МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Ярославский государственный университет им. П.Г. Демидова»

Кафедра информационных сетей и технологий

Сдано на кафедру

«\_\_» \_\_\_\_\_\_\_\_ 2018 г.

Зав. кафедрой, к.ф.-м.н., декан

\_\_\_\_\_\_\_\_ Чалый Д.Ю.

# Анализ методов машинного обучения для решения задачи

**“Predict future sales”**

Бакалавриат Прикладная информатика - 09.03.03

Научный руководитель:

Зав. Кафедрой, к.ф.-м.н.

\_\_\_\_\_\_\_\_ Чалый Д..Ю.

«\_\_» \_\_\_\_\_\_\_\_ 2018 г.

Студент группы ПИЭ - 31 БО

\_\_\_\_\_\_\_\_ Езжев Н.А.

«\_\_» \_\_\_\_\_\_\_\_ 2018 г.

Ярославль 2018 г.

**Содержание.**

1. **Введение.**
   1. О задаче.
   2. Постановка задачи.
   3. Используемые программные средства.
2. **Теоретическая часть.**
   1. Машинное обучение.
   2. Подходы и методы машинного обучения.
      1. Статистическая классификация
      2. Классификация на основе сходства
      3. Классификация на основе разделимости
      4. Нейронные сети
      5. Индукция правил(поиск закономерностей)
      6. Кластеризация
      7. Регрессия
      8. Алгоритмические композиции
      9. Сокращение размерности
3. **Решение задачи.**
4. **Результаты работы.**
5. **Список используемой литературы.**
6. **Введение**
   1. **О задаче**

Для исследования методов машинного обучения мной была выбрана задача “Predict future sales” выложенная на ресурсе kaggle.com. Цель задачи – определение будущих месячных продаж товаров на основе статистики продаж за несколько лет.

В ходе решения задачи предстоит проанализировать эффективность нескольких алгоритмов машинного обучения, чтобы выявить тот который лучше всего подходит для решения данной задачи.

* 1. **Постановка задачи**

Необходимо предсказать продажи в следующем месяце на основе статистики о дневных продажах товаров.

Набор данных предоставленный для решения поставленной задачи включает следующие файлы:

* **sales\_train.csv** – файл с обучающим набором данных, включает дневную статистику продаж с января 2013 по октябрь 2015 года;
* **test.csv** – тренировочный файл, содержащий индексы магазинов для которых должны быть предсказаны продажи;
* **items.csv –** файл содержит подробную информацию о товарах;
* **item\_categories.csv** – файл содержит подробную информацию о категориях товаров;
* **shops.csv** – файл содержит подробную информацию о магазинах.

Наборы данных состоят из следующего набора полей:

1. **ID** – поле-идентификатор для строки значений в таблице
2. **shop\_id** – уникальный идентификатор магазина
3. **item\_id** – уникальный идентификатор товара
4. **item\_category\_id** – уникальный идентификатор категории
5. **item\_cnt\_day –** дневные продажи товара
6. **item\_price –** стоимость товара
7. **date –** дата продажи в формате dd/mm/yyyy
8. **date\_block\_num –** номер месяца в последовательности. Январь 2013 – 0,…, Октябрь 2015 – 33
9. **item\_name –** название товара
10. **shop\_name –** название магазина
11. **item\_category\_name –** название категории товара
    1. **Используемые программные средства**

В качестве основного инструмента для анализа будет использоваться язык программирования Python с дополнительным набором библиотек. При первичной обработке данных будут использоваться библиотеки NumPy, Pandas. Для визуализации служат библиотеки seaborn и matplotlib.

При решении задачи будут использоваться модели машинного обучения, реализованные в библиотеке scikit-learn.

1. **Теоретическая часть.**
   1. **Машинное обучение**

Машинное обучение – подраздел искусственного интеллекта, изучающий методы построения обучающихся алгоритмов. Выделяют два способа обучения. Обучение по прецедентам или индуктивное обучение, основано на выявлении общих закономерностей в наборах данных. Дедуктивное обучение предполагает формализацию знаний экспертов и их перенос в компьютер в виде базы знаний. Дедуктивное обучение принято относить к области экспертных систем, поэтому термины машинное обучение и обучение по прецедентам можно считать синонимами.

* + 1. **Обучение по прецедентам**

Задача обучения по прецедентам будет звучать следующим образом дано конечное множество прецедентов (объектов, ситуаций), по каждому из которых собраны (измерены) некоторые данные. Данные о прецеденте называют также его описанием. Совокупность всех имеющихся описаний прецедентов называется обучающей выборкой. Требуется по этим частным данным выявить общие зависимости, закономерности, взаимосвязи, присущие не только этой конкретной выборке, но вообще всем прецедентам, в том числе тем, которые еще не наблюдались.

