

# БЭГГИНГ НА ОТСУТСТВИИ ВОСПРОИЗВОДИМОСТИ ОБУЧЕНИЯ В НЕЙРОСЕТИ

**Зенков В.В.**

*Институт проблем управления им. В.А. Трапезникова РАН, Москва, Россия  
zenkov-v@yandex.ru*

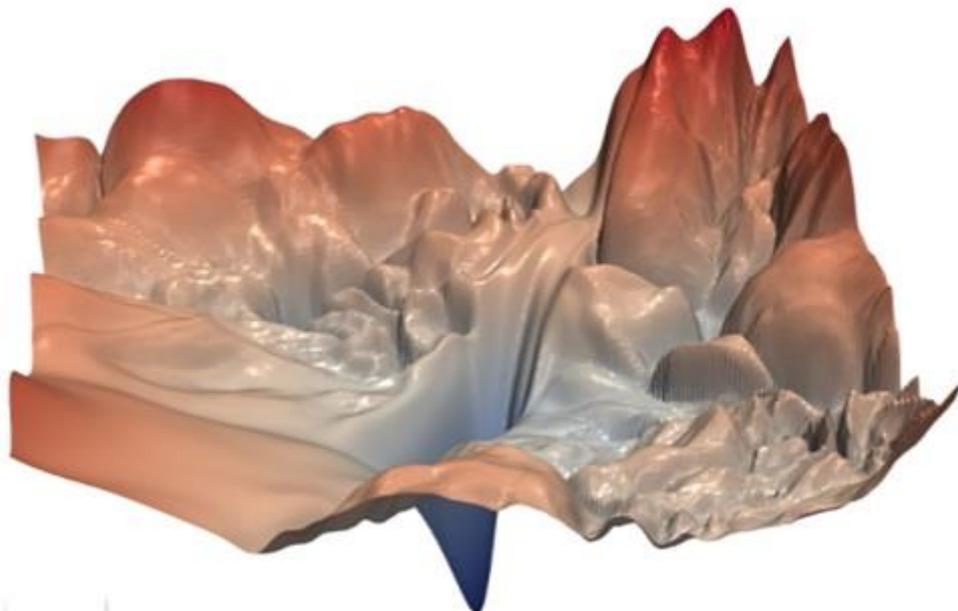
*Аннотация. Изменяя параметр псевдослучайного процесса оптимизации нейросети, генерируется несколько вариантов решения задачи. На их основе получается бэггинг - усредненное и менее подверженное переобучению решение. Приведен пример выполнения такого бэггинга и приведено его сравнение с методом классификации на основе дискриминантной функции Андерсона и тождественно связанной с ней апостериорной вероятностью класса.*

*Ключевые слова: искусственная нейросеть, обучающая выборка с учителем, бэггинг, дискриминантная функция Андерсона, апостериорная вероятность класса.*

## **Введение**

При повторном обучении нейросети на одной и той же обучающей выборке с учителем с одной и той же тестовой выборкой качество обучения, как обычно, бывает разным, иногда и очень разным. В работе [1] сообщается, что в 2016 году в журнале Nature приведен результат опроса 1576 исследователей о существовании кризиса воспроизводимости в обучении нейросети. По итогам опроса 52% исследователей считают, что имеет место существенный кризис, 38% – легкий кризис (итого 90% суммарно!),

Причина отсутствия воспроизводимости результата обучения заключается, по крайней мере, в псевдослучайном механизме, используемом в градиентном методе обучения нейросети. На Рис. 1, взятом из [2], показан вид функции потерь в пространстве двух параметров из нескольких десятков миллионов параметров оптимизации одной нейросети. Минимум функции потерь ищется градиентным методом с использованием случайного механизма. Очевидно, как бы ни был совершен поисковик, попасть в глобальный минимум с одной попытки ему вряд ли удастся. Большинство вариантов обучения при подобной функции потерь заканчивается попаданием в локальные минимумы со своими результатами обучения нейросети.



*Рис. 1. Пример функции потерь*

Улучшить несколько вид функции потерь для обучения нейросети градиентным методом можно путем удаления части связей между нейронами нейросети [2]. Но кардинально преобразовать нейросеть таким образом, чтобы функция потерь стала удобной для градиентного метода поиска экстремума, вряд ли возможно. Поэтому проще многократно повторить обучение исходной или преобразованной нейросети с целью получения хорошего результата обучения, либо нескольких хороших результатов для построения из них бэггинга, как сделано в нашей работе.

