Методы понижения размерности

Машинное обучение

Абдурахмон Садиев

ИСП РАН

20 марта 2025

Сингулярное разложение

Определение

Сингулярное разложение

Определение

Сингулярным разложением матрицы A размера $n \times d$ является произведение трех "простых" матриц:

$$A = USV^\top,$$

где

Сингулярное разложение

Определение

Сингулярным разложением матрицы A размера $n \times d$ является произведение трех "простых" матриц:

$$A = USV^\top,$$

где

- матрица U размера $n \times n$ ортогональна,

Сингулярное разложение

Определение

Сингулярным разложением матрицы A размера $n \times d$ является произведение трех "простых" матриц:

$$A = USV^\top,$$

где

- матрица U размера $n \times n$ ортогональна,
- матрица V размера $d \times d$ ортогональна,

Сингулярное разложение

Определение

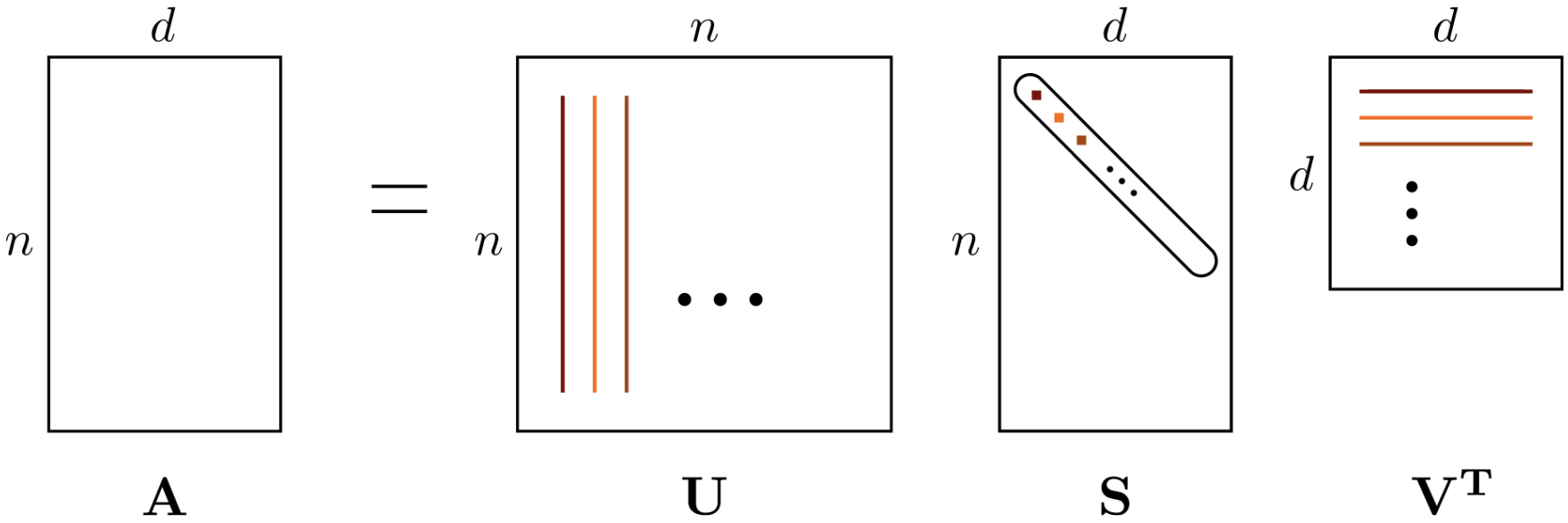
Сингулярным разложением матрицы A размера $n \times d$ является произведение трех "простых" матриц:

$$A = USV^\top,$$

где

- матрица U размера $n \times n$ ортогональна,
- матрица V размера $d \times d$ ортогональна,
- матрица S размера $n \times d$ такая, что $S_{ii} = \sigma_i = \sqrt{\lambda_i} \geq 0$, и $S_{ij} = 0$, если $i \neq j$ где $\{\lambda_i\}_{i=1}^k$ – собственные числа матрицы $A^\top A$ (и ненулевые собственные значения матрицы $AA^\top$).

Сингулярное разложение



Сингулярное разложение

Покажем, что сингулярное разложение существует.

Сингулярное разложение

Покажем, что сингулярное разложение существует.

Идея доказательства: случай $n \leq d$

Сингулярное разложение

Покажем, что сингулярное разложение существует.

Идея доказательства: случай $n \leq d$

- Рассмотрим $AA^\top$ - симметричная, неотрицательно определенная матрица.

Сингулярное разложение

Покажем, что сингулярное разложение существует.

Идея доказательства: случай $n \leq d$

- Рассмотрим $AA^\top$ - симметричная, неотрицательно определенная матрица.
- Существует ортогональная матрица U и диагональная матрица Λ: $AA^\top = U\Lambda U^\top$, причем $\Lambda = SS^\top$.

Сингулярное разложение

Покажем, что сингулярное разложение существует.

Идея доказательства: случай $n \leq d$

- Рассмотрим $AA^\top$ - симметричная, неотрицательно определенная матрица.
- Существует ортогональная матрица U и диагональная матрица Λ: $AA^\top = U\Lambda U^\top$, причем $\Lambda = SS^\top$.
- Заметив, что $A^\top u_i = \sigma_i v_i$, мы получаем матрицу $V^\top$.

Сингулярное разложение

Приложения:

