МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
Национальный исследовательский ядерный университет «МИФИ»

Институт интеллектуальных кибернетических систем

КАФЕДРА КИБЕРНЕТИКИ

БДЗ

по курсу "Теория нейронных сетей"

студента группы Б17-504

Вариант № 20

Оценка: _____

Подпись: _____

2020 г.

ОТЧЕТ № 1

по теме «Многослойные нейронные сети»

Вариант №20

ФИО студента Ярабаева Александра Павловна **Группа** Б17-504

Оценка:_____ Подпись:_____

Результаты обучения многослойной нейронной сети:

Метод обучения	Параметры метода обучения	Число эпох обучения	Ошибка на обучающей выборке, $E_{обуч}$	Ошибка на тестовой выборке, $E_{тест}$
GD	$\alpha = 0.1$	42	0.273	0.277
GDM	$\mu = 0.3$	34	0.274	0.277
NAG	$\mu = 0.3$	38	0.273	0.281
Fletcher-Reeeves		35	0.331	0.304
Polak-Ribiere		58	0.271	0.281
AdaGrad		78	0.231	0.253
RMSProp	$\rho = 0.85$	9	0.277	0.292
AdaDelta	$\rho = 0.85$	200	0.276	0.276
Adam	(0.9 , 0.999)	22	0.264	0.261
LM		1	0.261	0.261
BFGS		96	0.219	0.2325
Stochastic GD	Mini batch = 100	45	0.284	0.286

Выводы:

ОТЧЕТ № 2

по теме «Решение прикладных задач обработки данных на нейронных сетях»

Вариант №20

ФИО студента Ярабаева Александра Павловна **Группа** Б17-504

Оценка:_____ Подпись:_____

Показатели качества обученной нейросетевой модели:

Выводы:

I. Исходные данные

1.1. Описание исходных данных

Название набора данных – набор электронных писем, которые могут быть как спамом, так и нет. Спамом называется навязчивая, ненужная информация. Например, реклама продуктов/веб-сайтов, экономические пирамиды, письма-пересылок.

Коллекция данных является сбором различных частотных статистик по словам и буквам собранных из писем, с отметкой является данное письмо спамом или нет. Спам письма размечены с помощью почтового - офиса, используемого в компании Hewlett-Packard, а также с помощью ручного труда. Не-спам письма состоят из рабочей переписки компании и личных переписок (слова ‘george’ и ‘650’ являются индикаторами не-спам письма).

Объем выборки: 4601(спам = 39.4%(1813), не спам = 60.6%(2788))

Число признаков: 58 (57 - вещественные, 1 номинальный - метка класса)

Ссылка на источник: https://archive.ics.uci.edu/ml/datasets/Spambase

Пропуски и повторы в данных: нет

Список используемых слов и символов, статистика по которым считается в приложении №1.

Описание атрибутов:

- 48 вещественных атрибутов, описывающих частоту слова среди всех слов в письме (слова - любые последовательности цифр и букв);
- 6 вещественных атрибутов, описывающих частоту символа среди всех символов в письме;
- 1 атрибут, показывающий среднюю длину последовательностей, состоящих из заглавных букв в письме;
- 1 атрибут, показывающий максимальную длину последовательностей, состоящих из заглавных букв в письме;
- 1 атрибут, показывающий полное количество заглавных букв в письме;
- 1 атрибут, показывающий к какому классу (спам или не-спам) принадлежит письмо.

Необходимо решить задачу классификации писем на два класса: спам и не спам, то есть необходимо решить задачу выявления спам-писем среди всего потока писем. В качестве входной информации система должны брать 57 атрибутов, описанных выше, и выдавать результат принадлежности письма к классу (0 - не спам, 1 - спам).

1.2. Визуальный анализ исходных данных

а) Гистограммы распределения и диаграммы Box-and-Whisker

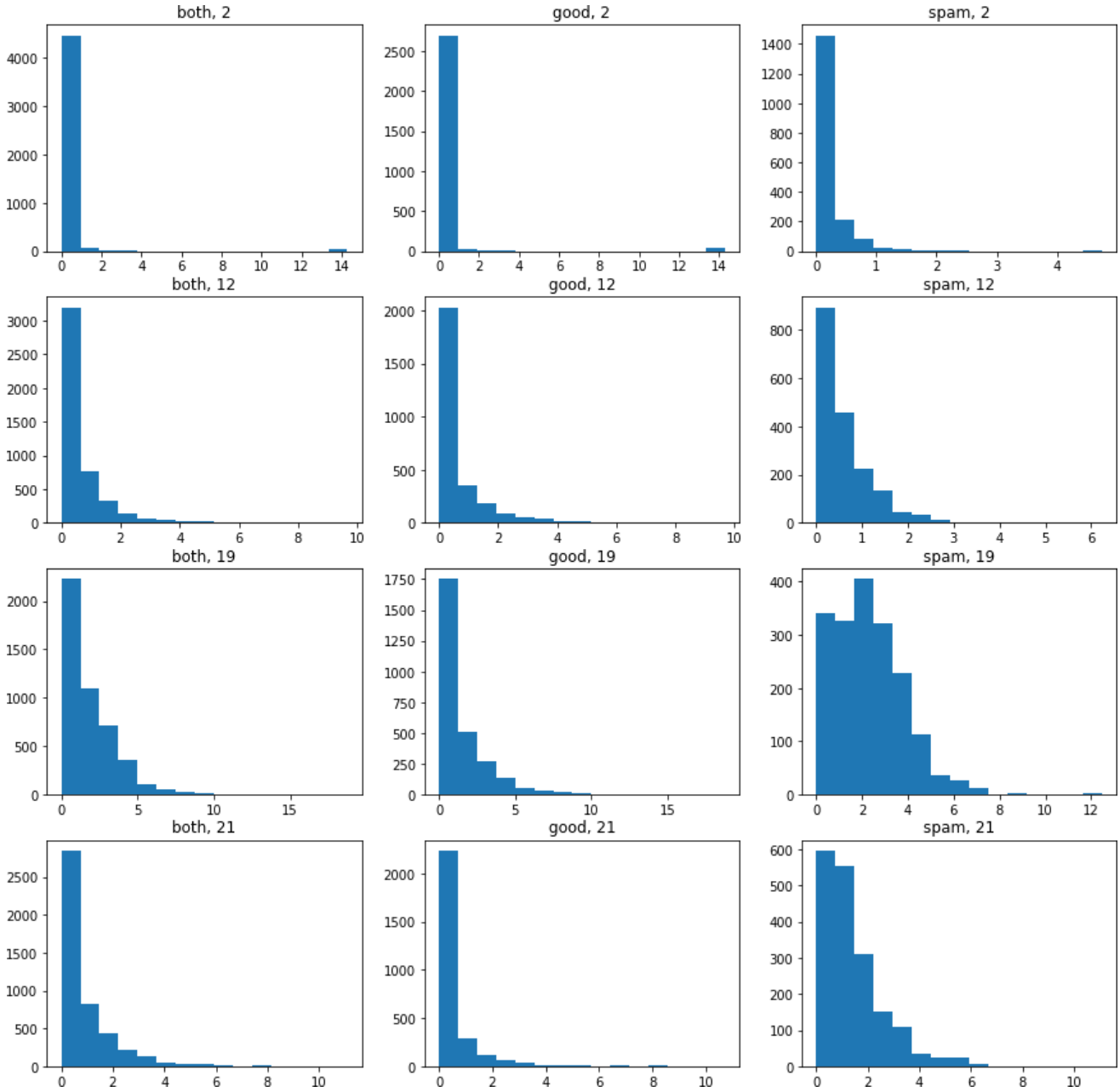


Рисунок 1 – Гистограммы для 2, 12, 19, 21 признаков.

