УДК: 004.67+004.912
ББК: 74.025
ГРНТИ: 14.01.77

**Нурматова Елена Вячеславовна**, *кандидат технических наук, доцент, ФГБОУ ВО «МИРЭА – Российский технологический университет», Москва, Россия, e-mail: nurmatova@mirea.ru*

**Nurmatova Elena V.**, *PhD in Engineering Sciences, Associate Professor, MIREA - Russian Technological University, Moscow, Russia, e-mail: nurmatova@mirea.ru*

**Зайцев Евгений Игоревич**, *кандидат технических наук, доцент, ФГБОУ ВО «МИРЭА – Российский технологический университет», Москва, Россия, e-mail: zajcev@mirea.ru*

**Zajcev Evgeniy I.**, *PhD in Engineering Sciences, Associate Professor, MIREA - Russian Technological University, Moscow, Russia, e-mail: zajcev@mirea.ru*

# Алгоритм прогнозирования вовлечённости в обучение на основе результатов анализа временного ряда

**Аннотация.** *В современном быстро развивающемся обществе всё больше уделяют внимание вопросам прогнозирования, играющим важнейшую роль в процессах как стратегического планирования, так и оперативного управления в различных сферах науки и техники. В проводимом исследовании стоит задача проанализировать данные и вовремя спрогнозировать вовлечённость обучающихся, которая зависит от актуальности обучения, мотивации. В первом случае, если участник образовательного процесса понимает, какие его личные проблемы поможет решить обучение, какие в результате будут преимущества, то уровень вклада его ресурсов и времени будет больше. Во втором, даже при понимании того, как обучение поможет развить важные компетенции для успешной реализации в профессии, постоянно нужны подкрепления импульса, запускающего и удерживающего степень вовлечённости.*

**Ключевые слова:** *методы машинного обучения, временной ряд, прогнозирование, модели прогнозирования, методы прогнозирования, архитектура модели, вовлечённость в обучение.*

# Algorithm for Predicting Learning Engagement Based on the Results of Time Series Analysis

**Abstract.** *In today's rapidly developing society, more and more attention is paid to the issues of forecasting, which play a crucial role in the processes of both strategic planning and operational management in various spheres of science and technology. In the current study, the task is to analyse data and predict the involvement of students in time, which depends on the relevance of learning and motivation. In the first case, if the participant of the educational process understands what his personal problems will help to solve the training, what will be the benefits as a result, the level of contribution of his resources and time will be greater. In the second case, even when understanding how training will help to develop important*

*competences for successful realisation in the profession, there is a constant need for reinforcement of the impulse that triggers and maintains the level of involvement.*

**Keywords:** *machine learning methods, time series, prediction, prediction models, prediction methods, model architecture, engaged learning.*

## Введение

Прогнозирование временных рядов представляет собой научный подход к обработке данных, основанный на применении интеллектуальных алгоритмов и современных компьютерных технологий. Этот метод используется для анализа прошлых наблюдений, предсказания будущих значений, а также для целей планирования, управления и оптимизации процессов. Анализ временных рядов включает исследование факторов, формирующих набор данных за определённый период времени. Чёткое понимание этих причин позволяет выявлять важные статистические закономерности и характеристики через моделирование, гипотезы и разложение данных на составляющие компоненты [3].

## Описание и предобработка данных

В работе представлен набор данных, который импортируется из СУБД PostgreSQL, состоящий из 11 колонок и 7043 строк, включающий следующие информативные признаки: дата, пол, возраст, семейное положение, уровень, форма и срок обучения, выделяется ли на обучение достаточно времени, процент выполнения заданий, количество успешно выполненных зданий в модуле, применение новых знаний на практике, уровень вовлечённости в обучение.

Проведена предподготовка на предмет выявления и обработки пропущенных, нетипичных, неинформативных (дубликатов) и несогласованных данных. Пропущенных значений не обнаружено, но есть несоответствие типа данных. В датасете присутствует большое количество переменных с текстовыми данными, применена описательная статистика категориальных признаков. Выбросы в данных не наблюдаются. В случае с уровнем вовлечённости в обучение (*EngagemenLearning*) можно наблюдать смещённость в сторону 0. Для графиков срока обучения (*StudyPeriod*) и количества успешно выполненных зданий в модуле (*SuccessfullyCompletedTasks*) налицо наличие явной кластеризации значений.

