СОДЕРЖАНИЕ

ВВЕДЕНИЕ 3
1 Обзор предметной области 4
1.1 Происхождение сигналов 4
1.2 Основные параметры сцинтилляторов 6
1.2.2 Физическая конверсионная эффективность 6
1.2.2 Время высвечивания 7
1.2.3 Средняя энергия 8
1.2.4 Техническая конверсионная эффективность 8
1.2.5 Эффективность регистрации 9
1.3 Подход к разделению сигналов как к задаче кластеризации 9
1.3.1 Постановка задачи 9
1.3.2 Показатели, связанные с задачей кластеризации 10
2 Разработка и обзор алгоритмов для определения типов сигналов 13
2.1 Выделение сигнала 13
2.2 Методы параметризации сигналов 15
2.2.1 Аппроксимация сигнала 15
2.2.2 Время высвечивания 16
2.2.3 Pulse shape discrimination (PSD) 24
2.2.4 Амплитуда и площадь под сигналом. 26
2.3 Подходы к разделению сигналов 27
2.3.1 Подход, основанный на разделении смеси распределений по гистограмме 27
2.3.2 Метод главных компонент 29
2.3.3 Метод композиции алгоритмов 32
3 Реализация программного средства и анализ результатов, полученных при его использовании. 35
3.1 Выбор средств и реализация 35
3.1.1 Язык программирования и среда разработки 35
3.1.2 Использованные прикладные библиотеки 35
3.2 Описание входных данных 37
3.3 Анализ результатов использования программного средства 38

3.3.1 Разделение методом главных компонент 38
3.3.2 Разделение по гистограмме PSD 42
3.3.3 Разделение по гистограмме времени высвечивания 44
3.3.4 Разделение композицией алгоритмов 46
3.3.5 Объединение результатов 48
ЗАКЛЮЧЕНИЕ 49
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 51

ВВЕДЕНИЕ

Современный подход к решению множества задач ядерной энергетики, начиная с контроля отработанного ядерного топлива и заканчивая измерением нейтринного фона в различных нейтринных детекторах, уделяет много внимания всевозможным методам регистрации отдельных составляющих исследуемого излучения. Например, часто требуется охарактеризовать излучение по доле присутствия в нём отдельных компонент, таких как нейтроны и гамма-кванты [1].

Зачастую заранее неизвестен профиль, по которому объект из общего множества может быть отнесён к какому-либо подмножеству. Возникает задача кластеризации как задача разбиения множества объектов на группы таким образом, чтобы объекты внутри обособленных групп были более похожи между собой, чем на объекты, находящиеся вне этих групп, называемых кластерами [2, 3]. Требуется построить алгоритм, способный отнести к одному из кластеров произвольный параметризованный объект из исходного множества. Обычно, для формализации такого анализа, исследуемый сигнал представляется в виде некоторого набора признаков или же метрик, являющихся функциями, зависящими от исходных значений сигналов. В качестве таких параметров могут использоваться как суперпозиции из значений сигнала в разные моменты времени, так и всевозможные статистические величины. Также признаками могут выступать величины, имеющие некоторый физический смысл в рамках исследуемой предметной области.

Для того, чтобы с уверенностью отнести объект к тому или иному кластеру, необходимо иметь достаточное количество информации о каждом из них. Для упрощения работы с большими объёмами данных, а также для улучшения качества кластеризации, было решено прибегнуть к использованию методов машинного обучения на ряду с классическими подходами к обработке сигналов подобной природы.

1 Обзор предметной области

1.1 Происхождение сигналов

В некоторых веществах в результате прохождения через них заряженных частиц возникают короткие вспышки света – сцинтилляции. Вещества, излучающие свет под действием ионизирующего излучения, называют сцинтилляторами. Сцинтилляции отличны от остальных видов свечения, образующегося во время взаимодействия частиц с веществом тем, что они возникают вследствие электронных переходов внутри так называемых центров свечения. Такими центрами свечения могут служить, например, атомы, ионы, молекулы. Каждая вспышка вызвана отдельной заряженной частицей и состоит из большого количества ($10^3 - 10^6$) фотонов.

Процесс сцинтилляции можно разбить на этапы: возбуждение вещества; перенос энергии, которую теряет частица, в веществе по направлению к центрам свечения; возбуждение центров свечения; высвечивание центров свечения [4].

Существуют следующие типы сцинтилляторов: органические кристаллы, органические жидкости, пластиковые сцинтилляторы, неорганические кристаллы, газообразные сцинтилляторы, стеклянные сцинтилляторы.

При получении данных, эксперименты с которыми производились в настоящей работе, был использован сцинтилляционный детектор, на основе органических кристаллов паратерфенила [5]. Перспективность сцинтилляторов на основе органических кристаллов паратерфенила в регистрации нейтронного излучения в присутствии гамма-фона обуславливает большое содержание атомов водорода в вышеупомянутых элементах систем детектирования. Вызвано это тем, что для регистрации нейтронов используют эффект упругого рассеяния нейтронов с ядром. При упругом рассеянии нейтронов на ядрах, наблюдается эффект отдачи кинетической энергии

$$E = \frac{4Mm}{(M+m)^2}\varepsilon * cos^2\theta \quad (1.1)$$

где m и ε – масса и энергия нейтрона соответственно; θ – угол, под которым вылетает ядро атома вещества по отношению к направлению налетающего нейтрона; M – масса ядра атома вещества. Таким образом, вещество, из которого должен быть сделан сцинтиллятор, выбирается в соответствии с тем требованием, чтобы кинетическая энергия была максимально возможной. Наибольшее значение данной энергии наблюдается для ядер водорода:

$$E = \varepsilon * cos^2\theta \quad (1.2)$$

Именно поэтому высокое содержание водорода в используемом сцинтилляторе играет весомую роль в процессе детектирования излучения, содержащего в себе нейтроны.

Для регистрации сцинтилляций, возникающих под действием отдельных ионизирующих частиц, обычно используются фотоэлектронные умножители (далее ФЭУ) [6]. Фотоны сцинтилляционной вспышки попадают на фотокатод ФЭУ и в результате фотоэффекта образуют фотоэлектроны. Фотоэлектроны, движущиеся под действием электрического поля, попадают на диноды ФЭУ. Схема детектора находится на рисунке 1.1.

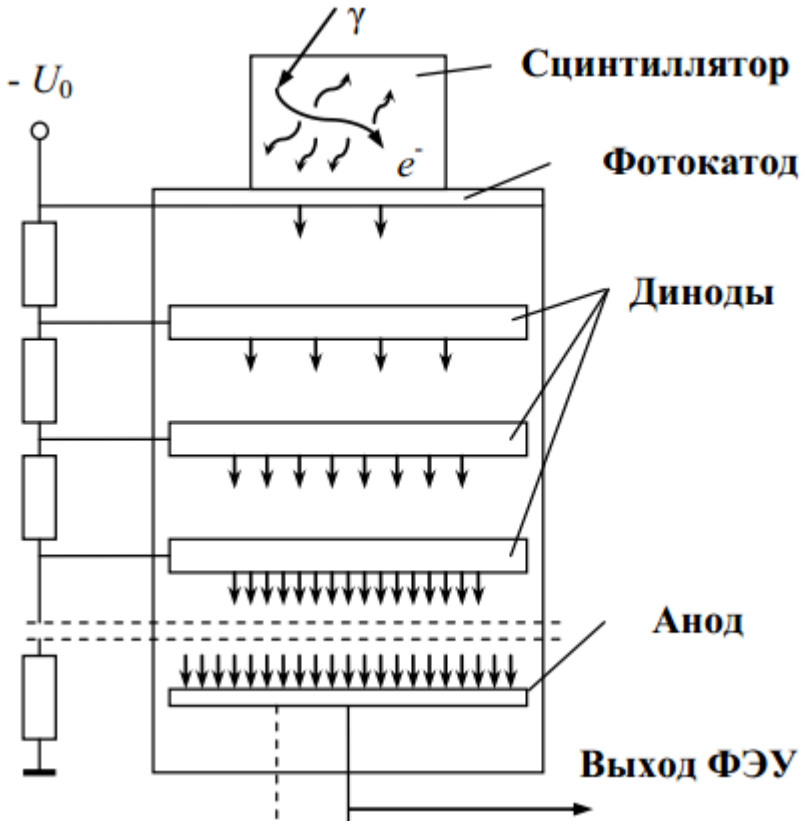

Рисунок 1.1 – Принцип действия сцинтилляционного детектора

В результате вторичной электронной эмиссии в динодах число электронов увеличивается в $10^6 - 10^9$ раз. Электроны собираются на аноде ФЭУ, вызывая электрический импульс, регистрируемый электронными схемами.

1.2 Основные параметры сцинтилляторов

1.2.2 Физическая конверсионная эффективность

Под физической конверсионной эффективностью (или же энергетическим выходом, световыходом) понимается отношение энергии вспышки к поглощенной в объёме сцинтиллятора энергии заряженной частицы. Чем выше конверсионная эффективность, тем большая доля энергии заряженной частицы преобразуется в световую вспышку и тем большая амплитуда будет у сигнала при одной и той же потерянной энергии.

$$\eta_k = \frac{E_{\text{св}}}{E_{\text{п}}} = N_{\text{ф}} \frac{h\nu_{\text{св}}}{E_{\text{п}}} \quad (1.3)$$

где $N_{\text{ф}}$ – полное число фотонов, образовавшихся в объёме сцинтиллятора под действием заряженной частицы; $E_{\text{св}}$ – энергия вспышки света; $E_{\text{п}}$ - энергия заряженной частицы, поглощенная в объёме сцинтиллятора; $h\nu_{\text{св}}$ – средняя энергия фотона сцинтилляции.

Известно, что конверсионная эффективность не является постоянной и зависит от удельных ионизационных потерь энергии частицы. Для одного и того же кристалла, амплитуда сигнала от электрона может быть в два раза выше, чем амплитуда от альфа-частицы той же энергии. Таким образом, сцинтилляционный детектор является пропорциональным только для частиц одного типа, а коэффициент пропорциональности между амплитудой на выходе ФЭУ и поглощенной энергией зависит от типа частиц.

1.2.2 Время высвечивания

Временем высвечивания τ сцинтиллятора называется время, в течение которого интенсивность свечения $dn_{\text{ф}}/dt$, то есть число фотонов во вспышке в единицу времени уменьшается в e раз. Например, если число фотонов во вспышке будет равным $N_{\text{ф}}$, а интенсивность вспышки уменьшается по экспоненциальному закону, то

$$\frac{dn_{\text{ф}}}{dt} = \frac{N_{\text{ф}}}{\tau} e^{-\frac{t}{\tau}} \quad (1.4)$$

1.2.3 Средняя энергия

Под средней энергией подразумевается энергия, расходуемая частицей на формирование одного сцинтилляционного фотона. Более формально данный параметр определяется следующим образом:

$$\omega_{\text{ф}} = \frac{E_{\text{п}}}{N_{\text{ф}}} = \frac{E_{\text{п}} h\nu_{\text{св}}}{\eta_k E_{\text{п}}} = \frac{h\nu_{\text{св}}}{\eta_k} \tag{1.5}$$

1.2.4 Техническая конверсионная эффективность

Определяется следующим соотношением:

$$\eta_{\text{кт}} = f * \eta_{\text{к}} \tag{1.6}$$

Здесь f – коэффициент, который отвечает за то, чтобы был учтён тот факт, что на фотокатод фотоэлектронного умножителя попадают не все испускаемые фотоны, образующиеся в сцинтилляторе.

С учётом всех записанных соотношений амплитуда получаемых на ФЭУ импульсов может быть записана в следующем виде:

$$A = E_{\text{п}} \frac{\eta_k f}{h\nu_{\text{св}}} \gamma M \tag{1.7}$$

где γ – эффективность фотокатода, которая равна вероятности того, что электрон вырвется из фотокатода под действием фотона; M – коэффициент усиления ФЭУ.

Если сделать допущение о том, что все коэффициенты в приведённой формуле не зависят от энергии частиц, то амплитуда на выходе должна быть пропорциональна поглощенной энергии. Однако в реальности такое допущение не подтверждается ввиду того, что физическая конверсионная эффективность зависит от ионизационных потерь частицы. Таким образом, детектор с одним и тем же сцинтиллятором может выдавать на выходе совершенно разные амплитуды для разных типов частиц, обладающих одинаковой энергией. Получаем, что сцинтилляционные детекторы обладают свойством пропорциональности только для частиц одного типа. Тем не менее, в рамках поставленной задачи это не имеет значения, более того, может сыграть на руку, когда речь идёт о кластеризации сигналов.

1.2.5 Эффективность регистрации

Под эффективностью в данном случае подразумевается отношение числа зарегистрированных частиц к общему числу частиц, попавших в сцинтиллятор.

Стоит отметить, что в связи со спецификой поставленной задачи, наибольшее внимание во время выполнения работы уделяется второй характеристике, а именно времени высвечивания.

1.3 Подход к разделению сигналов как к задаче кластеризации

1.3.1 Постановка задачи

Если подходить к вопросу, используя терминологию машинного обучения, то проще всего будет поставить задачу, рассмотрев разницу между обучением с учителем и обучением без учителя.

Когда идёт речь о задаче обучения с учителем (или же обучения на размеченных данных), то для обучающей выборки $X = (x_i; y_i)_{i=1}^{l}$ нужно найти такой алгоритм a, на котором будет достигаться минимум функционала ошибки:

$$Q(a, X) \rightarrow \min_{a}. \quad (1.8)$$

Что характерно для данного примера, каждый элемент обучающей выборки состоит из двух частей: признакового описания $x_i = (x^1, x^2, \dots, x^d)$ и ответа на объекте y_i. При этом признак – это число, которое характеризует объект, а признаковое описание является некоторым d-мерным вектором [7].

Таким образом есть объекты и истинные ответы на них. Задача же заключается в том, что по этим парам нужно восстановить общую зависимость.

Задача обучения без учителя – это такая задача, в которой есть только объекты, а ответов нет. Именно такой задачей является задача кластеризации, где по обучающей выборке, состоящей исключительно из признаковых описаний объектов, требуется построить алгоритм, расставляющий метки $y_1, y_2, \dots, y_l$ таким образом, чтобы похожие друг на друга объекты имели одинаковую метку, то есть разбить все объекты на некоторое количество групп [8, 9]. Именно поэтому при решении данного типа задачи мы лишаемся такого понятия как качество решения. Этим и различаются задачи классификации и кластеризации. При классификации также нужно делить объекты на группы, но в классификации группы, а точнее классы, фиксированы, и известны примеры объектов из разных групп [10].

