Моделирование частотных сканов

Богачёв А.М.

13 октября 2022 г.

Аннотация

В отчёте приведены результаты исследований...

Содержание

1 Введение **3**

2 Математические модели частотных сканов **5**
2.1 Модель сигнала релаксации ёмкости 5
2.2 Модель идеального частотного скана 6
2.3 Модель, учитывающая нелинейность и неэкспоненциальность 7
2.4 Модель многоэкспоненциального частотного скана 8

3 Реализация моделей **9**
3.1 Интерфейс и функционал 9
3.2 Алгоритм идентификации параметров моделей 11
3.3 Оценка качества модели 15

Список литературы **22**

1 Введение

Релаксационная спектроскопия глубоких уровней (РСГУ) – метод исследования электрически активных дефектов в полупроводниковых материалах. Данный метод обладает высокой чувствительностью к малым концентрациям ловушек носителей зарядов и является спектроскопическим. Существуют различные вариации РСГУ, например токовая и емкостная, так метод емкостной релаксационной спектроскопии глубоких уровней основан на исследовании процесса (сигнала) релаксации ёмкости барьерной структуры. Определив значения постоянной времени сигнала релаксации для разных температур образца, исследователь может определить энергию активации дефекта, вызывающего релаксацию.

В идеальном случае релаксация ёмкости носит экспоненциальный характер, однако так бывает не всегда. Согласно обзору [5] сигналы релаксации ёмкости барьерных структур можно условно разделить на три группы:

1. Моноэкспоненциальный сигнал релаксации, обусловленный одним единственным энергетическим уровнем в запрещённой зоне полупроводника.

2. Сигнал релаксации, состоящий из суммы нескольких моноэкспоненциальных сигналов релаксации.

3. Сигнал релаксации, характеризуемый непрерывным распределением скоростей эмиссии, представленным спектральной функцией $g(\lambda)$.

В последних двух случаях сигнал не является экспоненциальным.

Задача определения постоянной времени одной или нескольких экспоненциальных составляющих процесса релаксации ёмкости или спектральной функции $g(\lambda)$ относится к задачам экспоненциального анализа. Подходы к решению таких задач, а также фундаментальные ограничения и особенности сбора и обработки экспериментальных данных рассмотрены в исчерпывающем обзоре [5].

Во Владимирском Государственном университете создан измерительно-вычислительный комплекс релаксационной спектроскопии глубоких уровней, основным измерительным прибором которого служит спектрометр DLS-82E фирмы Semilab. Измерительно-вычислительный комплекс реализует метод емкостной РСГУ с частотным сканированием при постоянной температуре. В спектрометре аппаратно реализована корреляционная обработка сигнала релаксации с опорной функцией lock-in.

Технические решения, заложенные в названном измерительном оборудовании, требуют особых методов анализа полученных экспериментальных данных, в частности, разработки и идентификации моделей частотных сканов (экспериментальных данных, полученных на измерительно-вычислительном комплексе).

В следующих разделах отчёта будут рассмотрены модели частотных сканов, некоторые технические и методические вопросы их реализации и идентификации, а также их аппробация на экспериментальных данных.

В современной учебной и технической литературе набирают популярность термины «машинное обучение» и «статистическое обучение» (например в книгах [3], [10], [4]). За ними, как правило, скрывается процесс создания моделей, которые после идентификации их параметров на некой тренировачной выборке (экспериментальных данных с известными целевыми значениями), способны с некоторой точностью определять целевые значения для новых, ранее невстречавшихся данных. Таким образом, регрессия и классификация являются одними из самых распространенных задач машинного обучения [3]. В данном отчёте термин «машинное обучение» будет использоваться именно для обозначения процесса разработки и идентификации моделей, решения задачи регрессии.

2 Математические модели частотных сканов

В данном разделе представленны описания математических моделей сигнала релаксации ёмкости и моделей частотного скана.

2.1 Модель сигнала релаксации ёмкости

Согласно обзору [5], зависимость значения ёмкости от времени $f(t)$ для моноэкспоненциального сигнала релаксации имеет вид выражения 1.

$$f(t) = A \exp(-\lambda t), \tag{1}$$

где

A – амплитуда сигнала релаксации ёмкости;

λ – скорость экспоненциального спада, обратнопрпорциональная постоянной веремени сигнала релаксации τ (выражение 2).

$$\lambda = \tau^{-1} \tag{2}$$

Спектр моноэкспоненциального сигнала релаксации имеет вид, представленный на рисунке 1.

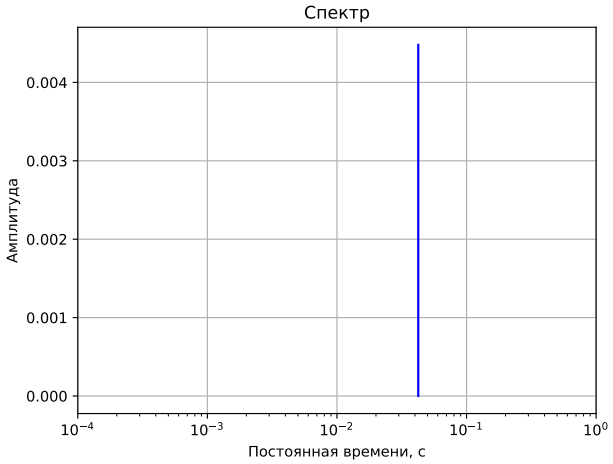

Рис. 1: Пример спектра моноэкспоненциального сигнала релаксации ёмкости. На графике амплитуда указана в пФ.

