Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Компьютерных сетей и систем

Кафедра Информатики

ЛАБОРАТОРНАЯ РАБОТА №1

«Логистическая регрессия в качестве нейронной сети»

Магистрант: Проверил:

гр. 956241 Заливако С. С.

Шуба И.А.

Минск, 2020

**ХОД РАБОТЫ**

**Задание.**

           Данные: В работе предлагается использовать набор данных notMNIST, который состоит из изображений размерностью 28×28 первых 10 букв латинского алфавита (A … J, соответственно). Обучающая выборка содержит порядка 500 тыс. изображений, а тестовая – около 19 тыс.

           Данные можно скачать по ссылке:

* <https://commondatastorage.googleapis.com/books1000/notMNIST_large.tar.gz> (большой набор данных);
* [https://commondatastorage.googleapis.com/books1000/notMNIST\_small.tar.gz](https://commondatastorage.googleapis.com/books1000/notMNIST_large.tar.gz) (маленький набор данных);

Описание данных на английском языке доступно по ссылке:

<http://yaroslavvb.blogspot.sg/2011/09/notmnist-dataset.html>

**Задание 1.**

Загрузите данные и отобразите на экране несколько из изображений с помощью языка Python;

**Задание 2.**

Проверьте, что классы являются сбалансированными, т.е. количество изображений, принадлежащих каждому из классов, примерно одинаково (В данной задаче 10 классов).

**Задание 3.**

Разделите данные на три подвыборки: обучающую (200 тыс. изображений), валидационную (10 тыс. изображений) и контрольную (тестовую) (19 тыс. изображений);

**Задание 4.**

Проверьте, что данные из обучающей выборки не пересекаются с данными из валидационной и контрольной выборок. Другими словами, избавьтесь от дубликатов в обучающей выборке.

**Задание 5.**

Постройте простейший классификатор (например, с помощью логистической регрессии). Постройте график зависимости точности классификатора от размера обучающей выборки (50, 100, 1000, 50000). Для построения классификатора можете использовать библиотеку SkLearn ([http://scikit-learn.org](http://scikit-learn.org/)).

**Результат выполнения:**

**Задание 1.** Загрузите данные и отобразите на экране несколько из изображений с помощью языка Python.

Настроим пути к скачанным датасетам.



Рисунок 1 – Выбор пути к датасетам

Отобразим несколько изображений.

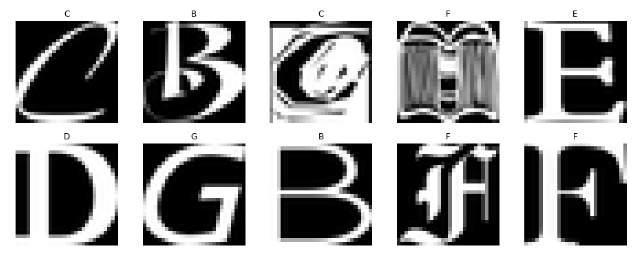
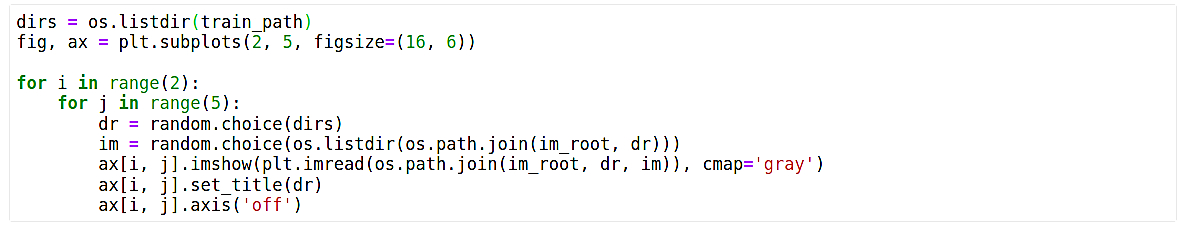


Рисунок 2 – Примеры изображений из датасета

**Задание 2.** Проверьте, что классы являются сбалансированными, т.е. количество изображений, принадлежащих каждому из классов, примерно одинаково (В данной задаче 10 классов).

