Алгоритм Laplacian Eigenmaps для точек вне обучающей выборки

Вельдяйкин Николай[1] Янович Юрий[1,2]

[1] Национальный исследовательский университет «Высшая школа экономики», Москва
[2] Институт проблем передачи информации им. А. А. Харкевича РАН, Москва

Аннотация Методы снижения размерности позволяют заменить многомерные описания данных на их низкоразмерные аналоги почти без потери информации, что способно упростить построение моделей по ним в рамках машинного обучения. Как правило, программные реализации алгоритмов снижения размерности строят лишь низкоразмерные описания для точек из обучающих выборок. Однако для последующего решения задач классификации и регрессии важно уметь строить вложение для новых точек вне обучающей выборки (out-of-sample extension), не перестраивая заново всю модель.
В статье записан алгоритм расширения алгоритма Laplacian Eigenmaps на точки вне обучающей выборки. Он реализован авторами в виде публичной ветки библиотеки scikit-learn. Для демонстрации качества работы кода проведен вычислительный эксперимент.

Ключевые слова: снижение размерности, расширение вне выборки

1 Введение

Понятие “большие данные” включает в себя не только большой объем данных, но и их высокую размерность, так как реальные данные обычно имеют очень высокую размерность (например, размерность цифровой черно-белой фотографии равна числу ее пикселей и может достигать сотен тысяч; изображения головного мозга, получаемые ежесекундно с помощью функциональной магнитно-резонансной томографии, имеют размерность порядка полутора миллионов). Однако многие традиционные методы и алгоритмы становятся неэффективными или просто неработоспособными для данных высокой размерности, и этот феномен назван проклятием размерности [1]. Известный статистик Д. Донохо сказал в 2000 году на конференции, посвященной математическим вызовам 21-го века: “мы можем с полной уверенностью сказать, что в наступающем веке анализ многомерных данных станет очень важным занятием, и совершенно новые методы многомерного анализа данных будут разработаны, просто мы еще не знаем, каковы они будут” [2].

Однако совокупность конкретных "реальных" данных, полученных из реальных источников, в силу наличия различных зависимостей между компонентами данных и ограничений на их возможные значения, занимает, как правило, малую часть высокоразмерного пространства наблюдений, имеющую невысокую внутреннюю размерность (например, множество всех черно-белых портретных изображений человеческих лиц с исходной размерностью порядка сотен тысяч, имеет внутреннюю размерность не выше ста). Следствием невысокой внутренней размерности является возможность построения низкоразмерной параметризации таких данных с минимальной потерей содержащейся в них информации. Поэтому многие алгоритмы для работы с высокоразмерными данными начинаются с решения задачи снижения размерности, результатом которого являются низкоразмерные описания таких данных.

На данный момент разработано множество алгоритмов снижения размерности: IsoMap [3], Locally-linear Embedding [4], Local Tangent Space Alignment [5], Laplacian Eigenmaps [6], Hessian Eigenmaps [7], Grassmann & Stiefel Eigenmaps [8], и др. Как правило, такие алгоритмы строят лишь низкоразмерные описания для точек из обучающих выборок. Однако для последующего решения задач классификации и регрессии важно уметь строить вложение для новых точек вне обучающей выборки, не перестраивая заново всю модель.

Статья посвящена реализации расширения алгоритма Laplacian Eigenmaps на точки вне обучающей выборки в рамках библиотеки scikit-learn на языке программирования Python.

2 Алгоритм Laplacian Eigenmaps

Задачу снижения размерности легко представить, как поиск отображения – действительно, нам необходимо для каждого набора точек из пространства высокой размерности получить набор точек в пространстве меньшей размерности. Идеальным случаем было бы найти универсальную биекцию, для которой существует обратная, но понятно, что задать отображение из исходного пространства размерности l в пространство размерности m, $\forall l, m : l > m$ – задача сложная. К тому же, осознавая цель данного отображения, а именно, упрощение анализа предоставленных данных для вычислительной машины, учитывая необходимость сохранения относительного расстояния между точками после отображения. То есть, если выбрать три любые точки и одну назвать выделенной, то ближняя к выделенной точка должна остаться ближней после отображения, а дальняя – дальней в смысле расстояния этих пространств. Указанную эвристику использует, в частности, алгоритм Laplacian Eigenmaps, описанный ниже.