Согласно источнику [5], зависимость сигнала релаксации ёмкости от времени $f(t)$ для сгинала, образованного несколькими дискретными экспоненциальными сигналами, определяется выражением 3.

$$f(t) = \sum_{i=1}^{n} A_i \exp(-\lambda_i t), \tag{3}$$

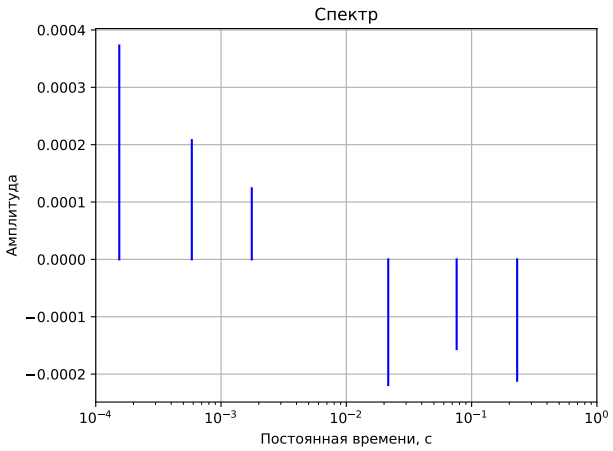

Рис. 2: Пример спектра сигнала релаксации ёмкости, содержащего несколько экспоненциальных составляющих. На графике амплитуда указана в пФ.

где n – количество экспоненциальных составляющих в спектре. Пример спектра такого сигнала показан на рисунке 2.

На спектрах амплитуды могут быть отрицательными. В данном слудче знак при амплитуде указывет лишь на направление изменения ёмкости [6].

2.2 Модель идеального частотного скана

Предполагается, что в идеальном случае сигнал релаксации ёмкости содержит одну экспоненциальную составляющую, а измерительный тракт спектрометра линеен. В таком случае модель частотного скана описывает аппаратные преобразования спектрометра DLS-82E учитывая только форму его опорной функции.

В спектрометре DLS-82E реализована корреляционная обработка сигнала релаксации ёмкости, таким образом сигнал на выходе аналогового тракта спектрометра определяется выражением 4, согласно публикации [5].

$$S\left[g(\lambda), t_c, t_d\right] = \frac{1}{t_c}\int_{t_d}^{t_d+t_c} f(t)W\left(t - t_d\right)dt, \tag{4}$$

где

$W(t)$ – весовая функция, определённая на интервале времени $[0, t_c]$,

t_c – период (длительность) весовой функции $W(t)$,

t_d – время задержки между началом сигнала релаксации и началом корреляционной обработки. Согласно обзору [5], время задержки t_d, обычно, вводится для улучшения избирательности или для снижения искажения сигнала из-за перегрузки измерительной системы.

$g(\lambda)$ – распределение скоростей экспоненциальных спадов, составляющих релаксационный сигнал.

Модель аппаратных преобразований (корреляционной обработки), учитывающая форму весовой функции, реализованной в спектрометре DLS-82E, для моноэкспоненциального сигнала определяется выражением 5 [6].

$$S\left(\tau, C_A, F_0, t_1\right) = C_A K_{BS} K_{LS} \phi\left(\tau, F_0, t_1\right), \tag{5}$$

где

C_A – амплитуда емкостного релаксационного сигнала,

K_{BS} – масштабный коэффициент, зависящий от чувствительности емкостного моста,

K_{LS} – масштабный коэффициент селектора,

τ – постоянная времени релаксации глубокого уровня,

F_0 – частота сканирования импульсов заполнения,

t_1 – длительность импульса заполнения,

$\phi\left(\tau, F_0, t_1\right)$ – функция определяемая выражением 6.

$$\phi\left(\tau, F_0, t_1\right) = M\tau F_0 e^{-\frac{0.05}{\tau F_0}} \left(1 - e^{\frac{t_1 F_0 - 0.45}{\tau F_0}} - e^{-\frac{0.5}{\tau F_0}} + e^{\frac{t_1 F_0 - 0.95}{\tau F_0}}\right), \tag{6}$$

где M – масштабный множитель.

Масштабный множитель M определяется выражением 7.

$$M(\tau, F_0, t_1) = \frac{1}{\max\left[\tau F_0 e^{-\frac{0.05}{\tau F_0}} \left(1 - e^{\frac{t_1 F_0 - 0.45}{\tau F_0}} - e^{-\frac{0.5}{\tau F_0}} + e^{\frac{t_1 F_0 - 0.95}{\tau F_0}}\right)\right]} \tag{7}$$

Введём коэффициент A (выражение 8), характеризующий амплитуду сигнала релаксации ёмкости и перепишем выражение 5 с учётом того, что длительность импульса заполнения t_1 является неизменной величиной, и получим выражение 9.

$$A = C_A K_{BS} K_{LS}. \tag{8}$$

$$S(\tau, A, F_0) = A\phi(\tau, F_0) \tag{9}$$

2.3 Модель, учитывающая нелинейность и неэкспоненциальность

Для одновременного учёта нелинейности аппаратного тракта и неэкспоненциальности сигнала релаксации, связанной с присутствием нескольких

экспоненциальных составляющих в модель вводят коэффициент нелинейности-неэкспоненциальности p [6], после чего выражение 9 приобретает вид выражения 10.

$$S(\tau, A, F_0, p) = A\left[\phi(\tau, F_0)\right]^p. \tag{10}$$

В случае моноэкспоненциального сигнала релаксации и линейного измерительного тракта коэффициент $p = 1$, в остальных случаях он отклоняется от 1. Данный коэффициент эмпирический и не имеет физического смысла. Он комплексно учитывает нелинейность и неэкспоненциальность, что позволяет повысить точность моделирования, но он не позволяет дать ответ на вопрос какой именно из этих двух факторов имел место.

2.4 Модель многоэкспоненциального частотного скана

Если предположить, что сигнал релаксации ёмкости состоит из нескольких экспоненциальных составляющих и определяется выражением 3, то опираясь на выражения 3, 4, 6 и 9, можно сделать выод, что частотный скан, созданный таким сигналом релаксации ёмкости определяется выражением 11.

$$Y = \sum_{i=1}^{n} A_i \phi(\tau_i, F_0), \tag{11}$$

где n – количество экспоненциальных составляющих в сигнале релаксации. Данная модель предполагает, что измерительный такт линеен, а сигнал релаксации создан ловушками носителей заряда, дающими линейчатый спектр, аналогичный спектру, представленному на рисунке 2.