Чтобы проверить сбалансированность классов, напишем функцию:



Рисунок 3 – Функция для разметки данных

Применим функцию к наборам данных:

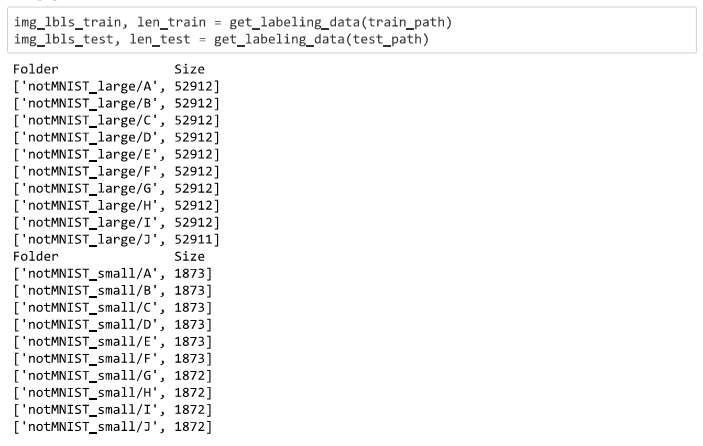


Рисунок 4 – Проверка сбалансированности классов

Видим, что все классы хорошо сбалансированы. В единственном классе «J» в «notMNIST\_large» на одно изображение меньше, чем в остальных, но это не существенное замечание. Наглядно это можно показать с помощью plt.bar. Покажем это для набора данных «notMNIST\_large».

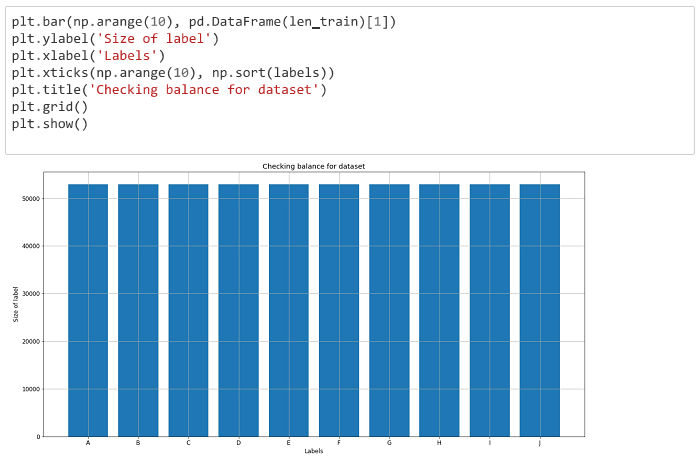
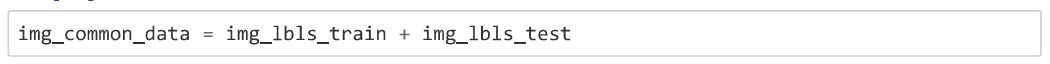


Рисунок 5 – Проверка сбалансированности с помощью функции bar

**Задание 3.** Разделите данные на три подвыборки: обучающую (200 тыс. изображений), валидационную (10 тыс. изображений) и контрольную (тестовую) (19 тыс. изображений).

Для того, чтобы задействовать максимальное число картинок для обучения, задействуем 2 набора данных. Полученный датасет рандомизируем. Далее рассортируем данные на данные для обучения (X) и их лейблы (Y).





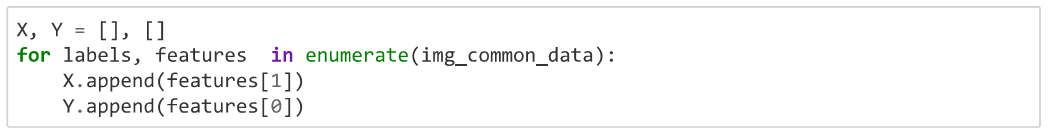


Рисунок 6 – Работа с датасетами

Чтобы каждый раз не работать с подготовкой датасета для обучения, сохраним его в удобный вид. Теперь его можно легко загрузить для выполнения последующих лабораторных работ.