Входными данными алгоритма являются:

– обучающая выборка $D = \{x_1, x_2, \ldots, x_n\} \subset \mathbb{R}^l$;

– функция близости $K(\cdot,\cdot)$: $\mathbb{R}^l \times \mathbb{R}^l \to [0,\infty)$. Например, $K(x,x') = 1$ для $|x-x'| < \varepsilon$ и 0 – иначе, где $\varepsilon > 0$ – параметр. Однако, функция близости может зависеть и от объектов обучающей выборки.

Выходом алгоритма являются низкоразмерные описания точек обучающей выборки:

– $y_1, \ldots, y_n \in \mathbb{R}^m$.

Algorithm 1. *Laplacian Eigenmaps [10]:*

1. *По выборке* $D = \{x_1, x_2, \ldots, x_n\} \subset \mathbb{R}^l$ *строим матрицу близости (affinity matrix)* $M = (M_{ij})_{i,j=1}^n$ *размера* $n \times n$: $M_{ij} = K(x_i, x_j)$.
2. *Определим диагональную матрицу* $Q : Q_{ii} = \sum_{j=1}^n M_{ij}$, *и вычисляем матрицу*

$$\tilde{M} = (\tilde{M}_{ij})_{i,j}^n = Q^{-0.5} \times M \times Q^{-0.5}.$$

3. *Находим* $m+1$ *наименьшее собственное значение*

$$\lambda_0 = 0 < \lambda_1 \leq \ldots \leq \lambda_m$$

матрицы $\tilde{M}$ *и соответствующие им собственные векторы*

$$V_0, \ldots, V_m \in \mathbb{R}^n.$$

4. *Используя полученную матрицу собственных векторов, получаем вложение:* $Y = (y_1|y_2|\ldots|y_n)^T = (V_1|\ldots|V_m)$, *где* $y_1, \ldots, y_n \in \mathbb{R}^m$, *где* $\cdot^T$ – *операция транспонирования.*

Этим алгоритмом решается задача минимизации

$$\sum_{i,j=1}^n K(x_i, x_j) \cdot |y_i - y_j|^2$$

при ограничениях $\sum_{i=1}^n y_i = 0$ и $\sum_{i=1}^n y_i^2 = 1$.

3 Расширение на точки вне обучающей выборки

В дополнение к данным со входа алгоритма Laplacian Eigenmaps, и данным, вычисленным в процессе его работы, для работы требуется новая точка $x \in \mathbb{R}^l$.

В качестве выходных данных выдаётся низкоразмерное описание $y \in \mathbb{R}^m$ для точки x.

Algorithm 2. *1. Вычисляем* $K(x, x_i)$, $i = 1, \ldots, n$. *Для* i: $x = x_i$ *присваиваем* $K(x, x_i) = \infty$.

2. *Определим*

$$\bar{K}(a,b) := \frac{1}{n} \frac{K(a,b)}{\sqrt{E_x\left(K(a,x)\right) E_{x'}\left(K(b,x')\right)}}.$$

Так как ни носитель выборки, ни мера неизвестны, то оцениваем математические ожидания простым усреднением по точкам обучающей выборки D.

3. *Вычисляем вектор* $\hat{M} = (\bar{K}(x,x_1),\ldots,\bar{K}(x,x_n))^T \in \mathbb{R}^n$.
4. *Вычисляем искомое* y *по формуле* $y = (V_1|\ldots|V_m)^T \cdot \hat{M}$.

Алгоритм расширения был разработан на основе статей [11] и [12], реализован как ветка популярной библиотеки scikit-learn и находится в открытом доступе [13].

4 Вычислительный эксперимент

Для демонстрации работы коды взята выборка точек из множества S-curve из библиотеки scikit-learn. Множество S-curve является подмножеством $\mathbb{R}^3$, поэтому $l = 3$. По построению, множество является двумерной поверхностью, поэтому естественно искать вложение для $m = 2$. Была сгенерирована i.i.d. выборка из $n = 2000$ точек. В качестве функции $K_D(\cdot,\cdot)$ использовались $k = 10$ ближайших соседей.

На Рисунке 1 представлена демонстрация корректности работы реализованного вложения: при расширении точки, близкие к обучающим, отображаются в близкие к обучающим (второй столбец). Более того, расширение примененное к обучающей выборке почти неразличимо с их вложением на этапе построения (третий столбец).