Можно также прекратить мучения с нейросетью и использовать иной способ решения задачи. Например, для решения задач классификации в виде альтернативы нейросетевому методу можно использовать наш метод, использующий аппроксимацию дискриминантной функции Андерсона (ДФА) в пространстве признаков классов. С ДФА тождественно связана апостериорная вероятность (АпоВ) класса в задаче классификации с двумя классами. АпоВ классов – исчерпывающая объективная информация для решения задач классификации. Если классов больше двух, и они несовместны, то методом один класс против остальных находятся АпоВ всех классов.

Непараметрический метод оценивания ДФА и АпоВ в точке пространства признаков классов избавляет от необходимости задавать вид аппроксимирующей зависимости ДФА. Оценивать ДФА в точке можно методом касательной по взвешенным точкам обучающей выборки, используя взвешенный метод наименьших квадратов (МНК), или просто применять средневзвешенную оценку Надарая-Ватсона [3]. При этом в качестве независимой переменной в выборке нужно использовать соответствующую разность стоимостей ошибок. Стоимости ошибок нужны для определения ДФА. При этом на результат пересчета ДФА в АпоВ классов выбор стоимостей ошибок не влияет.

Критерием качества обучения служит сумма квадратов разностей оценок ДФА и независимых переменных в тестовых точках, полученных аналогично независимым переменным в точках обучающей выборки. Критерий зависит от скалярного параметра весовой функции. Лучшее значение параметра находится простым перебором или градиентным методом. Зависимость критерия от параметра весовой функции, судя по решенным примером, унимодальная.

Наш метод дает воспроизводимый результат обучения. В силу простоты и лаконичности теория метода излагается на двух-четырёх страницах текста в зависимости от количества межстрочных интервалов.

Но вернемся к нейросети. Случайный процесс, генерируемый компьютером, является псевдослучайным и может быть воспроизведен повторно, если повторно использовать то же значение параметра, задающего этот псевдослучайный процесс. Параметр, определяющий псевдослучайный процесс, может быть случайным или быть заданным пользователем.

В случае с нашим очень простым примером нейросети для решения задачи классификации типичная проблема невоспроизводимости результатов обучения нейросети тоже имеет место. Однако можно обеспечить воспроизводимость серии результатов обучений нейросети, если перед каждым обращением к оптимизатору обучения нейросети в серии задавать программе свои значения параметра, запускающего псевдослучайные процессы, используемые оптимизатором в серии обучений.

Если нейросеть дала плохой результат обучения, не нужно торопиться менять ее гиперпараметры – количество слоев и количества нейронов в них. Вместо этого следует попробовать повторить обучения при разных значениях параметра, задающего воспроизводимый псевдослучайный процесс, используемый в оптимизаторе нейросети, с целью поиска подходящих вариантов обучения.

Зависимость результата обучения нейросети от параметра генерации псевдослучайного процесса мы используем для получения бэггинга. В отличие от известного бэггинга [4] мы не используем подвыборки из обучающей выборки с учителем. Вместо них мы запоминаем несколько “неплохо” обученных вариантов обучения нейросети со своими значениями параметра генерации псевдослучайного процесса. Далее отобранные варианты обучения нейросети параллельно используются для получения средних оценок АпоВ классов для классифицируемого объекта для помещения его в тот или иной класс при решении задачи классификации при субъективно заданном, например, байесовым или ином критерии классификации. Прием осреднения бэггинга позволяет надеяться на меньшую переобученность модели.

Для демонстрации бэггинга использован простой пример из работы [5]. Из нее взят для сравнения с нейросетью результат оценивания АпоВ класса методом аппроксимации ДФА.

Программа получения бэггинга написана на Питоне с использованием пакетов Keras и TensorFlow. Для обеспечения воспроизводимости результата обучения нейросети использовали две функции. Функция `numpy.random.seed(Vsp)` использовалась для воспроизводимой генерации обучающей выборки, если потребуется, (в примере  $Vsp=2$ ). Вторая функция `tensorflow.random.set_seed(Vspp)` использовалась для генерации воспроизводимых псевдослучайных процессов, используемых в обучении нейросети. Если  $Vspp=-1$ , то обращение к `tensorflow.random.set_seed(Vspp)` отменялось, и генерация псевдослучайного процесса для обучения нейросети выполнялась стандартным невоспроизводимым случайным образом.

