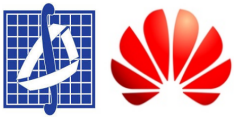
Введение в искусственный интеллект. Машинное обучение

Тема: Методы снижения размерности. Отбор признаков

Бабин Д.Н., Иванов И.Е.

кафедра Математической Теории Интеллектуальных Систем



План лекции

1. PCA
   - SVD-разложение
   - Kernel PCA
   - Sparse PCA
2. MDS
3. Isomap
4. Denoising autoencoder
5. Методы отбора признаков

Проклятие размерности

- Многие задачи машинного обучения содержат тысячи или даже миллионы признаков

Проклятие размерности

- Многие задачи машинного обучения содержат тысячи или даже миллионы признаков
- Типичный пример: если для восстановления плотности одномерной бинарной случайной величины нам потребуется $2 \times 100 = 200$ примеров, то для восстановления 100-мерной с той же точностью— $2^{100} \times 100$

Проклятие размерности

- Многие задачи машинного обучения содержат тысячи или даже миллионы признаков
- Типичный пример: если для восстановления плотности одномерной бинарной случайной величины нам потребуется $2 \times 100 = 200$ примеров, то для восстановления 100-мерной с той же точностью— $2^{100} \times 100$

Вывод

Надо уменьшать размерность данных

Для чего нужно уменьшать размерность данных

- Сжатие данных

Для чего нужно уменьшать размерность данных

- Сжатие данных
- Если данные лежат на многообразии меньшей размерности, то локальные координаты могут оказаться более информативными

Для чего нужно уменьшать размерность данных

- Сжатие данных
- Если данные лежат на многообразии меньшей размерности, то локальные координаты могут оказаться более информативными
- Удаление шума из данных

Для чего нужно уменьшать размерность данных

- Сжатие данных
- Если данные лежат на многообразии меньшей размерности, то локальные координаты могут оказаться более информативными
- Удаление шума из данных
- Выделение главных признаков

Для чего нужно уменьшать размерность данных

- Сжатие данных
- Если данные лежат на многообразии меньшей размерности, то локальные координаты могут оказаться более информативными
- Удаление шума из данных
- Выделение главных признаков
- Визуализация данных

Формальная постановка задачи

- Encoder — процедура сжатия

Формальная постановка задачи

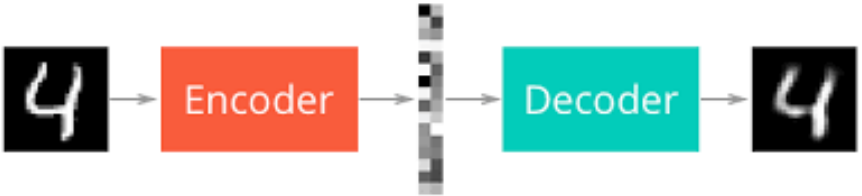
- Encoder — процедура сжатия
- Decoder — процедура восстановления

Формальная постановка задачи

- Encoder — процедура сжатия
- Decoder — процедура восстановления
- Оптимизационная задача $||X - D(E(X))||^2 \to \min$

Формальная постановка задачи

- Encoder — процедура сжатия
- Decoder — процедура восстановления
- Оптимизационная задача $||X - D(E(X))||^2 \to \min$



Manifold learning

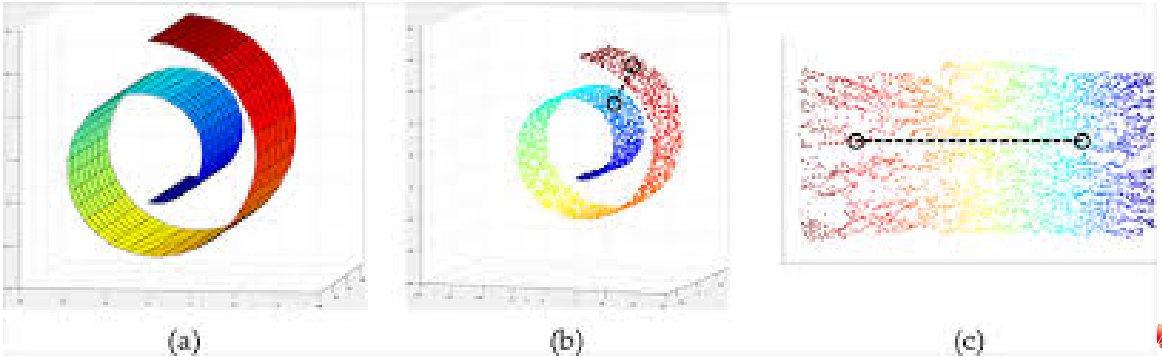
- Многие алгоритмы снижения размерности имеют некоторые предположения о типе многообразия

