МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН
Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ ИМЕНИ ГУМАРБЕКА ДАУКЕЕВА»
Институт Систем Управления и Информационных Технологии
Кафедра «Системы информационной безопасности»

«ДОПУЩЕН К ЗАЩИТЕ»
Зав.кафедрой к.п.н., доцент Бердибаев Рат Шындалиевич

_____ «8» июня 2020 г.
(подпись)

ДИПЛОМНЫЙ ПРОЕКТ

На тему: «Мониторинг пользовательских действий инструментами машинного обучения»_____

Специальность: Системы Информационной Безопасности_____

Выполнил: Сапарғали Есболат Еділұлы_____ Группа: СИБ-16-2___

Научный руководитель: к.т.н., доцент Сатимова Елена Григорьевна_____

Консультанты: старший преподаватель Зуева Екатерина Александровна_____
по специальной части:
старший преподаватель Дмитриева Маргарита Валерьевна_____

_____ «31» мая 2020г.
(подпись)

по безопасности жизнедеятельности:
д.х.н., профессор Приходько Николай Георгиевич_____

_____ «27» мая 2020г.
(подпись)

Нормоконтролер: старший преподаватель Дмитриева Маргарита Валерьевна_
(ученая степень, звание, Ф.И.О.)

_____ «8» июня 2020г.
(подпись)

Рецензент: научный сотрудник РГП «Институт информационных и вычислительных технологий» Комитета наук МОН РК PhD Бегимбаева Енлик Ериковна_____

_____ «8» июня 2020г.
(подпись)

Алматы 2020

Задание на выполнение дипломного проекта

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН
Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ ИМЕНИ ГУМАРБЕКА ДАУКЕЕВА»

Институт Систем Управления и Информационных_____

Кафедра «Системы Информационной Безопасности»_____

Специальность «Системы Информационной Безопасности»_____

ЗАДАНИЕ
на выполнение дипломного проекта

Студенту Сапарғали Есболат Еділұлы_____

Тема проекта «Мониторинг пользовательских действии инструментами машинного обучения»_____

Утверждена приказом по университету № 563 от «30» апреля 2020г.

Срок сдачи законченного проекта «8» июня 2020 г.

Исходные данные к проекту (требуемые параметры результатов исследования (проектирования) и исходные данные объекта): дистрибутив языка программирования Python «Anaconda» для разработки классификаторов; наборы данных Университет Карнеги — Меллона, Калифорниийского технологического института и Канадского института кибербезопасности для исследований закономерностей в машинном обучении.

Перечень вопросов, подлежащих разработке в дипломном проекте или краткое содержание дипломного проекта: изучение предметной области машинного обучения и алгоритмов «Деревьев решений»; исследование и подготовка больших наборов данных, построение модели классификаторов, разработка программных кодов для обнаружения аномалий DDOS атак, фишинга, сетевого трафика, в журналах логов предприятии; анализ результатов по завершению построений и использования моделей классификаторов; расчёт рисков ИБ; произведение расчетов, согласно стандартам БЖД; подведение итогов.

Перечень графического материала (с точным указанием обязательных чертежей): Глава 1 содержит 6 рисунков, в главе 2 представлено 7 рисунков,

в главе 3 представлено 56 рисунков, в 4 главе представлено 6 рисунков, в 5 главе представлен 3 рисунка.

Основная рекомендуемая литература: Джордан М.И. и Митчелл. Т.М. «Машинное обучение: тенденции, перспективы и перспективы»; Бучак А. и Гувен Э. «Обзор методов интеллектуального анализа данных и машинного обучения для обеспечения безопасности»; Байер У., Хабиби И., Бальзаротти Д., Кирда Э. и Крюгель К. «Взгляд на текущее поведение вредоносных программ».

Консультации по проекту с указанием относящихся к ним разделов проекта

Раздел	Консультант	Сроки	Подпись
Анализ рисков информационной безопасности	старший преподаватель Дмитриева Маргарита Валерьевна	17.02.2020 – 9.05.2020	
Безопасность жизнедеятельности	к.т.н. доцент Приходько Николай Георгиевич	17.02.2020 – 9.05.2020	

График
подготовки дипломного проекта

Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю	Примечание
Изучение предметной области	17.02.2020 – 25.02.2020	
Исследование и подготовка больших наборов данных	26.02.2020 – 4.03.2020	
Построение модели классификаторов	5.03.2020 – 24.03.2020	
Разработка программ для обнаружений аномалии	25.03.2020 - 7.04.2020	
Анализ результатов	8.04.2020 – 25.04.2020	
Расчет рисков ИБ	26.04.2020 - 9.05.2020	
БЖД	29.04.2020 - 9.05.2020	

Дата выдачи задания «30» апреля 2020г.

Заведующий кафедрой _____ Бердибаев Рат Шындалиевич
(подпись)

Научный руководитель проекта _____ Сатимова Елена Григориевна
(подпись)

Научный руководитель проекта _____ Зуева Екатерина Александровна
(подпись)

Задание принял к исполнению студент _____ Сапарғали Есболат Еділұлы
(подпись)

Аңдатпа

Бұл дипломдық жобада басып кірудің анықтау жүйелерінде қолданылатын машиналарды оқыту әдістері талданады. Қолданыстағы жүйелердің негізгі шектеулері анықталды және нейрондық желі бірнеше профильдерге жүктелген мәліметтерден маңызды қауіптерді анықтай алатын мәндер табылды.

Өміртіршілік қауіпсіздік бөлімінде машинаны оқыту алгоритмдерін талдау үшін оңтайлы жұмыс жағдайларының талдауы анықталған.

Тәуекелдерді бағалау бөлімінде Anaconda платформасы мен оның ресурстарының тәуекелдік сипаттамаларын анықтайды және екі параметрді бағалау әдісі көрсетілді.

Аннотация

В дипломном проекте проведен анализ методов машинного обучения, применяемых в системах обнаружения вторжений. Определились основные ограничения текущих систем и были найдены значения, при которых нейронная сеть может определять существенные угрозы с загруженных данных по нескольким профилям.

В разделе БЖД определен анализ оптимальных условий труда для разработки алгоритмов машинного обучения.

В разделе оценка рисков определены характеристики рисков платформы Anaconda и ее ресурсов. Показан способ оценки двумя параметрами.

Annotation

This graduation project analyzes machine learning methods used in intrusion detection systems. The main limitations of the current systems were determined and the values were found at which the neural network can identify significant threats from the downloaded data on several profiles.

In the section of life safety, an analysis of optimal working conditions for the development of machine learning algorithms is defined

The risk assessment section identifies the risk characteristics of the Anaconda platform and its resources. Method for assessing two parameters was shown.

Содержание

Введение

Машинное обучение применяется в широком спектре областей, где оно демонстрирует свое превосходство над традиционными алгоритмами на основе правил. Эти методы интегрируются в системах обнаружения с целью поддержки или замены первого уровня аналитиков безопасности. Для наилучшей эффективности в информационной безопасности машинное обучение должно быть оценено с должной осмотрительностью.

Чтобы не отставать от современных злоумышленников, информационная безопасность должна развиваться вместе с ними, не полагаясь на вмешательство человека. Вот где математика искусственного интеллекта и машинное обучение имеют преимущество. Классификация «доброкачественных» данных из «злонамеренных» на основе математических факторов риска позволяет научить компьютер производить соответствующую характеристику этих файлов в режиме реального времени. Ядром такого подхода является масштабируемый «мозг» для обработки данных, способный применять высокоразвитые математические модели к огромным объемам данных.

Масштаб используемых данных и тенденция к смещению количеству необходимых вычислений делают людей неспособными использовать эти данные, чтобы определить, является файл вредоносным. Большинство охранных компаний все еще полагаются на людей при принятии таких решений, нанимая большие команды для изучения миллионов файлов. У людей нет ни умственных способностей, ни физической выносливости, чтобы не отставать от объема и сложности современных угроз.

Машинное обучение фокусируется на прогнозировании, основанном на свойствах, полученных из более ранних данных. Таким образом, можно теперь отличить вредоносные файлы от доброкачественных. Интеллектуальный анализ данных фокусируется на обнаружении ранее неизвестных свойств данных, чтобы их можно было использовать в будущих решениях по машинному обучению.

1 Анализ предметной области

1.1 Введение в машинное обучение

Машинное обучение - это раздел искусственного интеллекта, которое предоставляет системам возможность автоматически учиться и совершенствоваться на основе опыта без явного программирования. Машинное обучение направлено на разработку компьютерных программ, которые могут получить доступ к данным и использовать их для обучения. С помощью машинного обучения компьютеры находят важную информацию без указания места поиска, делают это, используя алгоритмы, которые учатся на данных в итеративном процессе. На высоком уровне это способность адаптироваться к новым данным с помощью итераций. В основном, приложения извлекают уроки из предыдущих вычислений и используют распознавание образов для получения надежных и информированных результатов.

Хотя концепция машинного обучения существует уже давно, способность автоматизировать применение сложных математических вычислений к большим данным набирает силу только в последние несколько лет. Машинное обучение предоставляет разумные альтернативы анализу огромных объемов данных. Разрабатывая быстрые и эффективные алгоритмы и управляемые данными модели для обработки данных в режиме реального времени, машинное обучение может давать точные результаты и анализ.[1]

1.2 Алгоритмы машинного обучения. Типы алгоритмов

Из-за сложности обучения, машинное обучение было разделено на две основные области: контролируемое обучение и обучение без учителя. Каждый вид имеет определенную цель и действие в рамках машинного обучения, дает конкретные результаты и использует различные формы данных. Приблизительно 70 процентов машинного обучения - это обучение под наблюдением, в то время как обучение без учителя варьируется от 10 до 20 процентов.

При неконтролируемом обучении данные обучения неизвестны и не имеют маркировки. Эти данные поступают в алгоритм машинного обучения и используются для обучения модели. Обученная модель пытается найти шаблон и дать желаемый ответ. Кластеризация и ассоциация используется в настоящее время для неконтролируемого обучения.

Кластеризация указывают на сходные характеристики. Подходы включают k-средства и иерархическую кластеризацию. Методы кластеризации имеют ограниченную масштабируемость, но они представляют собой гибкое решение, который обычно используется в качестве предварительной фазы перед принятием контролируемого алгоритм или для целей обнаружения аномалий.

Ассоциация направлены на выявление неизвестных закономерностей между данными и подходят для целей прогнозирования. Тем не менее, они

имеют тенденцию производить вывод без действующих правил, следовательно, они должны быть объединены с точными осмотрами человека.

В контролируемом обучении используется известные или помеченные данные для данных обучения. Поскольку данные известны, обучение контролируется. Входные данные проходят через алгоритм машинного обучения и используются для обучения модели. Процесс обучения продолжается до тех пор, пока модель не достигнет желаемого уровня точности данных тренировки. Алгоритмы, используемые в настоящее время для контролируемого обучения:

— линейная регрессия используется для оценки реальных значений. На основе непрерывных переменных устанавливается связь между независимыми и зависимыми переменными, подбирая лучшую линию. Эта линия наилучшего соответствия называется линией регрессии и представлена линейным уравнением.

— логистическая регрессия является классификацией, а не алгоритм регрессией. Используется для оценки дискретных значений (двоичные значения, такие как 0/1, да / нет, истина / ложь) на основе заданного набора независимых переменных.

— деревья решений тип контролируемого алгоритма обучения, который в основном используется для задач классификации, который работает как для категориальных, так и для непрерывных зависимых переменных разбивая на два или более однородных множества. Это делается на основе наиболее значимых атрибутов / независимых переменных, чтобы создать как можно более четкие группы.

— байесовский алгоритм является вероятностным классификатором, в котором характеристики входного набора данных независимы друг от друга. Они масштабируемы и не требуют огромных наборов обучающих данных и производят заметные результаты.[1]

1.3 Алгоритмы Isolation forests, Random forests. Детектор аномалий

Алгоритмы «деревья решений» считаются наиболее часто используемыми методами обучения под наблюдением. Эти алгоритмы позволяют прогнозировать модели с высокой точностью, стабильностью и простотой интерпретацией. В отличие от линейных моделей довольно хорошо отображают нелинейные отношения. Они адаптируются при решении любых проблем (классификация или регрессия). Методы дерева решений строят модель решений, принятых на основе фактических значений атрибутов (рисунок 1.1).

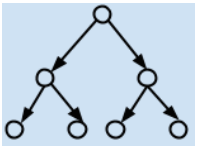


Рисунок 1.1 – Алгоритм «деревья решений»

Решения распадаются на древовидные структуры до тех пор, пока не будет принято решение для записи предсказаний. Деревья решений обучаются на данных для задач классификации и регрессии. Они часто бывают быстрыми и точными и являются большим фаворитом в машинном обучении.[2]

Наиболее популярные алгоритмы «Дерева решений»:

— случайный лес;
— решение «Пень»;
— деревья условных решений;
— изолированный лес;
— дерево классификации и регрессии;
— итеративный Дихотомизатор 3;
— Хи-квадрат;
— автоматическое обнаружение взаимодействия.

Случайный лес - это набор деревьев решений и рассматривает выходные данные каждого дерева перед предоставлением единого ответа. В Случайном Лесу есть ансамбль деревьев решений - Лес.

Каждое дерево решений является условным классификатором: на каждом узле данное условие проверяется на соответствие одному или нескольким признакам проанализированных данных. Эти методы эффективны для больших наборов данных и превосходны в мульти классовых задачах, но более глубокие деревья могут привести к переоснащению (рисунок 1.2).

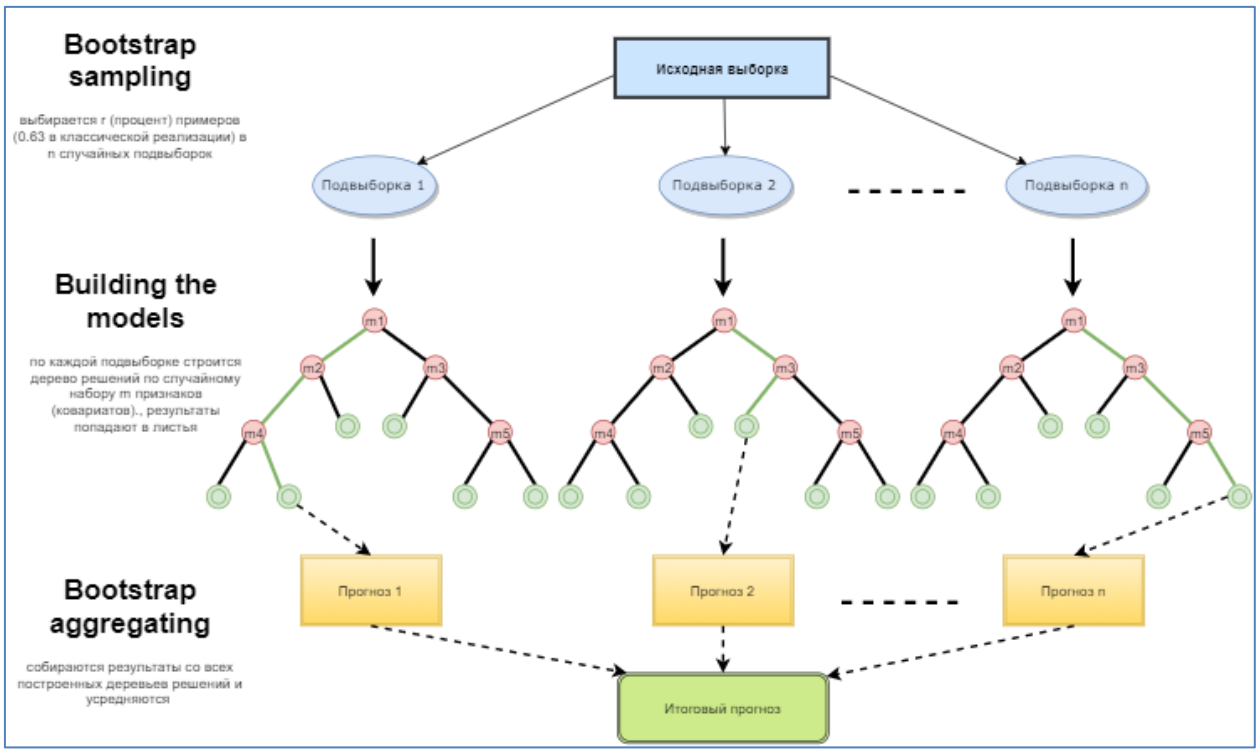


Рисунок 1.2 – Блок-схема Случайного Леса

Алгоритм случайного леса состоит из двух этапов: первый - создание случайного леса, второй - сделать прогноз на основе классификатора случайного леса, созданного на первом этапе:

— произвольно выбирается k объекты из общего числа m объектов, где k << m;

— среди функций k вычисляется узел d для оптимального разделения;

— разделяется узел на дочерние узлы для лучшего разбиения;

— повторяется шаги от a до c, пока не будет достигнуто число l;

— создается лес, повторяя шаги от a до d для n раз, чтобы создать n количество деревьев.

На рисунке 1.3 показан процесс случайного выбора функций:

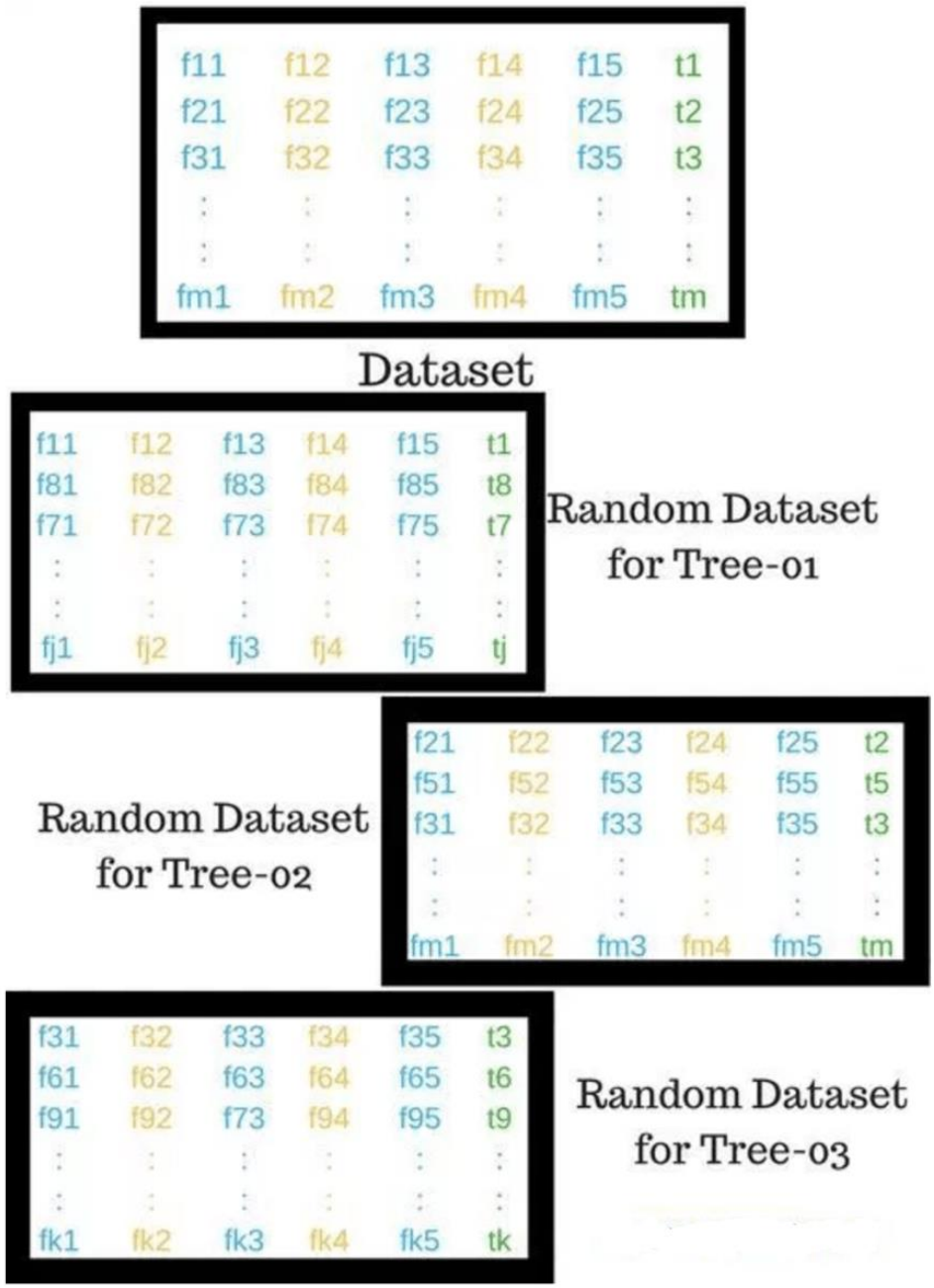


Рисунок 1.3 - Процесс случайного выбора функций

На следующем этапе с созданным классификатором случайных лесов делается прогноз. Псевдокод случайного лесного предсказания показан ниже:

— принимает возможности тестирования и используется правила каждого случайно созданное дерево решений, чтобы предсказать исход и сохраняют предсказанный исход (цель);

— подсчитывается голоса для каждой прогнозируемой цели;

— рассматривается прогнозируемая цель с высоким рейтингом как окончательный прогноз из алгоритма случайного леса.

Каждое дерево посажено и выращено следующим образом:

— если количество случаев в обучающем наборе равно N, тогда выборка из N случаев берется случайным образом, но с заменой. Этот образец будет обучающим набором для выращивания дерева.

— если имеется M входных переменных, число m << M указывается так, что в каждом узле m переменных выбирается случайным образом из M, и наилучшее разделение по этим m используется для разделения узла. Значение m поддерживается постоянным при выращивании леса.

— каждое дерево выращивается в максимально возможной степени. Здесь нет обрезки.[2]

Идентификация событий в наборе данных, которые не соответствуют ожидаемой схеме называется обнаружение аномалий. В приложениях эти события могут иметь решающее значение. Это могут быть случаи вторжений или мошенничества.

Для этих случаев часто используется «Изолированный лес», который опирается на наблюдение, чтобы легко выделить выбросы, в то время как сложнее описать нормальную точку данных. Изолированный лес - это гибкий, простой в использовании алгоритм машинного обучения, который дает большую часть времени даже без настройки гиперпараметров. Этот алгоритм был использован для выявления таких аномалий в виде эксперимента с обычными данными без приставки ИБ. На следующих шагах показано, как применялся алгоритм Isolation Forest для обнаружения аномалий.

Происходит импортирование необходимых библиотек и установка случайного начального числа с np.random.RandomState(15). Дальше, создается набор данных для обучения и тестовый набор (Рисунок 1.4).

```
X_train = 0.5 * random_seed.randn(400, 2)
X_train = np.r_[X_train + 3, X_train]
X_train = pd.DataFrame(X_train, columns = ['x', 'y'])
```

Рисунок 1.4 – Создания набора данных для обучения

Создан набор наблюдений за выбросами. Генерируются из распределения, отличающийся от обычных наблюдений (Рисунок 1.5).

```
X_nabl = random_seed.uniform(size=(20, 2))
X_nabl = pd.DataFrame(X_nabl, columns = ['x', 'y'])
```

Рисунок 1.5 – Создания набора данных для выбросов

На рисунке 1.6 показаны сгенерированные данные.

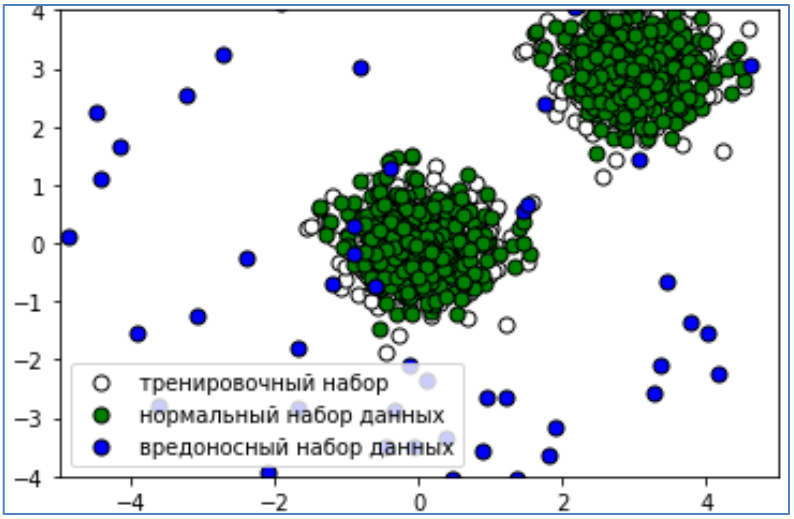


Рисунок 1.6 – График со сгенерированными данными

Обучена модель Isolation Forest на данных обучения и показано как работает алгоритм на рисунке 1.7:

```
from sklearn.ensemble import IsolationForest
clf = IsolationForest() clf.fit(X_train)
y_pred_train = clf.predict(X_train) y_pred_nabl = clf.predict(X_nabl)
X_nabl = X_outliers.assign(pred = y_pred_nabl)
```

Рисунок 1.7 – Обучения модели

Построен прогноз «Изоляционного леса», чтобы увидеть, сколько из выбросов было поймано. Это выполняется на обычных данных тестирования. Добавлена прогнозируемая метка x_testirovaniye. Выведены результаты, классификатор помечает нормальные данные тестирования (рисунок 1.8).

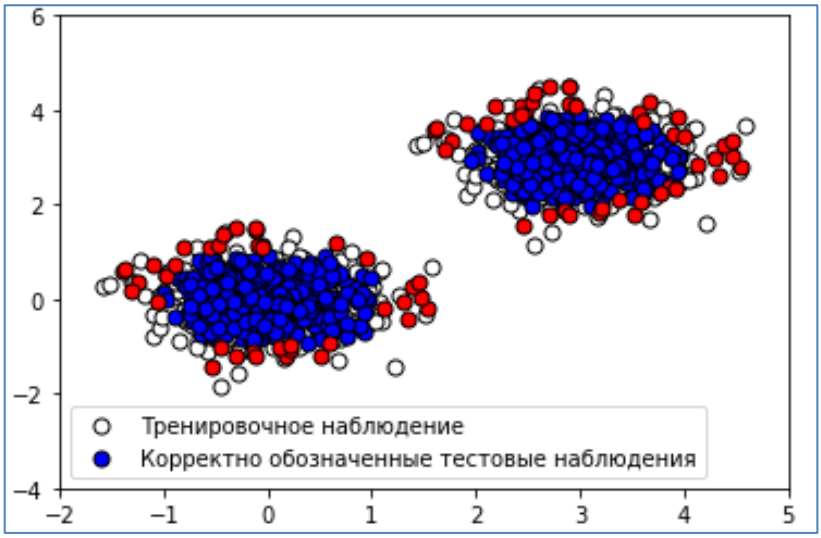


Рисунок 1.7 – Прогноз Изоляционного леса

Модель «Изоляционный лес» показала хорошие результаты при захвате аномальных точек. В случае, когда нормальные наблюдения были классифицированы как выбросы, было довольно много ложных негативов. После настроек параметров модели, было уменьшена выборка данных. Этот аномальный набор данных имеет распределение, отличающееся от данных обучения и остальных данных тестирования. Обрисовывая данные, некоторые выпадающие точки выглядят неотличимыми от нормальных точек. Это гарантирует, что классификатор будет иметь значительный процент ошибочных классификаций. Обученный экземпляр Isolation Forest предсказывает, являются ли данные тестирования нормальными или аномальными. Благодаря этому удалось зафиксировать большинство угроз.

1.4 Связь машинного обучения и информационной безопасности

Привлекательность и распространенность машинного обучения растет. Существующие методы совершенствуются и их способность понимать и отвечать на реальные вопросы оценивается в области ИБ. В некоторых сценариях алгоритмы машинного обучения представляют собой лучший выбор по сравнению с традиционными алгоритмами, основанными на правилах. Эта тенденция также влияет на область информационной безопасности, где используются системы обнаружения, модернизированные с компонентами машинного обучения. Аналитики первого уровня в сети могут извлечь выгоду из инструментов обнаружения и анализа на основе машинного обучения (рисунок 1.6).

Обнаружение вторжения направлено на обнаружение незаконных действий на компьютере или в сети через системы обнаружения вторжений (IDS). IDS широко используются в современных корпоративных сетях. Эти системы были традиционно основаны на шаблонах известных атак, но современные развертывания включают другие подходы к обнаружению аномалий и классификации на основе машинного обучения.

Обнаружение ботнета направлено на выявление связи между зараженным компьютером в контролируемой сети и внешним командно-управляющим сервером. Несмотря на множество исследовательских предложений и коммерческих инструментов, он автоматически генерирует доменные имена и часто используются зараженной машиной для связи с внешним сервером, периодически генерируя новые имена хостов. Он представляет реальную угрозу для организаций, которые опираются на методы обработки языка, основанных на статических черных списках доменных имен.

Анализ вредоносных программ является чрезвычайно актуальной проблемой, потому что современные вредоносные программы могут автоматически генерировать новые варианты с такими же вредоносными эффектами, но появляющиеся как совершенно разные исполняемые файлы. Эти полиморфные и метаморфические особенности превосходят традиционные правила идентификации вредоносных программ. Методы машинного обучения могут использоваться для анализа и приписывания их к классу вредоносных программ.

Обнаружение спама и фишинга включает в себя большой набор методов, направленных на снижение потенциальной опасности, вызванной нежелательными электронными письмами. В наше время, нежелательные электронные письма, представляют собой предпочтительный способ фишинга. Злоумышленник устанавливает первую точку опоры в сети предприятия. Фишинговые письма состоят из вредоносных программ или из ссылок на скомпрометированные веб-сайты. Обнаружение спама и фишинга становится все труднее из-за передовых стратегий уклонения, используемых злоумышленниками для обхода традиционных фильтров. Подходы машинного обучения могут улучшить процесс обнаружения спама.[2]

Рисунок 1.6 – Этапы машинного обучения

1.5 Программирование с основными инструментами машинного обучения в Python. Анализ библиотек Pandas, Matplotlib и Scikit-learn

Python - самый быстрорастущий язык программирования. В частности, исследователи данных используют эффективный синтаксис Python, обучаемость и простую интеграцию с другими языками, такими как C и C ++.

С недавним всплеском интереса к машинному обучению и искусственному интеллекту создаются мощные библиотеки с открытым исходным кодом для машинного обучения и приложений обработки данных. Среди них Numpy, Pandas, SciKit Learn.

Numpy расшифровывается как числовой питон. Это библиотека с открытым исходным кодом для языка программирования Python. Numpy добавляет поддержку больших многомерных матриц и массивов, а также гигантский набор математических функций для работы с этими массивами и матрицами. Цель библиотеки состоит в том, чтобы упростить преобразование сложных функций или вычисление некоторого анализа данных.

Pandas - это библиотека с открытым исходным кодом, которая позволяет легко использовать структуры данных и инструменты анализа данных для языка программирования Python. Используется не только для анализа данных, но и для машинного обучения. Pandas построена вокруг объектов DataFrame. Все данные поступают в один большой DataFrame, где

можно выбрать некоторые образцы или другие манипуляции с данными, если это необходимо. Функции Pandas:

— чтение и запись данных между структурами данных в памяти и различными форматами, такими как csv, текстовые файлы, файлы excel, базы данных sql;

— высокопроизводительное объединение наборов данных;

— выравнивание данных и интегрированная обработка отсутствующих данных.

SciKit Learn - это библиотека, которая содержит множество классических алгоритмов машинного обучения, таких как, машины опорных векторов, map, классификаторы ближайших соседей, случайные леса и алгоритмы регрессии. Как и Pandas и Numpy, это библиотека Python, но SciKit более специфична для машинного обучения. SciKit Learn включает в себя от манипулирования наборами данных до обработки метрик. Такие функции, как классификация, регрессия, кластеризация, режим, выбор модели встроены в эту библиотеку. Включает в себя варианты обучения, как под наблюдением, так и без него. Таким образом, это в конечном итоге эффективный инструмент для статистического моделирования.[3]

Выводы

В данной главе была рассмотрена предметная область дипломного проекта и алгоритмы машинного обучения. Был подробно расписан тип контролируемого обучения - деревья решений. Показаны разные способы реализации деревьев решений: «Изолированный лес» и «Случайный лес». Был произведен поиск пересечения граней машинного обучения и информационной безопасности, основных функций библиотеки Python, используемых в машинном обучении.

С помощью машинного обучения разрабатываются программы, которые могут получить доступ к данным и используются их для обучения. Алгоритмы с Деревьями решений считаются одним из лучших и наиболее часто используемых методов обучения под наблюдением. В отличие от линейных моделей, они довольно хорошо отображают нелинейные отношения и адаптируются при решении любых проблем (классификация или регрессия). Методы Деревьев решений строят модель, принятой на основе фактических значений атрибутов в данных.