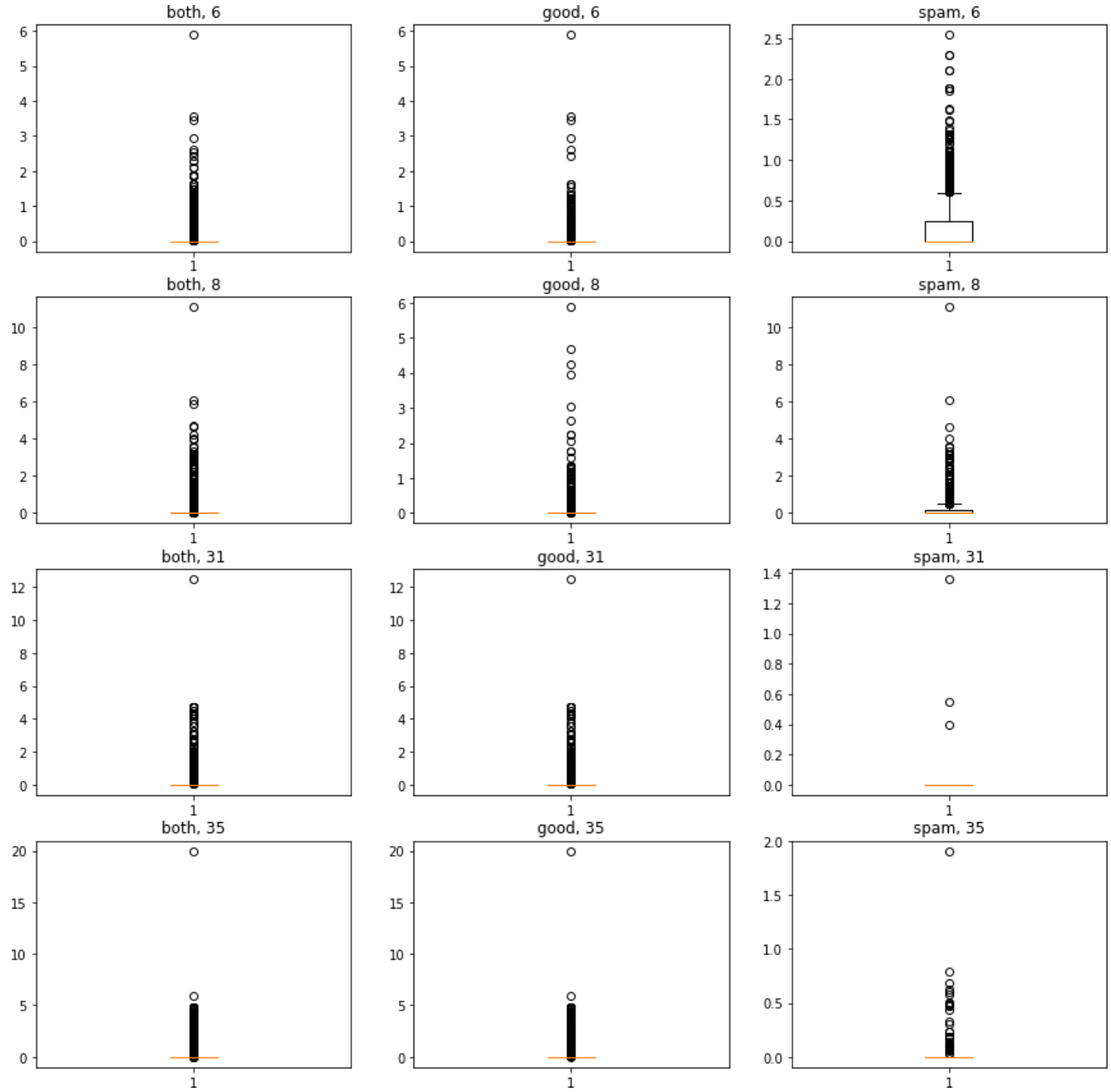


Рисунок 2 – Диаграмма «Box-and-Whisker» для 6, 8, 31, 35 признаков.

Анализ гистограмм всех атрибутов показал, что данные имеют экспоненциальное распределение и среднее значение у большинства из них в нуле с малой дисперсией. Исходя из рис.1 можно сделать вывод, что атрибуты в случае спам письма и не-спам письма распределены по-разному. По рис.2 можно заключить, что статистики в категориях спам и хороших писем отличаются, также в выборке есть выбросы и они не привязаны к какому-либо классу, то есть выбросы есть и среди спама, и среди не спама.

б) Корреляционная матрица признаков

Визуализировать корреляционную матрицу признаков (использовать heatmap), сделать выводы.

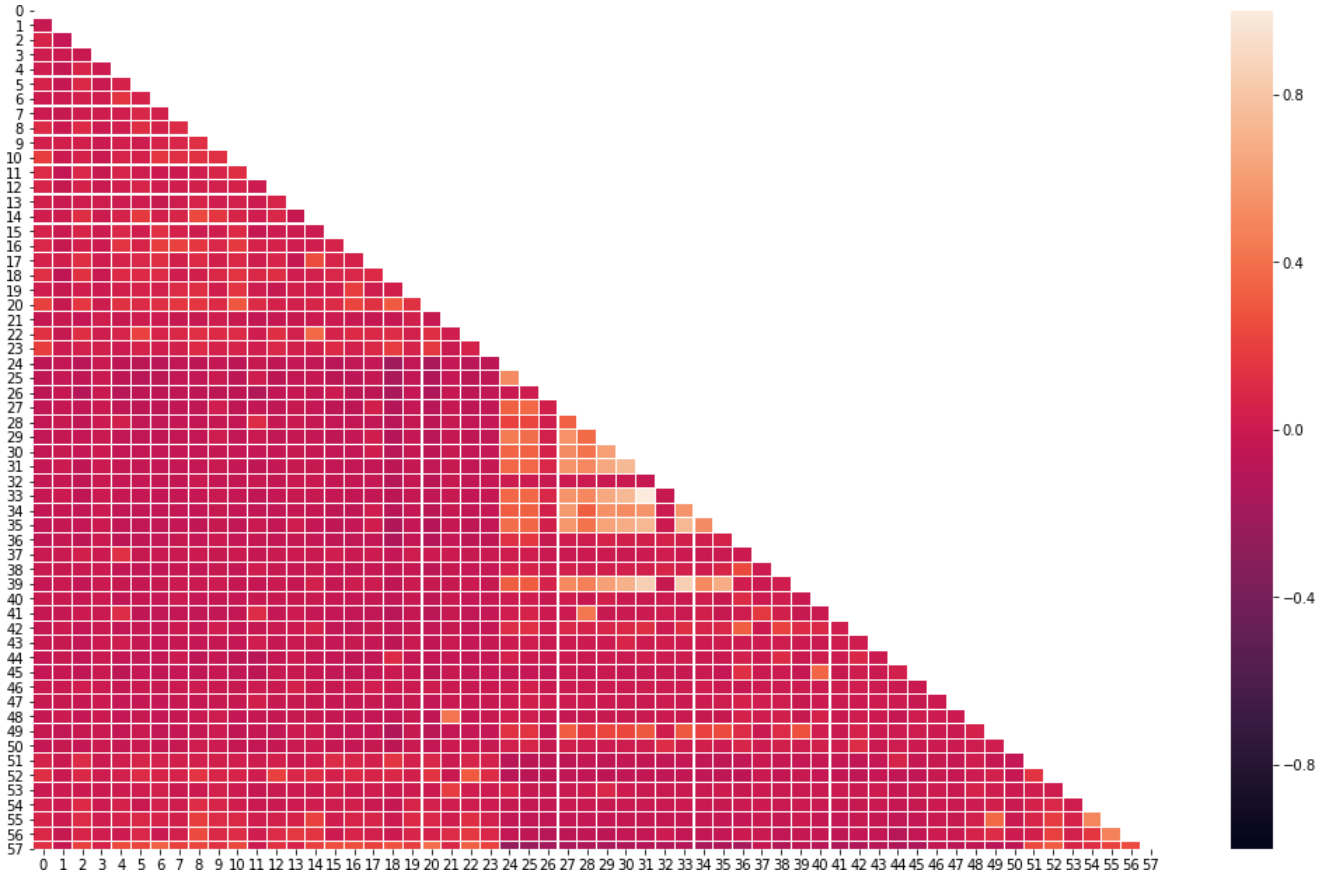


Рисунок 3 - Корреляционная матрица признаков

По корреляционной матрице на рис.3 можно сделать вывод о том, что набор данных несильно коррелирован. В выборке данных имеются атрибуты, которые сильно коррелируют между собой. Например, атрибут #31 и #39 сильно коррелируют между собой и им соответствуют частоты встречаемости слов ‘telnet’ и ‘pm’, но это вполне логично и можно рассматривать наличия двух этих атрибутов, как излишки.

в) Диаграммы рассеяния

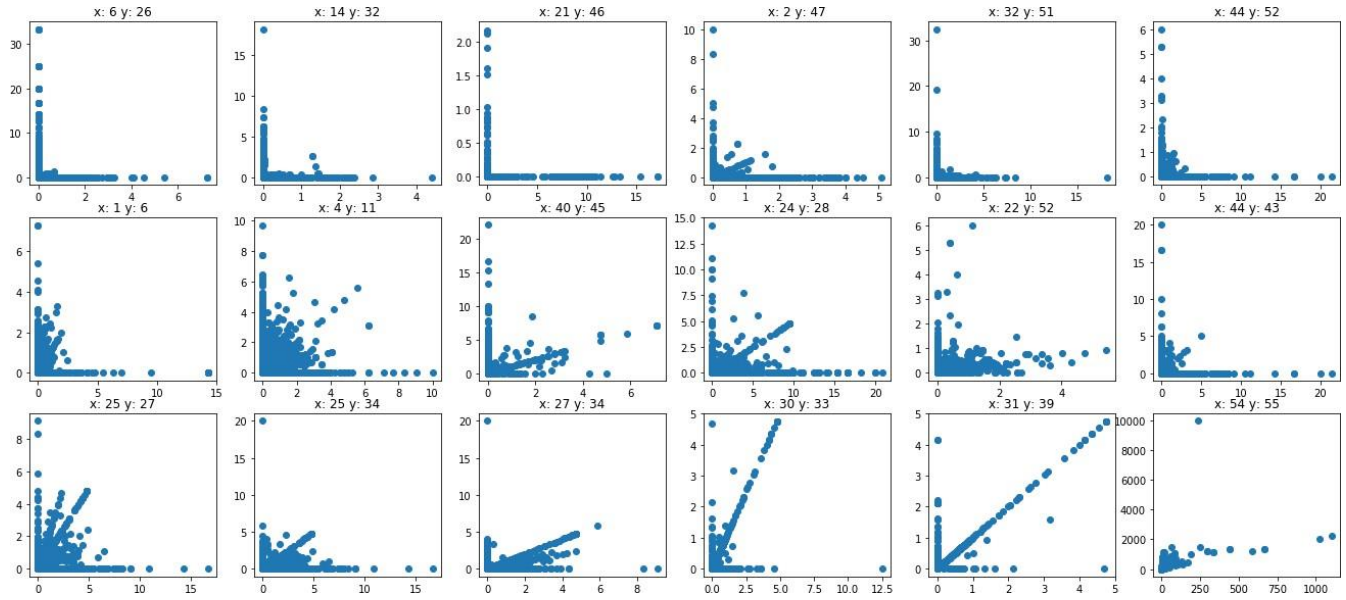


Рисунок 4 – Диаграммы рассеивания

Для составления диаграмм рассеивания пары атрибутов выбирались случайно. На первой строке представлены данные с низким уровнем корреляции, на второй – со средним, на третьей – с высоким. Анализируя получившиеся диаграммы, можно увидеть, что данные были взяты из одной генеральной совокупности (нет кластеров точек находящийся на значительном расстоянии друг от друга), а также, что высокий коэффициент корреляции не всегда соответствует линейной связи между признаками (например, для признаков 25 и 34).