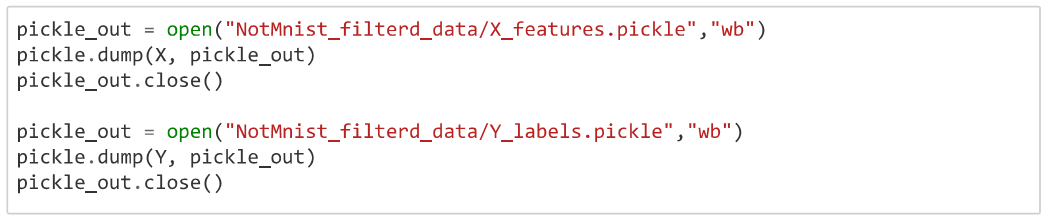


Рисунок 7 – Сохранение подготовленного датасета



Рисунок 8 – Загрузка подготовленного датасета

Нормализуем датасет путем деления каждого пикселя на его максимальное значение 255. Делаем это для того, чтобы нейронные сети и другие алгоритмы машинного обучения не работали с большими числами. Также изменим размер одного изображения как 28х28.

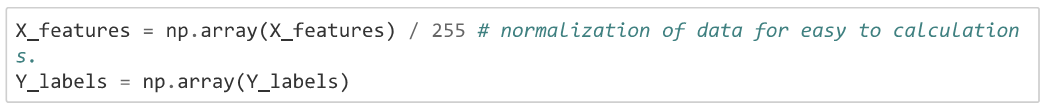


Рисунок 9 – Нормализация и изменение формы изображений в датасете

Зафиксируем константы для разделения датасета на тренировочный, валидационный и контрольный. Далее разделим датасет.

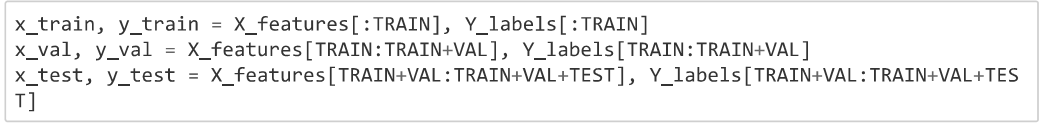


Рисунок 10 - Разделение датасета на тренировочный,

валидационный и контрольный

Результат разделения:

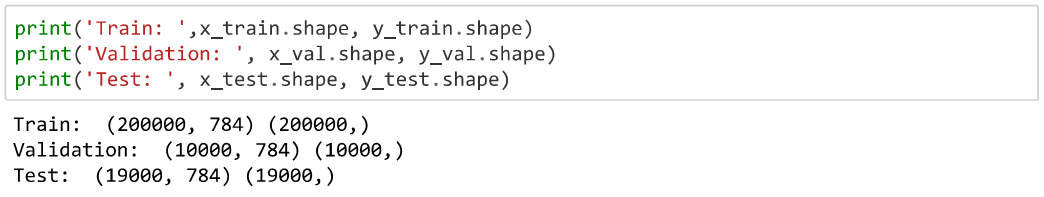


Рисунок 11 – Результат разделения

**Задание 4.** Проверьте, что данные из обучающей выборки не пересекаются с данными из валидационной и контрольной выборок. Другими словами, избавьтесь от дубликатов в обучающей выборке.

