Министерство науки и высшего образования Российской Федерации
Федеральное государственное автомное образовательное учреждение высшего образрвания
«Национальный исследовательский ядерный университет «МИФИ»

УДК 539.12.01

ОТЧЕТ
О НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

ПРИМЕНЕНИЕ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ ДЛЯ ИДЕНТИФИКАЦИИ СТРУЙ, ОБРАЗОВАННЫХ W-БОЗОНОМ

Научный руководитель
к.ф-м.н. _____ Мягков Алексей Григорьевич

Студент _____ Ван Алина Маошэновна

Москва
2025

Содержание

Введение

Стандартная модель – это современная теория в физике элементарных частиц, описывающая сильное, слабое и электромагнитное взаимодействия. Однако, несмотря на все свои преимущества, Стандартная модель не считается полной теорией всего, так как она не включает в себя гравитационное взаимодействие и не описывает некоторые экспериментальные факты, такие как скрытая масса, темная энергия, барионная асимметрия и т.д. Предполагается, что Стандартная модель является частью более общей теории. Поэтому одной из главных задач на LHC является поиск новой физики за рамками Стандартной модели.

Идентификация толстых струй, образованных W-бозонами, распавшимися по адронной моде, является одним из главных составляющих этапов анализа данных с экспериментов по поиску новой физики. К примеру, в эксперименте по поиску возбужденного лептона с конечным состоянием evJ процессом, привносящем основной вклад в фон, является образование пары топ-антитоп. Так как топ-кварк распадается преимущественно по слабому взаимодействию, идентификация струй, образованных W-бозоном, является необходимой для подавления фона.

В предыдущей работе для идентификации W-бозона использовалось ограничение по переменной D2 и инвариантной массе струи. Однако для выделения струй от векторных бозонов существуют и другие дискриминирующие переменные. Для улучшения точности идентификации толстой струи рассмотрено решение этой задачи с помощью методов машинного обучения. В контексте машинного обучения задача идентификации частицы – это задача бинарной классификации, т.е. предсказание по полученным признакам к какому из двух классов принадлежит объект: фону или сигналу.

Ранее проводилось изучение таких моделей машинного обучения, как BDT с градиентным и адаптивным алгоритмами бустинга, полносвязные и сверточные нейронные сети. В рамках изучения моделей сделаны выводы, что несмотря на хорошую интерпретируемость и быстрый темп обучения, модели бустированных деревьев решений имеют свойство быстро переобучаться, в отличие от простых нейронных сетей. Лучшие результаты по AUC-метрике дают полносвязные нейронные сети с алгоритмом оптимизации Adam.

Цель работы.

Целью работы является использование различных техник отбора признаков для обучения нейронной сети для решения задачи идентификации струй, образованных W-бозоном.

В соответствии с поставленной целью задачами данной работы были:

- Ознакомление с различными техниками отбора признаков;
- Обучение и тестирование моделей;
- Резюме результатов.

1 Машинное обучение с учителем

Машинное обучение – это область прикладной математики, изучающая методы решения задач с использованием обучающих данных. В данной работе используются методы машинного обучения с учителем.

Алгоритмы машинного обучения с учителем изучают соответствие между входными и выходными данными. Постановка задачи для обучения с учителем следующая: на вход алгоритма подаются размеченные данные вида (матрица признаков; отклик). Эти данные называются обучающей выборкой, они используются для настройки модели. В зависимости от задачи обучающие данные также могут быть разделены и на валидационную выборку, которая необходима для оценки обобщающих способностей обученных моделей и выбора лучшей модели. Тестовая выборка используется для финальной оценки качества модели. Цель машинного обучения с учителем состоит не в моделировании данных из пространства признаков, а в предсказании откликов на новых данных.

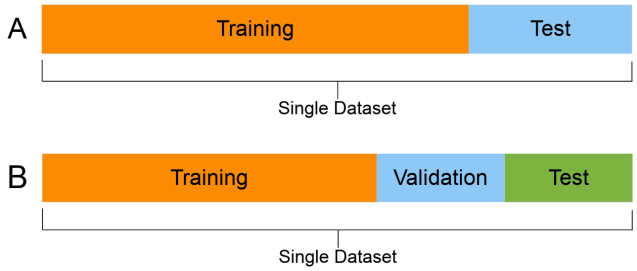

Рис. 1: Разделение датасета

Основными задачами машинного обучения с учителем являются задачи регрессии и задачи классификации. При регрессии алгоритм учится предсказывать непрерывное числовое значение. То есть цель алгоритма для задачи регрессии установить функциональную зависимость между независимыми переменными(признаками) и откликом. При классификации модель учится предсказывать класс принадлежности события(метку класса), то есть целевая переменная - это категориальное значение. Основной целью моделей для решения задачи классификации является обобщение обучения таким образом, чтобы делать точные прогнозы на тестовых данных.