Задавая значения  $Vspp \geq 0$ , получили различные по качеству воспроизводимые результаты обучения нейросети, часть из которых использовалась для бэггинга.

## 1. Демонстрационный пример

В качестве примера решения задачи классификации использована полносвязная нейронная сеть [5] с двумя скрытыми слоями по 4 нейрона в каждом слое. Один нейрон в выходном слое. Функция активации после каждого скрытого слоя `relu`. Функция активации в выходном слое - `sigmoid`.

Работу нейросети выполняли операторы:

Компиляция модели:

```
model.compile(loss = "binary_crossentropy", optimizer = "adam", metrics = ['accuracy']).
```

Обучение производилось по обучающей выборке с учителем, состоящей из двух векторов:  $X$  – значения одномерного признака,  $Y$  – единицы, если соответствующие значения признака  $x \in X$  принадлежали классу А, иначе нули, это оценки Апоб класса А от признака  $x \in X$ :

```
model.fit(X, Y, epochs = 200, batch_size=32, verbose=0).
```

Обучение и его результат `scores` по обучающей выборке  $X, Y$ :

```
scores = model.evaluate(X, Y).
```

При подаче на вход обученной нейросети вектора тестовой выборки  $X_t$  на выходе нейросети получается вектор  $Y_t$  оценок Апоб класса А в точках  $x \in X_t$ :

```
Y_t = model.predict(X_t).
```

В примере решалась задача классификации с тремя несовместными классами А, В и С. Априорные условные распределения признака классов и априорные вероятности классов известны. По известной в примере априорной информации получены по формуле Байеса истинные Апоб классов для оценки в примере качества обучения.

Для обучений нейросети по известным априорным данным сгенерирована обучающая выборка с учителем заданного объема и ассортимента в виде вектора  $X$  – значения признака и вектора  $Y^1$  – метки трех классов, которым принадлежали соответствующие значения признака в  $X$ .

В машинном обучении решение многоклассовой задачи классификации в стохастической постановке с несовместными классами сводится к решению последовательности задач с двумя классами по принципу один класс, Апоб которого находятся, против остальных классов, принимаемых за другой класс. В таком случае вектор  $Y$ , используемый для обучения, получается из вектора  $Y^1$  путем, например, замены в  $Y^1$  меток А на единицы, а остальных меток на нули.

Таким образом находятся оценки Апоб каждого класса в заданной точке пространства признаков. Далее заданная точка (объект классификации) с учетом Апоб классов относится в один из классов в соответствии с субъективными стоимостями ошибок классификации, субъективным критерием или критериями классификации.

Мы ограничились рассмотрением одного объективного этапа решения задачи – оценкой Апоб класса А. Классы В и С объединены в другой класс В+С.

Выполняли заданное количество обучений вышеописанной нейросети для решения задачи – получение оценок Апоб класса А в точках тестовой выборки  $X_t$  и строили бэггинг по отобранным “хорошим” вариантам обучения. Обучения получали невозпроизводимым способом при  $Vspp=-1$ , запоминая “хорошие” варианты обученной нейросети для бэггинга. Выполняли также обучения воспроизводимым способом, задавая некоторую последовательность значений параметра  $Vspp \geq 0$ .

Предполагалось, что условные распределения одномерного признака по классам  $p(x|A)$ ,  $p(x|B)$ ,  $p(x|C)$  нормальные. Параметры условных распределений известны. Средние классов А, В, С соответственно равны 0, -3, 3. Дисперсии всех условных распределений  $x$  равны единице. Априорные вероятности классов  $P(A)$ ,  $P(B)$ ,  $P(C)$ , соответственно равны 0.5, 0.25, 0.25.

В качестве критерия качества рассматриваемого этапа решения задачи использовали среднюю квадратичную ошибку оценки нейросетью Апоб класса А, зная по априорным данным истинную зависимость Апоб класса А от  $x$ ,  $p(A|x)$ .

Обучающую выборку сгенерировали последовательно по классам А, В, С функцией `numpy.random.normal()` в количестве 120 точек и в нужном ассортименте по классам 60, 30 и 30 точек соответственно априорным вероятностям классов. Перед генерацией обучающей выборки выполнили обращение в функции `numpy.random.seed(2)` для воспроизведения обучающей выборки, если она может кому-либо потребоваться.