3 Реализация моделей

В данном разделе приводится краткое описание программной реализации моделей, примеры рассчитанных частотных сканов, примеры результатов идентификации их моделей, рассматриваются некоторые методические вопросы.

3.1 Интерфейс и функционал

Модели оформлены в виде пакета модулей на языке программирования Python 3. Интерфейс моделей совместим с одной из самых популярных библиотек для машинного обучения scikit-learn [7]. Такое техническое решение позволяет использовать имеющиеся в библиотеке инструменты предобработки данных, оценки качества и оптимизации параметров моделей, а также интегрировать модели в другое программное обеспечение и конвейеры обработки данных.

Разработанные модели (выражения 9, 10, 11), выполняют две функции:

1. Вычисление частотного скана с заданными параметрами.
2. Идентификация параметров модели частотного скана по экспериментальным данным.

Идентификация параметров моделей реализована методом градиентного спуска. Имеется возможность вывода значений параметров модели на каждой итерации. Реализована автоматическая остановка идентификации при достижении заданного модуля разницы между значениями функции потерь на текущей и предыдущей итерации.

Модели реализуют единообразный интерфейс. При инициализации каждая модель получает длительность импульса заполнения (параметр filling_pulse), которая считается неизменной при измерениях) и параметры алгоритма идентификации. Параметры алгоритма идентификации варьируются:

- модель идеального частотного скана и модель частотного скана с показателем нелинейности-неэкспоненциальности (выражения 9, 10):

  fit_p_coef – если параметр принимает значение True (логическую единицу), тогда при идентификации параметров модели определяется показатель нелинейности-неэкспоненциальности p, иначе (False – логический ноль) p считается равным 1 и не изменяется при идентификации параметров модели;

  learning_rate – скорость градиентного спуска;

  n_iters – максимальное количесво итераций при идентификации;

stop_val – модуль разницы между значениями функции потерь на текущей и предыдущей итерации, при достижении которого происходит остановка идентификации;

verbose – если параметр принимает значение True, то в стандартный поток вывода печатаются значения параметров модели на каждой итерации.

- модель многоэкспоненциального частотного скана (выражение 11):

  n_exps – количество экспоненецальных составляющих в сигнале релаксации (значение n в выражении 11);

  learning_rate – скорость градиентного спуска;

  n_iters – максимальное количесво итераций при идентификации;

  stop_val – модуль разницы между значениями функции потерь на текущей и предыдущей итерации, при достижении которого происходит остановка идентификации;

  verbose – если параметр принимает значение True, то в стандартный поток вывода печатаются значения параметров модели на каждой итерации.

При идентификации параметров модели идеального частотного скана определяется амплитуда и десятичный логарифм постоянной времени ($\log_{10}(\tau)$) сигнала релаксации. Причины замены постоянной времени её десятичном логарифмом объясняются в следующем разделе.

При идентификации модели с показателем нелинейности-неэкспоненциальности к амплитуде и десятичному логарифму постоянной времени сигнала релаксации добавляется значение показателя нелинейности-неэкспоненциальности p.

При идентификации многоэкспоненциальной модели определяется n пар амплитуда — десятичный логарифм постоянной времени сигнала релаксации. По паре для каждой из n экспоненциальных составляющих.

Параметры модели частотного скана могут быть не только идентифицированны, но и заданы пользователем.

Чтобы выполнить идентификацию параметров модели, нужно передать ей набор экспериментальных (тренировачных) данных, состоящих из вектора значений десятичных логарифимов частоты опорной функции ($\log_{10}(F_0)$) и соответствующего вектора значений сигнала DLTS – сигнала на выходе аппаратного тракта спектрометра DLS-82E. При идентификации параметров модели по умолчанию их начальные параметры выбираются случайным образом, однако могут быть и определены пользователем.

При идентификации модели значения её параметров на каждой итерации сохраняются в атрибуте fit_results_, что позволяет пользователю получить и проанализировать данные о работе алгоритма идентификации.

При вычислении частотного скана с заданными параметрами модели передаётся вектор значений десятичных логарифимов частоты опорной функции, для каждого из которых модель вычислит значение сигнала DLTS.

3.2 Алгоритм идентификации параметров моделей

Идентификация параметров модели производится методом градиетного спуска, при этом минимизируется среднеквадратическая ошибка между значениями, полученными в результате измерений, и результатами моделирования (выражение 12).

$$E = \frac{1}{m}\sum_{i=1}^{m}\left(y_i - y_i^*\right)^2, \tag{12}$$

где

y_i – значения, полученные в результате измерений,

y_i^* – значения, полученные в результате моделирования,

m – количество измерений.

При каждом обновлении постоянной времени сигнала релаксации (как при идентификации модели, на каждой итерации, так и при обновлении постоянной времени пользователем) вычисляется масштабный множитель $M(\tau, F_0, t_1)$, определяемый выражением 7. Таким образом модель всякий раз вычисляет значение $\max\left[\tau F_0 e^{-\frac{0.05}{\tau F_0}}\left(1 - e^{\frac{t_1 F_0 - 0.45}{\tau F_0}} - e^{-\frac{0.5}{\tau F_0}} + e^{\frac{t_1 F_0 - 0.95}{\tau F_0}}\right)\right]$. В текущей реализации моделей данный максимум вычисляется приблизительно с помощью градиентного спуска. Не смотря на то, что применяемый итеративный алгоритм находит максимум очень быстро, эти вычисления занимают довольно много веремени при идентификации моделей, потому что выполняются на каждой итерации (при каждом обновлений постоянной времени τ). В случае многоэкспоненциальной модели данный процесс повторяется еще и для каждой экспоненциальной составляющей, что заметно снижает скорость идентификации модели при значениях параметра n_exps больше 10. Таким образом в будущем вычисления можно будет оптимизировать за счёт одного из следующих решений:

1. найти точное аналитическое выражение для расчёта $M(\tau, F_0, t_1)$,
2. найти для $M(\tau, F_0, t_1)$ аппроксимирующую функцию, позволяющую вычислять мастабный множитель с приемлимой точностью.