Оценка качества модели связана с функцией потерь. Функция потерь – это мера неточности модели, она сравнивает цели и выходы модели. Задача алгоритма машинного обучения с учителем - минимизация функции потерь.

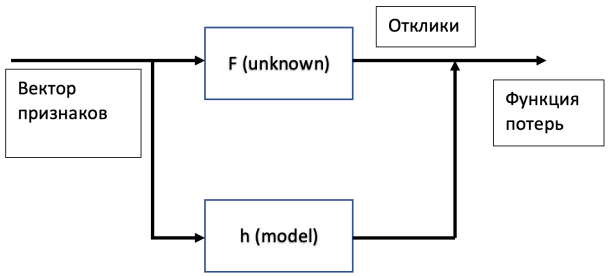

Рис. 2: Схема алгоритма обучения с учителем

1.1 Бинарная классификация

Задача идентификации частицы – это задача бинарной классификации. То есть отклики в данной модели дискретны, они могут принимать только два значения, в нашем случае – сигнал или фон. Задача бинарной классификации [1] состоит в нахождении гиперплоскости, размерность которой равна $N-1$, где N - количество признаков. Эта гиперплоскость должна оптимально разделять объекты одного класса от другого, то есть отличать сигнал от фона.

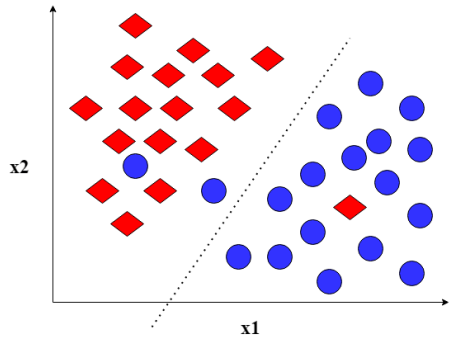

Рис. 3: Бинарная классификация

Функция потерь в данной задаче – это некоторая мера неточности классификатора в предсказании метки класса для объекта. Меньшие потери означают большую точность в предсказании метки класса. В задаче бинарной классификации функция потерь определяется как функция от разности между целями и выходами модели. То есть если объект классифицирован верно – вклад его в функцию потерь будет минимален, если классификатор ошибся, то вклад в потери будет большим. Однако среднего значения функции потерь по выборке (эмпирического риска) не достаточно для оценки ошибки модели из-за дискретного характера откликов. Поэтому необходимы дополнительные метрики для оценивания качества классификатора.

Одними из таких характеристик качества классификации являются ROC-кривая и AUC. ROC-кривая – это график зависимости доли истинных положительно определенных результатов от доли ложных положительно определенных результатов для различных значений контрольных точек проверки гипотезы. AUC – это площадь под ROC-кривой, которая является статистической характеристикой того, насколько доля истинных положительно определенных результатов превышает ложные положительно определенные результаты. То есть чем лучше классификатор разделяет два класса, тем больше AUC и тем выше ROC-кривая.

Для решения задачи бинарной классификации существует множество алгоритмов: от простых моделей, как логистическая регрессия, до глубоких нейронных сетей. В работе использованы алгоритмы MLP и CNN, речь о которых пойдет далее.

1.2 Многослойный персептрон (MLP)

Нейронная сеть – это сложная дифференцируемая функция, задающая отображение из признакового пространства в пространство откликов, все параметры которой могут настраиваться одновременно и взаимосвязано. Сложную функцию обычно представляют в виде композиции простых функций, которые называют слоями. Общая длина цепочки слоев определяет глубину модели.

Нейронную сеть, в которой есть только линейные слои и различные функции активации, называют многослойным персептроном (MLP). Многослойный персептрон состоит из входного, скрытого и выходного уровней взаимосвязанных нейронов.

Архитектура MLP следующая:

- Структура нейронов в входном слое определяется количеством признаков в данных. На входном уровне каждому нейрону сопоставляется один из признаков обучающих данных, далее эти значения распределяются по нейронам скрытых слоев.
- На скрытых слоях происходят сами вычисления. Каждый нейрон в скрытом слое анализирует информацию, полученную из предыдущего слоя. Происходят преобразования данных из пространства одной размерности в пространство другой размерности.
- В выходном слое нейроны генерируют предсказания модели. Структура выходного слоя зависит от поставленной задачи: для задачи бинарной классификации нейронов в выходном слое 2.

