки точки близкими.

Близость векторов задается графом с матрицей смежности W:

- если $W_{ij} = 1$, то x_i и x_j близки (являются соседями),
- если $W_{ij} = 0$, то x_i и x_j далеки (не являются соседями).

Обычно это граф ближайших соседей в смысле расстояния $\mathbb{R}^D$.

(Почти) формальная постановка

Найти вектора $y_1, \ldots, y_N \in \mathbb{R}^P$ так, чтобы

$$\sum_{i=1..N} \sum_{j=1..N} \|y_i - y_j\|^2 W_{ij} \to \min$$

при некоторых дополнительных условиях.

Laplacian Eigenmaps для $P = 1$

(Почти) формальная постановка

Найти вектора $y_1, \ldots, y_N \in \mathbb{R}^P$ так, чтобы

$$\sum_{i=1..N} \sum_{j=1..N} \|y_i - y_j\|^2 W_{ij} \to \min$$

при некоторых дополнительных условиях.

Начнем со случая $P = 1$. Это значит, что мы хотим сопоставить каждому вектору x_n такое число y_n, чтобы

$$F(y) := \sum_{i,j} |y_i - y_j|^2 W_{ij} \to \min, \qquad \text{где } y = (y_1, .., y_N)^T.$$

Две проблемы

1. Если $\alpha < 1$, то $F(\alpha y) < F(y)$. Для того, чтобы избежать такого поведения, накладываем условие $\|y\| = 1$.
2. Даже с ограничением $\|y\| = 1$ у задачи есть тривиальное решение $y_1 = y_2 = ... = y_N = 1/\sqrt{N}$. Полагаем $y \perp \mathbf{1}$.

Laplacian Eigenmaps для $P = 1$

Итак, при $P = 1$ формальная постановка задачи такова:

$$y = \underset{\|y\|=1, y \perp \mathbf{1}}{\arg\min} \sum_{i,j} |y_i - y_j|^2 W_{ij}$$

Введем обозначение. Пусть $d_i = \sum_{j=1..N} W_{ij}$ — степень вершины x_i в графе. Пусть $D = \mathrm{diag}(d_1, .., d_N)$.

Перепишем целевую функцию в другом виде

$$\sum_{i,j} |y_i - y_j|^2 W_{ij} = \sum_{i,j} \Big(y_i^2 - 2 y_i y_j + y_j^2 \Big) W_{ij}$$

$$= \sum_i y_i^2 \sum_j W_{ij} - 2 \sum_{i,j} y_i y_j W_{ij} + \sum_j y_j^2 \sum_i W_{ij}$$

$$= \sum_i y_i D_{ii} y_i - 2 \sum_{i,j} y_i W_{ij} y_j + \sum_j y_j D_{jj} y_j = 2 y^T (D - W) y.$$

Обозначим матрицу $D - W$ символом $\mathcal{L}$.

Laplacian Eigenmaps для $P = 1$

Получили, что

$$y = \underset{\|y\|=1, y\perp\mathbf{1}}{\arg\min} \sum_{i,j} |y_i - y_j|^2 W_{ij} = \underset{\|y\|=1, y\perp\mathbf{1}}{\arg\min}\ y^T \mathcal{L} y,$$

где $\mathcal{L} := D - W$ называется оператором Лапласа на графе.

Задача

$$y = \underset{\|y\|=1, y\perp\mathbf{1}}{\arg\min}\ y^T \mathcal{L} y,$$

является задачей оптимизации квадратичной формы $\mathcal{L}$.

Она стандартна: ее решением должен быть один из собственных векторов матрицы $\mathcal{L}$.

Laplacian Eigenmaps для $P = 1$

Оператор $\mathcal{L}$ — неотрицательно определенный.
Его ядро одномерно, благодаря связности графа.

Пусть

- $0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_N$ — собственные числа $\mathcal{L}$.
- $u_1, u_2, \ldots, u_N$ — соответствующие им собственные вектора.

Очевидно, что оптимизационную задачу без ограничения $y \perp \mathbf{1}$ решает вектор u_1. Конечно же, $u_1 = \frac{1}{\sqrt{N}}\mathbf{1}$.

Поэтому исходную задачу

$$y = \underset{\|y\|=1, y\perp\mathbf{1}}{\arg\min} \sum_{i,j} |y_i - y_j|^2 W_{ij} = \underset{\|y\|=1, y\perp\mathbf{1}}{\arg\min}\ y^T \mathcal{L} y,$$

решает вектор $y = u_2$, соответствующий самому маленькому <u>ненулевому</u> собственному числу λ_2 оператора $\mathcal{L}$.

Задача решена!

Laplacian Eigenmaps для $P = 1$: итог

Итог

Вектор y чисел $y_1, .., y_N \in \mathbb{R}$, который решает оптимизационную задачу

$$y = \underset{\|y\|=1, y \perp \mathbf{1}}{\arg\min} \sum_{i,j} |y_i - y_j|^2 W_{ij} = \underset{\|y\|=1, y \perp \mathbf{1}}{\arg\min}\, y^T \mathcal{L} y,$$

является собственным вектором матрицы $\mathcal{L} = D - W$, соответствующим ее наименьшему ненулевому собственному числу. Здесь $D = \mathrm{diag}(d_1, .., d_N)$, где d_n — степень вершины x_n.