Manifold learning

- Многие алгоритмы снижения размерности имеют некоторые предположения о типе многообразия
- Если есть априорные знания о данных, это может сильно помочь

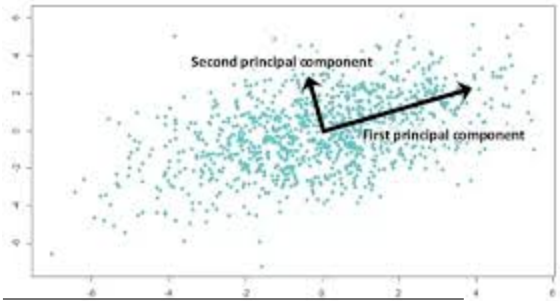
Manifold learning

- Многие алгоритмы снижения размерности имеют некоторые предположения о типе многообразия
- Если есть априорные знания о данных, это может сильно помочь
- Типичный пример



PCA

- Основное предположение о многообразии — гиперплоскость [1]
- Оптимизационнае задачи
  - Наименьшее отклонение от плоскости: $\sum_i dist^2(x_i, L_k) \to \min$
  - Наибольшое среднеквадратическое отклонение проекции на плоскость
  - Encoder и decoder — линейные функции



[1] Pearson, K. (1901). "On Lines and Planes of Closest Fit to Systems of Points in Space". Philosophical

Обозначения

Пусть $x_1, ..., x_\ell$ — наблюдения из $\mathbb{R}^n$. Пусть $\bar{X} = 0$.
Пусть $u_1, ..., u_k$ — ортонормированный базис некоторого подпространства L_k, k < n.

PCA

Обозначения

Пусть $x_1, ..., x_\ell$ — наблюдения из $\mathbb{R}^n$. Пусть $\bar{X} = 0$.
Пусть $u_1, ..., u_k$ — ортонормированный базис некоторого подпространства L_k, k < n.

Задача для k=1

Для текущих наблюдений найти такой u_1, что $||u_1|| = 1$ и выполнено

$$\sum_i ||u_1^T x_i||^2 \to \max$$

Обозначения

Пусть $x_1, ..., x_\ell$ — наблюдения из $\mathbb{R}^n$. Пусть $\bar{X} = 0$.
Пусть $u_1, ..., u_k$ — ортонормированный базис некоторого подпространства L_k, k < n.

Задача для k=1

Для текущих наблюдений найти такой u_1, что $||u_1|| = 1$ и выполнено

$$\sum_i ||u_1^T x_i||^2 \to \max$$

Замечание

$$||x_i - u_1(u_1, x_i)||^2 = ||x_i||^2 - (u_1, x_i)^2$$

Поэтому последенее условие эквивалентно $\sum_i ||x_i - u_1(u_1, x_i)||^2 \to \min$

PCA при k=1

Решение оптимизационной задачи

1. $\frac{1}{\ell}\sum_i ||u_1^T x_i||^2 = u_1^T S u_1$, где $S = \frac{1}{\ell}\sum_i x_i x_i^T$ — матрица ковариаций.