1.3. Выводы

По результатам предварительного визуального анализа исходных данных статистики для разных классов (писем спам и не-спам) отличаются, из чего можно сделать вывод о возможности построения адекватного классификатора. Полученные гистограммы и диаграммы показали, что данные необходимо подготовить для обучения, так как у большинства атрибутов среднее значение находится в нуле, поэтому вероятно значительное число элементов выборки будут содержать атрибуты со значением нуль; число анализируемых атрибутов довольно велико и между некоторыми из них наблюдается корреляция, но полагаться только на коэффициент корреляции будет нецелесообразно, так как присутствуют примеры ошибочно высокого коэффициента корреляции.

II. Предобработка данных

2.1. Очистка данных

а) Обнаружение и устранение дубликатов

Для устранения дубликатов использовалась библиотека Pandas и встроенная в нее функция drop_dublicates(), которая находит все дубликаты и устраняет все дублирующиеся строки кроме первого вхождения. После ее применения к набору данных количество строк уменьшилось и стало равняться 4210, что свидетельствует о том, что данные из некоторых писем заносились в базу несколько раз.

б) Обнаружение и устранение выбросов

Исследуемые признаки не распределены по нормальному закону, поэтому использование методов, которые основываются на среднем и дисперсии не подходят. Также возможно ручное выделение выбросов с помощью Box-and-Whisker, однако это занимает много времени.

Для выявления был выбран метод, который основан на методе машинного обучения: метод одноклассовой классификации на основе SVM с гауссовским ядром. Производится обучение на отличие представителей класса от не представителей и подсчет расстояний до разделяющей плоскости у элементов выборки (положительное если расстояние уходит внутрь класса, отрицательно если из класса), после чего подсчитывается необходимый квантиль и устраняется все что ниже этого значения.

Этим методом с квантилем (5%) было выявлено 211 выбросов, что является 5.01% от всего количества выборки.

в) Пропущенные значения

В данных нет пропущенных значений, об этом говорится в паспорте данных. И после проверки это подтвердилось.

г) Визуальный анализ очищенных данных

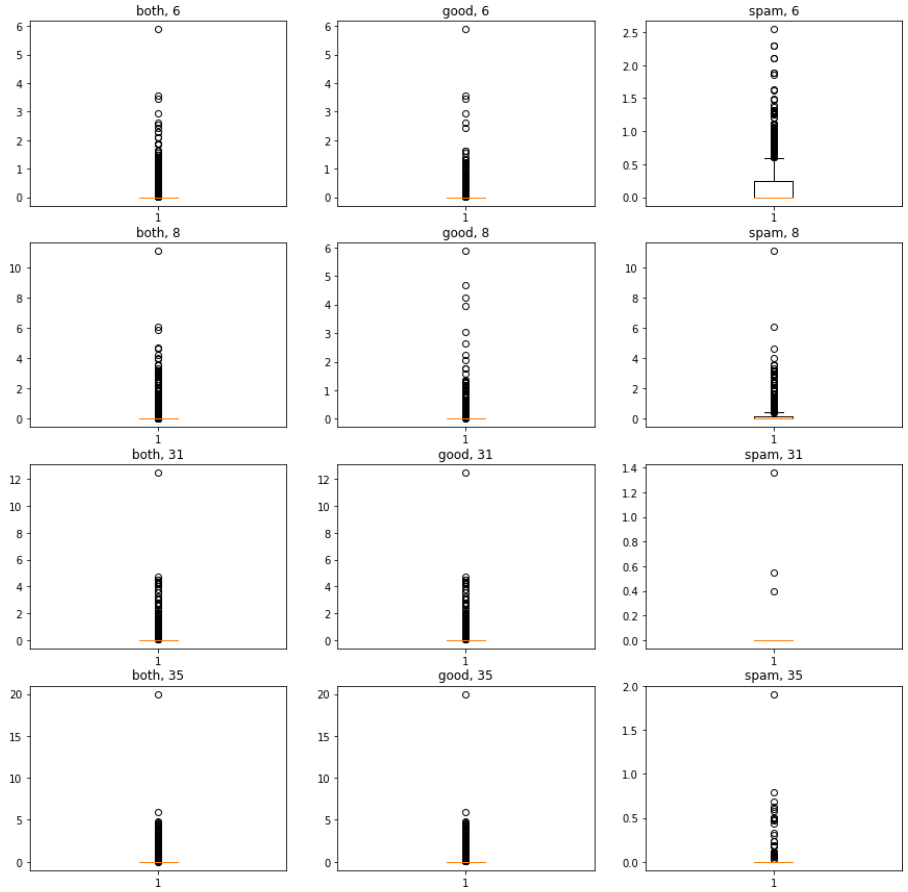


Рисунок 5 – Box-and-Whisker после предобработки

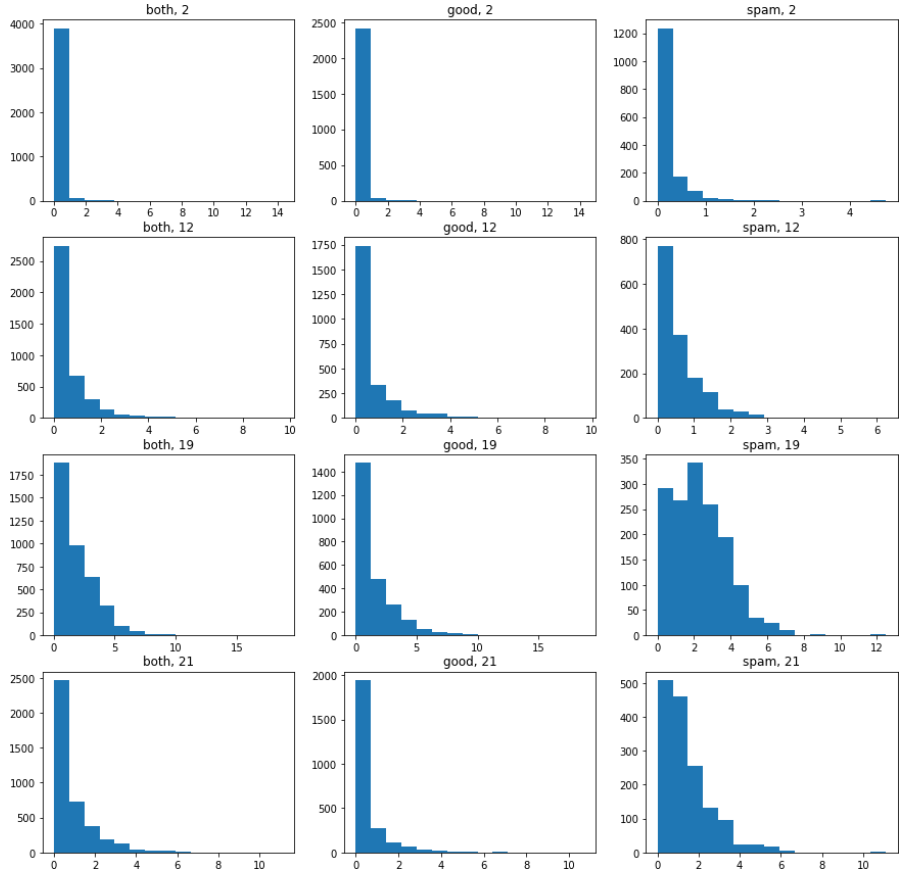


Рисунок 6 – Гистограммы после предобработки

Визуальный анализ диаграмм данных после предобработки и диаграмм из пункта 1.2 показал, что большинство выбросов были устранены.

д) Выводы

Основными недостатками в данных были выбросы и дубликаты. Путем визуального анализа диаграмм Box-and-Whisker всех атрибутов при разных значениях убираемой доли выборки (1%, 3%, 5% и 10%) было принято решение устранить выбросы на уровне 5%, который показывает хорошие результаты в очистке от выбросов и при этом не убирает элементы выборки с адекватно большими значениями атрибутов. Гистограммы стали более отчетливо походить на экспоненциальное распределение, а на диаграммах Box-and-Whisker незначительных выбросов не наблюдается.

2.2. Преобразование данных

а) Преобразование входов

Данные предлагаемой выборки распределены экспоненциально и применение к ним нормализации не имеет смысла, так как часть значений уйдет в минус, что противоречит их природе. Поэтому в работе применяется преобразование в [0.1, 0.9] отрезок с запасом 0,1 с каждой стороны, чтобы обеспечить выходы вне диапазона. Параметров нет.

б) Преобразование выходов

Преобразование выходов не требуется, так он выход либо 0, либо 1.