Тестовую выборку создали из 30 точек, равномерно расположенных на отрезке  $[-4, 4]$  признака классов  $x$ .

Для сравнения с нейросетевыми решениями приведен воспроизводимый по своей сути результат решения, получаемого по той же обучающей выборке аппроксимацией ДФА и связанной с ней Апоб класса А [5]. В качестве метода обучения использовался взвешенный метод наименьших квадратов с весовой функцией экспонентой от расстояния точки выборки до заданной точки, умноженного на

отрицательный коэффициент – настраиваемый параметр обучения. Лучшее значение параметра обучения находилось простым перебором. Для обучения использовался метод LOO (leave-one-out), в котором при заданном значении параметра обучения последовательно каждая точка обучающей выборки с возвратом переводилась в тестовую и для нее по остальным точкам выборки находилась оценка АпоВ класса А [5].

В обучающей выборке с учителем для аппроксимации ДФА выполняется замена меток классов соответствующими разностями стоимостей ошибок классификации [5]. Выбор стоимостей ошибок, необходимых для определения ДФА, не влияет на результат расчета АпоВ классов [5]. Построение ДФА непараметрическим методом в точке в качестве касательной к ДФА взвешенным МНК или с помощью средневзвешенной оценки Надарая-Ватсона избавляет и от необходимости выбора аппроксимирующей ДФА зависимости, но вынуждает использовать всю обучающую выборку и после обучения для получения оценок ДФА и АпоВ классов во всех точках тестовой выборки.

Все эти сведения приведены с целью привлечения внимания читателя к простому методу классификации по ДФА и АпоВ, дающему воспроизводимые результаты работы. Стоит заметить, что при малом объеме обучающей выборки, например, в начале работы, сложно бывает делать предположения о виде аппроксимирующих зависимостей, а также делить выборку на обучающую и тестовую части вместо применения метода LOO.

Вначале для демонстрации невоспроизводимости результатов обучения нейросетью выполнили последовательно 20 сеансов обучения при  $Vspp=-1$ , т.е. без управления генерацией псевдослучайной последовательности. На Рис. 2 изображены результаты обучений нейросети в виде погрешностей оценивания АпоВ класса А по вариантам обучения от нулевого до 19-го. Невоспроизводимость результатов обучения лежат в большом диапазоне ошибок от 0.035 до 0.4. Это свидетельствует о невозможности решения задачи классификации нейросетевым методом за одно обучение.

На Рис. 3 представлены априорные условные распределения признака для трех классов, ДФА, АпоВ класса А - сплошная красная линия, а также пунктирные линии для ДФА и АпоВ класса А, полученные по выборочным оценкам параметров условных распределений признаков классов. Точки обучающей выборки по классам А, В и С расположены на уровнях  $-0.5$  и  $+0.5$ , как того требует метод аппроксимации ДФА при равных стоимостях ошибок классификации, задаваемых для определения ДФА, [5].

На Рис. 3 представлен и результат оценивания АпоВ класса А, (звездочки), полученный методом бэггинга по отобраным вариантам обучения, имеющих погрешности меньше 0.1. Их оказалось пять, см. Рис. 2. Воспроизвести их нельзя из-за  $Vspp=-1$ . На Рис. 2 ошибка бэггинга изображена пунктиром на отметке 0.059, расположенной ниже заданной граничной пунктирной линии 0.1. Наименьшая ошибка 0.035 оценки АпоВ класса А получена в шестой точке (отсчет с нуля).