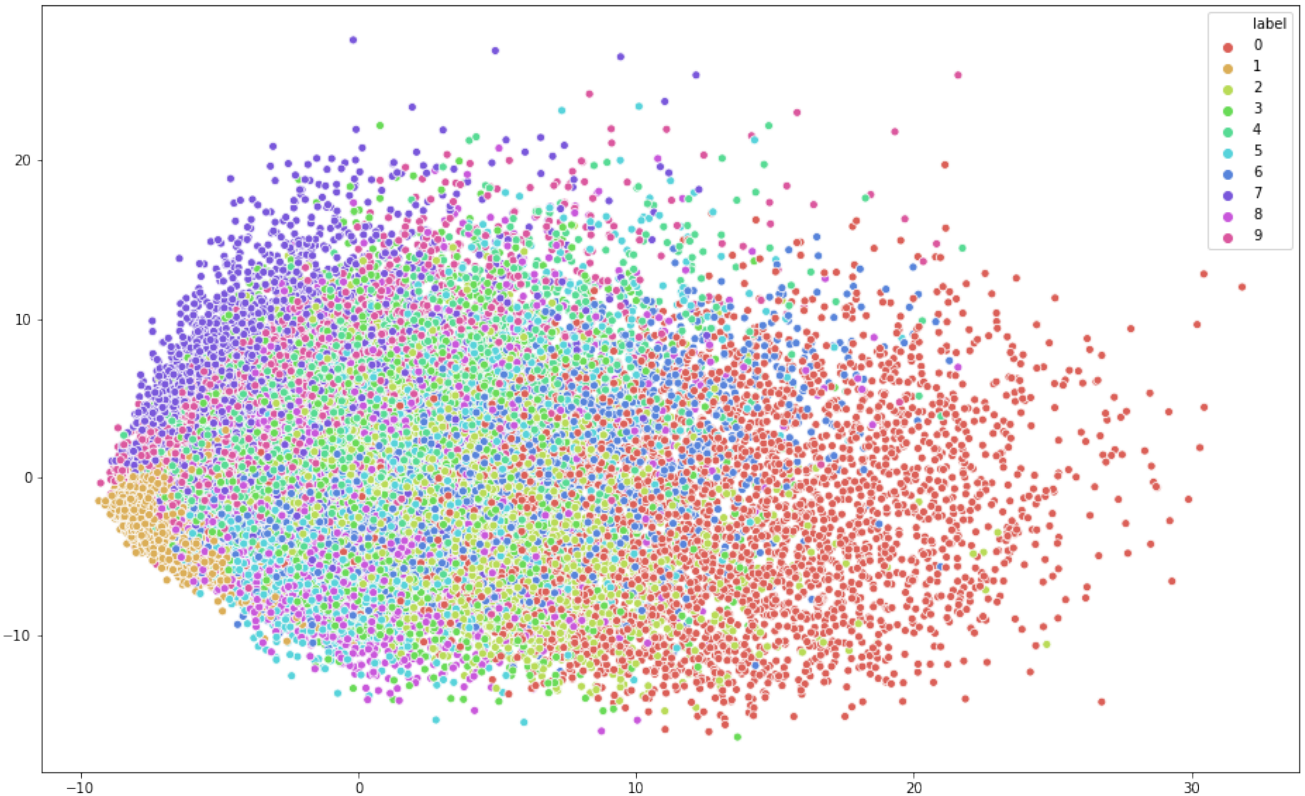
- Понижение размерности (PCA - метод главных компонент);
- Латентно-семантический анализ (LSA) в обработке естественного языка (NLP);
- Сжатие изображений;
- Подавление шума в обработке сигналов;
- Рекомендательные системы;
- Кластеризация и классификация данных;
- Решение обратных задач и вычисление псевдообратной матрицы;
- Квантовые вычисления и квантовая теория информации;
- Анализ экспрессии генов в биоинформатике;
- Распознавание лиц;

Мотивация

Задача: распознать цифры на картинке, для чего нужно выбрать компактное признаковое описание изображения.

Мотивация

После применения метода главных компонент количество признаков уменьшается, причем объекты одного класса образуют компактные области в пространстве признаков.



Постановка задачи

Постановка задачи

Задача *Снижение размерности данных*
Пусть $x_1, x_2, \ldots, x_n$ - выборка в пространстве $\mathbb{R}^d$.

Постановка задачи

Задача *Снижение размерности данных*
Пусть $x_1, x_2, \ldots, x_n$ - выборка в пространстве $\mathbb{R}^d$. Цель: выбрать пространство меньшей размерности $k < d$ так, чтобы схожие объекты оставались схожими и образовывали компактные области.

Постановка задачи

Задача *Снижение размерности данных*
Пусть $x_1, x_2, \ldots, x_n$ - выборка в пространстве $\mathbb{R}^d$. Цель: выбрать пространство меньшей размерности $k < d$ так, чтобы схожие объекты оставались схожими и образовывали компактные области. **Причины понижения размерности**

- уменьшение вычислительных затрат;
- сжатие данных для более эффективного хранения информации;
- борьба с мультиколлинеарностью;
- борьба с переобучением;
- визуализация и интерпретация данных;
- ...

Метод главных компонент (PCA)

Метод главных компонент (PCA)

Задача *Снижение размерности данных*

Пусть $x_1, x_2, \ldots, x_n$ - выборка в пространстве $\mathbb{R}^d$. Обозначим $X = (x_{ij})$ матрицу признаков размера $n \times d$.

Метод главных компонент (PCA)

Задача *Снижение размерности данных*

Пусть $x_1, x_2, \ldots, x_n$ - выборка в пространстве $\mathbb{R}^d$. Обозначим $X = (x_{ij})$ матрицу признаков размера $n \times d$.

Цель: Хотим перейти от признаков X к новым признакам $Z = (z_{ij})$, где Z -матрица размера $n \times k$, $k < d$.

Метод главных компонент (PCA)

Задача *Снижение размерности данных*

Пусть $x_1, x_2, \ldots, x_n$ - выборка в пространстве $\mathbb{R}^d$. Обозначим $X = (x_{ij})$ матрицу признаков размера $n \times d$.

Цель: Хотим перейти от признаков X к новым признакам $Z = (z_{ij})$, где Z -матрица размера $n \times k$, $k < d$. Помимо этого, мы хотим, чтобы старые признаки восстанавливались по новым с приемлемой точностью, т. е. существует матрица $V = (v_{ij})$ размера $d \times k$ такая, что

$$\hat{x}_j = \sum_{l=1}^{k} v_{jl} z_l \quad \approx \quad x_j.$$

Метод главных компонент (PCA)

Задача *Снижение размерности данных*

Пусть $x_1, x_2, \ldots, x_n$ - выборка в пространстве $\mathbb{R}^d$. Обозначим $X = (x_{ij})$ матрицу признаков размера $n \times d$.

Цель: Хотим перейти от признаков X к новым признакам $Z = (z_{ij})$, где Z -матрица размера $n \times k$, $k < d$. Помимо этого, мы хотим, чтобы старые признаки восстанавливались по новым с приемлемой точностью, т. е. существует матрица $V = (v_{ij})$ размера $d \times k$ такая, что

$$\hat{x}_j = \sum_{l=1}^{k} v_{jl} z_l \quad \approx \quad x_j.$$

Другими словами,

$$\ell(Z, V) = \sum_{j=1}^{k} \|\hat{x}_j - x_j\|^2 = \|X - ZV^\top\|_F^2 \to \min_{Z,V}$$

Свойства метода

Свойства метода

Теорема

Если $k < rank(X)$, то минимум $\ell(Z, V)$ достигается, когда столбцы матрицы V есть собственные вектора матрицы $X^\top X$, соответствующие k максимальным собственным значениям этой матрицы.

Свойства метода

Теорема

Если $k < rank(X)$, то минимум $\ell(Z, V)$ достигается, когда столбцы матрицы V есть собственные вектора матрицы $X^\top X$, соответствующие k максимальным собственным значениям этой матрицы.При этом $Z = XV$, а матрицы V и Z – ортогональны*.

Свойства метода

Теорема

Если $k < rank(X)$, то минимум $\ell(Z, V)$ достигается, когда столбцы матрицы V есть собственные вектора матрицы $X^\top X$, соответствующие k максимальным собственным значениям этой матрицы.При этом $Z = XV$, а матрицы V и Z – ортогональны*.

Свойства матриц V и Z:

Свойства метода

Теорема

Если $k < rank(X)$, то минимум $\ell(Z, V)$ достигается, когда столбцы матрицы V есть собственные вектора матрицы $X^\top X$, соответствующие k максимальным собственным значениям этой матрицы.При этом $Z = XV$, а матрицы V и Z – ортогональны*.