После преобразований набора данных, можно сделать следующие выводы:

- пропуск данных отсутствует, кроме показателя *EngagementLearning*; все значения заполнены корректно, без ошибок и вариаций формулировок;
- среди обучающихся преобладают женщины;
- возраст большинства участников варьируется от 19 до 38 лет;
- большая часть обучающихся не состоит в браке и не имеет иждивенцев;
- большинство активно используют дополнительные учебные материалы;

- медиана выполнения заданий среди всех обучающихся составляет 65,4 %;
- уровень трудностей в онлайн-образовании незначительно превышает 29 %;
- доля обучающихся с пониженным уровнем вовлечённости составила 26,5 %;
- выбросы в данных отсутствуют.

**Исследование зависимостей и формулирование гипотез**

Чтобы сформировать гипотезы необходимо изначально привести все данные к одному типу и удалить ненужные признаки.

Для разбиения на классы необходимо воспользоваться алгоритмом, применяемым, когда в наборе данных встречается большое количество признаков. А именно, задаётся итератор, который в рамках идентификаторов колонок проходит по ним и по условию для признаков, соответствующих типу данных object, заменяет значения 'Yes' и 'No' на 0 и 1. Далее итератор в рамках уникальных значений признака, по условию для замены нецифровых значений и условию компенсации порядкового номера (неуникальных значений больше 2) тех признаков в которых не было замечено 'Yes' и 'No', выполняет замену всех нецифровых значений на соответствующий порядковый номер.

Визуализация зависимости таких признаков, как процент вовлечённости обучаемых и количество месяцев обучения показала, что больше 20 % слушателей снизили уровень вовлечённости в первые месяцы обучения. Медианное значение группы с отсутствием вовлечённости оказалось выше, чем значение у группы активных обучаемых.

После исследования зависимостей и проведённого анализа можно сделать следующие выводы:

- Как оказалось, одной из основных причин снижения вовлечённости является недостаток технических навыков, изоляция в онлайн-обучении и готовность перенести результаты обучения в свою работу;
- По причине различных технических проблем слушатели со сниженной вовлечённостью не пользовались дополнительными материалами (записанными роликами вебинаров, образцами работ в облачных хранилищах);
- Около 20 % слушателей со сниженной вовлечённостью обучались только около месяца.

**Построение моделей для прогнозирования оттока**

Изначально импортировались библиотеки классификаторов, методы автоматической обработки данных и метрики качества классификации. Далее задаём параметры (рис. 1) базовых алгоритмов для каждого из методов параметров [1].

После определения параметров базовых алгоритмов, данные разделены на тренировочную и тестовую, для чего определена балансировка целевого признака: положительных значений – 1869, отрицательных – 5174. Положительных значений почти в

3 раза меньше отрицательных. При делении данных на тестовые и тренировочные применим стратификацию по *y*.

```
knn_params = {'n_neighbors' : np.arange(1, 10, 1)}  # Параметры для классификатора KNeighborsClassifier

gbc_params = {'learning_rate': np.arange(0.1, 0.6, 0.1)}   # Параметры для классификатора GradientBoostingClassifier

rfc_params = {'n_estimators': range(10, 100, 10),  # Параметры для классификатора RandomForestClassifier
              'min_samples_leaf': range(1, 7)}

svc_params = {'kernel': ['linear', 'rbf'],
'C': np.arange(0.1, 1, 0.2)}                    # Параметры для классификатора SVC

lr_params = {'C': np.arange(0.2, 1, 0.1)} # Параметры для классификатора LogisticRegression

skf = StratifiedKFold(n_splits=8, random_state=17) # Параметры для кросс-валидации
```

*Рисунок 1. Параметры базовых алгоритмов*

Далее для обучения моделей потребовалось определение объекта классификатора для каждого из них, определение объекта кросс-валидации. После обучения моделей получаем лучшие параметры для классификаторов, полученных на кросс-валидации.

## Выбор классификаторов и метрик

Задача моделирования снижения мотивации к обучению является бинарной классификацией – процессом отнесения объектов к одной из двух категорий на основе своих признаков, включающей определение, имеется или отсутствует вовлечённость в образовательный процесс. Лучшими классификаторами в решении данных проблем являются – логистическая регрессия, метод ближайших соседей (k-nearest neighbors algorithm, kNN), опорных векторов (support vector machines, SVM), случайного леса (random forest), а также градиентный бустинг (XGBoost) [2].

Выбор метрики зависит от типа алгоритма обучения и от решаемой бизнес-задачи [4].

