МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

им. Н.Э. Баумана

Факультет «Информатика и системы управления»

Кафедра «Систем обработки информации и управления»

ОТЧЕТ

**Лабораторная работа №\_\_1\_\_**

по дисциплине«Проектирование интеллектуальных систем»

Тема: «Введение в DL»

ИСПОЛНИТЕЛЬ:                 \_\_\_Ширшов А.С.\_\_\_

ФИО

группа ИУ5-24М                  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись

"17"\_февраля\_2024 г.

ПРЕПОДАВАТЕЛЬ:             \_\_\_Канев А.И.\_\_\_\_\_

ФИО

 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись

"\_\_"\_\_\_\_\_\_\_\_\_2024г.

Москва  -  2024

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Задание

Необходимо познакомиться с фреймворком машинного обучения PyTorch и выполнить три задания:

1. Регрессия по теореме универсальной аппроксимации, ручное дифференцирование
2. Бинарная классификация с помощью автодиффиренцирования PyTorch
3. Обучить полносвязную нейронную сеть классификации 3 классов изображений из набора данных CIFAR100 по варианту из примера и затем повысить точность на тестовой выборке.

Для задания 3 нужно сформировать свою подвыборку CIFAR100 по варианту. Вы должны использовать следующие классы из CIFAR100:

1. Номер группы + 15
2. Номер варианта + 56
3. ИУ5 (Номер варианта + 21); ГУИМЦ (80); Иностранцы (90)

Полученные варианты:

1. 4 + 15 = 19
2. 14 + 56 = 70
3. 14 + 21 = 35

# Часть 1. Задача регрессии по теореме универсальной аппроксимации, ручное дифференцирование

Выборка:

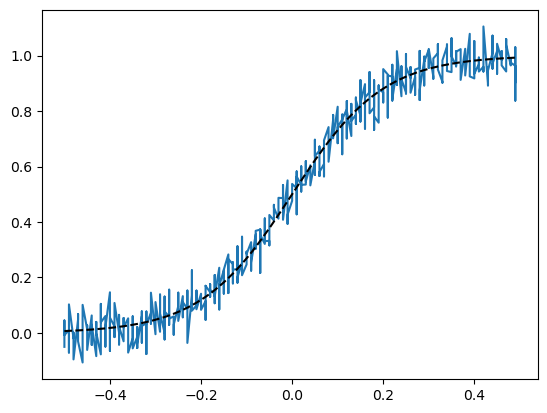


Рис. 1 – аппроксимируемая функция.

Решается задача аппроксимации функции y=11+e-10x – отмечена на рис.1 пунктиром. В значения y внесён сдвиг с ст. кв. откл. 0.05

Аппроксимация производится с помощью градиентного спуска. Функция потерь – сумма квадратов отклонения. Параметры НС: 1 вход, 1 скрытый слой из 64 нейронов, 1 выход.

На каждой итерации высчитывается производная по весам и смещению для обоих уровней НС, после чего значения сдвигаются на x\_new = x\_old - 0,0001\*grad(x).

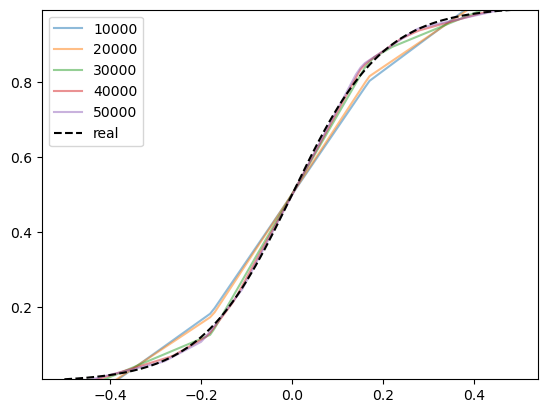


Рис. 2 – результат аппроксимации.

При прохождении 50000 итераций сумма квадратов потерь теперь составляет 0.0025866674. Результат аппроксимации указан на рис. 2.