В моделях градиентный спуск реализован при помощи библиотеки TensorFlow [8], главное преимущество которой в том, что она реализует алгоритм автоматического дифференцирования на графе вычислений. Таким образом,

при расчёте градиентов сначала производные берутся символьно (точно), а затем вычисляются их значения, поэтому точность вычисления градиента ограничена только разрядностью чисел [3], [10], [8].

Во всех моделях используется алгоритм идентификации с постоянной и одинаковой для всех параметров скоростью градиентного спуска, поэтому для ускорения идентификации параметров моделей и улучшения сходимости алгоритма необходима нормализация (прведение к единому масштабу) тренировочных данных.

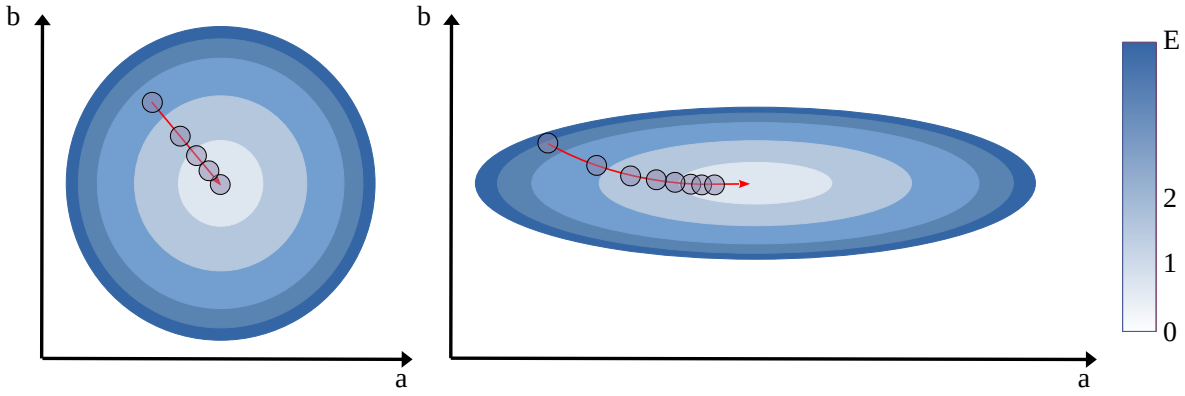

Рис. 3: Градиентный спуск с нормализацией тренировачных данных (график слева) и без неё (график справа) для модели с двумя параметрами. a и b – параметры модели, E – функция потерь.

Рассмотрим примеры на рисунке 3. В идеальном случае функция потерь (среднеквадратическая ошика – выражение 12) должна иметь форму «симметричной чаши», а изменение обоих параметров должно вносить одинаковый вклад в значение градиента. В таком случае алгоритм градиентного спуска достигнет минимума функции потерь кратчайшим путём, как на левом графике рисунка 3. Если же тренировочные данные имеют очень разные масштабы, функция потерь (гафик справа на рисунке 3) будет иметь форму «вытянутой чаши», а алгоритм идентификации будет долго подстраивать одни из параметров без существенного изменения функции потерь [3]. Чтобы минимизировать этот эффект, тренировочные данные нормализуют перед идентификацей параметров модели. После идентификации параметры модели и тренировочные данные можно вернуть в исходный масштаб.

Для разрабатанных моделей тренировочные данные данные имеют существенно отличающиеся масштабы: значение сигнала DLTS находятся в диапазоне долей пикофарад, а диапазон изменения частоты опорной функции спектрометра покрывает три декады (от 1 Гц до 2500 кГц). Таким образом, желательно выполнять нормализацию исходных данных перед идентификацией параметров модели. Кроме этого, изменение параметров модели (постоянной времени и амплитуды) вносят существенно различающийся

вклад в градиент функции потерь. Если снова обратиться к выражениям 6 и 9, можно заметить, что значение функции потерь пропорционально первой степени амплитуды A сигнала релаксации и экспоненте, возведенной в степень скорости экспоненциального спада, то есть $e^{\frac{1}{\tau}}$. Поэтому на вход моделей при идентификации подаётся десятичный логарифм частоты опорной функции ($\log_{10}(F_0)$), а идентификация производится не по постоянной времени τ, а по десятичному логарифму постоянной времени $\log_{10}(\tau)$. На рисунке 4 показан приемер идентификации идельного частотного скана. «Путь» параметров модели в процессе идентификации показывает красная линия с маркерами, а изолинии и градиент – значения функции потерь.