Laplacian Eigenmaps для $P > 1$

Пусть теперь $P > 1$ и мы ищем $y_1, .., y_N \in \mathbb{R}^P$.
Пусть $Y = (y_1, .., y_N)^\top$ — матрица размера $N \times P$, строками которой являются вектора y_n.

Формальной постановкой в этом случае становится

$$Y = \underset{Y^TY=I_{P\times P}, Y^T\mathbf{1}=0}{\arg\min} \sum_{i,j} \|y_i - y_j\|^2 W_{ij},$$

где мы требуем

- ортонормированность всех P столбцов матрицы Y,
- чтобы эти столбцы были ортогональны вектору $\mathbf{1}$.

Это легко мотивировать соотношением

$$\sum_{i,j} \|y_i - y_j\|^2 W_{ij} = \sum_{p=1}^{P} \sum_{i,j} |y_{ip} - y_{jp}|^2 W_{ij}.$$

Видно, что столбцы Y хотят “слипнуться” — мы это запрещаем.

Laplacian Eigenmaps для $P > 1$

Вполне естественно, что решением задачи

$$Y = \underset{Y^TY=I_{P\times P}, Y^T\mathbf{1}=0}{\arg\min} \sum_{i,j} \|y_i - y_j\|^2 W_{ij},$$

будет матрица Y, столбцы которой — собственные вектора $\mathcal{L}$, соответствующие наименьшим P ненулевым собств. числам.

Итог

Матрица Y, строками которой являются $y_1, .., y_N \in \mathbb{R}^P$, которая решает оптимизационную задачу

$$Y = \underset{Y^TY=I_{P\times P}, Y^T\mathbf{1}=0}{\arg\min} \sum_{i,j} \|y_i - y_j\|^2 W_{ij},$$

состоит из P столбцов, являющихся собственными векторами матрицы $\mathcal{L} = D - W$, соответствующими наименьшим P ненулевым собственным числам.

Laplacian Eigenmaps: замечания

1. Мы предполагали, что граф W связный и неориентированный.
   - Если граф не связный, рассматривают компоненты связности по отдельности.
   - Если граф ориентированный - все плохо (не надо использовать такие графы).
2. Граф также может быть взвешенным. Веса должны быть велики для близких точек
   Часто полагают $W_{ij} = e^{-\|x_i - x_j\|^2}$.
3. В оригинальной статье используются условия $Y^T D Y = I_{P\times P}, Y^T D\mathbf{1} = 0$ вместо $Y^T Y = I_{P\times P}, Y^T\mathbf{1} = 0$.
4. Авторы показывают, что Locally-Linear-Embeddings тоже занимается поиском собственных векторов оператора $\mathcal{L}$.
5. Получился Kernel PCA с ядром $\mathcal{L}^\dagger$.

Почему $\mathcal{L}$ — оператор Лапласа

В $\mathbb{R}^2$, например, оператор Лапласа Δ действует на функции

$$\Delta f = -\frac{\partial^2 f}{\partial x_1^2} - \frac{\partial^2 f}{\partial x_2^2}.$$

Простейшим приближением второй производной является

$$\frac{\partial^2 f}{\partial x_1^2}(x_1, x_2) \approx \frac{f(x_1 - h, x_2) + f(x_1 + h, x_2) - 2f(x_1, x_2)}{h^2},$$

значит

$$\Delta \approx \frac{4f(x_1, x_2) - f(x_1 - h, x_2) - f(x_1 + h, x_2) - f(x_1, x_2 - h) - f(x_1, x_2 + h)}{h^2}.$$

Если рассмотреть f как функцию на квадратной сетке в $\mathbb{R}^2$ с шагом h, то матрицей Δ будет именно $\mathcal{L}$.

Еще $\mathcal{L}$ определяет диффузию на графе:

$$\frac{\mathrm{d}\phi}{\mathrm{d}t} + \mathcal{L}\phi = 0 \quad \text{аналогично} \quad \frac{\mathrm{d}\phi}{\mathrm{d}t} + \Delta\phi = 0 \ \text{ в } \ \mathbb{R}^S.$$

Мотивация Laplacian Eigenmaps и римановы многообразия

Пусть (M, g) — компактное риманово многообразие.

До этого мы искали функцию из графа в $\mathbb{R}^P$, которая соседям на графе сопоставляет близкие точки (переводящая x_n в y_n).

Попробуем теперь найти $f : M \to \mathbb{R}$ так, чтобы $|f(z) - f(x)|$ было мало, когда расстояние $d_M(z, x)$ мало.

Главным инструментом для нас будет неравенство

$$|f(z) - f(x)| \leq d_M(z, x)\|\nabla f(x)\| + o(d_M(z, x)),$$

являющееся простым следствием формулы Тейлора.