# Часть 2. Бинарная классификация с помощью автодиффиренцирования PyTorch

Решается задача классификации. Классы отображены на рис.3. Параметры НС: 2 входа (соответственно x и y), 1 скрытый слой из 16 нейронов, 1 выход – класс.

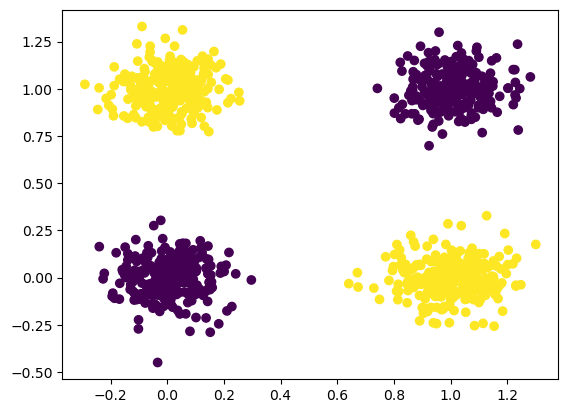


Рис. 3 – график классов.

Функция потерь – логарифмическая

Loss=-classtrue\* classpred -1-ytrue\* 1-ypred

чем дальше значение classtrue от classpred, тем ближе значение к 1. Соответственно для всего объёма значений берём сумму всех потерь.

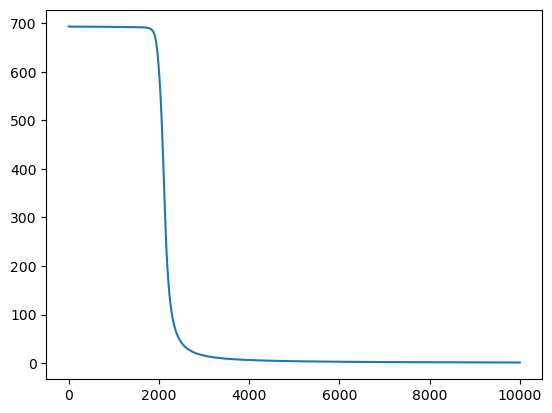


Рис. 4 – график общей функции потерь.

По рисунку видно, что после около 2.7 тысяч итераций НС начала сильно уменьшать функцию потерь, т.е. было найдено верное направление градиента. После 8000 итераций функция потерь находится у минимума. На рис.5 изображён график вывода нейронной сети в области [-0.5;0.5]

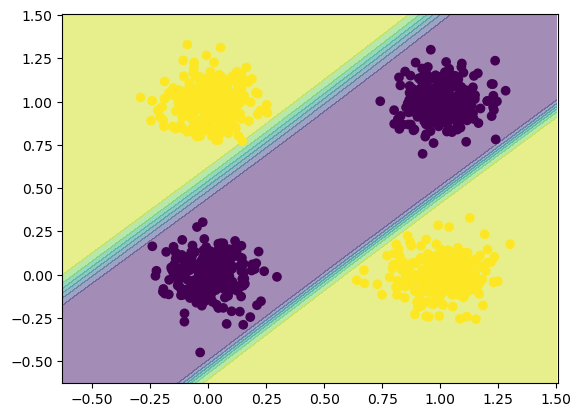


Рис. 5 – полученные области классификации.

Поскольку вначале графика потерь видим плато, можно говорить о потенциальном сокращении числа итераций до нахождения минимума, путём увеличения learning rate.

|  |  |  |
| --- | --- | --- |
| **C:\Users\ksarb\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\A21FE299.tmp** | C:\Users\ksarb\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\DB7334F5.tmp |  |
| Рис. 6 – функция ошибок с lr 1e-2 | Рис. 7 – функция ошибок с lr 2e-3 |  |