Рассмотрим наиболее распространённые метрики и их применение:

1) *Precision score* измеряет долю истинных положительных результатов (True Positives – TP) среди всех прогнозируемых положительных результатов (True Positives – TP и False Positives – FP). *Precision* используем в задачах с предпочтительным уменьшением количества ложно положительных результатов.

$$Precision = \frac{TP}{TP + FP},$$

2) *Recall score* измеряет долю истинных положительных результатов (TP) среди всех истинных положительных результатов (TP и False Negatives – FN). *Recall score* применяем в задачах с предпочтительным уменьшением количества ложно отрицательных результатов.

$$Recall = \frac{TP}{TP+FN},$$

3) *Accuracy score* измеряет долю правильных прогнозов среди всех прогнозов, является общей метрикой задач с сбалансированными классами. В противном случае может быть обманчивой метрикой, модель получает высокую точность, не предоставляя никакой реальной информации о решении задачи.

$$Acc = \frac{TP+TN}{TP+TN+FP+FN},$$

4) *ROC-AUC* измеряет площадь под ROC-кривой (*receiver operating characteristic),* представляющей кривую зависимости между TP-Rate и FP-Rate при различных порогах классификации. Метрика используется в задачах классификации, при оценке качества прогнозов по широкому диапазону порогов, подходит также для задач с несбалансированными классами.

Выбор метрики зависит от целей и особенностей конкретного проекта. Точность и полнота могут быть особенно важными в задачах, где отчуждение положительных примеров может иметь серьёзные последствия. В то же время, точность может быть лучшей метрикой для определения стабильных прогнозов в условиях, когда выборка сбалансирована. *ROC-AUC* может быть полезным в ситуациях, когда выборка несбалансированная, и в условиях, когда различия между классами менее очевидны. Т. е. в равной степени модель обучена работе с положительными и отрицательными примерами при существующем в обучающей выборке балансе классов.

## Сравнение качества моделей

После тренировки моделей используем метод *predict* и тестовый набор данных, чтобы получить прогнозы от каждой из моделей. Оценка работы классификаторов выполнена с помощью метрик, значения которых приведены в таблице 1.

*Таблица 1*

| Метод | accuracy % | roc-auc % | precision | recall |
|---|---|---|---|---|
| Логистическая регрессия | 80,2 | 83,3 | 0,66 | 0,526 |
| Метод опорных векторов | 80,2 | 79,6 | 0,705 | 0,435 |
| Случайные леса | 81,2 | 84,1 | 0,684 | 0,54 |
| Метод k-ближайших соседей | 78,7 | 79,3 | 0,654 | 0,415 |
| Градиентный бустинг | 80,5 | 84,8 | 0,67 | 0,528 |

Лучшие результаты по метрике качества roc-auc-score демонстрируют классификационные модели *GradientBoosting*, *LogisticRegression* и *RandomForest*. Так как после анализа была выбрана модель roc_auc_score, ориентироваться требуется на неё.

## Настройка работы в Airflow

Для автоматизации работы необходимо настроить Airflow. Основным рабочим элементом Airflow выступают файлы DAG (Directed Acyclic Graph – направленный

ациклический граф), представляющие собой инструкции, описывающие последовательность действий с данными [7]. Пользователь создаёт эти DAG'и на языке Python в удобном редакторе (IDE). DAG состоит из нескольких задач: импортируем необходимые библиотеки и инструменты для подключения к БД, задаём параметры запуска DAG'а, далее идут блоки задач, которые должен выполнять DAG, тут идёт обработка данных и предсказание. И, наконец, задаём последовательность выполнения операторов в DAG'е. После загрузки DAG отображается в списке Airflow под именем, заданным в параметре dag_id. При входе в Airflow отображается стартовая страница DAGs, где можно просмотреть список всех загруженных сценариев и краткую информацию по каждому из них. Основная функция Airflow не ограничивается только запуском, а также включает в себя мониторинг и управление работой DAG. Для этого существуют специальные инструменты. После запуска DAG можно видеть краткую статистику его работы. При переключении на имя DAG'а предоставляется расширенная информация. В каждой вкладке информационного окна можно ознакомиться с различными статистиками и данными.

При работе с большими объёмами данных, где каждый обучающий объект характеризуется множеством параметров (десятки и более), исчерпывающая обработка всего массива становится неэффективной как с точки зрения затрат времени, так и может негативно сказаться на точности анализа [2]. Поэтому разработка метода отбора ключевых параметров позволит определить наиболее значимые данные для дальнейшего анализа. Исходя из этого, принцип работы архитектуры для автоматизации непрерывной работы модели выглядит следующим образом: изначально берутся данные из базы данных PostgreSQL. Данные поступают на платформу для создания, планирования и мониторинга рабочих процессов Airflow, где с определённой периодичностью запускаются. Предложен следующий вариант архитектуры решения аналитической задачи (рисунок 2):

