**Банковская деятельность**

**Содержание отчета**

Содержание

[1. Постановка задачи 2](#_Toc62831270)

[2. Исходный код 2](#_Toc62831271)

[3. Описание метода машинного обучения: тот который вы используете 2](#_Toc62831272)

[4. Техническое задание 3](#_Toc62831273)

[1) Общие сведения 3](#_Toc62831274)

[2) Назначение и цели создания ПО «Предложить банковский продукт?» 3](#_Toc62831275)

[3) Характеристика датасета 4](#_Toc62831276)

[4) Требования к системе 4](#_Toc62831277)

[5) Состав и содержание работ по созданию системы 4](#_Toc62831278)

[6) Порядок контроля и приемки разработки 4](#_Toc62831279)

[7) Требования к документированию 5](#_Toc62831280)

[5. Некоторые примеры работы 5](#_Toc62831281)

[6. Выводы 7](#_Toc62831282)

1. **Постановка задачи**

Предоставлен набор данных, содержащий информацию о клиентах банка и предыдущей маркетинговой кампании банка. Задача заключается в том, чтобы предсказать, откликнется ли клиент на предложение банка по срочному вкладу или нет. Датасет состоит из обезличенных данных примерно 45000 клиентов банка.

Система должна задать сотруднику несколько вопросов в консольном режиме и выдать ответ, нужно ли звонить или писать данному клиенту. При этом автоматически должен использоваться алгоритм обучения, предлагающий наибольшую точность с предложенным датасетом.

На вход система получает в консоли ответы на вопросы в формате целочисленных значений. В некоторых случаях ответы должны ограничиваться 1 для ответа “Правда” или 0 для “Ложь”. На выход система выдаст модальное окно с результатом: стоит ли тратить время на клиента, выделенное зелёным или красным цветом для ускорения реакции сотрудников.

1. **Исходный код**

Для того, что бы посмотреть используемый код можно перейти по ссылке <https://github.com/schamberg97/mgimo-task-3>

1. **Описание метода машинного обучения: тот который вы используете**

Используемые пакеты для Python 3:

pandas

PySimpleGUIQt

sklearn

matplotlib.pyplot

Целевое Программное Обеспечение реализовано в первую очередь на основе библиотеки sklearn (SciKit Learn) для Python 3. В начале работы ПО сначала читает CSV файл с данными с помощью pandas, а затем тренирует модели на основе 5 алгоритмов и выбирает наилучше подходящий для нашей задачи по точности (accuracy). Среди тестируемых алгоритмов:

* Логистическая регрессия (ЛР)
* Линейный дискриминантный анализ (ЛДА, англ. *Linear Discriminant Analysis*, *LDA*)
* K-ближайших соседей K-Nearest Neighbor (KNN)
* Классификация на основе деревьев решений
* Наивный байесовский классификатор — простой вероятностный классификатор, основанный на применении теоремы Байеса со строгими (наивными) предположениями о независимости.

Как правило, по точности случае выигрывают либо ЛДА либо ЛР.

1. **Техническое задание**
   1. **Общие сведения**
      1. **Полное наименование системы и ее условное обозначение**

Полное наименование системы – «Программное обеспечение для оценки интереса потенциального клиента к новым банковским продуктам». Краткое наименование - ПО «Предложить банковский продукт?».

* + 1. **Наименование предприятия (объединения) разработчика, заказчика (пользователя) разработки и их реквизиты**

**Заказчик системы:** Yakushin FinSourcing

**Разработчик системы:** Шамберг Николай Вячеславович, старший сотрудник по работе с технологиями машинного обучения Yakushin FinSourcing

* + 1. **Перечень документов, на основании которых создается разработка**

Данная система создается на основании задания на выполнение проектного задания и заявки заказчика.

* + 1. **Плановые сроки начала и окончания работы по созданию системы**

Начало работ по созданию ПЗ: 28 января 2021

Окончание работ по созданию ПЗ: 29 января 2021

* + 1. **Порядок оформления и предъявления заказчику результатов работ по созданию разработки программного комплекса**

Результаты работ по созданию работы оформляются в отчет о выполнении поставленного задания и отправляются заказчику в форме отчета, подписанного Квалифицированной Электронной Подписью, вместе со ссылкой на приватный git репозиторий.