Для машинного обучения и приложений для обработки данных создаются мощные библиотеки с открытым исходным кодом. Среди них Numpy (поддержку больших многомерных матриц и массивов), Pandas (использовать структуры данных и инструменты анализа данных), SciKit Learn (содержит множество классических алгоритмов машинного обучения).

Алгоритмы машинного обучения влияют на область ИБ, где используются система обнаружения. Подходы машинного обучения могут улучшить процесс обнаружения спама и фишинга, в котором становится все труднее находить аномалии из-за передовых стратегий уклонения,

используемых злоумышленниками для обхода традиционных фильтров обнаружения спама.

2 Машинное обучение при мониторинге вредоносных событий

2.1 Обзор решений систем обнаружений, вторжений и аномалий

В начале 2000-х годов появилось второе поколение инструментов безопасности (рисунок 2.1). Это облегчало сортировку предупреждений путем сопоставления нескольких источников данных в озере данных безопасности, называемом инструментами управления информацией о событиях и безопасности (SIEM). SIEM-системы были успешными, но в эпоху Больших данных они работают медленно и не имеют интеллектуального уровня. Средства управления информацией и событиями безопасности объединяют средства управления информацией о безопасности и управления событиями безопасности, чтобы обеспечить надежный фасад для ваших систем. SIEM собирает связанные с безопасностью данные из различных источников и анализирует их.

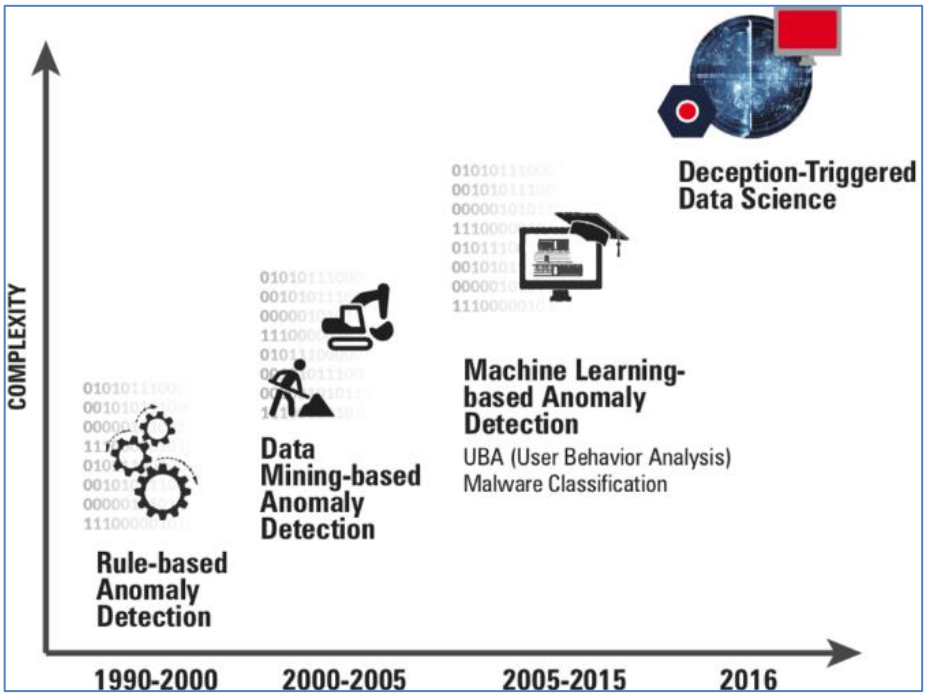


Рисунок 2.1 - Эволюция науки безопасности данных

Системы обнаружения и предотвращения вторжений работают, активно отслеживая сетевой трафик на наличие необычных шаблонов или подозрительного поведения. Основное различие между IDS и IPS заключается в том, что IDS будет оповещать о необычном трафике, это пассивная система, которая не предотвращает и не останавливает активность. В отличие от IPS, как правило, интегрирует функции, подобные брандмауэру, чтобы вносить активные изменения, предотвратить поток подозрительных данных и запретить трафик. Обе технологии в значительной степени основаны на сигнатурах и работают путем определения моделей

трафика, которые похожи на известные методы атаки. Это означает, что они могут быть неэффективны против последних угроз, если для атаки еще не установлена подпись.

Системы обнаружения аномалий основаны на моделях нормального поведения хостов и сетей. Всякий раз, когда есть существенное отклонение от нормального поведения, тогда они вызывают оповещения. Алгоритмы на основе аномалий используются в сетях для обнаружения:

— аномальных портов;
— необычного трафика с хоста;
— чрезмерных сбоев DNS;
— конечных точек, имеющих необычные процессы/приложения/ изменения реестра;
— пользователей/хостов, имеющих необычное поведение.

Важно отметить, IDS может только идентифицировать атаку. Сам по себе он не может предотвратить возможную атаку или остановить продолжающуюся атаку от достижения и/или компрометации цели. IDS объединяет данные трафика, выявляет любые аномалии и подозрительные действия в этих данных. IDS может вести логи и предупреждать администраторов в случае взлома или атаки.

Чтобы обеспечить полную защиту решения IDS и SIEM работают вместе. Инструменты IDS обнаруживают все виды подозрительных действий, нарушений или событий безопасности, которые происходят в пределах вашей системы и сети. Затем SIEM информирует о таких действиях, чтобы уведомить администраторов для принятия необходимых действий.[4]

2.2 Решение проблем безопасности данных с машинным обучением

Большинство систем обнаружения аномалий вызывают высокие ложные тревоги и нуждаются в большом внимании аналитиков безопасности для проверки предупреждений. С достижениями в области больших данных возникла новая форма науки о безопасности данных. Это дало начало анализу поведения пользователей и сущностей, в котором используются методы обнаружения аномалий для оповещений в реальном времени в случае подозрительного поведения хостов / пользователей в сети предприятия. Наука о данных используется для профилирования поведения противника и его движения в сети.

Наука о данных коррелирует данные о событиях безопасности с ложными предупреждениями высокой точности, чтобы сформировать лучшего понимания поведения противника (рисунок 2.2).

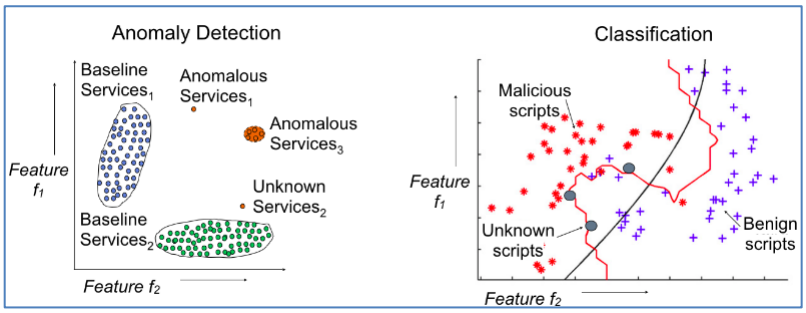


Рисунок 2.2 - Методы защиты данных безопасности

Когда злоумышленники попадают в сеть предприятия, им сначала необходимо выяснить, где они находятся. После этого они двигаются к своим целям и проводят атаку. Во время этих разведывательных запросов и движений злоумышленники обычно оставляют некоторые следы или сигналы. Эти сигналы присутствуют в данных, и их присутствие может быть обнаружено с помощью инструментов машинного обучения для своевременного оповещения.

Ранее для переноса всех данных, использовались SIEM-системы. Используя алгоритмы, можно соединить точки и найти шаблоны, которые раньше было трудно найти вручную из-за отсутствия аналитиков безопасности.

Большинство данные безопасности не имеют меток, что затрудняет применение сетей глубокого обучения для большого числа случаев использования ИБ. Тем не менее, отрасль ИБ решает эту проблему путем создания меток классов для нескольких вариантов одновременного использования. Например, обнаружение вредоносных программ и ранжирование вредоносных веб-сайтов и доменов DNS в основном выполняется с использованием методов машинного обучения. Еще одним успешным примером использования науки о данных для обеспечения безопасности является определение базового уровня каждого пользователя или устройства или ненормального поведения и возникновения аномалий.[5]

Эти аномалии на основе поведения пользователя более 100 раз меньше, чем аномалии на основе правил. Однако их величины все еще довольно высоки, и многие из них оказываются ложноположительными**.**

Чтобы реализовать обнаружение вторжения типов аномалий, раньше извлекали характеристики несанкционированного доступа из наблюдаемых событий и использовали это в качестве подсказки для классификации как нормальной или несанкционированной связи**.** Вероятность также может быть рассчитана во время классификации, что является предпочтительным при работе. Эта задача использует логистическую регрессию для реализации системы обнаружения вторжений с вышеуказанной концепцией, но не высокой вероятностью.

Для проверки текущего состояния науки данных, проведен анализ алгоритма линейной регрессии. Поскольку логистическая регрессия включает контролируемое обучение, необходимо заранее подготовить

тренировочные данные, которые включают в себя характеристики различных атак.

Набор данных был создан в 2010 году, поэтому его содержимое немного устарело, но, поскольку оно содержит нормальную связь и различные виды ненормальной связи, её использует в качестве обучающих данных для системы обнаружения вторжений:

from sklearn import linear_model

from sklearn import metrics

linear_model - пакет логистической регрессии scikit-learn. Этот пакет содержит различные классы для выполнения логистической регрессии.

A metrics является пакетом для оценки точности классификации. Из данных обучения получают столбцы и метку, связанную с выбранным количеством признаков. Здесь значение данных каждой функции нормализуется для повышения точности обучения ((X_train - X_train.mean()) / X_train.mean()) (Рисунок 2.3).

```
df_train = pd.read_csv('..\\dataset\\kddcup_train.csv')
X_train = df_train.iloc[:, [0, 7, 11, 12, 13, 34, 37, 38]]
X_train = (X_train - X_train.mean()) / X_train.mean()
y_train = df_train.iloc[:, [42]]
```

Рисунок 2.3 – Нормализация данных

Получены данные тестирования, данные для обучения (X_test, y_test): Определена модель логистической регрессии: Logreg.fit(X, train, y_train). fit - метод аргумента, в котором каждая величина функции обучается. Это создает модель логистической регрессии. Получение результатов классификации: y, pred = logreg, predict(X,test). Модель predict выполняет классификацию тестовых данных и выдает результат классификации (рисунок 2.4).

```
train_time   : 10.235417526819267 [sec]
predict_time : 0.023557101303234518 [sec]
score : 0.8996588708933895
--------------------------------------------------
label            predict         probability
normal.          normal.         0.9998065736764143
normal.          normal.         0.9998065736764143
normal.          normal.         0.9998065736764143
normal.          normal.         0.9917140229898096
```

```
...snip...

guess_passwd.    guess_passwd.   0.478403815
guess_passwd.    guess_passwd.   0.499999391
guess_passwd.    guess_passwd.   0.38504377
guess_passwd.    normal.         0.999790317
```

Рисунок 2.4 - Выводы результата классификации

Score указывает на точность классификации модели, которая составляет 89,9%.

Точность классификации во многом зависит от точности выбора признаков или алгоритмов. В этой модели использован набор данных 2012 года в качестве обучающих данных.

Модель будет собирать информацию, чтобы не упускать из виду функции, полезные для обнаружения вторжений.

2.3 Обзор наборов данных. Формат CSV

Сотрудники, работающие с любыми данными, редко получают структурированные табличные данные. Таким образом, любой специалист по данным должен знать о различных форматах файлов, общих проблемах с ними и о эффективных способах обработки этих данных.

Формат файла - это стандартный способ кодирования информации для хранения в файле. Во-первых, формат файла указывает, является ли файл двоичным или ASCII-файлом. Во-вторых, это показывает, как организована информация. Например, формат файла значений с разделителями-запятыми хранит табличные данные в виде простого текста.

В формате файла электронной таблицы данные хранятся в ячейках. Каждая ячейка организована в строки и столбцы. Столбец в файле электронной таблицы может иметь разные типы. Например, столбец может иметь строковый тип, тип даты или целочисленный тип. Некоторые из наиболее популярных форматов файлов электронных таблиц - это значения, разделенные запятыми (CSV), электронная таблица Microsoft Excel (xls) и электронная таблица Microsoft Excel Open (xlsx).

Файлы CSV можно использовать с большинством программ для работы с электронными таблицами, например, Microsoft Excel или Google Spreadsheets. Они отличаются от других типов файлов электронных таблиц, поскольку в файле может быть только один лист, также они не могут сохранить ячейку, столбец или строку. Каждая строка в файле CSV представляет собой наблюдение, или она обычно называется записью. Каждая запись может содержать одно или несколько полей, разделенных запятой.

Файл с разделителями-запятыми (CSV) представляет собой простой текстовый файл, который содержит список данных. Эти файлы часто используются для обмена данными между различными приложениями. Например, базы данных и менеджеры контактов часто поддерживают файлы CSV.

В основном они используют запятую для разделения данных, но иногда используют другие символы, такие как точки с запятой. Идея состоит в том, что вы можете экспортировать сложные данные из одного приложения в файл CSV, а затем импортировать данные из этого файла CSV в другое приложение.

Файл CSV имеет довольно простую структуру. Это список данных, разделенных запятыми. Например, предположим, у вас есть несколько контактов в менеджере контактов, и вы экспортируете их как файл CSV. Вы получите файл, содержащий текст, подобный этому:

Имя, электронная почта, номер телефона, адрес

Сапаргали Есболат, Есбол @mail.ru, 1234567890,123 Бейкер Стрит

Файлы могут быть более сложными и содержать тысячи строк, больше записей в каждой строке или длинные строки текста. Некоторые файлы CSV

могут даже не иметь заголовков в верхней части, а некоторые используют кавычки для окружения каждого бита данных, но это основной формат.

CSV-файлы предназначены для простого экспорта данных и их импорта в другие программы. Полученные данные читаются человеком и просматриваются с помощью текстового редактора (рисунок 2.5), такого как Блокнот, или программы для работы с электронными таблицами, такой как Microsoft Excel.[3]

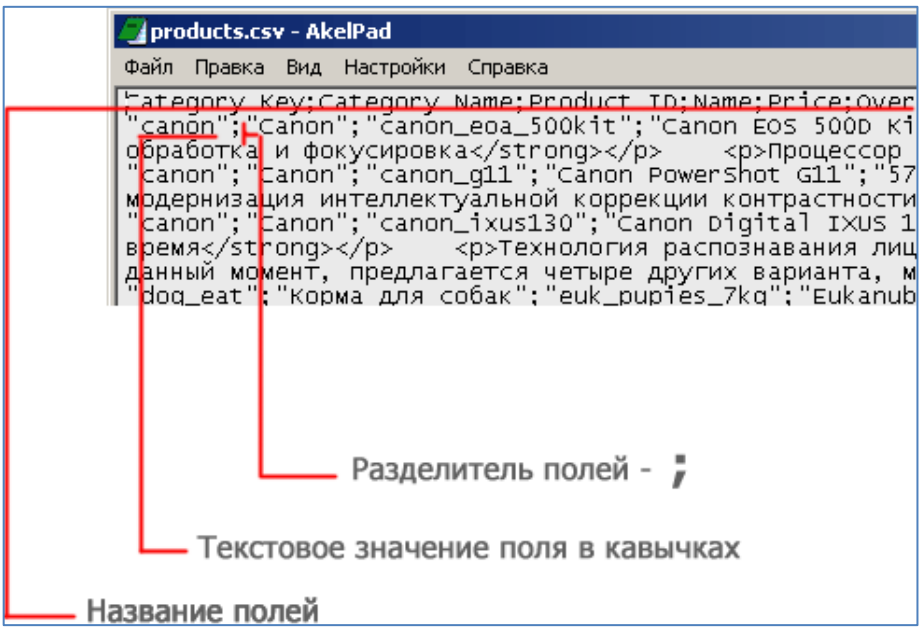


Рисунок 2.5 – Разбор CSV-файл

Особенности CSV файлов:

— CSV-файлы представляют собой текстовые файлы, что облегчает их создание разработчиком веб-сайта;

— поскольку они представляют собой простой текст, их проще импортировать в электронную таблицу или другую базу данных хранения независимо от того, какое программное обеспечение вы используете;

— первая запись в файле может представлять имена следующих столбцов данных и обычно называется заголовками столбцов. Каждая запись в файле с заголовками столбцов может содержать меньше полей, чем количество заголовков столбцов;

— начальные и конечные пробельные символы, запятые и символы табуляции, расположенные рядом с запятыми или разделителями записей, обрезаются;

— символы конца строки, используемые для разделителей записей, иногда заменяются другими символами, такими как точка с запятой;

— некоторые пользователи считают полезным, чтобы анализатор мог игнорировать пустое поле в данных и вместо этого возвращать следующее непустое поле.

2.4 Anaconda: особенности архитектуры и работа с iPython

Anaconda является менеджером пакетов, менеджером среды, дистрибутивом данных Python. Непосредственно с платформы можно быстро разрабатывать и внедрять модели искусственного интеллекта и машинного обучения в мой дипломный проект. Anaconda предоставляет инструменты, необходимые для:

— сбора данных из файлов, баз данных и датасетов;

— управления средами с помощью conda;

— воспроизведения и развертывания проектов в производство одним нажатием кнопки.

Jupyter Notebook и его гибкий интерфейс расширяет возможность текстового редактора от кода до визуализации, мультимедиа, работы дипломного проекта.

При вводе команды ipython выдает интерфейс IPython, работающий в терминале:

```
while True:
    code = input("> ")
    exec(code)
```

Модель похожа на пример кода: запрашивать у пользователя какой-то код, и когда он его введет, выполняет его в том же процессе. Эту модель часто называют Read-Eval-Print-Loop. Остальные интерфейсы – текстовый редактор, консоль Qt в терминале и сторонние интерфейсы - используют ядро IPython. Ядро IPython - отдельный процесс, отвечает за выполнение пользовательского кода и вычисление возможных завершений. Интерфейсы, взаимодействуют с ядром IPython с помощью сообщений JavaScript Object Notation, отправляемых через сокеты ZeroMQ (рисунок 2.6).

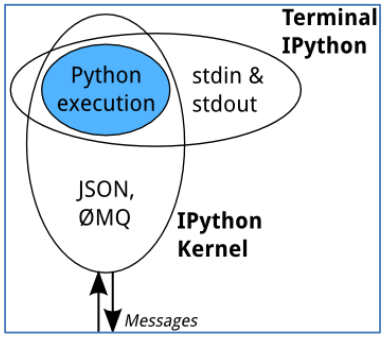


Рисунок 2.6 - Механизм выполнения ядра для совместного использования с терминалом IPython

Процесс ядра может быть подключен к нескольким интерфейсам одновременно. В этом случае разные интерфейсы будут иметь доступ к одним и тем же переменным.

Этот дизайн был предназначен для упрощения разработки различных интерфейсов на основе одного и того же ядра, но он также позволил

поддерживать новые языки в одних и тех же интерфейсах, разрабатывая ядра на этих языках, и IPython для практичности.

Интерфейс Notebook в дополнение к выполнению кода сохраняет код и выводит вместе с примечаниями по уценке в редактируемом документе, называемом блокнотом. При сохранении он отправляется из браузера на сервер редактора, который сохраняет его на диске (рисунок 2.7) в виде файла JavaScript Object Notation с .ipynb расширением.

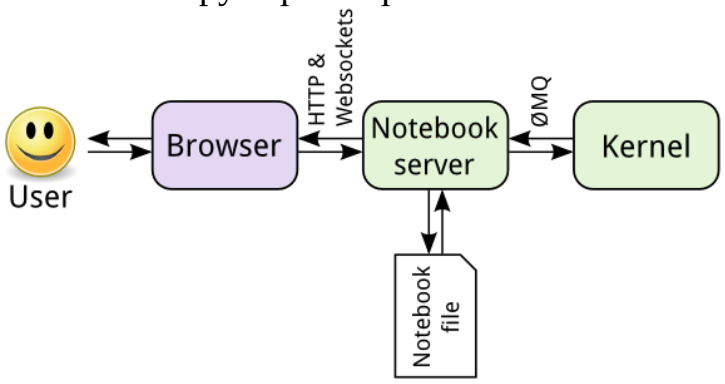


Рисунок 2.7 – Работа Notebook

Сервер записных книжек отвечает за сохранение и загрузку записных книжек, поэтому редактируются записные книжки, даже если нет ядра для этого языка. Ядро ничего не знает о документе ноутбука: оно просто отправляет ячейки кода для выполнения при запуске.

Инструмент Nbconvert в Jupyter преобразует файлы записной книжки в другие форматы, такие как reStructuredText. Это преобразование проходит через ряд шагов (рисунок 2.8):

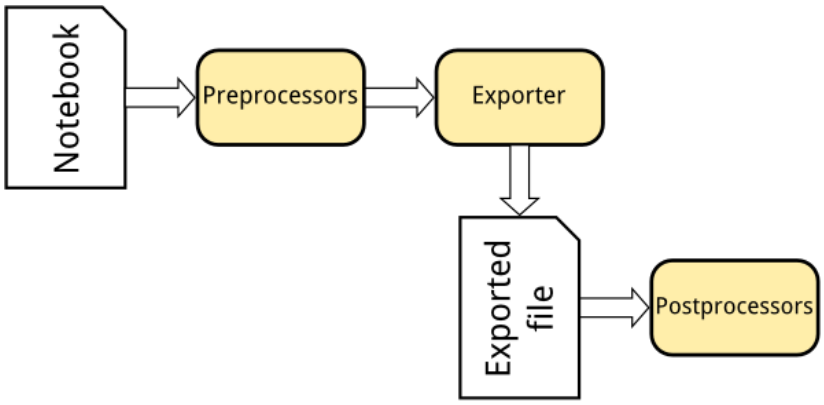


Рисунок 2.8 – Инструмент Nbconvert в Jupyter

— препроцессоры модифицируют текстовый редактор в памяти. Например, executepreprocessor запускает код в записной книжке и обновляет вывод;

— экспортер преобразует записную книжку в другой формат файла, для этого большинство экспортеров используют шаблоны;

— постпроцессоры работают с- файлом, созданным путем экспорта.

IPython также включает в себя инфраструктуру параллельных вычислений IPython.parallel . Это позволяет управлять многими отдельными движками, которые являются расширенной версией ядра IPython.[6]

Выводы

В данной главе был рассмотрен мониторинг вредоносных событий с использованием машинного обучения, сделан обзор доступных решений систем обнаружений, вторжений и аномалий. Изучены решение текущих проблем безопасности данных с помощью машинного обучения. Для проверки текущего состояния науки о данных, проведен анализ алгоритма линейной регрессии. Показана среда разработки Anaconda и её взаимодействие с языком программирования Python. Были объяснены используемые типы наборов данных.

Разъяснена ситуация по нахождению злоумышленников в сети предприятия. Во время разведывательных запросов и движений злоумышленники обычно оставляют некоторые следы. Эти следы присутствуют в данных, и их присутствие может быть обнаружено с помощью науки о данных инструментами машинного обучения для своевременного оповещения. Используя алгоритмы, можно соединить точки и найти шаблоны, которые раньше было трудно найти вручную без аналитиков безопасности.

Непосредственно с платформы Anaconda можно быстро разрабатывать и внедрять модели искусственного интеллекта и машинного обучения в дипломный проект. Anaconda предоставляет инструменты, необходимые для сбора данных в формате CSV и помогает воспроизводить, развертывать проекты одним нажатием кнопки.

Модель взаимодействия Anaconda и Python похожа на пример кода: запрашивается у пользователя определенный код, и когда он его введет, выполняет его в том же процессе. Ядро IPython - отдельный процесс, отвечающий за выполнение пользовательского кода и вычисление возможных завершений. Интерфейсы, взаимодействуют с ядром IPython с помощью сообщений, отправляемых через сокеты ZeroMQ.

3 Практическая часть

3.1 Исследование и подготовка данных первичного запуска в ipynb

3.1.1 Демонстрация Anaconda и создание листа iPynb

Был запущен дистрибутив для работы с данными Python. Его интерфейс с инструментами платформы показан на рисунке 3.1.

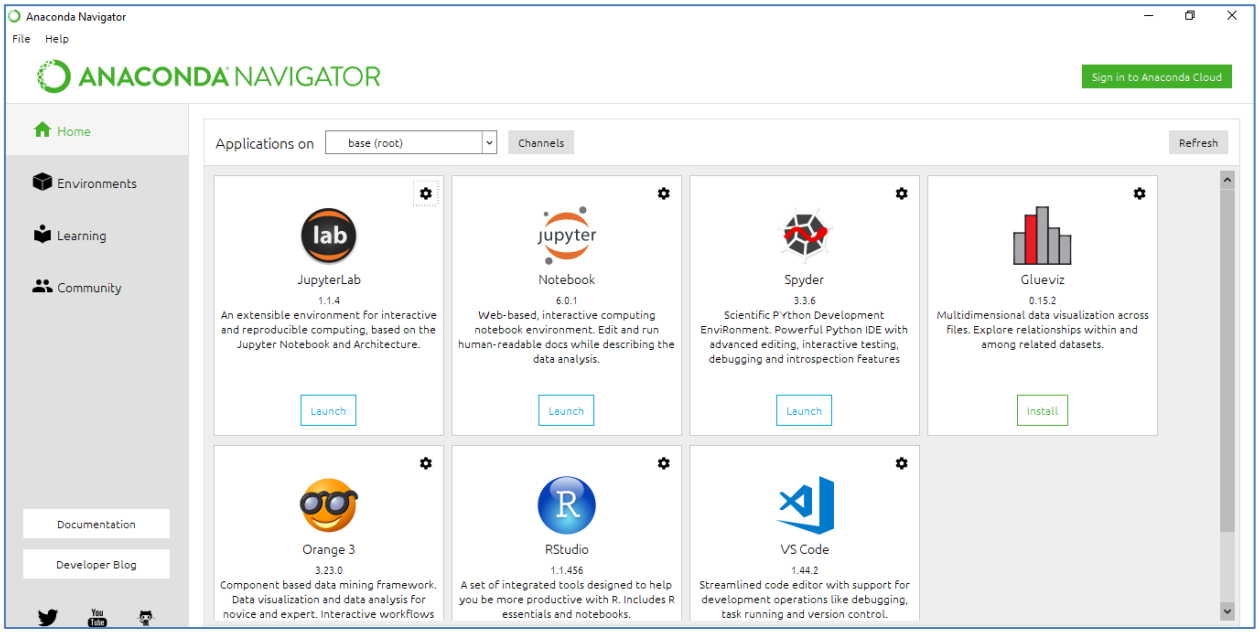


Рисунок 3.1 – Главный интерфейс программы Jupyter

Нужен расширенный текстовый редактор JupyterLab для последующего написания программного кода (рисунок 3.2).

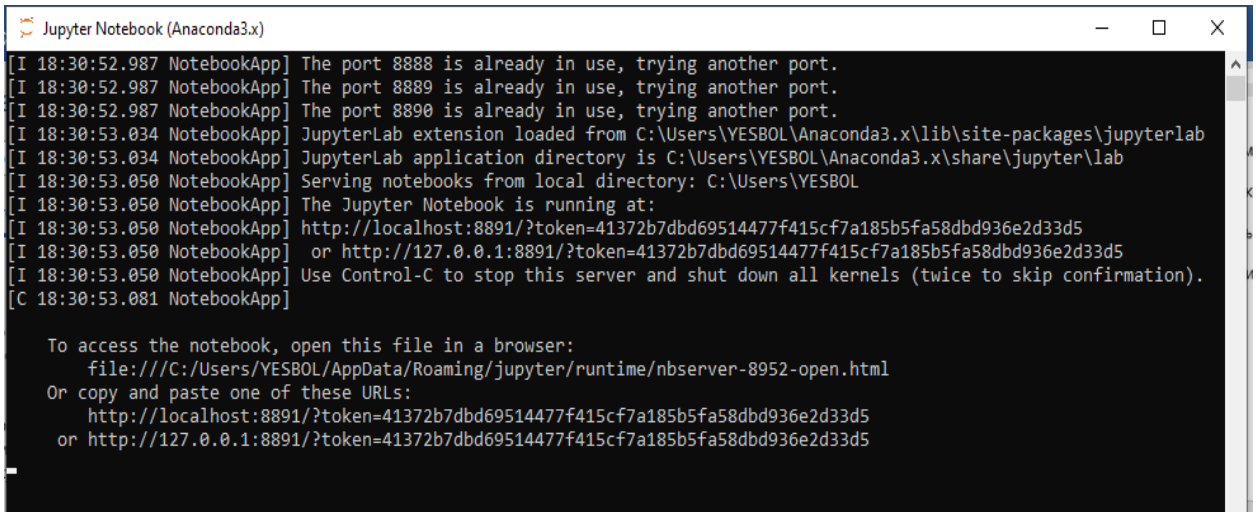


Рисунок 3.2 – Запуск программы через командную строку

Его также можно запустить с меню Пуска, где у редактора есть собственный ярлык (рисунок 3.3).

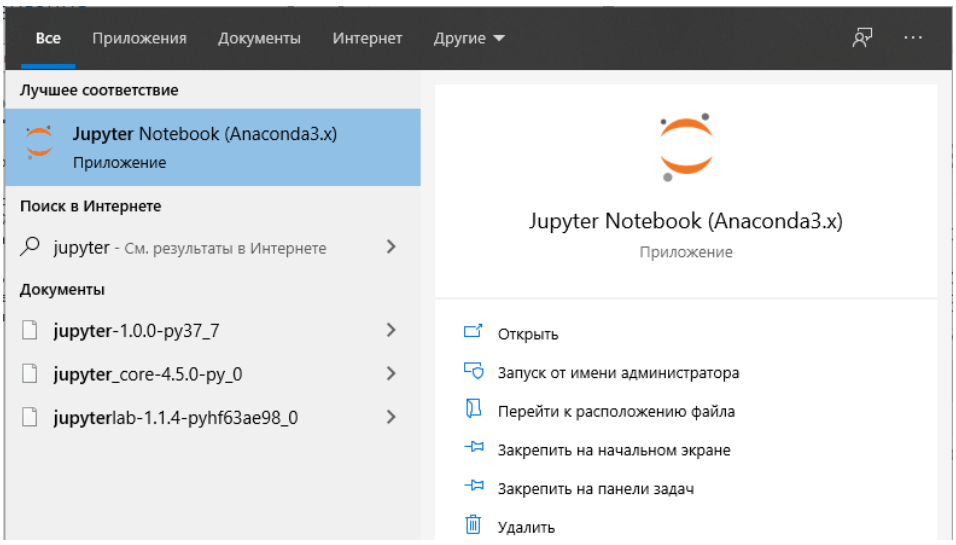


Рисунок 3.3 – Ярлык текстового редактора

В начальной странице Jupyter показывается каталог компьютера (рисунок 3.4).

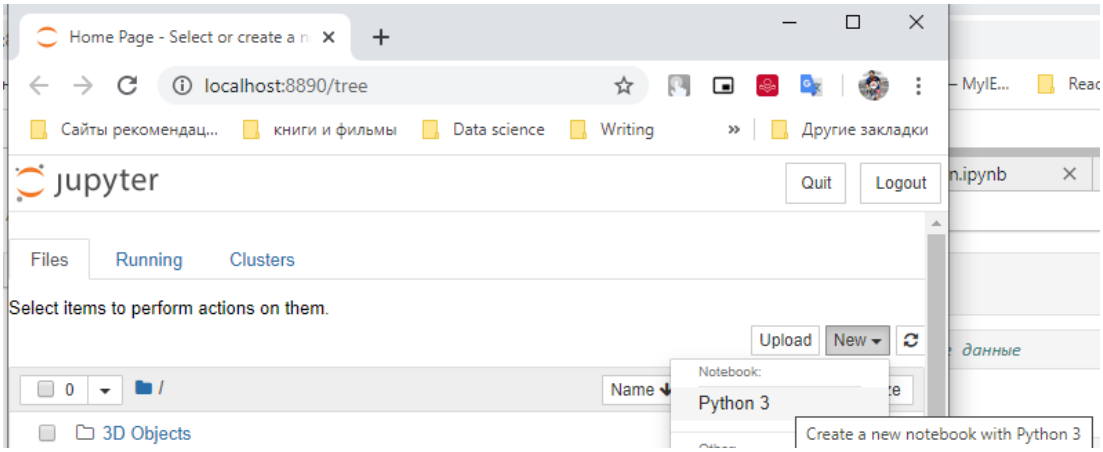


Рисунок 3.4 – Создание нового текстового документа

На рисунке 3.5 представлен новый текстовый документ.