PCA при k=1

Решение оптимизационной задачи

1. $\frac{1}{\ell}\sum_i ||u_1^T x_i||^2 = u_1^T S u_1$, где $S = \frac{1}{\ell}\sum_i x_i x_i^T$ — матрица ковариаций.
2. Так как оптимизационная задача решается при условии $u_1^T u_1 = 1$, то перейдём к безусловной задачи максимизации (метод множителей Лагранжа):

$$u_1^T S u_1 + \lambda_1(1 - u_1^T u_1)$$

PCA при k=1

Решение оптимизационной задачи

1. $\frac{1}{\ell}\sum_i ||u_1^T x_i||^2 = u_1^T S u_1$, где $S = \frac{1}{\ell}\sum_i x_i x_i^T$ — матрица ковариаций.
2. Так как оптимизационная задача решается при условии $u_1^T u_1 = 1$, то перейдём к безусловной задачи максимизации (метод множителей Лагранжа):

$$u_1^T S u_1 + \lambda_1(1 - u_1^T u_1)$$

3. Дифференцируя по параметру и приравнивая к нулю, получаем: $Su_1 = \lambda_1 u_1$ и $u_1^T S u_1 = \lambda_1$
4. u_1 — собственный вектор матрицы S, соотвествующий максимальному собственному значению.

PCA в общем случае

Индукция по числу компонент

Применив индукцию получаем, что $u_1, ..., u_k$ — собственные векторы матрицы S соотвествующие максимальным собственным значениям.

PCA в общем случае

Индукция по числу компонент

Применив индукцию получаем, что $u_1, ..., u_k$ — собственные векторы матрицы S соотвествующие максимальным собственным значениям.

Определение

Направления соотвествующие $u_1, ..., u_k$ называются главными

Основная теорема метода главных компонент

Теорема

Если $k < rkX$, то минимум $||GU^T - X||^2$ достигается, когда столбцы U — это собственные векторы матрицы $X^T X$ соотвествующие максимальным значениям $\lambda_1, ..., \lambda_k$, а матрица $G = XU$. При этом выполнено:

1. $U^T U = I_k$
2. матрица G ортогональна: $G^T G = \Lambda = diag(\lambda_1, ..., \lambda_k)$
3. $U\Lambda = X^T XU$, $G\Lambda = XX^T G$
4. $||GU^T - X||^2 = \sum_{i=k+1}^{n} \lambda_i$

Сингулярное разложение

Следствие

Если в предыдущей теореме взять $k = n$, то

$$X = V\sqrt{\Lambda}U^T,$$

где $U^TU = I_k$, $V^TV = I_k$.

Сингулярное разложение

Следствие

Если в предыдущей теореме взять $k = n$, то

$$X = V\sqrt{\Lambda}U^T,$$

где $U^TU = I_k$, $V^TV = I_k$.

SVD

Как правило большинство реализаций PCA используют SVD разложения, для нахождения главных компонент.