в) Визуальный анализ преобразованных данных

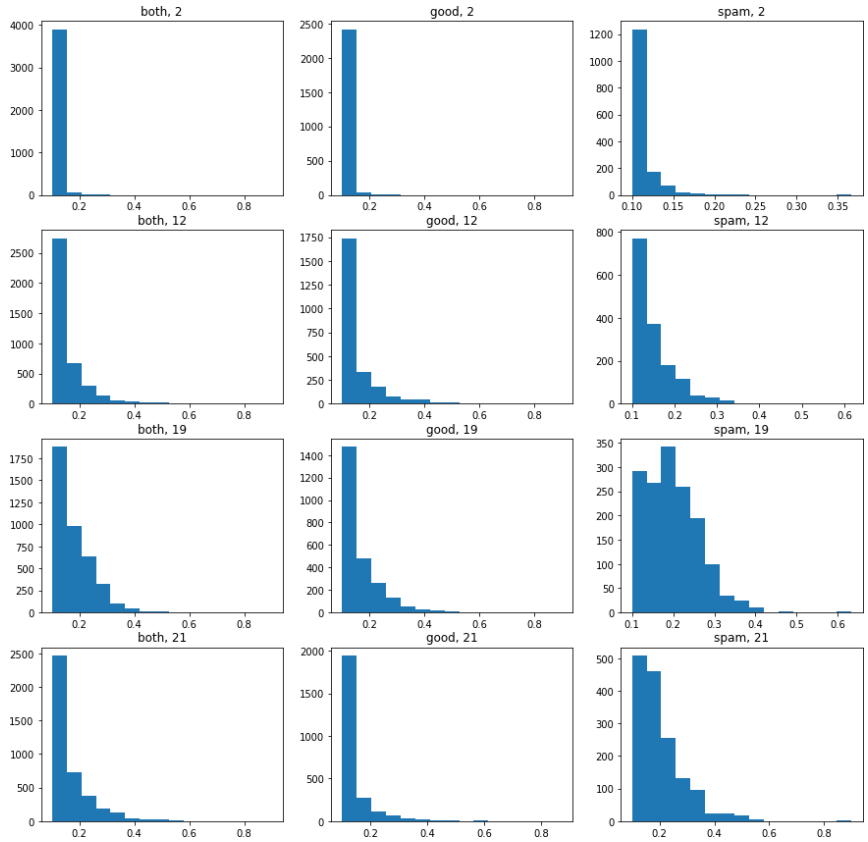


Рисунок 7 – Гистограммы после преобразования

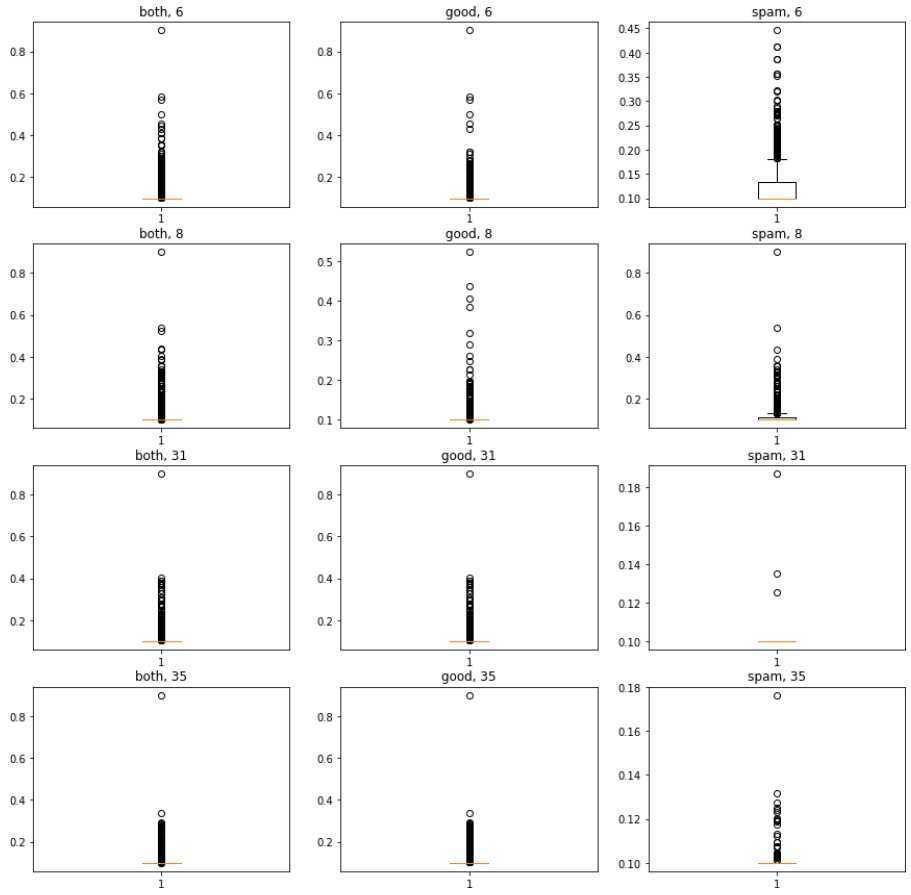


Рисунок 8 – Box-and-Whisker после преобразования

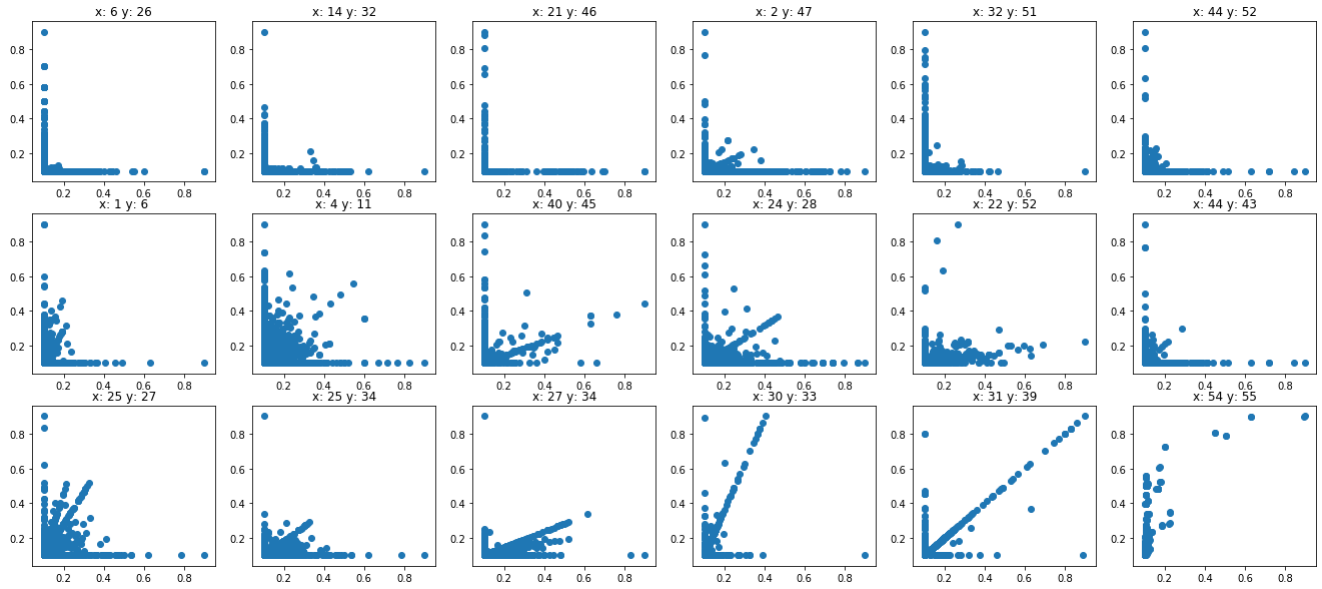


Рисунок 9 – Диаграммы рассеивания после преобразования

2.3. Выводы

Гистограммы данных после преобразования не отличаются от гистограмм, полученных в пункте 2.1. На диаграммах Box-and-Whisker видно, что максимальные и минимальные значения лежат в пределах [0, 1]. После удаления из выборки выбросов осталось 3998 примеров. Корреляция данных стала лучше наблюдаться на диаграммах рассеивания после преобразования.

III. Формирование признаков

3.1. Сокращение числа признаков

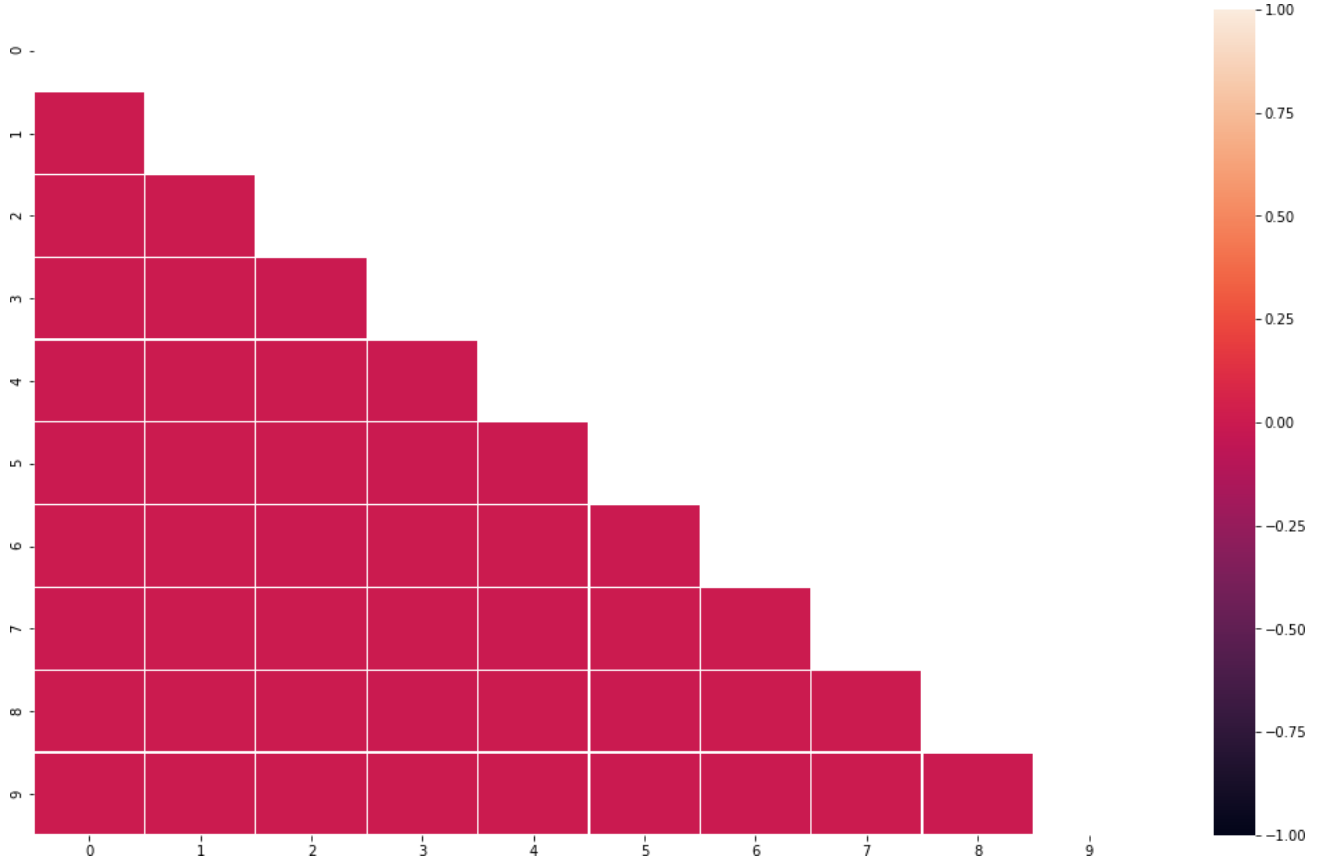
В предоставленном наборе данных имеется 57 признаков, некоторые из которых коррелируют между собой. Таким образом, в целях уменьшения вычислительных операций, было принято решение сократить количество рассматриваемых признаков с помощью метода главных компонент PCA. На корреляционной матрице наблюдается около 14 пар атрибутов, коррелирующих между собой. Эмпирическим путем было принято решение оставить 10 признаков для дальнейшей работы.