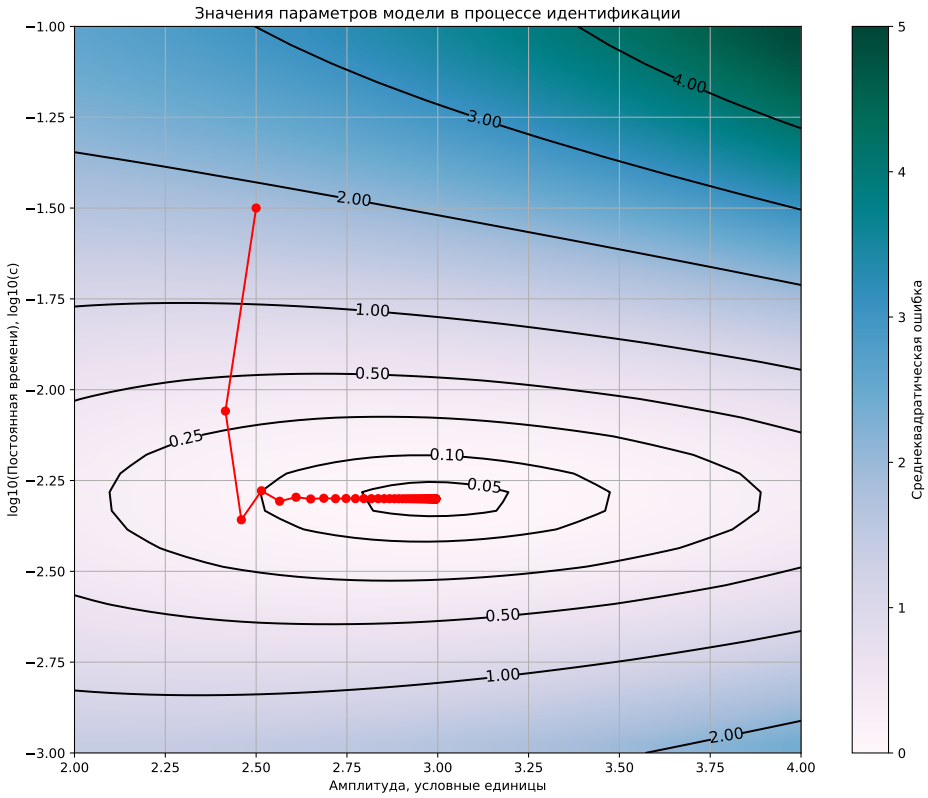

Рис. 4: «Путь» значений параметров при идентификации.

На рисунке 4 форма изолиний отличается от концентрических окружностей, а «путь» параметров не похож на прямую линию, что говорит о том, что возможна дальнейшая оптимизация алгоритма идентификации, однако, названные выше меры позволили значительно смягчить эффект от различных масштабов в экспериментальных данных и в параметрах модели. Также возможно использование одного из алгоритмов адаптивного градиентного спуска, то есть алгоритмов изменяющих скорость по мере приближения к оптимальным значениям параметров. Такое решение не изменит форму функции потерь, но оптимизирует «путь» и количество итераций.

Рисунки 5 и 6 показывают примеры результатов идентификации модели идеального частотного скана и модели многоэкспоненциального частотного

скана соответственно на расчитанных тренировочных данных.

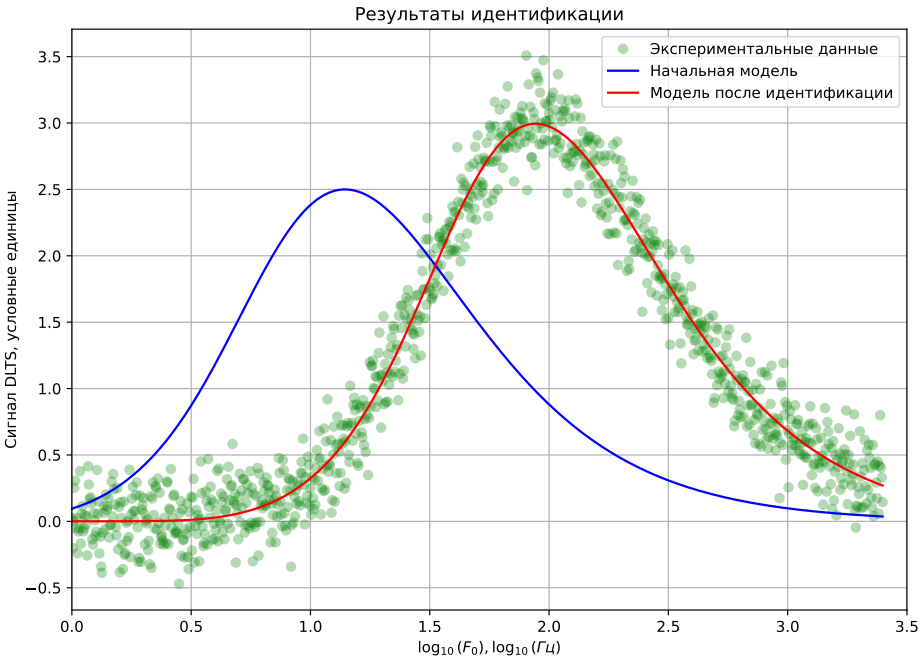

Рис. 5: Пример результата идентификации модели идеального частотного скана.

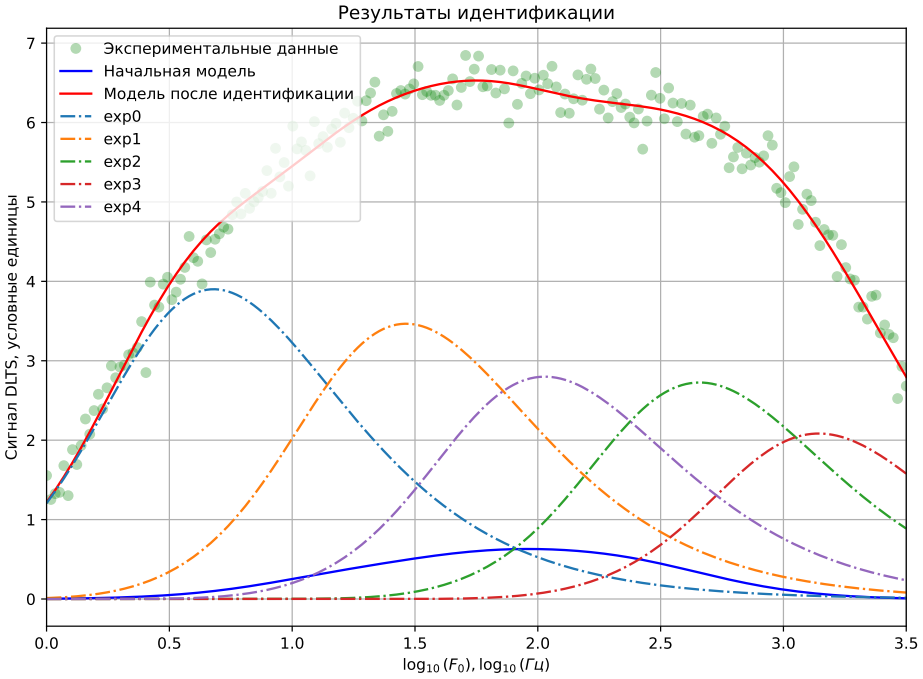

Рис. 6: Пример результата идентификации модели многоэкспоненциального частотного скана. Штрихпунктирными линиями показаны частотные сканы для отдельных экспоненциальных составляющих сигнала релаксации.