Наиболее распространенным способом описания прецедентов является признаковое описание. Фиксируется совокупность n показателей, измеряемых у всех прецедентов. Если все n показателей числовые, то признаковые описания представляют собой числовые векторы размерности n. Возможны и более сложные случаи, когда прецеденты описываются временными рядами или сигналами, изображениями, видеорядами, текстами, попарными отношениями сходства или интенсивности взаимодействия и т.д.

Для решения задачи обучения по прецедентам в первую очередь фиксируется модель восстанавливаемой зависимости. Затем вводится функционал качества, значение которого показывает, насколько хорошо модель описывает наблюдаемые данные. Алгоритм обучения ищет такой набор параметров модели, при котором функционал качества на заданной обучающей выборке принимает оптимальное значение. Процесс настройки модели по выборке данных в большинстве случаев сводится к применению численных методов оптимизации.

Существует следующая типология задач обучения по прецедентам:

* Обучение с учителем (supervised learning) - наиболее распространённый случай. Каждый прецедент представляет собой пару «объект, ответ». Требуется найти функциональную зависимость ответов от описаний объектов и построить алгоритм, принимающий на входе описание объекта и выдающий на выходе ответ. Функционал качества обычно определяется как средняя ошибка ответов, выданных алгоритмом, по всем объектам выборки.
* Обучение без учителя (unsupervised learning) - в этом случае ответы не задаются, и требуется искать зависимости между объектами.
* Частичное обучение (semi-supervised learning) занимает промежуточное положение между обучением с учителем и без учителя. Каждый прецедент представляет собой пару «объект, ответ», но ответы известны только на части прецедентов. Пример прикладной задачи — автоматическая рубрикация большого количества текстов при условии, что некоторые из них уже отнесены к каким-то рубрикам.
* Трансдуктивное обучение (transductive learning). Дана конечная обучающая выборка прецедентов. Требуется по этим частным данным сделать предсказания относительно других частных данных – тестовой выборки. В отличие от стандартной постановки, здесь не требуется выявлять общую закономерность, поскольку известно, что новых тестовых прецедентов не будет. С другой стороны, появляется возможность улучшить качество предсказаний за счёт анализа всей тестовой выборки целиком, например, путём её кластеризации. Во многих приложениях трансдуктивное обучение практически не отличается от частичного обучения.
* Обучение с подкреплением (reinforcement learning). Роль объектов играют пары «ситуация, принятое решение», ответами являются значения функционала качества, характеризующего правильность принятых решений (реакцию среды). Как и в задачах прогнозирования, здесь существенную роль играет фактор времени. Примеры прикладных задач: формирование инвестиционных стратегий, автоматическое управление технологическими процессами, самообучение роботов, и т.д.
* Динамическое обучение (online learning) может быть как обучением с учителем, так и без учителя. Специфика в том, что прецеденты поступают потоком. Требуется немедленно принимать решение по каждому прецеденту и одновременно доучивать модель зависимости с учётом новых прецедентов. Как и в задачах прогнозирования, здесь существенную роль играет фактор времени.
* Активное обучение (active learning) отличается тем, что обучаемый имеет возможность самостоятельно назначать следующий прецедент, который станет известен.
* Метаобучение (meta-learning или learning-to-learn) отличается тем, что прецедентами являются ранее решённые задачи обучения. Требуется определить, какие из используемых в них эвристик работают более эффективно. Конечная цель — обеспечить постоянное автоматическое совершенствование алгоритма обучения с течением времени.
  1. **Подходы и методы машинного обучения.**

Подход к задачам обучения — это концепция, точка зрения на процесс обучения, приводящая к набору базовых предположений, гипотез, эвристик, на основе которых строится модель, функционал качества и методы его оптимизации.

Разделение методов «по подходам» довольно условно. Разные подходы могут приводить к одной и той же модели, но разным методам её обучения. В некоторых случаях эти методы отличаются очень сильно, в других — совсем немного и «плавно трансформируются» друг в друга путём незначительных модификаций.

### **Статистическая классификация**

В статистике решение задач классификации принято называть дискриминантным анализом.

Байесовская теория классификации основана на применении оптимального байесовского классификатора и оценивании плотностей распределения классов по обучающей выборке. Различные методы оценивания плотности порождают большое разнообразие байесовских классификаторов. Среди них можно выделить три группы методов:

**Параметрическое оценивание плотности:**

* квадратичный дискриминант;
* линейный дискриминант Фишера;

**Непараметрическое оценивание плотности**

* + - метод парзеновского окна;

**Оценивание плотности как смеси параметрических плотностей**

* разделение смеси распределений, EM-алгоритм;
* метод радиальных базисных функций.