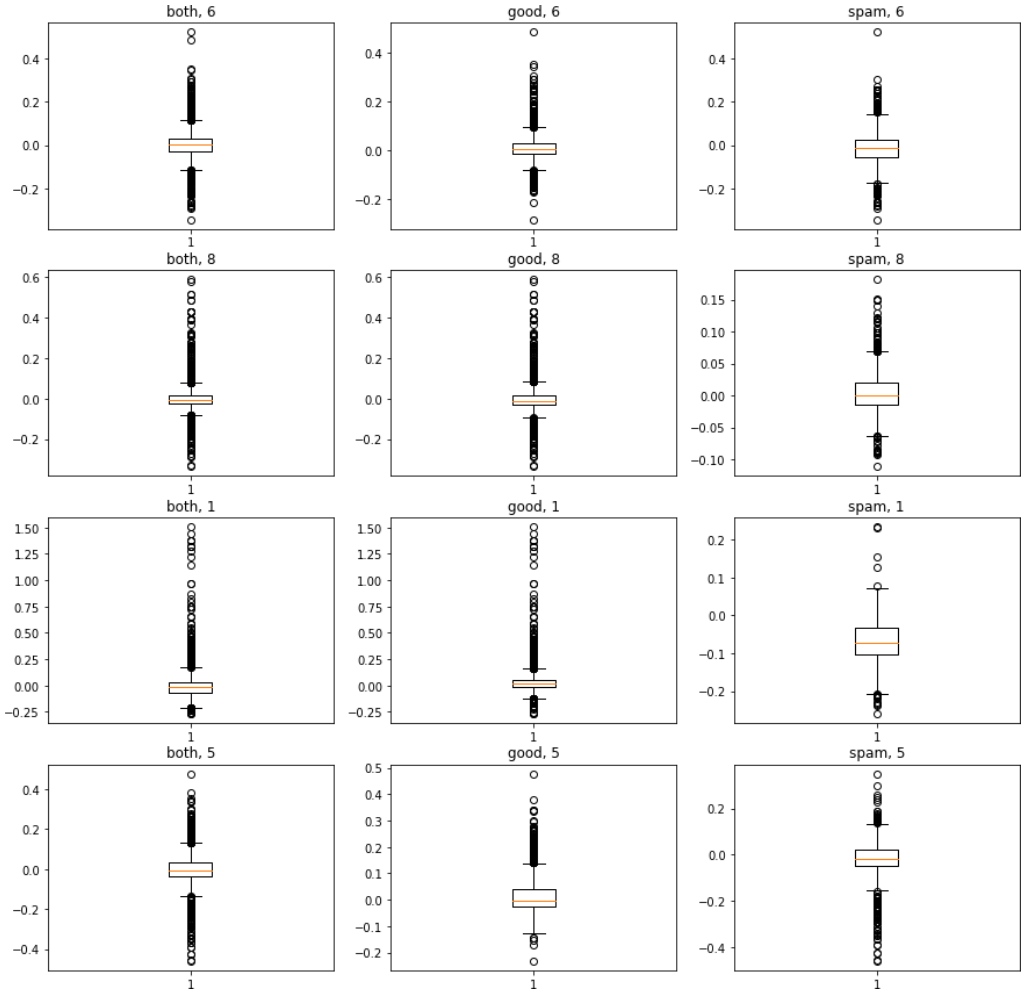
3.2. Конструирование новых признаков

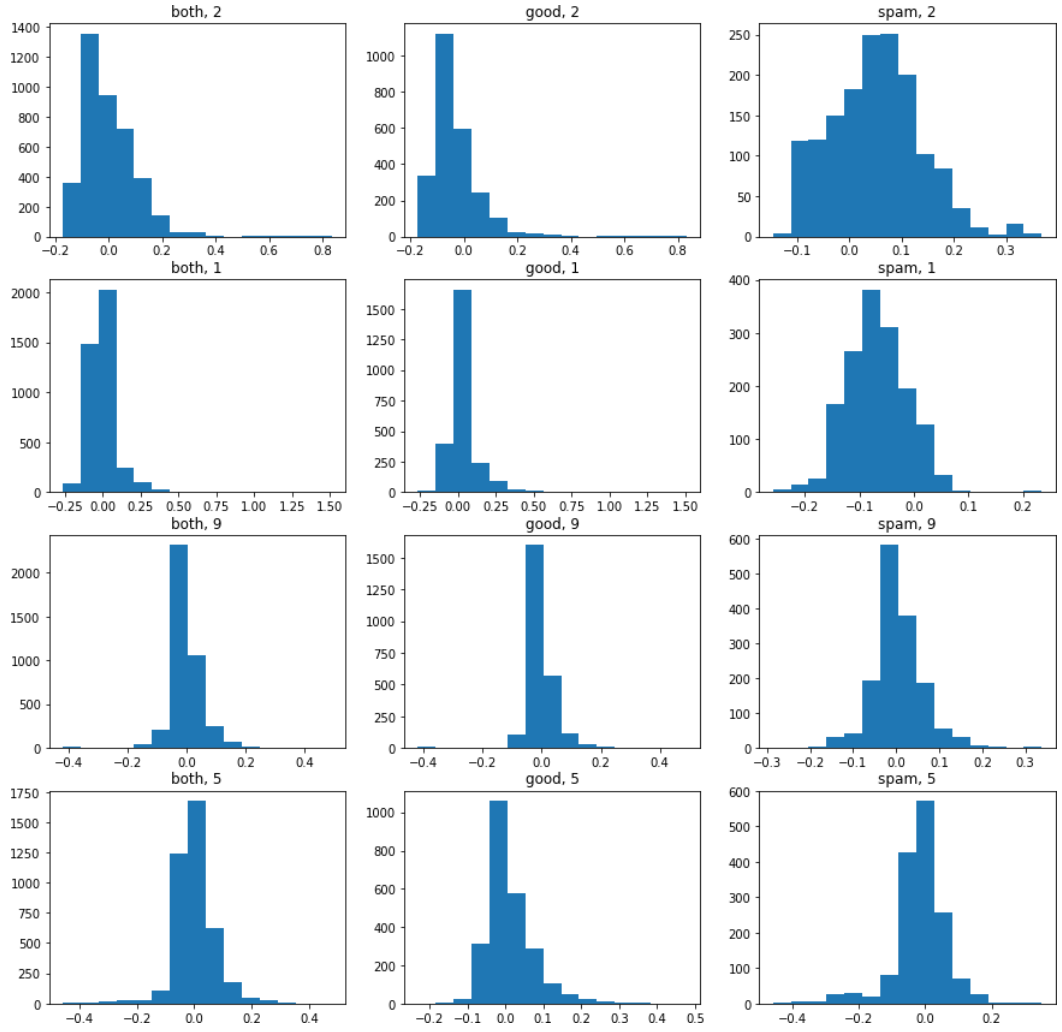
Рассматриваемых признаков достаточно много, поэтому конструирование новых не планировалось.

3.3. Выводы

После применения PCA метода было получено 0 признаков без линейной корреляции, как видно на корреляционной матрице. Остальные итоговые диаграммы приведены в приложении.




both, 6
good, 6
spam, 6
both, 8
good, 8
spam, 8
both, 1
good, 1
spam, 1
both, 5
good, 5
spam, 5



IV. Построение и исследование нейросетевых моделей

4.1. Параметры архитектуры и обучения многослойной нейронной сети

Параметр	Значение
Функция потерь	Binary-Cross-Entropy
Число входов сети	10
Число выходов сети	1
Число скрытых слоев сети*	3
Число и АХ нейронов 1-го скрытого слоя*	20, tanh
Число и АХ нейронов 2-го скрытого слоя*	10, tanh
Число и АХ нейронов 3-го скрытого слоя*	10,logistic
АХ нейронов выходного слоя	logistic

Кросс-валидация	Holdout (60/30/10)
Объёмы обучающей / валидационной / тестовой выборок	60%/ 30% /10%
Режим обучения*	Mini-batch, bs=50
Метод инициализации весов	метод Хавьера
Критерий останова	Epochs = 200
Ранний останов	да

* Определяется вариантом задания.

4.2. Исследование простого градиентного метода обучения

№ п/п	Скорость обучения, α	Число эпох обучения	Ошибка на обучающей выборке, $E_{обуч}$	Ошибка на тестовой выборке, $E_{тест}$
1	0.001	200	0.663	0.663
2	0.01	168	0.278	0.286
3	0.1	42	0.273	0.277
4	0.5	26	0.276	0.28
5	2	5	0.467	0.461

Анализируя полученные данные, лучшей обучающей скоростью оказалась 0.1.