Оно говорит, что f разумно искать, решая оптимизационную задачу

$$f = \operatorname*{arg\,min}_{\|f\|_{L^2}=1, f \perp \mathbf{1}} \int_M \|\nabla f(x)\|^2 \, \mathrm{d}x$$

Мотивация Laplacian Eigenmaps и римановы многообразия

Итак, имеется оптимизационная задача

$$f = \underset{\|f\|_{L^2}=1, f\perp\mathbf{1}}{\arg\min} \int_M \|\nabla f(x)\|^2 \,\mathrm{d}x.$$

Формула ~~Стокса~~ интегрирования по частям дает

$$\int_M \|\nabla f(x)\|^2 \,\mathrm{d}x = \int_M f(x)(\Delta f)(x) \,\mathrm{d}x.$$

Наконец, задача

$$f = \underset{\|f\|_{L^2}=1, f\perp\mathbf{1}}{\arg\min} \int_M f(x)(\Delta f)(x) \,\mathrm{d}x$$

оптимизации квадр. формы положительно определенного оператора Δ решается как в конечномерном случае.

Решение — собств. функция оператора Лапласа–Бельтрами, соответствующая наименьшему ненулевому собств. числу.

Мотивация Laplacian Eigenmaps и римановы многообразия

Аналогично дискретному случаю, если бы мы искали $f : M \to \mathbb{R}^P$, решающую оптимизационную задачу

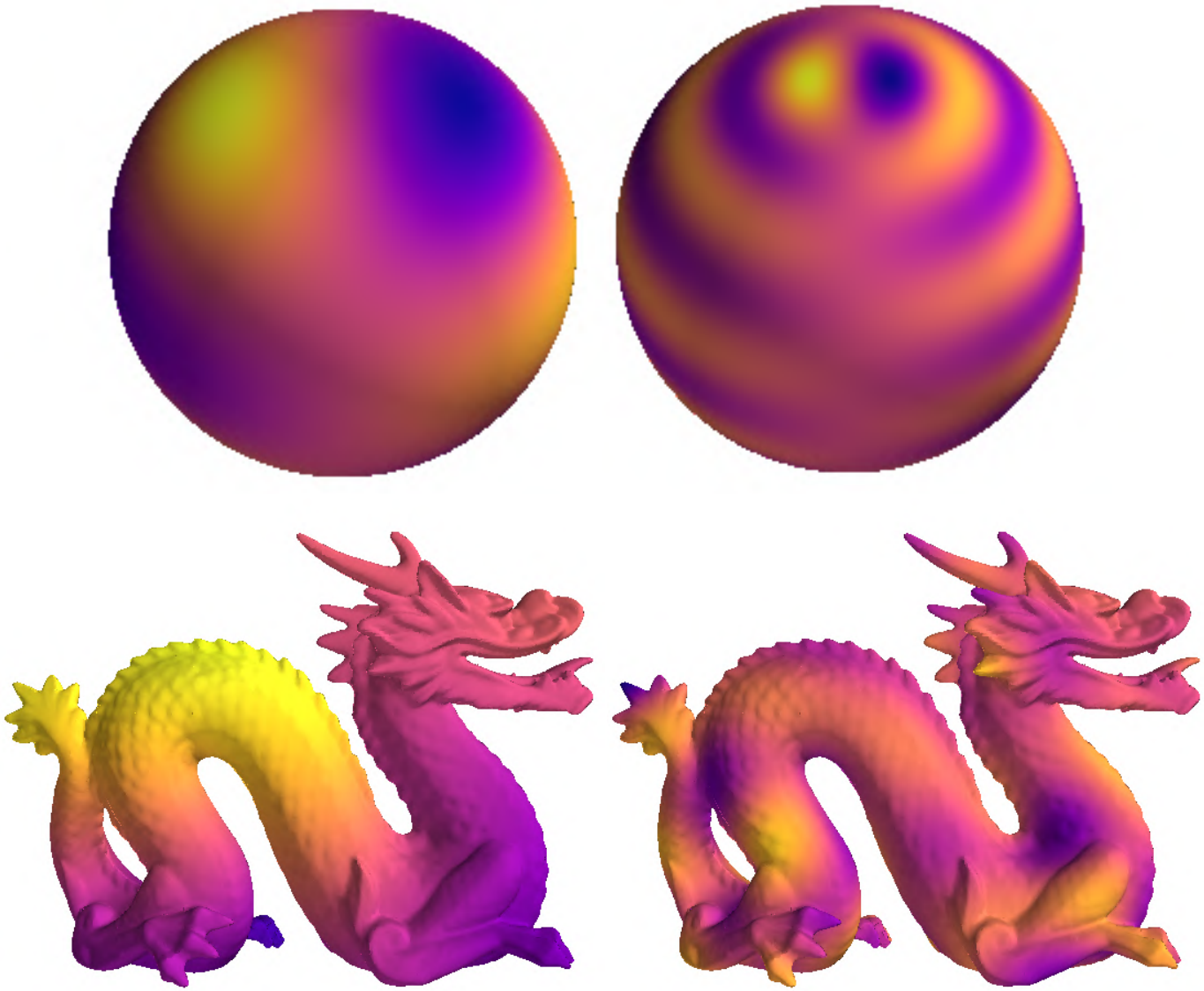
$$f = \underset{\|f_p\|_{L^2}=1 \text{ и } f_p \perp f_s}{\arg\min} \int_M \|\nabla f(x)\|^2 \, \mathrm{d}x,$$

то решением была бы $f = (f_1, .., f_P)$, где $\{f_p\}_{p=1}^P$ — собственные функции оператора Лапласа-Бельтрами, соответствующие P наименьшим ненулевым собственным числам.