Структура слоев в MLP:

- Линейный слой — линейное преобразование над входящими данными (его обучаемые параметры — это матрица W и вектор b). Такой слой преобразует d-мерные векторы в k-мерные.
- Функция активации[2] — нелинейное преобразование, поэлементно применяющееся к пришедшим на вход данным. Благодаря функциям активации нейронные сети способны порождать более информативные признаковые описания, преобразуя данные нелинейным образом.

1.2.1 Функции активации

Главные требования для функции активации: быть монотонной и иметь первую производную почти всюду (необходимо для обратного распространения ошибки). В качестве функции активации могут использоваться разные функции, у каждой из которых есть свои плюсы и минусы.

Сигмоида

Функция сигмоиды преобразовывает поступающие в неё значения в вещественный диапазон $[0, 1]$. То есть, если входные данные окажутся большими положительными значениями, то после преобразования они будут равны примерно единице, а отрицательные числа станут близки к нулю. Это довольно популярная функция, которую можно интерпретировать как частоту возбуждения нейрона. Однако сигмоида имеет несколько недостатков. Во-первых, область значений данной функции смещена относительно 0. Во-вторых, на хвостах сигмоиды происходит затухание градиента, что неприятно для обратного распространения ошибки. В-третьих, просчет экспоненты вычислительно сложен.

Гиперболический тангенс

Гиперболический тангенс используется в случаях, когда необходимо ограничить выходные данные в диапазоне от -1 до 1. Он часто применяется в скрытых слоях нейронных сетей. В отличие от сигмоиды он не смещен относительно 0. Однако значения градиента при обратном распространении по-прежнему могут обнуляться. Тем не менее, использование тангенса обычно более предпочтительно.

ReLU

ReLU является наиболее популярной функцией активации и широко используется в современных нейронных сетях. Она позволяет избежать проблемы затухания градиента. Она вычисляет функцию $f(x) = max(0, x)$, то есть просто выдаёт значения «ноль» и «не ноль».

Кроме того, ReLU очень просто вычисляется: примерно в шесть раз быстрее сигмоиды и тангенса. Однако, в ней отсутствует нулевое центрирование.

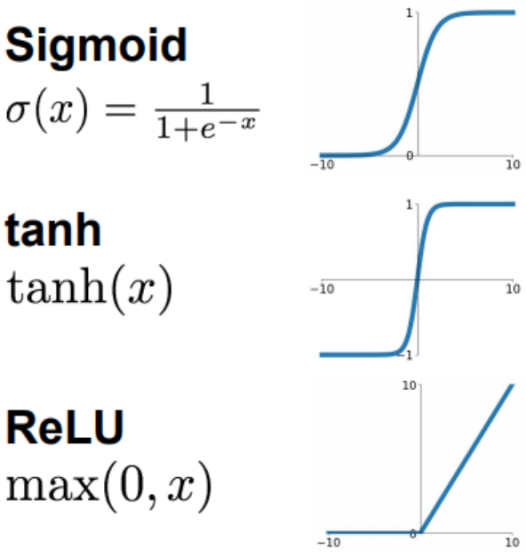

Рис. 4: Функции активации

1.2.2 Метод обратного распространения ошибки

Нейронная сеть обучается с помощью какой-либо модификации градиентного спуска. А для этого ей необходимо вычислять градиенты от функции потерь по всем обучающим параметрам. Нейронную сеть, как сложную функцию, можно представить в виде вычислительного графа, в которых узлы – это вычислительные операции или простые функции.

Применение нейронной сети к данным (вычисление выхода по заданному входу) называют прямым проходом. На этом этапе происходит преобразование исходного представления данных в выходное и последовательно строятся промежуточные представления данных — результаты применения слоёв к предыдущим представлениям. При обратном проходе информация движется от финального представления (от функции потерь) к исходному через все преобразования. Механизм обратного распространения ошибки, играющий важнейшую роль в обучении нейронных сетей, как раз предполагает обратное движение по вычислительному графу сети.

Метод обратного распространения ошибки заключается в рекурсивном использовании правила дифференцирования сложной функции для вычисления градиента в каждом узле, начиная с конечного узла. Информацию о величине градиента для каждого из узлов мы берем из forward pass.

