Титульный лист материалов по дисциплине

(заполняется по каждому виду учебного материала)

ДИСЦИПЛИНА	**Технологии извлечения знаний из больших данных** (полное наименование дисциплины без сокращений)
ИНСТИТУТ	ИКБ
КАФЕДРА	**КБ-4 «Интеллектуальные системы информационной безопасности»** полное наименование кафедры)
ВИД УЧЕБНОГО МАТЕРИАЛА	**Лекция** (в соответствии с пп.1-11)
ПРЕПОДАВАТЕЛЬ	**Никонов В.В.** (фамилия, имя, отчество)
СЕМЕСТР	**3 семестр 2023/2024 уч. года** (указать семестр обучения, учебный год)

Понижение размерности

Задача понижения размерности — это скорее вспомогательная задача для решения других задач машинного обучения, например классификации, регрессии или кластеризации, однако и самостоятельные применения у нее тоже есть.

Суть задачи понижения размерности состоит в том, чтобы имея данные с большим количеством признаков (столбцов), преобразовать их в новую таблицу с меньшим количеством столбцов. Количество строк (объектов) при этом останется неизменным.

Название «понижение размерности» означает уменьшение количества признаков, описывающих каждый объект. Как мы видим, после применения понижения размерности данные имеют такой же вид, как до применения алгоритма, а значит, к ним можно применять все изученные ранее алгоритмы. Так для чего же нужно понижение размерности?

Сжатие данных

Большая таблица данных может занимать много места на жестком диске, и уменьшив количество признаков в N раз, мы уменьшим размер файла с данными в те же самые N раз.

Ускорение предсказаний

Если алгоритму предсказания (например, алгоритму предсказания ухода клиента или алгоритму кластеризации) нужно обработать тысячи признаков, это будет занимать гораздо больше времени, чем если он будет обрабатывать десятки признаков. При этом во многих задачах, особенно связанных с онлайн-сервисами, существуют ограничения на время выполнения предсказаний, например поисковой запрос должен выполняться за доли секунды.

Визуализация данных

Если алгоритм понижения размерности составит новую таблицу с двумя столбцами, такие данные будет легко визуализировать, отложив по осям два новых признака.

Более компактное и «правильное» описание объектов

Новые признаки могут описывать объекты более емко, чем исходные, что упростит работу другим алгоритмам. Более того, среди исходных признаков могут быть такие, которые ухудшают предсказания, и при понижении размерности эти признаки будут удалены.

Далее мы рассмотрим основные группы алгоритмов понижения размерности, каждая из которых была разработана для выполнения одной или двух из указанных выше целей.

Отбор признаков

Обычно выделяют два типа алгоритмов понижения размерности: алгоритмы отбора признаков и алгоритмы выделения новых признаков на основе исходных. Первые просто удаляют столбцы из таблицы, а вторые вычисляют новые столбцы по формулам, включающим все исходные столбцы. Сначала сосредоточимся на первом типе.