3.3 Оценка качества модели

Вычисление корня из среднеквадратической ошибки (англ. root-mean-square error – RMSE) между экспериментальными данными и данными, полученными с помощью моделирвоания, – один из самых известных и распространенных способов оценки качества регрессии данных разработанной моделью. Однако, одного этого показателя часто бывает недостаточно, более того, **оценка этого показателя только на тренировочных данных** при разработке модели (**особенно эмпирической модели**) – методическая ошибка [3], [2]. Данные утверждения обусловлены следующим:

- сам по себе корень из среднеквадратической ошибки не отвечает на вопрос о «фундаментальном» соответствии данных модели, а также не показывает какова вероятность того, что полученное значение – результат случайного совпадения;

- оценивая данную метрику только на тренировочных данных, есть риск получить слишком оптимистическую оценку и столкнуться с проблемой, называемой «оверфитинг» (от англ. overfitting)[3], [2], [10].

Для начала, для иллюстрации приведённых аргументов обратимся к набору данных, известному под названием «Квартет Энскомба». Это набор данных был создан математематиком Фрэнком Энскомбом специально для того, чтобы показать важность визуализации данных в графиках. Особенность этого набора в том, что он состоит из четырёх групп данных, имеющих одинаковые статистические характеристики, но абсолютно разные графики [9], [1]. Графики приведены на рисунке 7, статистические характеристки – в табилице 1.

Таблица 1: Статистические свойства наборов данных в Квартете Энскомба

Характеристика	Значение
Среднее значение переменной x	9.0
Дисперсия переменной x	10.0
Среднее значение переменной y	7.5
Дисперсия переменной y	3.75
Корреляция между переменными x и y	0.816
Уравнение линейной регрессии	$y = 3 + 0.5x$
Коэффициент детерминации линейной регрессии (R^2)	0.67

Что примечательно, все 4 подгруппы данных имеют одно и то же урванение линейной регрессии (см. таблицу 1) и, как показано в таблице 2, очень близкие занчения корня из среднеквадратической ошибки.

В таблице 3 для справки приведены данные из «Квартета Энскомба».

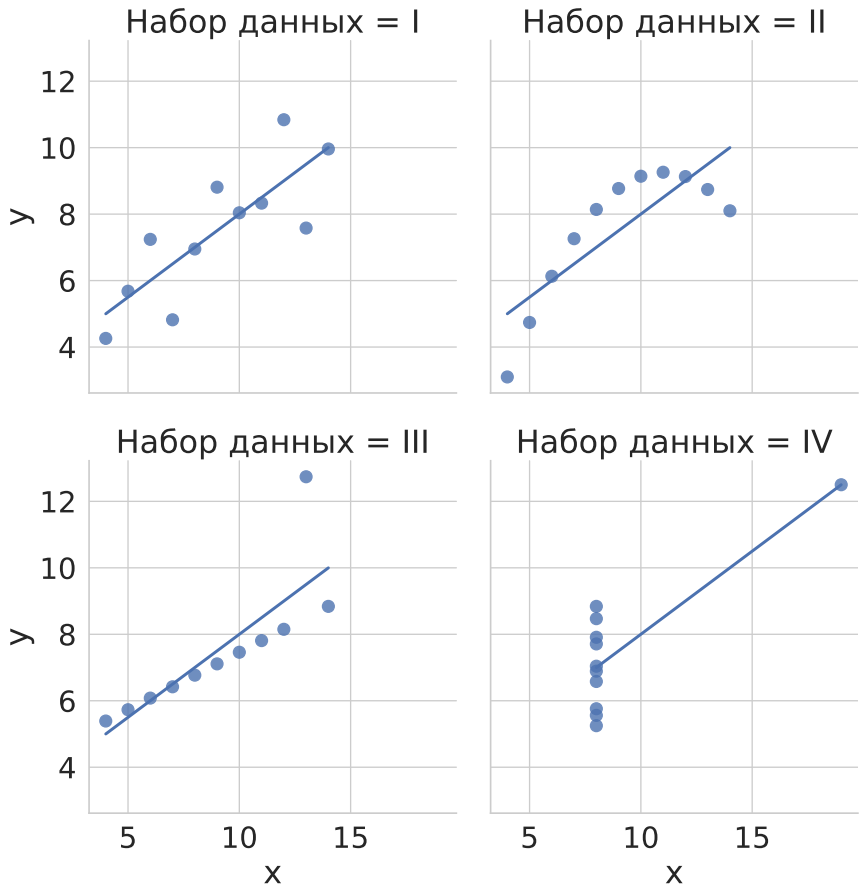

Рис. 7: Квартет Энскомба.

Таблица 2: Значения корней из среднеквадратической ошибки

Набор данных	I	II	III	IV
Значение корня среднеквадратической ошибки (RMSE)	1.119	1.119	1.118	1.118

Таблица 3: Квартет Энскомба

I		II		III		IV	
x	y	x	y	x	y	x	y
10,0	8,04	10,0	9,14	10,0	7,46	8,0	6,58
8,0	6,95	8,0	8,14	8,0	6,77	8,0	5,76
13,0	7,58	13,0	8,74	13,0	12,74	8,0	7,71
9,0	8,81	9,0	8,77	9,0	7,11	8,0	8,84
11,0	8,33	11,0	9,26	11,0	7,81	8,0	8,47
14,0	9,96	14,0	8,10	14,0	8,84	8,0	7,04
6,0	7,24	6,0	6,13	6,0	6,08	8,0	5,25
4,0	4,26	4,0	3,10	4,0	5,39	19,0	12,50
12,0	10,84	12,0	9,13	12,0	8,15	8,0	5,56
7,0	4,82	7,0	7,26	7,0	6,42	8,0	7,91
5,0	5,68	5,0	4,74	5,0	5,73	8,0	6,89

Квартет Энскомба – не единственный набор данных, имеющий такие свойства, но, пожалуй, самый известный [9].

