**IT-Колледж “Сириус”**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**ДОКЛАД**

по дисциплине “Введение в специальность”

на тему “Python в ML”

Выполнил:  
Студент группы

1.9.7.1  
Загора Анна Витальевна

Принял:

Старший преподаватель  
Тенигин Альберт Андреевич

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

IT-Колледж “Сириус”  
2022

**Содержание**

**Содержание**

[**Глоссарий** 3](#_Toc117675870)

[**Введение** 4](#_Toc117675871)

[**Актуальность** 5](#_Toc117675872)

[**Без учителя (Unsupervised machine learning)** 6](#_Toc117675873)

[**С учителем (Supervised machine learning).** 7](#_Toc117675874)

[**Глубокое обучение (Deep learning).** 8](#_Toc117675875)

[**Литература** 9](#_Toc117675876)

# **Глоссарий**

1. МО – машинное обучение
2. Глубокое обучение – DeepLearning
3. ИИ – искусственный интелект
4. Без учителя (Unsupervised machine learning).
5. С учителем (Supervised machine learning).
6. Глубокое обучение (Deep learning).

# **Введение**

Машинное обучение (ML) – это создание моделей, несущее в себе определенные методики, которые позволяют системе учиться, благодаря решению множества сходных задач, анализу и предоставленным данным. Позже, обучившись на определенных инструкциях, способны самостоятельно делать запрашиваемую похожую работу без вмешательства человека.

Сначала строиться модель, потом внедряются и применяются определенные алгоритмы. У каждой модели машинного обучения есть свой алгоритм и своя формула.

Существует три вида машинного обучения:

* Без учителя (Unsupervised machine learning).
* С учителем (Supervised machine learning).
* Глубокое обучение (Deep learning).

Python - это высокоуровневый скриптовый язык программирования. Это интерпретируемый язык. Он используется в разнообразных сферах IT. Примером служит: разработка приложений (мобильных и десктопных), парсинг, web, машинное обучение и т.д.  Он был разработан голландским программистов в начале 90-х годов Гвидо ван Россумом.

Код можно писать практически в любом редакторе или использовать специальные IDE:

* Visual Studio Code — популярный текстовый редактор от Microsoft.
* IDLE — стандартный текстовый редактор в составе языка.
* SublimeText — текстовый редактор с множеством плагинов.
* PyCharm — мощная среда разработки от JetBrains.
* Spyder — IDE, оптимизированная для работы в Data Science. Идёт в пакете с Anaconda.

# **Актуальность**

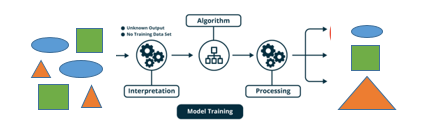
Машинное обучение поддерживает множество вариантов использования в современном мире. Оно обладает потенциалом не только для применения в науке, здравоохранении, строительстве и энергетике, но также в розничной торговле, электронной коммерции и финансовых услугах. Так же машинное обучение может предоставить безопасность благодаря прогнозированию, например, предусмотреть снижение эксплуатационных расходов и автоматизировать какие-либо процессы, в которых человек смог бы допустить ошибку по своей невнимательности. При использовании данного подхода повышается риск увеличения доходов, ибо будет улучшена производительность взаимодействий с пользователем.

Методы углубленного обучения, такие как нейронные сети, часто используются для классификации и распределения иллюстраций, поскольку они могут эффективно идентифицировать соответствующие особенности изображения при наличии потенциальных осложнений, или поспешно выявить схожесть черт лица преступников.

Машинное обучение приносит в себе большое количество ключевых вариантов использования в бизнесе. А именно - обеспечивает конкурентное преимущество. Оно улучшает видимость бизнеса и сотрудничество. Среди наиболее распространенных возможностей машинного обучения - его способность автоматизировать и потратить как можно меньше времени для принятия решений и время получения ценности.

# **Без учителя (Unsupervised machine learning)**

Машинное обучение без учителя – это алгоритм, используемый для обучения набора без структурированных данных. То есть, ИИ сама старается строить логическую классификацию информации. Зачастую используется для определения и структурирования материалов.



Например, на предоставленной картинке, входными данными являются фигуры. Машина не может различить их только по цвету, форме или тому, что их объединяет. Ей предстоит группировать их по различным категориям.

Существует несколько типов алгоритмов неконтролируемого машинного обучения.

**1. Кластеризация k-средних**

Данный процесс разделяет точки данных или наблюдения на какое-то количество неизвестных скоплений, по другому называемому кластеру, так чтобы каждое наблюдение что-то связывало. Единственный недостаток – данный алгоритм не запуститься без уточнения пользователем необходимого количества скоплений. Является самым простым и актуальным среди всех.