4.3. Исследование методов GDM и NAG

№ п/п	Метод	Момент, μ	Число эпох обучения	Ошибка на обучающей выборке, $E_{обуч}$	Ошибка на тестовой выборке, $E_{тест}$
1	GDM	0	39	0.274	0.278
2	NAG		31	0.276	0.279
3	GDM	0.1	40	0.272	0.277
4	NAG		38	0.274	0.279
5	GDM	0.3	34	0.274	0.277
6	NAG		38	0.273	0.281
7	GDM	0.9	38	0.262	0.278

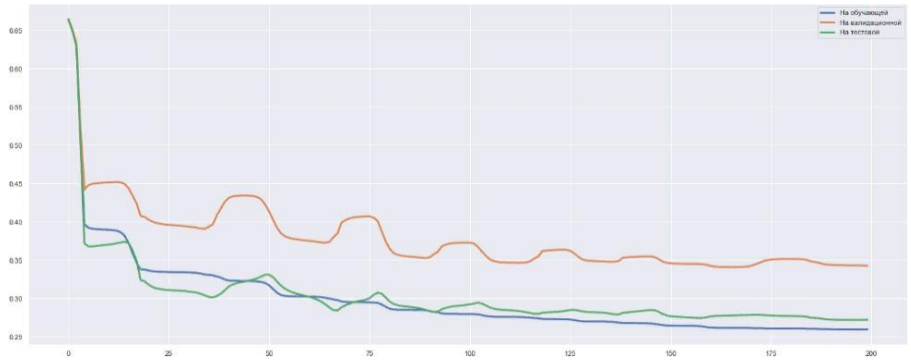
8	NAG		44	0.241	0.257

Отличие результатов двух алгоритмов наблюдается только при значении 0.9 в остальных случаях они отличаются на малую величину, скорость схождения также не сильно отличаются. При значении параметра (0.9) наблюдается явное переобучение. Наилучший результат на тестовой выборке был получена на GDM с параметром равным 0.3.

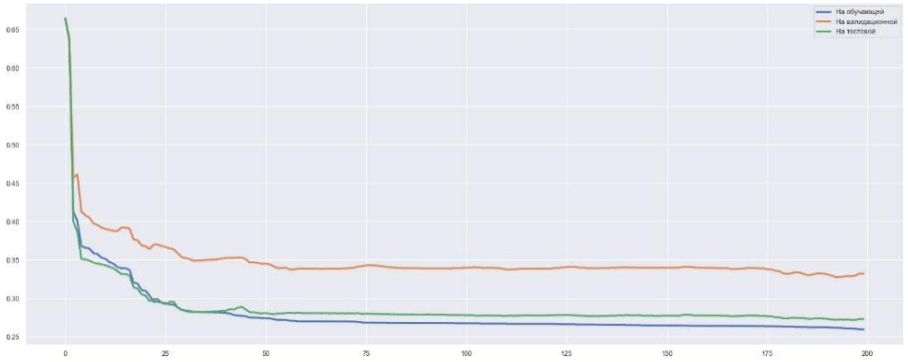
4.4. Исследование методов наискорейшего спуска и сопряжённых градиентов

а) Сравнение кривых обучения

Fletcher-Reeeves



Polak-Ribiere



б) Заполнить таблицу по результатам обучения

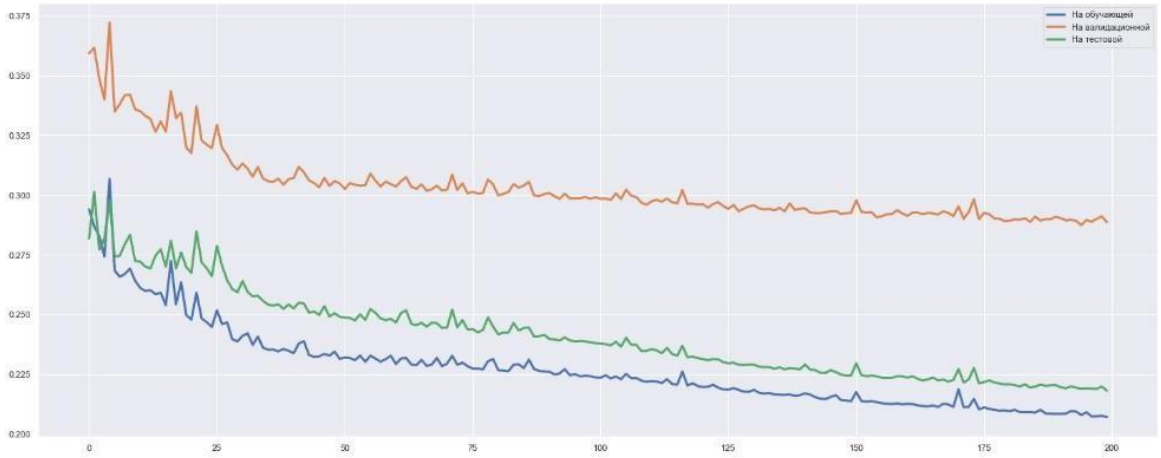
№ п/п	Метод	Число эпох обучения	Ошибка на обучающей выборке, $E_{обуч}$	Ошибка на тестовой выборке, $E_{тест}$
1	Fletcher-Reeeves	35	0.331	0.304
2	Polak-Ribiere	58	0.271	0.281

в) Выводы

Наилучший результат показал метод Polak-Ribiere. Методы сопряженных градиентов оба переобучились в пределах 100 итераций, метод слишком хорошо подстроился под обучающую выборку, необходима регуляризация.

4.5. Исследование метода AdaGrad

а) Кривые обучения



в) Заполнить таблицу по результатам обучения

№ п/п	Метод	Число эпох обучения	Ошибка на обучающей выборке, $E_{обуч}$	Ошибка на тестовой выборке, $E_{тест}$
1	AdaGrad	78	0.231	0.253

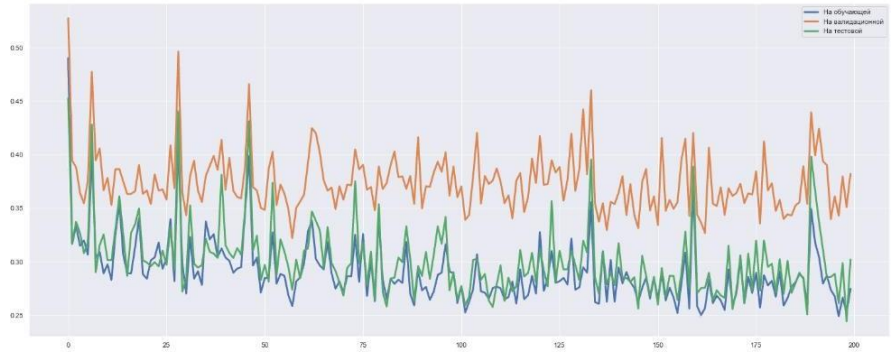
г) Выводы

Переобучения наблюдается: ошибки на выборках сошлись к разным значениям.

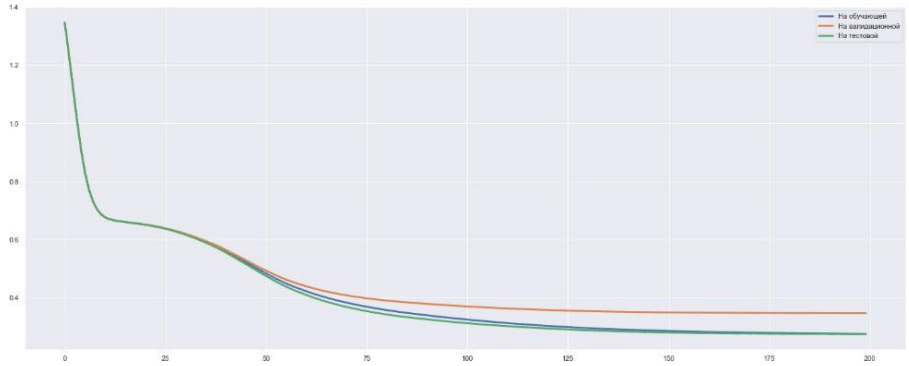
4.6. Исследование методов RMSProp и AdaDelta

а) Сравнение кривых обучения

0.85 RMSProp



0.85 AdaDelta



Больше графиков в файле 4part.ipynb

в) Заполнить таблицу по результатам обучения

№ п/п	Метод	Параметр сглаживания, ρ	Число эпох обучения	Ошибка на обучающей выборке, $E_{обуч}$	Ошибка на тестовой выборке, $E_{тест}$
1	RMSProp	0.8	21	0.272	0.294
2	AdaDelta		200	0.285	0.28
3	RMSProp	0.85	9	0.277	0.292
4	AdaDelta		200	0.276	0.276
5	RMSProp	0.9	12	0.286	0.281
6	AdaDelta		200	0.276	0.28
7	RMSProp	0.95	14	0.666	0.666
8	AdaDelta		199	0.272	0.279

г) Выводы

RMSProp:

Метод при всех значениях параметра переобучился. Наименьшее переобучение наблюдается при значении параметра 0.9 возможно при более высоком значении параметра сглаживания переобучение удастся избежать.

Adadelta:

На всех параметрах переобучения не наблюдается. Наилучший результат показал метод с параметром сглаживания равным 0.85.

Общий:

Adadelta как и ожидалось из-за балансирующего элемента в числителе при тех же значениях параметра сглаживания, не переобучалась и держала скорости в определенном диапазоне, если

судить по окончательным значениям Adadelta показал себя лучше, чем RMSProp. Однако RMSProp сошелся гораздо быстрей к значению ошибку заметно меньше, чем Adadelta. Скорее всего, если улучшить критерий останова и увеличить коэффициент сглаживания в RMSProp на этих данных он покажет результат лучше, чем Adadelta.