Несколько особняком стоит наивный байесовский классификатор, который может быть как параметрическим, так и непараметрическим. Он основан на нереалистичном предположении о статистической независимости признаков. Благодаря этому метод чрезвычайно прост.

**Другие теоретико-вероятностные и статистические подходы:**

* + - скрытая марковская цепь;
* байесовская сеть.

### **Классификация на основе сходства**

Метрические алгоритмы классификации применяются в тех задачах, где удаётся естественным образом задавать объекты не их признаковыми описаниями, а матрицей попарных расстояний между объектами. Классификация объектов по их сходству основана на гипотезе компактности, которая гласит, что в «хорошей задаче» схожие объекты чаще лежат в одном классе, чем в разных.

Метрические алгоритмы относятся к методам рассуждения на основе прецедентов, они способны дать ответ на вопрос «почему объект *u* был отнесён к классу *y*?». Алгоритм может дать понятный эксперту ответ: «потому, что имеются прецеденты — схожие с ним объекты, принадлежащие классу *y*», и предъявить список этих прецедентов.

Наиболее известные метрические алгоритмы классификации:

* метод ближайших соседей;
* метод парзеновского окна;
* метод потенциальных функций;

### **Классификация на основе разделимости**

Большая группа методов классификации основана на явном построении разделяющей поверхности в пространстве объектов. Из них чаще всех применяются Линейные классификаторы:

* **однослойный персептрон** — это линейный алгоритм классификации, принцип работы которого основан на модели нервной клетки - нейрона. Представляет собой пример нейронной сети с одним скрытым слоем
* **логистическая регрессия** - метод построения линейного классификатора, позволяющий оценивать апостериорные вероятности принадлежности объектов классам;
* **машина опорных векторов (SVM)** - является одной из наиболее популярных методологий обучения по прецедентам. Основная идея метода — перевод исходных векторов в пространство более высокой размерности и поиск разделяющей гиперплоскости с максимальным зазором в этом пространстве.

### **Нейронные сети**

Нейронные сети основаны на принципе коннективизма — в них соединяется большое количество относительно простых элементов, а обучение сводится к построению оптимальной структуры связей и настройке параметров связей.

* **однослойный персептрон**;
* **многослойный персептрон** – сеть прямого распространения где входной сигнал распространяется от слоя к слою. Многослойный персептрон представляет собой обобщение однослойного персептрона;
* **Нейронная сеть Кохонена** - ласс нейронных сетей, основным элементом которых является *слой Кохонена*. Слой Кохонена состоит из адаптивных линейных сумматоров. Как правило, выходные сигналы слоя Кохонена обрабатываются по правилу: наибольший сигнал превращается в единичный, остальные обращаются в ноль.;

### **Индукция правил (поиск закономерностей)**

Логические алгоритмы классификации представляют собой композиции простых, легко интерпретируемых правил.

* **решающее дерево** - метод решение задачи обучения с учителем, основанный на том, как решает задачи прогнозирования человек. В общем случае — это k-ичное дерево с решающими правилами в нелистовых вершинах (узлах) и некотором заключении о целевой функции в листовых вершинах (прогнозом). Решающее правило — некоторая функция от объекта, позволяющее определить, в какую из дочерних вершин нужно поместить рассматриваемый объект;
* **решающий лес** - это множество решающих деревьев. В задаче регрессии их ответы усредняются, в задаче классификации принимается решение голосованием по большинству;

* + 1. **Кластеризация**

**Кластеризация** - задача разбиения заданной выборки объектов (ситуаций) на непересекающиеся подмножества, называемые кластерами, так, чтобы каждый кластер состоял из схожих объектов, а объекты разных кластеров существенно отличались.

* **Алгоритм k средних (k-means)** - это метод кластерного анализа, цель которого является разделение m наблюдений на k кластеров, при этом каждое наблюдение относится к тому кластеру, к центру (центроиду) которого оно ближе всего;
* **Нейронная сеть Кохонена**;

### **Регрессия**

* **линейная регрессия** - это метод, используемый для моделирования отношений между одной независимой входной переменной (переменной функции) и выходной зависимой переменной;
* **нелинейная регрессия** - это вид регрессионного анализа, в котором экспериментальные данные моделируются функцией, являющейся нелинейной комбинацией параметров модели и зависящей от одной и более независимых переменных. Данные аппроксимируются методом последовательных приближений.;
* логистическая регрессия.