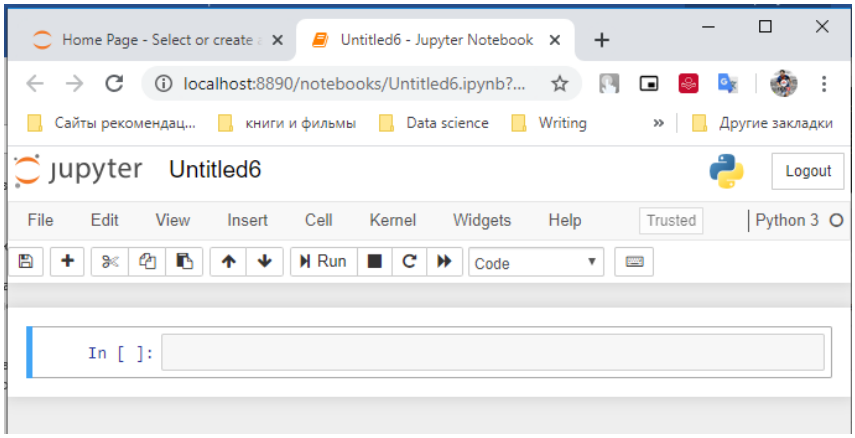


Рисунок 3.5 – Новый текстовый документ

Режимы для программного кода и описание кода переключаются через вкладку Code-Markdown (рисунок 3.6).

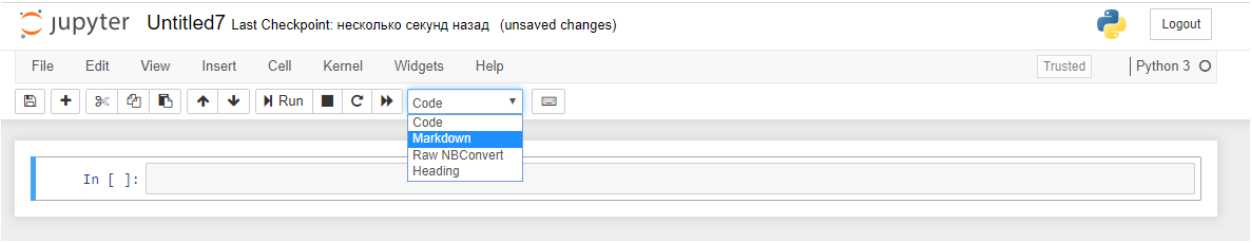


Рисунок 3.6 – Переключение режима ячеек

Разница между ячейками Code и Markdown показана на рисунке 3.7.

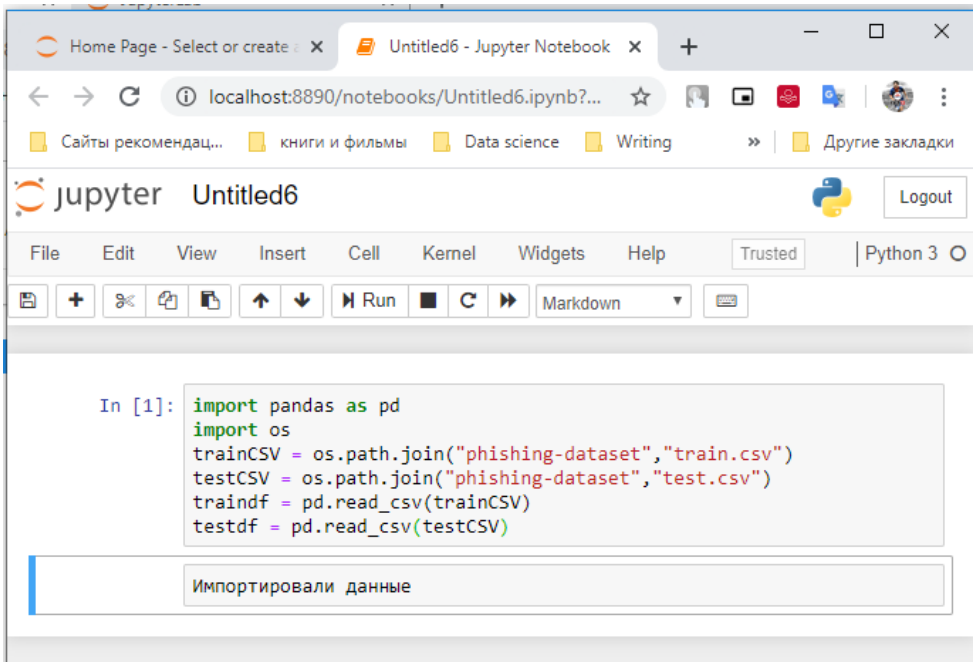


Рисунок 3.7 – Показ разницы ячеек между Code и Markdown

3.1.2 Исследование данных для нахождения алгоритмов обнаружения аномалий

В этой части дипломного проекта исследуется и классифицируется (определяются) файлы Portable Executable как «доброкачественные» или «вредоносные». Основной мотивацией является изучение взаимосвязи IPython, Pandas и scikit-learn с классификацией в качестве средства исследования. На рисунке 3.8 представлен импорт необходимых библиотек.

```
import os
import sklearn.feature_extraction

import pandas as pd

import numpy as np
```

Рисунок 3.8 - Импортирование библиотек

Для PE-файлов нужно сделать переход от необработанных двоичных файлов к DataFrame. Для PE-файлов существует множество отличных инструментов, есть модуль pyfile python. После идет создание функция для записи, загружаю 50 вредосносных и доброкачественных файлов:

f_list = [os.path.join('dataset/bad', child) for child in os.listdir('data/bad')]

threat_features = load_files(file_list)

f_list = [os.path.join('dataset/good', child) for child in os.listdir('data/good')]

features = load_files(file_list)

Осуществляется переход от списка словарей Python к Pandas DataFrame. У Pandas есть разные способы создания фрейма данных (рисунок 3.9).

```
# Ввод функций в пандас dataframe
import pandas as pd
df_bad = pd.DataFrame.from_records(bad_features)
df_bad['label'] = 'bad'
df_good = pd.DataFrame.from_records(good_features)
df_good['label'] = 'good'
df_good.head()
```

	check_sum	compile_date	datadir_IMAGE_DIRECTORY_ENTRY_BASERELOC_size	datadir_IMAGE_DIRECTORY_ENTRY_EXPORT_size
0	97308	1383744221	3044	0
1	103233	1383102953	60	0
2	26573	1386271379	360	0
3	0	1373925025	12	0
4	50003	1378865704	360	0

5 rows × 108 columns

Рисунок 3.9 – Создание фрейма данных

Используется некоторые полезные функции в фрейме данных Pandas для просмотра на обработанные данные (рисунки 3.10-3.11).

df_badd['sum'].hist(,label='bad',bins=30)

df_goodd['sum'].hist(='good',bins=30)

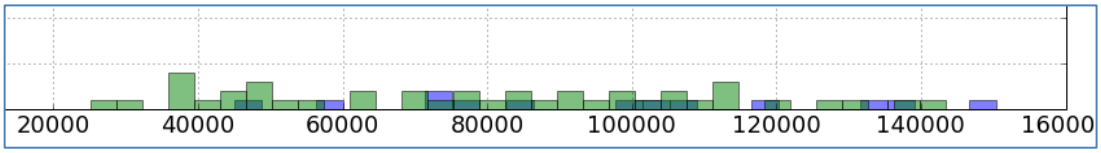


Рисунок 3.10 – Просмотр обработанных данных

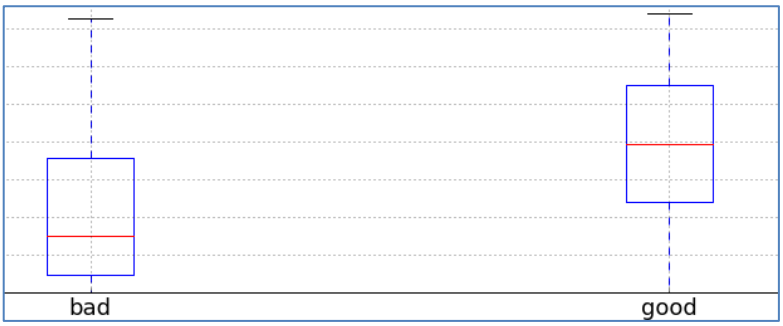


Рисунок 3.11 – Boxplots показывает распределение данных

Разделяется классы для установки цвета, размер, метки:

goodd = df[cond]

badd = df[~cond]

plt.scatter(good['количество импортированных символов], goodd['number_of_sections'], s=180, label='Good', alpha=.3)

plt.scatter(bad['количество импортированных символов'], badd['number_of_sections'], s=60, label='Bad', alpha=.3). Получаю рисунок 3.12.

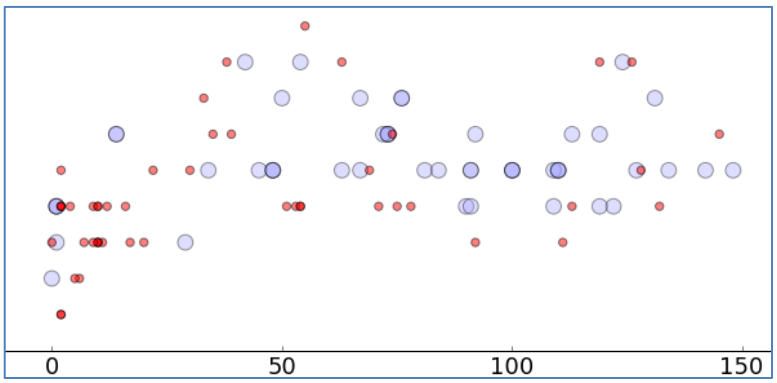


Рисунок 3.12 – Разделение классов

Переход от Pandas DataFrame к X Matrix и любому вектору, согу используется все алгоритмы научного обучения.

X = df.as_matrix(['импортирвоание символов',
'импортирование секторов'])

import sklearn.ensemble

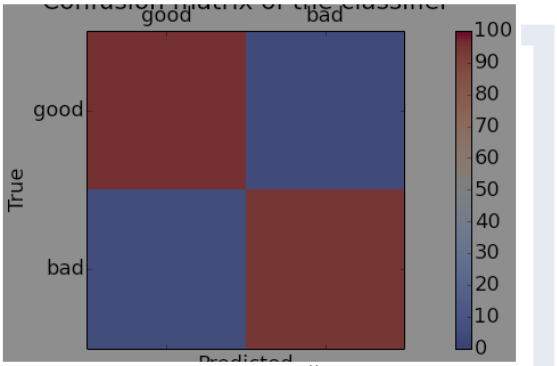
rfc = sklearn.ensemble.RandomForestClassifier(n_estimators=50, compute_importances=True)

Теперь можно использовать перекрестную проверку Scikit Learn для оценки эффективности прогнозирования.

Здесь показан способ, в котором настраиваивается и связывается с вводом функции в алгоритм машинного обучения, либо с методами вероятности предсказания, которые есть во многих классах в scikit-learn (рисунок 3.13). Ввод функции в алгоритм МО:

```
no_label = list(df.columns.values)
no_label.remove('label')
X = df.as_matrix(no_label)
# 60/40 делю в такой пропорции для теста прогнозирования
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=my_tsize, random
_state=my_seed)
clf.fit(X_train, y_train)
y_pred = clf.predict(X_test)
cm = confusion_matrix(y_test, y_pred, labels)
plot_cm(cm, labels)
```

Статистика матрицы ошибок
good/good: 95.45% (21/22)
good/bad: 4.55% (1/22)
bad/good: 5.56% (1/18)
bad/bad: 94.44% (17/18)



Матрица несоответствий

Рисунок 3.13 – Вывод матрицы несоответствий

Вычисляется вероятность предусловий для минимизации ложных сраба тываний (рисунок 3.14):

```
y_probs = clf.predict_proba(X_test)[:,0]
y_pred[y_probs<thres] = 'good'
y_pred[y_probs>=thres] = 'bad'
cm = confusion_matrix(y_test, y_pred, labels)
plot_cm(cm, labels)
```

Статистика матрицы ошибок
goodd/goodd: 100.00% (22/22)
goodd/badd: 0.00% (0/22)
badd/goodd: 16.67% (3/18)
badd/badd: 83.33% (15/18)

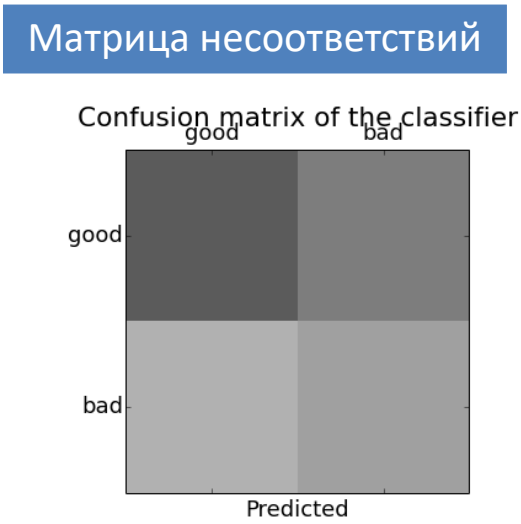


Рисунок 3.14 – Вывод матрицы несоответствий

Комбинация IPython, Pandas и Scikit Learn позволяет извлекать файлы, функции, составлять графики, изучать их с помощью машинного обучения.

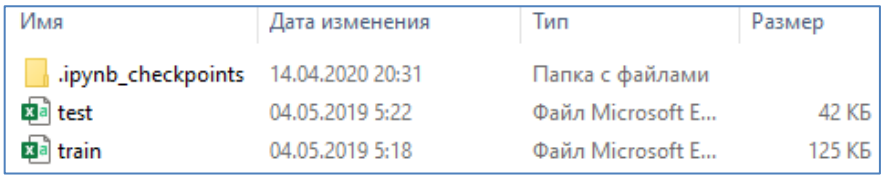
3.1.3 Краткое описание журнала логов, действия пользователей внутри предприятия

Данные по рискам инсайдеров содержит крупнейший общедоступный архив сценариев «красной команды». Симуляция построена на сочетании реальных исследований рисков, связанных с внутренним миром, с фактическими нейтральными клиентами.

Набор данных по рискам (рисунки 3.15-3.21) инсайдеров Института разработки программного обеспечения Университета Карнеги-Меллона представляет месяцы трафика в одной инжиниринговой компании из Интернета, телефона, входа в систему, папок и доступа к системе (dtaa.com). В компании работают несколько тысяч человек, каждый из которых выполняет в среднем 1000 зарегистрированных операций в день.

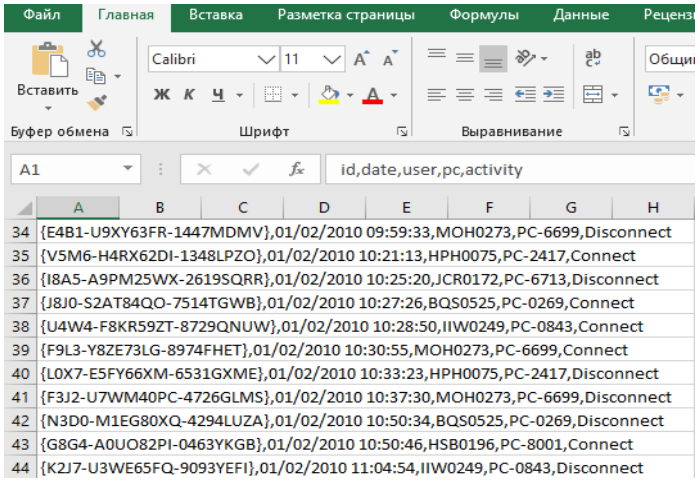
Имя	Дата изменения	Тип	Размер
.ipynb_checkpoints	11.04.2020 16:38	Папка с файлами	
LDAP	11.04.2020 13:53	Папка с файлами	
device	02.02.2012 17:28	Файл Microsoft E...	28 304 КБ
email	02.02.2012 17:35	Файл Microsoft E...	1 330 178 КБ
file	02.02.2012 17:29	Файл Microsoft E...	188 531 КБ
http	02.02.2012 18:20	Файл Microsoft E...	14 195 564 ...
license	11.11.2011 17:08	Текстовый докум...	4 КБ
logon	02.02.2012 17:28	Файл Microsoft E...	57 144 КБ
psychometric	02.02.2012 13:59	Файл Microsoft E...	44 КБ
readme	06.02.2012 16:58	Текстовый докум...	7 КБ

Рисунок 3.15 – Набор данных внутренних угроз



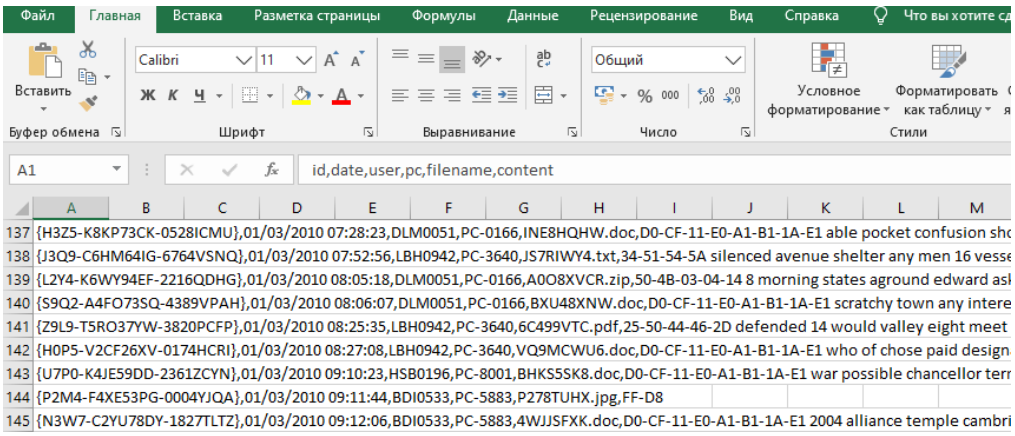
Имя	Дата изменения	Тип	Размер
.ipynb_checkpoints	14.04.2020 20:31	Папка с файлами	
test	04.05.2019 5:22	Файл Microsoft E...	42 КБ
train	04.05.2019 5:18	Файл Microsoft E...	125 КБ

Рисунок 3.16 – Данные для обнаружения фишинга



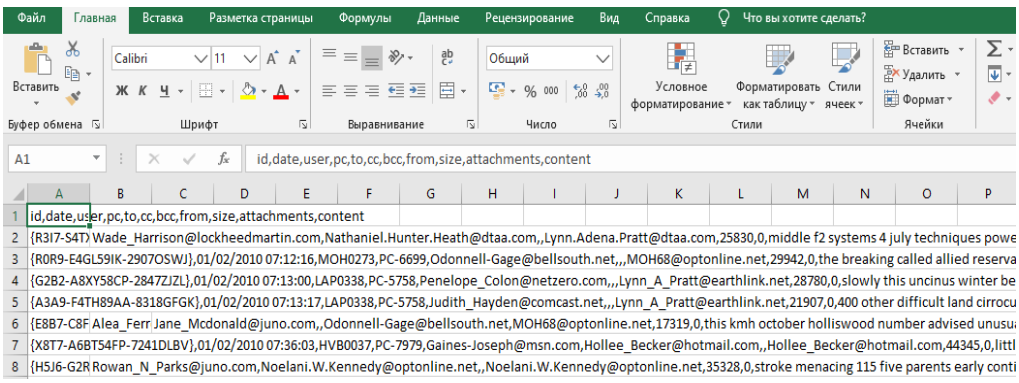
	A
34	{E4B1-U9XY63FR-1447MDMV},01/02/2010 09:59:33,MOH0273,PC-6699,Disconnect
35	{V5M6-H4RX62DI-1348LPZO},01/02/2010 10:21:13,HPH0075,PC-2417,Connect
36	{I8A5-A9PM25WX-2619SQRR},01/02/2010 10:25:20,JCR0172,PC-6713,Disconnect
37	{J8J0-S2AT84QO-7514TGWB},01/02/2010 10:27:26,BQS0525,PC-0269,Connect
38	{U4W4-F8KR59ZT-8729QNUW},01/02/2010 10:28:50,IIW0249,PC-0843,Connect
39	{F9L3-Y8ZE73LG-8974FHET},01/02/2010 10:30:55,MOH0273,PC-6699,Connect
40	{L0X7-E5FY66XM-6531GXME},01/02/2010 10:33:23,HPH0075,PC-2417,Disconnect
41	{F3J2-U7WM40PC-4726GLMS},01/02/2010 10:37:30,MOH0273,PC-6699,Disconnect
42	{N3D0-M1EG80XQ-4294LUZA},01/02/2010 10:50:34,BQS0525,PC-0269,Disconnect
43	{G8G4-A0UO82PI-0463YKGB},01/02/2010 10:50:46,HSB0196,PC-8001,Connect
44	{K2J7-U3WE65FQ-9093YEFI},01/02/2010 11:04:54,IIW0249,PC-0843,Disconnect

Рисунок 3.17 – Данные об использовании устройств



	A
137	{H3Z5-K8KP73CK-0528ICMU},01/03/2010 07:28:23,DLM0051,PC-0166,INE8HQHW.doc,D0-CF-11-E0-A1-B1-1A-E1 able pocket confusion sho
138	{J3Q9-C6HM64IG-6764VSNQ},01/03/2010 07:52:56,LBH0942,PC-3640,JS7RIWY4.txt,34-51-54-5A silenced avenue shelter any men 16 vesse
139	{L2Y4-K6WY94EF-2216QDHG},01/03/2010 08:05:18,DLM0051,PC-0166,A0O8XVCR.zip,50-4B-03-04-14 8 morning states aground edward ask
140	{S9Q2-A4FO73SQ-4389VPAH},01/03/2010 08:06:07,DLM0051,PC-0166,BXU48XNW.doc,D0-CF-11-E0-A1-B1-1A-E1 scratchy town any intere
141	{Z9L9-T5RO37YW-3820PCFP},01/03/2010 08:25:35,LBH0942,PC-3640,6C499VTC.pdf,25-50-44-46-2D defended 14 would valley eight meet
142	{H0P5-V2CF26XV-0174HCRI},01/03/2010 08:27:08,LBH0942,PC-3640,VQ9MCWU6.doc,D0-CF-11-E0-A1-B1-1A-E1 who of chose paid design
143	{U7P0-K4JE59DD-2361ZCYN},01/03/2010 09:10:23,HSB0196,PC-8001,BHKS5SK8.doc,D0-CF-11-E0-A1-B1-1A-E1 war possible chancellor terr
144	{P2M4-F4XE53PG-0004YJQA},01/03/2010 09:11:44,BDI0533,PC-5883,P278TUHX.jpg,FF-D8
145	{N3W7-C2YU78DY-1827TLTZ},01/03/2010 09:12:06,BDI0533,PC-5883,4WJJSFXK.doc,D0-CF-11-E0-A1-B1-1A-E1 2004 alliance temple cambri

Рисунок 3.18 – Данные о передаче файлов



	A
1	id,date,user,pc,to,cc,bcc,from,size,attachments,content
2	{R3I7-S4T} Wade_Harrison@lockheedmartin.com,Nathaniel.Hunter.Heath@dtaa.com,,Lynn.Adena.Pratt@dtaa.com,25830,0,middle f2 systems 4 july techniques powe
3	{R0R9-E4GL59IK-2907OSWJ},01/02/2010 07:12:16,MOH0273,PC-6699,Odonnell-Gage@bellsouth.net,,,MOH68@optonline.net,29942,0,the breaking called allied reserva
4	{G2B2-A8XY58CP-2847ZJZL},01/02/2010 07:13:00,LAP0338,PC-5758,Penelope_Colon@netzero.com,,,Lynn_A_Pratt@earthlink.net,28780,0,slowly this uncinus winter be
5	{A3A9-F4TH89AA-8318GFGK},01/02/2010 07:13:17,LAP0338,PC-5758,Judith_Hayden@comcast.net,,,Lynn_A_Pratt@earthlink.net,21907,0,400 other difficult land cirrocu
6	{E8B7-C8F Alea_Ferr Jane_Mcdonald@juno.com,,Odonnell-Gage@bellsouth.net,MOH68@optonline.net,17319,0,this kmh october holliswood number advised unusu
7	{X8T7-A6BT54FP-7241DLBV},01/02/2010 07:36:03,HVB0037,PC-7979,Gaines-Joseph@msn.com,Hollee_Becker@hotmail.com,,Hollee_Becker@hotmail.com,44345,0,littl
8	{H5J6-G2R Rowan_N_Parks@juno.com,Noelani.W.Kennedy@optonline.net,,Noelani.W.Kennedy@optonline.net,35328,0,stroke menacing 115 five parents early conti

Рисунок 3.19 – Данные о передаче сообщении через email

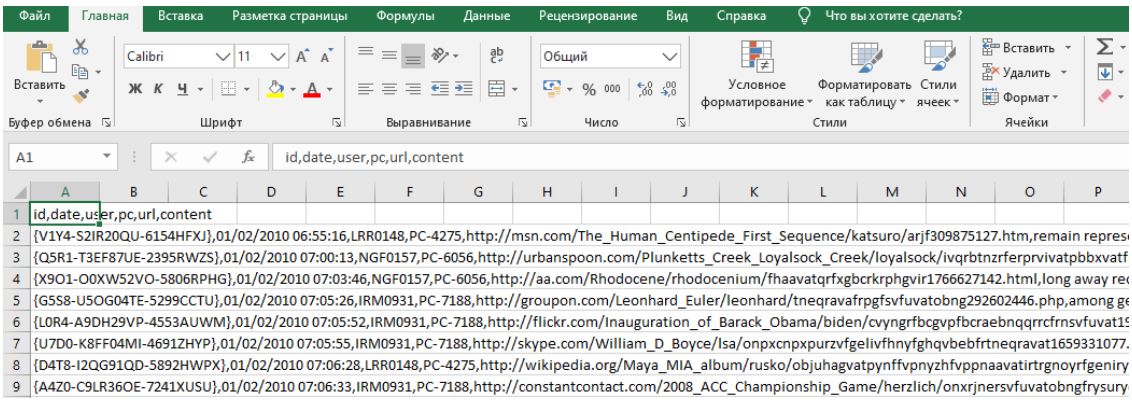


Рисунок 3.20 – Данные о http

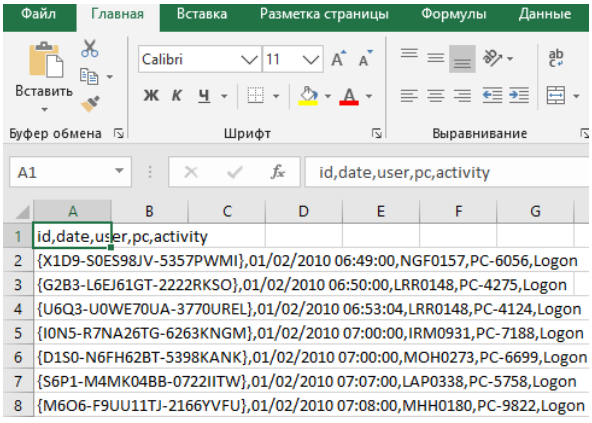
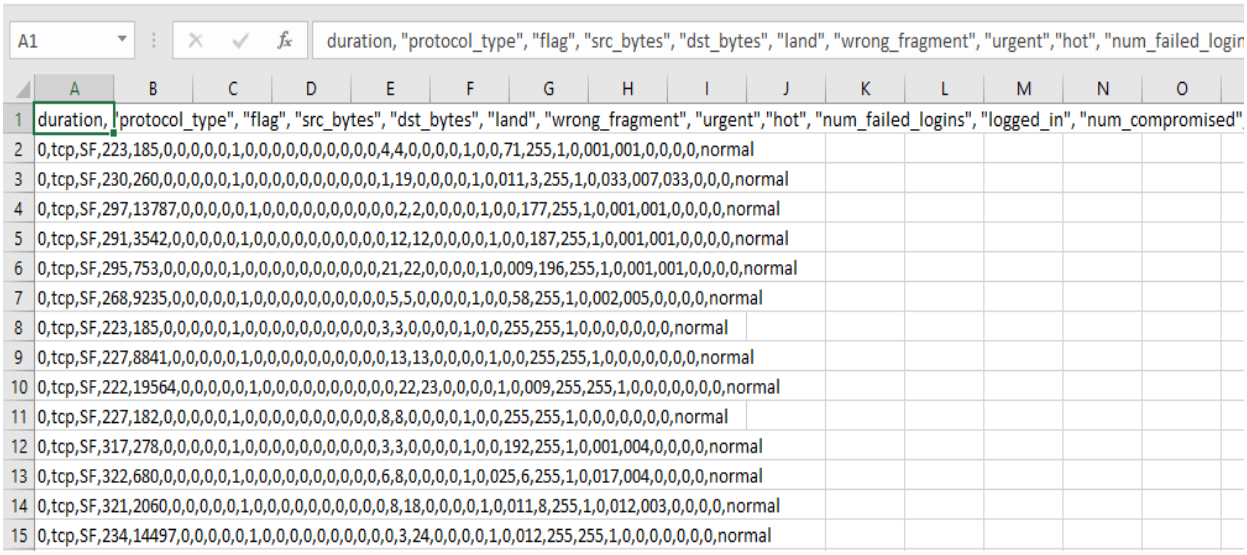


Рисунок 3.21 – Данные о входах/выходах по времени

Обнаружение аномалий поведения сети - это постоянный мониторинг сети на наличие необычных событий или тенденций. В идеале программа отслеживает критические характеристики сети в режиме реального времени и генерирует сигнал тревоги, если обнаруживается странное событие или тенденция, указывающая на угрозу.

Этот набор данных (Информационно-компьютерный университет Калифорнии) содержит стандартный набор данных для аудита, который включает в себя широкий спектр вторжений, моделируемых в среде военной сети. Он состоит из 41237 записей http-соединений за семь недель сетевого трафика. Соединение - это последовательность пакетов, начинающихся и заканчивающихся в некоторые четко определенные моменты времени, между которыми потоки данных направляются от исходного IP-адреса к отправленному IP-адресу и от него по некоторому четко определенному протоколу (рисунок 3.22).