Вероятностная интерпретация PCA

Вероятностная модель

$$p(z) = N(z|0, I)$$

$$p(x|z) = N(x|Wz + \mu, \sigma^2 I)$$

Вероятностная интерпретация PCA

Вероятностная модель

$$p(z) = N(z|0, I)$$

$$p(x|z) = N(x|Wz + \mu, \sigma^2 I)$$

$$x = Wz + \mu + \varepsilon$$

Вероятностная интерпретация PCA

Вероятностная модель

$$p(z) = N(z|0, I)$$

$$p(x|z) = N(x|Wz + \mu, \sigma^2 I)$$

$$x = Wz + \mu + \varepsilon$$

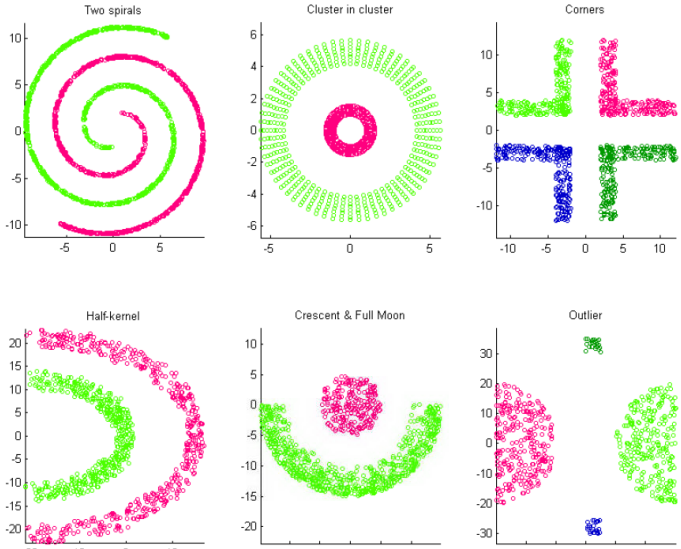
Следствие

Вероятностная интерпретация позволяет обобщить метод PCA и применять к нему вероятностные техники (например, EM-алгорим)

Обобщения PCA

- Kernel PCA
- Sparse PCA

Когда линейные методы не работает



Ядерный PCA (kernel trick)

Если в исходном пространстве сложно разделить выборку, то попробуем перейти в пространство большей размерности[2] $\varphi : X \to H$ и применить линейный PCA там.

[2] Schölkopf, Bernhard (1998). "Nonlinear Component Analysis as a Kernel Eigenvalue Problem". Neural Computation. 10 (5): 1299–1319

Ядерный PCA (kernel trick)

Если в исходном пространстве сложно разделить выборку, то попробуем перейти в пространство большей размерности[2] $\varphi : X \to H$ и применить линейный PCA там.
Ядро – функция $K : X \times X \to \mathbb{R}$, т.ч. $K(x_1, x_2) = \langle \varphi(x_1), \varphi(x_2) \rangle$ при некотором $\varphi : X \to H$, где H – гильбертово пространство.

[2] Schölkopf, Bernhard (1998). "Nonlinear Component Analysis as a Kernel Eigenvalue Problem". Neural Computation. 10 (5): 1299–1319

Ядерный PCA (kernel trick)

Если в исходном пространстве сложно разделить выборку, то попробуем перейти в пространство большей размерности[2] $\varphi : X \to H$ и применить линейный PCA там.
Ядро – функция $K : X \times X \to \mathbb{R}$, т.ч. $K(x_1, x_2) = \langle \varphi(x_1), \varphi(x_2) \rangle$ при некотором $\varphi : X \to H$, где H – гильбертово пространство.

Теорема Мерсера

Функция $K(x_1, x_2)$ является ядром $\Leftrightarrow$ 1) Она симметрична $K(x_1, x_2) = K(x_2, x_1)$ и 2) Неотрицательно определена $\int_X \int_X K(x_1, x_2) f(x_1) f(x_2) dx_1 dx_2 \geq 0$ для любой функции $f : X \to \mathbb{R}$.

[2] Schölkopf, Bernhard (1998). "Nonlinear Component Analysis as a Kernel Eigenvalue Problem". Neural Computation. 10 (5): 1299–1319

Операции над ядрами

- Скалярное произведение $K(x_1, x_2) = \langle x_1, x_2 \rangle$ – ядро
- Константа $K(x_1, x_2) = c$ – ядро
- Произведение ядер $K(x_1, x_2) = K_1(x_1, x_2)K_2(x_1, x_2)$ – ядро
- Для любой $\varphi : X \to \mathbb{R}$ сепарабельная $K(x_1, x_2) = \varphi(x_1)\varphi(x_2)$ – ядро
- Линейная $K(x_1, x_2) = \alpha_1 K_1(x_1, x_2) + \alpha_2 K_2(x_1, x_2)$ – ядро при $\alpha_1, \alpha_2 > 0$, K_1, K_2 - ядрах
- Для любой $\varphi : X \to X$ подстановка $K(x_1, x_2) = K_1(\varphi(x_1), \varphi(x_2))$ – ядро при K_1 - ядро

Примеры ядер

- Полиномиальное ядро с мономами степени d: $K(x_1, x_2) = \langle x_1, x_2 \rangle^d$

Примеры ядер

- Полиномиальное ядро с мономами степени d: $K(x_1, x_2) = \langle x_1, x_2 \rangle^d$
- Полиномиальное ядро с мономами степени $\leq d$: $K(x_1, x_2) = (\langle x_1, x_2 \rangle + 1)^d$