### **Алгоритмические композиции**

**Алгоритмические композиции** – класс алгоритмов машинного обучения являющихся комбинацией нескольких алгоритмов.

* **Бустинг** - это процедура последовательного построения композиции алгоритмов машинного обучения, когда каждый следующий алгоритм стремится компенсировать недостатки композиции всех предыдущих алгоритмов. Бустинг представляет собой жадный алгоритм построения композиции алгоритмов. Изначально понятие бустинга возникло в работах по вероятно почти корректному обучению в связи с вопросом: возможно ли, имея множество плохих (незначительно отличающихся от случайных) алгоритмов обучения, получить хороший;
* **Бэггинг** - это технология классификации, использующая композиции алгоритмов, каждый из которых обучается независимо. Результат классификации определяется путем голосования. Бэггинг позволяет снизить процент ошибки классификации в случае, когда высока дисперсия ошибки базового метода;

### **Сокращение размерности**

**Уменьшение размерности** - это преобразование данных очень высокой размерности в данные с гораздо меньшей размерностью, так что каждый из нижних размеров передает гораздо больше информации.

Это обычно делается при решении проблем машинного обучения, чтобы получить лучшие функции для задачи классификации или регрессии.

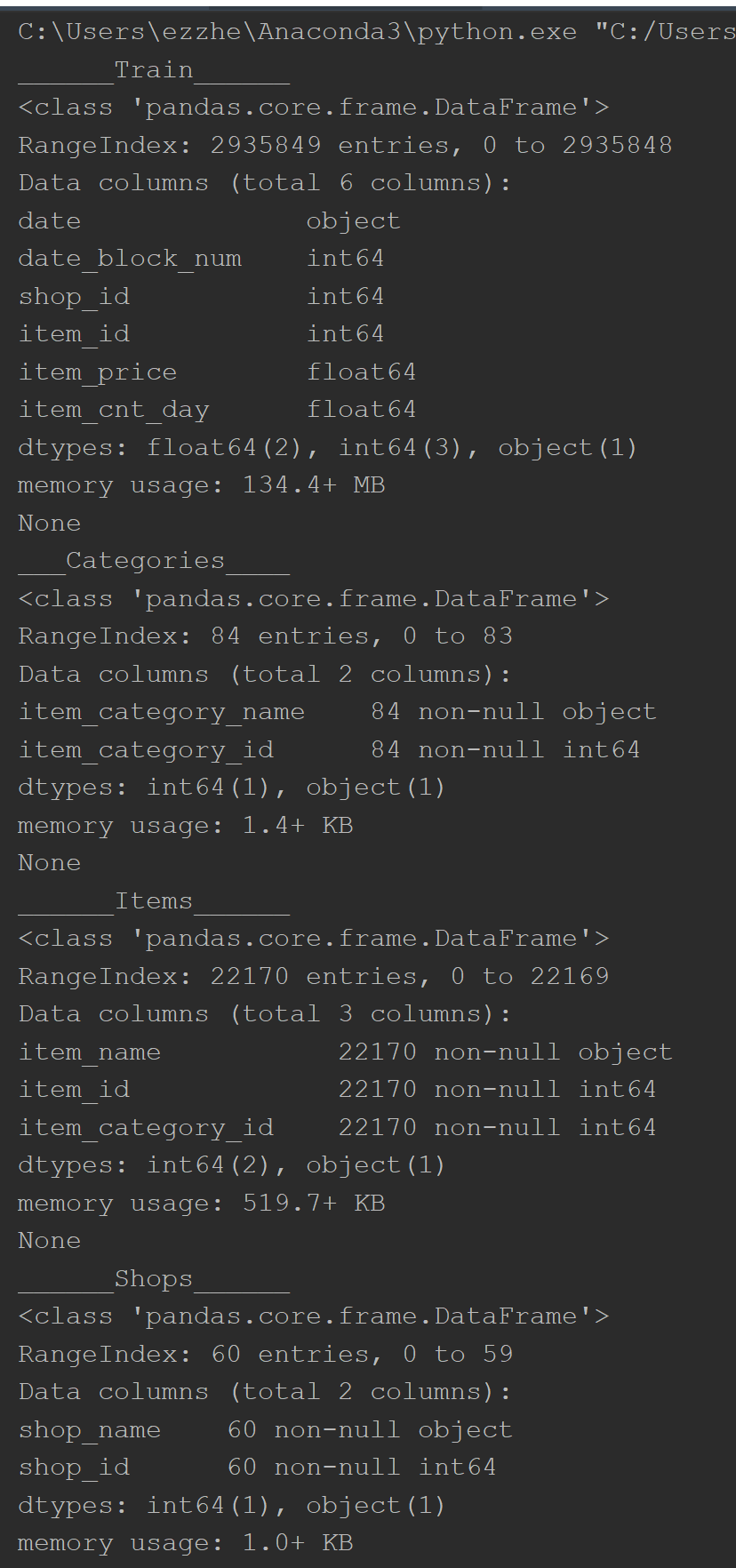
Уменьшение размерности производится с помощью следующих приемов:

* селекция признаков = отбор признаков;
* метод главных компонент;
* метод независимых компонент;
* многомерное шкалирование.
  1. **Выводы по разделу**

Раздел 2 содержит теоретическую часть посвященную машинному обучению. Он описывает основные задачи, которые решаются методами машинного обучения. Содержит описание основных методов и подходов к решению таких задач. В разделе 3 будет происходить практическое применение данных методов и алгоритмов для решения задачи.

1. **Решение задачи.**

Решение задачи “Predict future sales” заключается в определение объема товаров, которые будут проданы магазинами в следующем месяце на основе данных продаж прошлых месяцев. Процесс решения задачи “Predict future sales” можно разбить на четыре основных этапа:

* Анализ исходных данных
* Подготовка обучающей выборки
* Обучение моделей
* Анализ работы алгоритмов
  1. **Анализ исходных данных**

Первый шаг анализа исходных данных - проверка файла с тренировочной выборкой на наличие неопределенных значений (Рисунок 1). Как видим, необходимые данные не содержат пропусков и неопределенных значений.

Следующий шаг - это проверка и удаление дубликатов строк в обучающей выборке, если таковые присутствуют. Как видно по Рисунку 2, в результате в выборке было найдено и удалено 24 дублирующие строки.

Рисунок 1

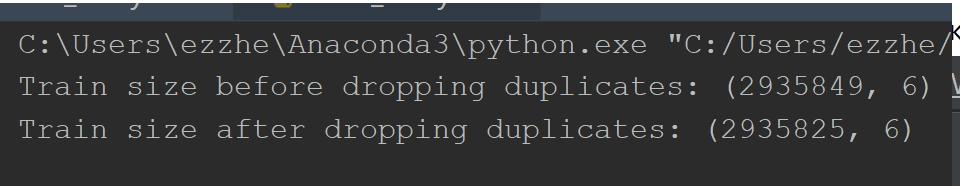


Рисунок 2

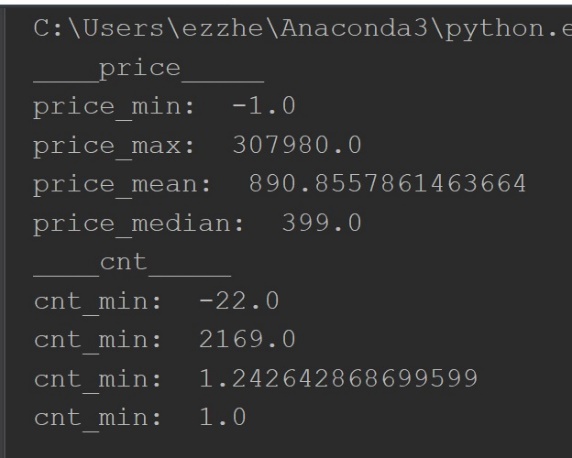
Теперь необходимо проверить не содержатся ли в выборке, какие-то нелогичные значения, выбросы. Для этого найдём минимальные и максимальные значения столбцов, а также среднее и медианное значение столбца (Рисунок 3). В результате получаем противоречивые данные по следующим столбцам:

Рисунок 3

* item\_price
* item\_cnt\_day

Очевидно, что цена на товар, как и число купленных товаров не могут быть меньше 0, скорее всего эти значения ошибочны. Так же вызывает сомнение достоверность максимального значения стоимости товаров, так как оно слишком большое.

Построим график, чтобы посмотреть является ли данное значение выбросом (Рисунок 4).