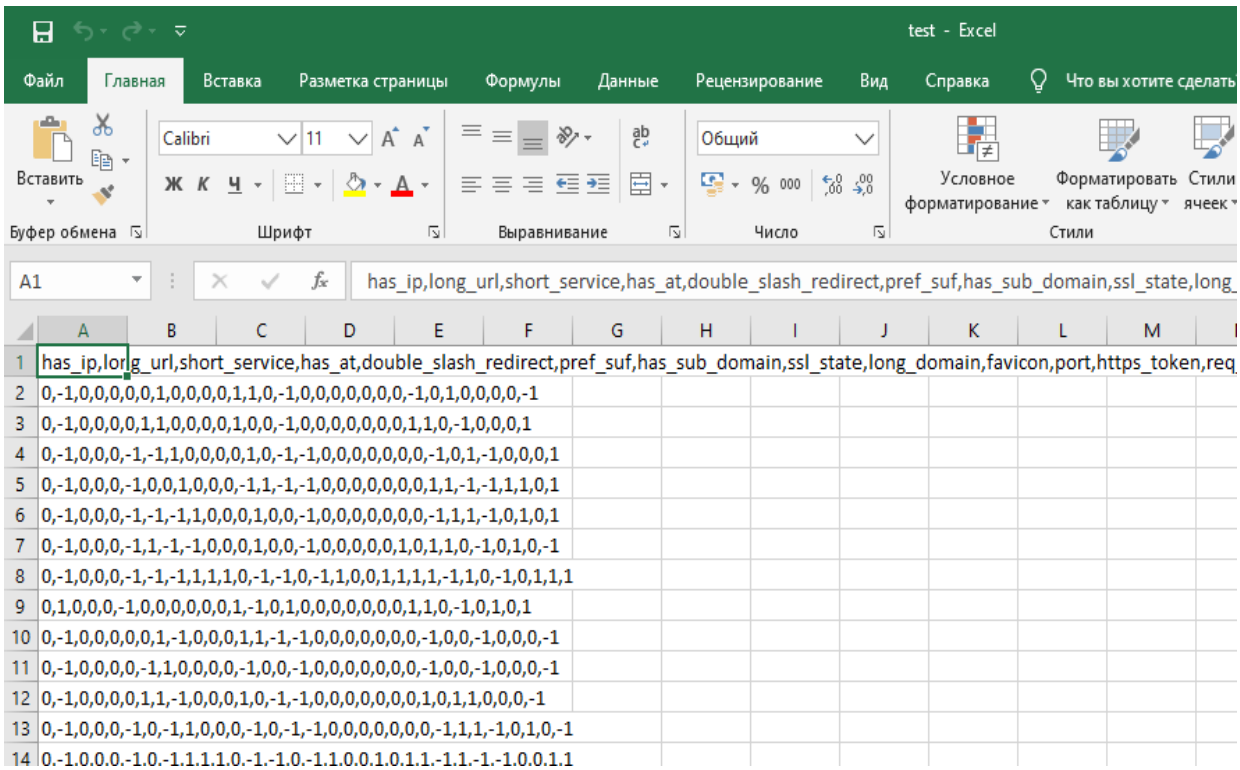
Набор данных содержит 41 функцию отдельных соединений TCP, функции контента и функции трафика. Каждое соединение помечается как нормальное или как атака с точно определенным типом атаки.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	duration,	protocol_type", "flag", "src_bytes", "dst_bytes", "land", "wrong_fragment", "urgent","hot", "num_failed_logins", "logged_in", "num_compromised",													
2	0,tcp,SF,223,185,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,4,4,0,0,0,0,1,0,0,71,255,1,0,001,001,0,0,0,0,normal														
3	0,tcp,SF,230,260,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,19,0,0,0,0,1,0,011,3,255,1,0,033,007,033,0,0,0,normal														
4	0,tcp,SF,297,13787,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,2,2,0,0,0,0,1,0,0,177,255,1,0,001,001,0,0,0,0,normal														
5	0,tcp,SF,291,3542,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,12,12,0,0,0,0,1,0,0,187,255,1,0,001,001,0,0,0,0,normal														
6	0,tcp,SF,295,753,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,21,22,0,0,0,0,1,0,009,196,255,1,0,001,001,0,0,0,0,normal														
7	0,tcp,SF,268,9235,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,5,5,0,0,0,0,1,0,0,58,255,1,0,002,005,0,0,0,0,normal														
8	0,tcp,SF,223,185,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,3,3,0,0,0,0,1,0,0,255,255,1,0,0,0,0,0,0,0,normal														
9	0,tcp,SF,227,8841,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,13,13,0,0,0,0,1,0,0,255,255,1,0,0,0,0,0,0,0,normal														
10	0,tcp,SF,222,19564,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,22,23,0,0,0,0,1,0,009,255,255,1,0,0,0,0,0,0,0,0,normal														
11	0,tcp,SF,227,182,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,8,8,0,0,0,0,1,0,0,255,255,1,0,0,0,0,0,0,0,normal														
12	0,tcp,SF,317,278,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,3,3,0,0,0,0,1,0,0,192,255,1,0,001,004,0,0,0,0,normal														
13	0,tcp,SF,322,680,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,6,8,0,0,0,0,1,0,025,6,255,1,0,017,004,0,0,0,0,normal														
14	0,tcp,SF,321,2060,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,8,18,0,0,0,0,1,0,011,8,255,1,0,012,003,0,0,0,0,normal														
15	0,tcp,SF,234,14497,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,3,24,0,0,0,0,1,0,012,255,255,1,0,0,0,0,0,0,0,normal														

Рисунок 3.22 – Данные для Internet Behavior Traffic

Веб-сайт с фишингом, который пытается получить пароль учетной записи или другую личную информацию, заставляя вас думать, что вы находитесь на законном веб-сайте. Некоторые фишинговые адреса отличаются от предполагаемого адреса одним символом, специально выбранным для увеличения вероятности опечатки, в то время как другие используют другие каналы для генерации трафика. Наборы данных были загружены из каталога с CIC IDS (2018), что проиллюстрировано на рисунках 3.23-3.24.



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	has_ip,long_url,short_service,has_at,double_slash_redirect,pref_suf,has_sub_domain,ssl_state,long_domain,favicon,port,https_token,req												
2	0,-1,0,0,0,0,0,1,0,0,0,0,1,1,0,-1,0,0,0,0,0,0,0,0,-1,0,1,0,0,0,0,-1												
3	0,-1,0,0,0,0,1,1,0,0,0,0,1,0,0,-1,0,0,0,0,0,0,0,0,1,1,0,-1,0,0,0,1												
4	0,-1,0,0,0,-1,-1,1,0,0,0,0,1,0,-1,-1,0,0,0,0,0,0,0,0,-1,0,1,-1,0,0,0,1												
5	0,-1,0,0,0,-1,0,0,1,0,0,0,-1,1,-1,-1,0,0,0,0,0,0,0,0,1,1,-1,-1,1,1,0,1												
6	0,-1,0,0,0,-1,-1,-1,1,0,0,0,1,0,0,-1,0,0,0,0,0,0,0,0,-1,1,1,-1,0,1,0,1												
7	0,-1,0,0,0,-1,1,-1,-1,0,0,0,1,0,0,-1,0,0,0,0,0,0,1,0,1,1,0,-1,0,1,0,-1												
8	0,-1,0,0,0,-1,-1,-1,1,1,1,0,-1,-1,0,-1,1,0,0,1,1,1,1,-1,1,0,-1,0,1,1,1												
9	0,1,0,0,0,-1,0,0,0,0,0,0,1,-1,0,1,0,0,0,0,0,0,0,0,1,1,0,-1,0,1,0,1												
10	0,-1,0,0,0,0,0,1,-1,0,0,0,1,1,-1,-1,0,0,0,0,0,0,0,0,-1,0,0,-1,0,0,0,-1												
11	0,-1,0,0,0,0,-1,1,0,0,0,0,-1,0,0,-1,0,0,0,0,0,0,0,0,-1,0,0,-1,0,0,0,-1												
12	0,-1,0,0,0,0,1,1,-1,0,0,0,1,0,-1,-1,0,0,0,0,0,0,0,0,1,0,1,1,0,0,0,-1												
13	0,-1,0,0,0,-1,0,-1,1,0,0,0,-1,0,-1,-1,0,0,0,0,0,0,0,0,0,-1,1,1,-1,0,1,0,-1												
14	0,-1,0,0,0,-1,0,-1,1,1,1,0,-1,-1,0,-1,1,0,0,0,1,0,1,1,-1,1,-1,-1,0,0,1,1												

Рисунок 3.23 – Данные для тестирования

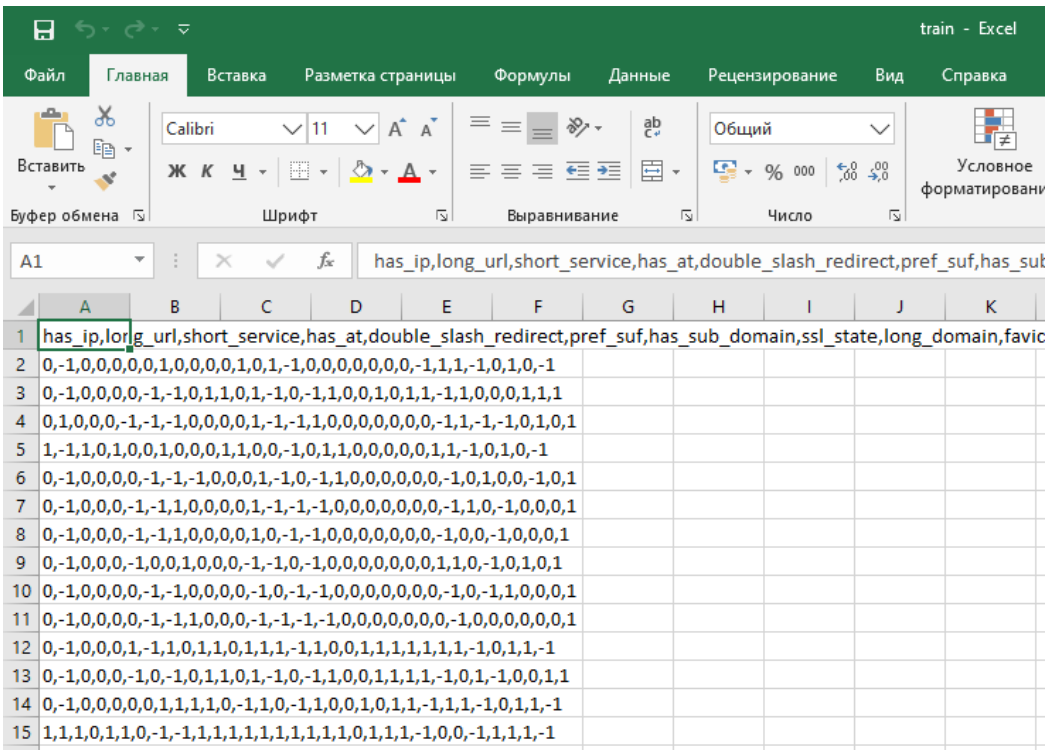
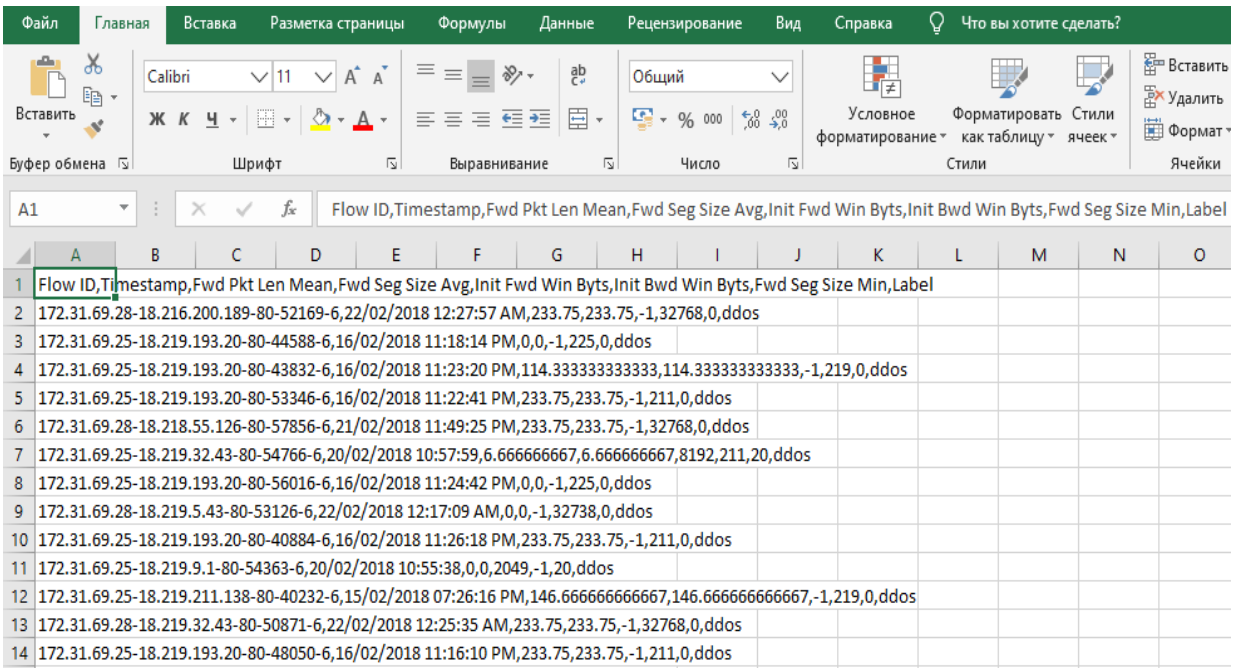


Рисунок 3.24 – Данные для обучения

DDoS или отказ в обслуживании - это атака, при которой трафик из разных источников затопляет жертву, что приводит к прерыванию обслуживания. Существует много типов DDoS-атак, подпадающих под три основные категории: атаки на уровне приложений, протоколы и объемные атаки. Большая часть защиты от DDoS сегодня - ручная. Определенные IP-адреса или домены идентифицируются, а затем блокируются. Поскольку DDoS-боты становятся все более изощренными, такие подходы становятся устаревшими. Машинное обучение предлагает автоматизированное решение.

Поскольку основной предпосылкой DDoS-атак с малым объемом является их способность воздействовать на службу без значительных ресурсов на стороне злоумышленника, атаки были произведены с достаточным трафиком для воздействия на целевую службу. Атаки были остановлены, когда сервер перестал отвечать на запросы.

Произведены 4 типа атак с использованием различных инструментов, получив 8 различных трассировок DDoS-атак на уровне приложений. Для каждого дня были записаны необработанные данные, включая сетевой трафик и журналы событий (рисунки 3.25-3.26).



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Flow ID,Timestamp,Fwd Pkt Len Mean,Fwd Seg Size Avg,Init Fwd Win Byts,Init Bwd Win Byts,Fwd Seg Size Min,Label														
2	172.31.69.28-18.216.200.189-80-52169-6,22/02/2018 12:27:57 AM,233.75,233.75,-1,32768,0,ddos														
3	172.31.69.25-18.219.193.20-80-44588-6,16/02/2018 11:18:14 PM,0,0,-1,225,0,ddos														
4	172.31.69.25-18.219.193.20-80-43832-6,16/02/2018 11:23:20 PM,114.333333333333,114.333333333333,-1,219,0,ddos														
5	172.31.69.25-18.219.193.20-80-53346-6,16/02/2018 11:22:41 PM,233.75,233.75,-1,211,0,ddos														
6	172.31.69.28-18.218.55.126-80-57856-6,21/02/2018 11:49:25 PM,233.75,233.75,-1,32768,0,ddos														
7	172.31.69.25-18.219.32.43-80-54766-6,20/02/2018 10:57:59,6.666666667,6.666666667,8192,211,20,ddos														
8	172.31.69.25-18.219.193.20-80-56016-6,16/02/2018 11:24:42 PM,0,0,-1,225,0,ddos														
9	172.31.69.28-18.219.5.43-80-53126-6,22/02/2018 12:17:09 AM,0,0,-1,32738,0,ddos														
10	172.31.69.25-18.219.193.20-80-40884-6,16/02/2018 11:26:18 PM,233.75,233.75,-1,211,0,ddos														
11	172.31.69.25-18.219.9.1-80-54363-6,20/02/2018 10:55:38,0,0,2049,-1,20,ddos														
12	172.31.69.25-18.219.211.138-80-40232-6,15/02/2018 07:26:16 PM,146.666666666667,146.666666666667,-1,219,0,ddos														
13	172.31.69.28-18.219.32.43-80-50871-6,22/02/2018 12:25:35 AM,233.75,233.75,-1,32768,0,ddos														
14	172.31.69.25-18.219.193.20-80-48050-6,16/02/2018 11:16:10 PM,233.75,233.75,-1,211,0,ddos														

Рисунок 3.25 – Набор данных для определения DDOS

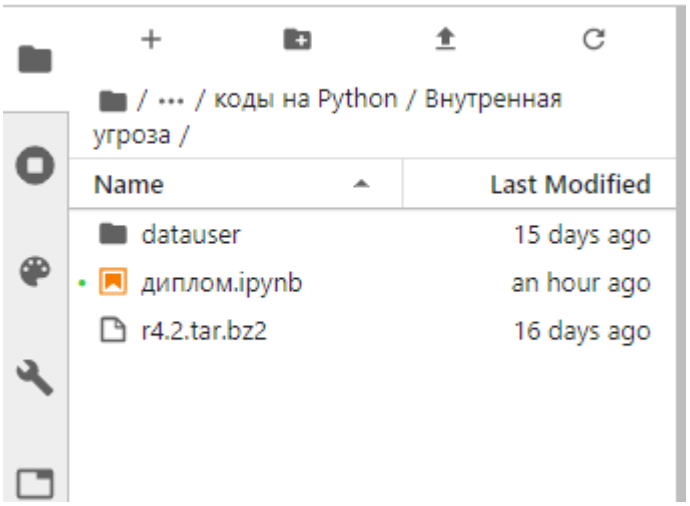
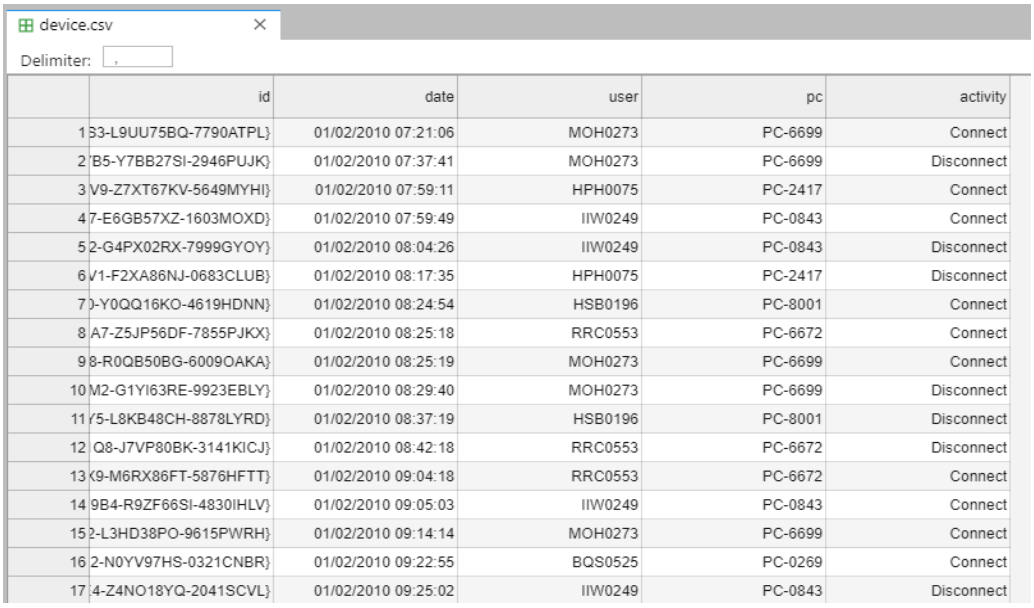


Рисунок 3.26 – Просмотр директорий через Jupyter

По сравнению с Excel данные CSV формата в Jupyter более структурированы (рисунки 3.27-3.28).

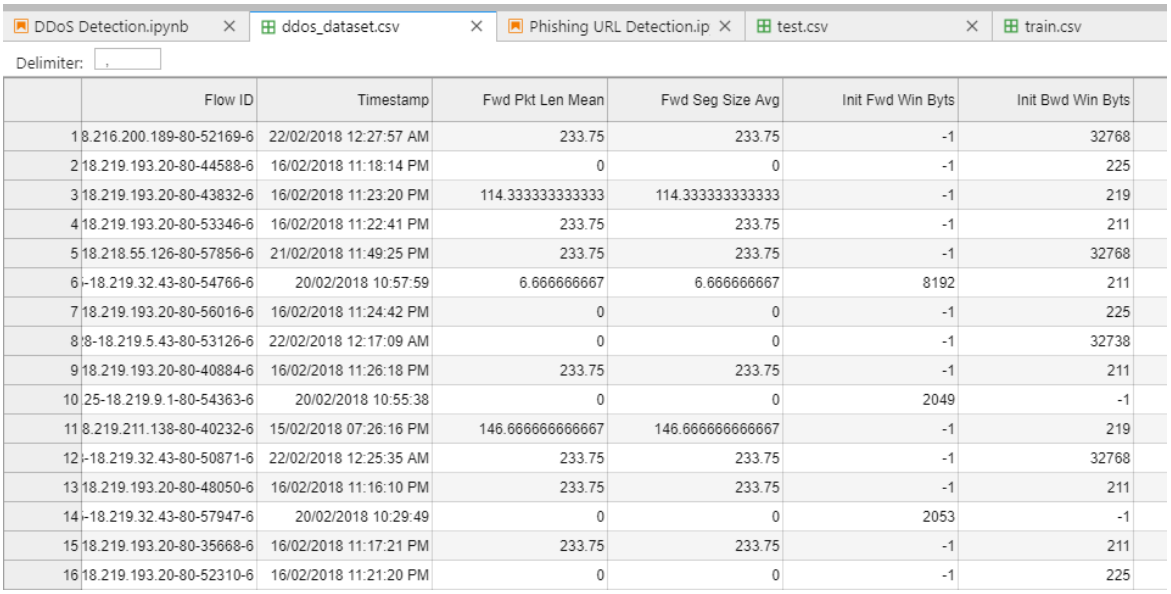


device.csv

Delimiter: ,

	id	date	user	pc	activity
1	33-L9UU75BQ-7790ATPL}	01/02/2010 07:21:06	MOH0273	PC-6699	Connect
2	B5-Y7BB27SI-2946PUJK}	01/02/2010 07:37:41	MOH0273	PC-6699	Disconnect
3	V9-Z7XT67KV-5649MYHI}	01/02/2010 07:59:11	HPH0075	PC-2417	Connect
4	7-E6GB57XZ-1603MOXD}	01/02/2010 07:59:49	IIW0249	PC-0843	Connect
5	2-G4PX02RX-7999GYOY}	01/02/2010 08:04:26	IIW0249	PC-0843	Disconnect
6	V1-F2XA86NJ-0683CLUB}	01/02/2010 08:17:35	HPH0075	PC-2417	Disconnect
7	)-Y0QQ16KO-4619HDNN}	01/02/2010 08:24:54	HSB0196	PC-8001	Connect
8	A7-Z5JP56DF-7855PJKX}	01/02/2010 08:25:18	RRC0553	PC-6672	Connect
9	8-R0QB50BG-6009OAKA}	01/02/2010 08:25:19	MOH0273	PC-6699	Connect
10	M2-G1YI63RE-9923EBLY}	01/02/2010 08:29:40	MOH0273	PC-6699	Disconnect
11	Y5-L8KB48CH-8878LYRD}	01/02/2010 08:37:19	HSB0196	PC-8001	Disconnect
12	Q8-J7VP80BK-3141KICJ}	01/02/2010 08:42:18	RRC0553	PC-6672	Disconnect
13	K9-M6RX86FT-5876HFTT}	01/02/2010 09:04:18	RRC0553	PC-6672	Connect
14	9B4-R9ZF66SI-4830IHLV}	01/02/2010 09:05:03	IIW0249	PC-0843	Connect
15	2-L3HD38PO-9615PWRH}	01/02/2010 09:14:14	MOH0273	PC-6699	Connect
16	2-N0YV97HS-0321CNBR}	01/02/2010 09:22:55	BQS0525	PC-0269	Connect
17	4-Z4NO18YQ-2041SCVL}	01/02/2010 09:25:02	IIW0249	PC-0843	Disconnect

Рисунок 3.27 – Просмотр данных через Jupyter



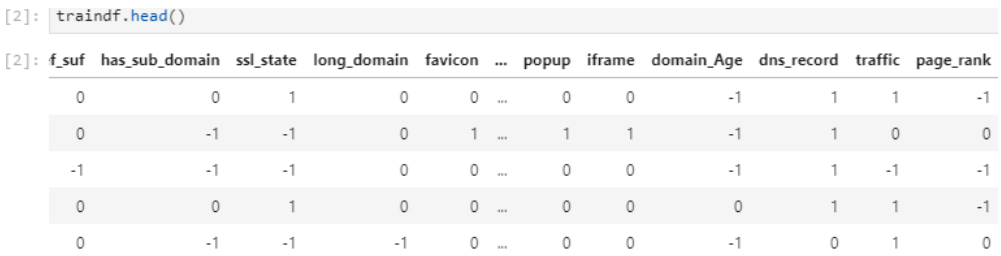
DDoS Detection.ipynb | ddos_dataset.csv | Phishing URL Detection.ip | test.csv | train.csv

Delimiter: ,

	Flow ID	Timestamp	Fwd Pkt Len Mean	Fwd Seg Size Avg	Init Fwd Win Byts	Init Bwd Win Byts
1	8.216.200.189-80-52169-6	22/02/2018 12:27:57 AM	233.75	233.75	-1	32768
2	18.219.193.20-80-44588-6	16/02/2018 11:18:14 PM	0	0	-1	225
3	18.219.193.20-80-43832-6	16/02/2018 11:23:20 PM	114.333333333333	114.333333333333	-1	219
4	18.219.193.20-80-53346-6	16/02/2018 11:22:41 PM	233.75	233.75	-1	211
5	18.218.55.126-80-57856-6	21/02/2018 11:49:25 PM	233.75	233.75	-1	32768
6	-18.219.32.43-80-54766-6	20/02/2018 10:57:59	6.666666667	6.666666667	8192	211
7	18.219.193.20-80-56016-6	16/02/2018 11:24:42 PM	0	0	-1	225
8	8-18.219.5.43-80-53126-6	22/02/2018 12:17:09 AM	0	0	-1	32738
9	18.219.193.20-80-40884-6	16/02/2018 11:26:18 PM	233.75	233.75	-1	211
10	25-18.219.9.1-80-54363-6	20/02/2018 10:55:38	0	0	2049	-1
11	8.219.211.138-80-40232-6	15/02/2018 07:26:16 PM	146.666666666667	146.666666666667	-1	219
12	-18.219.32.43-80-50871-6	22/02/2018 12:25:35 AM	233.75	233.75	-1	32768
13	18.219.193.20-80-48050-6	16/02/2018 11:16:10 PM	233.75	233.75	-1	211
14	-18.219.32.43-80-57947-6	20/02/2018 10:29:49	0	0	2053	-1
15	18.219.193.20-80-35668-6	16/02/2018 11:17:21 PM	233.75	233.75	-1	211
16	18.219.193.20-80-52310-6	16/02/2018 11:21:20 PM	0	0	-1	225

Рисунок 3.28 – Просмотр данных через Jupyter

Просмотр первых 5 строк данных осуществляется функцией head (рисунок 3.29).



```
[2]: traindf.head()
```

[2]:

f_suf	has_sub_domain	ssl_state	long_domain	favicon	...	popup	iframe	domain_Age	dns_record	traffic	page_rank
0	0	1	0	0	...	0	0	-1	1	1	-1
0	-1	-1	0	1	...	1	1	-1	1	0	0
-1	-1	-1	0	0	...	0	0	-1	1	-1	-1
0	0	1	0	0	...	0	0	0	1	1	-1
0	-1	-1	-1	0	...	0	0	-1	0	1	0

Рисунок 3.29 – Функция head

3.2 Формирование признаков адаптирования данных под алгоритм машинного обучения

3.2.1 Импорт библиотеки Pandas и данных из CSV-файла

В 4 вариантах программных кодов анализируются разные подходы и классификаторы. Но алгоритм действий машинного обучения одинаков в каждом случае. В первую очередь это сбор и загрузка данных из csv файлов, представляющие себя из табличные данные. Импортируется библиотека pandas и numpy (рисунок 3.30). Далее загружается данные для обнаружения фишинга из каталога, прочитал данные для обучения и тестирования, используя pandas. Полный листинг кода в Приложении А.

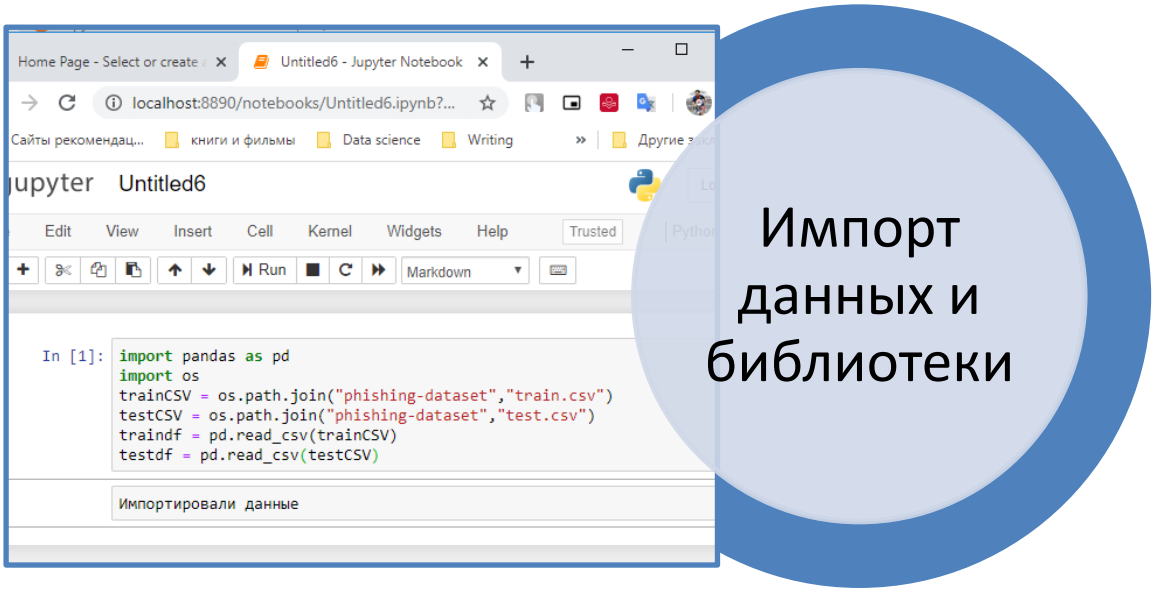


Рисунок 3.30 - Импорт данных и библиотеки

Данные представляют собой набор с плотной иглой, что означает более высокую частоту внутренних угроз, чем другие наборы данных:

```
import numpy as np
import pandas as pd
In []:
path_to_dataset = "./datauser/"      #путь где находится загруженные данные
```

Далее берутся столбцы сетевого трафика с угрозами, чтобы в конце практической части убедиться в верности расчетов и после считывания типов угроз трафика (рисунок 3.31-3.32).

```
traindf.head()
```

Рисунок 3.31 – Показ данных

	has_ip	long_url	short_service	has_at	double_slash_redirect	pref_suf	has_sub_domain	ssl_state	long_domain	favicon	...	popup	iframe
0	0	-1	0	0	0	0	0	1	0	0	...	0	0
1	0	-1	0	0	0	0	-1	-1	0	1	...	1	1
2	0	1	0	0	0	-1	-1	-1	0	0	...	0	0
3	1	-1	1	0	1	0	0	1	0	0	...	0	0
4	0	-1	0	0	0	0	-1	-1	-1	0	...	0	0

5 rows × 31 columns

Рисунок 3.32 – Показ данных

3.2.2 Выборка данных для исследования и их упорядочивания. Адаптированные данных. Выделение столбцов в листы

Поскольку набор данных настолько массивен, его удобно фильтровать и отбирать. Создается переменная, указывающей на этот набор данных. (Приложение В):

TypLoga = ["ustroistvo","email","file","vhod","http"]

:ogPoleList = [["date","Polz","deystive"],["date","Polz","to","cc","bcc"], ["date","Polz","imyafila"], ["date","Polz","deystive"], ["date", "Polz", "url"]]

Указываем файлы .csv и какие из столбцов нам надо читать

df = pd.read_csv("dataset.csv", usecols=TypLoga, dtype=dtypes, LogList=logPoleList,index_col=None).

Начинается считываться данные для удобного изучения и манипулирования. Дальше, набор данных помещается в массивы для подготовки к машинному обучению. Набор данных состоит из нескольких тысяч векторов признаков для фишинговых адресов. Есть 30 функций, чьи имена и значения приведены на рисунке 3.33:

Attributes	Values	Column name
Having an IP address	{ 1,0 }	has_ip
Having a long URL	{ 1,0,-1 }	long_url
Uses Shortening Service	{ 0,1 }	short_service
Having the '@' symbol	{ 0,1 }	has_at
Double slash redirecting	{ 0,1 }	double_slash_redirect
Having a prefix and suffix	{ -1,0,1 }	pref_suf
Having a subdomain	{ -1,0,1 }	has_sub_domain
SSLfinal state	{ -1,1,0 }	ssl_state
Domain registration length	{ 0,1,-1 }	long_domain
Favicon	{ 0,1 }	favicon
s a standard port	{ 0,1 }	port
Uses HTTPS tokens	{ 0,1 }	https_token
Request_URL	{ 1,-1 }	req_url
Abnormal URL anchor	{ -1,0,1 }	url_of_anchor
Links_in_tags	{ 1,-1,0 }	tag_links
SFH	{ -1,1 }	SFH
Submitting to email	{ 1,0 }	submit_to_email
Abnormal URL	{ 1,0 }	abnormal_url
Redirect	{ 0,1 }	redirect
On mouseover	{ 0,1 }	mouseover
Right-click	{ 0,1 }	right_click
Pop-up window	{ 0,1 }	popup
Iframe	{ 0,1 }	iframe
Age of domain	{ -1,0,1 }	domain_age
DNS record	{ 1,0 }	dns_record
Web traffic	{ -1,0,1 }	traffic
Page rank	{ -1,0,1 }	page_rank
Google index	{ 0,1 }	google_index
Links pointing to page	{ 1,0,-1 }	links_to_page
Statistical report	{ 1,0 }	stats_report
Result	{ 1,-1 }	target

30 функций

Рисунок 3.33 – Таблица функций DDOS

Осуществляется отбор данных по критериям, то есть перебирается файлы .csv, содержащие журналы, и считываются во фрейма данных.