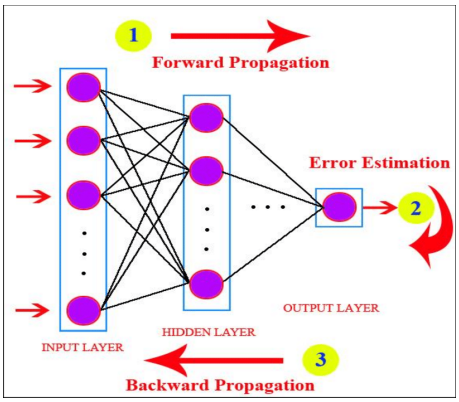

Рис. 5: Метод обратного распространения ошибки

1.2.3 Гиперпараметры модели

Выше была упомянута такая настройка MLP, как выбор функции активации для слоев сети. Однако не только выбор функции активации является гиперпараметром модели. Реализация градиентного спуска на больших наборах данных является затратной операцией как для оперативной памяти, так и по времени, поэтому существуют различные алгоритмы оптимизации весов: стохастический градиентный спуск, Adam и т.д. Выбор алгоритма оптимизации также является настройкой модели, вместе с выбором лосс-функции, которую надо минимизировать. По умолчанию лосс-функция для задачи бинарной классификации - это бинарная кросс-энтропия. Ширина шага, с которой градиент спускается в поисках минимума, называется скоростью обучения. Обычно скорость обучения варьируется от 0.001 до 0.1.

Пропустить через всю сеть целую обучающую выборку будет не только затратным, но и неоптимальным решением. Поэтому данные делят на маленькие части - батчи. Размер батча также является гиперпараметром: чем меньше батч, тем больше количество итераций. Для обновления весов модели необходимо более чем раз провести данные через сеть в прямом и обратном направлении. Эпоха - это один цикл, когда вся обучающая выборка проходит в прямом и обратном направлении через нейронную сеть. Обычно оптимальное количество эпох для модели - это плато между состояниями переобучения и недообучения.

1.3 Переобучение и недообучение

При самом процессе обучения модель может быть недостаточно обучена или переобучена. Недообучение возникает в том случае, когда модель не способна точно предсказывать отклики и моделировать закономерности в данных. Когда модель настроена на избыточную классификацию, тогда можно сказать, что модель переобучена.

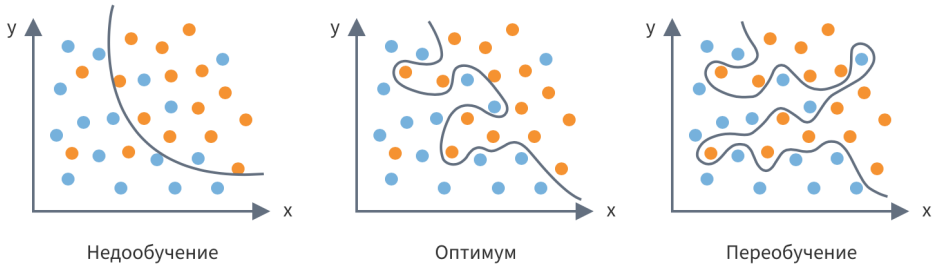

Рис. 6: Примеры недообучения и переобучения

Зафиксировать состояние недообучения или переобучения можно с помощью графиков зависимости ошибок на обучающих данных (далее loss), ошибок на валидационной выборке (val_loss) и номеру эпохи. Для устранения от недообучения применяют следующие подходы: увеличивают количество эпох, меняют архитектуру модели, увеличивают обучающую выборку, усложняют модель. Для устранения переобучения также существуют решения:

- Останавливают процесс обучения на эпохе, при которой val_loss еще не начинает возрастать;
- Уменьшают количество параметров модели, т.е. делают модель менее сложной;
- Применяют dropout: отключают указанную часть нейронов в слое, таким образом прореживая слой;
- Регуляризация: добавляют в функцию потерь дополнительное слагаемое, которое ставит дополнительные ограничения на вектор весов.

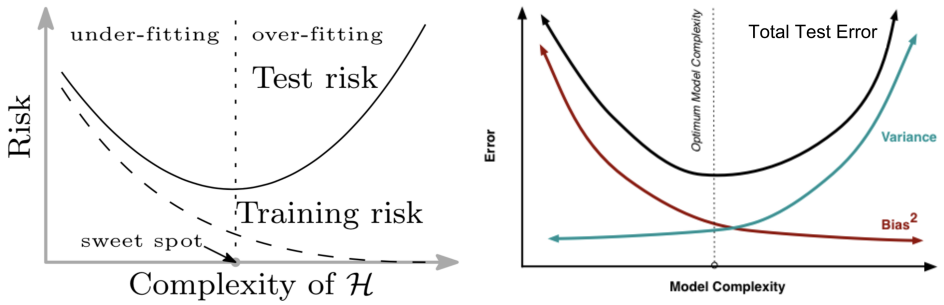

Рис. 7: История обучения

2 Дискриминирующие переменные

Первые два выбранных признака [3], которые будут поданы на вход алгоритмов МО – это инвариантная масса струи и поперечный импульс струи. Ранее для выделения W-струй из большого фона КХД использовалась переменная D2. Определение переменной:

$$D_2 = E_{CF3} \times \Big(\frac{E_{CF1}}{E_{CF2}}\Big)^3, \tag{1}$$

где энергетические корреляционные функции задаются формулами:

$$E_{CF1} = \sum_i^n p_{T,i}; E_{CF2} = \sum_{i,j}^n p_{T,i} p_{T,j} \Delta R_{ij}; E_{CF3} = \sum_{i,j,k}^n p_{T,i} p_{T,j} p_{T,k} \Delta R_{ij} \Delta R_{jk} \Delta R_{ki}. \tag{2}$$

ΔR_{ij} определяется как:

$$\Delta R_{ij}^2 = (\eta_i - \eta_j)^2 + (\phi_i - \phi_j)^2. \tag{3}$$

Также иногда вместо переменной D2 для выделения толстой струи используют переменную C2, которая задана следующей формулой:

$$C_2 = \frac{E_{CF1} \times E_{CF2}}{E_{CF3}^2} \tag{4}$$

В работе использованы данные переменные:

1. Энергетические корреляционные функции $D2, C2, ECF2, ECF3$
2. Масса и поперечный импульс струи p_t, m
3. Planar flow (измеряет степень, в которой энергия струи равномерно распределяется по плоскости поперек поверхности струи по сравнению с линейной распределением по поверхности струи)
4. N-subjettiness (для эффективного подсчета подструй в струе) $\tau 1$
5. Апланарность A
6. Момент Фокса-Вольфрама R_2
7. Angularity (переменная, чувствительная к степени симметрии потока энергии внутри струи)
8. KtDR
9. Разделяющая мера d_{12}

3 Процесс работы и результаты

Процесс обучения модели машинного обучения состоит из следующих этапов: обработка и стандартизация данных, разбиение данных на тестовую, обучающую и валидационную выборки, подбор оптимальных гиперпараметров и обучение, тестирование модели и оценка ее результативности с помощью выбранной метрики.

3.1 Подготовка данных

Работа проводится с данными, полученными методом МК моделирования протон-протонного столкновения в детекторе ATLAS на LHC с энергией в системе центра масс 13 ТэВ.

Сигнальное дерево строится на данных фонового процесса образования пары топ-антитоп, так как топ преимущественно распадается через W-бозон. Ограничения, поставленные на сигнальное дерево, схожи с контрольной областью пары топ-антитоп: отбираются события с как минимум одной b-меченной струей, поперечный импульс которой больше 30 ГэВ и модуль псевдобыстроты меньше 2.5.

В фоновом дереве не должно быть W, распадающегося по адронной моде. Поэтому фоновое дерево строится на данных фонового процесса распада Z бозона в электрон-позитрон. Также поставлены ограничения на отсутствие b-меченных струй в событиях данного дерева.

И на фоновое, и на сигнальные деревья наложены ограничения на поперечный импульс толстой струи и на ее инвариантную массу. Ограничения представлены ниже:

$$60\text{GeV} < m < 110\text{GeV}, p_t > 200\text{GeV}$$

Для некоторых событий вычисление ECF3 не возможно. Переменные D2, C2 задавались по формулам, поэтому для событий с ECF3 равным 0 значения этих переменных были не определены. Для дальнейшей работы с данными и для последующего обучения моделей необходимо было удалить события, содержащие неопределенные значения.

Перед обучением проведена стандартизация датасета.

3.2 Отбор признаков и анализ результатов MLP

3.2.1 Метод фильтрации: Корреляция Пирсона

Метод фильтрации с использованием корреляции Пирсона в машинном обучении применяется для оценки зависимостей между переменными. Этот метод позволяет выявить, какие признаки имеют значительное влияние на целевую переменную. Корреляция Пирсона измеряет степень и направление линейной связи между двумя переменными, принимая значения от -1 до 1. Значение, близкое к 1, указывает на сильную положительную корреляцию; значение, близкое к -1, указывает на сильную отрицательную корреляцию. Использование корреляции Пирсона позволяет отбирать наиболее значимые признаки, что способствует улучшению производительности модели и снижению риска переобучения, как следствие из устранения проблемы мультиколлинеарности.