1.3.2 Показатели, связанные с задачей кластеризации

Ранее было описано, почему при решении задачи кластеризации невозможно воспользоваться таким понятием, как качество решения. Однако существуют показатели, которые косвенно позволяют оценить то, насколько хорошо были разделены данные.

Calinski-Harabasz Index. Ненормированный показатель, большее значение которого присваивается лучшему разделению на кластеры. Из того, что показатель не нормирован, следует, что по нему возможно лишь построить отношение порядка между разделениями, полученными в результате обработки данных различными алгоритмами. Для данных, разделённых на k кластеров показатель рассчитывается следующим образом:

$$s(k) = \frac{Tr(B_k)}{Tr(W_k)} \times \frac{N-k}{k-1} \tag{1.9}$$

где B_k – межкластерная дисперсионная матрица, W_k – внутрикластерная дисперсионная матрица. При этом:

$$W_k = \sum_{q=1}^{k} \sum_{x \in C_q} (x - c_q)(x - c_q)^T \tag{1.10}$$

$$B_k = \sum_{q} n_q (c_q - c)(c_q - c)^T \tag{1.11}$$

где N – количество точек в данных, C_q – множество точек в кластере q, c_q – геометрический центр кластера q, c – геометрический центр данных, n_q – количество точек в кластере q.

Коэффициент силуэта. Нормированный показатель, большее значение которого присваивается лучшему разделению на кластеры. Принимает значения от -1, что соответствует некорректному разделению на кластеры, до 1, что соответствует приемлемому разделению. Также стоит отметить, что

значениям вблизи нуля соответствуют разделения с перекрывающимися кластерами. Отрицательные значения говорят о том, что большое количество точек было отнесено в неверные кластеры. Более формально коэффициент выглядит следующим образом:

$$Sil(C) = \frac{1}{N} \sum_{c_k \in C} \sum_{x_i \in c_k} \frac{b(x_i, c_k) - a(x_i, c_k)}{max\{a(x_i, c_k), b(x_i, c_k)\}} \tag{1.12}$$

где

$$a(x_i, c_k) = \frac{1}{|c_k| \sum_{x_j \in c_k} d_e(x_i, c_j)} \tag{1.13}$$

$$b(x_i, c_k) = \min_{c_l \in C \setminus c_k} \left\{ \frac{1}{|c_l| \sum_{x_j \in c_l} d_e(x_i, c_j)} \right\} \tag{1.14}$$

среднее внутрикластерное расстояние и среднее расстояние до ближайшего соседнего кластера соответственно, а d_e – обозначение евклидова расстояния между точками.

В дальнейших изысканиях будем использовать именно эти показатели ввиду того, что было показано, что решение оптимизационных задач с такими целевыми функциями на различных наборах данных показывало лучшие результаты [11].

2 Разработка и обзор алгоритмов для определения типов сигналов

2.1 Выделение сигнала

Сигналы имеют различия в периоде спада, в то время как период нарастания не несёт в себе информации, по которой можно было бы отличить одну кривую от другой ввиду того, что процесс нарастания происходит слишком быстро. Видно, что в правом и левом хвостах кривой не содержится информации, необходимой для решения задачи кластеризации. Тем не менее, по одному из этих хвостов можно вычислить такие показатели, как среднее значение и стандартное отклонение шума в детекторе. По этим характеристикам мы можем понять, где начинается и заканчивается сигнал в каждом снятом измерении.

Нулевой линией сигнала назовём среднее значение в левом хвосте кривой. Среднее значение в данном случае будет вычисляться по первым пятидесяти отсчётам.

Таким образом, сигналом в контексте поставленной задачи назовём подмножество значений кривой, содержащееся между максимумом и местом, где значение кривой начало отставать от нулевой линии на три стандартных отклонения (Считаем, что в момент, когда уровень энергии достиг такого значения, сигнал окончился, и начался шум. Даже если сигнал имеет большую амплитуду и долго затухает, в момент, когда он приближается к нулевой линии, шум начинает вносить слишком большой вклад в текущие показания детектора). Формально область, названную сигналом, можно обозначить следующим образом:

$$Signal_j = Raw\ Data_j \left[i_{max}; i_{m_j + 3\sigma_j} \right] \quad (2.1)$$

где $Raw\ Data_j$ – упорядоченное множество, соответствующее исходной кривой; $Signal_j$ – выделенный сигнал; i_{max} – индекс элемента множества, соответствующего максимальному значению; $i_{m_j+3\sigma_j}$ – индекс элемента множества, соответствующего значению, отстающему на три стандартных отклонения от нулевой линии; операция $Set[a; b]$ аналогична операции среза в структуре данных под названием массив (результатом является подмножество из Set, заключенное между элементами с индексами a и b).

Однако, выше представлен не единственный допустимый способ выделения сигнала. Временную зависимость интенсивности высвечивания можно представить в виде суммы двух компонент, экспоненциально спадающих с постоянным временем. Для каждый из компонент это время будет своим, из-за этого они условно названы быстрой и медленной. Описанная зависимость может быть формализована следующим образом [12]:

$$h(t) = u_1 e^{\frac{-t}{\tau_б}} + u_2 e^{\frac{-t}{\tau_м}} \quad (2.2)$$

где $\tau_б, \tau_м$ – времена высвечивания быстрой и медленной компонент соответственно; u_1, u_2 – коэффициенты интенсивности компонент.

Для того, чтобы сигнал было проще параметризовать, его началом в некоторых случаях может считаться не максимум кривой, а некоторый такт, который стоит уже после максимума. Способ выбора этого такта зависит от условий поставленной задачи. В настоящей работе отступ будет задаваться двумя способами: начало сигнала фиксируется спустя определённое количество тактов, заданное вручную (в таком случае, вне зависимости от вида исходных данных, сигнал будет начинаться, например, через три отсчёта от максимума) либо началом сигнала будет считаться отсчёт, отстоящий от максимума по амплитуде на определённое количество процентов (в таком

случае число отсчётов, которое будет отступаться от максимума – число, меняющееся в зависимости от исходных данных).

То, каким образом выбирать конкретный способ выделения сигнала, какой процент амплитуды или какое число отсчётов необходимо отступить – выбирается вручную методом подбора. Все подобные решения принимаются исходя из максимизации качества разделения сигналов. Конкретные примеры подбора параметров будут приведены ниже.

2.2 Методы параметризации сигналов

2.2.1 Аппроксимация сигнала

Как было упомянуто ранее, целесообразнее всего аппроксимировать полученные сигналы экспоненциальной зависимостью. В таком случае одним из параметров этой зависимости в явном виде будет время высвечивания сцинтиллятора.

На рисунке 2.1 приведён пример сигнала и его аппроксимации:

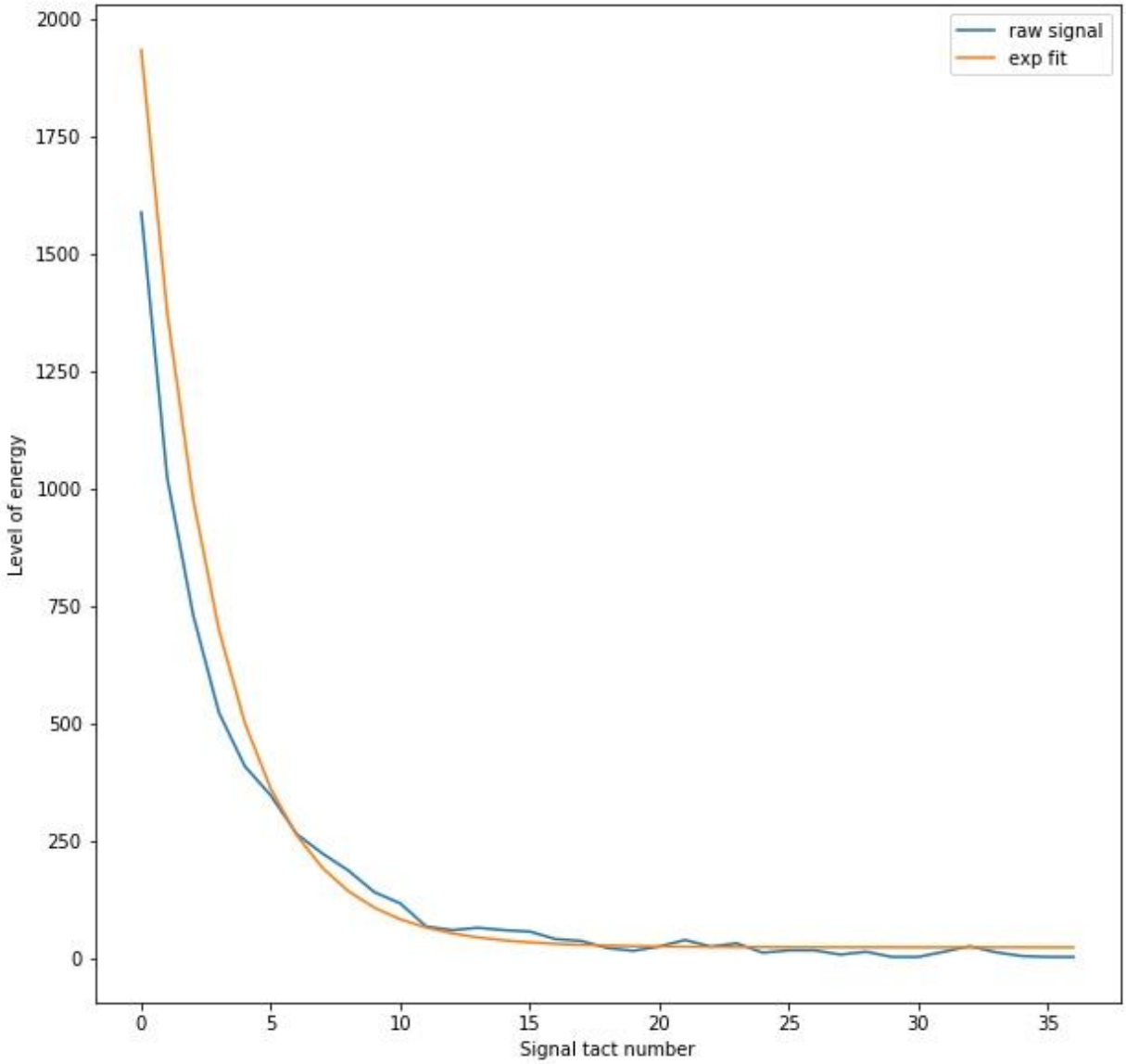

Рисунок 2.1 – Сигнал и его экспоненциальная аппроксимация

2.2.2 Время высвечивания

Классическим методом иллюстрации множества сигналов, иллюстрирующим присутствие сигналов различного происхождения, является изображение гистограммы для времени высвечивания. На такой визуализации можно заметить наличие пиков гистограммы, что свидетельствует о том, что приходящие на детектор сигналы в среднем имеют разное характерное время высвечивания.

Как было описано ранее, время высвечивания является одном из параметров при условии, если сигнал аппроксимируется экспоненциальной зависимостью. То, какого качества будет аппроксимация, а далее и разделение сигналов по такому методу, в большой степени зависит от правильного выбора отступа от начала сигнала. Это обстоятельство вызвано тем, что сигнал представляет из себя сумму двух экспонент, и в начале сигнала наибольший вклад вносит быстрая компонента. Одним из наиболее наглядных критериев при подборе величины отступа является высота перешейка между поучаемыми пиками гистограммы.

Таким образом, если проводить аппроксимацию, отступая от максимума сигналов на разные доли амплитуды, можно получать разное качество разделения. В данном случае, о качестве разделения будет свидетельствовать высота точки, которая на рисунке 2.2 и 2.3 названа «split point» или же разделяющая точка.