Свойства матриц V и Z:

1. Матрица V ортонормирована, т.е. $V^\top V = I$;

Свойства метода

Теорема

Если $k < rank(X)$, то минимум $\ell(Z, V)$ достигается, когда столбцы матрицы V есть собственные вектора матрицы $X^\top X$, соответствующие k максимальным собственным значениям этой матрицы.При этом $Z = XV$, а матрицы V и Z – ортогональны*.

Свойства матриц V и Z:

1. Матрица V ортонормирована, т.е. $V^\top V = I$;
2. $Z^\top Z = \Lambda = \text{diag}(\lambda_1, \ldots, \lambda_k)$,где $\lambda_1 \geq \cdots \geq \lambda_k$ –k максимальных собственных значений матрицы $X^\top X$.

Свойства метода

Теорема

Если $k < rank(X)$, то минимум $\ell(Z, V)$ достигается, когда столбцы матрицы V есть собственные вектора матрицы $X^\top X$, соответствующие k максимальным собственным значениям этой матрицы.При этом $Z = XV$, а матрицы V и Z – ортогональны*.

Свойства матриц V и Z:

1. Матрица V ортонормирована, т.е. $V^\top V = I$;
2. $Z^\top Z = \Lambda = \text{diag}(\lambda_1, \ldots, \lambda_k)$,где $\lambda_1 \geq \cdots \geq \lambda_k$ –k максимальных собственных значений матрицы $X^\top X$.
3. $X^\top XV = V\Lambda$, $X^\top Z = Z\Lambda$;

Свойства метода

Теорема

Если $k < rank(X)$, то минимум $\ell(Z, V)$ достигается, когда столбцы матрицы V есть собственные вектора матрицы $X^\top X$, соответствующие k максимальным собственным значениям этой матрицы.При этом $Z = XV$, а матрицы V и Z – ортогональны*.

Свойства матриц V и Z:

1. Матрица V ортонормирована, т.е. $V^\top V = I$;
2. $Z^\top Z = \Lambda = \text{diag}(\lambda_1, \ldots, \lambda_k)$,где $\lambda_1 \geq \cdots \geq \lambda_k$ –k максимальных собственных значений матрицы $X^\top X$.
3. $X^\top XV = V\Lambda,\ X^\top Z = Z\Lambda$;
4. $\|ZV^\top - X\|_F^2 = \|X\| - \text{tr}(\Lambda) = \sum_{j=k+1}^{d} \lambda_j.$

Выбор количества признаков

Выбор количества признаков

Наблюдение

Пусть

$$E_k = \frac{\sum_{j=k+1}^{d} \lambda_j}{\sum_{j=1}^{d} \lambda_j}.$$

Выбор количества признаков

Наблюдение

Пусть

$$E_k = \frac{\sum_{j=k+1}^{d} \lambda_j}{\sum_{j=1}^{d} \lambda_j}.$$

Чем меньше E_k, тем лучше новые признаки приближают старые.

Выбор количества признаков

Наблюдение

Пусть

$$E_k = \frac{\sum_{j=k+1}^{d} \lambda_j}{\sum_{j=1}^{d} \lambda_j}.$$

Чем меньше E_k, тем лучше новые признаки приближают старые.

- $\widetilde{k} = \min_k E_k < \varepsilon$ - эффективная размерность пространства признаков X.

Выбор количества признаков

Наблюдение

Пусть

$$E_k = \frac{\sum_{j=k+1}^{d} \lambda_j}{\sum_{j=1}^{d} \lambda_j}.$$

Чем меньше E_k, тем лучше новые признаки приближают старые.

- $\widetilde{k} = \min_k E_k < \varepsilon$ - эффективная размерность пространства признаков X.
- **метода крутого склона**: если E_{k+1} достаточно мало и $E_k >> E_{k+1}$, то в качестве эффективной размерности берем k.

Выбор количества признаков

Наблюдение

Пусть

$$E_k = \frac{\sum_{j=k+1}^{d} \lambda_j}{\sum_{j=1}^{d} \lambda_j}.$$

Чем меньше E_k, тем лучше новые признаки приближают старые.

- $\widetilde{k} = \min_k E_k < \varepsilon$ - эффективная размерность пространства признаков X.
- **метода крутого склона**: если E_{k+1} достаточно мало и $E_k >> E_{k+1}$, то в качестве эффективной размерности берем k.
- **метод сломанной трости**: $\bar{k} = \inf\{k : \frac{\lambda_k}{\sum_{i=1}^{k} \lambda_i} < \frac{1}{d} \sum_{j=k}^{d} \frac{1}{j}\}$

Главные компоненты

Главные компоненты

Главные компоненты – это собственные векторы матрицы $X^\top X$, соответствующие k максимальным собственным значениям этой матрицы.

Главные компоненты

Главные компоненты – это собственные векторы матрицы $X^\top X$, соответствующие k максимальным собственным значениям этой матрицы.
В качестве новых признаков $\{z_j\}_{j=1}^k$ модели мы выбираем проекции старых объектов на эти собственные векторы.

Главные компоненты

Главные компоненты – это собственные векторы матрицы $X^\top X$, соответствующие k максимальным собственным значениям этой матрицы.
В качестве новых признаков $\{z_j\}_{j=1}^k$ модели мы выбираем проекции старых объектов на эти собственные векторы.

Вероятностная интерпретация: Проекции объектов на первую главную компоненту c_1 имеют наибольшую выборочную дисперсию среди проекций на всевозможные направления $d \in \mathbb{R}^k$.

Главные компоненты

Главные компоненты – это собственные векторы матрицы $X^\top X$, соответствующие k максимальным собственным значениям этой матрицы.
В качестве новых признаков $\{z_j\}_{j=1}^k$ модели мы выбираем проекции старых объектов на эти собственные векторы.

Вероятностная интерпретация: Проекции объектов на первую главную компоненту c_1 имеют наибольшую выборочную дисперсию среди проекций на всевозможные направления $d \in \mathbb{R}^k$. Далее, $\forall j \geq 2$ проекции объектов на c_j – j-тую главную компоненту – имеют наибольшую выборочную дисперсию среди проекций на всевозможные направления $d \in \mathbb{R}^k$, перпендикулярные $c_1, \ldots, c_{j1}$.

Особенности

Особенности

- Выбросы могут сильно помешать работе алгоритма (потому что метод линейный), поэтому стоит их удалить.

Особенности

- Выбросы могут сильно помешать работе алгоритма (потому что метод линейный), поэтому стоит их удалить.
- Если 2 признака имеют очень большую корреляцию, то один из признаков тоже стоит удалить, иначе матрицы будут плохо обращаться.

Особенности

- Выбросы могут сильно помешать работе алгоритма (потому что метод линейный), поэтому стоит их удалить.
- Если 2 признака имеют очень большую корреляцию, то один из признаков тоже стоит удалить, иначе матрицы будут плохо обращаться.
- Если признаки – в различных шкалах, то стандартизация данных обязательна!

