ТЕХНОСФЕРА

Занятие № 6

Преобразование и получение признаков

Евгений Некрасов

План занятия

1. Извлечение признаков
2. Преобразование признаков
3. Работа с пропущенными данными
4. Представление признаков
5. Отбор признаков
6. Сжатие признаков
7. Преобразование целевой переменной

Извлечение признаков

Сампл (**пример**) - это вектор чисел.

Извлечение признаков - представление реального или цифрового объекта в виде вектора чисел.

Извлечение признаков

Данные бывают:

1. Числовые
2. Дата и время
3. Геоданные (latitude, longitude)
4. Временные ряды
5. Текстовые данные
6. Графические изображения
7. Звук
8. Видео

Геоданные

Что можно извлечь:

1. Что находится по заданной координате
2. Расстояния до особых объектов

Дата и время

Что можно извлечь:

1. Абсолютное время
2. Периодичность (час, день, месяц ...)
3. Временной интервал до особого события

Временные ряды

Что можно извлечь:

1. Среднее значение за период
2. Стандартное отклонение за период
3. Тренд за период
4. Количество пиков за период

Признаки

Признаки бывают:

- **Количественные**
- **Порядковые**
- **Категориальные**
- **Бинарные**

Извлечение признаков

Год выпуска	**2011**
Пробег	**98 000 км**
Кузов	Внедорожник 5 дв.
Цвет	Белый
Двигатель	6.2 л / 409 л.с. / бензин
Коробка	Автоматическая
Привод	Полный
Руль	Левый
Состояние	Не требует ремонта
Владельцы	3 владельца
ПТС	Оригинал
Владение	9 месяцев
Таможня	Растаможен
VIN	XWFS47EF*C0****62
Автокод	Без ограничений

Характеристики модели в каталоге

Зачем преобразовывать признаки?

1. Чтобы конкретный алгоритм машинного обучения их правильно интерпретировал
2. Чтобы конкретный алгоритм машинного обучения эффективно находил взаимосвязи
3. Чтобы внести априорные знания о наборе данных или свойствах признаков

Стандартизация

Для каждого признака в наборе вычитаем среднее и делим на стандартное отклонение.

Применяется к **количественным**, **порядковым** и **бинарным** признакам

Актуально для:

Линейные модели

Метод ближайших соседей

Масштабирование

Значения каждого признака в наборе приводят к диапазону [0,1].

Применяется к **количественным**, **порядковым** и **бинарным** признакам

Актуально для:

Линейные модели

Метод ближайших соседей

Монотонные преобразования

Применение монотонного преобразования к признаку (например: логарифмирование, возведение в степень)

Применяется к **количественным** и **порядковым** признакам

Актуально для:

Линейные модели

Метод ближайших соседей

Линеаризация (регрессия)

Применяем нелинейное преобразование к одному или более признакам чтобы получить новый признак, линейно зависящий от целевой переменной (например: физические законы).

Актуально для:

Линейные модели

Полиномиальные признаки

Заменяем исходный набор признаков полиномом от исходных признаков.

$$(x_1,\ x_2) \to (x_1,\ x_2,\ x_1^2,\ x_2^2,\ x_1 x_2)$$

Применяется к **количественным**, **порядковым** и **бинарным** признакам

Актуально для:

Линейные модели

Метод ближайших соседей

Решающие деревья

Бинаризация

Область значений **количественного** признака делим на N участков и представляем в виде N бинарных признаков.

Применяется к **количественным** и **порядковым** признакам

Актуально для:

Линейные модели

One hot encoding

Представление признака с N категорий как N бинарных признаков.

Применяется к **категориальным** и **порядковым** признакам

Актуально для:

Линейные модели

Метод ближайших соседей

Некоторых типов решающих деревьев

Хэширование признаков

Не всегда можно рассчитывать, что категориальные признаки не будут принимать новых значений.

$x' = hash(x)$

произвольное значение → натуральное число 0..N

Задача

Алгоритм: k ближайших соседей с евклидовым расстоянием

Признаки:

1. категория кинотеатра [1..43]
2. день недели [1..7]
3. час суток [0..24]
4. цена билета [100..1000]

Целевая переменная: заполненность зала в %

Что делать?

Работа с пропущенными данными

Простые решения:

- Некоторые алгоритмы поддерживают работу с пропущенными данными "из коробки".
- Закодировать пропущенные данные особым значением (0, -999 и т.п.).
- Закодировать пропущенные данные типичным значением (среднее, медиана, наиболее частое значение).

Работа с пропущенными данными

Более сложные решения:

- Для временных рядов можно брать соседнее значение соседей.
- Можно использовать модель для заполнения пропущенных данных (например алгоритм k ближайших соседей).

Отбор признаков, зачем?

- Меньше признаков - выше производительность.
- Меньше признаков - проще их сбор и преобразование.
- Снижение количества признаков может приводить как к снижению так и к росту точности модели.

Каждый признак - это сигнал + шум

Статистический подход

- Выбрасываем признаки, значение которых константно на тренировочном наборе данных (всем или большей части)
- Выбрасываем признаки, которые слабо статистически связаны с целевой переменной (например: корреляция)

Проблема статистической связи

Признак 1	Признак 2	Целевая переменная
1	0	1
1	0	1
1	0	1
0	1	1
1	1	0
0	0	0
0	0	0
0	0	0

Отбор признаков по важности для модели

Смотрим на какие признаки модель опирается при принятии решений.

- Веса при признаках для линейной модели
- Как часто происходят сплиты по признакам в Random Forest
- Нулевые коэффициенты в l1 регуляризованной линейной модели

Последовательный отброс признаков

Пусть N - количество признаков

1. Из полного набора признаков по очереди выбрасываем каждый и таким образом получаем N новых наборов признаков.
2. Оцениваем качество модели на каждом из N наборов признаков.
3. Выбираем набор с наилучшей точностью.
4. Переходим на шаг 1 (опционально если точность улучшилась).