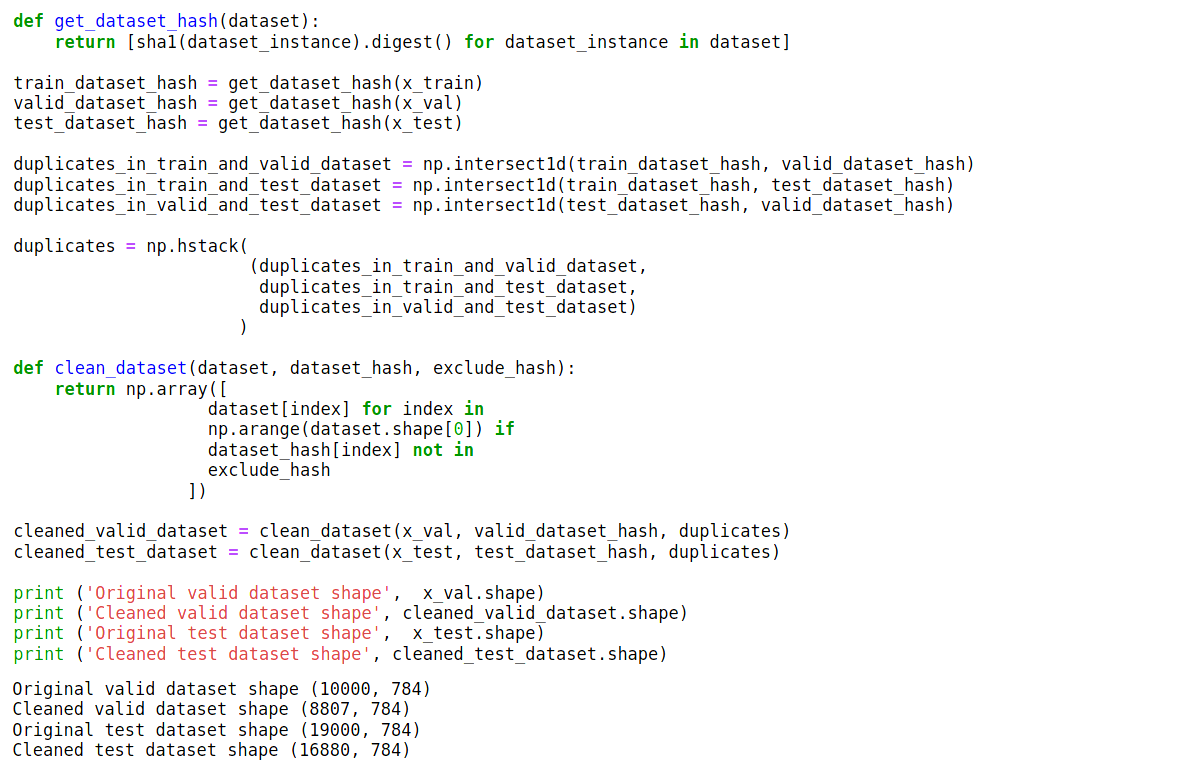


Рисунок 12 – Очистка тренировочного и контрольного

датасетов от дубликатов

**Задание 5.** Постройте простейший классификатор (например, с помощью логистической регрессии). Постройте график зависимости точности классификатора от размера обучающей выборки (50, 100, 1000, 50000). Для построения классификатора можете использовать библиотеку SkLearn ([http://scikit-learn.org](http://scikit-learn.org/)).

Построим классификатор с помощью LogisticRegression.

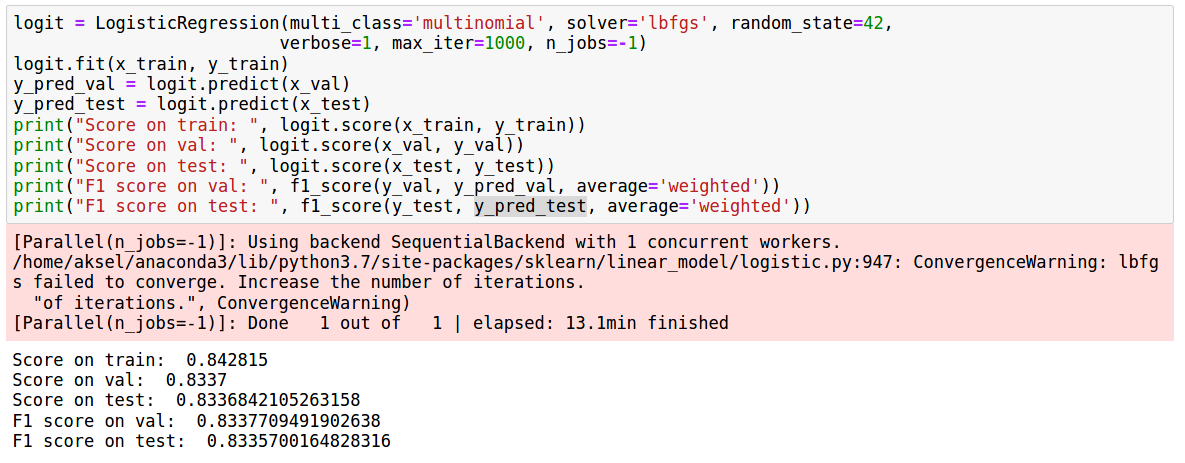


Рисунок 13 – Простейший классификатор на основе LogisticRegression

Построим confusion matrix:

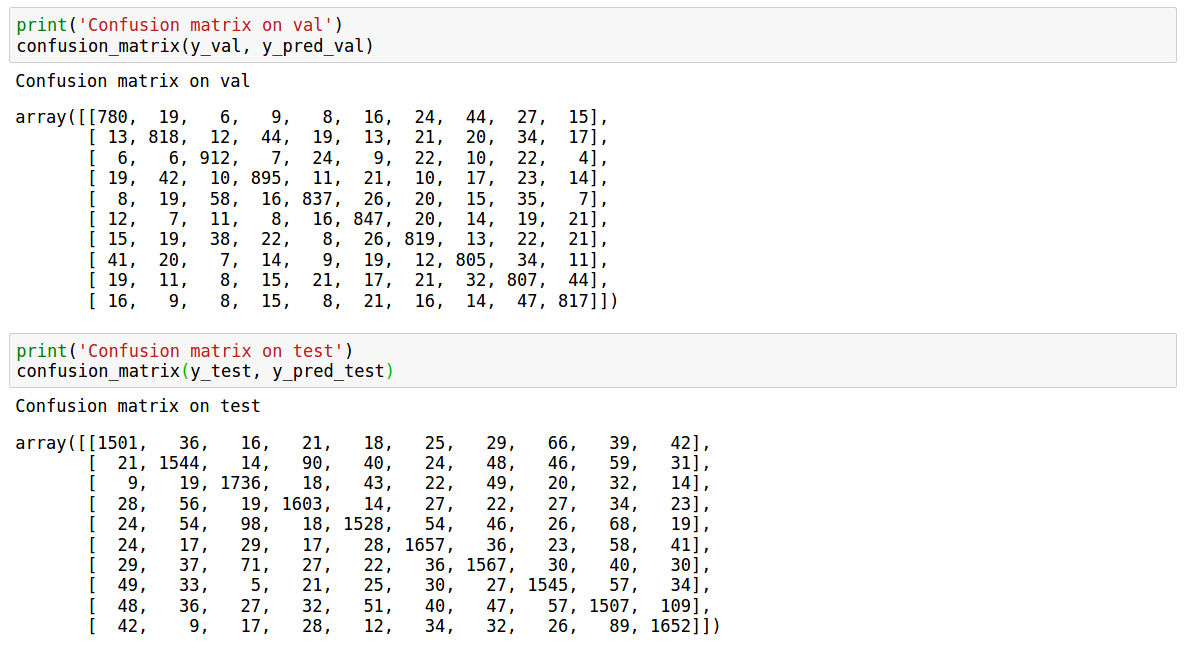


Рисунок 14 - Сonfusion matrix для валидационной и контрольной выборки

Построим график зависимости точности классификатора от размера обучающей выборки (50, 100, 1000,10000, 50000).