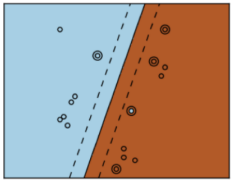
Примеры ядер

- Полиномиальное ядро с мономами степени d: $K(x_1, x_2) = \langle x_1, x_2 \rangle^d$
- Полиномиальное ядро с мономами степени $\leq d$: $K(x_1, x_2) = (\langle x_1, x_2 \rangle + 1)^d$
- Радиальное ядро (RBF): $K(x_1, x_2) = \exp(-\gamma ||x_1 - x_2||^2)$ (наиболее универсальное)

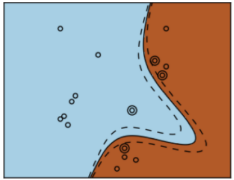
Примеры ядер

- Полиномиальное ядро с мономами степени d: $K(x_1, x_2) = \langle x_1, x_2 \rangle^d$
- Полиномиальное ядро с мономами степени $\leq d$: $K(x_1, x_2) = (\langle x_1, x_2 \rangle + 1)^d$
- Радиальное ядро (RBF): $K(x_1, x_2) = \exp(-\gamma ||x_1 - x_2||^2)$ (наиболее универсальное)

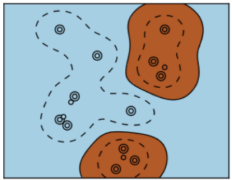
Линейное ядро
$K(x_1, x_2) = \langle x_1, x_2 \rangle$



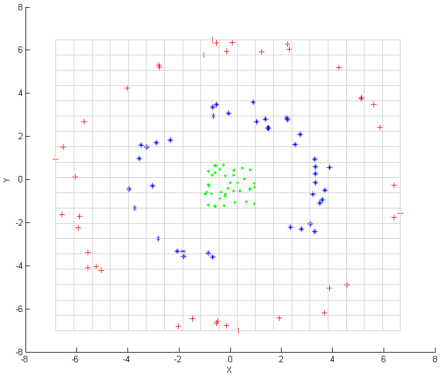
Полиномиальное ядро
$K(x_1, x_2) = (\langle x_1, x_2 \rangle + 1)^3$

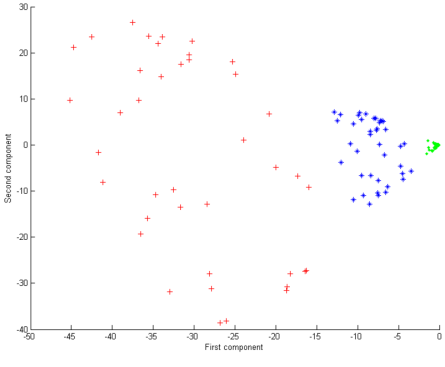


Радиальное ядро
$K(x_1, x_2) = \exp(-||x_1 - x_2||^2)$



Пример работы с ядром $(x^T y + 1)^2$





Sparse PCA

Недостаток PCA

При применении PCA обычно получаются компоненты с небольшим числом нулей.
Обычно это затрудняет интерпретируемость компонент.

Sparse PCA

Недостаток PCA

При применении PCA обычно получаются компоненты с небольшим числом нулей. Обычно это затрудняет интерпретируемость компонент.

Решение

Добавить l1-регуляризацию

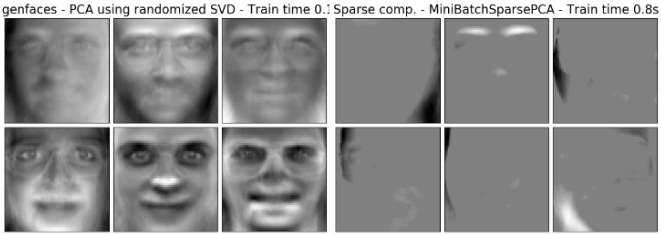
Sparse PCA

Недостаток PCA

При применении PCA обычно получаются компоненты с небольшим числом нулей. Обычно это затрудняет интерпретируемость компонент.