Мотивация Laplacian Eigenmaps и римановы многообразия: замечания

1. Собственные функции Лапласа–Бельтрами — “самые линейные”. Это видно по целевой функции.
2. “Самую линейность” можно использовать, чтобы обобщить PCA на многообразия:
   - Обыкновенный PCA — Kernel PCA с ядром, порожденным линейными функциями.
   - На многообразии, по аналогии, можно использовать ядро, порожденное собственными функциями Лапласа.
3. Про евклидов случай.
   - Рассмотрим собственные функции Δ на R^S.
   - Как для компактных многообразий и графов, выкинем константу.
   - Тогда в ядре Δ останутся линейные многочлены.

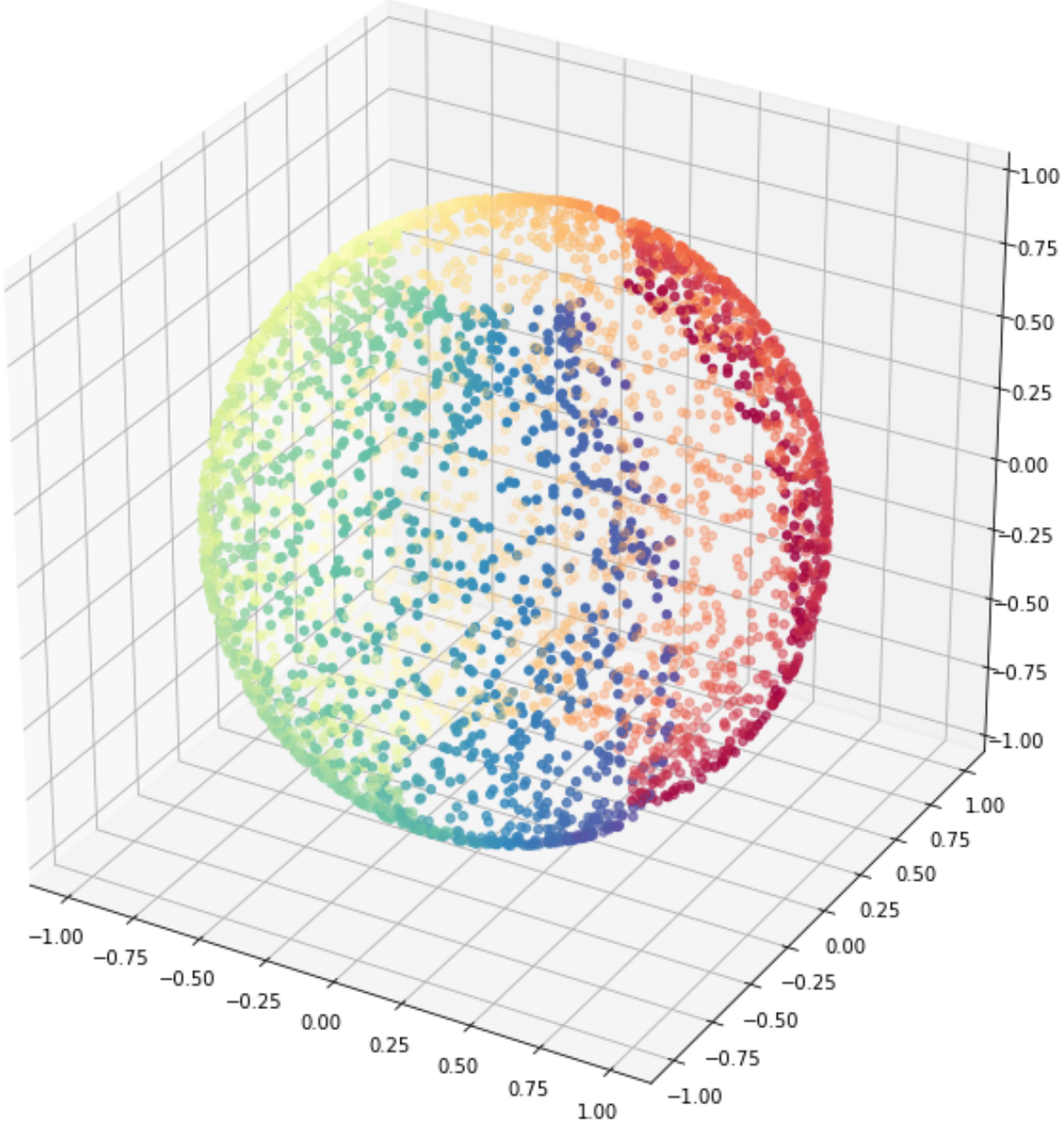
Мотивация Laplacian Eigenmaps и римановы многообразия: примеры собственных функций