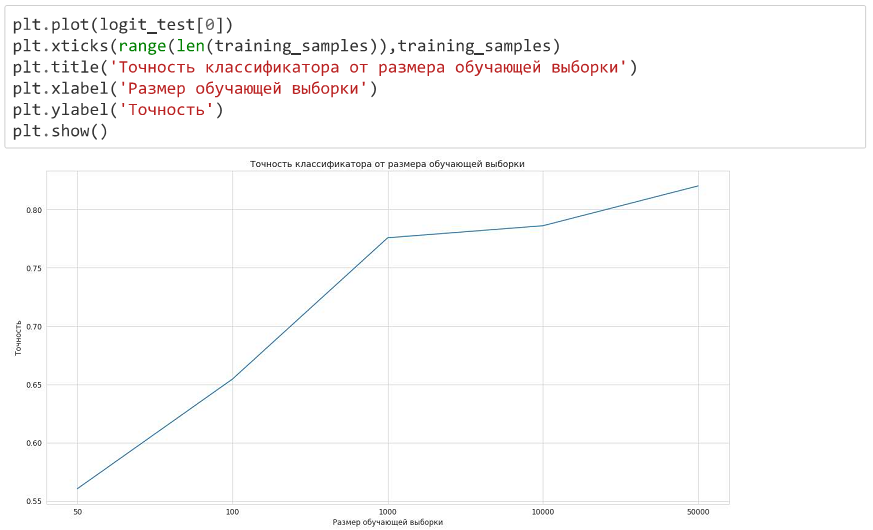
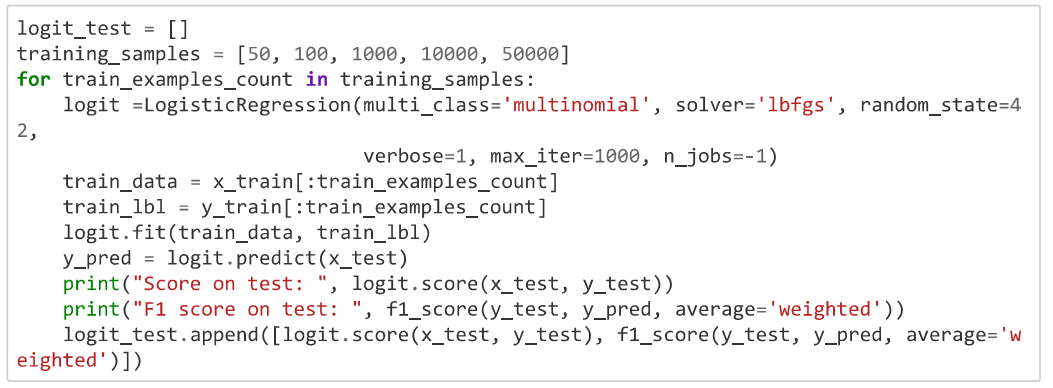


Рисунок 15 – Зависимость точности классификатора от размера

обучающей выборки для метрики Accuracy

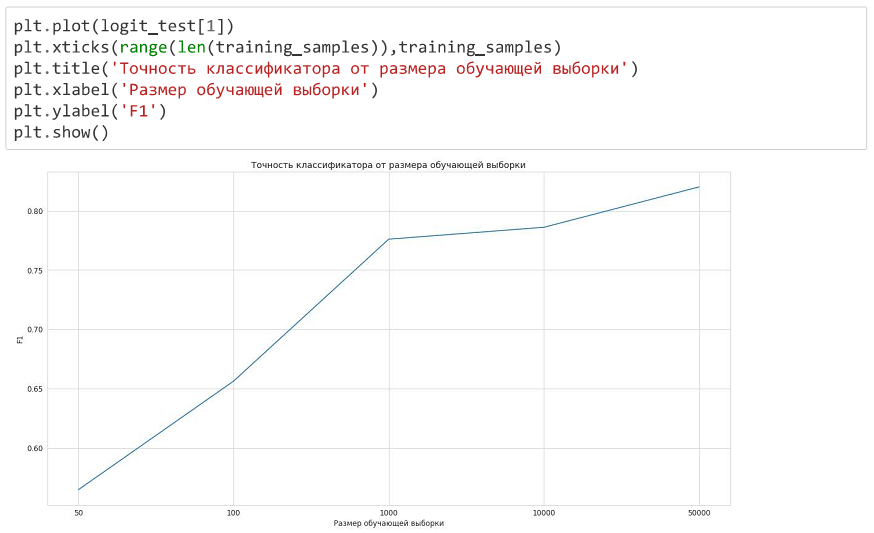


Рисунок 16 – Зависимость точности классификатора от размера

обучающей выборки для метрики F1

**Вывод:**

В ходе выполнения лабораторной работы были изучены подходы к предварительной обработке данных: проверка сбалансированности, удаление дубликатов, нормализация, изменение формы данных. Был построен простейший классификатор на основе логистической регрессии, с помощью подбора параметров были подобраны наилучшие параметры для реализации модели, при этом скор на валидационной и контрольной выборке составил примерно 0,83, а метрика F1 – 0,83 в обеих выборках. Была получена зависимость точности модели от количества данных для обучения, ее вид напоминает логарифмический рост.