ОБЗОР МЕТОДОВ ПРЕДОБРАБОТКИ ДАННЫХ

Методам предобработки в литературе по машинному обучению и анализу данных (МО и АД) отводят достаточно скромное место. Бóльшая часть материала посвящена описанию самих алгоритмов МО и АД и их применению на чистых, уже подготовленных данных. Как правило, упоминаются лишь самые математически нагруженные методы предобработки, например, методы сокращения пространства признаков в контексте задачи улучшения показателей качества модели.

При этом, практикам анализа данных хорошо известно, насколько значим вклад предобработки в успешное решение задачи. Вот цитата авторитетного исследователя данных Григория Пятецкого-Шапиро: *«Наиболее сложные этапы анализа — очистка данных, предобработка, выбор переменных. Они отнимают много времени, но если всё выполнено на должном уровне, результат не заставит себя долго ждать».*

Попыток системного и всеобъемлющего описания методов предобработки немного. Информация об отдельных методах предобработки содержится в статьях с результатами анализа данных в различных областях, представлена в небольших практических примерах решения задач на реальных данных на kaggle, рассыпана в постах на хабре ...

Знакомство с этими материалами показывает многообразие методов предобработки, их ориентированность на обязательное понимание физического смысла как самих данных, так и их дефектов, знание предметной области, особенностей источников данных. Именно в силу различной природы данных методы обработки пропусков в социологическом исследовании будут принципиально отличаться от методов обработки пропусков в данных от промышленного датчика!

С развитием цифровых производств Индустрии 4.0, чистота данных становится критически важным требованием; в среде специалистов по данным выделилась отдельная группа - Data Quality Engineer, инженеры по качеству данных. Задача таких специалистов - создание автоматизированной системы контроля качества данных в реальном времени для их использования в системах принятия решения различного уровня.

О первичности чистых данных по отношению к их количеству, и даже к алгоритмам МО и АД, образно сказал Peter Norvig: *«More data beats clever algorithms, but better data beats more data» - Больше данных лучше умных алгоритмов, но лучшие данные лучше их большого количества.*

В этой памятке предпринята попытка сделать систематизированное, краткое и ёмкое описательное представление методов предобработки данных. И если в ходе своих собственных исследований Вы хотя бы раз обратитесь к этой шпаргалке, чтобы воскресить в памяти подходы к обеспечению чистоты, полноты, безызбыточности и непротиворечивости данных, то автор будет считать цель достигнутой. Главное, помните о принципе GIGO: Garbage In – Garbage Out, мусор на входе – мусор на выходе (даже при верных алгоритмах)!

Если вы хотите узнать о методах предобработки подробнее, с формулами и разбором примеров в Python, Вы можете заказать в нашей компании специализированный курс.

С уважением, Ваша Analytera!

МЕСТО ПРЕДОБРАБОТКИ В ТИПОВОМ ПРОЦЕССЕ РЕШЕНИЯ ЗАДАЧ НА ДАННЫХ

ЭТАП 1: ИЗУЧЕНИЕ ДАННЫХ

Цель этапа: обеспечить ясность данных. Этап включает в себя следующие процедуры.

a. Понимание данных в предметной области

Перед выполнением любых процедур на данных, постарайтесь иметь ответы на следующие вопросы:

- какой объект или процесс описывают данные?
- к какой предметной области относится этот объект / процесс?
- какие специалисты являются экспертами в данной предметной области?
- какими параметрами и в каком диапазоне значений описывается этот объект / процесс в своей предметной области?
- насколько этот объект / процесс стационарен во времени?
- как бы сформулировали цель исследования специалисты в предметной области?
- как можно сформулировать цель исследования используя параметры, входящие в имеющийся набор данных?

b. Оценка полноты данных

Чтобы оценить достаточность имеющихся данных для решения задачи, постарайтесь ответить на следующие вопросы:

- насколько полно имеющиеся переменные описывают исследуемый объект / процесс?
- какие внешние факторы могут оказывать влияние на исследуемый объект / процесс?

c. Формальное описание

Чтобы иметь формальное, технологическое описание имеющегося набора данных, необходимо знать:

- размер набора;
- состав и тип переменных;
- статистическое описание переменных;
- количество пропусков в данных.

d. Визуальный анализ

Чтобы получить представление о зависимостях и структурах в данных, используйте различные методы визуализации:

- графики $y = f(x)$, $x = f(t)$, чтобы увидеть характер зависимости между переменными и ход их значений на временной оси;
- гистограммы, чтобы увидеть распределение значений параметров в диапазоне имеющихся значений;
- парные диаграммы рассеяния, чтобы увидеть характер зависимостей между переменными;
- диаграммы размаха ("ящик с усами"), чтобы оценить распределение вероятностей значений переменной и выбросы;
- матрицы корреляции, чтобы получить численные оценки зависимости переменных;
- кластерные и иерархические представления, чтобы увидеть структуру в данных;

Замечания: a) в некоторых источниках визуальный анализ рассматривается как самостоятельное направление в анализе данных – Visual Mining; b) визуализации используются на всех этапах АД и МО; c) если о природе данных ничего не известно или цель исследования не может быть сформулирована в терминах предметной области, визуализации помогут сформулировать вопросы формально.

Рекомендации:

- *на этапе изучения данных активно взаимодействуйте со специалистами в предметной области, старайтесь получить представление об изучаемом объекте / процессе, выходящее за описание в рамках имеющихся переменных;*
- *на этапе изучения данных избегайте соблазна сделать какие-либо выводы касательно основного вопроса исследования.*

ЭТАП 2: ОЧИСТКА ДАННЫХ

Цель этапа: обеспечить полноту, истинность, корректность, непротиворечивость данных. Этап может включать в себя следующие процедуры.

a. Заполнение пропусков

Основные методы обработки пропусков:

- ничего не делать, если алгоритм нечувствителен к пропускам или способен сам восстановить значения в пропусках (например, XGBoost, Light GBM);
- удалить строки с пропусками, если данных много;
- удалить столбец с пропусками, если параметр не имеет существенного значения;
- заполнить пропуски соответствующими логике значениями;
- заполнить самым частым значением или константой;
- заполнить расчётными значениями: среднее, медиана, мода;
- заполнить значениями случайной выборки из аналогичного распределения;
- заполнить пропуски методом «горячей колоды» (Hot Deck);
- восстановить значения функцией регрессии;
- восстановление полиномиальной аппроксимацией;
- восстановить значение методом классификации (например, k-NN);
- восстановить значения нейросетевым методом;
- маркировка пропусков (пропуск становится признаком).

Замечание: a) форматы представления отсутствующих значений могут быть различными: пустые ячейки, прочерки, нули, NaN / NaT (Not a Number / Not a Time), символы, радикальные выбросы (999), строковые переменные («неизвестно») и т.п.

b. Обработка невозможных значений

Поиск невозможных значений требует понимания физического смысла переменных, а выбор метода их обработки - понимания причин появления таких значений.

Подходы к обработке невозможных значений аналогичны подходам к обработке пропусков.

c. Обработка дубликатов записей

Обрабатывать дубликаты нужно, если они возникли в результате технического сбоя, ошибок ввода, или этого требует подход к решению задачи.

d. Исправление форматов ввода

Ошибки форматов данных чаще всего возникают при ручном вводе или интеграции данных.

Размерности параметров целесообразно приводить к размерностям, используемым на практике.

e. Сглаживание выбросов

Простейший метод поиска выбросов – визуальный анализ.

Основные приёмы обработки:

- удалить записи с выбросами, если методика исследования это допускает;
- провести сглаживание.

Замечания: a) большинство алгоритмов чувствительны к выбросам.