3.2.3 Создание функции для обнаружения угроз

На следующем шаге созданы функции, которые, помогут классификаторам ловит инсайдерские угрозы (Приложение В). Создаются удобные функцию для кодирования объектов, чтобы словарь мог их отслеживать:

```
Funcias = 0
Upor_po_funcia = {}
def dobavFuncia(name):
    if name not in Upor_po_funcia:
        global Funcias
        Upor_po_funcia[name] = Funcias
        Funcias+=1
```

Добавляю функции, которые буду использовать в словаре:

In[]:

```
dobavFuncia("Vyhodnoi_Vhod_Normal","Vyhodnoi_Vhod_After","Weeke
nd_Vhod"
```

,"Vyhod" ,"Connect_Normal","Connect_After","Connect_Weekend","Disco nnect", “Otpravitel”, “Poluchatel’)

На следующем шаге создается функцию, которые будут отслеживать тип файла, скопированного на съемный носитель. (рисунок 3.34).

```
def fileFeatures(row):
    """Функция которая будет записивать типы файла скопированные на съемный носитель"""
    if row["filename"].endswith(".exe"):
        return feature_map["File_exe"]
    if row["filename"].endswith(".jpg"):
        return feature_map["File_jpg"]
    if row["filename"].endswith(".zip"):
        return feature_map["File_zip"]
    if row["filename"].endswith(".txt"):
        return feature_map["File_txt"]
    if row["filename"].endswith(".doc"):
        return feature_map["File_doc"]
    if row["filename"].endswith(".pdf"):
        return feature_map["File_pdf"]
    else:
        return feature_map["File_other"]
```

Рисунок 3.34 – Функция для записывания типов файлов

Ссоздаётся е одна функцию для отслеживания, использовал ли сотрудник съемное устройство в нерабочее время (рисунок 3.35).

```
def ustroistvoFuncias(stroka):
    """определяем использовал ли сотрудник съемный носитель в нерабочее время
    """
    if stroka["devstive"] == "Connect":
        if stroka["date"].Vyhodnoi() < 5:
            if stroka["date"].hour >= 8 and stroka["date"].hour < 18:
                return Upor_po_funcia["Connect_Normal"]
            else:
                return Upor_po_funcia["Connect_After"]
        else:
            return Upor_po_funcia["Connect_Weekend"]
    else:
        return Upor_po_funcia["Disconnect"]
```

Рисунок 3.35 – Функция ustroistvoFuncia

Эта функция отслеживает вошел ли сотрудник в устройство в не-рабоч ее время (рисунок 3.36).

```
def vhodFuncias(stroka):
    """Определите функцию, чтобы отмечать, вошел ли сотрудник на компьютер в
нерабочее время:"""
    if stroka["deystive"] == "Vhod":
        if stroka["date"].Vyhodnoi() < 5:
            if stroka["date"].hour >= 8 and stroka["date"].hour < 18:
                return Upor_po_funcia["Vyhodnoi_Vhod_Normal"]
            else:
                return Upor_po_funcia["Vyhodnoi_Vhod_After"]
        else:
            return Upor_po_funcia["Weekend_Vhod"]
    else: #Is Vyhod
        return Upor_po_funcia["Vyhod"]
```

Рисунок 3.36 – Функция vhodFuncia

Следующая функция будет отслеживать подозрительность адрес сотрудника по электронной почте. Используется адреса, посещаемые сотрудниками, которые могут указывать на злонамеренное поведение (рисунок 3.37).

```
def emailFeatures(row):
  """Функция для определеение: отправил ли сотрудник электронное письмо
на несоответствующее письмо"""
  otpravitel = False
  if not pd.isnull(row["to"]):
    for address in row["to"].split(";"):
      if not address.endswith("standart.kz"):
        otpravitel = True
  if otpravitel:
    return feature_map["Otpravitel"]
  else:
    return feature_map["Poluchatel"]
```

Рисунок 3.37 – Функция emailFeatures

Затем упрощается данные, используется только даты, а не полная временная метка в подробных данных. Данные будут сохранены в сутках или один рабочий в день (рисунок 3.38).

```
def dateToDen(stroka):
    """из даты сохраняем только день"""
    tolkoden= stroka["date"].date()
    return tolkoden
```

Рисунок 3.38 – Функция dateToDen

3.2.4 Разделение данных на train и test. Векторизация данных для о бучения и тестирования

После завершения первой итерации этапа разработки функций нужно посмотреть дает ли хорошие результаты данная стратегия. Тестовые наборы нужны для последующей проверки параметров, которые будут использоваться и определятся насколько они чувствительные к аномалиям. Данные будут выглядеть как продолжительность общения между двумя устройства типы протокола являющийся TCP, http, которые будет уже преобразованы под форму для алгоритма. Затем делается с другой информации и меткой, которая нужна для нормального трафика или атаки. (Приложение С).

Разделив данные на подмножества для обучения и тестирования, состоящие из первых 80% и последних 20% данных, получаем (рисунок 3.39):

```
df2 = df.sort_values("Timestamp")
```

```
In[]:
d = len(df2.index)
obuch_df = df2.head(int(d*0.8))
test_df = df2.tail(int(d*0.2))
```

Подготовляю метки:

```
In[]:
from collections import Counter
print(Counter(obuch_df[‘metka’].values))
print(Counter(test_df[‘metka’].values))
```

```
Counter({'Vse': 332658, 'ddos': 66549})
Counter({'Vse': 66234, 'ddos': 33598})
```

```
In[]:
y_obuch = obuch_df.pop(‘metka’).values
y_test = test_df.pop(‘metka’).values
```

Подготовляю векторы объектов:

```
In[]:
X_obuch = obuch_df.values
X_test = test_df.values
```

```
traindf.head()
```

ssl_state	long_domain	favicon	...	popup	iframe	domain_Age	dns_record	traffic	page_rank	google_index	links_to_page	stats_report	target
1	0	0	...	0	0	-1	1	1	-1	0	1	0	-1
-1	0	1	...	1	1	-1	1	0	0	0	1	1	1
-1	0	0	...	0	0	-1	1	-1	-1	0	1	0	1
1	0	0	...	0	0	0	1	1	-1	0	1	0	-1
-1	-1	0	...	0	0	-1	0	1	0	0	-1	0	1

```
y_train = traindf.pop("target").values
y_test = testdf.pop("target").values
```

```
X_train = traindf.values
X_test = testdf.values
```

Рисунок 3.39 – Данные разделенные на train и test

Считываются упрощенные данные во фрейм данных pandas. Затем редактируются текущий формат даты, чтобы соответствовал формату pandas, а затем берутся все новые функции, отбрасывая старые. Преобразую данные во временные ряды, дельта которых - отдельные дни (рисунок 3.40):

```
for i in range(len(TypLogas)):
    TypLoga = TypLogas[i]
    logPoles = logPolesList[i]
    logFunciaFuncia = logFunciaFuncias[i]
    fr = pd.read_csv(put_v_dataset + TypLoga+".csv", usecols=logPoles, index_
col=None)
    #Конвертация со стандартной библиотеки
    dateFormat = "%m/%d/%Y %H:%M:%S"
    fr["date"] = pd.to_datetime( df["date"], format=dateFormat)
    #создаю новую функцию и убираю остальные, кроме date, Polz и новой функци
и
    newFuncia = df.apply(logFunciaFuncia, axis=1)
    fr["Funcia"]=newFuncia

    stol_s_sohr = ["date", "Polz", "Funcia"]
    fr = fr[stol_s_sohr]
    #Преобразуем дату в день
    fr["date"]=df.apply(dateToDen, axis=1)
    fr.append(df)
```

Рисунок 3.40 – Счетчик для записи словаря функции

Все данные объединяются в один большой отсортированный фрейм данных.

In[]:

joint = pd.concat(dfs)

In[]:

joint = joint.sort_values(Po="date")

На рисунке 3.41 перечисляю всех участников угроз.

joint

	date	user	feature
0	2010-01-02	MOH0273	6
1	2010-01-02	MOH0273	7
2	2010-01-02	HPH0075	6
3	2010-01-02	IIW0249	6
4	2010-01-02	IIW0249	7
...	...	...	...
28434418	2011-05-16	BRM0995	17
28434419	2011-05-16	BRM0995	17
28434420	2011-05-16	ZSB0649	17
28434421	2011-05-16	BAM0636	17
28434422	2011-05-16	CGB0637	17

32770222 rows × 3 columns

Рисунок 3.41 – Список участников угроз

Создается индексацию для дат, чтобы 0 соответствовало начальной дате, 1 - следующему дню. В последующих шагах функция определяется для чтения во всем временном ряду набора данных, фильтрации отдельных пользователей, а затем векторизации временного ряда для каждого пользователя (рисунок 3.42):

In[]:

obuch_df = joint[joint["date"]<=d]

test_df = joint[joint["date"]>=d]

```
def collectDataset(df, Upor_po_funcia, dateK_Index):
    Polzs = set(df["Polz"].values)
    X = np.zeros((len(Polzs),len(Upor_po_funcia)*timeWindow))
    PolzK_Index = {}
    Index_KPolz = {}
    i = 0
    for Polz in Polzs:
        x = OrgPolzTime(Polz, df, Upor_po_funcia, dateK_Index)
        X[i,:]=x.flatten()
        PolzK_Index[Polz]=i
        Index_KPolz[i]=Polz
        i+=1
    return X, PolzK_Index, Index_KPolz
```

Рисунок 3.42 – Функция collect Dataset

Далее векторизуется набор данных и разбивается для тестирования и обучения. Данные изменены, чтобы иметь возможность передать их в классификатор изолированного леса (рисунок 3.43).

In[]:

X_obuch, obuch_PolzK_Index, obuch_Index_KPolz = collectDataset(obuch_df, Upor_po_funcia, obuch_dateK_Index)

X_test, test_PolzK_Index, test_Index_KPolz = collectDataset(test_df, Upor_po_funcia, test_dateK_Index)

#создаю листы для инциндентов

IstUgrozIndicesObuch = set([])

IstUgrozIndicesTest = set([])

for IstUgroz in IstUgrozs:

    if IstUgroz in obuch_PolzK_Index:

        IstUgrozIndicesObuch.dobav(obuch_PolzK_Index[IstUgroz])

    if IstUgroz in test_PolzK_Index:

        IstUgrozIndicesTest.dobav(test_PolzK_Index[IstUgroz])

In[]:

#Изменяю форму векторизованных данных:

obuch_normalIndices = set(obuch_Index_KPolz.keys()) - IstUgrozIndicesObuch

test_normalIndices = set(test_Index_KPolz.keys()) - IstUgrozIndicesTest

In[]:

#Obuch-test разделяет векторизованные данные:

y_obuch = np.zeros(len(X_obuch))

y_obuch[list(IstUgrozIndicesObuch)]=1

y_test = np.zeros(len(X_test))

y_test[list(IstUgrozIndicesTest)]=1

```
#Разделяю наборы данных обучения и тестирования на подгруппы угроз и не-угроз
X_train_normal = X_train[list(train_normalIndices),:]
print(X_train_normal.shape)
X_train_threat = X_train[list(threatActorIndicesTrain),:]
print(X_train_threat.shape)
X_test_normal = X_test[list(test_normalIndices),:]
print(X_test_normal.shape)
X_test_threat = X_test[list(threatActorIndicesTest),:]
print(X_test_threat.shape)
```

```
(930, 4518)
(70, 4518)
(891, 4518)
(51, 4518)
```

Рисунок 3.43 – Просмотр разделенных данных

3.3 Анализ результатов по завершению построений и использования моделей классификаторов

3.3.1 Импортирование библиотеки Scikit-learn. Установка параметров для создания классификаторов Isolation Forests и Random forests

Завершив этап разработки функций, благодаря машинному обучению создается модель и можно классифицировать неконтролируемый поток данных. Импортируются и создается экземпляр классификатора случайного леса. Случайный лес также является очень удобным алгоритмом. Используемые по умолчанию гиперпараметры часто дают хороший результат прогнозирования (рисунки 3.44-3.45).

```
[11]: from sklearn.ensemble import RandomForestClassifier
      clf=RandomForestClassifier(n_estimators=50)

[12]: clf.fit(X_train, y_train)

[12]: RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
                             max_depth=None, max_features='auto', max_leaf_nodes=None,
                             min_impurity_decrease=0.0, min_impurity_split=None,
                             min_samples_leaf=1, min_samples_split=2,
                             min_weight_fraction_leaf=0.0, n_estimators=50,
                             n_jobs=None, oob_score=False, random_state=None,
                             verbose=0, warm_start=False)

[13]: clf.score(X_train, y_train)

[13]: 0.99688

[14]: clf.score(X_test, y_test)

[14]: 0.6906
```

Рисунок 3.44 - Использование алгоритма случайного леса.

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_rez, confusion_matrix
import itertools
aues = RandomForestClassifier()
aues.fit(X_obuch, y_obuch)
y_test_pred = aues.predict(X_test)
```

Рисунок 3.45 – Описание функций предсказания

Вот так выглядит установленный параметрами алгоритм:
from sklearn.ensemble import RandomForestClassifier
aues=RandomForestClassifier(n_estimators=45)

In []:
aues.fit(X_obuch, y_obuch)
RandomForestClassifier (bootstrap=True, class_weight=None,

max_depth=None, max_Funcias='auto', max_leaf_nodes=None,
min_impurity_decrease=0.0, min_impurity_split=None,
min_samples_leaf=1, min_samples_split=2,
min_weight_fraction_leaf=0.0, n_estimators=45, n_jobs=None,
oob_rez=False, random_state=None, verbose=0,
warm_start=False)

Следующий использующийся алгоритм называется «Изолированный Лес». Алгоритм зависит от наблюдения, когда аномальные данные лучше изолировать от обычных событий. Например, попытка описать среднюю нормальную распределенную точку деревьев решений. Для описания аномальных точек, где средняя длина пути деревьев больше, точка намного выделяется чем в логической структуре (рисунок 3.46).

```
#Определяем и создаем экземпляр классификатора изолированного леса:
from sklearn.ensemble import IsolationForest
zagrParametr = 0.07
IF = IsolationForest(n_estimators=100, max_samples=256,zagr=zagrParameter)
IF.fit(X_obuch)
normalRezs = IF.reshenive_funcia(X_obuch_normal)
```

Рисунок 3.46 – Установка параметров на модель «Изолированный лес»

Эти параметры вводится в алгоритм Изолированного леса:

— n_estimators: количество деревьев для использования, предлагается количество 100 деревьев, потому что длины пути обычно сходятся задолго до этого.

— max_samples: количество выборок, которые можно нарисовать при построении одного дерева. Этот параметр предлагается max_samples = 256, поскольку он обычно предоставляет достаточно деталей для обнаружения аномалий в широком диапазоне данных.

— contamination: количество загрязнения набора данных, то есть доля выбросов в наборе данных. Используется при подгонке для определения порога функции принятия решения (рисунок 3.47).

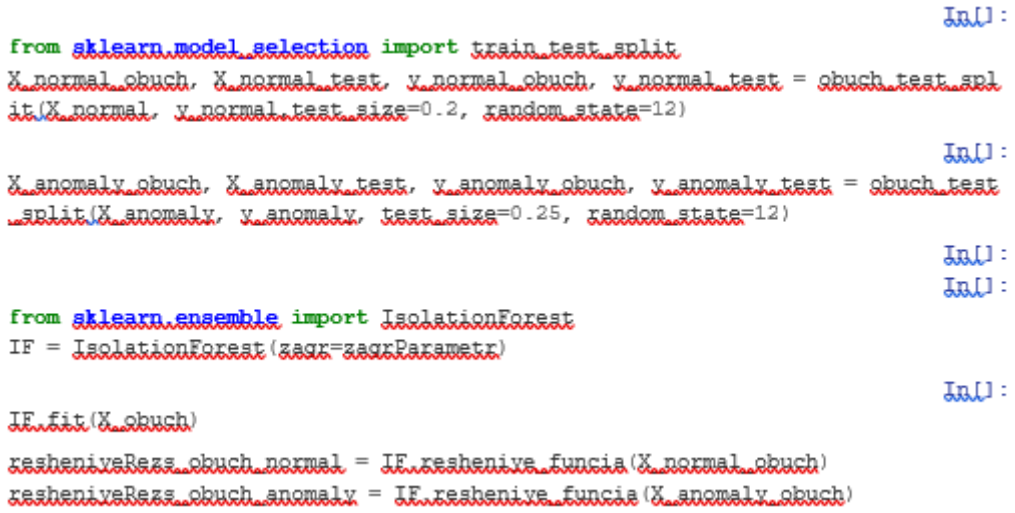


Рисунок 3.47 – Установка параметров «Изолированный лес» на разделенные данные

3.3.2 Построение графика для определения области вредоносных данных

С помощью алгоритма изолированного леса определяется событий с угрозами и другими событиями, где ничего аномального не произошло.

```
import matplotlib.mlab as mlab
import matplotlib.pyplot as stat
normal = stat.hist(normalRezs, 45,density=True)
stat.xlabel('Аномальные значения')
stat.ylabel('Проценты')
stat.title("Распределение аномальных значений для обычных событий")
stat.figure(figsize=(10, 5), dpi=500)
#делаем тоже самое для инициаторов угроз
anomaly = stat.hist(anomalyRez, 45,density=True)
stat.xlabel('Аномальные значения')
stat.ylabel('Проценты')
stat.title("Распределение аномальных значений для угроз")
Out []:
Text('Распределение аномальных значений для угроз')
```

График оценки решения нормального подмножества данных обучения представлен на рисунке 3.48. График распределения угроз иллюстрируется рисунком 3.49.

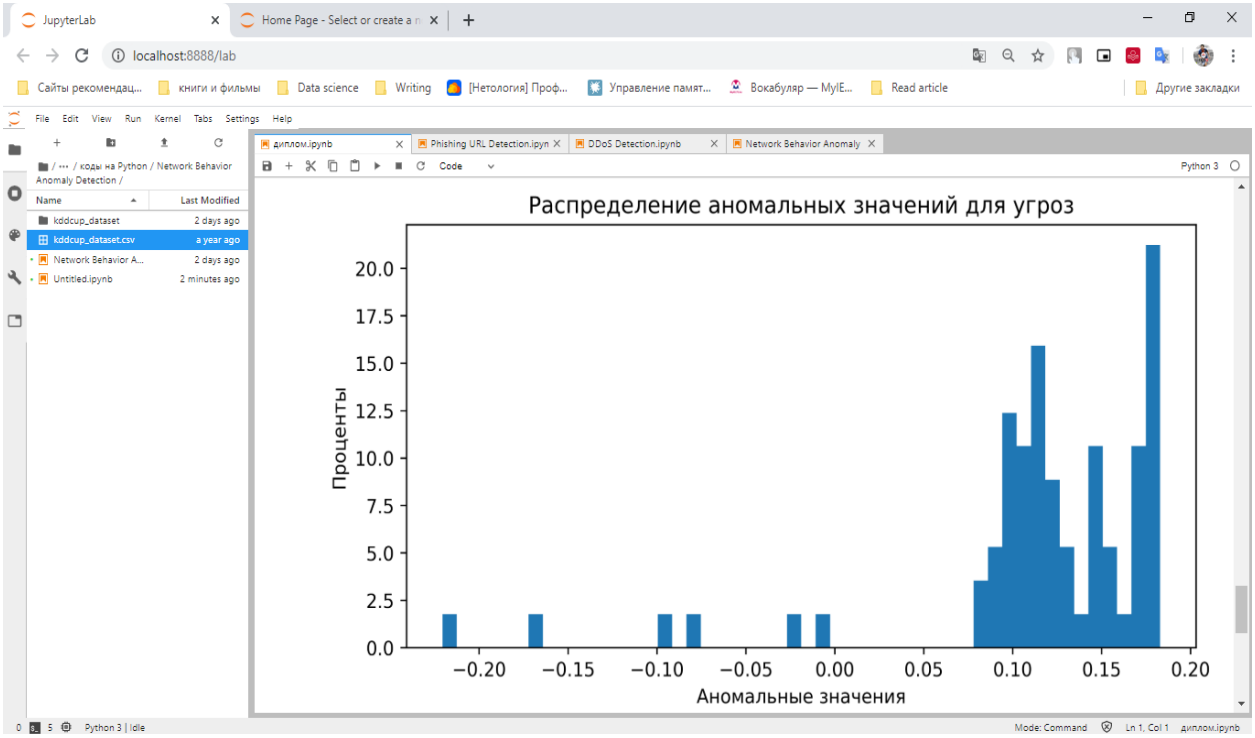


Рисунок 3.48 – Процесс компиляции на Jupyter

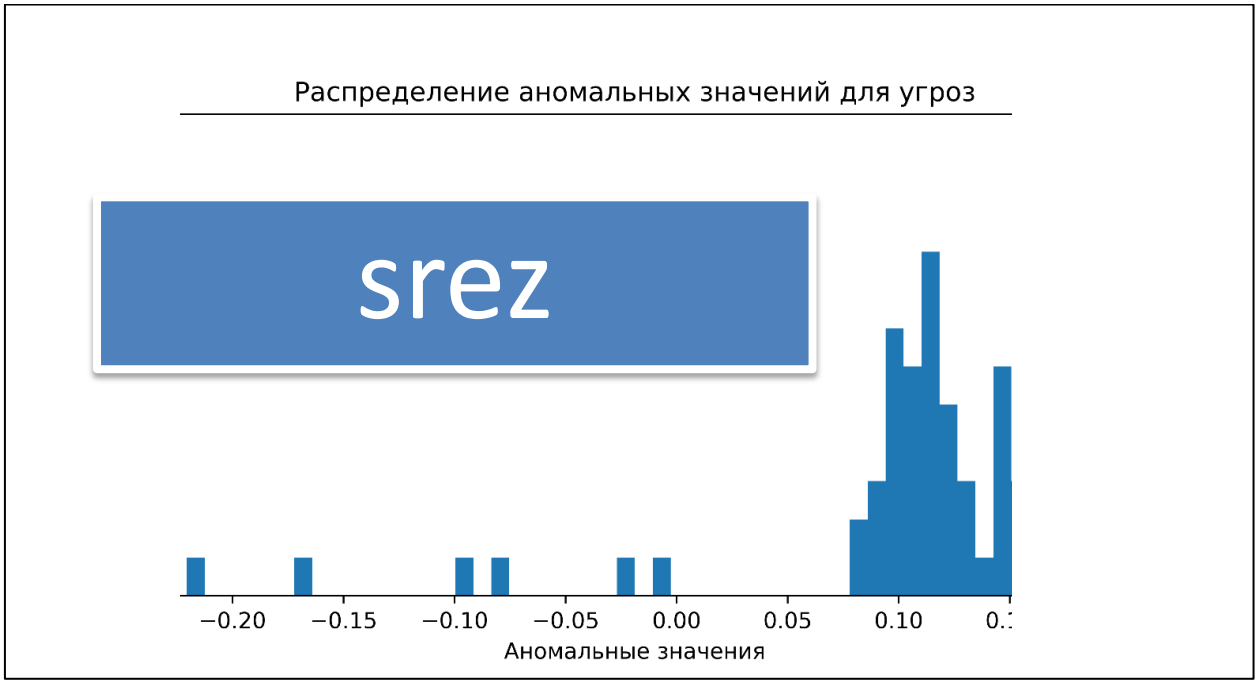


Рисунок 3.49 – Распределение аномальных значений для угроз

Устанавливается случайный лес для обучающих данных для последующей оценки по данным тестирования (рисунок 3.50):

from sklearn.model_selection import train_test_split

X_normal_obuch, X_normal_test, y_normal_obuch, y_normal_test = obuch _test_split(X_normal, y_normal,test_size=0.2, random_state=12)

In[]:

X_anomaly_obuch, X_anomaly_test, y_anomaly_obuch, y_anomaly_test = obuch_test_split(X_anomaly, y_anomaly, test_size=0.25, random_state=12)

In[]:
```
from sklearn.ensemble import IsolationForest
IF = IsolationForest(zagr=zagrParametr)
```

In[]:
```
IF.fit(X_obuch)
```

```
resheniyeRezs_obuch_normal = IF.resheniye_funcia(X_normal_obuch)
resheniyeRezs_obuch_anomaly = IF.resheniye_funcia(X_anomaly_obuch)
```

In[]:
```
import matplotlib.pyplot as stat
%matplotlib inline
stat.figure(figsize=(20, 10))
_ = stat.hist(resheniyeRezs_obuch_normal, bins=45)
```

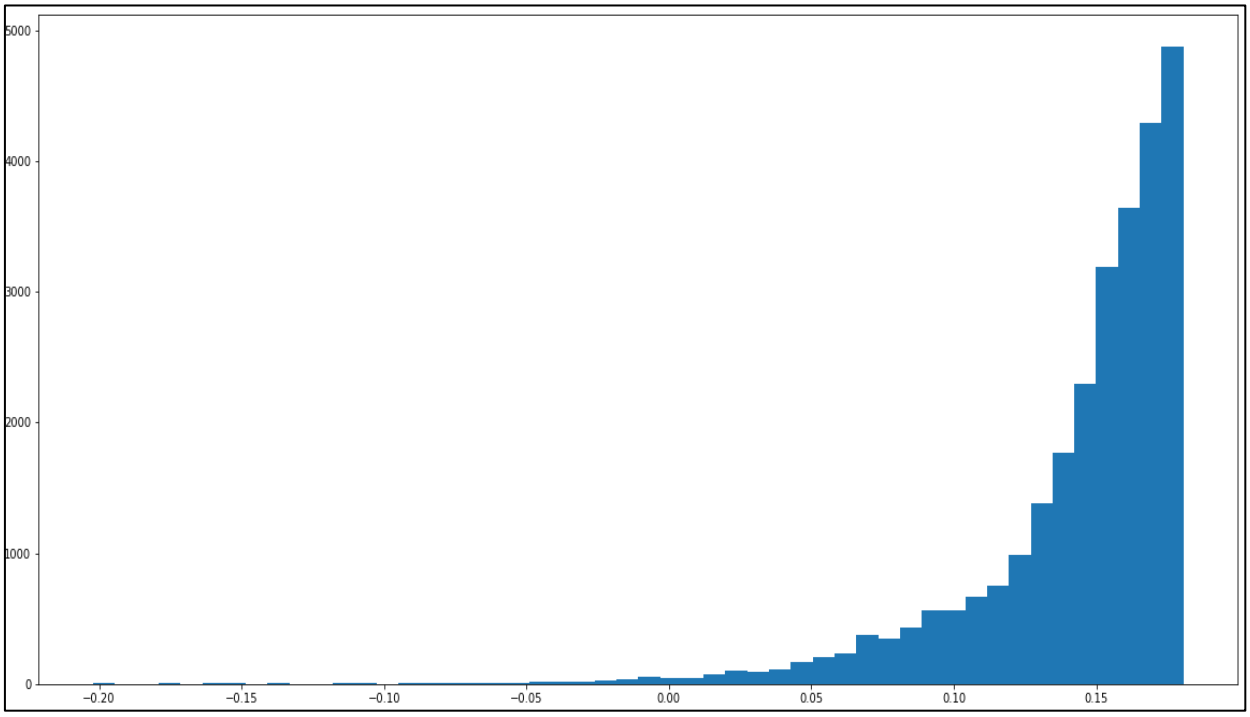


Рисунок 3.50 – Распределение опасного сетевого трафика

Получаю распределение аномалий (рисунок 3.51):
In[]:
```
stat.figure(figsize=(25, 15))
_ = stat.hist(resheniyeRezs_obuch_anomaly, bins=45)
```

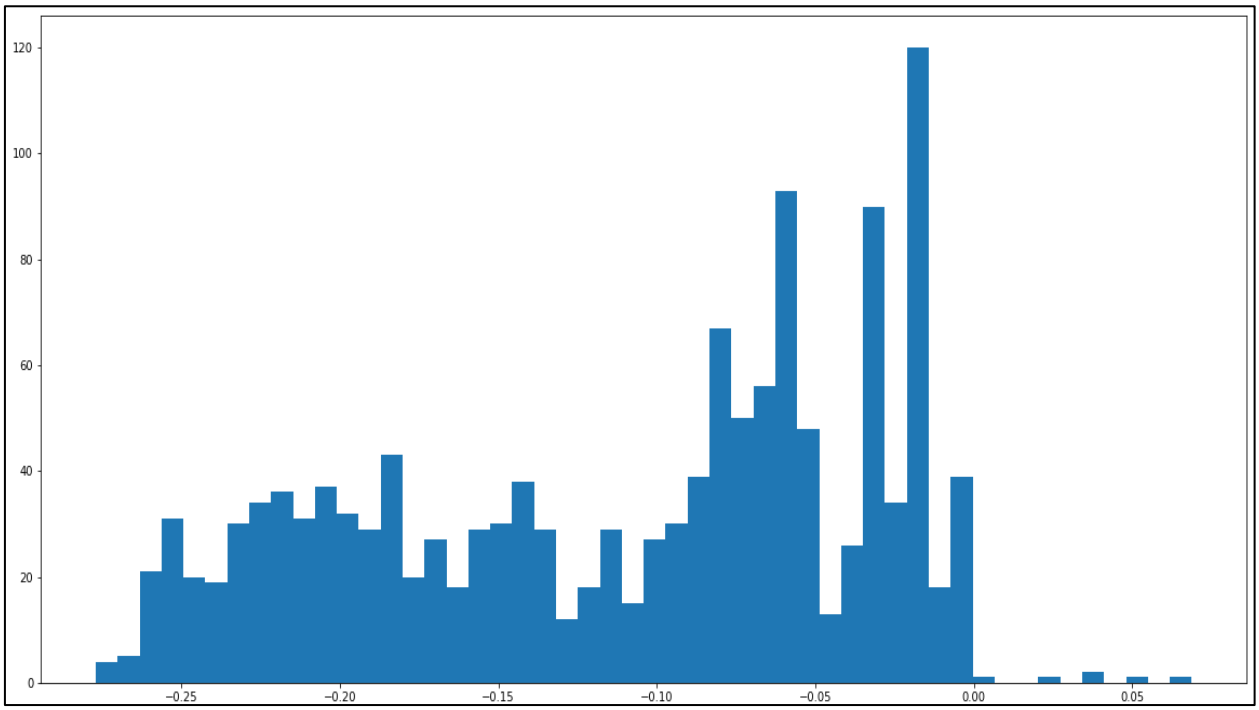


Рисунок 3.51 – Распределение аномалий

3.3.3 Анализ полученных результатов через коэффициент cutoff и матрицы ошибок

Оцениваю классификатор в матрице ошибок. С целью сопоставления предсказаний и реальности в машинном обучении используется матрица ошибок (confusion matrix) - с различными комбинациями прогнозируемых и фактических значений, листинг и результат представлены на рисунке 3.52.

```
y_test_pred = rfc.predict(X_test)
print("Точность классификации")
print(accuracy_score(y_test, y_test_pred))
```

```
Точность классификации
0.9837133550488599
```

```
Оцениваю классификатор в матрице ошибок. С целью сопоставления предсказаний и реальности в машинном обучении используется
матрица ошибок (confusion matrix) – таблица с 4 различными комбинациями прогнозируемых и фактических значений.
```

```
print("Матрица ошибок")
print(confusion_matrix(y_test, y_test_pred))
```

```
Матрица ошибок
[[344   3]
 [  7 260]]
```

Рисунок 3.52 – Распределение аномалий

Точность классификации показал 0.9837. В зависимости от применения достигнутая точность является хорошей отправной точкой. Это результат дает доверия на изучения результатов матрицы ошибок. Результат: первая ячейка показывает в сетевом трафика, 344 - обычные события, а трех классификатор сомневается. Из вредоносных данных в семи он сомневается, а в 260 уверен что произошел фишинг.

В первом коде сделали оценку модели фишинга и определились схожие показатель с меткой.