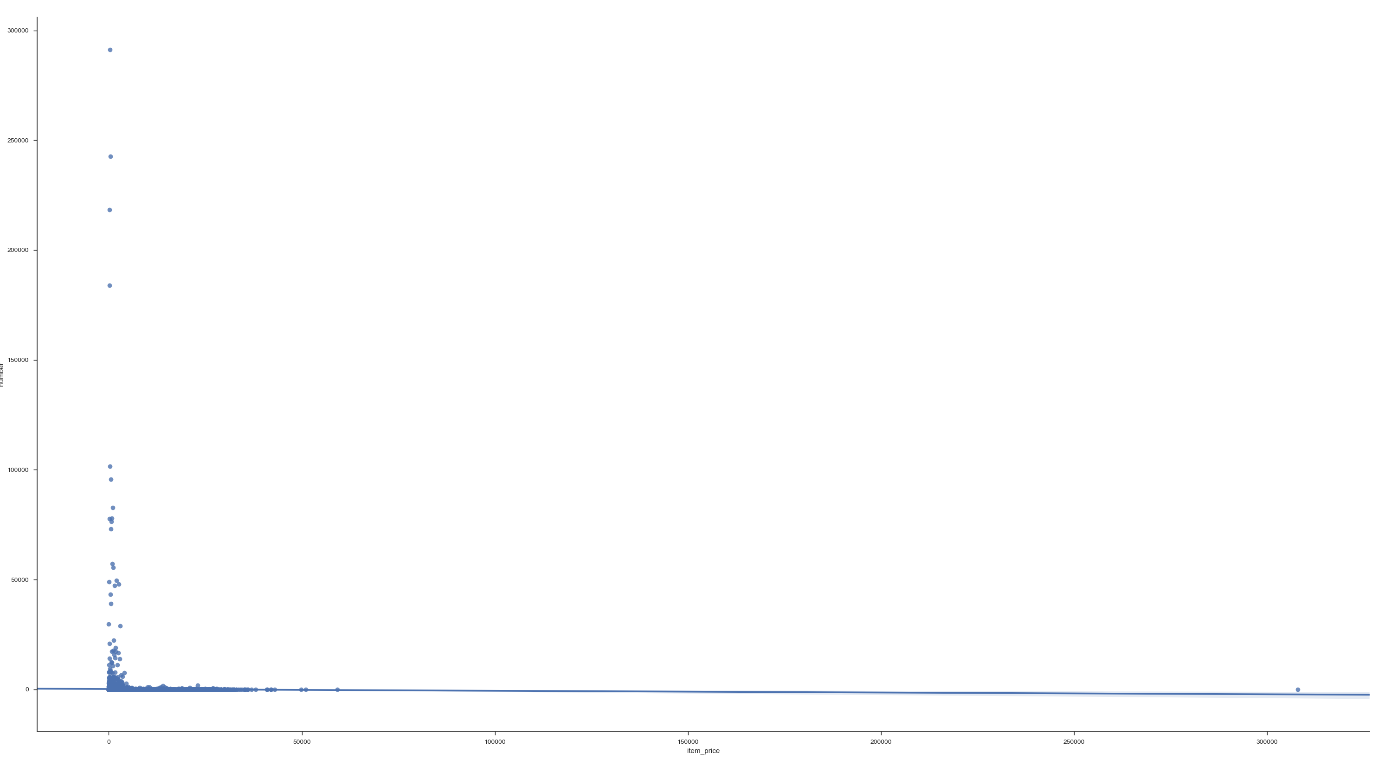


Рисунок 4

Проанализировав полученный график, можно сказать, что значение стоимости 307980 является выбросом, поэтому его можно удалить из обучающего набора данных. Так же удаляем строки, в которых значения столбцов «item\_price» и «item\_cnt\_day» меньше нуля.

Тестовая выборка содержит поле «Date», которое необходимо привести к единому определенному формату даты для дальнейшего удобства при формировании тестовых данных для моделей машинного обучения.

* 1. **Подготовка обучающей выборки**

По условию задачи необходимо предсказывать результаты месячных продаж товаров. Исходные данные содержат ежедневные объемы продаж, поэтому необходимо сформировать выборку для предсказания будущих продаж по месяцам.

Набор данных формируется из всех уникальных пар товар-магазин для каждого уникального значения колонки «date\_block\_num». Таким образом, получаем, что для каждого месяца у нас есть всевозможные сочетания товаров и магазинов.

Далее нам необходимо подсчитать, какое количество каждой единицы товара было куплено в каждом магазине за месяц и добавить эти данные в формируемую выборку. Данную операцию производим путем сложения данных по количеству проданных товаров за день, для текущих магазина (shop\_id), товара (item\_id) и номера периода (date\_block\_num).

Следующим шагом будет разбиение полей столбца «Date» на три отдельные столбца:

* month,
* year,
* days\_of\_month

и добавление их к формируемой выборке.

Последним шагом перед построением моделей будет процесс стандартизации атрибутов.