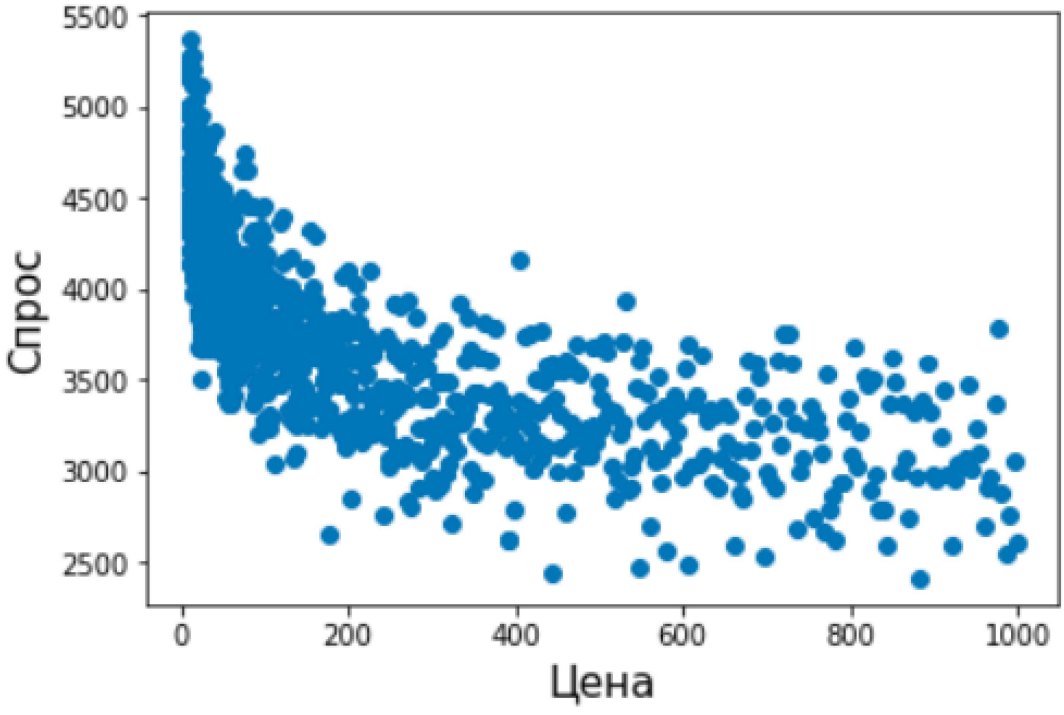
Казалось бы, здесь все просто: нужно удалить часть столбцов из таблицы, чтобы она занимала меньше места, а предсказания выполнялись быстрее. Но как определить, какие признаки удалять? Для примера будем рассматривать задачу предсказания спроса на товар в интернет-магазине: объект — товар; нужно предсказать, какое количество товара купят в течение

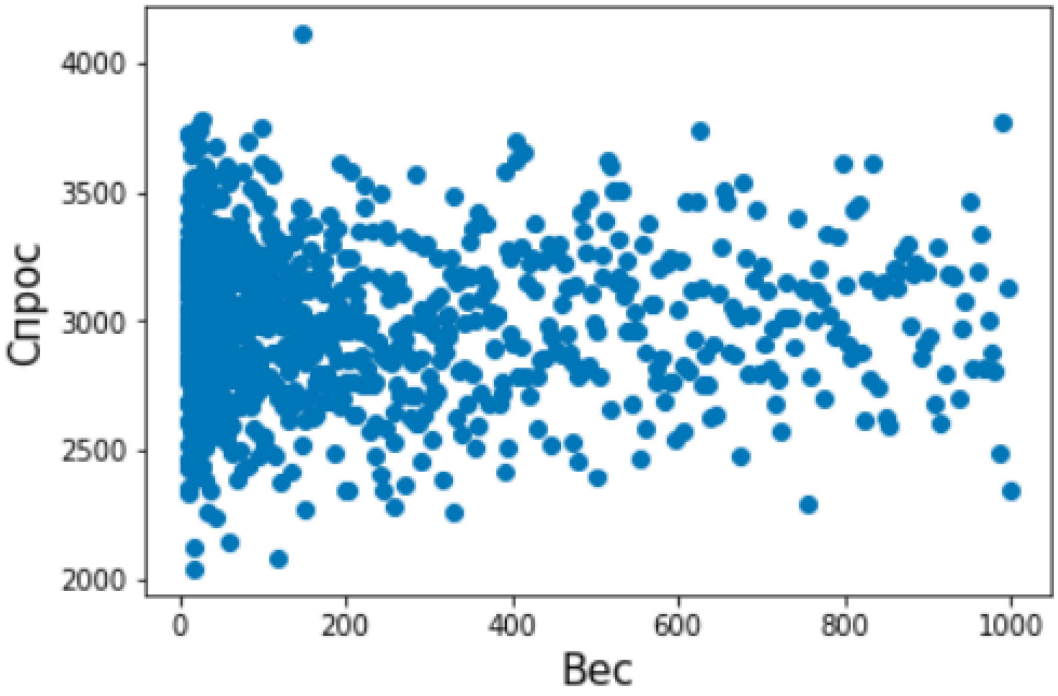
следующего месяца, чтобы спланировать закупки, это задача регрессии. Мы рассматриваем задачу регрессии неспроста: методы отбора признаков, как правило, применяют именно в задачах классификации и регрессии, то есть в задачах обучения с учителем.