```
ddos=rfc.score(X_test, y_test)
print(ddos)
```

```
0.6906
```

```
Здесь показываю на каком этапе классификатор нашел угрозы:
```

```
print(rfc.feature_importances_)
```

```
[0.16911444 0.15088217 0.10376505 0.31881767 0.25742067]
```

Рисунок 3.53 – Функция feature importances

Функция feature importances определила угрозы DDOS атаки в строке общее количество байтов, отправленных в начальном окне в обратном направлении.

В третьем и четвертом используется функция Decision_function, которая обеспечивается счет производной средней длины пути образцов в модели. Результаты отсечения по тренировочным данным: srez = -0.05

In[]:

```
from collections import Counter
c= IF.resheniye_funcia(X_obuch)
print(Counter(y_obuch[srez>c]))
```

```
Counter({0.0: 31, 1.0: 3})
```

In[]:

```
c = IF.resheniye_funcia(X_test)
print(Counter(y_test[srez>c]))
```

```
Counter({0.0: 45})
srez = 0
```

In[]:

```
print(Counter(y_test))
print(Counter(y_test[srez>IF.resheniye_funcia(X_test)]))
```

```
Counter({0: 11875, 1: 597})
Counter({1: 595, 0: 85})
```

Дальше происходит процесс экспортирования данных в html-файл и их визуализация, которые показаны на рисунках 3.53-3.55.

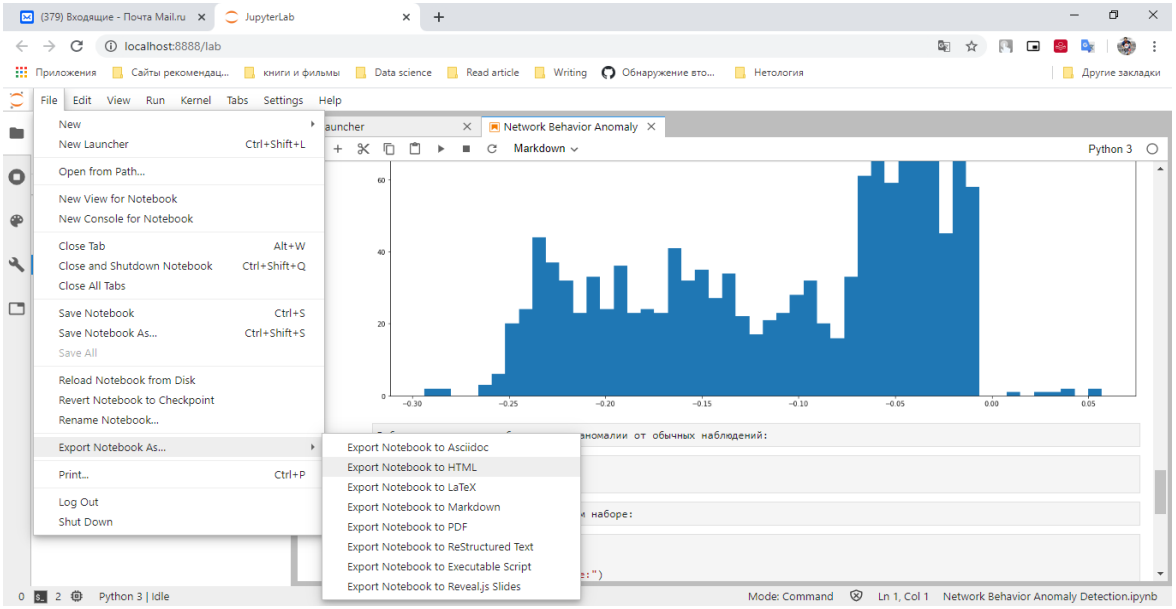


Рисунок 3.54 – Процесс экспортирования

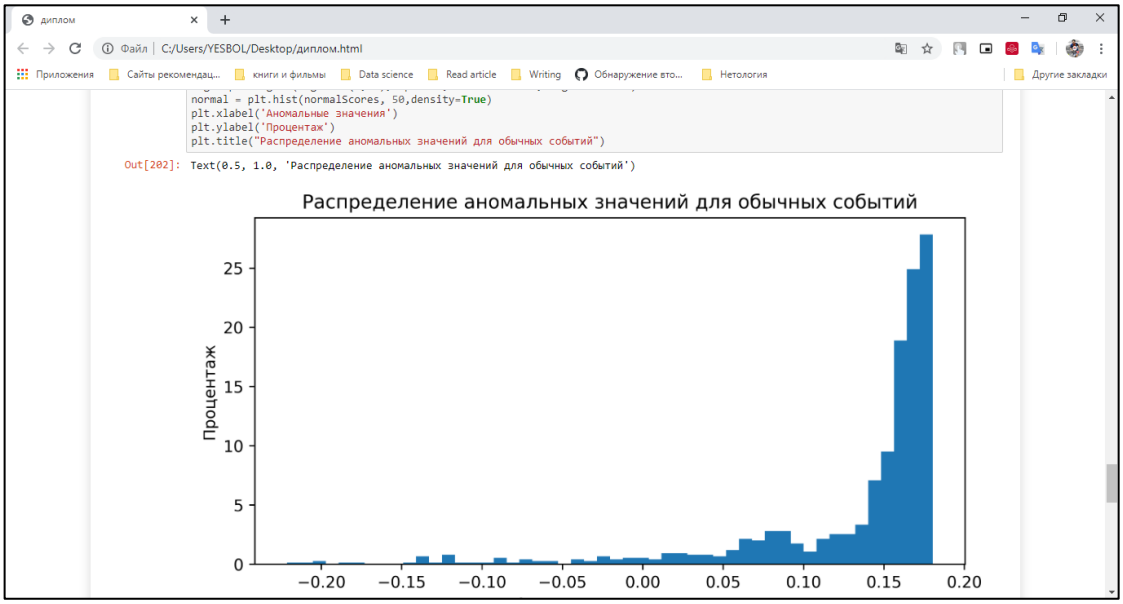


Рисунок 3.55 – Визуализация в браузере

Сопоставив два графика, выбрали отсечение для отделения аномалии от обычных наблюдений. Определяем угрозы из оставшийся данных после отсечения, на которые стоит среагировать аналитику безопасности.