Решение

Добавить l1-регуляризацию



Multidimentional Scaling (MDS)

Дано

Дана матрица попарных расстояний между объектами d_{ij}

Задача

Найти $z_1, z_2, ..., z_n$ удовлетворяющие:

$$\min_{z_1,...,z_n} \sum_{i,j} (d_{ij} - ||z_i - z_j||)^2$$

Вариации MDS

Least squares / Kruskal-Shephard scaling

$$\min_{z_1,\dots,z_n} \sum_{i,j} (d_{ij} - ||z_i - z_j||)^2$$

Sammon mapping

$$\min_{z_1,\dots,z_n} \sum_{i \neq j} \frac{(d_{ij} - ||z_i - z_j||)^2}{d_{ij}}$$

Classical scaling

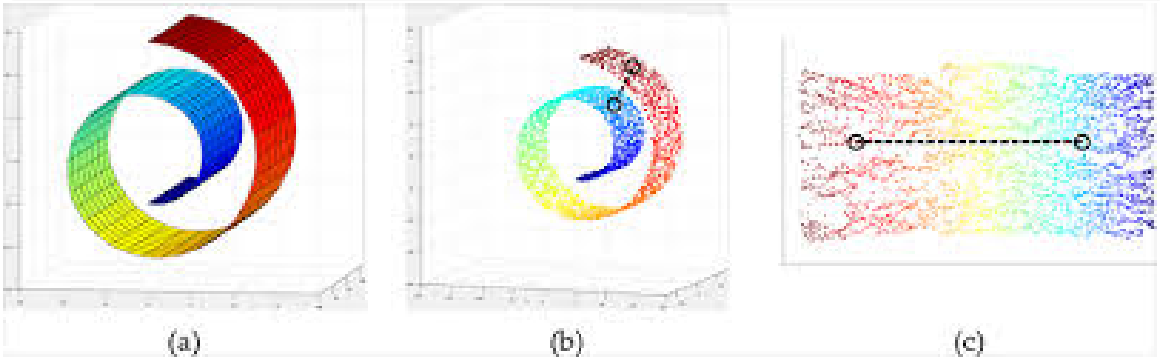
$$d_{ij} = (x_i - \bar{x}, x_j - \bar{x})$$

$$\min_{z_1,\dots,z_n} \sum_{i \neq j} (d_{ij} - (z_i - \bar{z}, z_j - \bar{z}))^2$$

Isomap

Идея

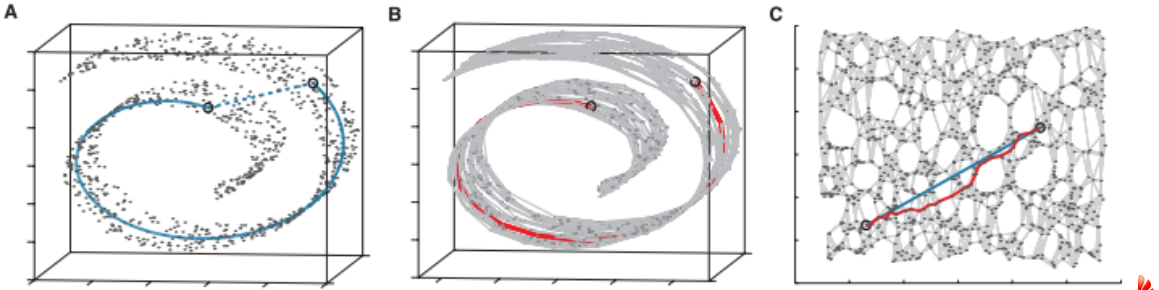
Вместо расстояний использовать геодезические