4.7. *Исследование метода Adam*

а) Сравнение кривых обучения

(0.9, 0.999)

Больше графиков см. в 4part.ipynb.

в) Заполнить таблицу по результатам обучения

№ п/п	β_1	β_2	Число эпох обучения	Ошибка на обучающей выборке, $E_{обуч}$	Ошибка на тестовой выборке, $E_{тест}$
1	0.9	0.9	24	0.281	0.285
2	0.85	0.85	21	0.277	0.3
3	0.9	0.999	22	0.264	0.261
4	0.85	0.9	12	0.265	0.276
5	0.95	0.9	19	0.278	0.28
6	0.999	0.999	10	0.295	0.302

г) Выводы

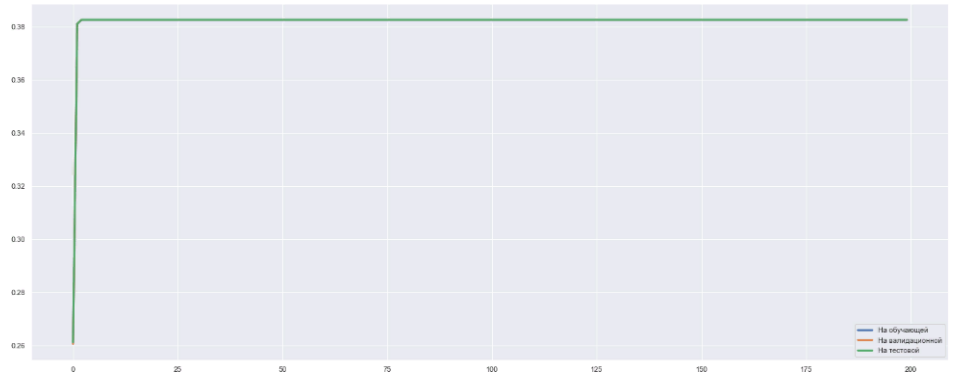
При всех значениях метод на обучающей выборке сходится. На всех наблюдается явное переобучение. При параметрах равных (0.999, 0.999) возможно из-за излишне большим значением параметра момента, алгоритм не остановился на оптимальном минимуме. (0.85, 0.9) показал хороший результат, но не лучший. Лучше всех себя показала пара параметров (0.9,

0.999), также данная пара совпадает со значением, которые рекомендуют авторы метода - (0.9, 0.999).

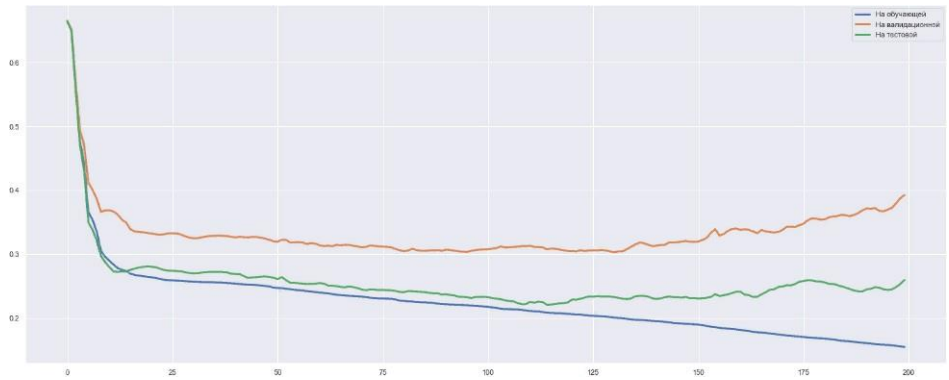
4.9. Исследование методов Левенберга-Маркардта и BFGS

а) Кривые обучения

LM



BFGS



б) Заполнить таблицу по результатам обучения

№ п/п	Метод	Число эпох обучения	Ошибка на обучающей выборке, $E_{обуч}$	Ошибка на тестовой выборке, $E_{тест}$
1	LM	1	0.261	0.261
2	BFGS	96	0.2199	0.2325

в) Выводы

LM метод расходится на данной выборке скорее всего играет роль, то что LM метод разрабатывался для минимизации квадрата ошибки, а не кросс-энтропии. BFGS переобучился

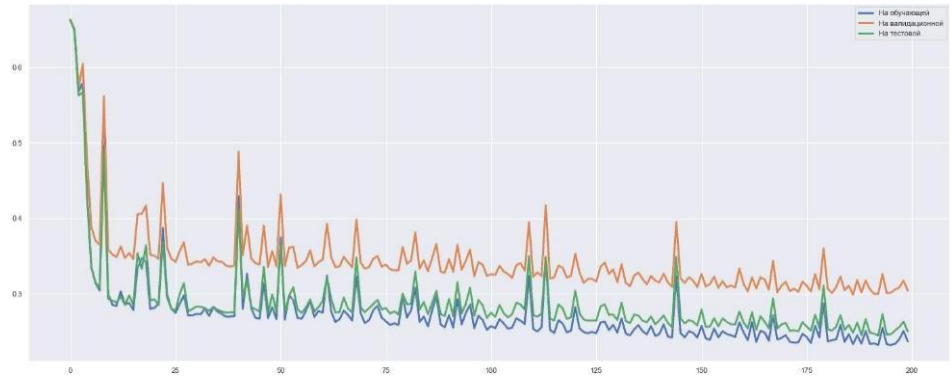
из визуального анализа графика видно, что метод сильно подстроился под обучающую выборку, скорее всего необходимы методы регуляризации для предотвращения переобучения.

4.10. Исследование метода стохастического градиента

а) Сравнение кривых обучения

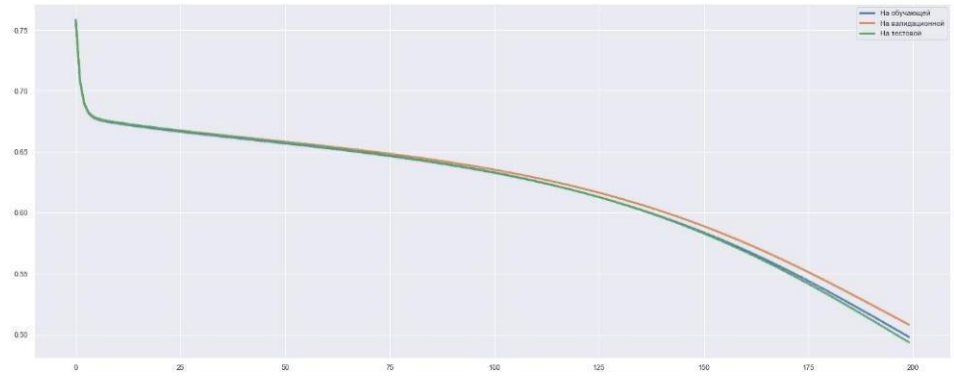
Размер батча - 1:

Размер батча - 20:



Размер батча - 100:

Размер батча - 2398:



б) Заполнить таблицу по результатам обучения

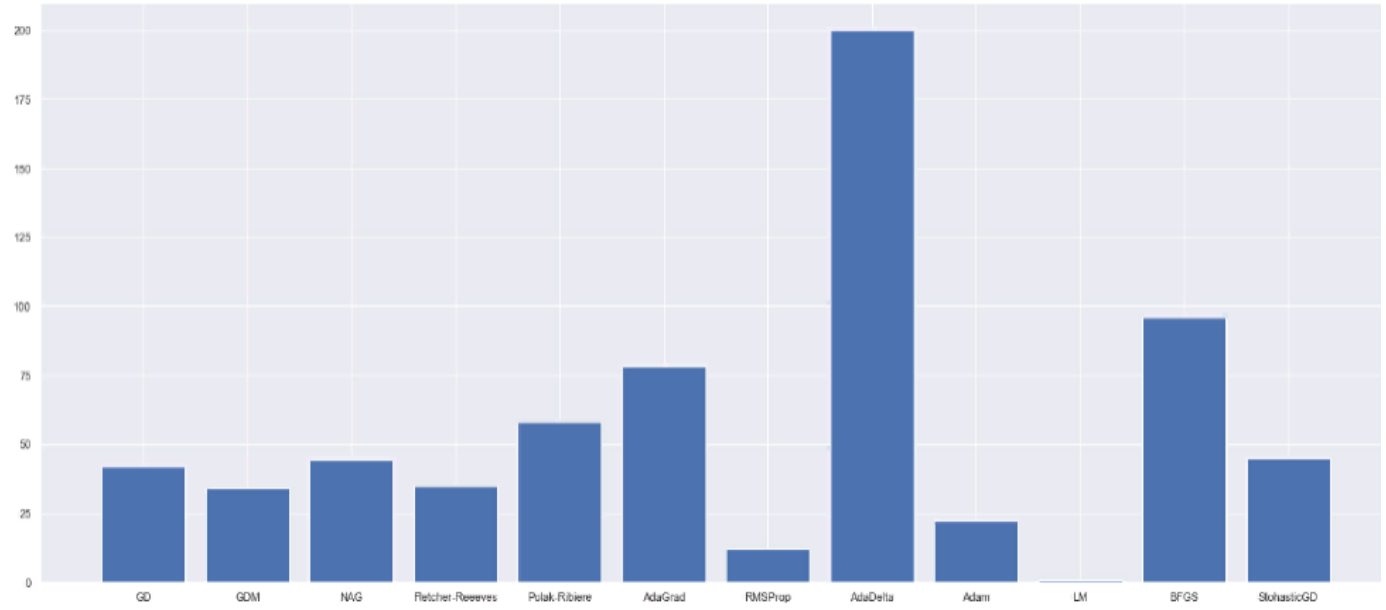
№ п/п	Метод	Размер mini-batch'a	Число эпох обучения	Ошибка на обучающей выборке, $E_{обуч}$	Ошибка на тестовой выборке, $E_{тест}$
1	GD	1	34	0.261	0.274
2	GD	20	35	0.273	0.276
3	GD	100	45	0.284	0.286
4	GD	2398	200	0.453	0.441