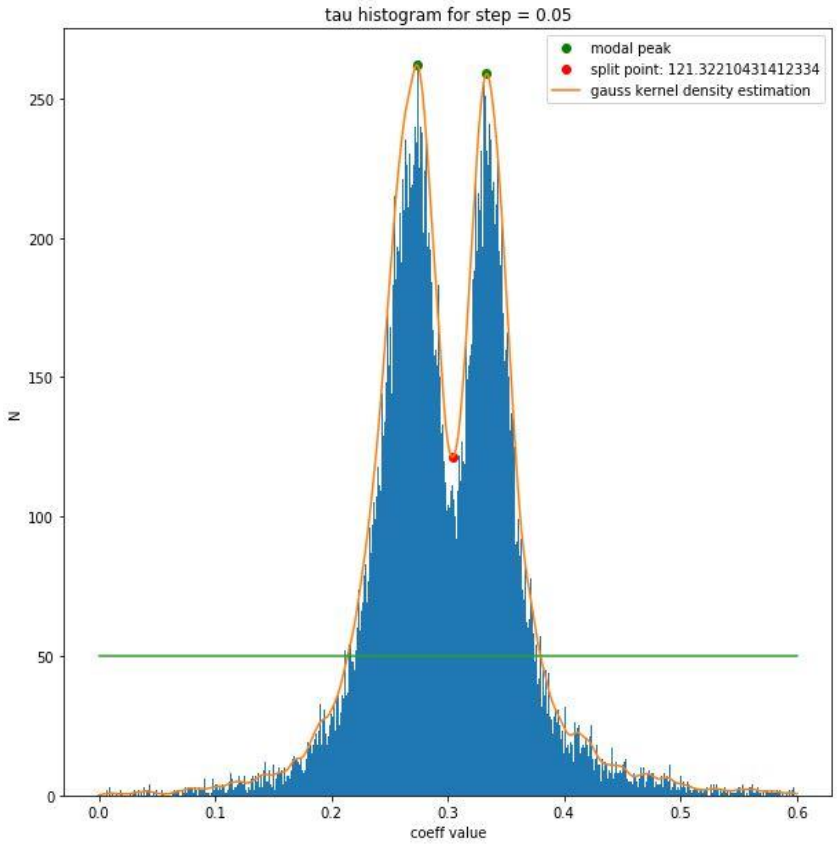

Рисунок 2.2 - Вид гистограммы времени высвечивания для отступа в 5%

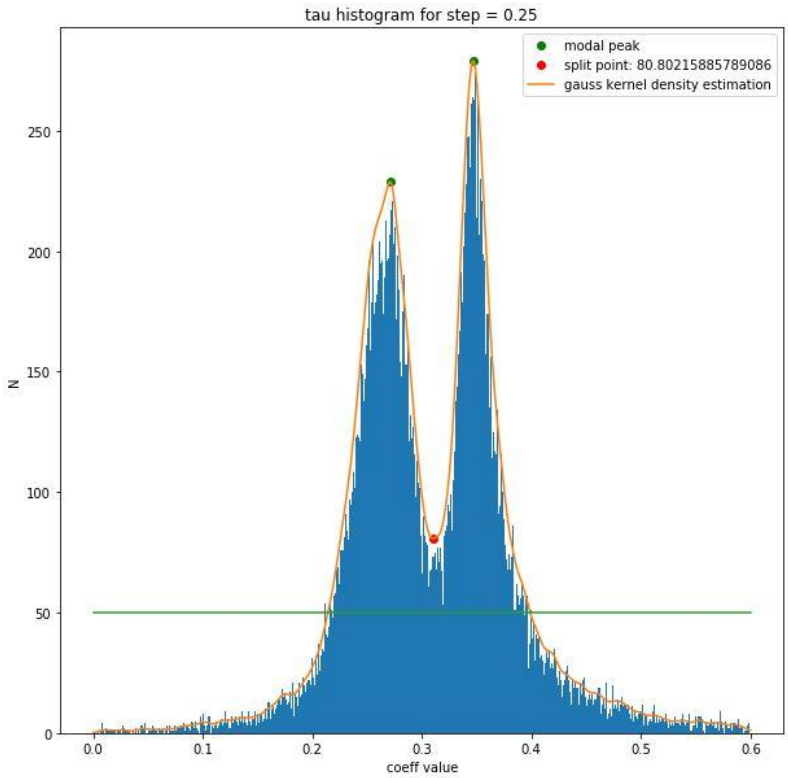

Рисунок 2.3 - Вид гистограммы времени высвечивания для отступа в 25%

Как видно из рисунков, первое время качество разделения растёт с увеличением отступа, однако далее наступит момент, когда эта закономерность перестанет работать. На рисунке 2.4 приведён график зависимости для высоты разделяющей точки от доли амплитуды, на которую отступили от максимума сигнала.

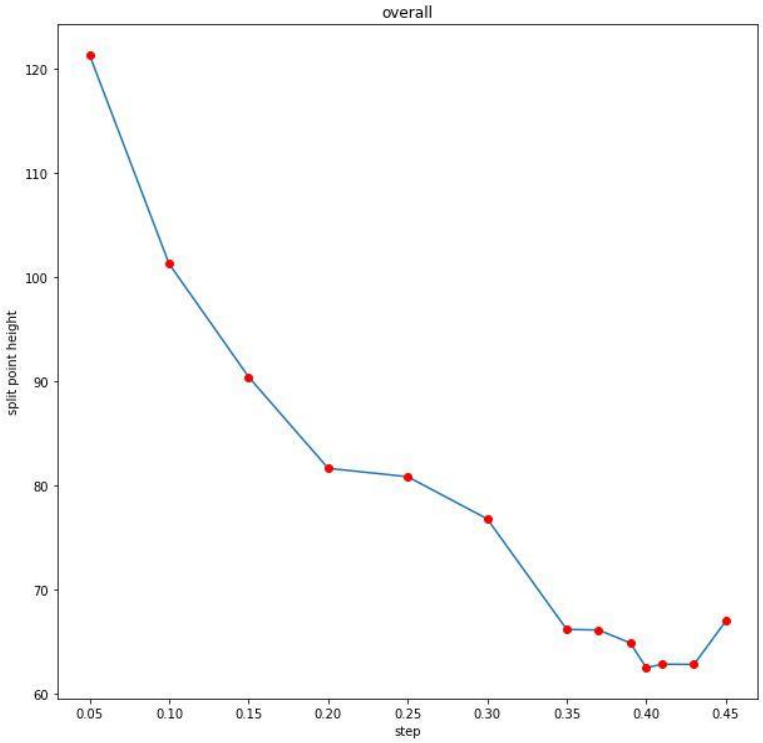

Рисунок 2.4 – График зависимости высоты разделяющей точки от отступа от максимума сигнала

Таким образом, наилучшее разделение получается, если отступать от начала сигнала 40% его амплитуды. Далее, каждый раз, когда будет заходить речь о разделении с использованием гистограммы для времени высвечивания, будет подразумеваться, что получена она была путём извлечения множества значений параметров из экспоненциальных зависимостей, где первоначальные данные были получены путём выделения сигналов с отступом от максимума в размере 40% от амплитуды.

На рисунке 2.5 приведена гистограмма времени высвечивания для такого отступа:

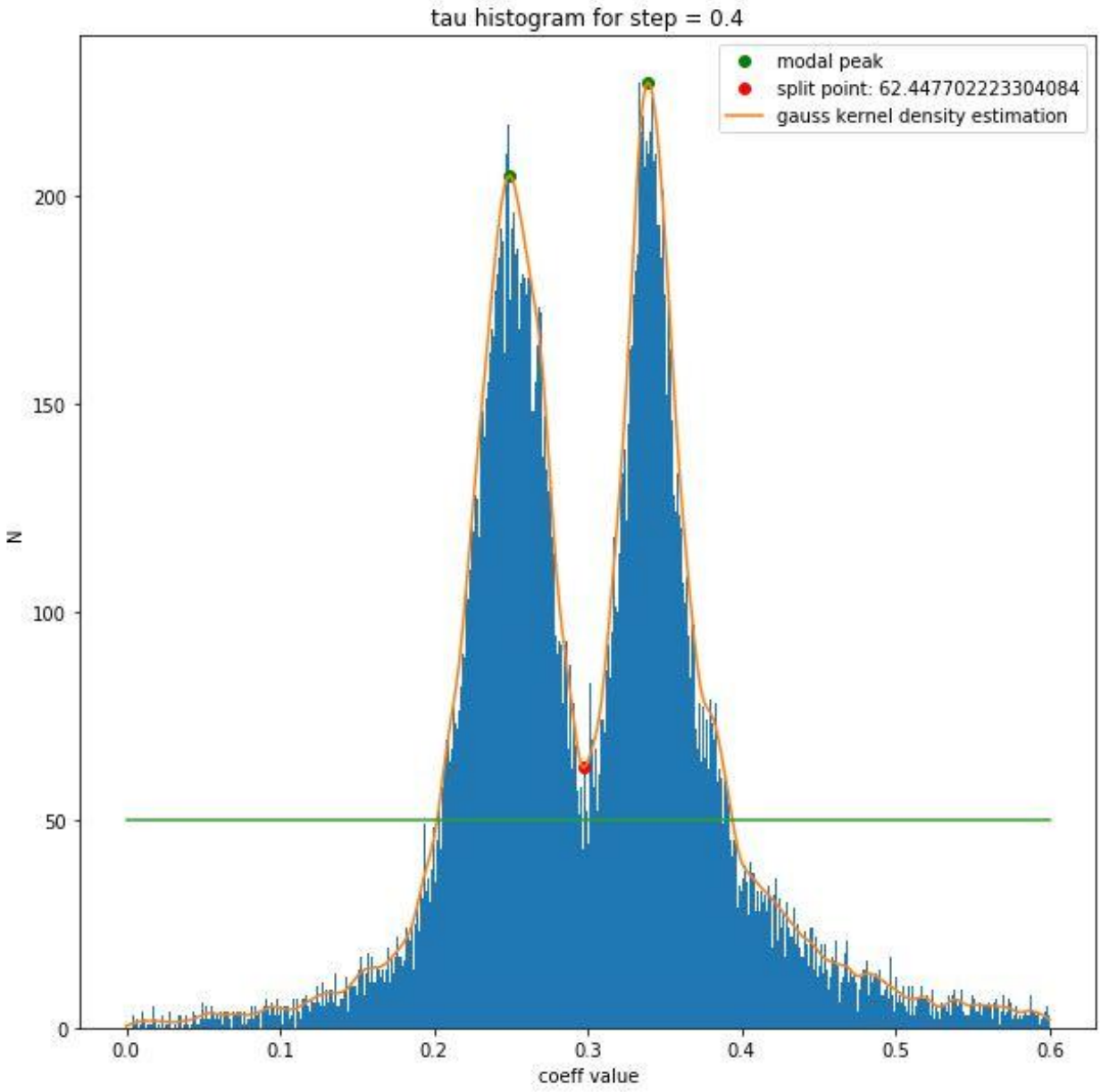

Рисунок 2.5 – Вид гистограммы времени высвечивания для отступа в 40%

Ядерная оценка при нахождении разделяющей точки.

Разделяющая точка для каждой из гистограмм находится при помощи ядерного сглаживания, заключающегося в решении задачи восстановления регрессии.

Постановка задачи выглядит следующим образом. Пусть задано пространство объектов X и множество ответов Y. Существует неизвестная

целевая зависимость $y^*: X \to Y$, значения которой известны только на обучающей выборке $X^m = (x_i; y_i)_{i=1}^m$. Требуется построить алгоритм $a: X \to Y$, аппроксимирующий целевую зависимость.

Принцип решения задачи заключается в представлении последовательности весов $\{W_{mi}(x)\}_{i=1}^m$ и описании формы весовой функции $W_{mi}(x)$ посредствам функции плотности со скалярным параметром, который регулирующим размер и форму весов около x. Именно эту функцию и принято называть ядром. Полученные таким образом веса далее используются для представления величины $a(x)$ в виде взвешенной суммы значений y_i обучающей выборки.

Последовательность весов для ядерных оценок в одномерном случае определяется следующим образом [13]:

$$W_{mi}(x) = \frac{K_{h_m}(x - X_i)}{\hat{f}_{h_m}(x)} \tag{2.3}$$

где

$$\hat{f}_{h_m}(x) = \frac{1}{m}\sum_{i=1}^{m} K_{h_m}(x - X_i) \tag{2.4}$$

$$K_{h_m}(u) = \frac{1}{h_m} K\left(\frac{u}{h_m}\right) \tag{2.5}$$

представляет собой ядро с параметром h_m. Этот параметр называют шириной окна, и именно он регулирует размер и форму весов в окрестности точки.

Функция $\hat{f}_{h_m}(x)$ является ядерной оценкой Парзена – Розенблатта для плотности переменной x. Как видно на рисунке 2.6, этому методу можно дать неформальную трактовку.

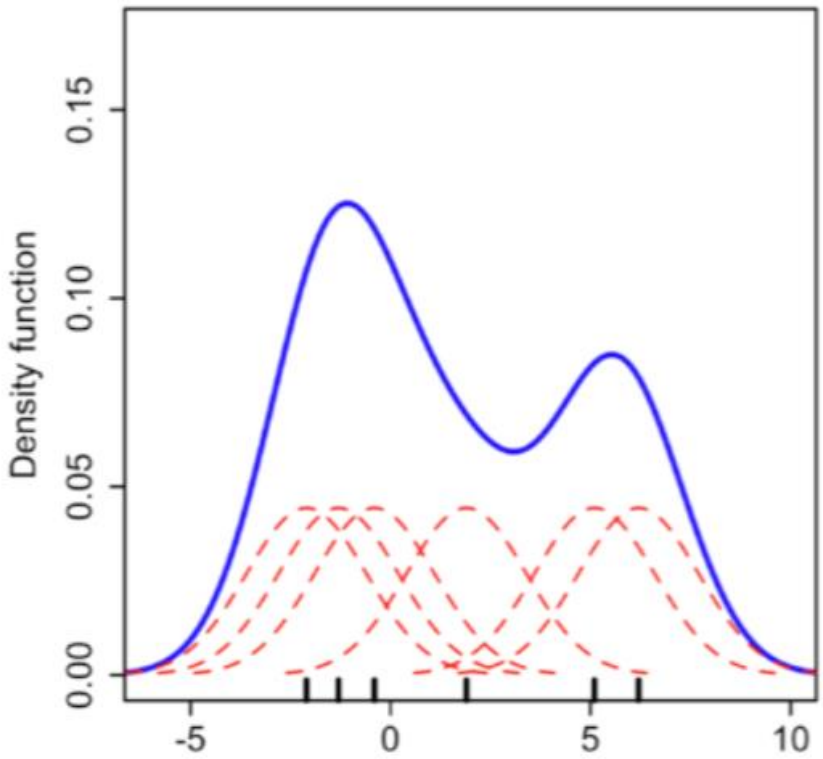

Рисунок 2.6 – Пример непараметрического восстановления плотности

Пусть имеется одномерная выборка (объекты выборки обозначены засечками по оси абсцисс). В каждой точке обучающей выборки помещают центр небольшой гауссианы (показаны красной линией), таким образом всем точкам на оси присваиваются некоторые вероятности. Далее, в каждой точке эти гауссианы суммируются, и получается итоговое распределение (на рисунке показано синим). Именно это делается при помощи формулы Парзена – Розенблатта.

Остановимся подробнее на таком параметре как ширина окна.