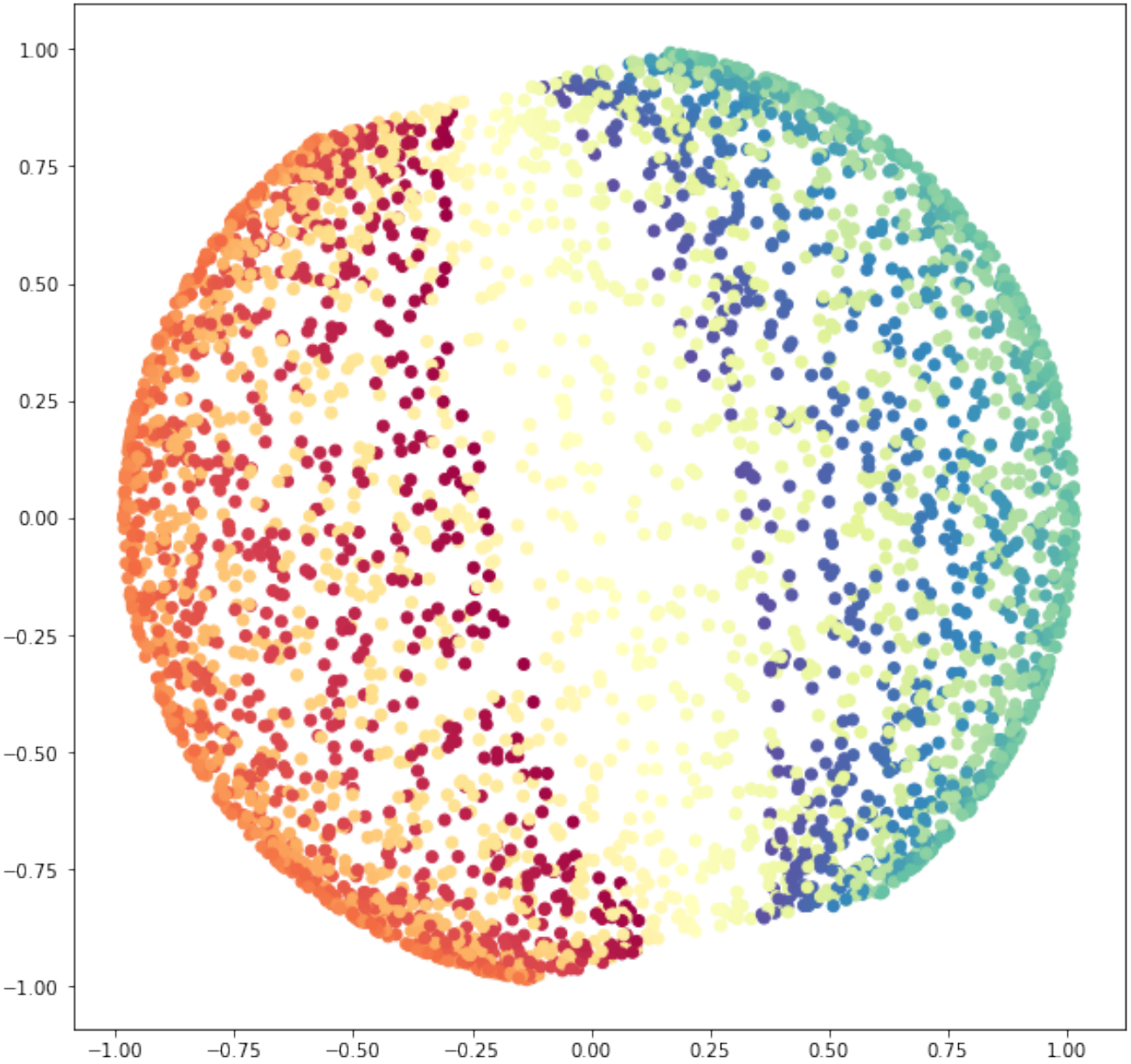
Пример

Заимствован из
https://juanitorduz.github.io/laplacian