Конечно, достаточно просто построить график и попробовать идентифицировать разные виды моделей на одном и том же наборе данных, чтобы исключить такую ошибку и выбрать наиболее подходящие модели.

Далее, обратимся к проблеме, называемой «оверфитинг». Как уже было сказано, оценка RMSE только на тренировочных данных – методическая ошибка. Если исследователь на этапе выбора модели (например, выбора между линейной и полиномиальной регрессией) или на этапе выбора параметров модели (параметров, не зависящих от данных, например степени полинома) выполняет оценку данной метрики на том же наборе, на котором выполняет идентификацию параметров модели, появляется риск настроить модель таким образом, что она будет демонстрировать превосходные результаты на тренировочных данных (показатель RMSE может буквально равняться нулю), но будет абсолютно бесполезна на новых не входивших в тренировочный набор. Проще говоря, появляется высокий риск выбрать слишком сложную модель, которая примет случайные отклонения в тренировочных данных за закономеронсть, либо же излишне адаптировать модель под тренировочный набор [3], [10].

Для того чтобы проиллюстрировать эту проблему, подготовим небольшой набор данных, имитирующий результаты измерения величин x и $y(x)$, связанных выражением 13.

$$y(x) = 1 + 0.5x + \epsilon, \tag{13}$$

где ϵ – нормально распределённая случайная величниа со средним, значением равным 0, и среднеквадратическим отклонением равным 0.4. В данном случае, зависимость $y(x)$ «фундаментально» линейна. Расчитаем для данной зависмости 10 точек в диапазоне от $x = 0.5$ до $x = 3.5$, затем, выполним регрессию полученного набора данных линейной функцией и алгебраическим полиномом десятой степени, в конце, оценим корень из среднеквадратической ошибки для обеих моделей. Рисунок 8 демонстрирует результаты регрессий.

После идентификации для модели линейной регрессии получились следующие параметры:

a – коэффициент, характеризующий наклон, равен приблизительно 0.43;

b – коэффициент, характеризующий смещение, равен приблизительно 0.92;

RMSE – корень из среднеквадратической ошибки, равный приблизительно 0.48.

Как и ожидалось, полученные значения очень близки к параметрам выражения 13.

Рис. 8: Иллюстрация оверфитинга в случае полиномиальной модели. Использовался полином десятой степени.

Не будем приводить все коэффициенты полиномиальной регрессии, корень из среднеквадратической ошибки для данной модели получился равным $4.24 \cdot 10^{-11}$, на рисунке 8 видно, что график полиномиальной регрессии проходит через все точки исходных данных, при этом крайне маловероятно, что данная модель будет адекватно предсказывать значения $y(x)$, на значениях x, которых не было в тренировочном наборе.

Для сравнения попробуем идентифицировать многоэкспоненциальную модель на том же наборе данных, примем параметр n_exps равным 10. Результаты идентификации такой модели представлены на рисунке 9.

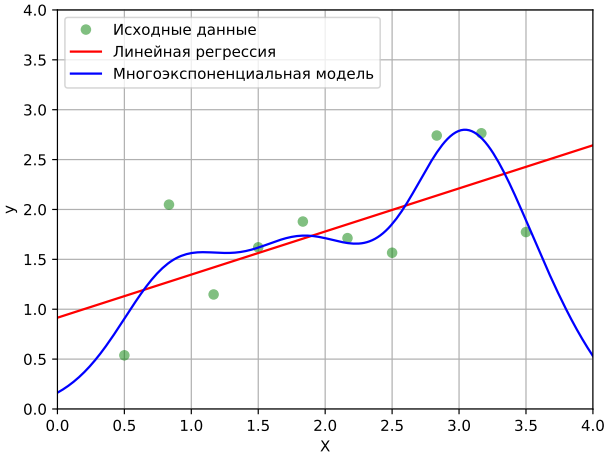

Рис. 9: Иллюстрация оверфитинга в случае многоэкспоненциальной модели. Использовалась модель с параметром n_exps = 10.

Для полученной многоэкспоненциальной модели с параметром n_exps, равным 10 корень из среднеквадратической ошибки получается приблизительно равным 0.28, что значительно больше, чем в случае полиномиальной регрессии, и всё еще меньше, чем в случае линейной регрессии. При этом, хочется подчеркнуть, что данный набор точек является линейной функ-

цией с добавлением нормально распределённой случайной составляющей (выражение 13).

Существует несколько способов борьбы с оверфитингом, ниже приведены некоторые из них:

- собрать больше данных (для того чтобы предыдущие примеры были показательны был намеренно сделан набор из 10 точек, получить аналогичный эффект на наборе из 1000 точек сложнее);
- использовать модель попроще;
- использовать регуляризацию;
- оценивать качество модели не на тренировочном наборе, а на отдельном валидационном наборе (англ. validation set), случайно отобранной контрольной группе;
- использовать приём, называемый кросвалидацией (англ. cross-validation) [3], [2].

В задачах машинного обучения для того, чтобы оценить как точно модель будет предсказывать целевые значения на новых данных, исходный набор разделяют на две группы: тренировочную и тестовою. На тренировачном наборе данных проводится идентификация модели, а на тестовом только оценивают качество самой финальной модели. Таким образом, применительно к задаче регрессии, ожидается, что на новых, невходивших в тренировочный набор данных модель будет демонстрировать среднеквадратическую ошибку, близкую к полученной на тестовом наборе.

Если у модели есть параметр, не зависящий от данных, например, степень алгебраического полинома или количество экспоненциальных составляющих многоэкспоненциальной модели (параметр n_exps), и необходимо найти оптимальное значение данного параметра, то из тренировочного набора выделяют еще один набор данных – валидационный набор, на котором оценивают оценивают модель при каждом значении подстраемового параметра, в итоге выбор останавливают на параметре с которым модель показала лучший результат (наименьшее значение среднеквадратической ошибки). При этом модель не тренируют на валидационном наборе, то есть тренировочный набор уменьшается.