Для полученных данных ниже представлена матрица корреляции для 13 признаков и целевой переменной.

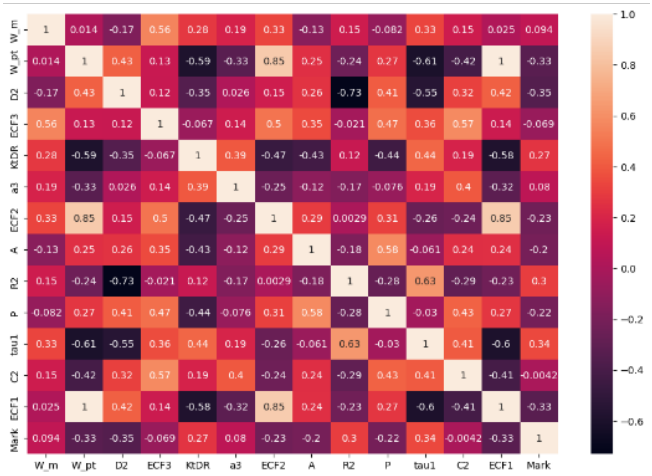

Рис. 8: Корреляционная матрица

С порогом по абсолютной величине 0.1 выявлено 9 признаков, имеющих некоторую зависимость с целевой переменной. Сформированы обучающая и тестовые выборки, включающие только 9 признаков. В качестве модели для обученя была использована полносвязная нейронная сеть с тремя слоями. Гипрерапаметры данной сети следующие:

- Оптимизатор: Adam
- Скорость обучения: 0.001
- Размер батча: 32
- Функция активации: Relu

Для обученной модели с данным набором признаков результаты представлены ниже.

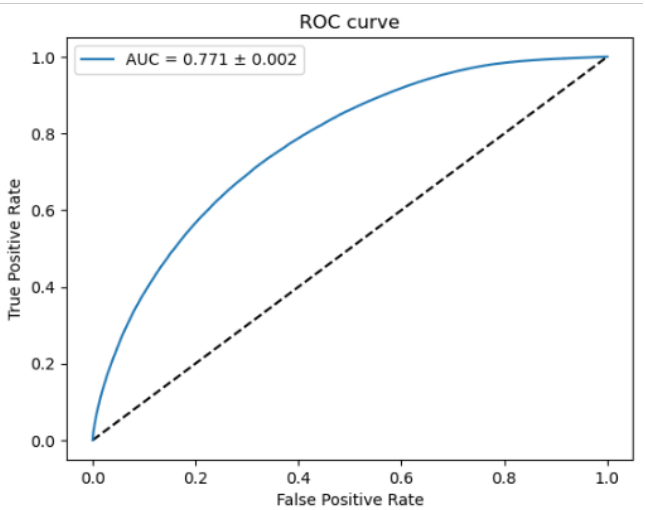

Рис. 9: ROC-кривая

3.2.2 Метод отбора признаков с использованием дерева решений

Основная идея деревьев решений заключается в том, чтобы разбить данные на подмножества, основываясь на значениях признаков. Каждый узел дерева представляет собой условие на определенный признак, а ветви — возможные исходы этого условия. В конечных узлах находятся предсказания модели.

Важность признаков может быть оценена на основе критериев, таких как уменьшение энтропии или индекс Джини, что позволяет количественно оценить вклад каждого признака в модель.

Для оценки важности признаков вся полученная выборка проходила через дерево решений с критерием индекса Джини. Получена важность данных признаков, представленная ниже.

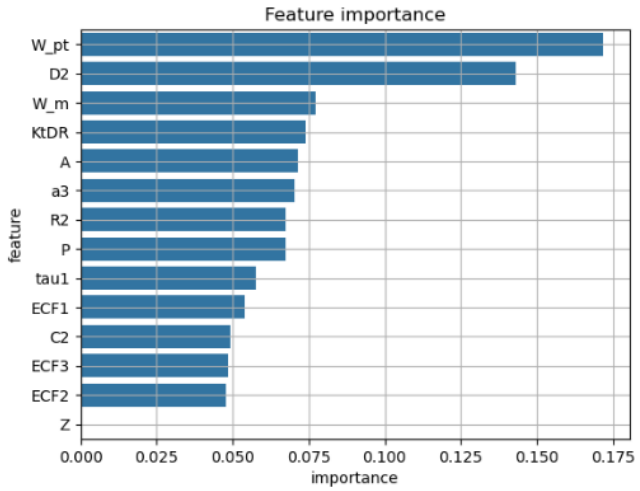

Рис. 10: Важность признаков

С порогом важности 0.05 было выбрано 10 более значимых признаков для последующего обучения полносвязной нейронной сети. Результаты классификации модели представлены на рисунке.