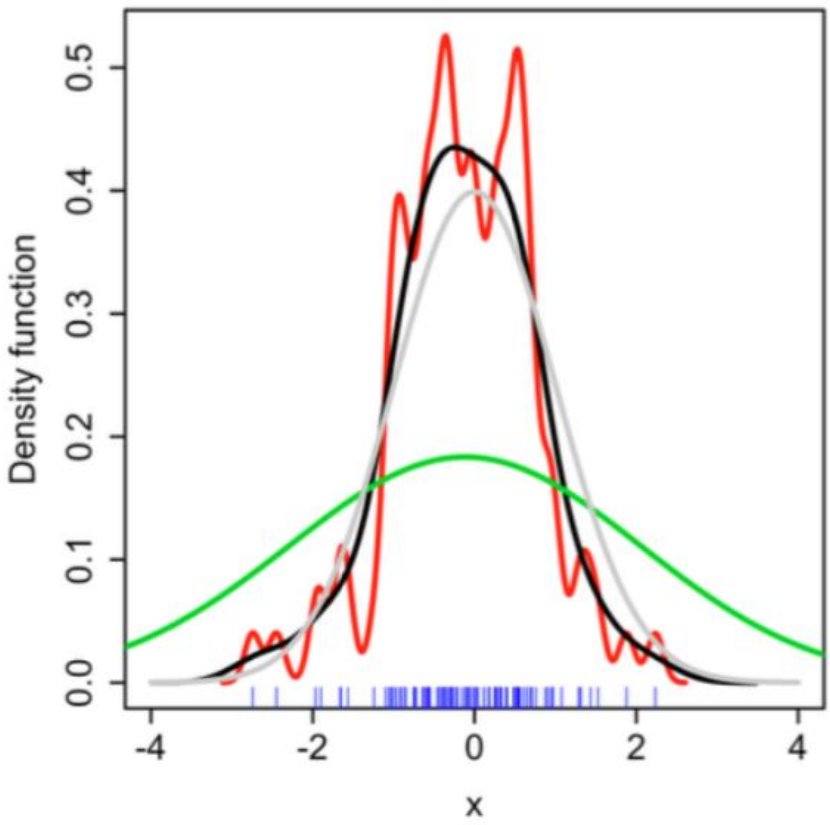

Рисунок 2.7 – Влияние ширины окна на вид восстанавливаемой функции

Влияние этого параметра на результирующую плотность вероятности можно рассмотреть на примере выборки, изображённой на рисунке 2.7. Элементы выборки изображены синими засечками на оси абсцисс. Красной, черной и зеленой кривыми показаны непараметрические оценки с гауссовским ядром для разных значений ширины окна. Красная кривая соответствует малой ширине окна, результирующая плотность чувствительна к ближним точкам. Она получается не очень гладкой и, скорее всего, переобученной. Черная кривая соответствует более высокому значению ширины окна. Эта плотность очень неплохо восстанавливает нормальное распределение, из которого были сгенерированы данные (на рисунке показана серым цветом). Зеленая кривая соответствует оценке плотности с очень большой шириной окна, результирующая плотность характеризуется большой дисперсией.

2.2.3 Pulse shape discrimination (PSD)

Ещё одним способом получить распределение, иллюстрирующее наличие двух разновидностей сигналов является построение PSD гистограммы. Показатель PSD рассчитывается следующим образом [14]:

$$PSD = \frac{long - short}{long} \tag{2.6}$$

где long и short – площади под определёнными участками сигнала, которые выбираются таким образом, что long чаще всего соответствует весь сигнал, а short лишь небольшой его фрагмент в окрестности максимума (то есть в самом начале наших сигналов).

Идея метода заключается в том, чтобы параметризовать форму сигналов-пульсаций, исходя из того, что сигналы, чья природа возникновения в данном случае сходна, будут иметь различное «распределение» площади. То есть, как показано на рисунке 2.8, если сигнал обладает более пологой формой, то площадь, приходящаяся на так называемый long gate будет доминировать в соотношении, характеризующем данный показатель, он будет стремиться к единице. Сигнал, имеющий более острый пик, получает меньшее значение показателя. Стоит отметить, что параметризуется именно форма сигнала ввиду того, что показатель нормирован. Таким образом при анализе разделения по показателю PSD получается игнорировать тот факт, что физическая конверсионная эффективность зависит от ионизационных потерь частицы, и для частиц разного типа с одинаковой энергией может получиться совершенно разная амплитуда сигнала.

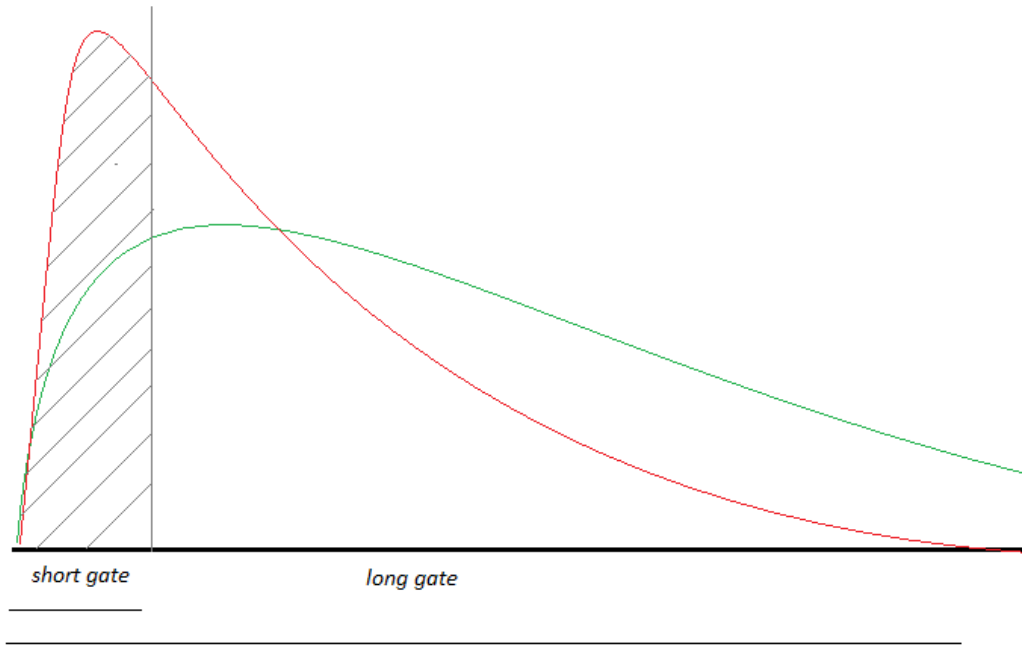

Рисунок 2.8 – Иллюстрация различия PSD для сигналов разной формы

В данной работе long соответствует весь сигнал. Short подбирается экспериментальным путём. Начало окна для short совпадает с началом long. Отметим, что длина short подбирается исходя из тех же соображений, что подбирался отступ от максимума сигнала при построении гистограммы для времени высвечивания. Стоит заострить внимание на том, что для корректного подсчёта PSD от максимума сигнала также необходимо отступить, однако в данном случае отступ будет производиться не по доле амплитуды, а по времени. Таким образом для того, чтобы построить качественную гистограмму для PSD, необходимо варьировать сразу два параметра: длину short gate и количество тактов, отступаемых от максимума. Ввиду того, что рассуждения о выборе подходящей гистограммы повторяют рассуждения, приведённые в пункте 2.3.1, пропустим этот пункт и перейдём сразу к виду полученной гистограммы.

По результатам вычислений получаем гистограмму на рисунке 2.9:

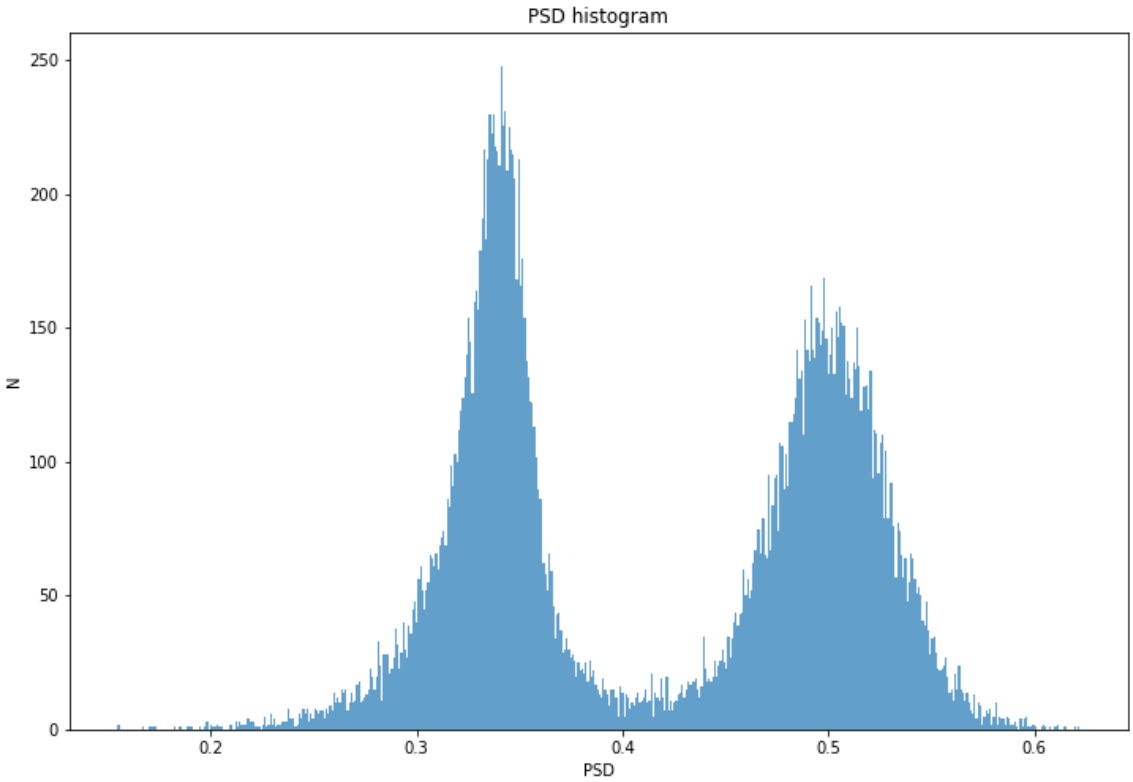

Рисунок 2.9 – Гистограмма для PSD.

2.2.4 Амплитуда и площадь под сигналом.

Построим диаграмму рассеяния с отображением плотности точек для зависимости амплитуды сигнала от площади под ним, для большей наглядности приведём её масштабированную версию на рисунке 2.10:

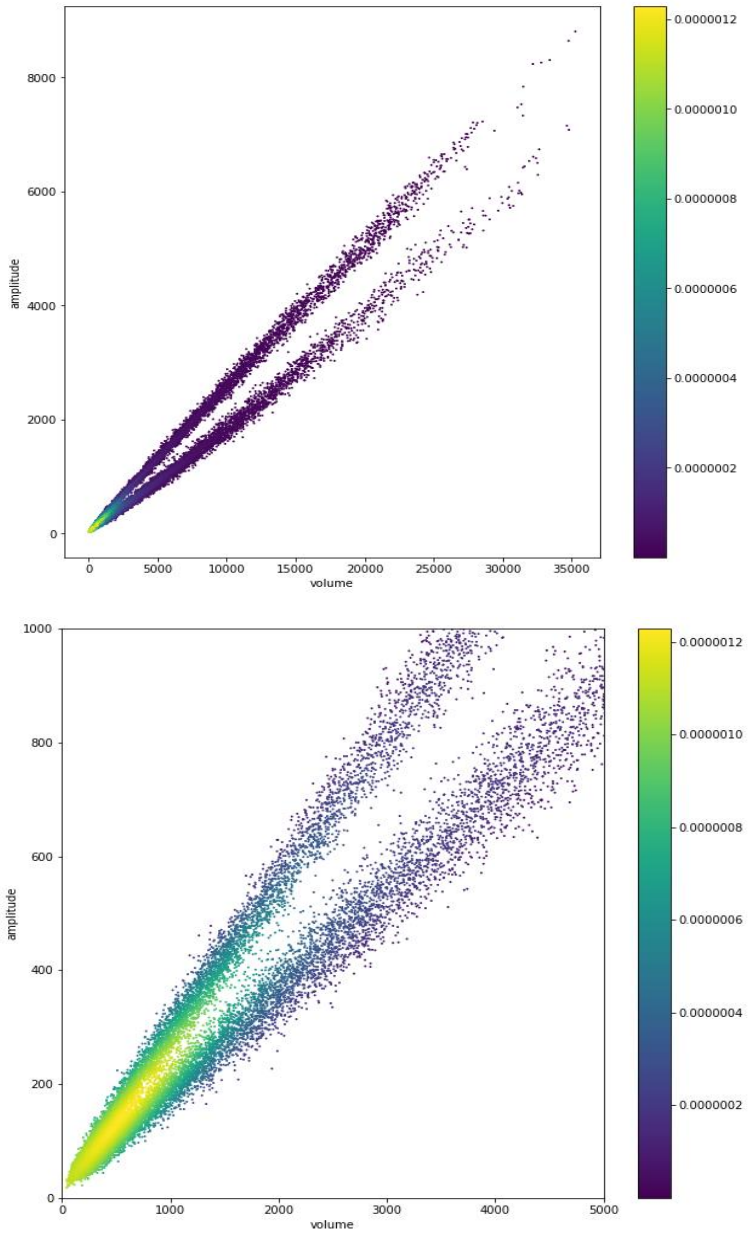

Рисунок 2.10 – Диаграммы рассеяния для зависимости амплитуды от площади под сигналом с отображением плотности точек.

2.3 Подходы к разделению сигналов

2.3.1 Подход, основанный на разделении смеси распределений по гистограмме

Классический подход применяется, когда речь идет о разделении сигналов на основании их различий во времени высвечивания или различий в

PSD. Заключается он в выборе порогового значения из области между пиками гистограммы, после чего все сигналы, имеющие значение рассматриваемого показателя ниже определённого ранее порога, относятся к одному кластеру, а те, что имеют значение выше – к другому. На примере с гистограммой для времени высвечивания отметим, что пороговое значение принимается равным значению абсциссы точки разделения, обозначенной на рисунке 2.5 как split point. Аналогичные рассуждения верны и для PSD.

Однако при всей своей простоте у данного метода есть очевидный недостаток. При таком подходе к разделению неизбежно совершается ошибка по отношению к сигналам, чьи значения соответственных коэффициентов находятся вблизи порога разделения. Происходит это из-за того, что получаемые гистограммы представляют собой ничто иное как смесь некоторых распределений, близких к нормальным. Схематично это можно было бы проиллюстрировать следующим образом:

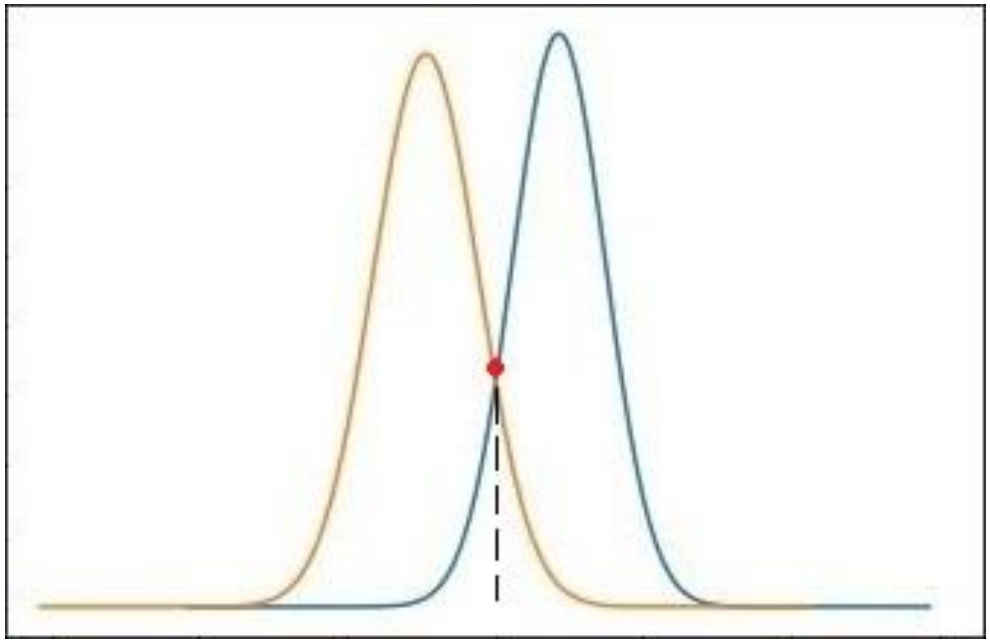

Рисунок 2.11 – Схематичное представление разделяемых распределений.