**2. Кластеризация DBSCAN (Основанная на плотности пространственная кластеризация для приложений с шумами)**

Алгоритм немного похож на Кластеризацию k-средних, но несет в себе две основные концепции.

1. Достижимость по плотности
2. Связность по плотности

Оба данных подхода помогают алгоритму разделять данные на скопления. Для осуществления его все начинается с определения двух главных границ– параметра радиуса eps (ϵ) и минимального количества точек в пределах радиуса (м), а дальше идет распределение, которое отталкивается от данных точек.

1. **Анализ главных компонентов (PCA)**

Данный тип алгоритмов уменьшает размерности, то есть снижает количество переменных в большом потоке информации, предоставленной для обучения машины. Делает он это путем преобразования группы маленьких переменных в одну общую по определенным характеристикам. Процесс происходит с практически нулевой потерей информации. Единственный минус - есть риск потери точности данных.

1. **Правила ассоциации**

Это набор условных операторов, которые стараются показать связь между элементами данных в больших наборах информации. Они полезны при анализе типичных данных, например, связанных с торговлей, где собранные с помощью сканеров штрих-кода в супермаркетах. Эти данные обычно содержат многочисленные записи транзакций. В целом, это помогает в правильном распределении предметов на сайтах или внутри магазинов.

В целом, оно лучше всего подходит для более сложных задач, включая описательное моделирование и обнаружение шаблонов, чем контролируемое обучение.

# **С учителем (Supervised machine learning).**

Машинное обучение с учителем (контролируемое машинное обучение) – это алгоритм, основанный на обработке данных из прошлого опыта. Входная переменная (x) сопоставляется выходной переменной (y) благодаря функции отображения, которая прорабатывается и изучается моделью.

Дабы получить желаемый результат, модель должна создать функцию Y= f(x). Алгоритм имеет целевую или конечную переменную (зависимую переменную), которая определяется из предоставленного набора предикторов, так называемых независимых переменных.

Это может занять какое-то время для ожидания, пересмотра данных и итерации. Она связывает две переменные с достижением цели, чтобы предсказать правильную метку для входных данных. Этот процесс обучения повторяется до тех пор, пока модель не достигнет высокого уровня точности.

Данным алгоритмом активно пользуются, ибо он умеет превращать данные в полезную информацию для достижения желаемых результатов, так же приносит благо большому количеству отраслей.  Это возможно при предоставлении качественных обучающих данных. Если действительно хорошо обучить модель, то это приведет к улучшению производительности, что даст преимущество перед конкурентами в различных сферах.

В целом, его недостатки заключаются в том, что массивные данные в контролируемом машинном обучении маркировать трудно и очень долго. Так же тяжело предугадать действительно верно результат, если распределение тестовых данных значительно отличается обучающего набора.

# **Глубокое обучение (Deep learning).**

## Глубокое обучение - это определенная методика разработки и обучения нейронных сетей. Структура нейронной сети опирается на структуру человеческого мозга. Подобно коре головного мозга искусственная нейронная сеть несет в себе много слоев взаимосвязанных персептронов.

глубокое обучение опирается на нелинейный подход обучающих машин к обработке данных, точнее - подаются в систему глубокого обучения, проходят через взаимосвязанную паутину скрытых уровней. Нейронная сеть представляет из себя скрытые слои, которые модифицируют, обрабатывают и анализируют, определяя связь с целевыми переменными.

# **Python в ML**

В ML используются разные языки программирования, например это **Python, Julia, R, Java**и другие. Каждые из них несут в себе определенную концепцию и подход к работе.

Один из распространенных – это Python. Он популярен благодаря полезным инструментам и большому количеству различных библиотек, которые помогают облегчить рабочую нагрузку.

Python предоставляет пользователям одни из лучших **гибких возможностей и функций**, которые повышают производительность и **качество кода**.

Особенности, которые ставят Python на первое место лучших языков программирования для машинного обучения, глубокого обучения и искусственного интеллекта:

* **Различные библиотеки** демонстрируют, что для каждой существующей проблемы есть решение.
* **Доступный исходный код** делает искусственный интеллект гарантирует улучшения в долгосрочной перспективе.
* **Постепенное внедрение и интеграция** делают его доступным для людей с различным уровнем квалификации, чтобы адаптировать его.
* Повышение производительности за счет **сокращения времени на кодирование и отладку.**
* **Легко работает с модулями кода на C и C ++**.
* Может **использоваться для простых вычислений**, а также для обработки естественного языка.