Особенности

- Выбросы могут сильно помешать работе алгоритма (потому что метод линейный), поэтому стоит их удалить.
- Если 2 признака имеют очень большую корреляцию, то один из признаков тоже стоит удалить, иначе матрицы будут плохо обращаться.
- Если признаки – в различных шкалах, то стандартизация данных обязательна!
- PCA способен находить только линейные подпространства исходного пространства.

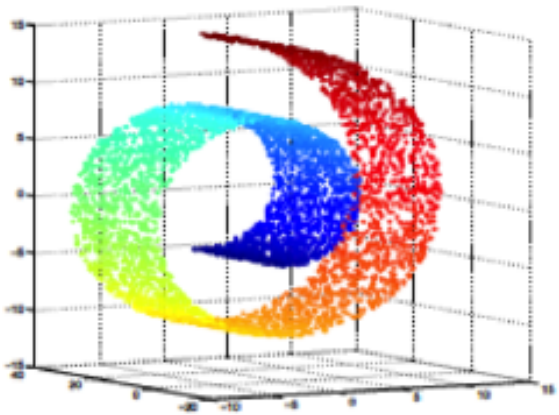
Особенности

- Выбросы могут сильно помешать работе алгоритма (потому что метод линейный), поэтому стоит их удалить.
- Если 2 признака имеют очень большую корреляцию, то один из признаков тоже стоит удалить, иначе матрицы будут плохо обращаться.
- Если признаки – в различных шкалах, то стандартизация данных обязательна!
- PCA способен находить только линейные подпространства исходного пространства.
- PCA является инвариантным относительно поворота координат в пространстве переменных.

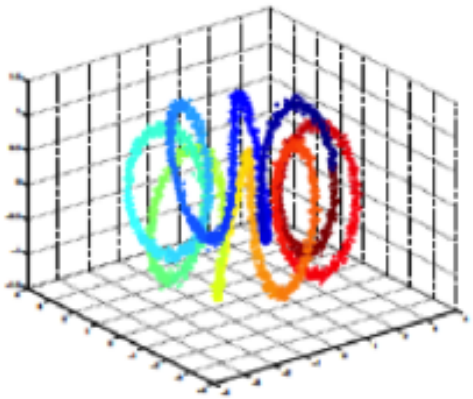
Особенности

- Выбросы могут сильно помешать работе алгоритма (потому что метод линейный), поэтому стоит их удалить.
- Если 2 признака имеют очень большую корреляцию, то один из признаков тоже стоит удалить, иначе матрицы будут плохо обращаться.
- Если признаки – в различных шкалах, то стандартизация данных обязательна!
- PCA способен находить только линейные подпространства исходного пространства.
- PCA является инвариантным относительно поворота координат в пространстве переменных.
- Если некоторые собственные значения матрицы $X^\top X$, совпали, то новое пространство признаков может определяться неоднозначно.

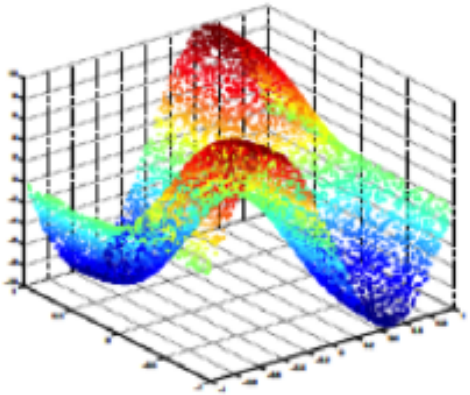
Нелинейные методы понижения размерности



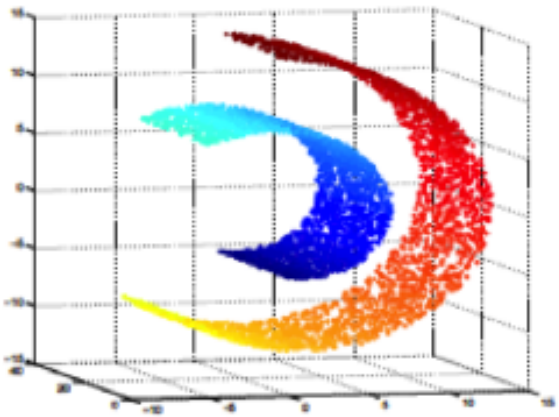
(a) Swiss roll dataset.



(b) Helix dataset.



(c) Twinpeaks dataset.



(d) Broken Swiss roll dataset.

Нелинейные методы понижения размерности

Нелинейные методы понижения размерности

- **Kernel PCA**: вместо стандартного скалярного произведения рассмотрим скалярное произведение $K(x, y) = \langle \Phi(x), \Phi(y) \rangle$.

Нелинейные методы понижения размерности

- **Kernel PCA**: вместо стандартного скалярного произведения рассмотрим скалярное произведение $K(x, y) = \langle \Phi(x), \Phi(y) \rangle$.
- **Многомерное шкалирование (multi-dimensional scaling)**: минимизировать суммарное расхождение между расстояниями между объектами, $d_{ij} = \rho(x_i, x_j)$, и расстояниями между объектами в новом пространстве признаков,$\delta_{ij} = \rho(z_i, z_j)$.

Нелинейные методы понижения размерности

- **Kernel PCA**: вместо стандартного скалярного произведения рассмотрим скалярное произведение $K(x, y) = \langle \Phi(x), \Phi(y) \rangle$.
- **Многомерное шкалирование (multi-dimensional scaling)**: минимизировать суммарное расхождение между расстояниями между объектами, $d_{ij} = \rho(x_i, x_j)$, и расстояниями между объектами в новом пространстве признаков,$\delta_{ij} = \rho(z_i, z_j)$.
- **Isomap**: минимизируется суммарное расхождение между расстояниями между объектами в пространстве новых признаков и расстояниями по поверхности (по геодезическим) в исходном пространстве.

t-SNE

t-SNE (t-distributed Stochastic Neighbor Embedding) – один из лучших алгоритмов понижения размерности, существующих на настоящий момент, в частности, хорошо подходящий для визуализации данных.

Постановка задачи

Задача *Снижение размерности данных*

Постановка задачи

Задача *Снижение размерности данных*
Пусть $x_1, x_2, \ldots, x_n$ - выборка в пространстве $\mathbb{R}^d$. Хотим перейти от признаков X к новым признакам $Z = (z_{ij})$, где Z -матрица размера $n \times k,\ k < d$.

Постановка задачи

Задача *Снижение размерности данных*
Пусть $x_1, x_2, \ldots, x_n$ - выборка в пространстве $\mathbb{R}^d$. Хотим перейти от признаков X к новым признакам $Z = (z_{ij})$, где Z -матрица размера $n \times k,\ k < d$.
Цель отобразить кластерную структуру, сохранив кластеры без сохранения пространственных взаимоотношений кластеров.