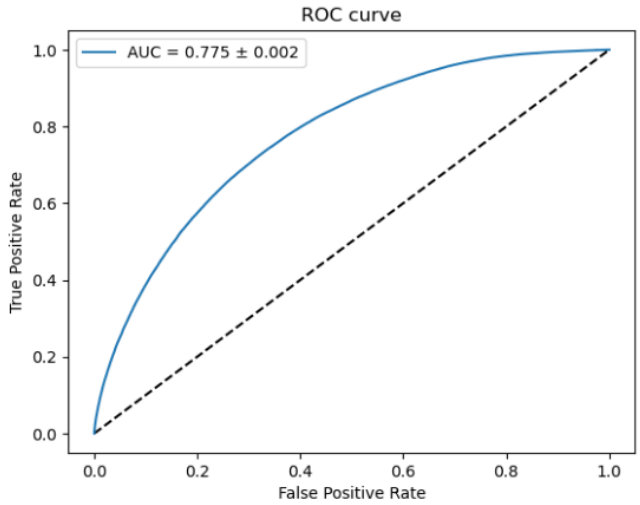

Рис. 11: ROC-кривая

3.2.3 Метод отбора признаков с использованием логистической регрессии и регуляризации L1

Логистическая регрессия находит взаимосвязь между целевой переменной и одной или несколькими независимыми переменными, оценивая вероятности с помощью своей логит-функции. Линейные модели работают так, что коэффициенты при признаках указывают насколько сильно изменение признака влияет на целевую переменную. При большом признаковом пространстве для уменьшения риска переобучения используют методы регуляризации. В частности Lasso регуляризация, а именно прибавление к функции потерь суммы абсолютных значений коэффициентов, позволяет обнулять коэффициенты неинформативных признаков.

С порогом значений коэффициентов 0.1 по модулю было выбрано 5 более значимых признаков для последующего обучения полносвязной нейронной сети.

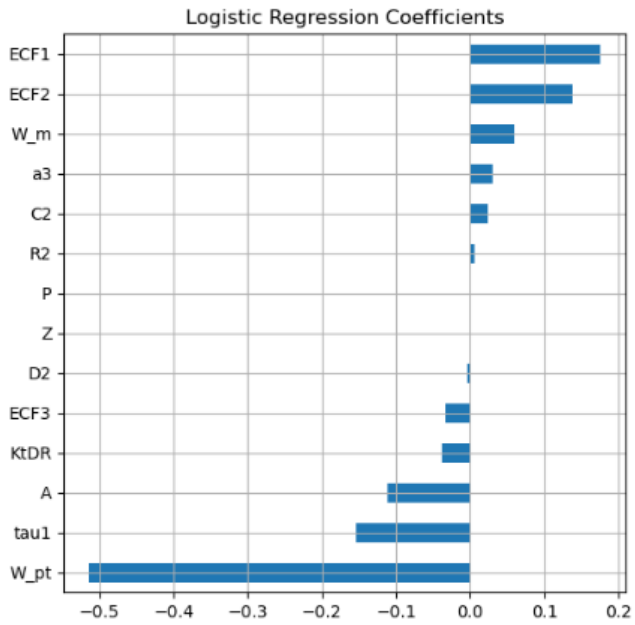

Рис. 12: Коэффициенты логистической регрессии при независимых переменных

Результаты классификации модели представлены на рисунке.