На Рисунке 2 представлена демонстрация неустойчивости модели Laplacian Eigenmaps к изменениям выборки. В качестве обучающей выборки использовались вся выборка (первый столбец), только точки с четными индексами – второй столбец и только точки с нечетными индексами – третий столбец. Можно заметить, что модель по полной выборке имеет отличный от других моделей масштаб, модели по по подвыборкам, хотя и имеют один масштаб, но значительно отличаются, в то время как реализованное расширение близко к исходной выборке. Черным квадратами на графике обозначены две точки с четными индексами, черными треугольниками – две точки с нечетными индексами. Можно заметить, что несмотря на похожесть второго и третьего столбцов, положения данных точек в разных столбцах сильно отличается. Следовательно, при использовании стандартного в машинном обучении разбиения на обучающую и тестовую подвыборки следует обучать модель снижения размерности на обучающей выборке, а для тестовой производить расширение. Остальные подходы (например, обучение на обучающей, а затем обучения на совокупности обучающей и тестовой) могут давать существенно различные результаты даже для близких точек обеих подвыборок.

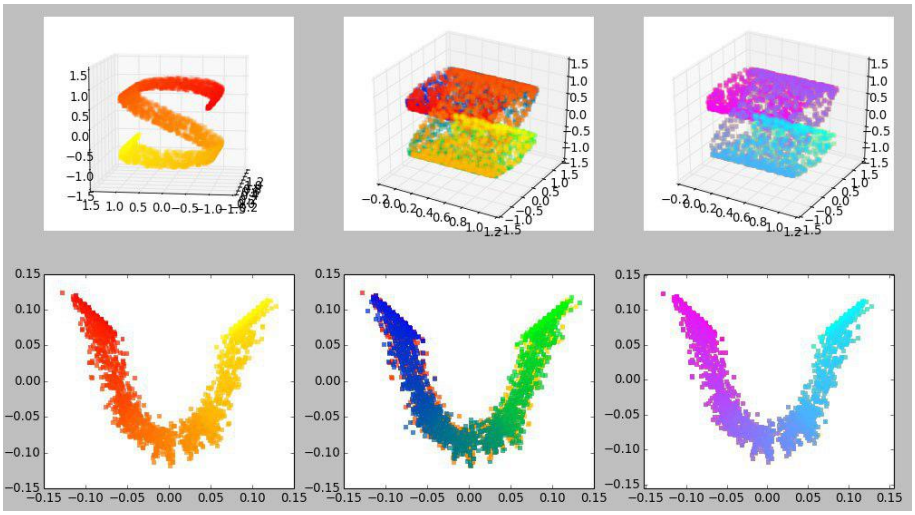

Рис. 1. Корректность реализованного вложения. Верхний левый рисунок – точки выборки с четными индексами в исходном пространстве (желто-красные точки). Под ним – результат снижения размерности до двух. Во втором столбце на первый наложена подвыборка с нечетными индексами (сине-зеленые точки), вложение для нее построено алгоритмом расширения. В третьем столбце – результат наложения на первый подвыборки с четными индексами (сине-фиолетовые точки). На верхнем графике они в точности совпали с исходными желто-красными точками, на втором – с точностью до неразличимых глазом невязок.

5 Заключение

В работе выписано расширение алгоритма Laplacian Eigenmaps для точек вне обучающей выборки. Оно реализовано программно в рамках популярной библиотеки scikit-learn и находится в открытом доступе в сети. С помощью данного расширения был произведен иллюстративный вычислительный эксперимент, в котором алгоритм демонстрирует себя устойчивым по входным точкам и, следовательно, может быть использован для последующего решения задач классификации и регрессии.

Благодарности

Работа была выполнена при поддержке гранта РФФИ 16-29-09649 офи_м.

Также, хочется выразить признательность студентам Горбачеву Сергею, Жижину Петру, Ребенко Ярославу и Хайдурову Руслану, участвовавшим вместе с Вельдяйкиным Николаем в проектном семинаре НИУ ВШЭ под руководством Яновича Юрия, в рамках которого была разработана реализация, за содержательные обсуждения, плодотворные дискуссии и совместную подготовку pull-request-а в scikit-learn.

