Отчет по курсу по выбору:

Инструментарий машинного обучения и анализа данных

**“Распознавание рукописных цифр”**

Студентки 11-404 группы:

Ермолаевой К.С.

*Инструментарий машинного обучения и анализа данных*

Введение

**“Распознавание рукописных цифр”**

<https://www.kaggle.com/c/digit-recognizer>

В последнее время увеличилась активность ботов в различных социальных сетях и мессенджерах, и большинство людей задается вопросом, как их проверить. Поэтому в будущем мы хотим использовать полученные данные для разработки инструмента,который будет проверять ботов.

*Инструментарий машинного обучения и анализа данных*

Словесное описание набора данных

Наборы данных содержат в себе черно-белые изображения рукописных цифр от нуля до девяти.

Каждое изображение имеет размерность 28 на 28 пикселей, что в целом дает 784 пикселя. Каждый пиксель имеет по одному связанному с ним значению, которое обозначает оттенок пикселя: светлый или темный (часть цифры обозначают темные пиксели). Значение пикселя - целое число от 0 до 255.

Таблица тренировочных данных имеет 785 колонок. Первая колонка - это сама цифра, которая изображена на картинке. Последующие 784 колонки содержат значения пикселей изображения.

Каждый пиксель колонки тренировочной таблицы имеет имя наподобие pixelX, где X - целое число между 0 и 783. Для сопоставления значения пикселя с конкретным пикселем на изображении используется формула x = i \* 28 + j, где i и j - целые числа в диапазоне от 0 до 27, обозначающие высоту и ширину изображения, соответственно.

*Инструментарий машинного обучения и анализа данных*

Словесное описание преобразований, которые совершаются с данными, и их обоснование

Входные данные представлены в удобном для выполнения над ними предсказывающих алгоритмов виде и не требуют преобразований.

Значения пикселей тестовых изображений уже распределены по таблице, каждый пиксель в своей ячейке, соответственно. Это позволяет удобно работать со всеми входными данными и способствует обеспечению оптимальной скорости работы алгоритмов.

*Инструментарий машинного обучения и анализа данных*

Словесное описание методов машинного обучения, которые мы применяем.

Для реализации программного кода приложения и, в частности, алгоритмов машинного обучения, планируется использовать классификаторы

* **Naive Bayes**
* **Random Forest**

По окончанию реализации для обоих методов - Naive Bayes и Random Forest - будут произведены тестовые запуски программы на различных входных данных с целью выявления наиболее производительного и точного алгоритма для данной задачи - распознавания рукописных цифр.

Для данной задачи нам необходимо классифицировать каждый новый набор входных данных (картинку, представленную в виде набора пикселей) - определить его принадлежность к одному из 10 классов, отображающих цифры от 0 до 9 соответственно.

**Naive Bayes** — простой вероятностный [классификатор](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%B4%D0%B0%D1%87%D0%B0_%D0%BA%D0%BB%D0%B0%D1%81%D1%81%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D0%B8), основанный на применении [Теоремы Байеса](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BE%D1%80%D0%B5%D0%BC%D0%B0_%D0%91%D0%B0%D0%B9%D0%B5%D1%81%D0%B0) со строгими (наивными) предположениями о [независимости](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B0%D1%82%D0%B8%D1%81%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%BD%D0%B5%D0%B7%D0%B0%D0%B2%D0%B8%D1%81%D0%B8%D0%BC%D0%BE%D1%81%D1%82%D1%8C).

Достоинством наивного байесовского классификатора является малое количество данных для обучения, необходимых для оценки параметров, требуемых для классификации.

Её смысл на обывательском уровне можно выразить следующим образом. Теорема Байеса позволяет переставить местами причину и следствие. Зная с какой вероятностью причина приводит к некоему событию, эта теорема позволяет рассчитать вероятность того что именно эта причина привела к наблюдаемому событию.

Цель классификации состоит в том чтобы понять к какому классу принадлежит изображение, поэтому нам нужна не сама вероятность, а наиболее вероятный класс. Байесовский классификатор использует [оценку апостериорного максимума](http://ru.wikipedia.org/wiki/%D0%9E%D1%86%D0%B5%D0%BD%D0%BA%D0%B0_%D0%B0%D0%BF%D0%BE%D1%81%D1%82%D0%B5%D1%80%D0%B8%D0%BE%D1%80%D0%BD%D0%BE%D0%B3%D0%BE_%D0%BC%D0%B0%D0%BA%D1%81%D0%B8%D0%BC%D1%83%D0%BC%D0%B0) (Maximum a posteriori estimation) для определения наиболее вероятного класса. Грубо говоря, это класс с максимальной вероятностью. Следовательно, нам надо рассчитать вероятность для всех классов и выбрать тот класс, который обладает максимальной вероятностью.

**Random Forest** - для полученных данных алгоритм создает множество деревьев принятия решений и потом усредняет результат их предсказаний. Важным моментом тут является элемент случайности в создании каждого дерева. Ведь понятно, что если мы создадим много одинаковых деревьев, то результат их усреднения будет обладать точностью одного дерева.Предположим, у нас есть некие данные на входе. Каждая колонка соответствует некоторому параметру, каждая строка соответствует некоторому элементу данных. Мы можем выбрать, случайным образом, из всего набора данных некоторое количество столбцов и строк и построить по ним дерево принятия решений. Дальше мы можем повторить эту процедуру много-много раз и получить множество различных деревьев. Алгоритм построения дерева очень быстр. И поэтому нам не составит большого труда сделать столько деревьев, сколько будет нужно. При этом, все эти деревья в некотором смысле случайны, ведь для создания каждого из них мы выбирали случайное подмножество данных.

*Инструментарий машинного обучения и анализа данных*

Описание оценки качества нашего алгоритма

В нашем проекте реализуется алгоритм **receiver operating characteristic** или **ROC-кривая**(Кривая ошибок) – кривая, которая наиболее часто используется для представления результатов бинарной классификации в машинном обучении. Название пришло из систем обработки сигналов. Поскольку классов два, один из них называется классом с положительными исходами, второй – с отрицательными исходами. ROC-кривая показывает зависимость количества верно классифицированных положительных примеров от количества неверно классифицированных отрицательных примеров. В терминологии ROC-анализа первые называются истинно положительным, вторые – ложно отрицательным множеством. При этом предполагается, что у классификатора имеется некоторый параметр, варьируя который, мы будем получать то или иное разбиение на два класса. Этот параметр часто называют порогом, или точкой отсечения (cut-off value).

Для каждого из 10 классов будет проводиться проверка качества бинарной классификации - с помощью алгоритма ROC-кривая: принадлежит ли картинка с изображением рукописной цифры к данному классу или же нет.

*Инструментарий машинного обучения и анализа данных*

Вывод из работы

В самом начале нашего проекта мы планировали использовать данную работу для проверки ботов в социальных сетях. Исследовав данную область и все алгоритмы в этой области, мы выбрали алгоритм ROC-кривую и методы машинного обучения Naive Bayes и Random Forest. В результате мы получили работающий модуль для программы Распознавания рукописных цифр.Примерно при проверке было выявлено 3-5% ошибок. А это очень хороший результат, следовательно, наши планы осуществимы.