Последовательный набор признаков

Пусть дано M основных и N дополнительных признаков.

1. Из N дополнительных признаков по очереди добавляем каждый к основному набору из M признаков и таким образом получаем N новых наборов признаков.
2. Оцениваем качество модели на каждом из N наборов признаков.
3. Выбираем набор с наилучшей точностью.
4. Переходим на шаг 1 (опционально если точность улучшилась).

Сжатие признаков

Признак = сигнал + шум

Идея:

Давайте из большого количества признаков извлечем общий сигнал

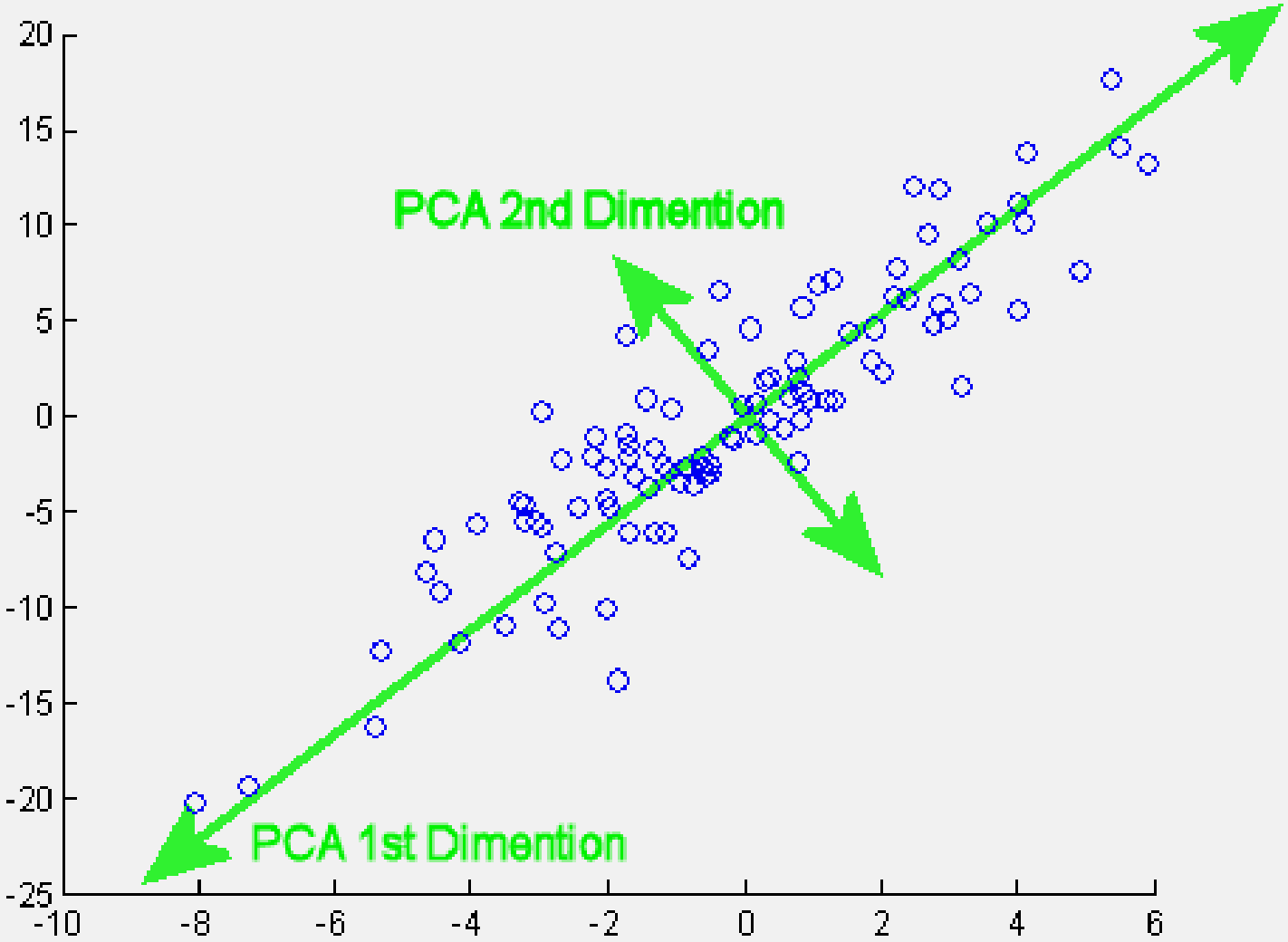
Сжатие признаков PCA

Метод главных компонент (PCA) - один из основных способов уменьшить размерность данных, потеряв наименьшее количество информации.

PCA - это линейное преобразование.

Задача PCA - найти подпространства меньшей размерности, в ортогональной проекции на которые разброс данных максимален.

Сжатие признаков PCA



Сжатие признаков, другие методы

- Kernel PCA – не линейный вариант PCA
- ICA – выявление независимых компонент сигнала
- Автоэнкодеры – тип искусственной нейронной сети
- и другие

Преобразование целевой переменной

Иногда целевую метрику нельзя оптимизировать напрямую (например ROC-AUC)

В таком случае монотонные преобразования целевой переменной могут повысить точность по целевой метрике

Соревнование "Property prices"

https://www.kaggle.com/c/msu-introduction-to-machine-learning-property

Домашнее задание №#6

- Сделать сабмит решения
- Выложить решение на github.com
- Прислать ссылку на код решения, свой профиль kaggle

Срок сдачи

12 ноября 2017

ТЕХНОСФЕРА

Спасибо за внимание!

Евгений Некрасов

e.nekrasov@corp.mail.ru