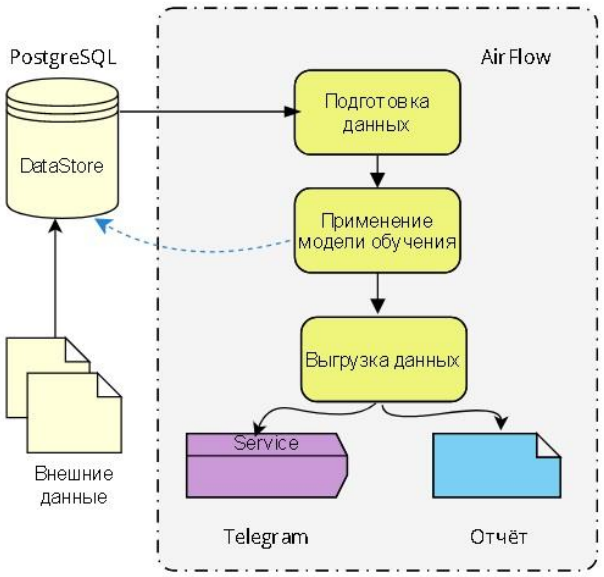



*Рисунок 2. Схема автоматизации работы*

Далее происходит обработка данных, после чего будет применяться обученная модель и происходить выгрузка данных в виде отчёта и уведомления в Telegram.

Исходя из информации было видно, что больше 20 % обучаемых понизили уровень вовлечённости в первые месяцы обучения. Таким ставим параметр schedule_interval='@monthly', этот параметр отвечает, как часто будет происходить запуск DAG'a, в нашем случае ежемесячно.

После запусков приходит оповещение в Telegram о том, что DAG отработал и загружен новый отчёт со списком обучаемых, которые могут снизить вовлечённость в процесс обучения.

## Заключение

Улучшение работы систем сбора и обработки данных требует обязательного внедрения методов прогнозирования временных рядов.

В качестве результатов можно также отметить разработанное приложение для определения вовлечённости в процесс обучения согласно предложенной архитектуре для автоматизации непрерывной работы модели. В процессе исследования описаны, проанализированы и обработаны данные по вовлечённости в обучение, проведено исследование зависимостей и формулирование гипотез. Построены модели для прогнозирования снижения вовлечённости, произведено сравнение и выбор наилучшей модели, а также описана настройка работы в AirFlow.

## Список литературы

1. *Афанасьев В. Н., Юзбашев М. М.* Анализ временных рядов и прогнозирование. М.: Финансы и статистика; ИНФРА-М, 2015. 320 с.

2. Журнал EduTech. Вовлечённость в обучение: разумные подходы к мотивации. 2020. № 6 (37). URL: https://sberuniversity.ru/upload/iblock/873/EduTech_37_web.pdf (дата обращения: 23.12.2024).

3. *Маккинни У.* Python и анализ данных. Первичная обработка данных с применением pandas, NumPy и Jupiter. М.: ДМК Пресс, 2023. 536 с.

4. *Мюллер А., Гвидо С.* Введение в машинное обучение с помощью Python. Руководство для специалистов по работе с данными / пер. с англ. А. В. Груздева. СПб: ООО «Альфа-книга», 2017. 480 с.

5. *Нильсен Э.* Практический анализ временных рядов: прогнозирование со статистикой и машинное обучение / пер. с англ. Д. А. Клюшина. СПб: ООО «Диалектика», 2021. 544 с.

6. *Харенслак Б., де Руйтер Дж.* Apache Airflow и конвейеры обработки данных / пер. с англ. Д. Беликова. М.: ДМК Пресс, 2021. 502 с.

## References

1. Afanasev V. N., Yuzbashev M. M. *Analiz vremennykh ryadov i prognozirovanie*. Moscow: Finansy i statistika; INFRA-M, 2015. 320 p.

2. *Zhurnal EduTech. Vovlechennost v obuchenie: razumnye podkhody k motivatsii*. 2020, No. 6 (37). Available at: https://sberuniversity.ru/upload/iblock/873/EduTech_37_web.pdf (accessed: 23.12.2024).

3. McKinney W. *Python i analiz dannykh. Pervichnaya obrabotka dannykh s primeneniem pandas, NumPy i Jupiter*. Moscow: DMK Press, 2023. 536 p.

4. Mueller A., Guido S. *Vvedenie v mashinnoe obuchenie s pomoshchyu Python. Rukovodstvo dlya spetsialistov po rabote s dannymi*. Transl. from English by A. V. Gruzdev. St. Petersburg: OOO “Alfa-kniga”, 2017. 480 p.

5. Nielsen A. *Prakticheskiy analiz vremennykh ryadov: prognozirovanie so statistikoy i mashinnoe obuchenie*. Transl. from English by D. A. Klyushin. St. Petersburg: OOO “Dialektika”, 2021. 544 p.

6. Harenslak B., de Ruiter J. *Apache Airflow i konveyery obrabotki dannykh*. Transl. from English by D. Belikov. Moscow: DMK Press, 2021. 502 p.