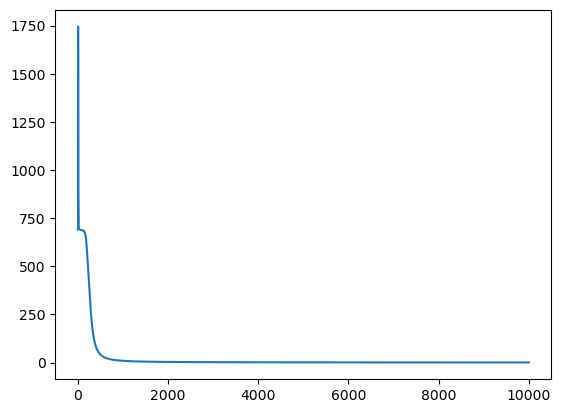


Рис. 8 – функция ошибок с lr 3e-3

Для learning rate  0,003 (рис. 8) видим почти мгновенное начало уменьшения функции ошибок, причём не попадаем в локальный минимум, наблюдаемый при lr1e-2 (рис.6), и не наблюдая плато, как на рис.7.

Проверка для lr 3e-3:

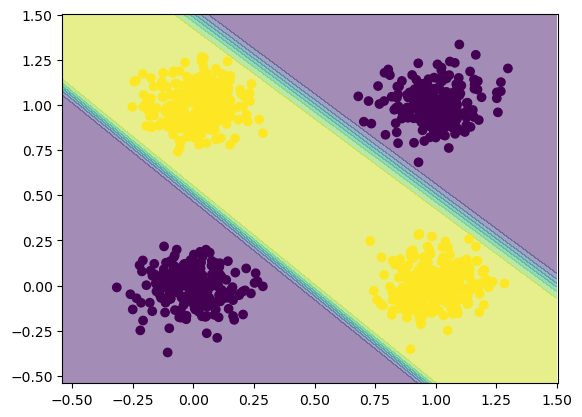


Рис. 9 – полученные области для lr 3e-3.

Сравнивая рис. 5 и рис. 9 можно наблюдать один из главных плюсов нейронных сетей – возможность поиска оптимального решения несколькими путями. Поскольку наши классы расположены симметрично относительно вращения на 180° относительно точки (0.5, 0.5), то и найденные решения также симметричны.

# Часть 3. Классификация изображений CIFAR100

Вариант: CLASSES = [19, 70, 35] - Рогатый скот, Девочки, Цветы

Взглянем ближе на классы изображений, которые должна распознать модель. Ниже представлены рисунки изображений каждого класса:



Рис. 12 – класс с рогатым скотом.



Рис. 13 – класс с девочками.



Рис. 14 – класс с цветами.

Для базового варианта нейронной сети, получаем показатели метрик, представленные в таблице 1.

Таблица 1. Показатели метрик для базовой модели

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.992 | 0.99 | 0.991 |
| Девочки | 0.994 | 0.998 | 0.996 |
| Цветы | 0.996 | 0.994 | 0.995 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7935 | 0.73 | 0.7604 |
| Девочки | 0.7037 | 0.76 | 0.7308 |
| Цветы | 0.78 | 0.78 | 0.78 |

Видим, что показатели обучающей выборки довольны высоки, однако для тестовой выборки показатели примерно равны 75%. Скорее всего происходит переобучение НС.

Проверим происходит ли переобучение модели. Действительно, минимальная ошибка на тестовых данных достигается на 49 эпохе, со значением 0.525.

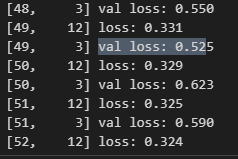


Рис. 15 – значения val loss для базовой модели.

Попробуем обучить модель до этой эпохи (49) и сравним результаты. Результаты обучения представлены в таблице 2.