* 1. **Назначение и цели создания ПО «Предложить банковский продукт?»**
     1. **Назначение ПО «Предложить банковский продукт?»**

Система разрабатывается для облегчения принятия решений по обзвону клиентов сотрудниками отдела маркетинга банков-клиентов Yakushin FinSourcing.

* + 1. **Цели и задачи создания ПО «Предложить банковский продукт?»**

Основной целью создания ПО «Предложить банковский продукт?» является оценка заинтересованности потенциального клиента в продуктах банков-партнеров.

* 1. **Характеристика датасета**

Датасет содержит около 45,000 обезличенных записей, содержащих персональные данные клиентов, а именно сведения о финансовом состоянии, наличии образования, долгах, кредитных дефолтах и т.д.

* 1. **Требования к системе**
     1. **Требования к функционированию ПО «Предложить банковский продукт?»**

Общение пользователя с системой должно происходить в диалоговом режиме в консоли.

* + 1. **Требования к вводу и выводу**

Данные в приложение будут вводиться в формате целочисленных значений. В некоторых случаях ответы должны быть 1 для ответа “Правда” или 0 для “Ложь”.

На выход система выдаст модальное окно с результатом: стоит ли тратить время на клиента, выделенное зелёным или красным цветом для ускорения реакции сотрудников.

* + 1. **Требования к программному обеспечению**

Для корректного функционирования ПО «Предложить банковский продукт» необходимо наличие компьютера с ОС Windows/macOS/Linux и установленным интерпретатором Python версии не ниже 3.8. Для компьютеров с ОС Linux может также требоваться X11/WayLand или другая оконная система.

* + 1. **Требования к техническому обеспечению**

Любой компьютер, способный запускать Python 3.8 и выше и имеющий 64-битный процессор и графический интерфейс.

* 1. **Состав и содержание работ по созданию системы**

Разработка ПО «Предложить банковский продукт» должна содержать следующие этапы:

* + 1. **Постановка задачи**
    2. **Описание машинного обучения**
    3. **Техническое задание**
    4. **Примеры предсказаний модели.**
    5. **Выводы**

По окончании всех видов работ составляется отчет, содержащий вышеуказанные разделы.

* 1. **Порядок контроля и приемки разработки**

Контроль и приемка работы должна происходить на 10 контрольных примерах, обеспечивающих просмотр работоспособности всех функций разработки. Готовая разработка представляется преподавателю и сопровождается отчетом.

* 1. **Требования к документированию**

Пояснительная записка к проекту должна содержать совокупность документов, структура и содержание которых регламентируется руководящим документом по стандартизации РД 50-34.698-90 (введенным 01.01.92) МЕТОДИЧЕСКИЕ УКАЗАНИЯ. ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ. АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ. ТРЕБОВАНИЯ К СОДЕРЖАНИЮ ДОКУМЕНТОВ.

Схемы алгоритмов, программ данных и систем, условные обозначения и правила выполнения регламентируются ГОСТ 19.701-90.

Общие требования к текстовым документам регламентируются ГОСТ 2.105-95

Техническое задание на создание автоматизированной системы выполняется на основании ГОСТ 34.602-89.

1. **Некоторые примеры работы**

Пример 1:

Text

Description automatically generatedGraphical user interface, text, application

Description automatically generated

Пример 2:Text

Description automatically generatedGraphical user interface, text, application, chat or text message

Description automatically generated

Пример 3:

Text

Description automatically generated

Graphical user interface, text, application, chat or text message

Description automatically generated

1. **Выводы**

Мы использовали примерно 45000 клиентских записей для обучения нейросети. В результате наших тестов нам удалось достичь точности, близкой к 90%. Учитывая, что речь идёт о реклмае банковских вкладах, данная точность приемлема для наших целей.

Сеть обучена, ввод принимается программой без ошибок. Программа исправна.