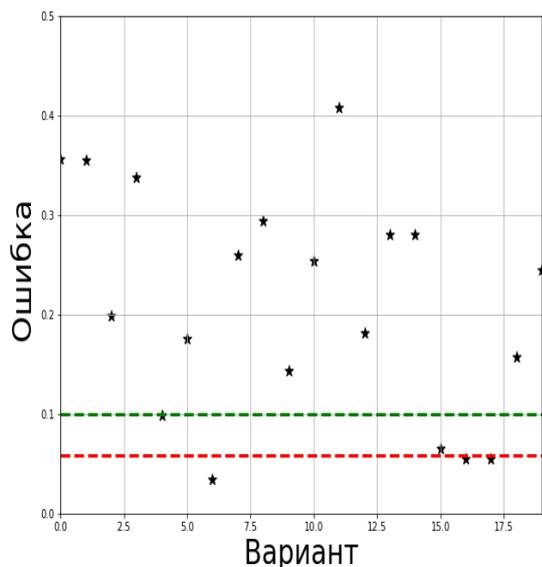


Рис. 2. Ошибки 20-ти вариантов

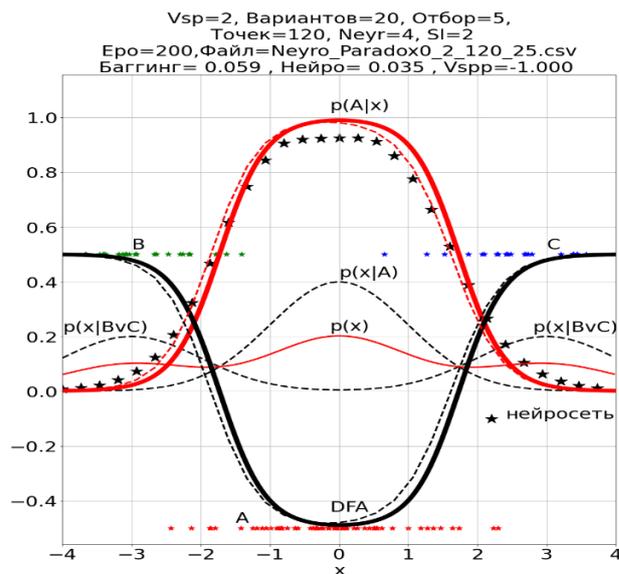


Рис. 3. ДФА,  $p(A|x)$ , ошибки вариантов

Проведены также 20 воспроизводимых вариантов обучения нейросети с параметром воспроизводимости  $Vspp$ , изменяющимся от нуля до 19-ти с единичным шагом. Результаты расчета ошибок оценки АпоВ класса А по вариантам воспроизводимости  $Vspp$  представлены на Рис. 4.

Отобраны для бэггинга варианты с ошибками не более 0.1. На Рис. 4 они лежат ниже пунктирной горизонтали на отметке 0.1. В каждой тестовой точке оценку бэггинга получаем как среднее значение оценок по отобранным вариантам обучения при  $V_{spp}=(1,3,7,8,15,17,18)$ . Ошибка бэггинга в оценке АпоВ класса А составила 0.080. Минимальная ошибка нейросети в варианте обучения  $V_{spp}=8$  составила 0.066.

Для сравнения, ошибка при аппроксимации ДФА методом касательной составила 0.058, методом Надарая-Ватсона 0.055 [5].