Isomap

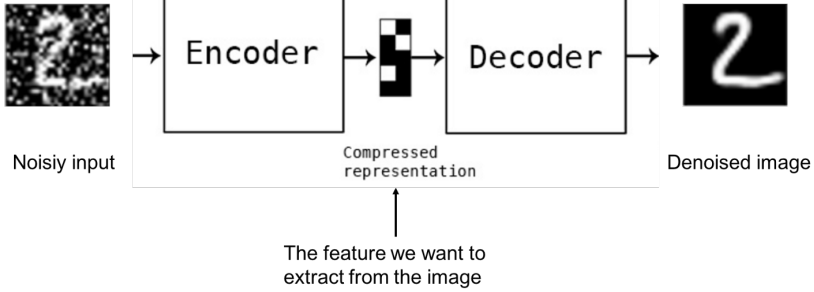
Схема алгоритма

- Построения графа соседства
- Вычисление геодезических на графе
- MDS



Denoising autoencoder

- Модель автоэнкодера — довольно общая архитектура снижения размерности данных
- Может использоваться для генерации новых признаков
- Может использоваться для отчистки данных от шума



Методы отбора признаков

- Статистические
- Основанные на важности признаков для конкретного алгоритма машинного обучения
- Переборные

Методы отбора признаков: полный перебор

Алгоритм

Для каждой сложности наборов искать лучший набор признаков.

Методы отбора признаков: полный перебор

Алгоритм

Для каждой сложности наборов искать лучший набор признаков.

Преимущества

- простота реализации
- гарантированный результат

Методы отбора признаков: полный перебор

Алгоритм

Для каждой сложности наборов искать лучший набор признаков.

Преимущества

- простота реализации
- гарантированный результат

Недостатки

- очень долго работает
- переобучение

Методы отбора признаков: жадные алгоритмы перебор

Алгоритм

На каждой итерации алгоритма добавляется/удаляется наиболее выгодный признак

Методы отбора признаков: жадные алгоритмы перебор

Алгоритм

На каждой итерации алгоритма добавляется/удаляется наиболее выгодный признак

Преимущества

- простота реализации
- работает быстро

Методы отбора признаков: жадные алгоритмы перебор

Алгоритм

На каждой итерации алгоритма добавляется/удаляется наиболее выгодный признак

Преимущества

- простота реализации
- работает быстро

Недостатки

- склонен включать в набор лишние признаки

Методы отбора признаков: генетический алгоритм

Вход: множество F, критерий Q, параметры: d, p_m,
B — размер популяции, T — число поколений;

1: инициализировать случайную популяцию из B наборов:
$B_1 := B$; $R_1 := \left\{J_1^1, \ldots, J_1^{B_1}\right\}$; $Q^* := Q(\varnothing)$;
2: **для всех** $t = 1, \ldots, T$, где t — номер поколения:
3: ранжирование индивидов: $Q(J_t^1) \leqslant \ldots \leqslant Q(J_t^{B_t})$;
4: **если** $B_t > B$ **то**
5: селекция: $R_t := \left\{J_t^1, \ldots, J_t^B\right\}$;
6: **если** $Q(J_t^1) < Q^*$ **то** $t^* := t$; $Q^* := Q(J_t^1)$;
7: **если** $t - t^* \geqslant d$ **то вернуть** $J_{t^*}^1$;
8: породить $t{+}1$-е поколение путём скрещиваний и мутаций:
$R_{t+1} := \left\{\sim(J' \times J'') \;\middle|\; J', J'' \in R_t\right\} \cup R_t$;

Эвристики генетического алгоритма

- Увеличивать вероятности перехода признаков от более успешного родителя к потомку
- Накапливать оценки информативности признаков. Чем более информативен признак, тем выше вероятность его включения в набор во время мутации
- Скрещивать только лучшие индивиды (элитаризм)
- Переносить лучшие индивиды в следующее поколение
- В случае стагнации увеличивать количество мутаций
- Параллельно выращивать несколько изолированных популяций

Заключение

- Метод главных компонент — рабочий инструмент по уменьшению размерности
- Метод главных компонент имеет огромное число обобщений, но не всегда они работают на реальных данных
- Автоэнкодер — универсальная модель для уменьшения размерности
- Отбор признаков и их ранжирование по важности — ключ к пониманию данных
- Точные алгоритмы по отбору признаков не работают на реальных данных, надо использовать эвристики

Источники

На основе материалов сайта http://www.machinelearning.ru.