Как видно из рисунка 2.11 разделении подобных гистограмм, мы не можем сделать ничего лучше, чем выбрать порог в точке пересечения между

пиками (на рисунке точка отмечена красным) и при этом неизбежно ошибиться в выводах относительно принадлежности некоторого числа точек по одну и по другую сторону от порога разделения.

2.3.2 Метод главных компонент

Метод главных компонент – широко используемый метод понижения размерности, заключающийся в аппроксимации данных линейными многообразиями меньшей размерности. Однако в данном случае рассматриваемый метод не будет использован в качестве способа понизить размерность данных. Если обратить внимание на то, как выглядят исходные данные на рисунке 2.10, то можно сделать предположение о том, что итоговое линейное многообразие (в нашем случае просто прямая) будет некоторым образом разделять выборку на две части, проходя между двумя «хвостами». О том, что прямая будет проходить именно таким образом, свидетельствуют постановки задачи метода главных компонент. Следовательно, полученная прямая может служить естественной границей раздела для двух кластеров.

Существует несколько постановок задачи, которые приводят к одному и тому же результату и являются полностью эквивалентными [15]:

- Поиск подпространства меньшей размерности, в ортогональной проекции на которое дисперсия выборки будет максимальной
- Поиск такого подпространства меньшей размерности, чтобы сумма ошибок проецирования по всей выборке на это подпространство была минимальной
- Поиск подпространства меньшей размерности, где каждый новый признак будет линейно выражаться через исходные

Приведём пояснения к первой из постановок. Формально дисперсию выборки после проецирования можно записать следующим образом:

$$\sum_{j=1}^{d} w_j^T X^T X w_j \rightarrow max_W \quad (2.7)$$

Где X – матрица размерности (m, n) – исходная выборка; W – искомая матрица размерности (n, d), составленная из направляющих векторов, задающих новое пространство. Причём такое выражение будет означать выборочную дисперсию только в том случае, если выборка центрирована. Также вводится ограничение, обеспечивающее наличие единственного решения:

$$W^T W = I \quad (2.8)$$

В контексте решаемой задачи нам потребуется выделить первую главную компоненту, перепишем условие задачи для случая, когда вся выборка проецируется на одно направление:

$$\begin{cases} w_1^T X^T X w_1 \rightarrow max_{w_1} \\ w_1^T w_1 = 1 \end{cases} \quad (2.9)$$

Для решения задачи необходимо выписать лагранжиан:

$$L(w_1, \lambda) = w_1^T X^T X w_1 - \lambda(w_1^T w_1 - 1) \quad (2.10)$$

Продифференцируем лагранжиан по искомой величине:

$$\frac{dL}{dw_1} = 2X^T X w_1 - 2\lambda w_1 = 0 \qquad (2.11)$$

После преобразований получаем:

$$X^T X w_1 = \lambda w_1 \qquad (2.12)$$

Отсюда следует, что w_1 – собственный вектор для матрицы $X^T X$. Подставив полученное выражение в функционал решаемой задачи получим, что дисперсия выборки после проецирования в точности равна собственному значению, соответствующему выбранному собственному вектору:

$$w_1^T X^T X w_1 = \lambda \qquad (2.13)$$

Таким образом, поскольку требуется максимизировать дисперсию, необходимо выбирать максимальное собственное значение и собственный вектор, который соответствует этому значению.

Итак, получили, что первая компонента — это собственный вектор ковариационной матрицы $X^T X$, который соответствует наибольшему собственному значению этой матрицы.

2.3.3 Метод композиции алгоритмов

Одним из способов нивелировать недостатки и упрочить сильные стороны при отнесении сигналов к одному или другому кластеру является построение композиции алгоритмов. В основе такого подхода лежит то, что для построения вывода о принадлежности сигнала к кластеру используется информация, полученная от нескольких алгоритмов.

Подойдём к вопросу более формально. Композицией алгоритмов назовём объединение N алгоритмов $f_1(x), f_2(x), \dots, f_N(x)$ в один. Суть заключается в независимом построении $f_1(x), f_2(x), \dots, f_N(x)$, а затем в усреднении их ответов.

$$a(x) = \frac{1}{N}\sum_{k=1}^{N} f_k(x) \tag{2.14}$$

Если бы перед нами стояла задача регрессии (то есть задача предсказания действительного числа), то данное выражение давало бы нам искомый ответ. В нашем случае искомым ответом будет знак от получившегося выражения (в случае, если метки кластеров будут 1 и -1):

$$a(x) = sign\frac{1}{N}\sum_{k=1}^{N} f_k(x) \tag{2.15}$$

Алгоритм $a(x)$, возвращающий среднее или его знак, называется композицией N алгоритмов $f_1(x), f_2(x), \dots, f_N(x)$, а они сами называются базовыми алгоритмами.

Проще говоря, объект относится к одному или другому кластеру тогда, когда за него «голосует» большинство базовых алгоритмов. В нашем случае базовыми алгоритмами будут являться алгоритмы, разделяющие сигналы на основании: времени высвечивания, коэффициента PSD, метода главных компонент.

Для того, чтобы понять причины, почему композиции алгоритмов могут помочь при решении задач, разберём следующие понятия [16]:

- Шум – компонента ошибки, появляющаяся даже при применении идеальной модели. То есть, шум является характеристикой самих данных и будет иметь место вне зависимости от того, какая модель применяется.
- Разброс – дисперсия ответов моделей, которые были построены по различным выборкам данных (естественно при условии, что все выборки были получены из одного и того же распределения). Другими словами, разброс является мерой того, насколько сильно прогноз модели зависит от исходных данных и того факта, что они не идеально отображают генеральную совокупность.
- Смещение – отклонение усредненного по различным подвыборкам прогноза полученной модели от прогноза идеальной модели.

Проще всего продемонстрировать смысл данных понятий на следующем примере:

Рисунок 2.12 – Иллюстрация влияния разброса и смещения на предсказания модели

Таким образом, как показано на рисунке 2.12, даже если ответы моделей будут неточны ввиду наличия разброса, то в среднем они всё равно будут «попадать в десятку». Именно этой цели и служит построение композиции. Стоит отметить, что чаще всего разговор о подобных понятиях и методах ведётся в рамках решения задач, которые относятся к задачам обучения с учителем [17]. Однако интуитивно понятно, что в задаче кластеризации, когда мы не обладаем знаниями, какие примеры из выборки к каким кластерам отнесены, такой подход также может быть применим. Различие же заключается в том, что в первом случае куда проще оценить эффект применения композиции.

3 Реализация программного средства и анализ результатов, полученных при его использовании.

3.1 Выбор средств и реализация

3.1.1 Язык программирования и среда разработки

В качестве основного средства реализации был выбран такой язык программирования как Python.

Несмотря на то, что Python является интерпретируемым языком и не отличается быстродействием, он имеет ряд неоспоримых доказательств, таких как: простота синтаксиса и написания кода; простота чтения кода; большое количество доступных прикладных библиотек под самые разные нужды и требования; активное сообщество, развивающее язык и всегда готовое прийти на помощь как новичку, так и профессионалу, у которого появились затруднения [18].

В качестве среды разработки был выбран Jupyter, его основными преимуществами являются широкие возможности в части визуализации и постоянного проведения экспериментов. Это обусловлено модульной структурой рабочего пространства, которая позволяет разделять код по ячейкам, наблюдая, что происходит при выполнении кода в конкретной ячейке без ограничения области видимости переменных.

3.1.2 Использованные прикладные библиотеки

Ниже будут приведены названия библиотек, использованных в процессе разработки, с краткими описаниями:

- NumPy – основным назначением данной библиотеки является предоставление пользователю возможности работы с многомерными массивами. Библиотека содержит в себе расширение стандартной функциональности языка в области типов данных и работы с ними. Также стоит отметить, что введённые типы данных и операции с ними

используются в абсолютном большинстве библиотек Python и фактически являются стандартом в области обработки и анализа данных [19, 20]. Написана библиотека NumPy на языке C, что является причиной того, что многие даже простые вычисления, проведённые при помощи данной библиотеки, выполняются быстрее, чем аналогичные, проведённые стандартными средствами Python.

- Pandas – библиотека предназначена для работы с табличными данными. Основными преимуществами являются удобство сбора, очистки и прочих преобразований данных [19, 20]. Тесно связана с NumPy, являясь её надстройкой. Несмотря на то, что библиотека Pandas собрана поверх NumPy, её наиболее важные части также написаны на низкоуровневом языке программирования C, что обеспечивает её быстродействие. Прежде всего данная библиотека предназначена для первичной компоновки и оценки данных.
- SciPy – прикладная библиотека для выполнения научных и инженерных расчётов, имеющая в своём арсенале широкий спектр средств для: поиска экстремумов функций; вычисления интегралов; работы со специальными функциями; обработки сигналов; обработки изображений; работы с генетическими алгоритмами и так далее [19, 20]. Наиболее важные части данной библиотеки написаны на языках: C, C++, Fortran.
- Scikit-learn – библиотека, применяемая для работы с алгоритмами машинного обучения и всеми аспектами, окружающими эту предметную область [19, 20]. Написана на языках программирования: C, C++. Помимо того, что в библиотеке имеются реализации всех основных алгоритмов машинного обучения и средств работы с нейронными сетями, Scikit-learn обладает очень подробной документацией с большим количеством практических примеров, теоретическими сносками и пояснениями, что делает возможным освоение машинного

обучения посредствам ознакомления с вышеупомянутой документацией.

- Matlpotlib – библиотека, предназначенная для визуализации данных, написана на C++. Содержание библиотеки вдохновлено основными возможностями по визуализации, доступными в MATLAB. В качестве установленных по умолчанию доступны: обыкновенные графики, диаграммы рассеяния, гистограммы, круговые диаграммы, поля градиентов и так далее [19, 20].

3.2 Описание входных данных

Был получен текстовый файл, внутри которого находятся табличные данные формата 23000 x 504. В каждой из 23000 строк содержится 504 значения, первые 4 из которых являются метаданными, описывающими состояние ФЭУ в момент получения сигнала. Затем идёт 500 значений, описывающих показания детектора. При визуализации данных для разных строк, были получены изображения как на рисунке 3.1:

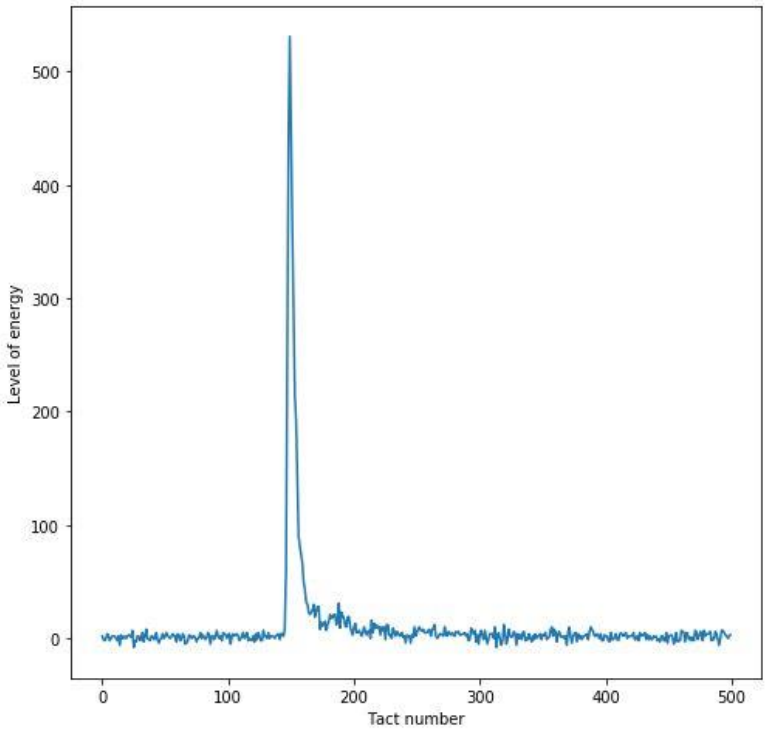

Рисунок 3.1 – Визуализация исходных данных

3.3 Анализ результатов использования программного средства

3.3.1 Разделение методом главных компонент

Ранее было выдвинуто предположение о том, что первая главная компонента, построенная по данным в пространстве, где по оси абсцисс отложена площадь под сигналом, а по оси ординат – амплитуда сигналов, может дать некоторое адекватное разделение данных на два подмножества. Применив описанный алгоритм, позволяющий получить разделение методом главных компонент, получаем следующий результат:

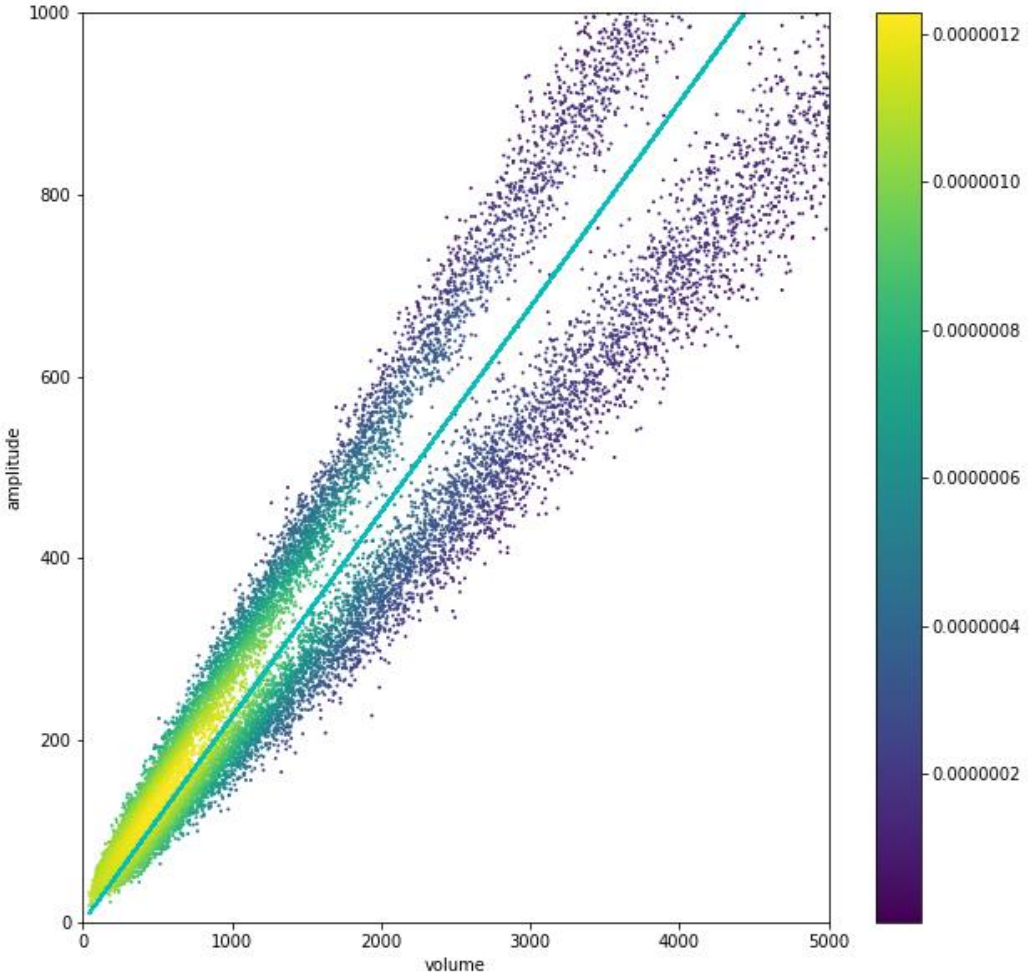

Рисунок 3.2 – Диаграмма рассеяния для зависимости амплитуды от площади под сигналом с отображением первой главной компоненты.

Разделение, проиллюстрированное на рисунке 3.2, будет выглядеть следующим образом:

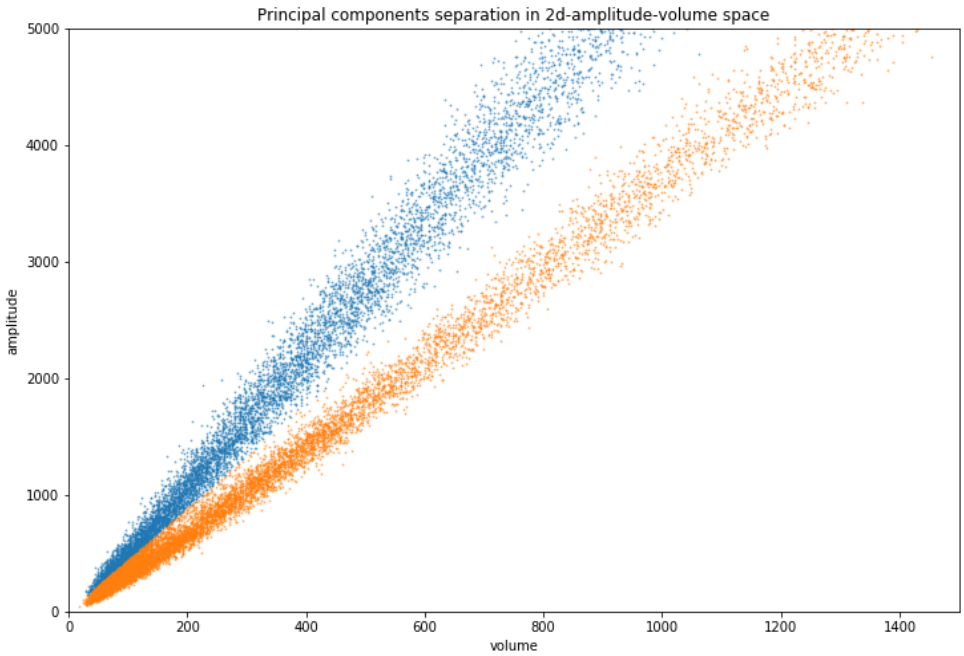

Рисунок 3.3 – Разделение методом главных компонент в двумерном пространстве амплитуды и площади под сигналом

Вспомним, что каждой точке в текущем пространстве, изображенном на рисунке 3.3, соответствует сигнал со своим показателем времени высвечивания и своим значением коэффициента PSD. Построим соответствующие гистограммы с учётом разделения, полученного на текущем шаге:

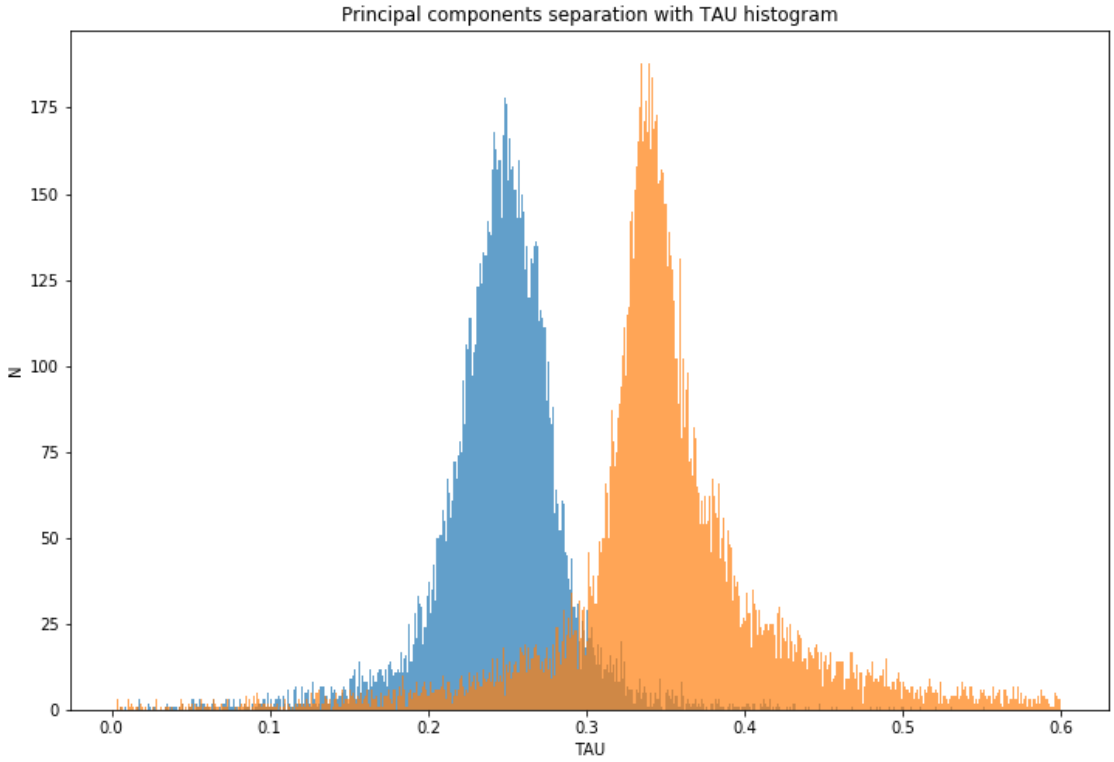

Рисунок 3.4 – Гистограммы для времени высвечивания, полученные при помощи метода главных компонент.

Из рисунка 3.4 видно, что первая главная компонента в пространстве амплитуды и площади под сигналом позволяет получить результат разделения, более соответствующий нашим представлениям о природе данных. Отчётливо просматриваются два распределения, с накладывающимися хвостами. Визуализация разделения посредствам гистограмм PSD изображена на рисунке 3.5:

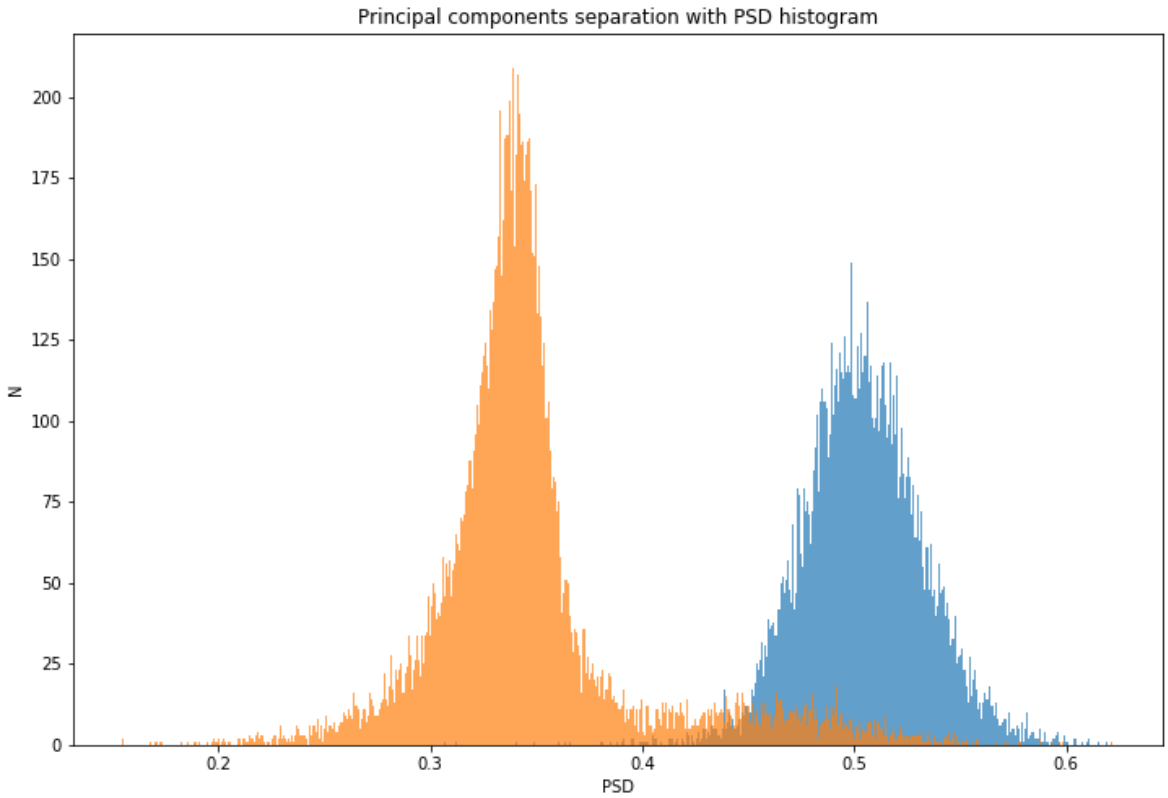

Рисунок 3.5 – Гистограммы PSD, полученные при помощи метода главных компонент.

Calinski-Harabaz score для такого разделения составил 93.82.

Коэффициент силуэта равен 0.027

3.3.2 Разделение по гистограмме PSD

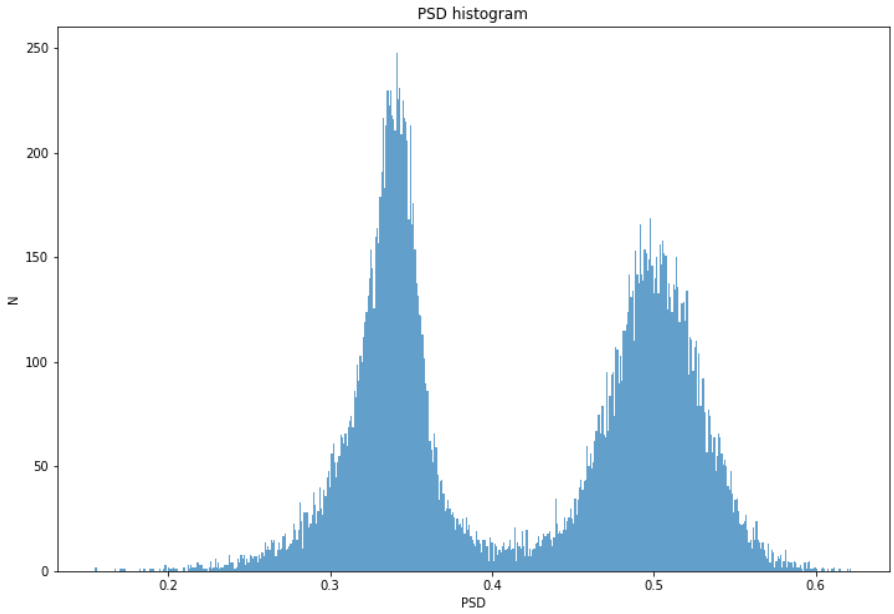

Рисунок 3.6 – Гистограмма PSD

Проведя вертикаль по гистограмме с рисунка 3.6, проиллюстрируем разделение посредствам гистограмм для времени высвечивания и разделения в двумерном пространстве амплитуды и площади под сигналом.