Таблица 2. Показатели метрик для базовой модели с уменьшенным количеством эпох.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8887 | 0.894 | 0.8913 |
| Девочки | 0.8507 | 0.832 | 0.8413 |
| Цветы | 0.8642 | 0.878 | 0.8710 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7865 | 0.7 | 0.7407 |
| Девочки | 0.6863 | 0.7 | 0.7395 |
| Цветы | 0.7523 | 0.82 | 0.7847 |

Результаты обучающей выборки ухудшились, как и следовало ожидать, так как теперь мы приблизились к моменту переобучения. Ожидалось, также улучшение результатов на тестовой выборке, однако, итоговые показатели метрик в целом демонстрируют значения ниже.

Попробуем увеличить кол-во эпох и уменьшить скорость обучения. Поставим кол-во эпох равным 98, а скорость обучения – 0.0025. Результаты показателей указаны в таблице 3.

Таблица 3. Показатели метрик для базовой модели с lr=0.025, epoch=98.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8708 | 0.89 | 0.8913 |
| Девочки | 0.8676 | 0.826 | 0.8413 |
| Цветы | 0.8635 | 0.886 | 0.8746 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7802 | 0.71 | 0.7435 |
| Девочки | 0.6735 | 0.66 | 0.6667 |
| Цветы | 0.7568 | 0.84 | 0.7962 |

Можно заметить, что значения остались примерно равны, предыдущим.

Попробуем теперь сделать 500 эпох, со скоростью обучения 0.025. Найдем момент переобучения. Минимальная ошибка достигается при 106.

Посмотрим результаты при кол-ве эпох равным 106.

Таблица 4. Показатели метрик для базовой модели с lr=0.025, epoch=106.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8798 | 0.878 | 0.8789 |
| Девочки | 0.8874 | 0.898 | 0.8926 |
| Цветы | 0.9071 | 0.898 | 0.9025 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8 | 0.68 | 0.7351 |
| Девочки | 0.6875 | 0.77 | 0.7264 |
| Цветы | 0.7864 | 0.81 | 0.7980 |

Попробуем изменить размер батча. Сначала возьмём размер равный 50, затем 200. В таблицах ниже представлены результаты.

Таблица 5. Показатели метрик для базовой модели с lr=0.025, epoch=106. Batch = 50.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.956 | 0.956 | 0.956 |
| Девочки | 0.9434 | 0.966 | 0.9545 |
| Цветы | 0.9713 | 0.948 | 0.9595 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7553 | 0.71 | 0.7320 |
| Девочки | 0.6509 | 0.69 | 0.6699 |
| Цветы | 0.79 | 0.79 | 0.79 |

Таблица 6. Показатели метрик для базовой модели с lr=0.025, epoch=106. Batch = 200.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8462 | 0.836 | 0.841 |
| Девочки | 0.8238 | 0.832 | 0.8279 |
| Цветы | 0.8663 | 0.868 | 0.8671 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7907 | 0.68 | 0.7312 |
| Девочки | 0.6667 | 0.68 | 0.6733 |
| Цветы | 0.7768 | 0.87 | 0.8208 |

Добавим слой из 10-ти нейронов, размер батча возьмём 128. Скорость обучения поставим 0.005, а кол-во эпох 250. На таких параметрах результаты f-меры составили в среднем 75 процентов, практически приближённо к прошлым результатам. Найдём, когда произошло переобучение. Минимальное значение ошибки при 83 эпохе. Ниже в таблице 7 представлены результаты для данной модели при 83 эпохах. А затем прогоним модель на различных размерах батча.

Таблица 7. Показатели метрик для модели с 3-мя слоями с lr=0.005, epoch=83. Batch = 128.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8594 | 0.868 | 0.8637 |
| Девочки | 0.8710 | 0.878 | 0.8745 |
| Цветы | 0.9084 | 0.892 | 0.9001 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7841 | 0.69 | 0.734 |
| Девочки | 0.6577 | 0.73 | 0.6919 |
| Цветы | 0.8119 | 0.82 | 0.8159 |

Таблица 8. Показатели метрик для модели с 3-мя слоями с lr=0.005, epoch=83. Batch = 50.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.9703 | 0.98 | 0.9751 |
| Девочки | 0.8821 | 0.988 | 0.985 |
| Цветы | 0.9959 | 0.98 | 0.9879 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7363 | 0.67 | 0.7016 |
| Девочки | 0.6765 | 0.69 | 0.6832 |
| Цветы | 0.7383 | 0.79 | 0.7633 |

Возникло переобучение на 17-ой эпохи. Меняем кол-во эпох на 17 смотрим.