***Рекомендации*:**

- *чтобы выбрать верный метод обработки дефекта, необходимо: а) понимать физический смысл переменной с дефектом, б) иметь гипотезу о появлении дефекта, в) знать характер дефекта: случайный, псевдослучайный или предсказуемый;*
- *отличайте нуль, как фактическое значение параметра, от нуля, обозначающего дефект данных: пропуск, невозможное значение и др.;*
- *отличайте пустую ячейку, как фактическое отсутствие значения переменной, от дефекта пропуска в данных;*
- *методология обработки дефектов на предикторах и целевых переменных могут быть различны.*

ЭТАП 3: ПРЕОБРАЗОВАНИЕ ДАННЫХ

Цель этапа: обеспечить структурированность, однородность, согласованность и безызбыточность данных. Этап может включать в себя следующие процедуры.

a. Приведение типов данных

Задача: выполнить требования алгоритмов к типам данным. Основной метод - кодирование номинативных переменных.
Замечание: некоторые алгоритмы нечувствительны к типам входных данных, например, деревья решений (Decision Tree), случайный лес (Random Forest).

b. Нормализация

Задача: улучшить качество работы алгоритмов за счёт приведения данных к нужному диапазону («в одну шкалу»). Основные методы:
- нормализация на максимум;
- нормализация на интервал;
- ранговая нормализация.

Замечание: существенно повышает эффективность метрических алгоритмов классификации: метод ближайшего соседа (k- Nearest Neighbors), метод k-средних (k-means), машина опорных вектором (Support Vector Machine).

c. Стандартизация

Задача: улучшить качество работы алгоритмов за счёт приведения данных к стандартному нормальному распределению, где:
математическое ожидание $\mu = 0$;
стандартное отклонение $\sigma = 1$.
Замечание: существенно повышает эффективность метрических алгоритмов классификации: метод ближайшего соседа (k- Nearest Neighbors), метод k-средних (k-means), машина опорных вектором (Support Vector Machine).

d. Создание новых признаков (Feature engineering)

Задача: дополнить данные новыми параметрами, позволяющими повысить интерпретируемость модели и выявить новые признаки. Основные методы:
- агрегирование, или вычисление статистик на наборе однотипных параметров;
- обобщение, или создание групповых описательных признаков;
- квантование, или кодирование интервалов вещественных значений признака для перевода к порядковым значениям;
- выделение временных интервалов на временном ряде.

e. Обогащение данных (Data enrichment)

Задача: дополнить имеющийся набор новыми данными, определённо влияющими на исследуемый объект или процесс, например:
- метеоусловия сказывается на посещаемости магазинов;
- курс валют влияет на спрос на зарубежные поездки.

Замечания: а) учитывайте стоимость и доступность данных, используемых для обогащения; б) учитывайте часовые пояса при интеграции данных на временных рядах из разных источников.

f. Оптимизация пространства признаков (Data reduction)

Задача: повысить производительность вычислений и улучшить интерпретируемость результатов за счёт сокращения пространства признаков. Основные методы:
- корреляционный анализ;
- метод главных компонент (PCA – Principal Component Analysis);
- многомерное шкалирование (MDS – Multidimensional Scaling);
- факторный анализ, t-SNE и другие методы.

Замечание: а) PCA выявляет линейные зависимости между переменными, а MDS нелинейные; б) методы сокращения пространства признаков используются в задачах визуализации многомерных данных.

ЭТАП 4: ОТБОР ПЕРЕМЕННЫХ

Цель этапа: обеспечить максимальную эффективность модели на подготовленном наборе данных.

a. Отбор переменных (Feature selection)

Задача: определить совокупность переменных, на которых будет получен наилучший результат предсказания.

В общем случае метод отбора зависит от алгоритма, на котором строится модель, например:
- для регрессионных моделей используют методы прямого, обратного, последовательного отбора, метод лучших подмножеств.