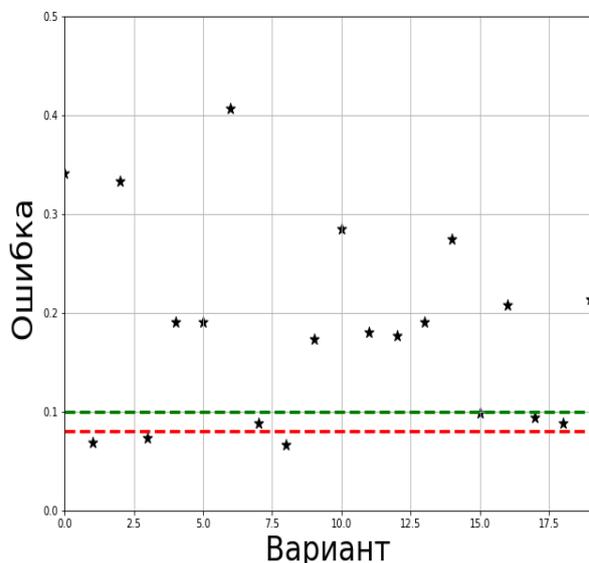


Рис. 4. Воспроизводимые 20 вариантов

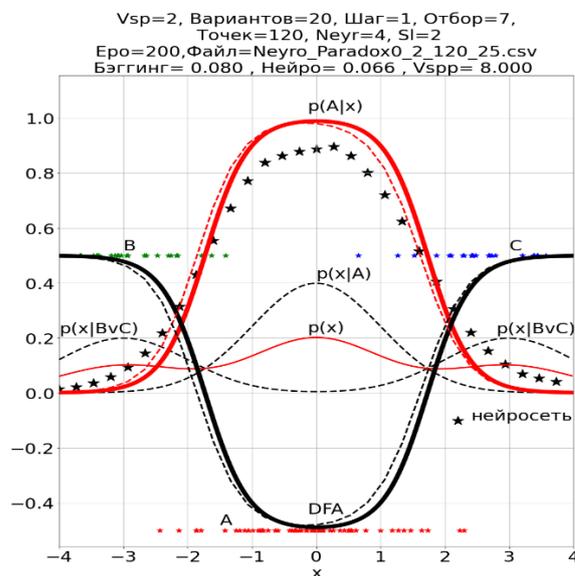


Рис. 5. Ошибки бэггинга, 20 вариантов

Аналогично получены решения для случая с  $V_{spp}$ , изменяющегося от нуля до 999 с шагом единица, на Рис. 6 и Рис. 7. Вариантов с ошибкой меньше 0.1 оказалось 236. Ошибка бэггинга 0.064 изображена пунктирной линией, расположенной ниже 0.1. Минимальная ошибка варианта обучения нейросети составила 0.029 при  $V_{spp}=559$ .

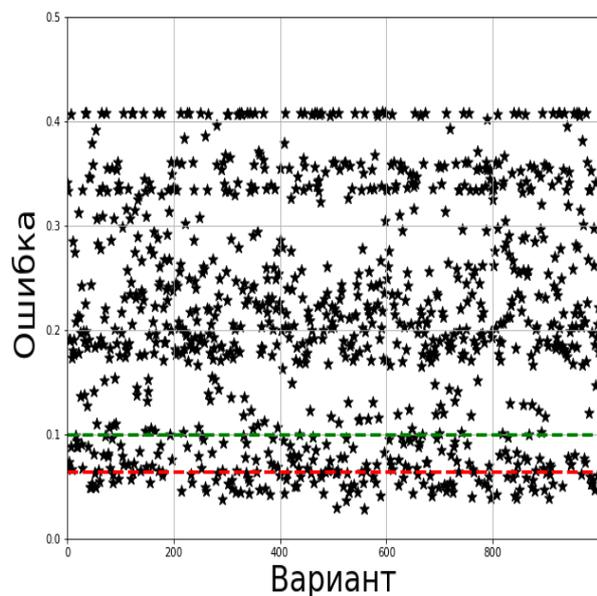


Рис. 6. Ошибки 1000 вариантов

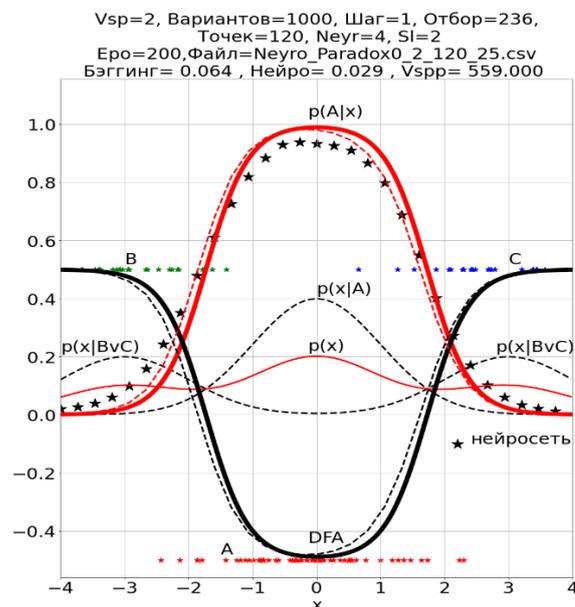


Рис. 7. Ошибки бэггинга, 1000 вариантов

Результаты для случая с  $V_{spp}$ , изменяющегося от нуля до 9999, бэггинга с шагом 0.1 представлены на Рис. 8 и Рис. 9. Вариантов с ошибкой меньше 0.1 оказалось 2274. Ошибка бэггинга 0.064, как в предыдущем случае. Минимальная ошибка варианта обучения нейросети составила 0.031 при  $V_{spp}=349.2$ .