Подход с применением валидационного набора позволяет исключить оверфитинг и прогнозировать эффективность модели. Однако, данный приём приводит к сокращению тренировочных данных, что может вызывать проблемы в случаях работы с наборами данных, содержащими небольшое количество наблюдений. В таких случаях прибегают к технике, называемой кросвалидацией [3], [2].

При использовании кросвалидации тестовый набор всё также применяется только для прогнозирования точности будующих предсказаний модели, на нем никогда не производится обучение модели, и не производится оценка модели во время настройки её параметров. Тестовый набор же разбивается на k подгрупп (в случае использования алгоритма k-fold cross-validation), затем модель в цикле из k шагов обучают на $k-1$ подгруппах, а на оставшейся оценивают модель (в случае регрессии считают среднеквадратическую ошибку), при этом на каждом шаге цикла для оценки модели берут новую подгруппу. В конце считают среднее значение метрики качества модели, полученной на каждом шаге цикла, это и будет результат кросвалидации. Таким образом, модель каждый раз оценивается на данных, которые она «не видела» во время обучения, за счёт этого получается исключить использование валидационного набора, однако приходится тренировать модель k-раз [2].

Рисунок 10 демонстрирует схему работы с данных, при применении кросвалидации. Зелёным отмечены наборы данных, на которых проводится обучение модели, синим – наборы, на которых выполняется оценка среднеквадратической ошибки [2].

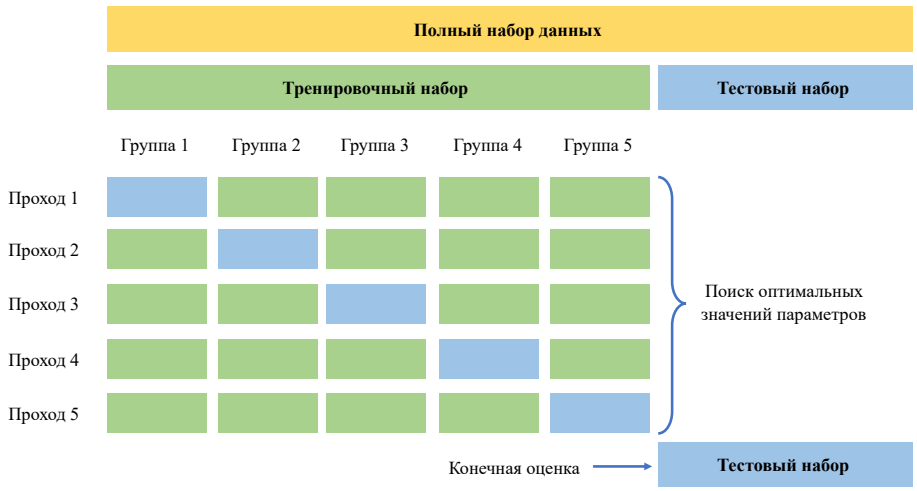

Рис. 10: Иллюстрация кросвалидации.

При оптимизации патаметров модели, не зависящих от данных, по сути, параметров алгоритма идентификации, например количество экспоненциальных составляющих многоэксопоненциальной модели, кросвалидацию выполняют для каждого комплекта значений параметров и выбирают комплект, с которым модель показала лучший результат (такая техника оптимизации в англоязычной литературе называется «grid search», то есть поиск по сетке). В конце, модель с победившим набором параметров еще раз идентифицируют на полном тренировочном наборе данных, после этого, если разработка модели закончена можно выполнить оценку на тестовом наборе [2].

Таким образом, при оценке модели не стоит ограничиваться только

расчётом среднеквадратической ошибки, целесообразно строить графики, применять дополнительные приёмы и метрики, которые на самом деле не ограничиваются названными в данном разделе.

В следующем разделе мы апробуем разработанные модели на экспериментальных данных и сравним результаты для разных моделей.

Список литературы

[1] F. J. Anscombe. «Graphs in Statistical Analysis». В: *The American Statistician* 27.1 (1973), с. 17—21. ISSN: 00031305. DOI: 10.2307/2682899. URL: http://www.jstor.org/stable/2682899 (дата обр. 11.10.2022).

[2] *Cross-validation: evaluating estimator performance*. URL: https://scikit-learn.org/stable/modules/cross_validation.html (дата обр. 12.10.2022).

[3] Aurelien Geron. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. 2nd. O'Reilly Media, Inc., 2019. ISBN: 1492032646.

[4] Trevor Hastie, Robert Tibshirani и Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York, NY, 2017. ISBN: 978-0-387-84858-7. DOI: 10.1007/978-0-387-84858-7.

[5] Andrei A. Istratov и Oleg F. Vyvenko. «Exponential analysis in physical phenomena». В: *Review of Scientific Instruments* 70.2 (1999), с. 1233—1257. DOI: 10.1063/1.1149581. eprint: https://doi.org/10.1063/1.1149581. URL: https://doi.org/10.1063/1.1149581.

[6] Vladimir Krylov, Aleksey Bogachev и T. Pronin. «Deep level relaxation spectroscopy and nondestructive testing of potential defects in the semiconductor electronic component base». В: *Radio industry (Russia)* 29 (май 2019), с. 35—44. DOI: 10.21778/2413-9599-2019-29-2-35-44.

[7] *scikit-learn: Machine Learning in Python*. URL: https://scikit-learn.org/ (дата обр. 10.10.2022).

[8] *TensorFlow*. URL: https://www.tensorflow.org/ (дата обр. 10.10.2022).

[9] *Квартет Энскомба*. URL: https://en.wikipedia.org/wiki/Anscombe%27s_quartet (дата обр. 12.10.2022).

[10] С. Николенко, А. Кадурин и Е. Архангельская. *Глубокое обучение*. СПБ.: Питер, 2018, с. 480. ISBN: 978-5-496-02536-2.