Концептуально мы хотим удалять такие признаки, которые никак не помогают нам правильно выполнить предсказание. Например, очевидно, что если у каждого товара есть уникальный артикул, не повторяющийся между товарами и имеющий вид случайного набора цифр, например 8433498 или 89923900, то никакой информации о спросе на товар данный артикул не несет и соответствующий столбец можно легко удалить из данных. Но с другими признаками все может быть не так просто.

- **Метод фильтрации признаков**

Самые простые методы отбора признаков подразумевают анализ каждого признака отдельно и называются методами фильтрации признаков. Например, если у нас есть признаки «стоимость товара» и «вес товара», то мы можем построить графики стоимость-спрос и вес-спрос и по ним постараться увидеть, зависит ли спрос от какой-либо величины.





По графикам видна явная зависимость между ценой и спросом (чем дороже товар, тем меньше его покупают), а вот зависимости между весом товара и спросом не видно, откуда напрашивается вывод, что столбец «вес товара» можно удалить. Но может оказаться так, что вес товара влияет на спрос в совокупности с другими признаками, например, более легкие туристические палатки покупают чаще, чем более тяжелые, а для канцелярских ручек наоборот — такие сложные зависимости методы фильтрации обычно не учитывают. Отметим, что в методах фильтрации используются не только визуальные, но и числовые методы, количественно оценивающие влияние признака на целевую переменную. Также отметим, что фильтры не используют никаких алгоритмов предсказания и работают только и непосредственно с данными.

- **Оберточные методы отбора признаков**

Другая группа отбора признаков называется оберточные методы (wrappermethods). Эти методы — более сложный отбор признаков, включающий обучение алгоритмов и измерение их качества на тестовой выборке. Например, чтобы оценить, важен ли признак «вес товара», оберточный метод сначала обучит алгоритм на данных, включающих этот признак, затем на данных без этого признака (столбец «вес товара» удален) и

сравнит, ухудшилось ли качество: если спрос на товары стал предсказываться менее точно, значит, признак «вес товара» важный, иначе он удаляется. Чтобы выполнить отбор признаков, оберточный метод может, например, начать с полной таблицы данных и по очереди удалять наименее важные признаки, то есть те, удаление которых меньше всего снижает точность предсказаний. Или наоборот: начать с пустой таблицы и по очереди добавлять в нее важные признаки.

- **Встроенные методы отбора**

Наконец, третья группа методов для отбора признаков называется встроенные методы (embeddedmethods). Это методы, подразумевающие, что алгоритм обучения сам может определить, какие признаки важные, а какие нет. Вспомним, что в блоке 4, посвященном задаче регрессии, мы обсуждали линейные модели: те, которые умножают значения признаков на веса и складывают. Если у какого-то признака после обучения получится вес, равный нулю, это значит, что признак не влияет на предсказание и его можно удалить — это и есть пример встроенного метода. А точнее, встроенным методом является регуляризация — специальный механизм, который помогает настроить в линейных моделях такие веса, что среди них будет много нулевых. Многие алгоритмы классификации и регрессии включают встроенные методы отбора признаков.

Итак, методы отбора признаков помогают определить, какие признаки можно удалить из данных, чтобы уменьшить размер данных и ускорить выполнение предсказаний. Для определения нерелевантных признаков можно анализировать признаки по отдельности (методы фильтрации), можно обучать алгоритмы с разными наборами признаков и оценивать важности признаков по изменению точности предсказаний (оберточные методы), а можно воспользоваться встроенными механизмами алгоритмов обучения. Помимо указанных преимуществ, отбор признаков повышает интерпретируемость алгоритмов: гораздо проще проанализировать алгоритм с десятком признаков, чем с тысячей. С другой стороны, неосторожный

отбор признаков может привести к снижению уровня качества (точности предсказаний). Подробнее прочитать о различных методах отбора признаков можно по ссылке.

Выделение признаков

Методы отбора признаков выполняют понятное действие — удаляют столбцы. Методы же выделения новых признаков делают более сложную операцию: они создают новые столбцы, которые вычисляются по формулам, зависящим от имеющихся в данных столбцов. Иными словами, мы получаем новые столбцы, в которых стоят какие-то числа. Что эти числа означают — как правило, известно только алгоритму, который их сделал, для человека это будет просто набор неинтерпретируемых чисел. Так для чего это делать?