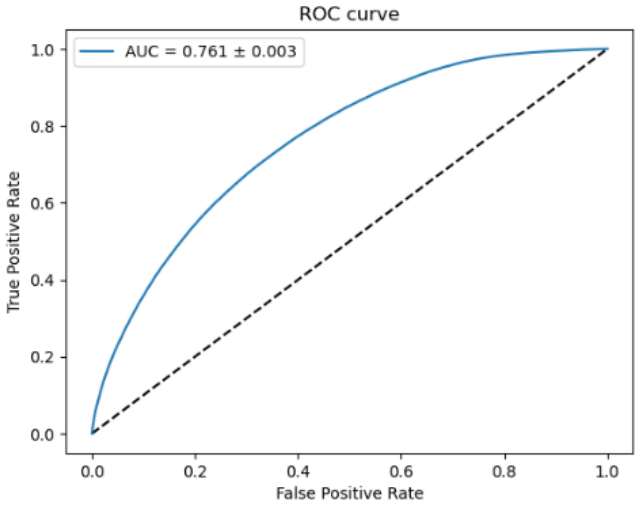

Рис. 13: ROC-кривая

3.2.4 Метод обертки: последовательный отбор признаков

Метод обертки - это процесс выбора признаков, основанный на конкретном алгоритме машинного обучения, который мы используем. Он следует подходу жадного поиска, оценивая все возможные комбинации признаков по определенному критерию. Методы оболочки обычно обеспечивают лучшую точность прогнозирования чем методы фильтрации. Оданко данные методы самые вычислительно затратные из всех, что описывались выше. В последовательном отборе признаков мы начинаем со всех доступных характеристик датасета и строим на их основе модель. Затем мы удаляем переменную из модели, которая

дает наихудшее значение меры оценки. Этот процесс продолжается до тех пор, пока не будет достигнут заданный критерий.

Для первой итерации признаков были получены следующие значения важности признаков.

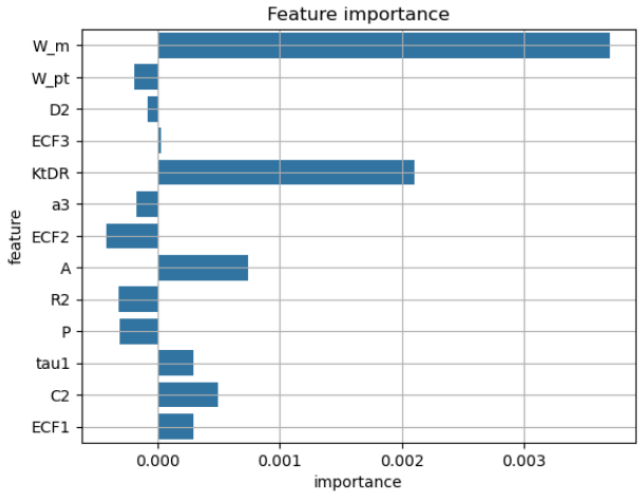

Рис. 14: Важность признаков в MLP

Результаты классификации модели с использованием всех признаков датасета представлены на рисунке.

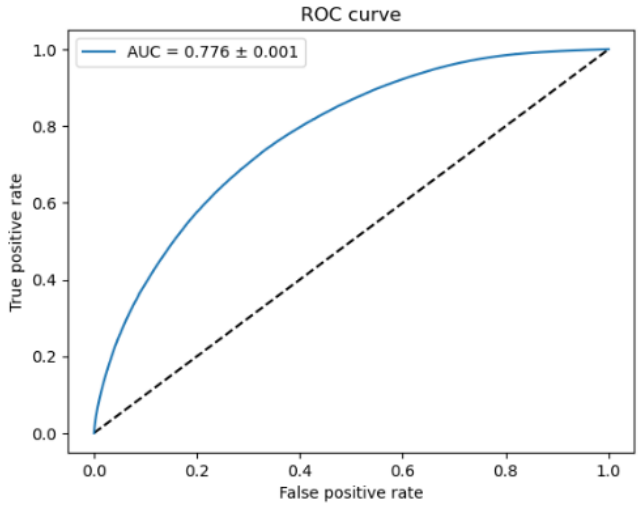

Рис. 15: ROC-кривая

4 Заключение

В рамках НИР за семестр проведено ознакомление с различными методами отбора признаков для дальнейшего обучения нейронной сети. Сформированы датасеты с различным набором признаков по результатам методов отбора. Проведено обучение и тестирование MLP для разных наборов признаков.

Наилучшее значение метрики AUC показала нейронная сеть, обученная на всех сформированных признаках.

Задачи для дальнейшей работы следующие:

1. Используя наилучший набор признаков, необходимо обучить нейронную сеть на всех фоновых процессах Монте-Карло данных с учетом веса событий;
2. Сравнить результаты работы обученной модели на реальных данных с работой модели на Монте-Карло симуляциях.

Список литературы

[1] Школа Анализа Данных. Учебник по машинному обучению. https://education.yandex.ru/handbook/ml/article/about.

[2] Stanford University School of Engineering. Свёрточные нейронные сети для визуального распознавания. https://www.reg.ru/blog/stenfordskij-kurs-lekciya-1-vvedenie/.

[3] Identification of Hadronically-Decaying W Bosons and Top Quarks Using High-Level Features as Input to Boosted Decision Trees and Deep Neural Networks in ATLAS at $\sqrt{s} = 13$ TeV. Technical report, CERN, Geneva, 2017. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2017-004.