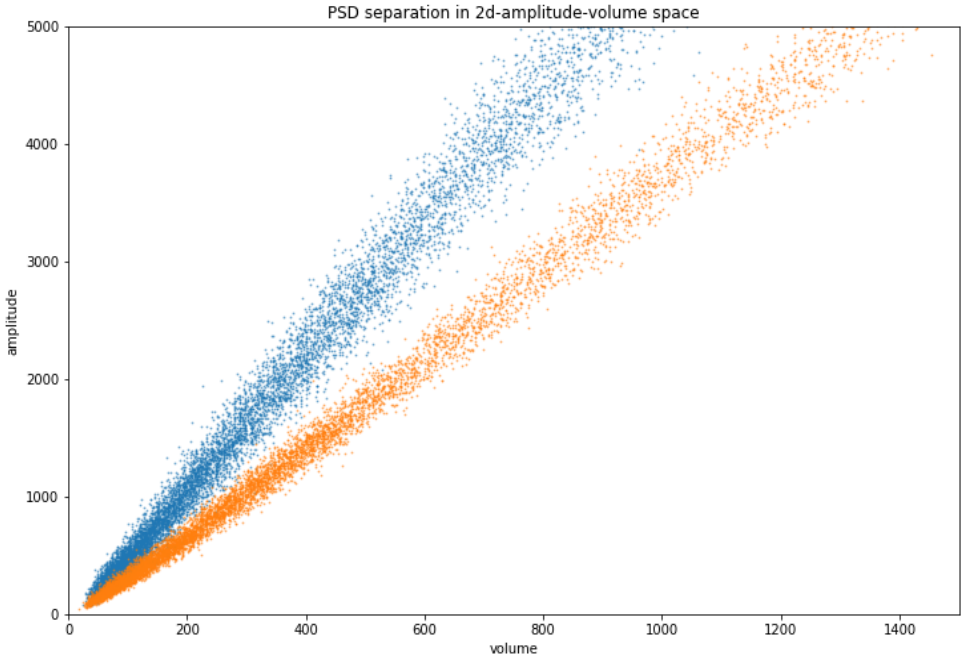

Рисунок 3.7 – Разделение по коэффициенту PSD в двумерном пространстве амплитуды и площади под сигналом

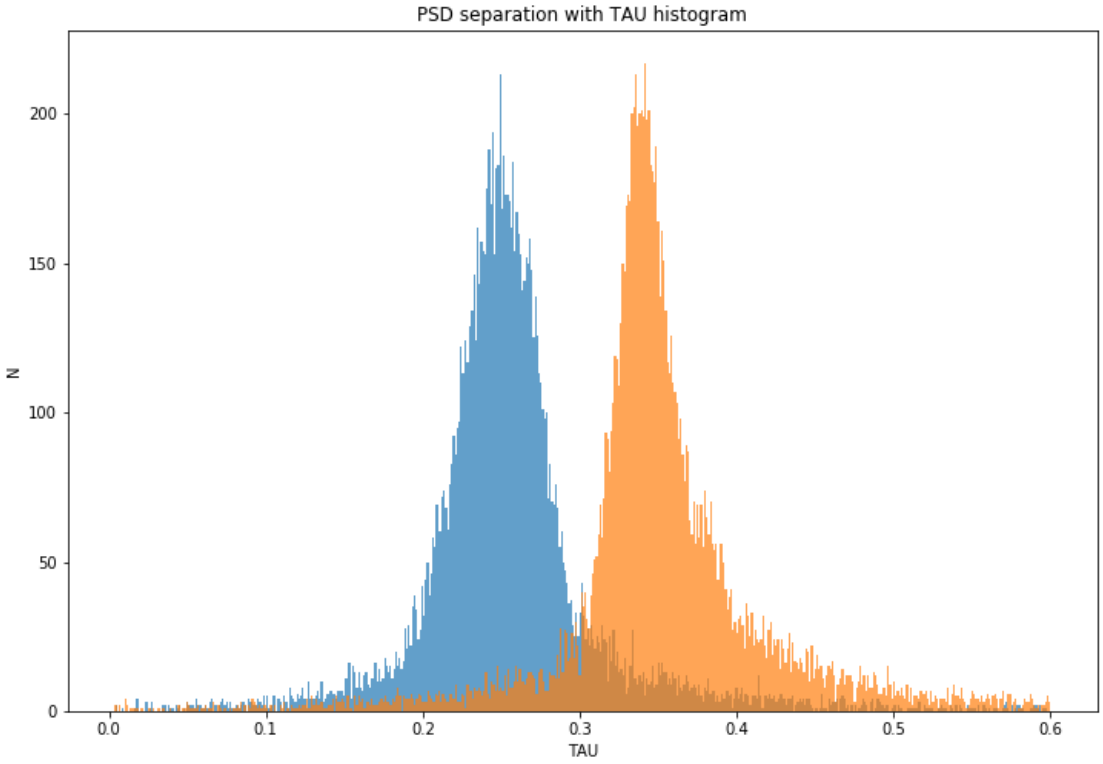

Рисунок 3.8 – Гистограммы для времени высвечивания, полученные при помощи разделения по PSD.

По разделению, проиллюстрированному на рисунках 3.7 и 3.8 были вычислены параметры разделения: Calinski-Harabaz score для такого разделения составил 38.72, коэффициент силуэта равен 0.01.

3.3.3 Разделение по гистограмме времени высвечивания

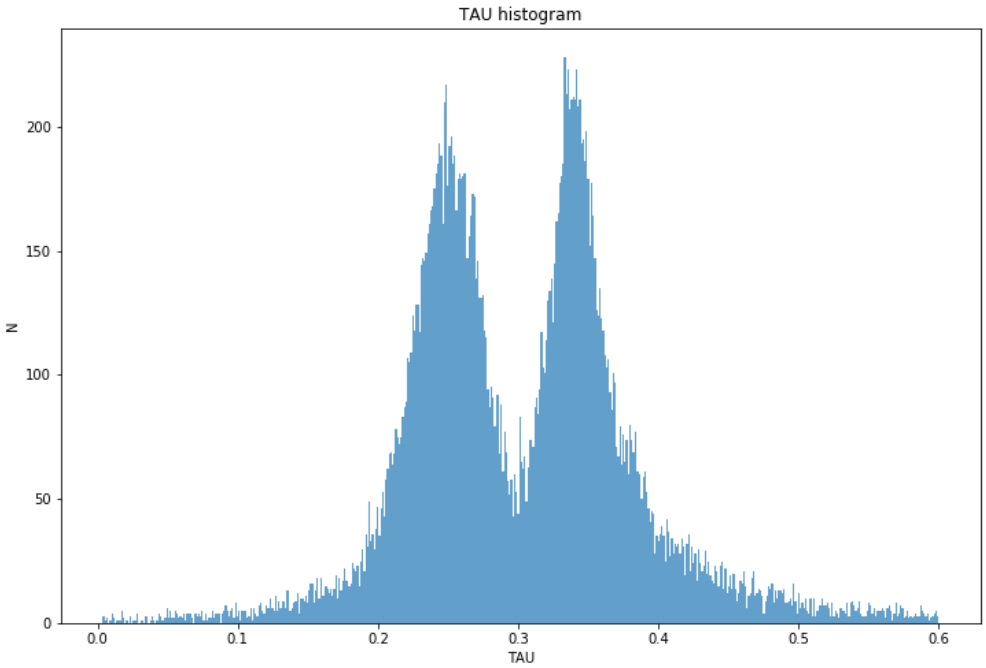

Рисунок 3.9 – Гистограмма времени высвечивания

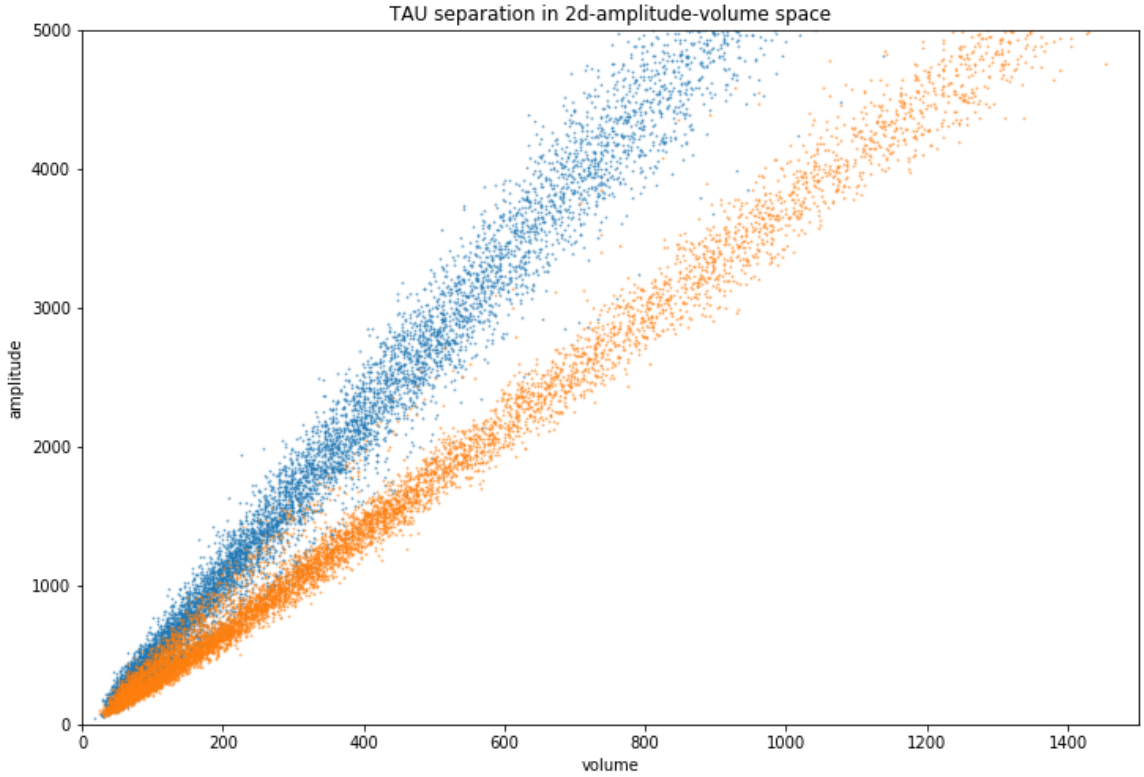

Рисунок 3.10 – Разделение по времени высвечивания в двумерном пространстве амплитуды и площади под сигналом

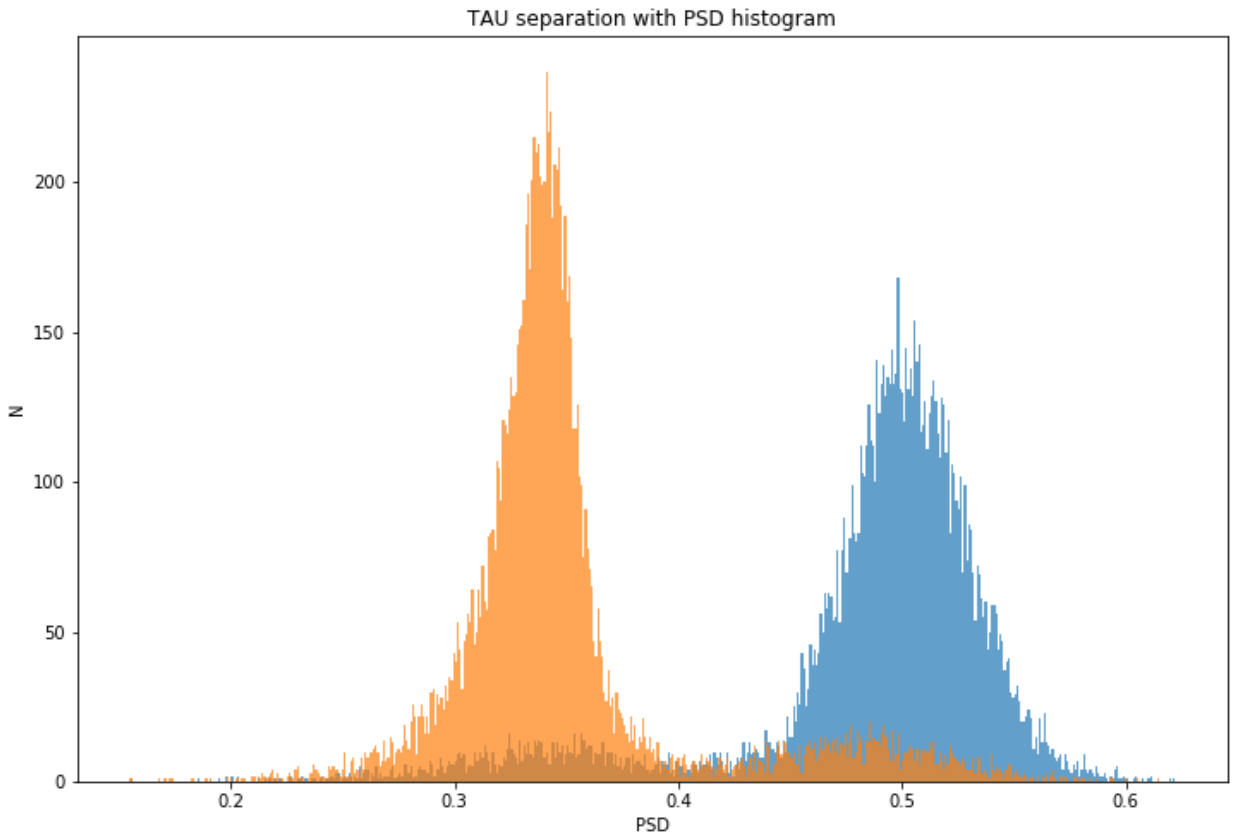

Рисунок 3.11 – Гистограммы для PSD, полученные при помощи разделения по времени высвечивания.

Проведя вертикаль по гистограмме с рисунка 3.9, проиллюстрируем разделение посредствам гистограмм PSD и разделения в двумерном пространстве амплитуды и площади под сигналом.

По разделению, проиллюстрированному на рисунках 3.10 и 3.11 были вычислены параметры разделения: Calinski-Harabaz score для такого разделения составил 24.01, коэффициент силуэта равен 0.008.