SNE

SNE

Определим вероятность того, что x_j - сосед x_i, пропорционально плотности $\mathcal{N}(x_i, \sigma_i^2)$ в точке x_j:

SNE

Определим вероятность того, что x_j - сосед x_i, пропорционально плотности $\mathcal{N}(x_i, \sigma_i^2)$ в точке x_j:

$$p_{j|i} = \frac{\exp\left(-\|x_j - x_i\|^2 / 2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-\|x_j - x_k\|^2 / 2\sigma_i^2\right)}.$$

SNE

Определим вероятность того, что x_j - сосед x_i, пропорционально плотности $\mathcal{N}(x_i, \sigma_i^2)$ в точке x_j:

$$p_{j|i} = \frac{\exp\left(-\|x_j - x_i\|^2/2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-\|x_j - x_k\|^2/2\sigma_i^2\right)}.$$

Определим вероятность того, что z_j - сосед z_i, пропорционально плотности $\mathcal{N}(z_i, 1/2)$ в точке z_j:

SNE

Определим вероятность того, что x_j - сосед x_i, пропорционально плотности $\mathcal{N}(x_i, \sigma_i^2)$ в точке x_j:

$$p_{j|i} = \frac{\exp\left(-\|x_j - x_i\|^2/2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-\|x_j - x_k\|^2/2\sigma_i^2\right)}.$$

Определим вероятность того, что z_j - сосед z_i, пропорционально плотности $\mathcal{N}(z_i, 1/2)$ в точке z_j:

$$q_{j|i} = \frac{\exp\left(-\|z_j - z_i\|^2\right)}{\sum_{k \neq i} \exp\left(-\|z_j - z_k\|^2\right)}.$$

SNE

Определим вероятность того, что x_j - сосед x_i, пропорционально плотности $\mathcal{N}(x_i, \sigma_i^2)$ в точке x_j:

$$p_{j|i} = \frac{\exp\left(-\|x_j - x_i\|^2/2\sigma_i^2\right)}{\sum_{k\neq i}\exp\left(-\|x_j - x_k\|^2/2\sigma_i^2\right)}.$$

Определим вероятность того, что z_j - сосед z_i, пропорционально плотности $\mathcal{N}(z_i, 1/2)$ в точке z_j:

$$q_{j|i} = \frac{\exp\left(-\|z_j - z_i\|^2\right)}{\sum_{k\neq i}\exp\left(-\|z_j - z_k\|^2\right)}.$$

Будем считать, что $p_{i|i} = 0$ и $q_{i|i} = 0$.

Как выбирать σ_i^2

Как выбирать σ_i^2

Рассмотрим $P_i = (p_{1|i}, p_{2|i}, \ldots, p_{n|i})$. Посчитаем Энтропию

Как выбирать σ_i^2

Рассмотрим $P_i = (p_{1|i}, p_{2|i}, \ldots, p_{n|i})$. Посчитаем Энтропию

$$H(\sigma_i) = -\sum_{j=1}^{n} p_{j|i} \log p_{j|i}.$$

Как выбирать σ_i^2

Рассмотрим $P_i = (p_{1|i}, p_{2|i}, \ldots, p_{n|i})$. Посчитаем Энтропию

$$H(\sigma_i) = -\sum_{j=1}^{n} p_{j|i} \log p_{j|i}.$$

Перплексия $\text{Perp}(\sigma_i) = 2^{H(\sigma_i)}$ имеет смысл сглаженного показателя эффективного числа соседей точки X_i.

Как выбирать σ_i^2

Рассмотрим $P_i = (p_{1|i}, p_{2|i}, \ldots, p_{n|i})$. Посчитаем Энтропию

$$H(\sigma_i) = -\sum_{j=1}^{n} p_{j|i} \log p_{j|i}.$$

Перплексия $\text{Perp}(\sigma_i) = 2^{H(\sigma_i)}$ имеет смысл сглаженного показателя эффективного числа соседей точки X_i.
Значение перплексии — гиперпараметр метода. Задается одинаковым для всех i. Числа σ_i подбираются на основе перплексии с помощью бинарного поиска.

Как выбирать σ_i^2

Рассмотрим $P_i = (p_{1|i}, p_{2|i}, \ldots, p_{n|i})$. Посчитаем Энтропию

$$H(\sigma_i) = -\sum_{j=1}^{n} p_{j|i} \log p_{j|i}.$$

Перплексия $\text{Perp}(\sigma_i) = 2^{H(\sigma_i)}$ имеет смысл сглаженного показателя эффективного числа соседей точки X_i.
Значение перплексии — гиперпараметр метода. Задается одинаковым для всех i. Числа σ_i подбираются на основе перплексии с помощью бинарного поиска.
Разная σ_i необходима из-за возможного наличия кластеров разных плотностей.

Оптимизационная задача

Оптимизационная задача

Хотим выбрать $z_1, z_2, \ldots, z_n$ так, чтобы вероятности $q_{j|i}$ как можно точнее описывали $p_{j|i}$.

Оптимизационная задача

Хотим выбрать $z_1, z_2, \ldots, z_n$ так, чтобы вероятности $q_{j|i}$ как можно точнее описывали $p_{j|i}$.

Функция потерь: дивергенция Кульбака-Лейблера между $p_{j|i}$ и$q_{j|i}$

Оптимизационная задача

Хотим выбрать $z_1, z_2, \ldots, z_n$ так, чтобы вероятности $q_{j|i}$ как можно точнее описывали $p_{j|i}$.

Функция потерь: дивергенция Кульбака-Лейблера между $p_{j|i}$ и$q_{j|i}$

$$\mathcal{L} = \sum_{i=1}^{n} \mathrm{KL}(P_i, Q_i) = \sum_{i=1}^{n} \sum_{j=1}^{n} p_{j|i} \log \frac{p_{j|i}}{q_{j|i}} \to \min_{Z}$$

Оптимизационная задача

Хотим выбрать $z_1, z_2, \ldots, z_n$ так, чтобы вероятности $q_{j|i}$ как можно точнее описывали $p_{j|i}$.

Функция потерь: дивергенция Кульбака-Лейблера между $p_{j|i}$ и$q_{j|i}$

$$\mathcal{L} = \sum_{i=1}^{n} \mathrm{KL}(P_i, Q_i) = \sum_{i=1}^{n} \sum_{j=1}^{n} p_{j|i} \log \frac{p_{j|i}}{q_{j|i}} \to \min_{Z}$$

Несимметричность:

- большой штраф, если близкие точки будут расположены далеко.
- малый штраф, если далекие точки будут расположены близко.