Данный процесс является общим для многих алгоритмов машинного обучения, так как большинство алгоритмов очень чувствительны к тому, что атрибуты учебной выборки представляли собой нормально распределенные данные. Так, например, многие функции, используемые в функциях алгоритма обучения (ядро опорных векторных машин или регуляризаторы линейных моделей L1 и L2), предполагают, что все атрибуты центрированы вокруг 0 и имеют дисперсию в том же порядке. В случае если атрибут имеет дисперсию, которая на несколько порядков больше, чем другие, он может доминировать над целевым атрибутом и сделать невозможным правильное обучение модели.

В данной задаче стандартизации подвергаются все поля обучающей выборки, кроме поля date\_block\_num.

* 1. **Обучение моделей**

В качестве моделей для решения данной задачи были выбраны следующие:

* **SVC** – машина опорных векторов
* **DecisionTreeClassifier –** дерево принятия решений
* **AdaBoostClassifier -** Этот алгоритм может использоваться в сочетании с несколькими алгоритмами классификации для улучшения их эффективности. Алгоритм усиливает классификаторы, объединяя их в «комитет».
* **RandomForestClassifier -** случайный лес
* **ExtraTreesClassifier**
* **GradientBoostingClassifier -** градиентный бустинг деревьев решений
* **MLPClassifier –** многослойный перцептрон
* **KNeighborsClassifier –** к - ближайших соседей
* **LogisticRegression –** логистическая регрессия
* **LinearDiscriminantAnalysis -** классификатор с линейной границей принятия решений, использующий подгонку условных плотностей данных и правила Байеса.

Обучение и тестирование моделей будет производится последовательно на обучающей выборке, разбитой на блоки с использованием алгоритма KFold. В качестве критерия качества моделей будем использовать accuracy алгоритма, то есть точность распознавания данных в тестовых наборах.

* 1. **Анализа работы алгоритмов**

Итоговый отчет по работе представлен в виде сводного графика (Рисунок 5), где по шкале X указывается точность распознавания данных обученной моделью, а по шкале Y названия тестируемых моделей и таблицы с более точными результатами.