_eigenmaps_dim_red/

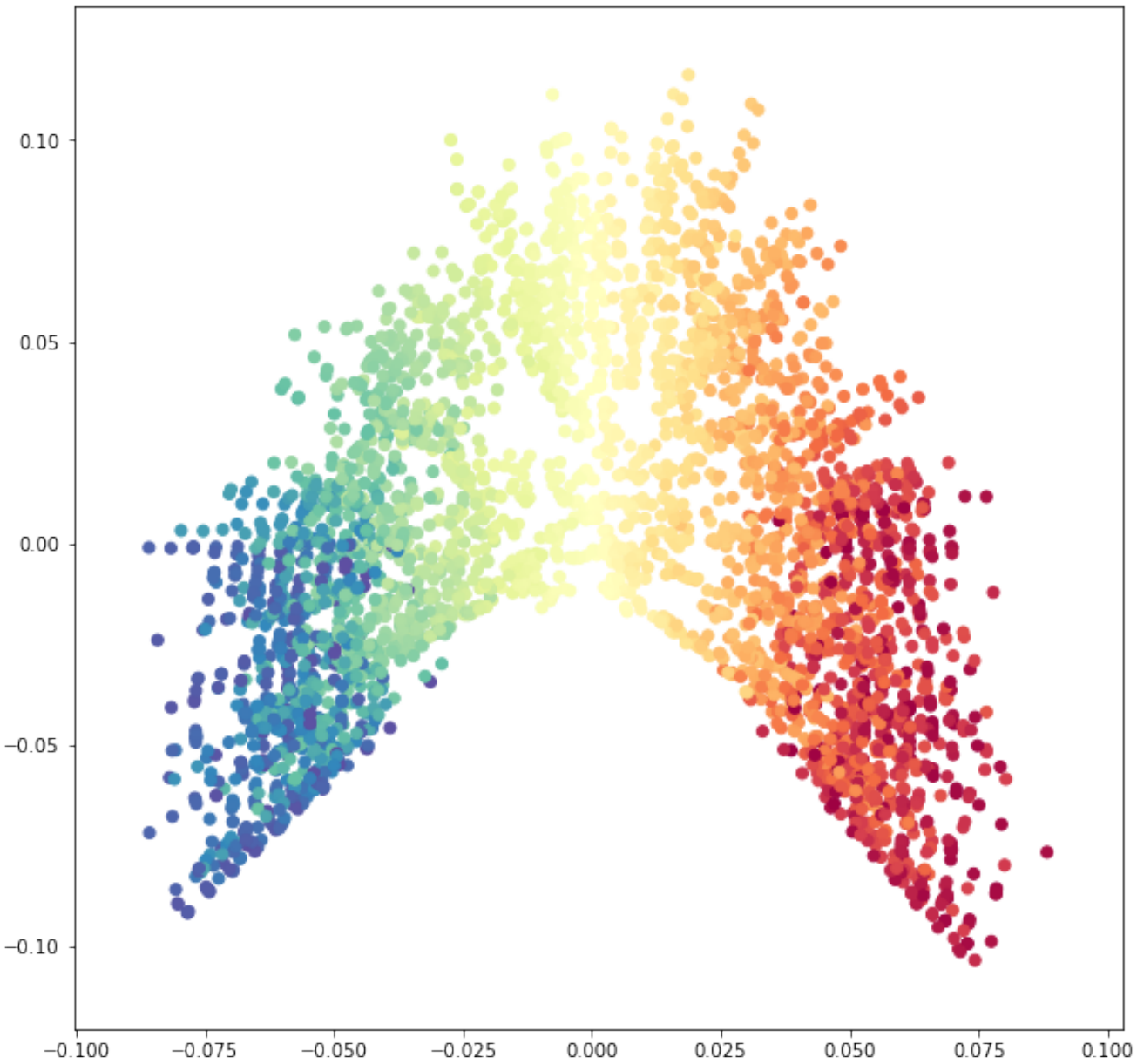
Пример: данные (облако точек в 3D)