Несмотря на то, что человеку новые признаки непонятны, они могут содержать гораздо больше информации об объектах, чем любой набор исходных признаков того же количества.

Пример использования

Пусть у нас есть таблица клиентов банка с признаками: число кредитов, заработная плата, суммы на счетах и вкладах, статистика пользования банковскими продуктами и т.д. Например, система выделения новых признаков выделила два признака: первый признак задает уровень склонности клиента к риску (0 — склонный к риску, 1 — предпочитает стабильность), а второй признак задает уровень состояния клиента (0 — бедный, 1— богатый). Ни одна пара признаков из исходных данных не отражает эту информацию полностью, например первый новый признак агрегирует информацию о кредитах, использовании брокерских продуктов и т.д., а второй — о заработной плате, суммах на вкладах и т.д. При этом новые признаки позволяют легко кластеризовать клиентов на четыре группы и упростить рекомендации продуктов для них, например богатым клиентам, предпочитающим стабильность, можно предлагать вложения в драгоценные металлы, а богатым, склонным к риску, — вложения в акции.

Методы выделения признаков используются в широком спектре задач, причем как в задачах обучения с учителем, так и в задачах обучения без учителя. Проблема только в том, что методы выделения признаков не позволяют интерпретировать признаки, но попытаться выяснить их значения, например, по итогам кластеризации, вполне возможно. Более того, новые признаки могут быть полезны сами по себе, без интерпретации. Например, они активно используются в поисковых системах: у двух похожих объектов будут похожие значения новых выделенных признаков, и для поиска в базе документов по запросу нужно найти новые признаки и выделить документы с наиболее похожими значениями новых признаков.

- **Метод главных компонент**

Один из наиболее популярных методов выделения признаков называется метод главных компонент (англ. PrincipalComponentAnalysis, PCA): он задает новые признаки как линейные формулы от исходных. Иными словами, каждый новый признак будет равен сумме исходных признаков, умноженных на веса, и веса настраиваются в процессе обучения. Очень похоже на линейные модели регрессии, только в регрессии целевая переменная известна, а здесь алгоритм сам «придумывает», что будет означать полученная сумма. Достоинства и недостатки такие же, как у других линейных методов: метод работает быстро, но выделяет слишком простые зависимости. Похожим образом на PCA работает другой алгоритм — SingularValueDecomposition, SVD.

- **Автокодировщик**

Автокодировщик (autoencoder) - это специальная архитектура нейронных сетей, которые используют вместо линейной формулы. Такая архитектура также позволяет осуществить обучение без учителя с использованием алгоритма обратного распространения ошибки. Про нейронные сети мы подробно поговорим в следующих темах, здесь отметим, что автокодировщики часто применяются для видео или изображений,

чтобы, например, убирать лишний шум, находить похожий контент или искать материалы по текстовому запросу.

Понижение размерности в работе с текстами

Для выделения признаков из текстов часто используется тематическое моделирование, например алгоритмы LSA (LatentSemanticAnalysis) и LDA (LatentDirichletAllocation). В этом случае сам текст представляется как набор слов (неупорядоченный), иными словами, вычисляются частоты слов. Новые признаки задают темы: один новый признак — одна тема, при этом каждый объект (текст) может относиться к нескольким темам. Например, текст может быть одновременно про политику, экономику и немножко литературу.

Выделенные признаки опять же используются для понимания структуры текстового набора (например, можно прочитать несколько текстов из каждой темы вместо того, чтобы читать все тексты в наборе данных) или для поиска похожих текстов (книг, постов в соц. сетях и т.д.). Тематические модели удобно использовать для группировки новостей или обращений клиентов по похожим тематикам, для этого к новым признакам нужно применить алгоритм кластеризации.

Измерение качества задачи

Как и в задаче кластеризации, в задаче понижения размерности, в частности, в выделении признаков, измерить качество очень сложно. Часто используют такой метод: оценивают, насколько точно по новым признакам можно восстановить старые (как правило, методы выделения признаков это позволяют). Другая хорошая альтернатива — использовать метрику качества той задачи, в которой используются новые признаки, например классификации.

Инструменты для задачи понижения размерности

Программное обеспечение для задачи понижения размерности такое же, как для предыдущих задач: язык программирования Python и библиотека Scikit-learn. Для тематического моделирования обычно используются отдельные библиотеки, напримерGensim.