Симметричный SNE

Симметричный SNE

Рассмотрим "симметричные" вероятности:

$$p_{ji} = \frac{\exp\left(-\|x_j - x_i\|^2/2\sigma_i^2\right)}{\sum_{j\neq i}\exp\left(-\|x_j - x_i\|^2/2\sigma_i^2\right)}, \quad q_{ji} = \frac{\exp\left(-\|z_j - z_i\|^2\right)}{\sum_{j\neq i}\exp\left(-\|z_j - z_i\|^2\right)}$$

Симметричный SNE

Рассмотрим "симметричные" вероятности:

$$p_{ji} = \frac{\exp\left(-\|x_j - x_i\|^2/2\sigma_i^2\right)}{\sum_{j\neq i}\exp\left(-\|x_j - x_i\|^2/2\sigma_i^2\right)}, \quad q_{ji} = \frac{\exp\left(-\|z_j - z_i\|^2\right)}{\sum_{j\neq i}\exp\left(-\|z_j - z_i\|^2\right)}$$

причем $p_{ii} = q_{ii} = 0$, функция потерь

Симметричный SNE

Рассмотрим "симметричные" вероятности:

$$p_{ji} = \frac{\exp\left(-\|x_j - x_i\|^2/2\sigma_i^2\right)}{\sum_{j\neq i}\exp\left(-\|x_j - x_i\|^2/2\sigma_i^2\right)}, \quad q_{ji} = \frac{\exp\left(-\|z_j - z_i\|^2\right)}{\sum_{j\neq i}\exp\left(-\|z_j - z_i\|^2\right)}$$

причем $p_{ii} = q_{ii} = 0$, функция потерь

$$\mathcal{L} = \mathrm{KL}(P, Q) = \sum_{i\neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}.$$

Симметричный SNE

Рассмотрим "симметричные" вероятности:

$$p_{ji} = \frac{\exp\left(-\|x_j - x_i\|^2/2\sigma_i^2\right)}{\sum_{j\neq i}\exp\left(-\|x_j - x_i\|^2/2\sigma_i^2\right)}, \quad q_{ji} = \frac{\exp\left(-\|z_j - z_i\|^2\right)}{\sum_{j\neq i}\exp\left(-\|z_j - z_i\|^2\right)}$$

причем $p_{ii} = q_{ii} = 0$, функция потерь

$$\mathcal{L} = \mathrm{KL}(P, Q) = \sum_{i\neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}.$$

Если x_i - выброс, то $\|x_i - x_j\| >> 0$ и $p_{ij} \approx 0$ для любого j. Значит расположение z_i почти не влияет на $\mathcal{L}$.

Симметричный SNE

Рассмотрим "симметричные" вероятности:

$$p_{ji} = \frac{\exp\left(-\|x_j - x_i\|^2/2\sigma_i^2\right)}{\sum_{j\neq i}\exp\left(-\|x_j - x_i\|^2/2\sigma_i^2\right)}, \quad q_{ji} = \frac{\exp\left(-\|z_j - z_i\|^2\right)}{\sum_{j\neq i}\exp\left(-\|z_j - z_i\|^2\right)}$$

причем $p_{ii} = q_{ii} = 0$, функция потерь

$$\mathcal{L} = \mathrm{KL}(P, Q) = \sum_{i\neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}.$$

Если x_i - выброс, то $\|x_i - x_j\| >> 0$ и $p_{ij} \approx 0$ для любого j. Значит расположение z_i почти не влияет на $\mathcal{L}$.
Вместо этого определим p_{ij} как $p_{ij} = \frac{p_{i|j}+p_{j|i}}{2n}$. Тогда $\sum_j p_{ij} > 1/2n$.

Crowding problem

Crowding problem

При вложении в пространство малой размерности при использовании нормального распределения действуют силы сжатия, из-за чего точки сильно сжимаются в кучу.

Crowding problem

При вложении в пространство малой размерности при использовании нормального распределения действуют силы сжатия, из-за чего точки сильно сжимаются в кучу.

Пример

Данные: облака точек в вершинах правильного тетраэдра в $\mathbb{R}^3$. В силу симметричности при сжатии в $\mathbb{R}^2$ на точки будут действовать "сжимающие силы" по диагоналям.

t-SNE

t-SNE

Для решения Crowding problem определим q_{ij} на основе **распределения Стьюдента**:

t-SNE

Для решения Crowding problem определим q_{ij} на основе **распределения Стьюдента**:

$$q_{ji} = \frac{\left(1 + \|z_i - z_j\|^2\right)^{-1}}{\sum_{i \neq j}(1 + \|z_i - z_j\|^2)^{-1}}$$

t-SNE

Для решения Crowding problem определим q_{ij} на основе **распределения Стьюдента**:

$$q_{ji} = \frac{\left(1 + \|z_i - z_j\|^2\right)^{-1}}{\sum_{i \neq j}(1 + \|z_i - z_j\|^2)^{-1}}$$

При этом $p_{ij} = \frac{p_{i|j} + p_{j|i}}{2n}$. Градиент в таком случае

$$\frac{\partial \mathcal{L}}{\partial z_i} = 4 \sum_j^n \frac{(p_{ij} - q_{ij})(z_i - z_j)}{1 + \|z_i - z_j\|^2}$$

t-SNE

Для решения Crowding problem определим q_{ij} на основе **распределения Стьюдента**:

$$q_{ji} = \frac{\left(1 + \|z_i - z_j\|^2\right)^{-1}}{\sum_{i \neq j}(1 + \|z_i - z_j\|^2)^{-1}}$$

При этом $p_{ij} = \frac{p_{i|j} + p_{j|i}}{2n}$. Градиент в таком случае

$$\frac{\partial \mathcal{L}}{\partial z_i} = 4 \sum_j^n \frac{(p_{ij} - q_{ij})(z_i - z_j)}{1 + \|z_i - z_j\|^2}$$

- Градиент делится на квадрат расстояния плюс 1.

t-SNE

Для решения Crowding problem определим q_{ij} на основе **распределения Стьюдента**:

$$q_{ji} = \frac{\left(1 + \|z_i - z_j\|^2\right)^{-1}}{\sum_{i \neq j}(1 + \|z_i - z_j\|^2)^{-1}}$$

При этом $p_{ij} = \frac{p_{i|j} + p_{j|i}}{2n}$. Градиент в таком случае

$$\frac{\partial \mathcal{L}}{\partial z_i} = 4 \sum_j^n \frac{(p_{ij} - q_{ij})(z_i - z_j)}{1 + \|z_i - z_j\|^2}$$

- Градиент делится на квадрат расстояния плюс 1.
- Если точки близки, то $\|z_i - z_j\|^2 \approx 0$, и сила остается прежней.

t-SNE

Для решения Crowding problem определим q_{ij} на основе **распределения Стьюдента**:

$$q_{ji} = \frac{\left(1 + \|z_i - z_j\|^2\right)^{-1}}{\sum_{i \neq j}(1 + \|z_i - z_j\|^2)^{-1}}$$