Таблица 9. Показатели метрик для модели с 3-мя слоями с lr=0.005, epoch=17. Batch = 50.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8309 | 0.7960 | 0.8131 |
| Девочки | 0.7385 | 0.706 | 0.7219 |
| Цветы | 0.7845 | 0.852 | 0.8169 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7791 | 0.67 | 0.7204 |
| Девочки | 0.6495 | 0.63 | 0.6396 |
| Цветы | 0.7265 | 0.85 | 0.7834 |

Таблица 10. Показатели метрик для модели с 3-мя слоями с lr=0.005, epoch=83. Batch = 200.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8164 | 0.818 | 0.8172 |
| Девочки | 0.7907 | 0.816 | 0.8031 |
| Цветы | 0.8571 | 0.828 | 0.8423 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7978 | 0.71 | 0.7513 |
| Девочки | 0.6789 | 0.74 | 0.7081 |
| Цветы | 0.8039 | 0.82 | 0.8119 |

Переобучение возникает на 79 эпохе, что достаточно близко к общему кол-ву, поэтому не будем прогонять заново на меньшем кол-ве эпох.

Таблица 11. Показатели метрик для модели с 3-мя слоями с lr=0.005, epoch=83. Batch = 200.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8164 | 0.818 | 0.8172 |
| Девочки | 0.7907 | 0.816 | 0.8031 |
| Цветы | 0.8571 | 0.828 | 0.8423 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7978 | 0.71 | 0.7513 |
| Девочки | 0.6789 | 0.74 | 0.7081 |
| Цветы | 0.8039 | 0.82 | 0.8119 |

Добавим ещё один слой на 20. Остальные параметры возьмём аналогично таблице выше. Из таблицы 12 и результатов прогона, можно отметить, что напрямую увеличение слоёв не даёт прироста напрямую, также при данных настройках не произошло переобучение. Попробуем ещё при batch=50. При таком показателе. Переобучение возникает на 38. Результаты в таблицах ниже.

Таблица 12. Показатели метрик для модели с 4-мя слоями с lr=0.005, epoch=83. Batch = 200.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7625 | 0.796 | 0.7789 |
| Девочки | 0.7643 | 0.7460 | 0.7551 |
| Цветы | 0.8102 | 0.7940 | 0.8020 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.764 | 0.68 | 0.7196 |
| Девочки | 0.6562 | 0.63 | 0.6429 |
| Цветы | 0.7043 | 0.81 | 0.7535 |

Таблица 13. Показатели метрик для модели с 4-мя слоями с lr=0.005, epoch=83. Batch = 50.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.9819 | 0.978 | 0.98 |
| Девочки | 0.9646 | 0.982 | 0.9732 |
| Цветы | 0.9817 | 0.9680 | 0.9748 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7935 | 0.73 | 0.7604 |
| Девочки | 0.6863 | 0.7 | 0.6931 |
| Цветы | 0.7736 | 0.82 | 0.7961 |

Таблица 14. Показатели метрик для модели с 4-мя слоями с lr=0.005, epoch=38. Batch = 50.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8273 | 0.862 | 0.8443 |
| Девочки | 0.8483 | 0.85 | 0.8492 |
| Цветы | 0.8996 | 0.86 | 0.8793 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8090 | 0.72 | 0.7619 |
| Девочки | 0.6937 | 0.77 | 0.7299 |
| Цветы | 0.81 | 0.81 | 0.81 |

**Исправления:**

Задание для каждой сети изменить размер батча соответственно с эпохами. При увеличении размера батча модель должна реже пересчитывать значения параметров, но с большей точностью. Так как частота пересчёта уменьшена, для сохранения кол-ва итераций, увеличим кол-во эпох и сравним.