г) Выводы

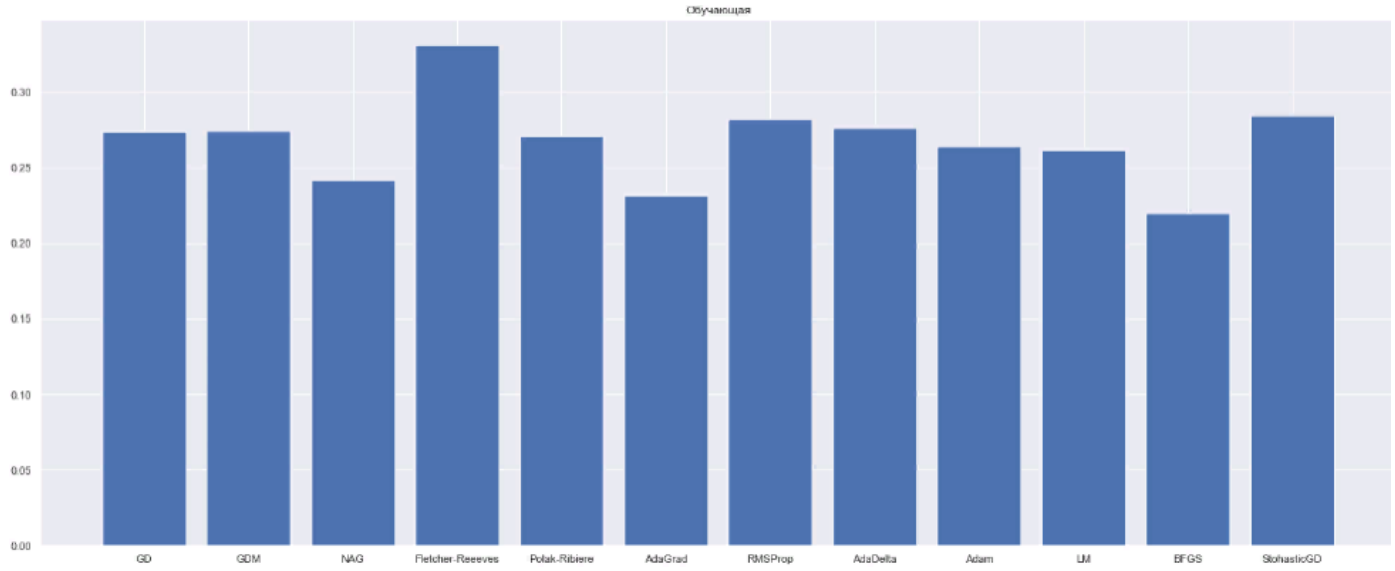
Размер mini-batch влияет на вероятность переобучиться у сети, это можно наблюдать на графках зависимости ошибки на данных от итераций обучения, на которых ошибки с batch size = 1, 20 практически с самого начала подвержены переобучению. Это объясняется тем, что в данной работе решается задача классификации, и хоть данные и перемешаны, но есть вероятность того, что в одном mini-batch'е может наблюдаться сильное доминирование одного из классов, что не позволит методу адекватно оценивать ошибку на данных в целом.

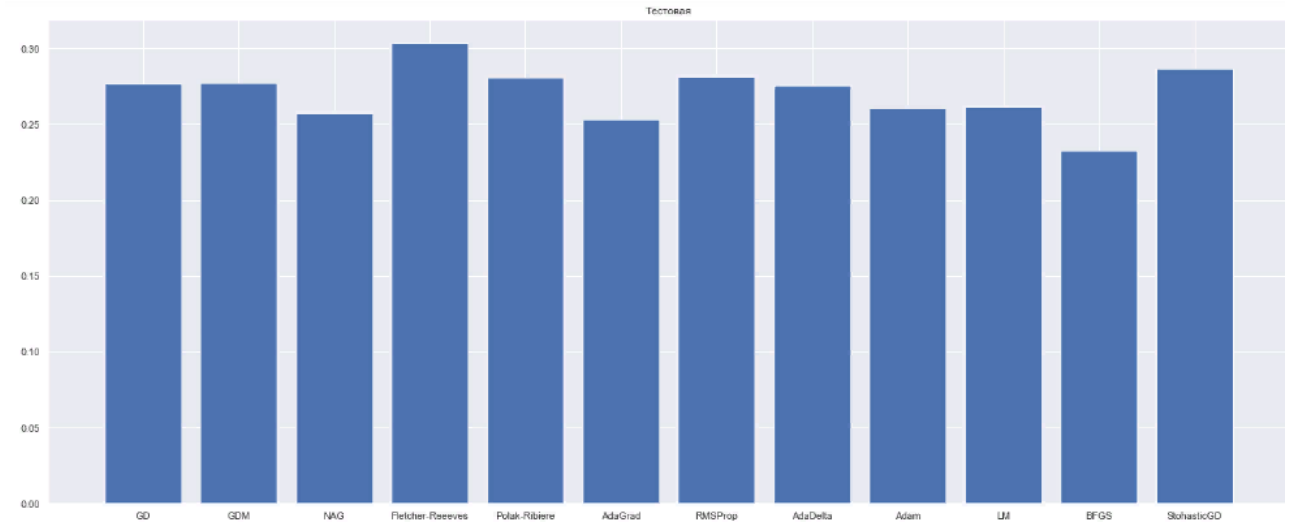
4.11. Сравнение методов обучения

а) Сравнение числа эпох обучения



б) Сравнение качества обученных нейросетевых моделей





в) Выводы

GD: Показал средние результаты как на обучающей выборке так и на тестовой, поэтому делается вывод о том, что с хорошим подбором параметра скорости обучения GD способен конкурировать с более ‘сложными’ методами, однако требует слишком много времени.

GDM: Показал тоже средние результаты, но наилучший результат получился при малом значении параметра μ, что говорит о том, что возможно момент для данной функции не совсем применим.

NAG: Аналогично GDM. Так как наилучшие результаты получились при малом значении параметра μ следственно отличий от GDM на наблюдалось.

Fletcher-Reeeves: Показал плохие результаты: наблюдается переобучение, и недообучение на обучающей выборке в сравнении с другими методами, однако количество итераций мало и минимальная тестовая ошибка (ошибка до начала переобучения) сравнима с другими методами, возможно, при улучшении критерия останова и применения ряда регуляризаций метод покажет себя.

Polak-Ribiere: Показал плохие результаты, но хуже, чем Fletcher-Reeeves переобучение меньше и результаты на обучающей выборке хуже, возможно повезло с выбором начальной точки. Требует больше времени. Соображения аналогичные Fletcher-Reeeves.

AdaGrad: Показал средние результаты при максимальном количестве итераций. Все скорости параметров довольно быстро, сошлись сильно замедлив обучение, ошибка на обучающей выборке выше, чем у методов с одной скоростью обучения на все параметры.

RMSProp: Показал плохие результаты: переобучение. Однако достиг минимума за малое количество итераций и анализ скоростей показал, что возможно при увеличении параметра сглаживая переобучения удастся избежать.

Adam: Результат схож с результатами RMSProp.

Adadelta: Показал один из наилучших результатов на тестовой выборке и хороший на обучающей.

LM: Показал наихудший результат, ошибка на обучающей выборке не сошлась.

BFGS: Наблюдается самое сильное переобучение, однако минимальная тестовая ошибка достаточно мала и, если улучшить критерий останова, его можно использовать в условиях ограниченности времени.

Stochastic GD: Сравним с GD, но занимает меньше памяти во время обучения.

4.13. Методы кросс-валидации

а) Заполнить таблицу по результатам кросс-валидации различными методами

Метод кросс-	Число запусков	Средняя ошибка на	Средняя ошибка на

валидации	обучения	обучающей выборке ± с.к.о., $\overline{E}_{обуч} \pm \sigma[E_{обуч}]$	тестовой выборке ± с.к.о., $\overline{E}_{тест} \pm \sigma[E_{тест}]$
Монте-Карло	10	0.29	0.007
Holdout 60/30/10	1	0	0
10-fold	10	0.295	0.081

Оценки модели методов 10-fold и Монте-Карло в целом схожи, но оценка алгоритма на тестовом наборе данных завышена у Монте-Карло в сравнении с 10-fold.

4.14. Исследование различных архитектур нейронных сетей

а) Исследование зависимости качества обучения от числа нейронов в скрытых слоях

Провести обучение нейронных сетей с различным числом нейронов в скрытых слоях. По результатам обучения заполнить таблицу.

№ п/п	Число нейронов в скрытых слоях	Средняя ошибка на обучающей выборке ± с.к.о., $\overline{E}_{обуч} \pm \sigma[E_{обуч}]$	Средняя ошибка на тестовой выборке ± с.к.о., $\overline{E}_{тест} \pm \sigma[E_{тест}]$
1	(20, 10, 10)	0.331	0.304
2	(25, 15, 10)	0.292	0.28
3	(20,20,20)	0.419	0.407
4	(20, 30, 10)	0.29	0.298
5	(30, 30, 30)	0.332	0.325
6	(30, 40, 20)	0.521	0.502

Замечание: использовался FR.

б) Исследование зависимости качества обучения от активационных характеристик нейронов

Наилучшей архитектурой из предыдущего из пункта а) является архитектура - [25, 15, 10]

АХ нейронов скрытых слоёв	Средняя ошибка на обучающей выборке ± с.к.о., $\overline{E}_{обуч} \pm \sigma[E_{обуч}]$	Средняя ошибка на тестовой выборке ± с.к.о., $\overline{E}_{тест} \pm \sigma[E_{тест}]$
logistic	0.292	0.28
tanh	0.557	0.514
linear	0.54	0.537
softsign	0.54	0.537

softplus	0.659	0.659