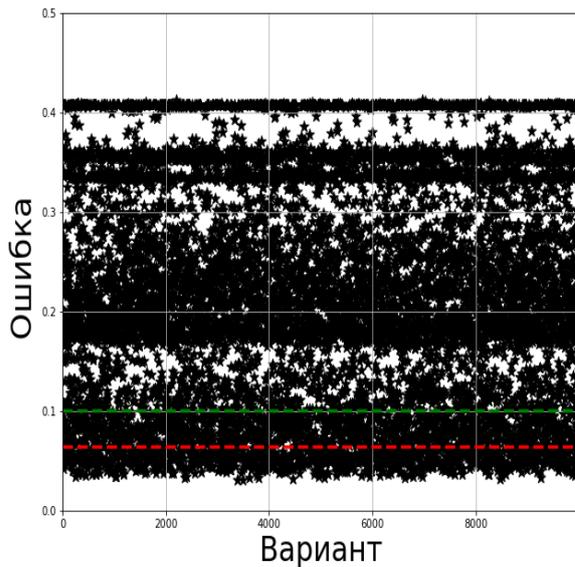


Рис. 8. Ошибки 10000 вариантов

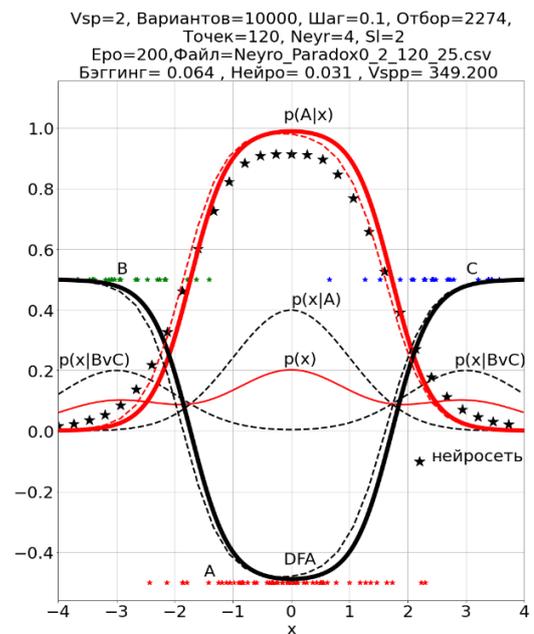


Рис. 9. Ошибки бэггинга, 10000 вариантов

## 2. Заключение

При обучении нейросети используется градиентный метод поиска экстремума многоэкстремальной функции. Это является причиной отсутствия воспроизводимости результатов обучения – при повторных обучении на одних и тех же данных результаты обучения получаются совершенно разные. Этот парадокс продемонстрирован на простой нейросети для решения задачи классификации.

Если нейросеть не дает желаемого результата, не стоит спешить с ее усложнением. Нужно многократно использовать повторное ее обучение в надежде получить приемлемый результат.

Для получения воспроизводимых результатов обучения в программе перед оптимизатором следует задавать значение параметру, определяющему генерацию псевдослучайного процесса, который используется оптимизатором. Иначе используется случайное значение этого параметра и результат обучения получается невоспроизводимым.

Предложено после проведения серии обучений нейросети из отобранных “хороших вариантов” создавать своеобразный бэггинг – получение усредненного решения задачи на отобранных вариантах нейросети. Это может снизить эффект переобучения.

В качестве сравнения с нейросетевым методом предложен наш метод решения задачи, основанный на аппроксимации по обучающей выборке с учителем дискриминантной функции Андерсона и тождественно связанной с ней апостериорной вероятностью класса. В нем для решения задачи при выбранной весовой функции используется метод наименьших квадратов, он дает воспроизводимый результат, имеет скалярный параметр настройки весовой функции (в наших примерах весовая функция – экспонента). При скалярном параметре настройки весовой функции оптимальное значение параметра настройки находится методом перебора.

## Литература

1. Deep Fake Science, кризис воспроизводимости и откуда берутся пустые репозитории // Хабр. <https://habr.com/ru/articles/480348/>.
2. Visualizing the Loss Landscape of Neural Nets [https://proceedings.neurips.cc/paper\\_files/paper/2018/file/a41b3bb3e6b050b6c9067c67f663b915-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2018/file/a41b3bb3e6b050b6c9067c67f663b915-Paper.pdf).
3. Воронцов К.В. Математические методы обучения по прецедентам (теория обучения машин) URL: <http://www.machinelearning.ru/wiki/images/6/6d/Voron-ML-1.pdf>.
4. Открытый курс машинного обучения. Тема 5. Композиции: бэггинг, случайный лес // Хабр <https://habr.com/ru/companies/ods/articles/324402/>.
5. Зенков В.В. Дискриминантная функция Андерсона или нейросеть для оценки апостериорных вероятностей классов при решении задач классификации в машинном обучении // Труды 15-й Международной конференции «Управление развитием крупномасштабных систем» (MLSD'2022). – М.: ИПУ РАН, 2022. – С. 1199-1208.