```
Весь трафик
Counter({0: 9812, 1: 597})
Только аномальное поведение
Counter({1: 596, 0: 66})
```

Рисунок 3.56 – Результаты ограничение в тестовом наборе

Большинство вредоносных событии классификатор смог распределить. И почти из 10000 событий он определил 650 вредоносных событий.

Если оценивать эффективность классификатора, то встречались ложные срабатывания. Несмотря на это, были обнаружено значительное коли чество внутренних угроз. Поскольку соотношение было не слишком высоким, классификатор может оказать большую помощь в информировании аналитиков о вероятных угрозах.

Вывод

Поскольку набор данных большой, даже импорт данных требует больших вычислительных ресурсов. По этой причине начинаем с определения подмножества функций, а также записи их типа данных, чтобы не пришлось преобразовывать. Затем переходим к чтению данных во фрейм данных. Следующем шаге сортируем данные по дате, поскольку проблема требует возможности прогнозировать события в будущем. Затем, выполняем разделение на тренировочные и тестируемые, имея в виду временную прогрессию. Создаем, подбираем и тестируем данные случайным лесным классификатором. В во втором коде функция feature importances определила угрозы DDOS атаки в одной из характеристик данных, а именно в строке общее количество байтов, отправленных в начальном окне обратном направлении. Результаты первого кода показали что в сетевом трафика из вредоносных событий в семи он сомневается, а в 260 уверен что произошел фишинг.

Также использовался другой экземпляр классификатора (изолированного лес). Для параметра загрязнения использовали значения, соответствующее соотношению угроз для не вредоносных событий. На следующих трех этапах изучили показатели принятия решений по изолированному лесу на доброкачественные и опасные субъекты и пришли к выводу, что посредством проверки, пороговое значение 0,05 в третьем коде и 0.01 в четвертом обнаруживает большую долю субъектов угроз без отметки обычных пользователей. Наконец, оценивая нашу эффективность в шагах увидел, что были некоторые ложные срабатывания, но также обнаружено значительное количество внутренних угроз. Поскольку соотношение было не слишком высоким, классификатор может оказать помощь в информировании аналитиков о вероятных угрозах.

Оценка рисков ИБ

4.1 Управление рисками проекта

Риск характеризуется определенными источниками или же основаниями и содержит результаты, оказывает воздействие на итоги проекта. Главными текстами в определении считаются:

— вероятность;
— событие;
— субъект;
— решение;
— потери.

Основными процедурами данного облика управления считаются:

— идентификация;
— оценка;
— планирование реагирования;
— мониторинг и контроль.

4.2 Анализ и оценка проектных рисков

Анализ и оценка рисков делаются с целью преобразование добытых в ходе идентификации сведений информации, позволяющая принимать серьезные решения. В ходе процесса высококачественного анализа производится ряд экспертных оценок не очень благоприятными результатами. В процессе количественного анализа ориентируются и устанавливаются значения характеристик вероятности появления угрожающих мероприятий. Количественный анализ настоятельно требует свойства входных данных, применения развитых математических моделей и больше высочайшей компетентности от персонала. На выходе аналитической работы менеджер проекта намерен получить:

— оценку рискованности проекта;
— сгруппированный по ценностям перечень рисков;
— список позиций дополнительного анализа.

Экспертные оценки это вероятность наступления не очень благоприятных событий и значения влияния на проект. Основным выходом процесса качественного анализа считается перечень ранжированных рисков с выполненными оценками или оформленная карта рисков. И вероятности, и воздействия разбиваются на категориальные группы в данном спектре значений.

Рисунок 4.1 - Матрица вероятности и воздействия

В результате оценок возводятся всевозможные особые матрицы, в ячейках которых помещаются итоги произведения значения вероятности на степень влияния. Полученные итоги разделяются на сегменты, которые служат базой для ранжирования угроз.

Матрица вероятности и последствий имеет комбинации вероятности и влияния и рискам дается определенный ранг: низкий, средний или высших.

Матрица содержит определение терминов или цифровые обозначения и строится на основании шкал оценки вероятности и оценки степени воздействия вероятности риска. Левый столбец матрицы имеет значения вероятности появления риска, в 1 строке размещена шкала со значениями возможных последствий. Ячейки заполняется результатами перемножения значений данных шкал.

Риски, имеющие довольно высокие вероятности, но незначительные последствия, а также риски, имеющие низкие вероятности и незначительные последствия, считаются рисками, не оказывающими воздействия. Риски с очень большими последствиями, но малой вероятностью, как и риски с незначительными последствиями и высокой вероятностью (клетки светло-серого цвета) имеют среднее воздействие на проект. Риски, которым необходимо уделять особое внимание, имеют достаточно высокую вероятность и существенные последствия (клетки таблицы, окрашенные темно-серым цветом).

Внизу представлены заполненные формы для моего проекта, для общего анализа проектных рисков, описание процедуры для управления рисками, форма запроса на регистрацию риска, описание процедуры управления рисками проекта.

Таблица 4.1 – Внутренняя информация о клиенте

Название клиента	ТОО Колеса
Основное контактное лицо	Колесник А.Н.
Код проекта	ML
Ответственный за проект	Сапаргали Е.Е.
Руководитель проекта	Зуева Е.А.

Таблица 4.2 – Одобрения рисков на фазах проекта

Фаза	**Анализ рисков провел**	**Анализ рисков одобрила**	**Дата одобрения**
Сбор и загрузка данных	Сапаргали Е.Е.	Дмитриева М.В.	2.02.2020
Предобработка данных	Сапаргали Е.Е.	Дмитриева М.В.	2.02.2020
Формирование признаков	Сапаргали Е.Е.	Дмитриева М.В.	4.02.2020
Построение модели (по алгоритму)	Сапаргали Е.Е.	Дмитриева М.В.	5.02.2020
Использование модели	Сапаргали Е.Е.	Дмитриева М.В.	7.02.2020
Анализ результатов	Сапаргали Е.Е.	Дмитриева М.В.	11.02.2020

Таблица 4.3 – Количество проектных рисков по фазам

Итог по фазе	**Высокие риски**	**Средние риски**	**Низкие риски**	**Кол-во рисков**
Сбор и загрузка наборов данных	1	2	0	3
Предобработка данных	0	2	0	2
Формирование признаков у данных	1	0	1	2
Построение модели (по алгоритму)	0	0	1	1
Использование модели	0	0	1	1
Анализ результатов	0	1	0	1
Всего	2	5	3	10

Таблица 4.4 - Запросы на регистрацию риска для первой фазы

Запрос на регистрацию риска **Номер в журнале рисков:** 1	
ФИО автора запроса: Сапаргали Е.Е. **Роль на проекте**: data-engineer **Наиме нование проекта**: ML **Фаза проекта**: Сбор и загрузка данных	**Приоритет**: Высокий **Дата запроса**: 2.02.2020 **Желаемая дата разрешения**: 5.02.2020
Описание риска: Ненадежные наборы данных. Данные для анализа могут взяты из ненадежных и некомпетентных источников. **Предпосылки**: Отсутствие формального процесса санкционирования общедоступной информации. **Последствия**: В конце полученные результаты могут быть не правильными самого раннего этапа и без возможности корректировки формул на определенных этапах. **Варианты решения**: -Изучение инженером надежности источника данных; -Формирование списка надженых источников на обсуждениях на коллегиальных дискуссиях; -Брать с надежных источников предаставленные например, институтом.	

Таблица 4.5 - Запросы на регистрацию риска

Запрос на регистрацию риска **Номер в журнале рисков: 2**	
ФИО автора запроса: Сапаргали Е.Е. **Роль на проекте**: разработчик **Наименование проекта**: ML **Фаза проекта**: Сбор и загрузка данных	**Приоритет**: Средний **Дата запроса**: 5.02.2020 **Желаемая дата разрешения**: 7.02.2020
Описание риска: Сбой программных средств. Сбой программных средств платформы Anaconda после запуска одновременного нескольких CSV. **Предпосылки**: Отсутствие "завершения, перезагрузки сеанса" при уходе с рабочего места. **Последствия**: Данные приходится убирать с установленного каталога и теряется действия с предыдущих сеансов в платформе. **Варианты решения**: -Ограничить одновременный запуск нескольких CSV файлов; -Переносить данные из каталога в другую папку, чтобы программа не смогла его найти.	

Таблица 4.6 - Запросы на регистрацию риска

Запрос на регистрацию риска **Номер в журнале рисков: 3**	
ФИО автора запроса: Сапаргали Е.Е. **Роль на проекте**: администратор **Наименование проекта**: ML **Фаза проекта**: Сбор и загрузка данных	**Приоритет**: Средний **Дата запроса**: 5.02.2020 **Желаемая дата разрешения**: 7.02.2020
Описание риска: Злоупотребления правами. Злоупотребление правами сотрудником рангом по ниже в платформе Anaconda. **Предпосылки**: Неверное распределение прав доступа в платформе Anaconda. **Последствия**: Сотрудник имеющий права на которые не имеют доступа права может делать недобросответвые дейтсвия. **Варианты решения**: - Определение прав для каждого сотрудника; - Root права должны распределять только определенным людям.	

Таблица 4.7 - Запросы на регистрацию риска

Запрос на регистрацию риска **Номер в журнале рисков: 4**	
ФИО автора запроса: Сапаргали Е.Е. **Роль на проекте**: администратор **Наименование проекта**: ML **Фаза проекта**: Предобработка данных	**Приоритет**: Средний **Дата запроса**: 6.02.2020 **Желаемая дата разрешения**: 9.02.2020
Описание риска: Потеря электропитания. Потеря электропитания блока питания аппаратного средства. **Предпосылки**: Чувствительность к колебаниям напряжения. **Последствия**: Сгорание блоки питания и теряется пред обработанные данные. **Варианты решения**: -Обеспечить использование резервных систем электропитания; -Делать резервное копирование и сохранение изменения программных кодов с этапа предобработки данных.	

Таблица 4.8 - Запросы на регистрацию риска

Запрос на регистрацию риска **Номер в журнале рисков: 5**	
ФИО автора запроса: Сапаргали Е.Е. **Роль на проекте**: data-engineer **Наименование проекта**: ML **Фаза проекта**: Предобработка данных	**Приоритет**: Средний **Дата запроса**: 9.02.2020 **Желаемая дата разрешения**: 11.02.2020
Описание риска: Смещение результатов. Смещение результатов в отношении определенных категорий данных в направлении не необходимых целям проекта. **Предпосылки**: Некорректное предобработка данных инженера. **Последствия**: Получение не совсем корректных результатов, а также результатов не коррелирующим целям проекта/ **Варианты решения**: - Разрешение проблем, когда машина не может понять человеческую сущность и человеческие мысли; - Считывать также грамматичекские ошибки; - Считывать весьма аномальные цифры в ячейках.	

Таблица 4.9 - Запросы на регистрацию риска

Запрос на регистрацию риска **Номер в журнале рисков: 6**	
ФИО автора запроса: Сапаргали Е.Е. **Роль на проекте**: безопасник **Наименование проекта**: ML **Фаза проекта**: Формирование признаков	**Приоритет**: Высокий **Дата запроса**: 2.02.2020 **Желаемая дата разрешения**: 9.02.2020
Описание риска: Хищение носителей данных. Хищение носителей данных аппаратного средства. **Предпосылки**: Незащищенное хранение данных аппаратного средства в предприятии. **Последствия**: Может получить доступ к внутренностям аппаратного средства и получить данные в котором были формированы признаки для машинного обучения и для последующих этапов. **Варианты решения**: -Установка системы контроля безопасности в помещения, где хранится аппаратное средство; -Защитить от несанкционированного доступа при хищении.	

Таблица 4.10 - Запросы на регистрацию риска

Запрос на регистрацию риска **Номер в журнале рисков: 7**	
ФИО автора запроса: Сапаргали Е.Е. **Роль на проекте**: data-engineer **Наименование проекта**: ML **Фаза проекта**: Формирование признаков	**Приоритет**: Низкий **Дата запроса**: 11.02.2020 **Желаемая дата разрешения**: 14.02.2020
Описание риска: Затруднительное применение МО. Затруднительное применение сетей машинного обучения для анализа данных. **Предпосылки**: Отсутствие меток в наборах данных. **Последствия**: Данные будет неправильно обучаться без ориентира. **Варианты решения**: -Создания классов меток для использования методов машинного обучения; -Не брать большие наборы данных со многими характеристиками без столбцов с метками.	

Таблица 4.11 - Запросы на регистрацию риска

Запрос на регистрацию риска **Номер в журнале рисков: 8**	
ФИО автора запроса: Сапаргали Е.Е. **Роль на проекте**: администратор **Наименование проекта**: ML **Фаза проекта**: Построения алгоритма	**Приоритет**: Низкий **Дата запроса**: 7.02.2020 **Желаемая дата разрешения**: 9.02.2020
Описание риска: Ухудшение состояния элементов. Ухудшение состояния элементов аппаратного средства (в том числе носителей данных). **Предпосылки**: Отсутствие периодического обслуживания аппаратного средства. **Последствия**: Ухудшение работоспособности системы, вплоть до отключение аппаратного средства. **Варианты решения**: - Поставить сотрудника, который будет периодический проводить диагностику аппаратного средства; - Поменять элемент аппаратного средства который не пригоден для работы.	

Таблица 4.12 - Запросы на регистрацию риска

Запрос на регистрацию риска **Номер в журнале рисков: 9**	
ФИО автора запроса: Сапаргали Е.Е. **Роль на проекте**: разработчик **Наименование проекта**: ML **Фаза проекта**: Использование модели	**Приоритет**: Низкий **Дата запроса**: 10.02.2020 **Желаемая дата разрешения**: 16.02.2020
Описание риска: Нелегальная обработка данных. Нелегальная обработка данных с встроенного текстового редактора Jupyter Notebook. **Предпосылки**: Активизация ненужных сервисов запущенных по умолчанию. **Последствия**: Использвание модели проекта посторонними людьми. **Вариант решения**: -Ручное отключение сервисов с определением каждого что не помещает работоспособности программы. Сервисы показаны в виде списка при не знаниях что делает сервис можно прочитать документацию.	

Таблица 4.13 - Запросы на регистрацию риска

Запрос на регистрацию риска **Номер в журнале рисков: 10**	
ФИО автора запроса: Сапаргали Е.Е. **Роль на проекте**: data scientist **Наименование проекта**: ML **Фаза проекта**: Обучение модели	**Приоритет**: Средний **Дата запроса**: 14.02.2020 **Желаемая дата разрешения**: 16.02.2020
Описание риска: Вывод ошибки Вывод сообщении ложным срабатываниям **Предпосылки**: Ошибка начальных расчетов алгоритмов машинного обучения **Последствия**: Не получим результата от данных этого проекта **Варианты решения**: - Определение оптимиальных значении параметров классификатора Случайного Леса или Изолированного Леса; - Корректный ввод n_estimator, max_samples, contaminationParameter для пострения модели.	

Таблица 4.14 - Описание процедуры управления рисками проекта

Риск	**Наименование и описание рисков**	**Предлагаемое действие**	**Ответсвенный**	**Срок**
ML-1	Ненадежные наборы данных. Данные для анализа могут взяты из ненадежных и некомпетентных источников. Владелец: Сапаргали Е.Е.	-Изучение инженером надежности источника данных;	data-engineer	2.02.2020
		- Обсуждениях на коллегиальных дискуссиях об надежности дискуссиях и формирование списка надежных источников на;	big-data менеджер	4.02.2020
		-Брать с фиксированных, заранее одобренных источников предаставленные например, институтом.	data-engineer	5.02.2020
ML-2	Сбой программных средств. Сбой программных средств платформы Anaconda после запуска одновременного нескольких CSV. Владелец: Сапаргали Е.Е.	-Ограничить одновременный запуск нескольких CSV файлов;	разработчик	5.02.2020
		-Переносить данные из каталога в другую папку, чтобы программа не смогла его найти	разработчик	7.02.2020
ML-3	Злоупотребления правами. Злоупотребление правами сотрудником рангом по ниже в платформе Anaconda. Владелец: Сапаргали Е.Е.	- Определение прав для каждого сотрудника;	администратор	5.02.2020
		- Root права должны распределять только определенным людям.	администратор	5.02.2020

Продолжение таблицы 4.14

ML-4	Потеря электропитания. Потеря электропитания блока питания аппаратного средства. Владелец: Сапаргали Е.Е.	-Обеспечить использование резервных систем электропитания	администратор	6.02.2020
		-Делать резервное копирование и сохранение изменения программных кодов с этапа предобработки данных	разрабочтик	9.02.2020
ML-5	Смещение результатов. Смещение результатов в отношении определенных категорий данных в направлении не необходимых целям проекта. Владелец: Сапаргали Е.Е.	- Разрешение проблем, когда машина не может понять человеческую сущность и человеческие мысли;	big-data manager	9.02.2020
		- Считывать также грамматичекские ошибки;	data-engineer	11.02.2020
		- Считывать весьма аномальные цифры в ячейках.	data-engineer	11.02.2020
ML-6	Хищение носителей данных. Хищение носителей данных аппаратного средства. Владелец: Сапаргали Е.Е.	-Установка системы контроля безопасности в помещения, где хранится аппаратное средство;	безопасник	9.02.2020
		-Защитить от несанкционированного доступа при хищении:	безопасник	9.02.2020
ML-7	Затруднительное применение МО. Затруднительное применение сетей машинного обучения для анализа данных. Владелец: Сапаргали Е.Е.	-Создания классов меток для использования методов машинного обучения;	data-engineer	14.02.2020
		-Не брать большие наборы данных со характеристиками без столбцов с метками	data-engineer	11.02.2020

Продолжение таблицы 4.14

ML-8	Ухудшение состояния элементов. Ухудшение состояния элементов аппаратного средства (в том числе носителей данных) Владелец: Сапаргали Е.Е.	- Поставить сотрудника, который будет периодический проводить диагностику аппаратного средства;	администратор	9.02.2020
		- Поменять элемент аппаратного средства который не пригоден для работы.	администратор	9.02.2020
ML-9	Нелегальная обработка данных. Нелегальная обработка данных с встроенного текстового редактора. Владелец: Сапаргали Е.Е.	-Ручное отключение сервисов с определением каждого что не помещает работоспособности программы. Сервисы показаны в виде списка при не знаниях что делает сервис можно прочитать документацию	разработчик	10.02.2020
ML-10	Вывод ошибки. Вывод сообщении ложным срабатываниям. Владелец: Сапаргали Е.Е.	- Определение оптимиальных значении параметров классификатора Случайного Леса или Изолированного Леса.	data-scientist	16.02.2020
		- Корректный ввод n_estimator, max_samples, contaminationParameter для построения модели	data scientist	16.02.2020

4.3 Анализ рисков с инструментом CORAS

При построении диаграмм следует определить центральный элемент диаграммы, вокруг которого будет строиться система вывода. Таким элементом может несколько активов системы или угрозы информационной безопасности системы. Анализ данных машинном обучением представляют собой математическую модель нейронных сетей живых существ. Работа

нейронной сети осуществляется на компьютере под управлением дистрибутива данных Python – Anaconda. Наборы данных из аппаратного средства загружается в Anaconda, для дальнейшего использования в машинном обучении.

На рисунке 4.2 представлена диаграмма, иллюстрирующая влияние некомпетентности персонала и администратора, а также нарушителя и самой системы на активы, интересующие владельцев системы. В диаграмме приняты следующие элементы: — угроза. Источники угрозы которые привели действиями к инцидентам безопасности системы; — сценарии угроз. Последовательность действий, которая реализует соответствующую атаку на систему; — нежелательный инцидент. Ситуация к которой привела реализация сценария угрозы или другой инцидент; — активы. Целевые элементы системы, оценку которых снижает реализация сценариев и инцидентов. Данная диаграмма имеет элементы «нарушитель», «персонал», «система», «администраторов» из множества угроз. Есть сценарии угроз — НСД в помещение, отсутствие периодического обслуживания, сбой в системе, злоупотребление правами сотрудником рангом по ниже, доступ информации через удаленный доступ. Сценарии приводят к нежелательным инцидентам, множество которых составляет — хищение информации, ухудшение состояние элементов аппаратного средства, сбой программных средств программ Анаконда, некорректный результат анализа данных, вывод сообщении ложным срабатыванием. Множество активов определено как Анаконда, аппаратные средства, нейронные сети, набор данных.

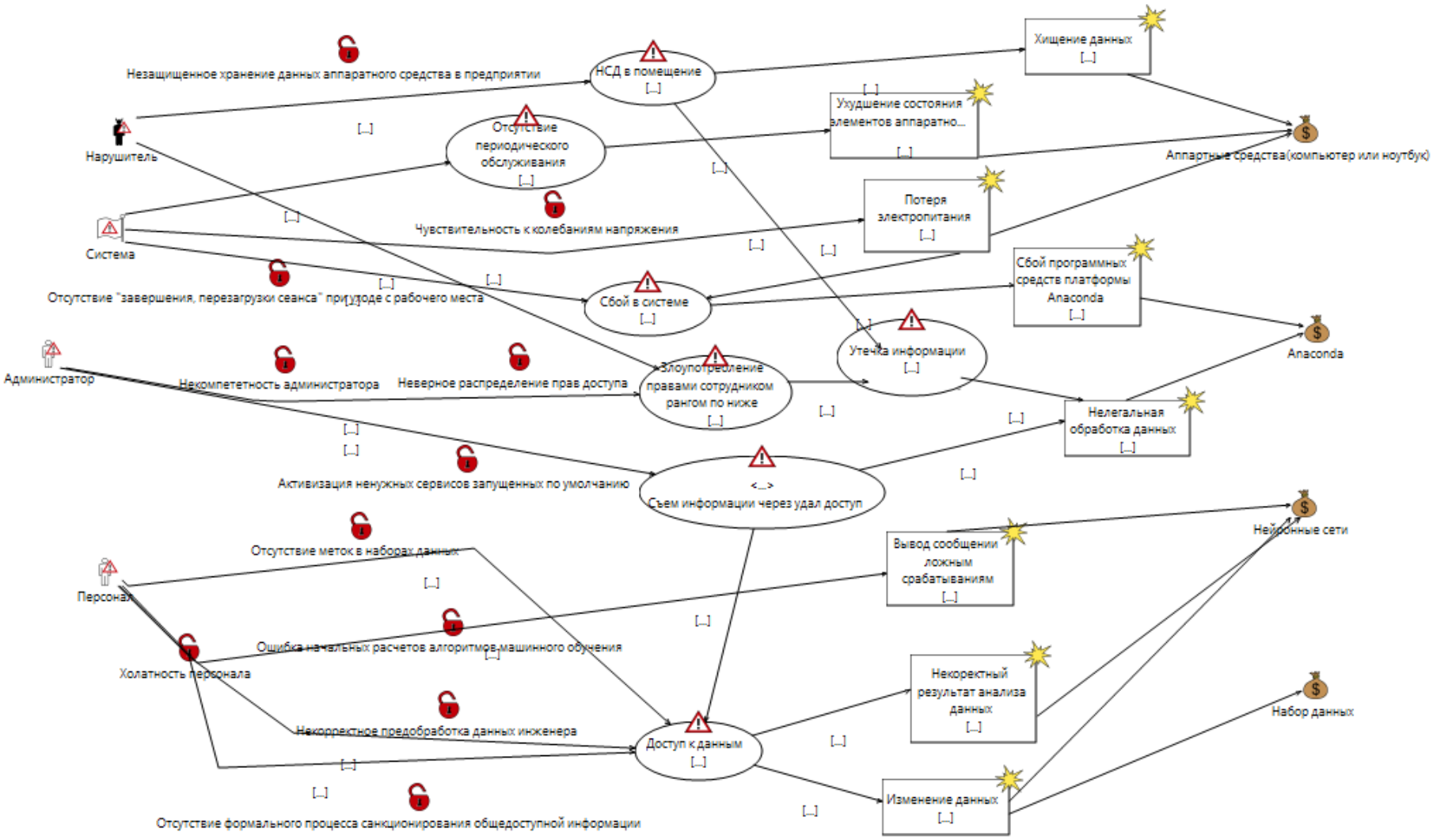


Рисунок 4.2 – Модель угроз

На полученную в предыдущем шаге модель наносим вероятность осуществления сценария нежелательного инцидента (высокий, средний, низкий). В результате получаем полную модель угроз. Для моего проекта эта модель угроз представлена на рисунке 4.3.

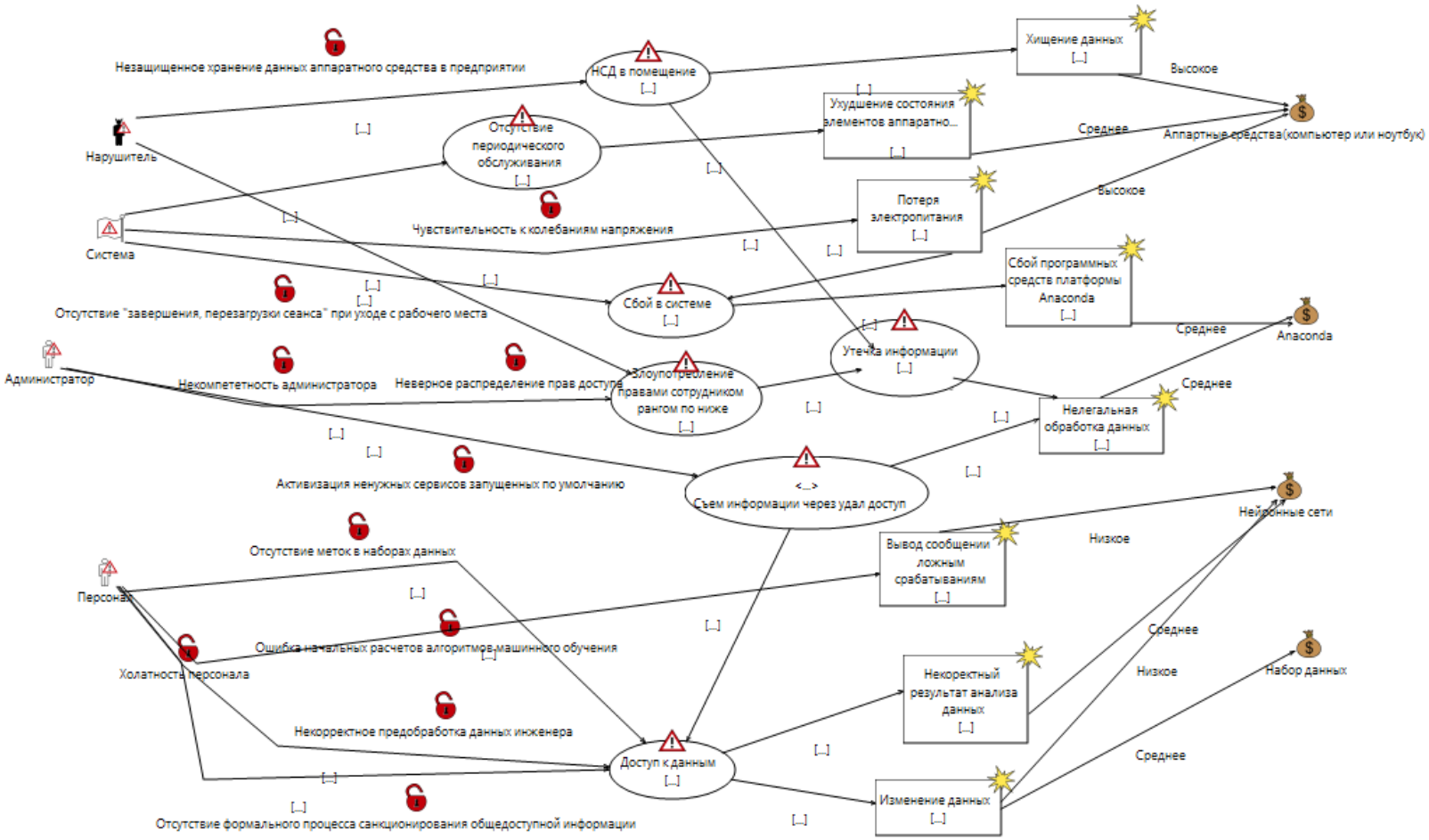


Рисунок 4.3 – Модель угроз с вероятностными характеристиками

На диаграмме угроз для каждой уязвимости ставим противодействие.

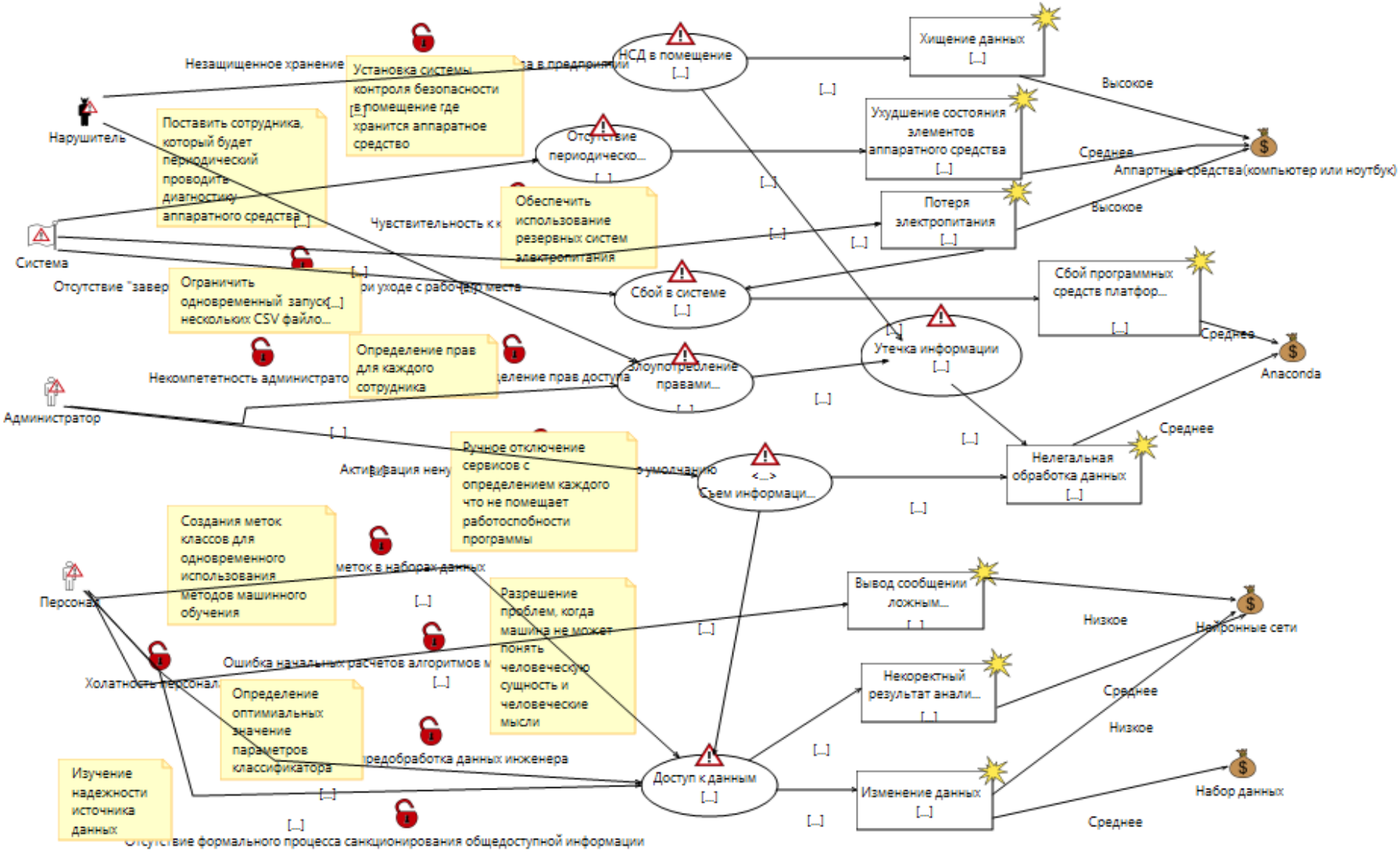


Рисунок 4.4 – Диаграмма угроз после добавления воздействия

Теперь по каждому риску для каждого актива определяем последствия в случае осуществления этого риска.

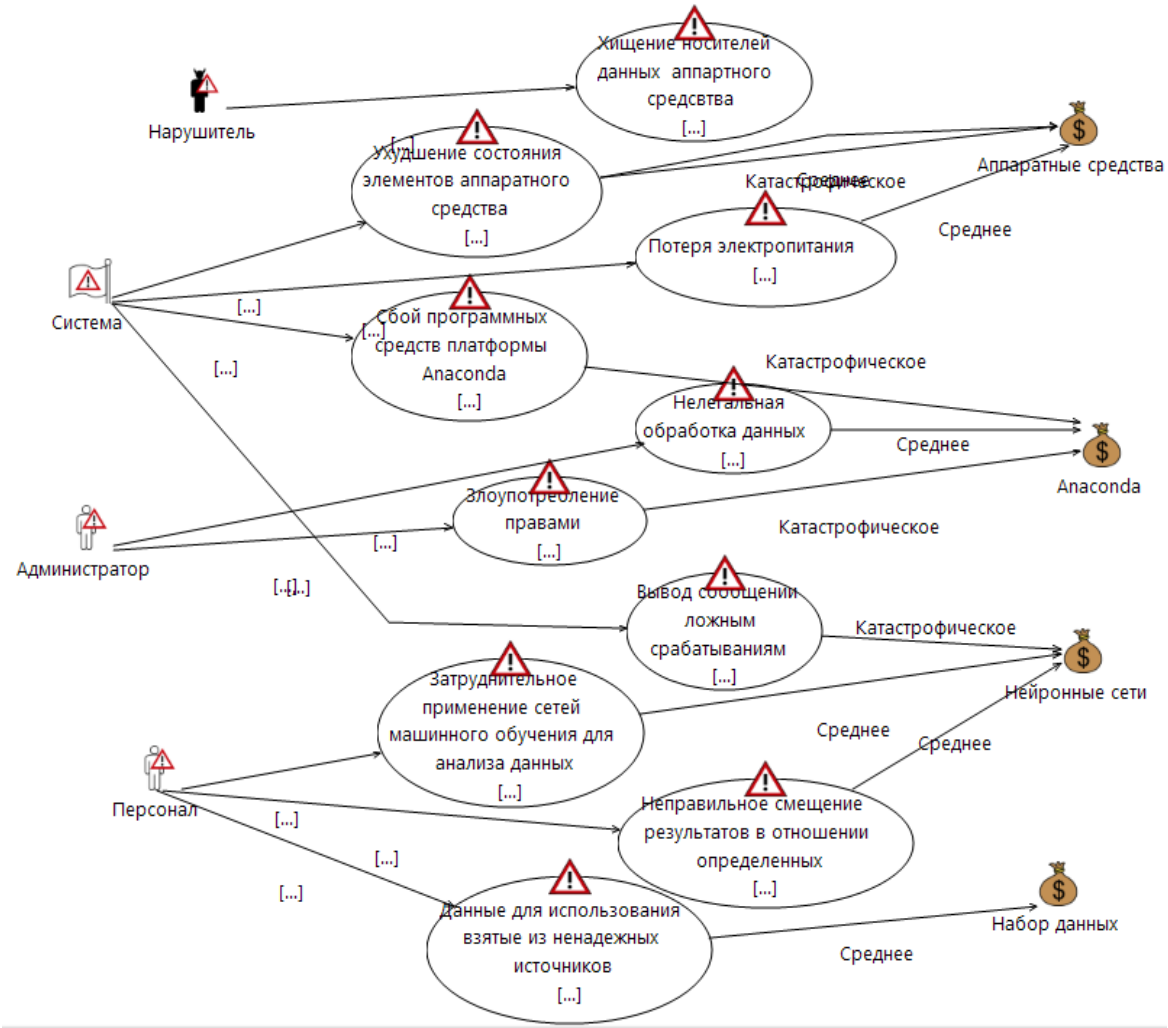


Рисунок 4.5 – Диаграмма рисков с характеристикой последствий осуществления угрозы

Внося поправки в соответствии с матрицей, получаем диаграмму неприемлемых рисков.

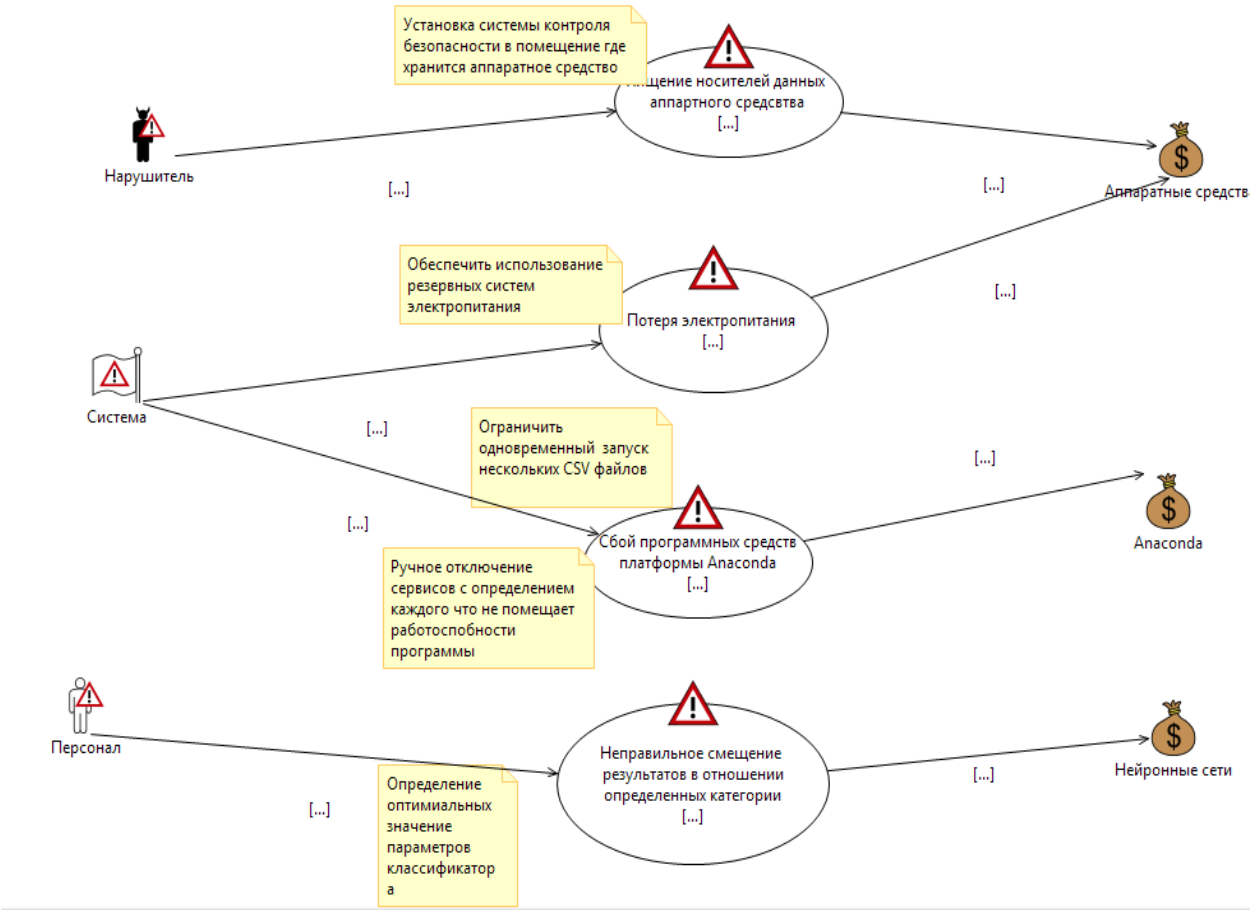


Рисунок 4.6 – Диаграмма неприемлемых рисков

На основании диаграммы неприемлемых рисков (рисунок 4.6) можно предложить следующие противодействия в порядке влияния на риски:

— Установка системы контроля безопасности в помещение где хранится аппаратное средство;

— Обеспечить использование резервных систем электропитания;

— Ограничить одновременный запуск нескольких CSV файлов;

— Ручное отключение сервисов с определением каждого что не помещает работоспособности программы;

— Определение оптимиальных значении параметров классификатора Случайного Леса или Изолированного Леса.

Вывод

В данном разделе было определены характеристики рисков(проектных) дипломного проекта платформы Anaconda и ее ресурсов. Были рассмотрены теоретические основы проектных рисков, показан способ оценки матрицой вероятности и воздействия, освещена методика расчета оценки проектных рисков. Были построены диаграммы с использованием программы Coras.

Для управления рисками использовали процессы, связанные с идентификацией, анализом рисков и принятием решений, которые включают максимизацию положительных и минимизацию отрицательных последствий наступления рисковых событий.

Количество рисков огромно, для успешного внедрения информационных систем выявил все риски и определил наиболее опасные. Именно на них обратил наиболее пристальное внимание.

Основными стратегиями по информационными рисками использовал следующие: принятие риска; предотвращение риска; снижение возможного ущерба от риска; предотвращение риска и снижение возможного ущерба от него.

5 Безопасность жизнедеятельности

5.1 Анализ потенциально опасных и вредных факторов офисного помещения, воздействующих на персонал

Задание: в первом разделе провести анализ наиболее характерных опасных и вредных факторов для работающего в офисе (например, электромагнитное излучение от компьютера, влияние компьютера на работающего – сидячая, длительная поза и др., шум, запыленность, химические вещества - озон от лазерного принтера и др., требования по микроклимату и др.), описать их негативное действие на организм человека и привести допустимые параметры в соответствии с нормами.

С каждым годом улучшаются показатели характеристик выпускаемых компьютеров. Несмотря на улучшенные качества техники и условий труда пользователей электронно-вычислительных машин и видео-дисплейных терминалов (ВДТ), есть воздействие от ПЭВМ и ВДТ на работников.

Основные воздействия вредных и опасных факторов на работников в офисных помещениях при работе за компьютерами происходит по следующим причинам:

- несоответствие размеров помещения количеству работников;
- не соответствие рабочих мест (планирование, размещение) требуемым нормам;
- непрерывная работа сотрудников офиса за компьютером в течении всей рабочей смены.

Работа с персональным компьютером - это воспроизведение визуальной информации на дисплее, которая должна быстро и точно восприниматься пользователем.

Основным фактором, влияющим на производительность труда людей, работающих с ПЭВМ и ВДТ, являются комфортные и безопасные условия труда.

Условия труда пользователя, работающего с персональным компьютетером, определяются:

- организацией и планированием рабочего места;
- условиями производственной среды (освещение, микроклимат (температура, влажность), шум, электромагнитные и электростатические поля);
- характеристиками информационного взаимодействия человека и персональных компьютеров (ПК).

При выполнении работ на персональном компьютере могут иметь место следующие факторы:

- повышенная или пониженная температура воздуха рабочей зоны;
- недостаточная искусственная освещенность рабочей зоны;
- отсутствие или недостаток естественного света;
- зрительное напряжение;
- повышенная температура поверхностей ПК;
- повышенная или пониженная влажность воздуха;

- выделение в воздухе рабочей зоны ряда химических веществ;
- повышенный или пониженный уровень отрицательных и положительных аэроионов;
- повышенное значение напряжения в электрической цепи, замыкание;
- повышенный уровень статического электричества;
- повышенный уровень электромагнитных излучений;
- повышенная напряженность электрического поля;
- повышенная яркость света;
- повышенная контрастность;
- прямая и отраженная блескость;
- монотонность трудового процесса;
- нервно-эмоциональные перегрузки.[8]

Работа на ПК сопровождается постоянным и значительным напряжением функций зрительного анализатора. Одной из основных особенностей является иной принцип чтения информации, при работе на ПК оператор считывает текст, почти не наклоняя голову, глаза смотрят прямо или почти прямо вперед, текст (источник - люминесцирующее вещество экрана) формируется по другую сторону экрана, поэтому пользователь не считывает отраженный текст, а смотрит непосредственно на источник света, что вынуждает глаза и орган зрения в целом работать в несвойственном ему стрессовом режиме длительное время.

Расстройство органов зрения резко увеличивается при работе более четырех часов в день.

Всемирная организация здравоохранения (ВОЗ) ввела понятие “компьютерный зрительный синдром” (КЗС), типовыми симптомами которого являются жжение в глазах, покраснение век, чувство инородного тела или песка под веками, боли в области глазниц и лба, затуманивание зрения, замедленная пере фокусировка с ближних объектов на дальние.

Нервно-эмоциональное напряжение при работе на ПК возникает вследствие дефицита времени, большого объема и плотности информации, особенностей диалогового режима общения человека и ПК, ответственности за безошибочность информации. Продолжительная работа на дисплее, особенно в диалоговом режиме, может привести к нервно-эмоциональному перенапряжению, нарушению сна, ухудшению общего состояния, снижению концентрации внимания и работоспособности, хронической головной боли, повышенной возбудимости нервной системы, депрессии.

Повышенные статические и динамические нагрузки у пользователей ПК приводят к боли в спине, шейном отделе позвоночника и руках. Из всех недомоганий, обусловленных работой на компьютерах, чаще встречаются те, которые связаны с использованием клавиатуры. В период выполнения операций ввода данных количество мелких стереотипных движений кистей и пальцев рук за смену может превысить 60 тыс., что в соответствии с гигиенической классификацией труда относится к категории вредных и опасных. Поскольку каждое нажатие на клавишу сопряжено с сокращением

мышц, сухожилия непрерывно скользят вдоль костей и соприкасаются с тканями, вследствие чего могут развиться болезненные воспалительные процессы. Воспалительные процессы тканей сухожилий получили общее название “травма повторяющихся нагрузок”.

Большинство работающих со временем начинают жаловаться на боли в шее и спине. Эти недомогания накапливаются постепенно и получили название “синдром длительных статических нагрузок” (СДСН). Другой причиной возникновения СДСН может быть длительное пребывание в положении сидя, которое приводит к сильному перенапряжению мышц спины и ног, в результате чего возникают боли и неприятные ощущения в нижней части спины. Основной причиной перенапряжения мышц спины и ног являются нерациональная высота рабочей поверхности стола и сидения, отсутствие опорной спинки и подлокотников, неудобное размещение монитора, клавиатуры и документов, отсутствие подставки для ног.

Для существенного уменьшения боли и неприятных ощущений, возникающих у пользователей ПК, необходимы достаточные перерывы в работе и эргономические усовершенствования, в том числе оборудование рабочего места для исключения неудобных поз и длительных напряжений.

К факторам ухудшающих состояние здоровья пользователей компьютеров, также относятся эргономические параметры расположения экрана монитора (дисплея), параметры мебели и характеристики помещения, где расположена компьютеры.

Психофизиологические вредные и опасные факторы: напряжение зрения и внимания; интеллектуальные, эмоциональные и длительные статические нагрузки; монотонность труда; большой объем информации, обрабатываемый в единицу времени; нерациональная организация рабочего места.

Типичными ощущениями, которые испытывают к концу рабочего дня пользователи ПК, являются: переутомление глаз, головная боль, тянущие боли в мышцах шеи, рук и спины, снижение концентрации внимания.

Уже в первые годы компьютеризации было отмечено специфическое зрительное утомление у пользователей ПК, получившее общее название «компьютерный зрительный синдром». Одной из причин служит, что сформировавшаяся за миллионы лет эволюции зрительная система человека приспособлена для восприятия объектов в отраженном свете (печатные тексты, рисунки и т.п.), а не для работы за дисплеем ПК. Изображение на дисплее принципиально отличается от привычных глазу объектов наблюдения - оно светится, мерцает, состоит из дискретных точек, а цветное компьютерное изображение не соответствует естественным цветам. Не только особенности изображения на экране вызывают зрительное утомление, большую нагрузку орган зрения испытывает при вводе информации, так как пользователь вынужден часто переводить взгляд с экрана на текст и клавиатуру, находящиеся на разном расстоянии и по-разному освещенные. Зрительное утомление проявляется жалобами на затуманивание зрения,

трудности при переносе взгляда с ближних предметов на дальние и с дальних на ближние, кажущиеся изменения окраски предметов, их двоение, чувство жжения, «песка» в глазах, покраснение век, боли при движении глаз.

Длительная и интенсивная работа на компьютере может стать источником тяжелых профессиональных заболеваний, таких, как травма повторяющихся нагрузок (ТПН), представляющая собой постепенно накапливающиеся недомогания, переходящие в заболевания нервов, мышц и сухожилий руки.

К профессиональным заболеваниям, связанным с ТПН, относятся:

- тендовагинит - воспаление сухожилий кисти, запястья, плеча;

- тендосиновит - воспаление синовиальной оболочки сухожильного основания кисти и запястья;

- синдром запястного канала (СЗК) – вызывается ущемлением срединного нерва в запястном канале. Накапливающаяся травма вызывает образование продуктов распада в области запястного канала, в результате чего вначале возникает отек, а затем СЗК.

Появляются жалобы на жгучую боль и покалывание в запястье, ладони, а также пальцах, кроме мизинца. Наблюдается болезненность и онемение, ослабление мышц, обеспечивающих движение большого пальца.

Эти заболевания обычно наступают в результате непрерывной работы на неправильно организованном рабочем месте.

Механизм нарушений, происходящих в организме под влиянием электромагнитных полей, обусловлен их специфическим (нетепловым) и тепловым действием.

Специфическое воздействие ЭМП отражает биохимические изменения, происходящие в клетках и тканях. Наиболее чувствительными являются центральная и сердечно-сосудистая системы. Возможны отклонения со стороны эндокринной системы.

В начальном периоде воздействия может повышаться возбудимость нервной системы, проявляющаяся раздражительностью, нарушением сна, эмоциональной неустойчивостью. В последующем развиваются астенические состояния, т.е. физическая и нервно-психическая слабость. Поэтому для хронического воздействия ЭМП характерны: головная боль, утомляемость, ухудшение самочувствия, гипотония (снижение артериального давления), брадикардия, боли в сердце. Указанные симптомы могут быть выражены в разной степени.

Тепловое воздействие ЭМП характеризуется повышением температуры тела, локальным избирательным нагревом клеток, тканей и органов вследствие перехода ЭМП в тепловую энергию. Интенсивность нагрева зависит от количества поглощенной энергии и скорости оттока тепла от облучаемых участков тела. Отток тепла затруднен в органах и тканях с плохим кровоснабжением. К ним в первую очередь относится хрусталик глаза, вследствие чего возможно развитие катаракты. Тепловому воздействию ЭМП подвергаются также паренхиматозные органы (печень,

поджелудочная железа) и полые органы, содержащие жидкость (мочевой пузырь, желудок). Нагревание их может вызвать обострение хронических заболеваний.

Офисные работники ежедневно испытывают на себе влияние множества негативных факторов, таких, как шум принтеров, выделения озона от работающей копировальной техники, сухой кондиционированный воздух, излучение монитора ПК, стрессы, малоподвижный, сидячий режим работы и др. Все это может постепенно привести к серьезным проблемам со здоровьем.

Кроме вредных факторов, сопряженных с работой оргтехники в офисах, элементарной проблемой является пыль офисных помещений. В зависимости от возраста здания пыль может содержать до 80% вредных веществ, среди которых угарный газ, затягиваемый с улицы, аллергены и возбудители заболеваний. Наибольшее количество пыли скапливается в вентиляционных системах, и даже если в офисе все чисто, воздух помещения может быть непригоден для сотрудников. Система воздуховодов загрязняется уже через год после начала эксплуатации. В результате развиваются микроорганизмы, которые попадают по вентиляционным системам во все помещения здания. Единственный верный способ борьбы с пылью — это качественные влажные уборки, регулярное проветривание, очистка и обеззараживание системы вентиляции. Кроме того, от пыли успешно можно применять ионизаторы, он генерирует отрицательно заряженные ионы, насыщая ими воздух, а пыль оседает на специальных алюминиевых пластинах ионизатора с которых она легко смывается водой. Таким образом можно предотвращать вредное влияние пыли на работающих в офисе. Офисная мебель тоже может быть опасной для здоровья сотрудников - ДСП и декоративная фанера способны выделять фенол, формальдегид. Поэтому обставлять офисные помещения желательно мебелью из натуральных материалов. Выделять вредные вещества способны и строительные материалы, применяемые при строительстве и отделке помещений. Наличие в воздухе офисных помещений паров фенола, формальдегида, используемых в производстве мебели, радиоактивных веществ (радона) повышает вероятность развития онкологических заболеваний. Источником радона может быть бетон, кирпич и другие строительные материалы, содержащие естественные радионуклиды.

В наше время понятия «работник офиса» и «малоподвижный образ жизни» практически неразделимы. Малоподвижный образ жизни вызван отсутствием регулярных физических нагрузок и сидячей работой. Последствия такого времени препровождения - слабый отток лимфы, атония кишечника, утомляемость, головная боль и множество других проблем со здоровьем. Причиной недомоганий могут быть так называемые токсины усталости, которые возникают в организме из-за малоподвижности офисного труда. Незаметно скапливаясь в организме, они вызывают многие заболевания, в том числе позвоночника, сердца, сосудов, снижение сексуальной активности и др. Поэтому, чтобы продуктивно работать и

избежать развития заболеваний, офисным работникам с малоподвижным трудом необходимо в перерывах делать физические упражнения восстанавливающие ток лимфы, крови, работу суставов в полном объеме.[9]

Допустимые параметры в соответствии с нормами. План размещения рабочих мест в офисном помещении с ВДТ и ПК должны учитывать расстояния между рабочими столами с видеомониторами (в направлении тыла поверхности одного видеомонитора и экрана другого видеомонитора), которое должно быть не менее 2,0 м, а расстояние между боковыми поверхностями видеомониторов - не менее 1,2 м.

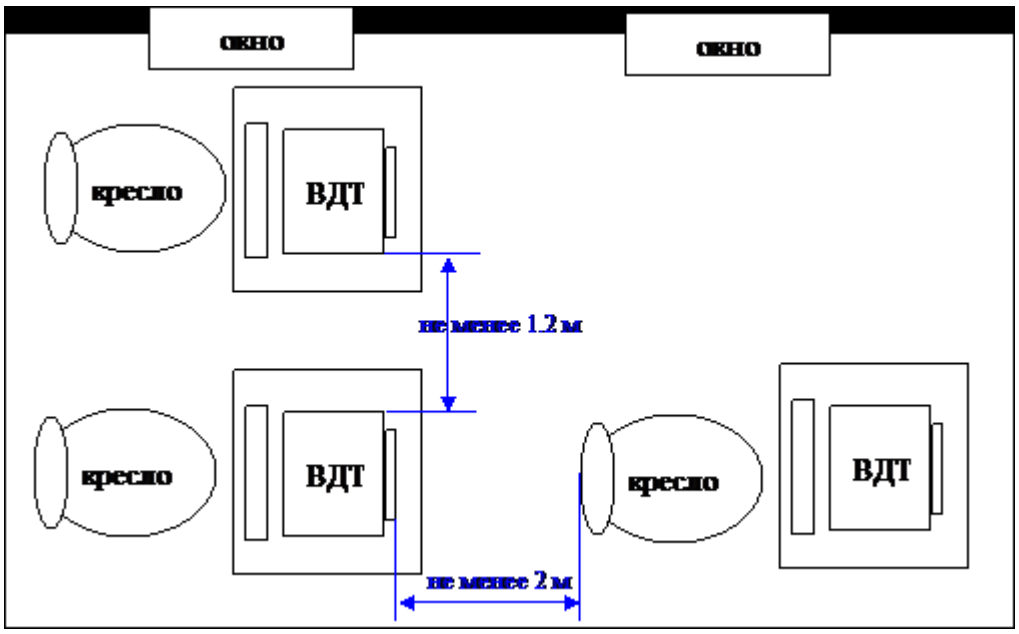


Рисунок 5.1 - Предлагаемый план размещения рабочих мест для офисного помещения на 3 (трех) работников

Выписка из Санитарных правил «Санитарно-эпидемиологические требования к условиям работы с источниками физических факторов (компьютеры и видеотерминалы), оказывающих воздействие на человека». Утверждены приказом Министра национальной экономики Республики Казахстан от 21 января 2015 года № 38.

Помещения для работы с ПК, ПлПК, ноутбуками и ВТ не размещаются в подвальных и цокольных помещениях. Рабочие места с ПК, ПлПК, ноутбуками и ВТ не размещаются в местах, где расположены силовые кабели, высоковольтные трансформаторы, технологическое оборудование.

Площадь на одно рабочее место пользователей ПК и ВТ на базе электронно-лучевой трубки (далее - ЭЛТ) составляет не менее 6 квадратных метров (далее - м^2) при рядном расположении, при центральном и периметра льном расположении - 4 м^2, при использовании ВТ на базе плоских дискретных экранов (жидкокристаллические, плазменные) при любом расположении - 4 м^2. Площадь на одно рабочее место пользователей ПлПК, ноутбуков 2,5 м^2.

Для отделки помещений применяют материалы, допускающие уборку влажным способом с применением моющих средств.

Поверхность пола в помещениях, где оборудуются ПК, ПлПК, ноутбуки и ВТ, выполняется без выбоин и щелей, из материалов, обладающих антистатическими свойствами. Помещения с использованием ПК, ПлПК, ноутбуками и ВТ, мебель и оборудование содержатся в порядке и чистоте. Дефекты в отделке помещения и поломки оборудования, мебели подлежат своевременному ремонту и замене.

Помещения, где размещаются ПК и ВТ, оборудуются защитным заземлением.

Расстановка компьютеров (ПК, планшетные персональные компьютеры, ноутбуки) используется одним из трех 3-х вариантов: периметральная, рядные (2-3-рядная), центральная.

При периметральной расстановке, расстояние между стеной с оконными проемами и столами 0,5 метров (далее - м), стеной и столами - 0,4 м.

При рядной расстановке расстояние между тылом поверхности одного видеомонитора и экраном другого - не менее 2 м, между боковыми поверхностями видеомониторов не менее 1,2 м, при двух-трехрядной расстановке одноместных столов с компьютерами расстояния в каждом ряду между боковыми поверхностями столов не менее 0,5 м.

При центральной расстановке рабочие столы с компьютерами устанавливаются в центре, в два ряда без разрыва и экраны видеомониторов обращены в противоположные стороны, располагаясь в шахматном порядке, или напротив друг друга тыльными сторонами мониторов, при этом расстояние между тылом поверхности одного видеомонитора и экраном другого - не менее 2 м.

Размеры рабочей поверхности:

- высота рабочей поверхности стола (от пола) регулируется в пределах 640- 800 миллиметров (далее - мм);

- ширину рабочей поверхности стола 800, 1000, 1200 и 1400 мм;

рабочий стол имеет пространство для ног высотой не менее 580 мм, шириной

- не менее 500 мм, глубиной - не менее 450 мм.

Экран видеомонитора находится от глаз пользователя на расстоянии 600-700 мм, но не ближе 500 мм с учетом размеров алфавитно-цифровых знаков и символов.

В помещениях, где для занятия с ПК, ПлПК, ноутбуками и ВТ оборудуются одноместными столами, предусматривают следующую конструкцию одноместного стола для работы с ПК, ПлПК, ноутбуков и ВТ:

- две раздельные поверхности: одну горизонтальную для размещения ПК с плавной регулировкой по высоте в пределах 520 - 760 мм и вторую подвижную для клавиатуры с регулировкой по высоте соответственно горизонтальной рабочей поверхности;

- ширина поверхностей для ПК, ПлПК, ноутбуков и ВТ клавиатуры составляет не менее 750 мм, глубина - не менее 550 мм;

- ширина пространства для ног не менее 500 мм, глубина не менее 450 мм, а высоту принимать в соответствие с ростом;

- увеличение ширины поверхностей до 1200 мм при оснащении рабочего места принтером.

Продолжительность непосредственной работы с ВТ и ПК, ПлПК и ноутбуками рекомендуется не более двух часов. В период работы проводятся профилактические мероприятия: упражнения для глаз через каждый 20-25 минут и физкультурная пауза через 45 минут во время перерыва.

Одновременное использование одного ВТ, ПК, ПлПК, ноутбуков двумя и более людьми, независимо от возраста не рекомендуется.

Не используются ПК, ВТ, ПлПК, ноутбуки без наличия документов, подтверждающих их качество и безопасность.

В качестве источников света при искусственном освещении используются люминесцентные лампы. В светильниках местного освещения допускается применение ламп накаливания, в том числе энергосберегающие.

Для предупреждения бликов на экране монитора, оконные проемы оборудуются защитными устройствами, не пропускающими дневной свет.

Минимизация воздействия вредных и опасных факторов на работников в офисных помещениях при работе за компьютерами должна происходит с учитом требований норм санитарных правил по основным производственным факторам (освещённость, шум, электромагнитные излучения, микроклимат) указаных в таблицах 5.1 - 5.4:

- Таблица 5.1. Допустимые параметры освещённости для помещений;
- Таблица 5.2. Допустимые уровни звукового давления в октавных полосах частот и уровня звука, создаваемого компьютерами и видеотерминалами;
- Таблица 5.3. Допустимые значения уровней неионизирующих электромагнитных излучений;
- Таблица 5.4 Допустимые параметры микроклимата для помещений.

Таблица 5.1 - Допустимые параметры освещенность на поверхности рабочего стола при работе за ПК

№№	Освещенность на поверхности рабочего стола составляет:	Допустимое значение
1	при комбинированном освещении не менее, от общей системы	300 люкс
2	от местной системы	500 люкс
3	при наличии только общей системы освещения	400 люкс
4	освещенность поверхности экрана не более	200 люкс
Освещение выполняется таким образом, чтобы обеспечить отсутствие бликов на поверхности экрана		

Таблица 5.2 - Допустимые уровни звукового давления в октавных полосах частот и уровня звука, создаваемого компьютерами и видеотерминалами

Уровни звукового давления (далее - дБ) в октавных полосах (далее - ОП) среднегеометрическими частотами Герц (далее - Гц) не более									Уровни звука в дБА не более
31,5 Гц	63 Гц	125 Гц	250 Гц	500 Гц	1000 Гц	2000 Гц	4000 Гц	8000 Гц	
86 дБ	71 дБ	61 дБ	54 дБ	49 дБ	45 дБ	42 дБ	40 дБ	38 дБ	50

Таблица 5.3 - Допустимые значения уровней неионизирующих электромагнитных излучений

Наименование параметров	Части ПК, ВТ	Контрольное расстояние, сантиметров (далее - см)	Допустимое значение
Напряженность электростатического поля для профессиональных пользователей	Монитор Клавиатура мышь	На уровне головы, На уровне рук - 1,0	20 килоВольт на метр (далее - кВ/м)
Напряженность электростатического поля на рабочих местах: - детских дошкольных оргциях; - учебных заведений; - компьютерных клубах	Монитор Клавиатура мышь	На уровне головы, На уровне рук - 1,0	20 кВ/м 15 кВ/м 15 кВ/м 15 кВ/м
Напряженность электрического поля вокруг ПК, ВТ: в диапазоне частот 5 - 2000 Герц (далее - Гц): в диапазоне частот 2 - 400 кГц:	Монитор Клавиатура мышь	На уровне головы На уровне рук - 1,0 На уровне головы На уровне рук - 1,0	25 Вольт на метр (далее - В/м) 2,5 В/м
Плотность магнитного потока вокруг ПК, ВТ: в диапазоне частот 5 -2000 Гц: в диапазоне частот 2-400 кГц:	Монитор Клавиатура мышь	На уровне головы, На уровне рук - 1,0 На уровне головы, На уровне рук - 1,0	250 наноТесла (далее - нТл) 25 нТл
Поверхностный электростатический потенциал от монитора, не более (при сертификационных испытаниях)	Монитор	Между дисплеем и установленной в 10 см от него заземленной измерительной пластиной	500 Вольт

Таблица 5.4 - Допустимые параметры микроклимата для помещений

Температура, C^{o}	Относительная влажность, не более, %	Скорость движения воздуха, м/с
1	2	3
18	66	<0,1
19	62	<0,1
20	58	<0,1
21	55	<0,1

5.2 Расчетные показатели по обеспечению комфортных условий труда для работающих в офисных помещениях

Задание. Привести два расчетных вопроса, обеспечивающие комфортные условия труда. Например, расчет кондиционирования, расчет шума от нескольких компьютеров шумоглушению, расчет освещенности (офиса), расчет вентиляции, расчет по электробезопасности (защитное заземление, зануление), и т.п. Выбрать два расчетных вопроса самостоятельно.

Согласно выданному заданию проведем расчеты офисного помещения по освещенности и кондиционированию.

При планировании освещения, в первую очередь определяем соответствующую нормам целевую освещенность и посчитать общий световой поток, который должны давать светильники в помещении. Согласно требованиям СП РК 2.04-104-2012 «Естественное и искусственное освещение», берем за основу требование по офисным помещениям с компьютерами – 400 лк.

Выписка из Санитарных правил «Санитарно-эпидемиологические требования к условиям работы с источниками физических факторов (компьютеры и видеотерминалы), оказывающих воздействие на человека». Утверждены приказом Министра национальной экономики Республики Казахстан от 21 января 2015 года № 38.[9]

Освещенность на поверхности рабочего стола составляет: при комбинированном освещении не менее 300 люкс (далее - лк) от общей системы, 500 лк от местной системы; при наличии только общей системы освещения - 400 лк. Освещение выполняется таким образом, чтобы обеспечить отсутствие бликов на поверхности экрана. Освещенность поверхности экрана не более 200 лк.

$$СП\ (световой\ поток) = А \times Б \times В \qquad (1)$$

где А - нормативное значение освещенности помещения;
Б - площадь помещения (комнаты) в м.кв.;

В - коэффициент высоты потолка (до 2,7 м - 1,0; 2,7-3,0 м - 1,2; 3,0-3,5 м - 1,5; 3,5-4,0 - 2,0).

Получившееся значение - это общий световой поток, необходимый на данное помещение. Теперь легко определить количество выбранных осветительных приборов.

Офисное помещение 5×5 м. В помещении требуется установить светодиодные светильники, которые имеют световой поток 4080 лм мощностью 39 Вт.

Задача: найти количество светильников для обеспечения освещенности 400 лк.

Количество светильников определяется по формуле 2:

$$N=S*Ko*P*Kкн/Uф \quad (2)$$

где S – площадь помещения;

Ко – коэффициент освещенности, Ко=Фл/Е, не путайте с коэффициентом одновременности;

Фл – световой поток светильника, лм;

Е – требуемая освещенность помещения, лк;

Р – мощность светильника, Вт; Uф – фазное напряжение;

Здесь очень важно использовать напряжение 230 В согласно ГОСТ Р 50.571-2017, а не 220 В, как это было ранее.

Ккн – постоянная Кноринга для административных помещений; Ккн=0,095.

Тогда, N=5*5*(4080/400)*39*0,095/230=4,1

Вывод:

для обеспечения требуемой освещенности необходимо 4 светильника.

Проверочный расчет в программе Dialux:

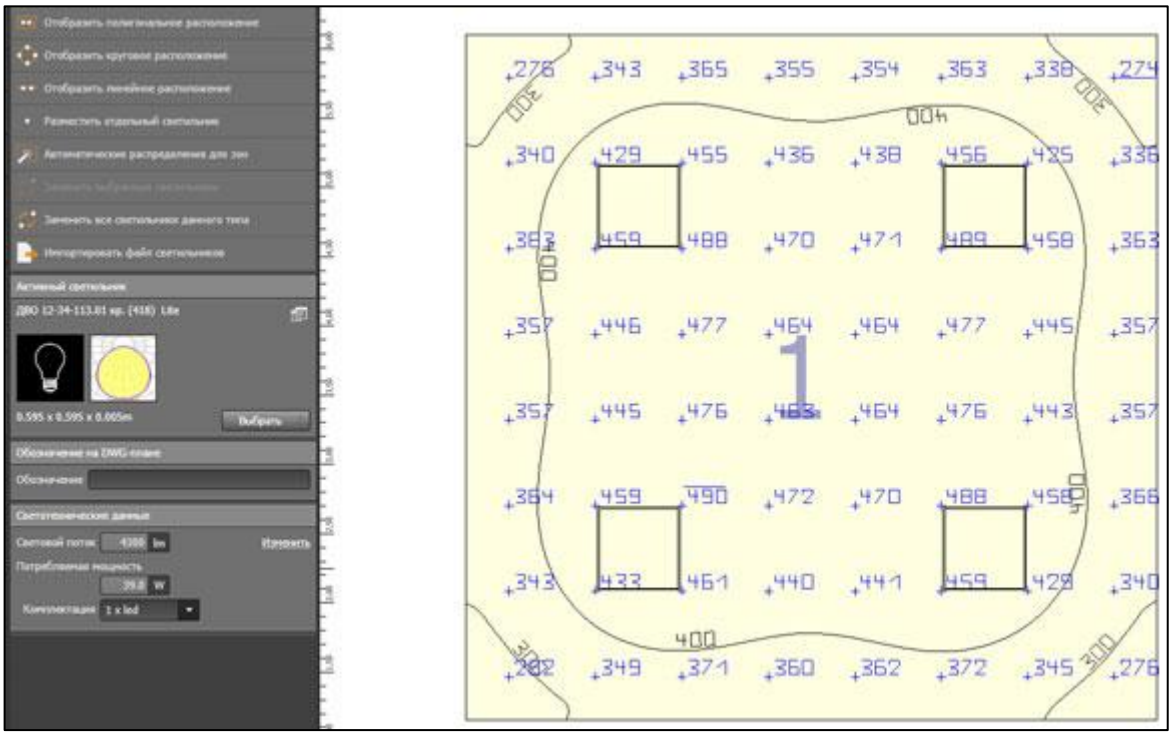


Рисунок 5.2 - Расчет светильников в Dialux

Проверочный расчет в программе:

A	B	H	H1	Фл	N	U	E	F	K	КПД
5	5	3	0,8	4080	1	1,2	400	3,13	0,7	1
Количество светильников, Nсв.:				4,2						

Рисунок 5.3 – Расчет необходимого количества светильников при заданной освещенности

Основной параметр, на который ориентируются при выборе кондиционера – мощность. Кондиционер с недостаточной мощностью не сможет выполнять свои задачи. Работу кондеционера в помещении регулируем с учитом требований норм санитарных правил по микроклимату указаных в таблице 5.4.

Перед тем, как рассчитать мощность кондиционера для помещения, необходимо подсчитать основные параметры, влияющие на производительность устройства. Существует 5 основных факторов, по которым подбирается мощность кондиционера:

- площадь. Самый важный параметр, от которого следует отталкиваться. На каждые 10 м.кв. площади помещения необходим 1 кВт мощности кондиционера. В противном случае, производительности устройства не будет хватать на покрытие всей территории помещения.

- высота потолков. Объемы пространства в помещении также оказывают немалое влияние на производительность кондиционера. Если высота потолков больше 3-х метров, то лучше предусмотреть запас по мощности (холодопроизводительности).

- количество человек, постоянно находящихся в помещении. Человеческое тело выделяет 100 Вт тепла в спокойном состоянии и 200 при физической активности. Например, если в гостиной постоянно находятся 2 человека, понадобится кондиционер на 200 Вт мощнее.

- размер и количество оконных проемов. Сквозь застекленные поверхности в помещение попадают солнечные лучи, они нагревают помещение. Перед тем, как рассчитать, какой кондиционер нужен для помещения, нужно учесть количество и площадь окон на солнечных сторонах.

- на каком этаже расположено помещение. На последних этажах, прямо под крышей, температура поднимается сильнее.

Стандартная формула 3 расчета подходящей мощности:

$$Q = Q1 + Q2 + Q3 \quad (3)$$

где Q1 - это мощность кондиционера для пустого помещения с учетом окон. Она рассчитывается по формуле Q1 =S x h x q/ 1000. Здесь S - площадь помещения, h - высота потолков, q - коэффициент освещенности помещения. Коэффициент освещенности принимается q = 30 для затененных помещений, 35 - для средней освещенности помещений, 40 - для хорошо освещенных помещений;

Q2 - сумма теплопритоков, идущих от человеческих тел, 0,1 - 0,3 кВт на человека;

Q3 - сумма теплопритоков, идущих от бытовой техники, 0,2 - 0,5 кВт на единицу техники.

Для подсчета общей мощности слит-системы складывают показатели основных факторов:

Теплопритоки от бытовой техники. Средний показатель рассчитывается в размере 30% от потребляемой прибором мощности. Например, компьютер мощностью в 0,9 кВт выделяет 0,3 кВт и т.д.

После сложения всех показателей получается средняя мощность, необходимая кондиционеру в конкретном помещении.[10]

Маркировка от производителей

Сплит-система одной и той же модели производится на различную площадь (соответственно разной мощности). Производители маркируют устройства по холодопроизводительности выраженной в кBTU (1000 BTU/h = 293 Вт). Исходя из этой маркировки, можно судить, подходит ли данный кондиционер под нужды будущего владельца или нет:

- 07 – мощность составляет 2 кВт. В среднем, такое устройство можно поставить в помещение площадью 18-20 м.кв.;
- 09 – кондиционеры на 2,5-2,6 кВт. Подходят для помещений площадью до 26 м.кв.;
- 12 – наиболее мощный вариант среди бытовых кондиционеров (3,5 кВт). Такую сплит-систему можно установить в помещении до 35 м.кв.

Маркировка 12 – кондиционер рассчитан на площадь большого помещения с высокими потолками.

Некоторые производители используют другие значения – например, Toshiba маркируют так же в BTU цифрами 10 и 13 (они чуть мощнее «девяток» и «двенашек» соответственно). А, например Mitsubishi в маркировке применяют цифры соответствующие площади помещения – 20, 25, 35 (что аналогично «семеркам», «девяткам» и «двенашкам» соответственно).

Ниже приведена таблица, в которой указана необходимая холодопроизводительность на определенную площадь помещения. Обратите внимание, что данная таблица учитывает только стандартную высоту потолков, малую освещенность, минимальное количество техники и людей.

Площадь комнаты	Необходимая мощность	Маркировка традиционная	Маркировка, указывающая на площадь
5-20 м.кв.	2 кВт	07	20
21-25 м.кв.	2,5 кВт	09 (10 у Toshiba)	25
26-35 м.кв.	3,5 кВт	12 (13 у Toshiba)	35
36-50 м.кв.	5 кВт	18	50
51-70 м.кв.	7 кВт	24	60-80

Рисунок 5.4 – Таблица необходимости холодопроизводительность на определенную площадь помещения

Если в помещении постоянно изменяется количество человек или работающей бытовой техники, солнце активно появляется лишь в определенное время суток, рекомендуется выбрать сплит-систему с функцией подстройки под окружающую среду (автоматический режим,

который есть практически в каждом современном приборе). Такие устройства способны поддерживать комфортный климат в зданиях без особого участия человека – алгоритм сам подбирает оптимальные параметры.

Формула расчета мощности кондиционера. Условно рассчитать мощность кондиционера можно, разделив площадь помещения на 10. Полученная величина и будет необходимой мощностью охлаждения в кВт. Это расчет для пустого помещения, без людей и техники. Также в данном расчете не учтена площадь и ориентация оконных проемов, площадь стен, полов и потолков. Человек, в зависимости от его деятельности в помещении, может выделять 0,1–0,3 кВт, компьютер — 0,3 кВт. Тепловую мощность остальной техники можно оценить как 50% от ее паспортной мощности. Мощность теплового излучения от людей и техники надо прибавить к условно рассчитанной мощности кондиционера. Этот расчет будет более близок к реальности.

Уточнения в расчет с учетом дополнительных параметров. Если вы хотите определить мощность кондиционера точнее, помимо теплового излучения от людей и техники, необходимо учесть и другие виды теплопритоков. В этой связи формула расчета мощности усложняется:

Расчет мощности кондиционера будет верным при среднем размера окна, равном 2-м кв. метрам. При большем остеклении к расчетной мощности следует прибавить 200–300 Вт при сильной инсоляции и 100–200 Вт при средней или низкой освещенности помещения. Кондиционер выбирается мощностью в диапазоне от -5% до +15% от расчетной.[10]

Расчет мощности кондиционера для офисного помещения.

Дано: офисное помещенье площадью 32 кв. метров с высотой потолка 2,5 метра. На данной площади работают три человека, в помещении размещены 3 компьютера, телевизор и маленький холодильник. Окна выходят на солнечную сторону, компьютер и телевизор одновременно не работают. Какой мощности нужен кондиционер в этих условиях?

По вышеприведенной формуле рассчитываем теплопотоки, принимая q=40.

Q1 = S x h x q / 1000 = 32 x 2,5 x 40/1000 = 3,2 кВт.

Q2 = 0,2 кВт, поскольку работники ведут спокойный, монотонный режим работы.

Для расчета Q3 берем мощность теплопотока от компьютера как большую. Это 0,3 кВт. От холодильника обычно исходит 30% тепла. Если холодильник маленький, его мощность составляет около 0,165 кВт.

Q3 = 0,3 кВт + 0,165 кВт x 0,3 = 0,35 кВт

Итого, расчетная мощность кондиционера составит:

Q = Q1 + Q2 + Q3 = 3,2 кВт + 0,2 кВт + 0,35 кВт = 3,75 кВт

Таким образом, выбрать мощность кондиционера для данных условий можно в интервале от 3,14 кВт до 3,80 кВт.

Заключение

В этом дипломном проекте был рассмотрен мониторинг вредоносных событий с использованием машинного обучения, сделан обзор доступных решении систем обнаружений, вторжений и аномалий. Изучены решения текущих проблем безопасности данных с помощью машинного обучения. Для проверки текущего состояние науки данных, проведен анализ алгоритма линейной регрессии. Было объяснены, какие типы наборов данных используются. Показана среда разработки Anaconda и его взаимодействия с языком программирования Python. Был осуществлен поиск пересечения машинного обучения и информационной безопасности, а также исследованы вопросы импорта основных библиотек Python, используемых в машинном обучении.

В проекте показаны разные способы реализации деревьев решений: «Изолированный лес» и «Случайный» лес для обнаружений аномалий в наборах данных и нахождение ранее необнаруживаемых системами угроз.

Злоумышленники в системе во время этих разведывательных запросов и движений оставляют некоторые следы. Именно по этим следам можно сделать вывод о данных, о траектории их поведения и их присутствие может быть обнаружено с помощью науки о данных инструментами машинного обучения для своевременного оповещения.

Поскольку набор данных большой, то выборка и исследование по представленному набору является качественной, в рамках которой определяются подмножества функций из набора данных, которые считаю наиболее перспективными. В процессе исследования происходит чтение данных во фрейм и последующая их сортировка по дате, поскольку проблема требует возможности прогнозировать события в будущем. Выполняется разделение на тренировочные и тестируемые события, то есть временная прогрессия. Создаются, подбираются и тестируются данные случайным лесным классификатором. В зависимости от применения достигнутая точность является хорошей отправной точкой. Также используется другой экземпляр классификатора (изолированный лес) после загрузки данных. Для параметра загрязнения используется значение, соответствующее соотношению угроз для невредоносных событий. На следующих трех этапах изучается показатели принятия решений по изолированному лесу на опасных субъектах и посредством проверки, пороговое значение 0,12 обнаруживает большую долю субъектов угроз без отметки обычных пользователей. Наконец, оценивая эффективность в шагах я увидел, что были некоторые ложные срабатывания, но также обнаружено значительное количество внутренних угроз.

Таким образом, в рамках дипломного проекта были поставлены и успешно реализованы задачи:

— изучена предметная область (машинное обучение);

— рассмотрены существующие алгоритмы машинного обучения (Линейная регрессия, Деревья решений);

— проанализированы и выбраны инструментарий анализа и разработки (Anaconda);

— исследованы и подготовлены данные CSV-формата до первичного запуска в листе;

— прописаны блок-схемы, сформированы признаки, адаптируя данные, под алгоритмы машинного обучения;

— построены модели классификаторов ML Isolation Forests и Random Forests;

— написаны программные коды для обнаружения аномалий DDOS-атак, фишинга, сетевого трафика;

— проанализированы результаты коэффициентом отсечения аномальных данных от всех наблюдений по завершению построений графика моделей классификаторов и матрицами ошибок;

— оценены риски ИБ;

— произведены расчеты, согласно стандартам БЖД;

— сделаны соответствующие выводы.

Достоинством в проведенном анализе методов машинного обучения является обнаружения ранее неизвестных свойств данных, которые можно использовать в последующих задачах. Используя алгоритмы, можно соединить точки и найти шаблоны, которые раньше было трудно найти вручную.

Список сокращений

БЖД – безопасность жизнедеятельности.

ВДТ - видео-дисплейный терминалы.

ВТ – вычислительная техника.

ВОЗ - Всемирная организация здравоохранения.

ИБ – Информационная безопасность.

ИИ – Искусственный интеллект.

КЗС - Компьютерный зрительный синдром.

МО – машинное обучение.

ПК – персональные компьютеры.

ПЭВМ - Пользователи электронно-вычислительных машин.

СДСН - синдром длительных статических нагрузок.

СЗК – синдром запятсного канала.

ТПН - травма повторяющихся нагрузок.

ЭМП – электромагнитное поле.

CSV – (Comma-Separated Values) - текстовый формат, предназначенный для представления табличных данных.

DDoS - (Distributed Denial of Service) - атака, при которой трафик из разных источников затопляет жертву, что приводит к прерыванию обслуживания.

DNS - (Domain Name System) - распределенная система для получения информации о доменах.

IDS - (Intrusion Detection System) - обнаружении и регистрации атак, оповещении при срабатывании определенного правила.

IPS – (Intrusion Prevention System) - программно-аппаратная система сетевой и компьютерной безопасности, обнаруживающая вторжения.

SIEM - (Security information and event management) - решения для осуществления мониторинга информационных систем.

TCP - (Transmission Control Protocol) - выполняет функции транспортного протокола с установлением логического соединения.

Список литературы

1 Джордан М.И. и Митчелл. Т.М. Машинное обучение: тенденции, перспективы и перспективы. - М.: Science, 2015. – 248 с.

2 Бучак А. и Гувен Э. Обзор методов интеллектуального анализа данных и машинного обучения для обеспечения безопасности. - М.: IEEE Communications Surveys & Tutorials, 2015. – 226 с.

3 Хан А., Бахарудин Б. и Ли Л.Х. Обзор алгоритмов машинного обучения для текстового классификация документов // Журнал достижений в области информационных технологий. – 2010. – 269 с.

4 Байер У., Хабиби И., Бальзаротти Д., Кирда Э. и Крюгель К. Взгляд на текущее поведение вредоносных программ. – М.: В ЛИТ, 2009. – 281 с.

5 Андерсон Б., Куист Д., Нил Дж., Сторли С. и Лейн Т. Обнаружение вредоносных программ с использованием динамического анализа. – М.: J Comput Virol, - 2011, - №7. – С. 247–258.

6 Перацци Ф., Апруззезе Г., Коладжанни М., Гвидо А. и Маркетти М. Масштабируемая архитектура для онлайн определение приоритетов киберугроз. – М.: Cycon, 2017. - 95с.

7 Международный стандарт ISO 27005:2013 «Информационные технологии - Методы обеспечения безопасности - Менеджмент риска информационной безопасности».

8 «Правила обязательной периодической аттестации производственных объектов по условиям труда». Приказ Министра здравоохранения и социального развития Республики Казахстан от 28 декабря 2015 года № 1057. Зарегистрирован в Министерстве юстиции Республики Казахстан 31 декабря 2015 года № 12743.

9 Санитарные правила «Санитарно-эпидемиологические требования к условиям работы с источниками физических факторов (компьютеры и видеотерминалы), оказывающих воздействие на человека». Утверждены приказом Министра национальной экономики Республики Казахстан от 21 января 2015 года № 38.

10 СП РК 2.04-104-2012 «Естественное и искусственное освещение».

Приложения А

DDOS phising