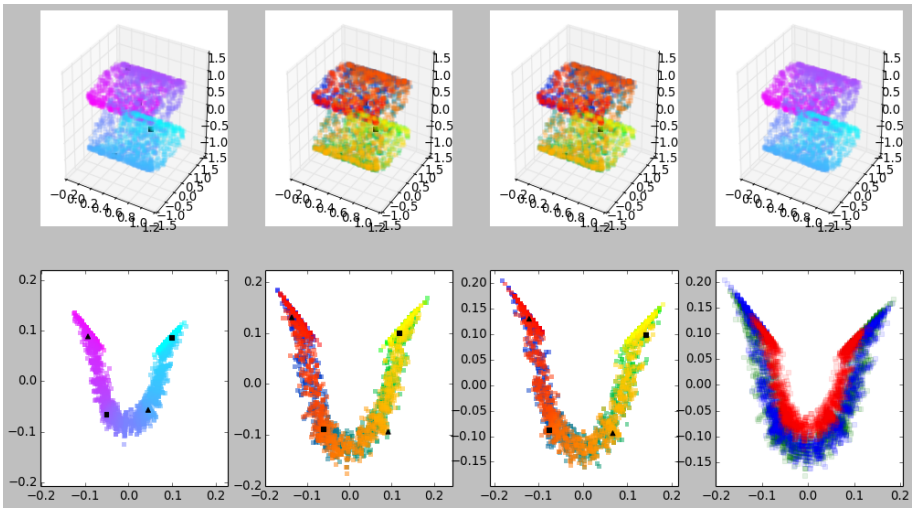

Рис. 2. Неустойчивость модели к изменению обучающей выборки. В верхнем ряду расположены графики для выборок в исходном трехмерном пространстве, в нижнем – результаты сжатия до размерности 2. В первом столбце объединенная выборка в исходном пространстве (сверху) и вложение для нее (снизу). Во втором обучение производилось на точках с четными индексами (сине-зеленые точки), а расширение на точках с нечетными индексами (желто-красные точки). В третьем столбце – обучение на точках с нечетными индексами (желто-красные), а расширение на точках с четными индексами (сине-зеленые точки). В четвертом – сверху полная выборка, а снизу все точки с первой картинки – красным цветом, все точки со второй картинки – зеленым цветом, все точки с третьей картинки – синим цветом

Список литературы

1. Bellman R. E. Dynamic programming // Princeton University Press, 1957.
2. Donoho D. L. High-dimensional data analysis: The curses and blessings of dimensionality // AMS conference on math challenges of 21st century. — 2000. — Pp. 1–31.
3. Tenenbaum J. B., de Silva V., Langford J. A Global Geometric Framework for Nonlinear Dimensionality Reduction // Science. — 2000. — Vol. 290, no. 5500. — Pp. 2319–2323.
4. Roweis S. T., Saul L. K. Nonlinear dimensionality reduction by locally linear embedding // Science. — 2000. — Vol. 290. — Pp. 2323–2326.
5. Zhang Z., Zha H. Principal Manifolds and Nonlinear Dimension Reduction via Local Tangent Space Alignment // SIAM Journal on Scientific Computing. — 2004. — Vol. 26, no. 1. — Pp. 313–338.
6. Belkin M., Niyogi P. Laplacian Eigenmaps for dimensionality reduction and data representation // Journal Neural Computation. — 2003. — Vol. 15, no. 6. — Pp. 1373–1396.

7. Donoho D. L., Grimes C. Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data // Proceedings of the National Academy of Sciences. — 2003. — may. — Vol. 100, no. 10. — Pp. 5591–5596.
8. Bernstein A., Kuleshov A. P. Manifold Learning: generalizing ability and tangent proximity // International Journal of Software and Informatics. — 2013. — Vol. 7, no. 3. — Pp. 359–390.
9. Scikit-learn library in Python [Online]. Available: http://scikit-learn.org
10. Ng A. Y., Jordan M. I., Weiss Y. On spectral clustering: Analysis and an algorithm //Advances in neural information processing systems. – 2002. – С. 849-856.
11. Von Luxburg, Ulrike: A tutorial on spectral clustering. // Statistics and computing, 17 (4), pp. 395–416 (2007)
12. Bengio Y. et al. Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering //Advances in neural information processing systems. – 2004. – С. 177-184.
13. Out-of-sample Laplacian Eigenmaps for scikit-learn [Online]. Available: https://github.com/hse-se-project/scikit-learn