Для сети FC(10), FC(3):

Таблица 15. Показатели метрик для модели с 2-мя слоями с lr=0.005, epoch=200. Batch = 75.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8805 | 0.884 | 0.8822 |
| Девочки | 0.8779 | 0.906 | 0.8917 |
| Цветы | 0.9212 | 0.888 | 0.9043 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.80 | 0.71 | 0.7553 |
| Девочки | 0.6667 | 0.68 | 0.6733 |
| Цветы | 0.7545 | 0.83 | 0.7905 |
|  | | | 0.7433 |

Таблица 16. Показатели метрик для модели с 2-мя слоями с lr=0.005, epoch=400. Batch = 150.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.9056 | 0.884 | 0.8737 |
| Девочки | 0.8733 | 0.91 | 0.8913 |
| Цветы | 0.9123 | 0.9360 | 0.9240 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8022 | 0.73 | 0.7644 |
| Девочки | 0.6972 | 0.76 | 0.7273 |
| Цветы | 0.81 | 0.81 | 0.8100 |
|  | | | 0.7667 |

Для сети FC(10), FC(10), FC(3):

Таблица 17. Показатели метрик для модели с 3-мя слоями с lr=0.005, epoch=200. Batch = 75.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8694 | 0.852 | 0.8606 |
| Девочки | 0.7981 | 0.846 | 0.8214 |
| Цветы | 0.8646 | 0.83 | 0.8469 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7802 | 0.71 | 0.7435 |
| Девочки | 0.6731 | 0.70 | 0.6863 |
| Цветы | 0.8 | 0.84 | 0.8195 |
|  | | | 0.75 |

Таблица 18. Показатели метрик для модели с 3-мя слоями с lr=0.005, epoch=200. Batch = 75.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8291 | 0.854 | 0.8414 |
| Девочки | 0.8324 | 0.854 | 0.8430 |
| Цветы | 0.8898 | 0.84 | 0.8642 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.814 | 0.70 | 0.7527 |
| Девочки | 0.6916 | 0.74 | 0.715 |
| Цветы | 0.8037 | 0.86 | 0.8309 |
|  | | | 0.7667 |

Для сети FC(10), FC(20), FC(10), FC(3):

Таблица 19. Показатели метрик для модели с 4-мя слоями с lr=0.005, epoch=200. Batch = 75.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7641 | 0.7580 | 0.761 |
| Девочки | 0.7720 | 0.8060 | 0.7886 |
| Цветы | 0.8548 | 0.8240 | 0.8391 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8293 | 0.68 | 0.7473 |
| Девочки | 0.7027 | 0.78 | 0.7393 |
| Цветы | 0.7664 | 0.82 | 0.7923 |
|  | | | 0.76 |

Таблица 20. Показатели метрик для модели с 4-мя слоями с lr=0.005, epoch=200. Batch = 75.

|  |  |  |  |
| --- | --- | --- | --- |
| **Train** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.8146 | 0.87 | 0.8414 |
| Девочки | 0.8004 | 0.834 | 0.8168 |
| Цветы | 0.8831 | 0.786 | 0.8317 |
| **Test** |  | | |
| Классы | **Precision** | **Recall** | **F1-score** |
| Рогатый скот | 0.7708 | 0.74 | 0.7551 |
| Девочки | 0.6952 | 0.73 | 0.7122 |
| Цветы | 0.8081 | 0.80 | 0.804 |
|  | | | 0.7567 |

Полученные результаты указаны в таб.20.