# **Библиотеки Python для машинного обучения**

Языков для машинного обучения и глубокого обучения на выбор достаточно много, но Python является одним из наиболее **востребованных среди разработчиков.** Данный язык пользуется поддержкой **множества библиотек на выбор**, которые подходят для каждого варианта использования, проблемы или проекта.

Лучшие библиотеки Python для машинного обучения:

# **TensorFlow**

Это одна из лучших библиотек, доступных для работы с машинном обучением на Python, так же с исходным кодом для исследований и производства. Отличается тем, что является быстрой, масштабируемой и гибкой. TensorFlow от Google упрощает создание и обучение моделей ML для людей любого уровня знаний. Благодаря этой библиотеке можно создавать и обучать модели на мобильных устройствах и серверах, а не только на компьютерах (TensorFlow Lite и TensorFlow Serving). Основные области ML и DL, в которых TensorFlow выделяется(помимо создания моделей глубокого машинного обучения): обработка естественного языка, возможности абстракции, простое и совместное использование идей и кода, дифференциальное уравнение в частных производных, распознавание изображений, речи и текста, работа с глубокими нейронными сетями.

* **Keras**

Одна из самых популярных библиотек с исходным кодом на Python с нейронными сетями. Изначально был разработан инженером Google по заказу для ONEIROS (Open-Ended Neuro Electronic Intelligent Robot Operating System). Позже был поддержан в самой библиотеке TensorFlow, которая выступала как базовая. Она включает в себя пару инструментов и строительных блоков, без которых нейронную сеть нельзя построить: нейронные слои, цели, функции активации и стоимости, объединение в пул, отсев, пакетная нормализация.

* **PyTorch**

Была разработана Facebook. Поддерживает не только Python, но и C++. Считается одним из претендентов на звание лучшей платформы глубокого и машинного обучения. Имеет множество плюсов, например, удобен в настраивании под работу, широко и часто используется в исследованиях глубокого обучения. Минусы состоят в том, что имеет гораздо меньше абстракции NLP, которые совершенно не оптимизированы для скорости.

* [**Scikit-learn**](https://scikit-learn.org/stable/user_guide.html)

Активно используемая библиотека на Python. Несет в себе понятную интеграцию с разными библиотеками программирования на языке ML, такими как NumPy и Pandas. Scikit-learn несет в себе алгоритмы: регрессия, уменьшение размеров, классификация, кластеризация, выбор модели и предварительная разработка.

Scikit-learn очень прост и гибок в использовании, так же он нацелен на моделирование данных.

# **Выводы**

В ходе работы над проектом были изучены различные методы изучения машинного обучения, его тонкости и его связь с искусственным интеллектом. Были рассмотрены языки программирования для обучения моделей, в том числе особое внимание уделено Python. Было рассмотрено большое количество библиотек языка программирования Python, которые были специально разработаны для создания моделей, отобранные из них – это самые популярные и часто используемые. Каждая библиотека изучалась отдельно.

На основании проведенной работы можно сделать выводы о том, что существует большое количество подходов к машинному обучению конкретно на данном популярном языке программирования. Пользователь, знающий лишь Python, сможет сделать много проектов или подойти к определенным проблемам с совершенно разных сторон, подключая библиотеки и разновидные подходы к ML.

Каждая из видов машинного обучения, написанная на Python заслуживает к себе внимания и приведет к желаемому результату.

# **Литература**

1. Алгоритмы машинного обучения // asu-analitika.ru URL: https://asu-analitika.ru/razlichnye-tipy-algoritmov-mashinnogo-obuchenija/ (дата обращения: 24.10.22).

2. Supervised learning and other machine learning tasks // blog.superannotate.com URL: https://blog.superannotate.com/supervised-learning-and-other-machine-learning-tasks/#supervised-learning (дата обращения: 24.10.22).

3. Best Python Libraries for Machine Learning and Deep Learning // towardsdatascience.com URL: https://towardsdatascience.com/best-python-libraries-for-machine-learning-and-deep-learning-b0bd40c7e8c (дата обращения: 24.10.22).

4. Machine Learning In Python // www.askpython.com URL: https://www.askpython.com/python/machine-learning-introduction (дата обращения: 25.10.22).

5. The Best Machine Learning Libraries in Python // stackabuse.com URL: https://stackabuse.com/the-best-machine-learning-libraries-in-python/ (дата обращения: 25.10.22).

6. Франсуа Шолле Глубокое обучение в Python. - 2-е изд. - СПб: Питер, 2019