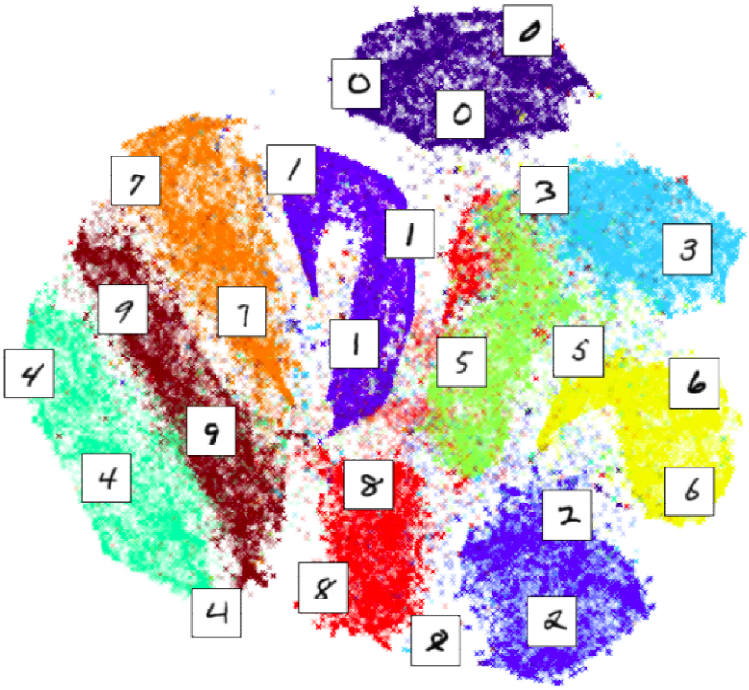
Визуализация данных

Отдельного внимания заслуживают методы визуализации данных. Они тоже относятся к методам выделения новых признаков, но ставят перед собой особую цель — найти два (максимум три) признака, такие, что в осях этих признаков получается информативная визуализация данных. Формально для этой цели можно использовать любые указанные ранее алгоритмы: PCA, автокодировщики или методы тематического моделирования, но на практике такие визуализации получаются не очень красивыми и понятными.

- **Многомерное шкалирование**

Метод под названием многомерное шкалирование (MultidimensionalScaling, MDS) старается найти такие новые признаки так, чтобы схожести между объектами, измеренные по исходным признакам), были примерно такими же, как схожести между объектами, измеренные по новым признакам. Про схожести между объектами мы говорили в блоке про кластеризацию: схожесть — это числовая величина, оценивающая, насколько похожи два объекта.

Метод MDS был усовершенствован, и получился **метод t-SNE** (t-Distributedstochasticneighborembedding) — самый популярный на сегодня метод визуализации объектов. С его помощью получаются такие визуализации:



На этом изображении приведена визуализация набора данных для распознавания рукописных цифр. По осям отложены два выделенных признака (два столбца), каждая точка — это один объект (строка таблицы данных), цветами отмечены разные цифры, в квадратах приведены примеры цифр, соответствующих рядом расположенным точкам. Видно, что цифры одного номера располагаются на визуализации рядом, более того, похожие цифры тоже располагаются рядом, например, 4, 9 и 7.

Применение визуализации

Аналогичные визуализации можно выполнять для любых данных, например визуализировать клиентов, товары в онлайн-магазине, дома на продажу. Как правило, если на визуализации видна зависимость, например классы разбиваются на отдельные области, как на изображении выше, то задача будет решаться с высоким уровнем качества, а если цвета перемешаны — то с не очень высоким.

Если выполнялась кластеризация объектов, на такой визуализации можно отобразить ее цветами и оценить, разделились ли объекты по цветам, как на изображении выше (качественная кластеризация), или цвета перемешаны (плохая кластеризация). Однако это будет лишь приблизительным способом оценки качества кластеризации.

Визуализация с помощью t-SNE позволяет посмотреть на данные «свысока», но окончательных выводов по ней лучше не делать, потому что значения осей интерпретировать невозможно.

Таким образом, методы выделения признаков позволяют найти новые признаки, агрегирующие информацию, хранящуюся в исходных признаках. Выделенные признаки можно использовать для определения схожести объектов, для решения других задач, например классификации или кластеризации, и для визуализации данных.