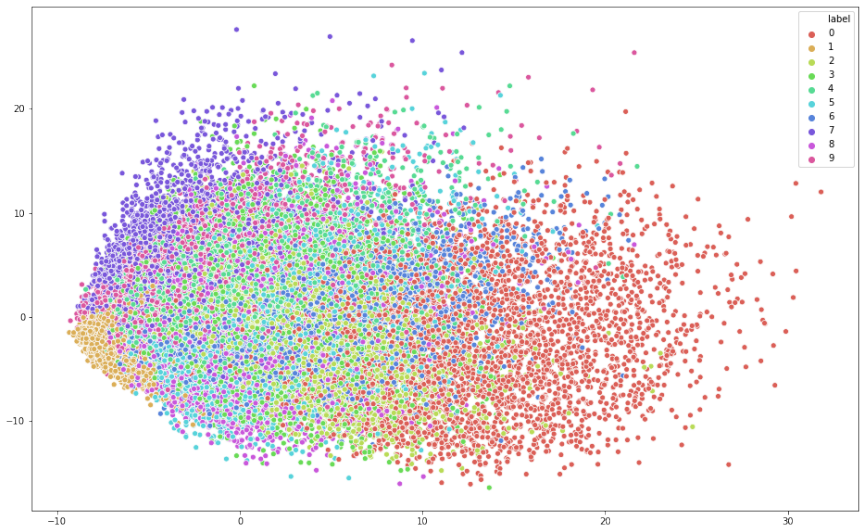
При этом $p_{ij} = \frac{p_{i|j} + p_{j|i}}{2n}$. Градиент в таком случае

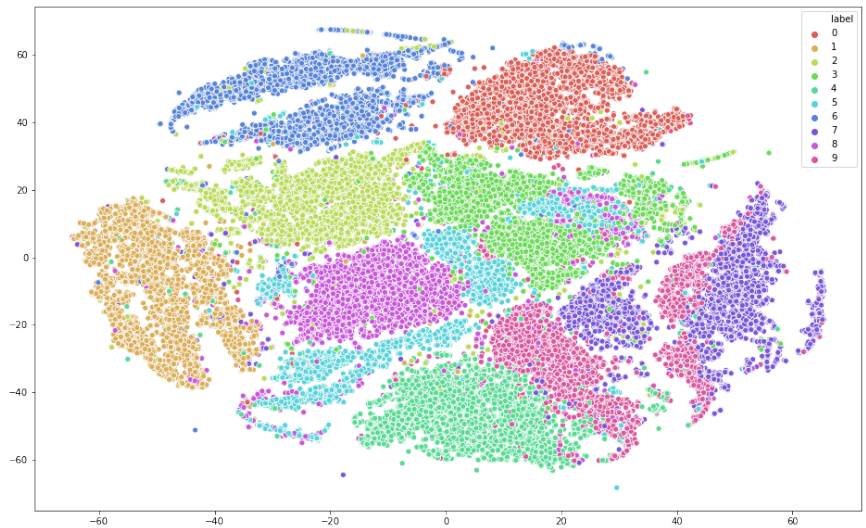
$$\frac{\partial \mathcal{L}}{\partial z_i} = 4 \sum_j^n \frac{(p_{ij} - q_{ij})(z_i - z_j)}{1 + \|z_i - z_j\|^2}$$

- Градиент делится на квадрат расстояния плюс 1.
- Если точки близки, то $\|z_i - z_j\|^2 \approx 0$, и сила остается прежней.
- Если точки далеки, то $\|z_i - z_j\|^2 >> 0$, сила сжатия становится существенно меньше, и сильного сжатия не происходит.

t-SNE

MNIST


label
0
1
2
3
4
5
6
7
8
9
20
10
0
−10
−10
0
10
20
30


label
0
1
2
3
4
5
6
7
8
9
60
40
20
0
−20
−40
−60
−60
−40
−20
0
20
40
60

UMAP

Uniform Manifold Approximation and Projection — метод, выполняющий нелинейное снижение размерности. Алгоритм предложен в 2018 году с целью получить аналог t-SNE, но с более сильным математическим обоснованием.

Ориентированный граф

Ориентированный граф

Пусть $X = (x_1, x_2, \ldots, x_n)$ - выборка в пространстве $\mathcal{X}$.

Ориентированный граф

Пусть $X = (x_1, x_2, \ldots, x_n)$ - выборка в пространстве $\mathcal{X}$.
$\rho(x, y)$ - метрика в $\mathcal{X}$.

Ориентированный граф

Пусть $X = (x_1, x_2, \ldots, x_n)$ - выборка в пространстве $\mathcal{X}$.
$\rho(x, y)$ - метрика в $\mathcal{X}$.
Определим случайный ориентированный граф на множестве вершин X. Считаем, что каждое ребро появляются независимо от других.

Ориентированный граф

Пусть $X = (x_1, x_2, \ldots, x_n)$ - выборка в пространстве $\mathcal{X}$.
$\rho(x, y)$ - метрика в $\mathcal{X}$.
Определим случайный ориентированный граф на множестве вершин X. Считаем, что каждое ребро появляются независимо от других.
Пусть $x_{i_1}, x_{i_2}, \ldots, x_{i_k}$ - k ближайших соседей объекта x_i в выборке X .
$r_i = \min_s \rho(x_i, x_{i_s})$ — расстояние до ближайшего соседа.

Ориентированный граф

Пусть $X = (x_1, x_2, \ldots, x_n)$ - выборка в пространстве $\mathcal{X}$.
$\rho(x, y)$ - метрика в $\mathcal{X}$.
Определим случайный ориентированный граф на множестве вершин X. Считаем, что каждое ребро появляются независимо от других.
Пусть $x_{i_1}, x_{i_2}, \ldots, x_{i_k}$ - k ближайших соседей объекта x_i в выборке X .
$r_i = \min_s \rho(x_i, x_{i_s})$ — расстояние до ближайшего соседа.
Вероятность ребра из x_i в x_{i_s} (для остальных ноль):

Ориентированный граф

Пусть $X = (x_1, x_2, \ldots, x_n)$ - выборка в пространстве $\mathcal{X}$.
$\rho(x, y)$ - метрика в $\mathcal{X}$.
Определим случайный ориентированный граф на множестве вершин X. Считаем, что каждое ребро появляются независимо от других.
Пусть $x_{i_1}, x_{i_2}, \ldots, x_{i_k}$ - k ближайших соседей объекта x_i в выборке X .
$r_i = \min_s \rho(x_i, x_{i_s})$ — расстояние до ближайшего соседа.
Вероятность ребра из x_i в x_{i_s} (для остальных ноль):

$$P\{x_i \to x_{i_s}\} = \exp\left(-\frac{\rho(x_i, x_{i_s}) - r_i}{\sigma_i}\right),$$

Ориентированный граф

Пусть $X = (x_1, x_2, \ldots, x_n)$ - выборка в пространстве $\mathcal{X}$.
$\rho(x, y)$ - метрика в $\mathcal{X}$.
Определим случайный ориентированный граф на множестве вершин X. Считаем, что каждое ребро появляются независимо от других.
Пусть $x_{i_1}, x_{i_2}, \ldots, x_{i_k}$ - k ближайших соседей объекта x_i в выборке X .
$r_i = \min_s \rho(x_i, x_{i_s})$ — расстояние до ближайшего соседа.
Вероятность ребра из x_i в x_{i_s} (для остальных ноль):

$$P\{x_i \to x_{i_s}\} = \exp\left(-\frac{\rho(x_i, x_{i_s}) - r_i}{\sigma_i}\right),$$