```
import pandas as pd
df = pd.read_csv("ddos_dataset.csv")
df2 = df.sort_values("Timestamp")
In[]:
d = len(df2.index)
obuch_df = df2.head(int(d*0.8))
test_df = df2.tail(int(d*0.2))
In[]:
from collections import Counter
print(Counter(obuch_df[‘metka’].values))
print(Counter(test_df[‘metka’].values))
Counter({'Vse': 332658, 'ddos': 66549})
Counter({'Vse': 66234, 'ddos': 33598})
In[]:
y_obuch = obuch_df.pop(‘metka’).values
y_test = test_df.pop(‘metka’).values
In[]:
X_obuch = obuch_df.values
X_test = test_df.values
In[]:
from sklearn.ensemble import RandomForestClassifier
aues=RandomForestClassifier(n_estimators=45)
In []:
aues.fit(X_obuch, y_obuch)
RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
    max_depth=None, max_Funcias='auto', max_leaf_nodes=None,
    min_impurity_decrease=0.0, min_impurity_split=None,
    min_samples_leaf=1, min_samples_split=2,
    min_weight_fraction_leaf=0.0, n_estimators=45, n_jobs=None,
    oob_rez=False, random_state=None, verbose=0,
    warm_start=False)
aues.rez(X_obuch, y_obuch)
In[]:
aues.rez(X_test, y_test)
```

Приложения В

URL Phising

```
import pandas as pd
import os
obuchCSV = os.path.join("phishing-dataset","obuch.csv")
testCSV = os.path.join("phishing-dataset","test.csv")
obuchdf = pd.read_csv(obuchCSV)
testdf = pd.read_csv(testCSV)
```

In[]:

```
y_obuch = obuchdf.pop("target").values
y_test = testdf.pop("target").values
```

In[]:

```
X_obuch = obuchdf.values
X_test = testdf.values
```

In[]:

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_rez, confusion_matrix
import itertools
aues = RandomForestClassifier()
aues.fit(X_obuch, y_obuch)
y_test_pred = aues.predict(X_test)
print(accuracy_rez(y_test, y_test_pred))
print(confusion_matrix(y_test, y_test_pred))
cm=confusion_matrix(y_test, y_test_pred)
```

Internet Trafic

```
import pandas as pd
dataset_df = pd.read_csv("dataset.csv", index_col=None)
```

In[]:

```
y = dataset_df[‘metka’].values
from collections import Counter
dataset_df[‘metka’] = dataset_df[‘metka’].apply(labelAnomalous)
```

In[]:

```
y = dataset_df[‘metka’].values
counts = Counter(y).most_common()
zagrParametr = counts[1][1]/(counts[0][1]+counts[1][1])
```

In[]:

```
from sklearn.preprocessing import LabelEncoder
slovar_kodirovpk = dict()
for c in dataset_df.columns:
    if dataset_df[c].dtype == "object":
        slovar_kodirovpk[c] = LabelEncoder()
```

```
        dataset_df[c] = slovar_kodirovpk[c].fit_transform(dataset_df[c])
In[]:
dataset_df_normal = dataset_df[dataset_df[‘metka’] == 0]
dataset_df_anomaly = dataset_df[dataset_df[‘metka’] == 1]
In[]:
y_normal = dataset_df_normal.pop(‘metka’).values
X_normal = dataset_df_normal.values
y_anomaly = dataset_df_anomaly.pop(‘metka’).values
X_anomaly = dataset_df_anomaly.values
In[]:
from sklearn.model_selection import train_test_split
X_normal_obuch, X_normal_test, y_normal_obuch, y_normal_test = obuch_test_s
plit(X_normal, y_normal,test_size=0.2, random_state=12)
In[]:
X_anomaly_obuch, X_anomaly_test, y_anomaly_obuch, y_anomaly_test = obuch_
test_split(X_anomaly, y_anomaly, test_size=0.25, random_state=12)
In[]:
In[]:
from sklearn.ensemble import IsolationForest
IF = IsolationForest(zagr=zagrParametr)
In[]:
IF.fit(X_obuch)
resheniyeRezs_obuch_normal = IF.resheniye_funcia(X_normal_obuch)
resheniyeRezs_obuch_anomaly = IF.resheniye_funcia(X_anomaly_obuch)
In[]:
import matplotlib.pyplot as stat
%matplotlib inline
stat.figure(figsize=(20, 10))
_ = stat.hist(resheniyeRezs_obuch_normal, bins=45)
In[]:
stat.figure(figsize=(25, 15))
_ = stat.hist(resheniyeRezs_obuch_anomaly, bins=45)
In[]:
srez = 0
In[]:
print(Counter(y_test))
print(Counter(y_test[srez>IF.resheniye_funcia(X_test)]))
```

Приложения С

```
import numpy as np
import pandas as pd
```

```
In[]:
put_v_dataset = "./dataPolz/"
```

```
In[]:
TypLoga = ["ustroistvo","email","file","vhod","http"]
logPoleList = [["date","Polz","deystive"],["date","Polz","to","cc","bcc"], ["date","
Polz","imyafila"], ["date","Polz","deystive"], ["date", "Polz", "url"]]
```

```
In[]:
Funcias = 0
Upor_po_funcia = {}
def dobavFuncia(name):
    if name not in Upor_po_funcia:
        global Funcias
        Upor_po_funcia[name] = Funcias
        Funcias+=1
```

```
In[]:
dobavFuncia("Vyhodnoi_Vhod_Normal","Vyhodnoi_Vhod_After","Weekend_Vh
od"
,"Vyhod" ,"Connect_Normal","Connect_After","Connect_Weekend","Disconnect"
)
def vhodFuncias(stroka):
    if stroka["deystive"] == "Vhod":
        if stroka["date"].Vyhodnoi() < 5:
            if stroka["date"].hour >= 8 and stroka["date"].hour < 18:
                return Upor_po_funcia["Vyhodnoi_Vhod_Normal"]
            else:
                return Upor_po_funcia["Vyhodnoi_Vhod_After"]
        else:
            return Upor_po_funcia["Weekend_Vhod"]
    else: #Is Vyhod
        return Upor_po_funcia["Vyhod"]
```

```
In[]:
logFunciaFuncias = [ustroistvoFuncias, emailFuncias, fileFuncias, vhodFuncias, ht
tpFuncias]
```

```
In[]:
dfs = []
for i in range(len(TypLogas)):
    TypLoga = TypLogas[i]
logPoles = logPolesList[i]
    logFunciaFuncia = logFunciaFuncias[i]
```

Продолжение Приложении С

```
 df = pd.read_csv(put_v_dataset + TypLoga+".csv", usecols=logPoles, index_col=
None)
    dateFormat = "%m/%d/%Y %H:%M:%S"
    df["date"] = pd.to_datetime( df["date"], format=dateFormat)
    newFuncia = df.apply(logFunciaFuncia, axis=1)
    df["Funcia"]=newFuncia
    cols_to_keep = ["date", "Polz", "Funcia"]
    df = df[cols_to_keep]
    #Преобразуем дату в день
    df["date"]=df.apply(dateToDen, axis=1)
    dfs.append(df)
In[]:
joint = pd.concat(dfs)
In[]:
joint = joint.sort_values(Po="date")
In[]:
obuch_df = joint[joint["date"]<=d]
test_df = joint[joint["date"]>=d]
In[]:
In[]:
def OrgDen(ts, Upor_po_funcia):
    fv = np.zeros(len(Upor_po_funcia))
    counts = ts["Funcia"].value_counts().to_dict()
    for Funcia in counts:
        fv[Funcia]=counts[Funcia]
    return fv
def OrgPolzTime(Polzname, df, Upor_po_funcia, dateK_Index):
    ts = izvlTimePoPolz(Polzname, df)
    x = np.zeros((len(Upor_po_funcia),timeWindow))
    for date in set(ts["date"].values):
        ts2 = izvlTimePoDate(date, ts)
        fv = OrgDen(ts2,Upor_po_funcia)
        x[:,dateK_Index[date]]=fv
    return x
In[]:
X_obuch, obuch_PolzK_Index, obuch_Index_KPolz = collectDataset(obuch_df, U
por_po_funcia, obuch_dateK_Index)
X_test, test_PolzK_Index, test_Index_KPolz = collectDataset(test_df, Upor_po_fu
ncia, test_dateK_Index)
IstUgrozIndicesObuch = set([])
IstUgrozIndicesTest = set([])
for IstUgroz in IstUgrozs:
```

Продолжение Приложении С

```
 if IstUgroz in obuch_PolzK_Index:
        IstUgrozIndicesObuch.dobav(obuch_PolzK_Index[IstUgroz])
    if IstUgroz in test_PolzK_Index:
        IstUgrozIndicesTest.dobav(test_PolzK_Index[IstUgroz])
```

In[]:

```
obuch_normalIndices = set(obuch_Index_KPolz.keys()) - IstUgrozIndicesObuch
test_normalIndices = set(test_Index_KPolz.keys()) - IstUgrozIndicesTest
```

In[]:

```
y_obuch = np.zeros(len(X_obuch))
y_obuch[list(IstUgrozIndicesObuch)]=1
y_test = np.zeros(len(X_test))
y_test[list(IstUgrozIndicesTest)]=1
```

In[]:

```
from sklearn.ensemble import IsolationForest
zagrParametr = 0.07
IF = IsolationForest(n_estimators=100, max_samples=256,zagr=zagrParameter)
IF.fit(X_obuch)
normalRezs = IF.resheniye_funcia(X_obuch_normal)
```

In[]:

```
import matplotlib.mlab as mlab
import matplotlib.pyplot as stat
normal = stat.hist(normalRezs, 45,density=True)
stat.xlabel('Аномальные значения')
stat.ylabel('Проценты')
stat.title("Распределение аномальных значений для обычных событий")
```

In[]:

```
c= IF.resheniye_funcia(X_obuch)
```

In[]:

```
c = IF.resheniye_funcia(X_test)
```