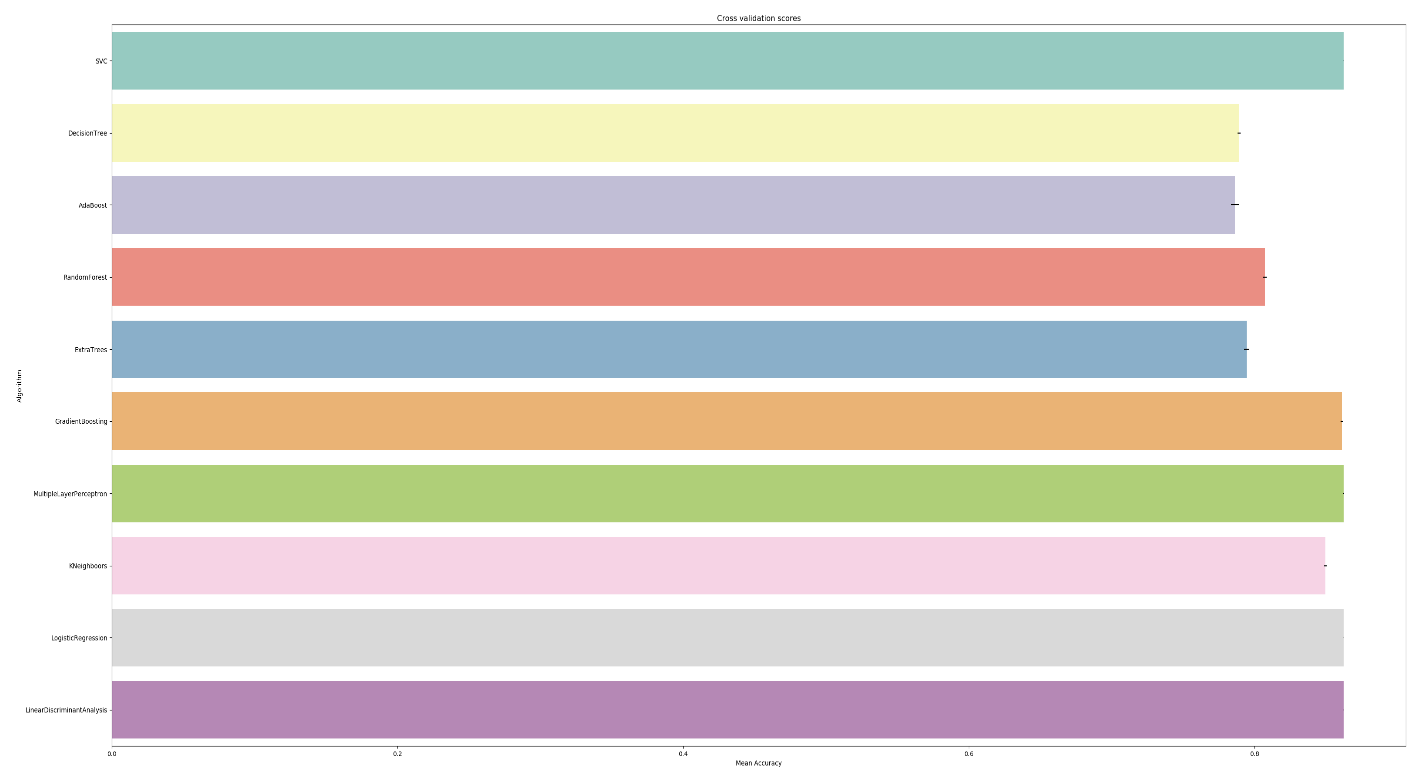


Рисунок 5

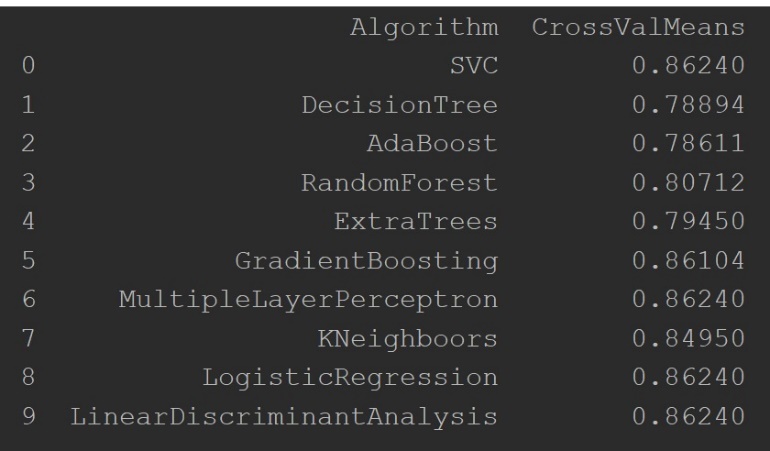


Рисунок 6

Как видим, 5 алгоритмов:

* SVC
* GradientBoosting
* MultipleLayerPerceptron
* LogisticRegression
* LinearDiscriminantAnalysis

показали наилучшие результаты с точностью распознавания данных около 86%.

Данные результаты получены на алгоритмах, которые не подвергались дополнительной настройке, путем указания каких-либо дополнительных параметров, способных иногда значительно улучшить качество обученных моделей. Отсюда следует, что данные результаты могут быть улучшены при проведении соответствующей настройки каждой из моделей, а так же путем настройки данных обучающей выборки наилучшим образом, подходящим под каждую из моделей.

1. **Результаты работы**

В ходе работы над решением задачи было выполнено следующее:

* Изучены методы и средства машинного обучения
* Изучены библиотеки python для работы со средствами машинного обучения
* Была решена задача **“Predict future sales”,** с применением моделей машинного обучения
* Проанализированы результаты работы различных алгоритмов машинного обучения, выявлены те которые решают поставленную задач наилучшим образом

1. **Список используемой литературы**
   1. <https://www.kaggle.com/c/competitive-data-science-predict-future-sales>
   2. <https://ru.coursera.org/specializations/machine-learning-data-analysis#courses>
   3. <http://www.machinelearning.ru/wiki/index.php?title=Machine_Learning#.D0.A0.D0.B5.D0.B3.D1.80.D0.B5.D1.81.D1.81.D0.B8.D1.8F>
   4. <https://ru.coursera.org/learn/vvedeniye-v-nauku-o-dannykh#>
   5. <http://scikit-learn.org/stable/documentation.html>
   6. <https://ru.wikipedia.org/wiki/Метод_опорных_векторов>
   7. <https://alexanderdyakonov.wordpress.com/2016/11/14/случайный-лес-random-forest/>
   8. <https://ru.wikipedia.org/wiki/Нелинейная_регрессия>
   9. <https://ru.wikipedia.org/wiki/Кластерный_анализ>
   10. Дьяконов, A. Г. [Анализ данных, обучение по прецедентам, логические игры, системы WEKA, RapidMiner и MatLab (практикум на эвм кафедры математических методов прогнозирования)](http://www.machinelearning.ru/wiki/images/7/7e/Dj2010up.pdf). — МАКСПресс, 2010. — 278 с.