3.3.4 Разделение композицией алгоритмов

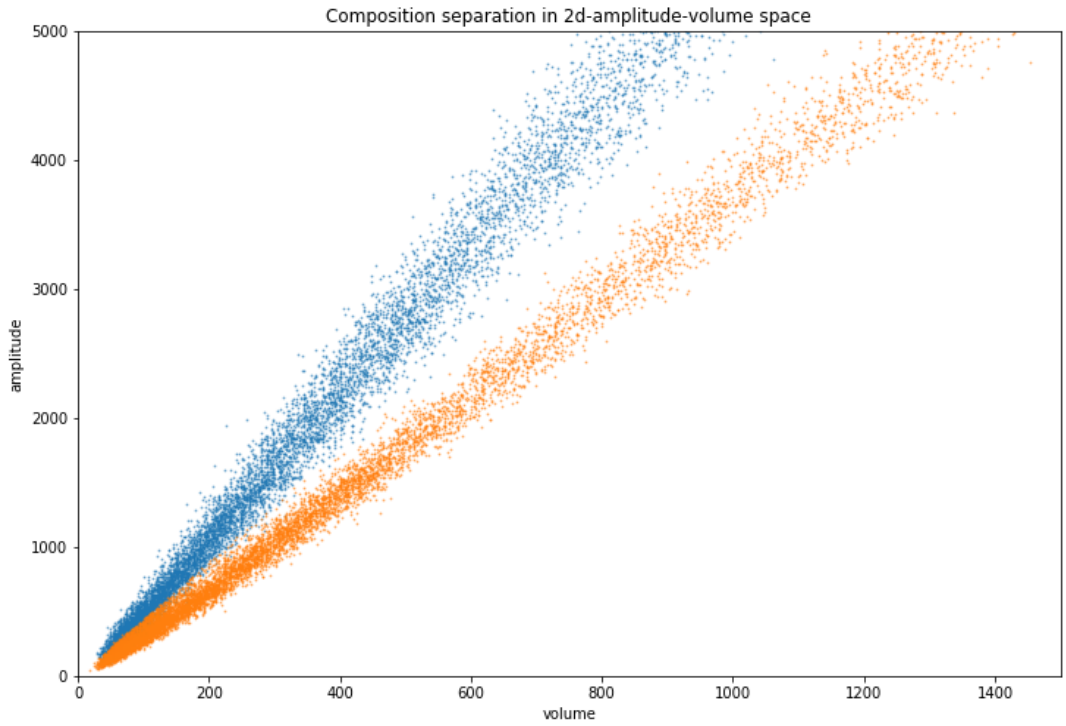

Рисунок 3.12 – Разделение композицией алгоритмов в двумерном пространстве амплитуды и площади под сигналом

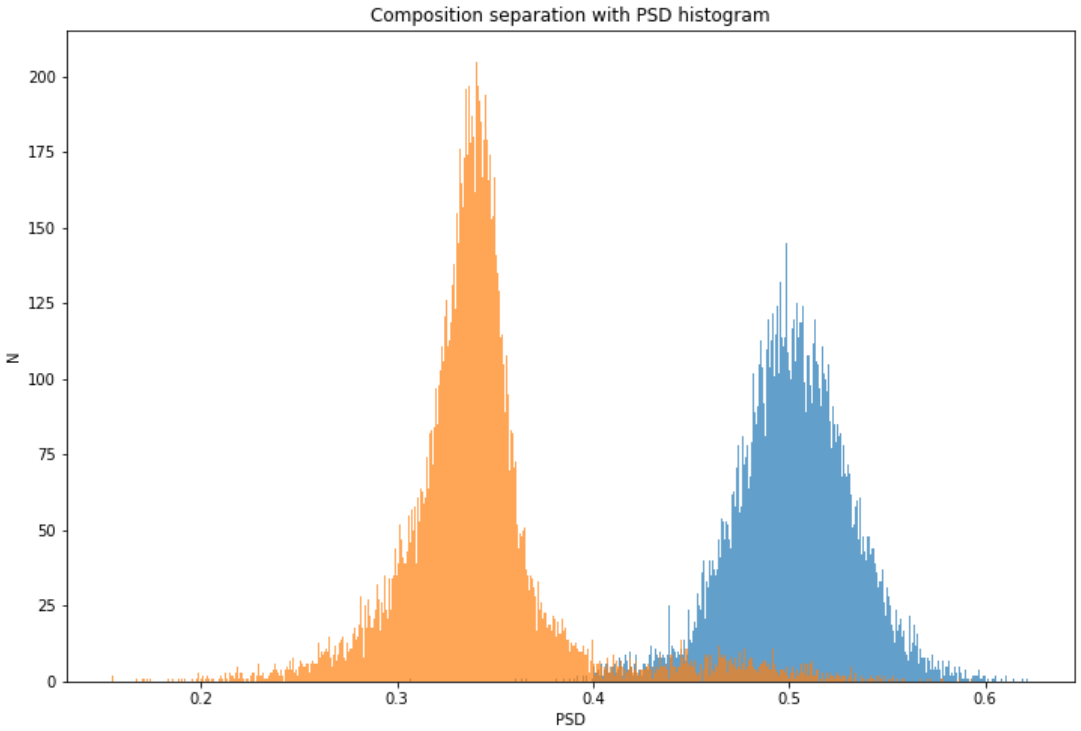

Рисунок 3.13 – Гистограммы для PSD, полученные при помощи разделения композицией алгоритмов.

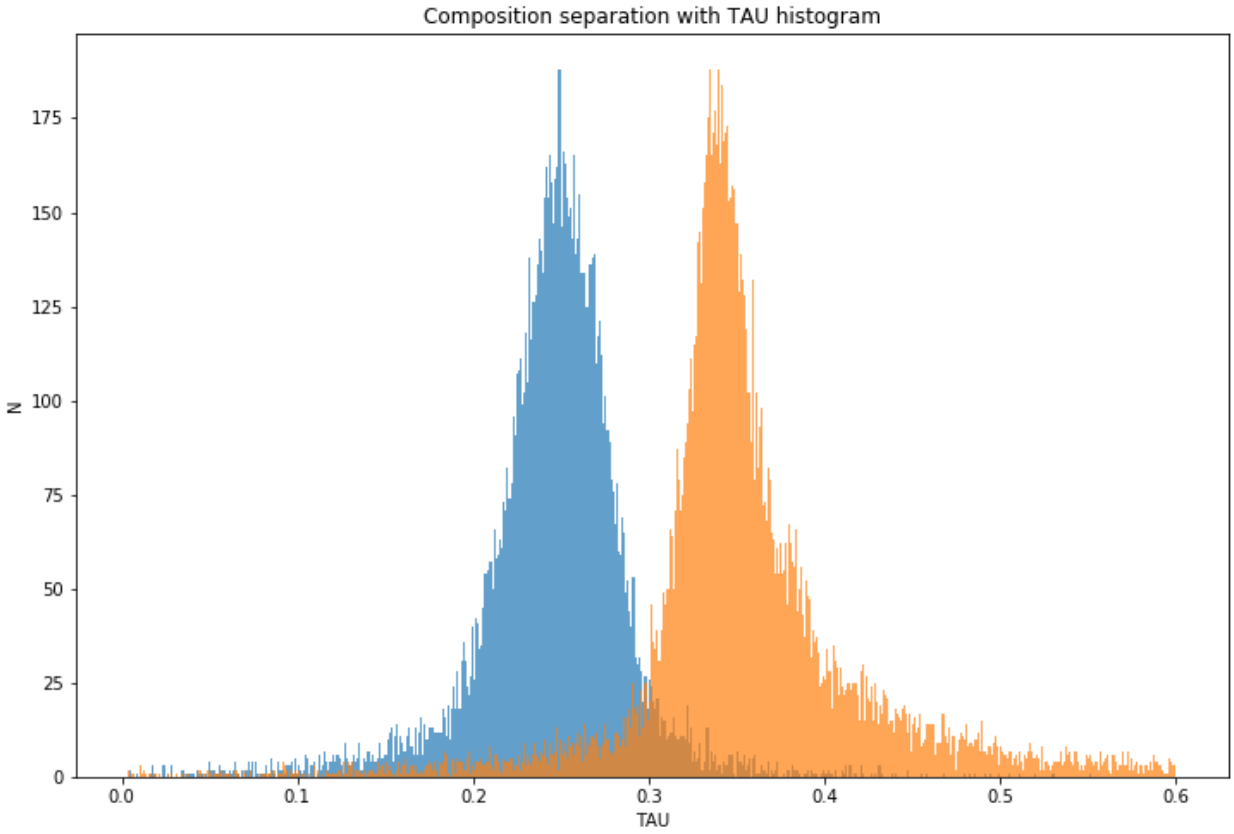

Рисунок 3.14 – Гистограммы для времени высвечивания, полученные при помощи разделения композицией алгоритмов.

По разделению, проиллюстрированному на рисунках 3.12, 3.13 и 3.14 были вычислены параметры разделения: Calinski-Harabaz score для такого разделения составил 30.04, коэффициент силуэта равен 0.014.

3.3.5 Объединение результатов

В итоге получаем таблицу, содержащую в себе информацию относительно комбинаций всех использованных методов и рассмотренных показателей кластеризации:

Таблица 1 – Итоговые значения показателей кластеризации

	Коэффициент силуэта	Calinski-Harabaz score
Метод главных компонент	0.027	93.82
Разделение по гистограмме PSD	0.01	38.72
Разделение по гистограмме времени высвечивания	0.008	24.01
Разделение композицией алгоритмов	0.014	30.04

ЗАКЛЮЧЕНИЕ

Работа посвящена проблеме определения типов сигналов сцинтилляционного детектора. Для решения задачи разделения сигналов на два подмножества было проведено ознакомление с двумя методами разделения, широко использующимися в промышленности: разделением по гистограмме времени высвечивания и разделением по гистограмме PSD. Также был разработан метод разделения, основанный на методе главных компонент, и выдвинуто предложение использовать композицию вышеперечисленных алгоритмов в качестве ещё одного способа определения типа сигналов.

В результате анализа предметной области были получены представления о характерных особенностях таких веществ, как сцинтилляторы, о принципах работы сцинтилляционных детекторов. В итоге это позволило получить понимание принципиальных особенностей работы алгоритмов для разделения сигналов. Также был сформирован подход к задаче разделения сигналов как к задаче кластеризации, что позволило ввести показатели, основываясь на которых стало возможно сравнение алгоритмов между собой.

Проделанная часть, посвященная разработке и обзору подходов к определению типов сигналов, является важной составляющей работы. В ходе выполнения этой части был дан обзор методов, используемых в настоящее время, а также методов, разработанных в процессе выполнения настоящей работы. Были приведены особенности реализации алгоритмов и методы, на которых основаны эти алгоритмы.

Анализ результатов показал, что при сравнении алгоритмов по таким показателям как коэффициент силуэта и Calinski-Harabasz index, наилучший результат продемонстрировал алгоритм разделения, основанный на методе главных компонент. Значения показателей для него составили 0.027 и 93.82

соответственно против 0.01 и 38.72 для метода разделения по гистограмме PSD, 0.008 и 24.01 для метода разделения по гистограмме времени высвечивания. Разделение композицией алгоритмов дало значения показателей 0.014 и 30.04, что в целом не оправдало ожиданий, однако полученные значения оказались выше, чем у метода разделения по гистограмме времени высвечивания. Возможно, что результат разделения, получаемый при использовании композиции улучшится с увеличением числа базовых алгоритмов или с введением процедуры взвешивания для алгоритмов.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Knoll G. F. Radiation Detection and Measurement [Текст] / G. F. Knoll // New York: John Wiley & Sons. – 2010. – С. 830.
2. Valente de Oliveira J. Advances in Fuzzy Clustering and its Applications [Текст] / J. Valente de Oliveira, W. Pedrycz // New York: John Wiley & Sons. – 2007. – С. 433.
3. Duda R.O. Pattern Classification [Текст] / R.O. Duda, P.E. Hart, D.G. Stork // New York: John Wiley & Sons. – 2000. – С. 653.
4. Кирсанов М. А. Сцинтилляционный детектор [Текст] / М.А. Кирсанов, В.В. Кушин, Н.А. Миханчук, С.Г. Покачалов // М: МИФИ. – 2006. – С. 25.
5. Chepurnov A.S. Study of the influence of ADC sampling rate on the efficiency of neutron-gamma discrimination by the pulse shape [Текст] / A. S. Chepurnov, O. I. Gavrilenko, M. A. Kirsanov, S. G. Klimanov, A. S. Kubankin // Journal of Physics: Conference Series. – V. 934 012054. – 2017.
6. Birks J. B. The Theory and Practice of Scintillation Counting [Текст] / J. B. Birks // Somerset: D.R. Hillman & Sons Ltd. – 1967. – С. 633.
7. Обучение на размеченных данных. Машинное обучение и линейные модели. [Электронный ресурс] URL: https://www.coursera.org/learn/supervised-learning/home/week/1 (дата обращения: 25.05.2019)
8. Поиск структуры в данных. Кластеризация. [Электронный ресурс] URL: https://www.coursera.org/learn/unsupervised-learning/home/week/1 (дата обращения 25.05.2019)
9. Halkidi M. On Clustering Validation Techniques [Текст] / M. Halkidi, Y. Batistakis, M. Vazirgiannis // Journal of Intelligent Information Systems. – V. 17. – 2001. –pp. 147-145.

10. Маккинни У. Python и анализ данных [Текст] / У. Маккинни // М: ДМК Пресс. – 2015. – С. 482.
11. Arbelaitz O. An extensive comparative study of cluster validity indices [Текст] / O. Arbelaitz, I. Gurrutxagan, J. Muguerza, J. M. Perez, I. Perona // Pattern Recognition. – V. 46. – 2013. –pp. 243-256.
12. Luo X. L. Neutron/Gamma Discrimination Utilizing Fuzzy C-Means Clustering of the Signal from the Liquid Scintillator [Текст] / X. L. Luo, G. Liu, J. Yang // Pervasive Computing Signal Processing and Applications. – 2010. –pp. 994-997.
13. Gramacki A. Nonparametric Kernel Density Estimation and Its Computational Aspects [Текст] / A. Gramacki // Cham: Springer. – 2018. – С. 197.
14. Cester D. Pulse shape discrimination with fast digitizers [Текст] / D. Cestera, M. Lunardona, G. Nebbiab, L. Stevanatoa, G. Viestia, S. Petruccic, C.Tintoric // Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment. – V. 748. – 2014. –pp. 33-38.
15. Поиск структуры в данных. Понижение размерности и матричные разложения. [Электронный ресурс] URL: https://www.coursera.org/learn/unsupervised-learning/home/week/2 (дата обращения: 27.05.2019)
16. Обучение на размеченных данных. Решающие деревья и композиции алгоритмов. [Электронный ресурс] URL: https://www.coursera.org/learn/supervised-learning/home/week/4 (дата обращения: 27.05.2019)
17. Shalev-Shwartz S. Understanding Machine Learning From Theory to Algorithms [Текст] / S. Shalev-Shwartz, S. Ben-David // Cambridge: Cambridge University Press. – 2014. – С. 449.
18. Доусон М. Программируем на Python [Текст] / М. Доусон // СПб: Питер. – 2014. – С. 416.

19. Мюллер А. Введение в машинное обучение с помощью Python. Руководство для специалистов по работе с данными [Текст] / А. Мюллер, С. Гвидо // СПб: Альфа-Книга. – 2017. – С. 480.
20. Жерон О. Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow. Концепции, инструменты и техники для создания интеллектуальных систем [Текст] / О. Жерон // М: Вильямс. – 2018. – С. 688.