Таблица 20 – итоговая таблица.

| **Конфигурация нейросети** | **Гиперпараметры** | **Точность** | **Комментарий** |
| --- | --- | --- | --- |
| FC(10), FC(3) | lr = 0.005, batch\_size = 128, epochs = 250 | test = 75%, train = 99% | Базовый вариант |
| FC(10), FC(3) | lr = 0.005, batch\_size = 128, epochs = 49 | test = 74%, train = 87% | В базовой модели было переобучение – уменьшили количество эпох. |
| FC(10), FC(3) | lr = 0.005, batch\_size = 128, epochs = 98 | test = 74%, train = 87% | Подняли кол-во эпох в два раза. |
| FC(10), FC(3) | lr = 0.0025, batch\_size = 128, epochs = 500 | test = 73%, train = 99% | Уменьшили learning rate для повышения точности. Соответственно увеличили количество эпох |
| FC(10), FC(3) | lr = 0.0025, batch\_size = 128, epochs = 106 | test = 70%, train = 85% | Обнаружено переобучение для прошлой итерации. Снизили количество эпох до 106. |
| FC(10), FC(3) | lr = 0.0025, batch\_size = 200, epochs = 106 | test = 73%, train = 84% | Изменили размер батча в большую сторону. |
| FC(10), FC(3) | lr = 0.0025, batch\_size = 50, epochs = 106 | test = 73%, train = 99% | Изменили размер батча в меньшую сторону. Получили переобучение |
| FC(10), FC(3) | lr = 0.0025, batch\_size = 50, epochs = 83 | test = 73%, train = 99% | Изменили кол-во эпох |
| FC(10),FC(10), FC(3) | lr = 0.0025, batch\_size = 128, epochs = 83 | test = 74%, train = 88% | Добавили слой |
| FC(10),FC(10), FC(3) | lr = 0.0025, batch\_size = 50, epochs = 83 | test = 72%, train = 97% | Уменьшили Батч |
| FC(10),FC(10), FC(3) | lr = 0.0025, batch\_size = 50, epochs = 17 | test = 72%, train = 88% | Переобучение. Уменьшили кол-во эпох |
| FC(10),FC(10), FC(3) | lr = 0.0025, batch\_size = 200, epochs = 83 | test = 74%, train = 88% | Пробуем большой батч |
| FC(10),FC(20), FC(10), FC(3) | lr = 0.0025, batch\_size = 200, epochs = 83 | test = 74%, train = 88% | Добавил слой. Batch 200 |
| FC(10),FC(20), FC(10), FC(3) | lr = 0.0025, batch\_size = 50, epochs = 83 | test = 75%, train = 99% | Изменили размер батча |
| FC(10),FC(20), FC(10), FC(3) | lr = 0.0025, batch\_size = 50, epochs = 38 | test = 77,6%, train = 88% | Изменили размер эпох |

**Вывод:** при обучении нейронных сетей очень важны гиперпараметры. Уменьшение learning rate позволяет модели найти минимум более точно, но для этого требуется обычно более большое кол-во эпох. При увеличении размера батча точность шагов увеличивается, однако, для сохранения кол-ва итерации необходимо также увеличить кол-во эпох. Эксперименты показали, что при таком случае результаты модели не ухудшаются, иногда даже становятся немного лучше. Также очень важно грамотно определить момент наступления переобучения, и отталкиваться в зависимости от него. Дополнительно, добавлять слои не приносит пользы напрямую, необходимо снова менять гиперпараметры, искать переобучение. Все обученные модели показывают precision на классе девочек явно хуже, чем на других. Это может быть связано с тем, что другие картинки могут содержать признаки, схожие с признаками картинок про девочек – например яркий цветок может ассоциироваться с платьем. Recall – метрика, показывающая, какую долю класса из всех объектов класса нашла модель. На многих экспериментах recall для рогатого скота меньше по сравнению с другими классами. Выходит коров чаще определяют как другой класс, может коричневый цвет коров, наличие ушей, может их сделать похожими на девочек для модели. Это тоже объясняет уменьшение precision для девочек, и recall для коров.



Рисунок 2 – Пример коровы, которую можно спутать с девочкой из-за бледного лица, ушей и коричневой кожи, или же с цветами из-за наличия зелени.