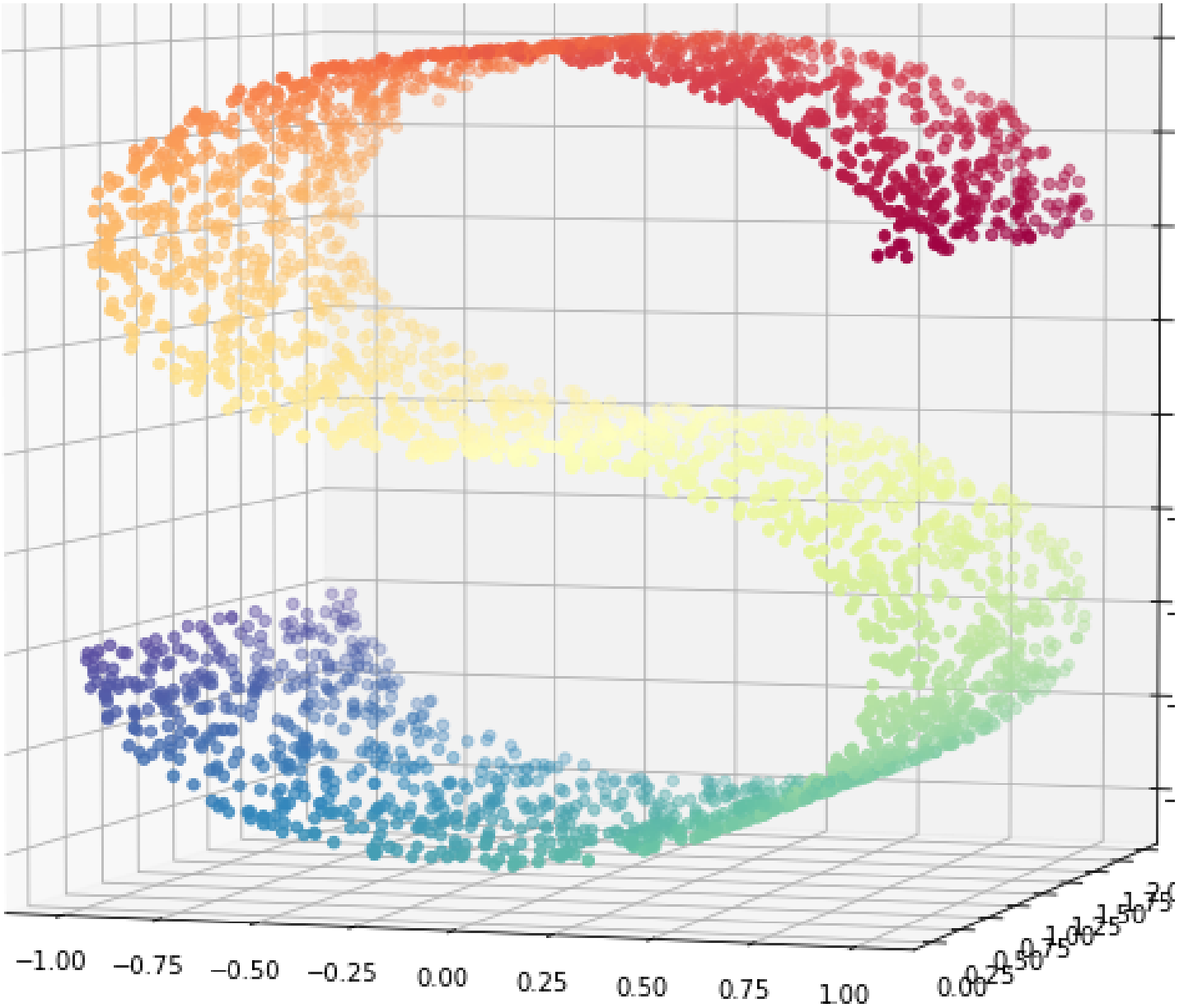
Пример: результат PCA



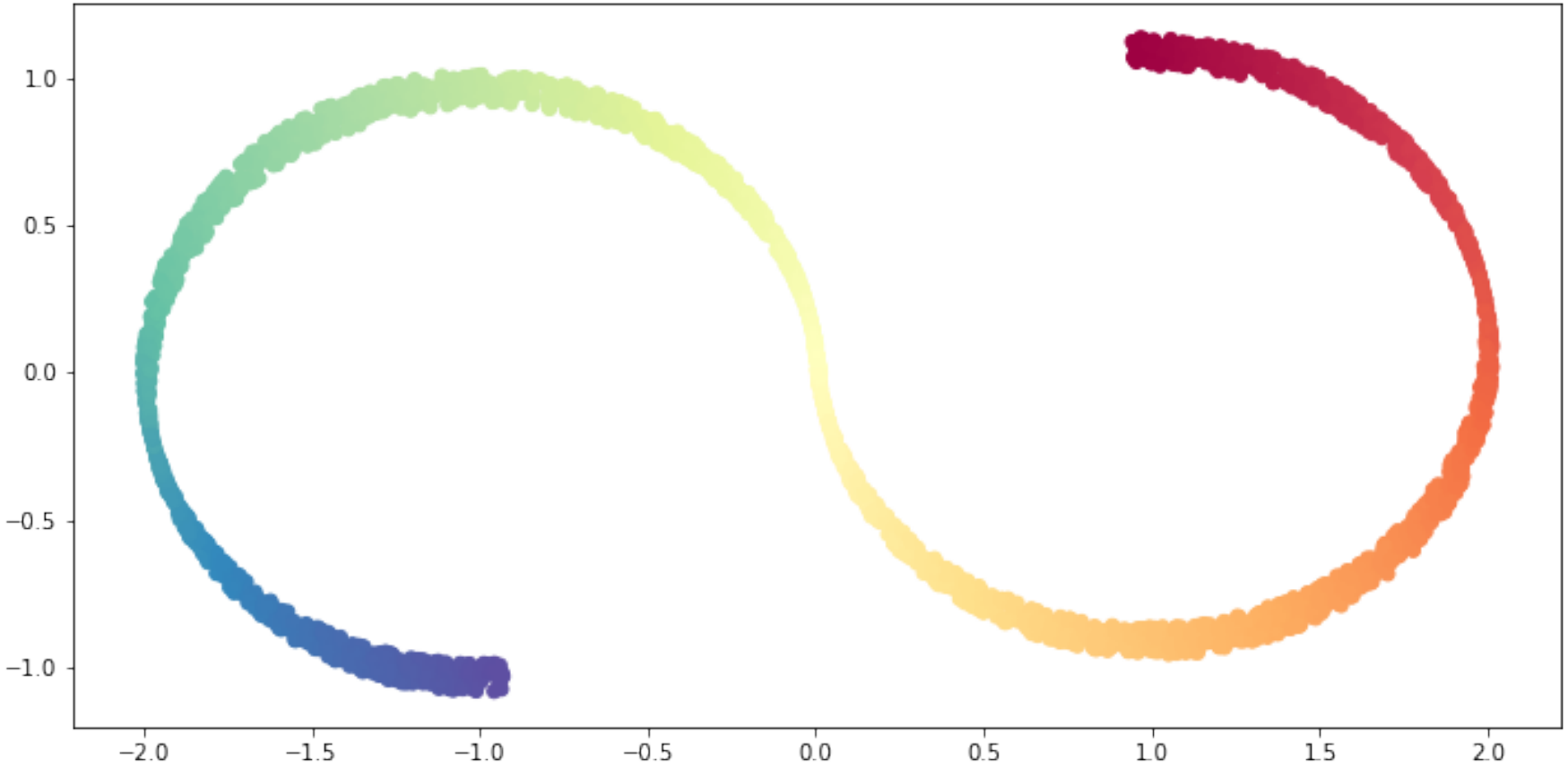
Пример: результат Laplacian Eigenmaps



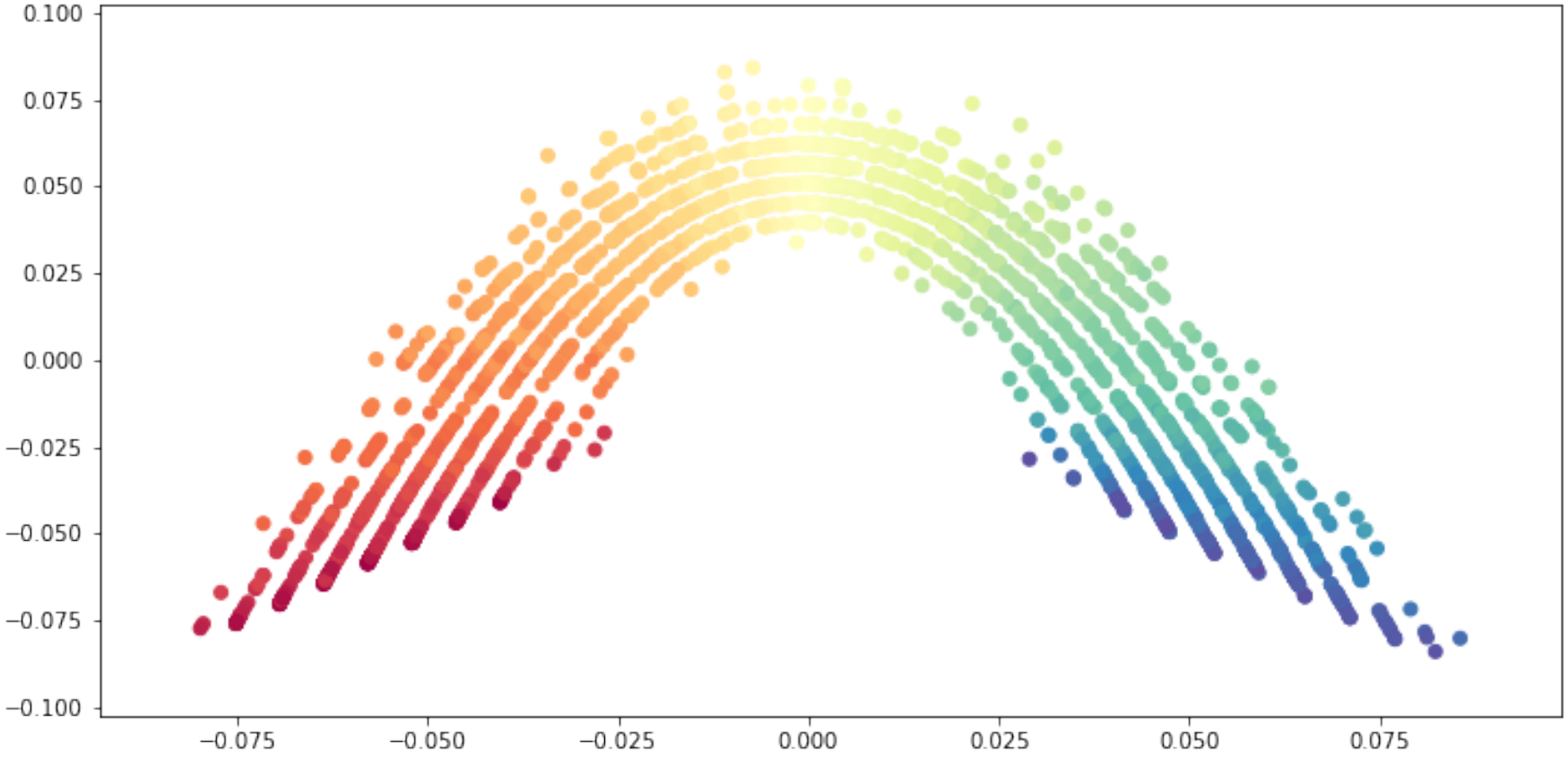
Второй пример: данные (облако точек в 3D)



Второй пример: результат PCA



Второй пример: результат Laplacian Eigenmaps



Основной источник

LETTER **Communicated by Joshua B. Tenenbaum**

Laplacian Eigenmaps for Dimensionality Reduction and Data Representation

Mikhail Belkin
misha@math.uchicago.edu
Department of Mathematics, University of Chicago, Chicago, IL 60637, U.S.A.

Partha Niyogi
niyogi@cs.uchicago.edu
Department of Computer Science and Statistics, University of Chicago, Chicago, IL 60637 U.S.A.

2002 год, Nature Communications. Рядом, но позже Isomap и Locally Linear Embeddings.