где σ_i определяется как решение уравнения $\sum_{s=1}^{k} P\{x_i \to x_{i_s}\} = \log_2 k$

Неориентированный граф

Неориентированный граф

На основе ориентированного графа построим неориентированный. X — множество вершин. Вероятность ребра между X_i и X_j :

Неориентированный граф

На основе ориентированного графа построим неориентированный. X — множество вершин. Вероятность ребра между X_i и X_j :

$P\left\{x_i \leftrightarrow x_{i_s}\right\}$

Неориентированный граф

На основе ориентированного графа построим неориентированный. X — множество вершин. Вероятность ребра между X_i и X_j :

$$P\{x_i \leftrightarrow x_{i_s}\} = P\left\{\{x_i \to x_{i_s}\} \bigcup \{x_{i_s} \to x_i\}\right\}$$

Неориентированный граф

На основе ориентированного графа построим неориентированный. X — множество вершин. Вероятность ребра между X_i и X_j :

$$\begin{aligned} P\{x_i \leftrightarrow x_{i_s}\} &= P\left\{\{x_i \to x_{i_s}\} \bigcup \{x_{i_s} \to x_i\}\right\} \\ &= P\{x_i \to x_{i_s}\} + P\{x_{i_s} \to x_i\} - P\{x_i \to x_{i_s}\} P\{x_{i_s} \to x_i\} \end{aligned}$$

Новые признаки

Новые признаки

Пусть $z_1, z_2, \ldots, z_n$ - новые признаки, которые хотим получить.

Новые признаки

Пусть $z_1, z_2, \ldots, z_n$ - новые признаки, которые хотим получить. На них определяем случаем неориентированный граф с вероятностями $P\{z_i \leftrightarrow z_j\} = (1 + a\|z_i - z_j\|_2^{2b})$, где a и b - гиперпараметры.

Новые признаки

Пусть $z_1, z_2, \ldots, z_n$ - новые признаки, которые хотим получить. На них определяем случаем неориентированный граф с вероятностями $P\{z_i \leftrightarrow z_j\} = (1 + a\|z_i - z_j\|_2^{2b})$, где a и b - гиперпараметры.

Минимизируем дивергенцию Кульбака-Лейблера:

Новые признаки

Пусть $z_1, z_2, \ldots, z_n$ - новые признаки, которые хотим получить. На них определяем случаем неориентированный граф с вероятностями $P\{z_i \leftrightarrow z_j\} = (1 + a\|z_i - z_j\|_2^{2b})$, где a и b - гиперпараметры.

Минимизируем дивергенцию Кульбака-Лейблера:

$$\begin{aligned} \mathrm{KL}(P_X, P_Y) &= \sum_{i,j} \Big[P\{x_i \leftrightarrow x_j\} \log \frac{P\{x_i \leftrightarrow x_j\}}{P\{z_i \leftrightarrow z_j\}} \\ &\quad + (1 - P\{x_i \leftrightarrow x_j\}) \log \frac{1 - P\{x_i \leftrightarrow x_j\}}{1 - P\{z_i \leftrightarrow z_j\}} \Big] \end{aligned}$$

Особенности

Особенности

- Введение графов и вероятностей на них эквивалентно использованию метрики локальной связности, которая устойчива к проклятию размерности. UMAP можно применять на данных огромных размерностей.

Особенности

- Введение графов и вероятностей на них эквивалентно использованию метрики локальной связности, которая устойчива к проклятию размерности. UMAP можно применять на данных огромных размерностей.
- Возможно сохранение пространственных взаимоотношений между кластерами.

Особенности

- Введение графов и вероятностей на них эквивалентно использованию метрики локальной связности, которая устойчива к проклятию размерности. UMAP можно применять на данных огромных размерностей.
- Возможно сохранение пространственных взаимоотношений между кластерами.
- Можно применять на новых данных и выполнять обратное преобразование.

Особенности

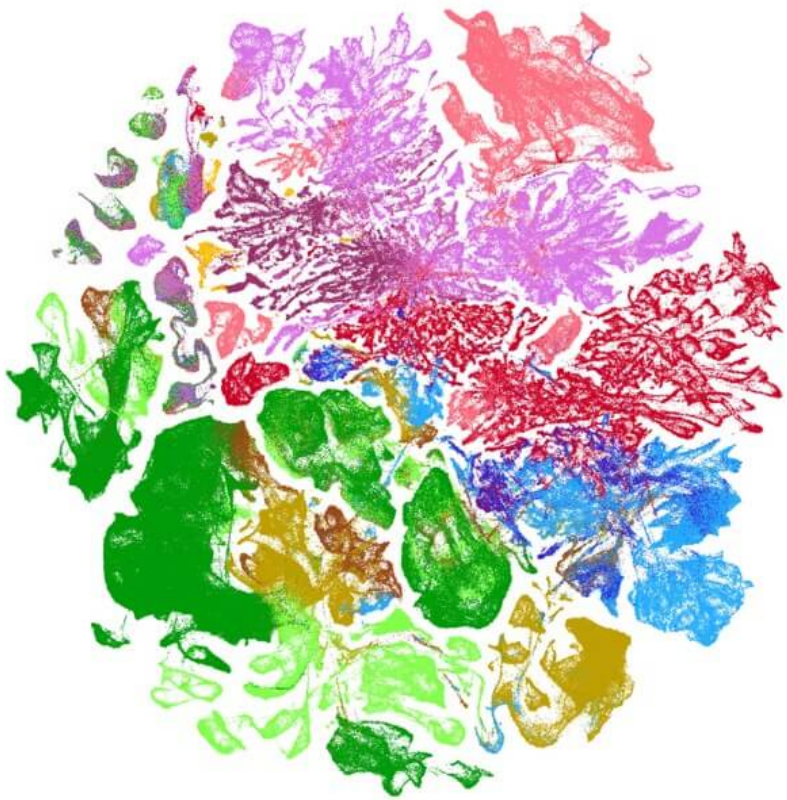
- Введение графов и вероятностей на них эквивалентно использованию метрики локальной связности, которая устойчива к проклятию размерности. UMAP можно применять на данных огромных размерностей.
- Возможно сохранение пространственных взаимоотношений между кластерами.
- Можно применять на новых данных и выполнять обратное преобразование.
- Работает в несколько раз быстрее t-SNE.

Особенности

- Введение графов и вероятностей на них эквивалентно использованию метрики локальной связности, которая устойчива к проклятию размерности. UMAP можно применять на данных огромных размерностей.
- Возможно сохранение пространственных взаимоотношений между кластерами.
- Можно применять на новых данных и выполнять обратное преобразование.
- Работает в несколько раз быстрее t-SNE.
- Поддерживает различные метрики.

Примеры

4 миллиона транскриптомов отдельных клеток взрослого мозга мыши, помеченных по исходному региону мозга и представленных с помощью UMAP (Yao Z. et al., 2023, bioRxiv).



Примеры

Fashion MNIST