Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ _____ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ_____(ИУ)___

КАФЕДРА ____СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ___(ИУ5)__

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

К НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

НА ТЕМУ:

Разработка стратегии удержания сотрудников через построение модели оттока сотрудников

Студент ___**ИУ5-31М**____ _____ ____**А.А. Яценко**____
(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель _____ __**Ю.Е. Гапанюк**___
(Подпись, дата) (И.О.Фамилия)

Консультант _____ _____
(Подпись, дата) (И.О.Фамилия)

2021 г.

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

УТВЕРЖДАЮ

Заведующий кафедрой ИУ-5
(Индекс)
_____ В.М.Черненький
(И.О.Фамилия)
« _____ » _____ 2021 г.

З А Д А Н И Е
на выполнение научно-исследовательской работы

по теме Разработка стратегии удержания сотрудников через построение модели оттока сотрудников

Студент группы ИУ5-31М

Яценко Антон Александрович
(Фамилия, имя, отчество)

Направленность НИР (учебная, исследовательская, практическая, производственная, др.)

учебная

Источник тематики (кафедра, предприятие, НИР) НИР

График выполнения НИР: 25% к 4 нед., 50% к 8 нед., 75% к 12 нед., 100% к 16 нед.

Техническое задание Построить модель оттока сотрудников для разработки стратегии удержания, сформулировать главные вопросы, ответы на которые позволят приблизиться к пониманию того, когда и почему сотрудники более склонны к уходу из компании. Воспользоваться методами машинного обучения

Оформление научно-исследовательской работы:

Расчетно-пояснительная записка на 33 листах формата А4.
Перечень графического (иллюстративного) материала (чертежи, плакаты, слайды и т.п.)
Двадцать девять (29) рисунков

Дата выдачи задания « 08 » октября 2021 г.

Руководитель НИР _____ _____ г. Ю.Е. Гапанюк
(Подпись, дата И.О.Фамилия)

Студент _____ _____ г. А.А.Яценко
(Подпись, дата И.О.Фамилия)

Примечание: Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

ОГЛАВЛЕНИЕ

ЦЕЛЬ НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЫ

Целью научно-исследовательской работы является разработка стратегии удержания сотрудников через построение модели оттока сотрудников.

Необходимо сформулировать главные вопросы, ответы на которые позволят приблизиться к пониманию того, когда и почему сотрудники более склонны к уходу из компании.

Для получения ответов на поставленные вопросы воспользоваться методами машинного обучения.

ВВЕДЕНИЕ

Как уже было сказано в предыдущих научно-исследовательских работах, увольнение сотрудников обходится компании довольно дорого. Особенно при утечке кадров с высокооплачиваемых и руководящих должностей. Чтобы качественнее удерживать сотрудников в компании необходимо понять, когда и почему они более всего склонны к уходу.

Для этого будет использован пошаговый систематический подход, применимый для решения всевозможных задач машинного обучения.

Ниже приведены вопросы, на которые необходимо дать ответ:

- Каковы сходства и ключевые индикаторы покидающих компанию сотрудников?
- Какие решения и стратегии могут быть приняты для увеличения удержания кадров?

АНАЛИЗ ИСХОДНЫХ ДАННЫХ

В этой научно-исследовательской работе используется набор данных 1470 сотрудников для предсказания их ухода, опираясь на основные факторы текучки кадров. Этот набор данных сгенерирован датасентистами IBM (один из крупнейших производителей и поставщиков ПО, IT-сервисов и консалтинговых услуг) для демонстрации возможностей IBM Watson Analytics (инновационный когнитивный сервис) в отношении увольнения сотрудников.

Описание данных

Для начала импортируем датасет и копируем исходный файл для анализа. Датасет состоит из 1470 строк и 35 колонок (см. рис. 1), содержащих как числовые, так и категориальные колонки, в которых представлены персональные данные и данные о трудовом опыте (см. рис. 2). Сгруппируем колонки по типу данных (т.е. int64, float64, object) (см. рис. 3).

Источник данных

В представленном наборе данных отсутствуют пропуски, что не удивительно, поскольку отделы кадров обычно все данные о занятости и личные данные сотрудников. Не стоит забывать, что подход к сбору и хранению (Excel-файлы, бумажные носители, всевозможные базы данных и проч.) серьезно влияют на доступность и точность этих данных.

Обзор числовых признаков

На основе информации и гистограмм для числовых признаков (см. рис. 4) можно сделать несколько наблюдений:

- Некоторые признаки имеют распределение с «тяжелыми хвостами» с перекосом вправо. Например, кол-во лет в компании, расстояние от дома и месячный доход. Могут потребоваться методы преобразование данных

для приведения распределения к нормальному перед тренировкой модели.

- Распределения возраста сотрудников нормальное с небольшим перекосом вправо. Основная масса сотрудников от 25 до 40 лет.
- Кол-во рабочих часов и кол-во сотрудников одинаковы для всех, что сигнализирует об их избыточности вышеперечисленных двух признаков.
- Идентификатор сотрудника – это уникальный номер, из-за чего он получил псевдоравномерное распределение.

Распределение признаков относительно целевой переменной

Возраст

Распределение по возрасту для работающих и уволившихся сотрудников отличается всего на один год. Средний возраст составляет 33,6 и 37,6 лет для бывших и работающих сотрудников соответственно.

На рисунке 5 представлен график ядерной оценки плотности с разбивкой по целевому признаку. Ядерная оценка плотности - это непараметрический способ оценки функции плотности вероятности случайной величины.

Гендер

Гендерное распределение демонстрирует высокую относительную долю бывших сотрудников-мужчин, чем бывших сотрудников-женщин. При этом нормализованное гендерное распределение бывших сотрудников в наборе данных составляет 17,0% для мужчин и 14,8% для женщин (см. рис. 6).

Семейное положение

Датасет включает три значения для семейного положения: в браке (673 сотрудника), холост/незамужняя (470 сотрудников), разведен(а) (327

сотрудников). Больше всего увольняющихся среди одиноких сотрудников - 25% (см. рис. 7).

Должность и Условия работы

Предварительный взгляд на взаимосвязь между частотой командировок и статусом ухода показывает, что существует самая большая нормализованная доля уволенных среди сотрудников, которые «часто» путешествуют (см. рис. 8). Показатели командировок со статусом «Деловая поездка» не разглашаются (т.е. неизвестно сколько часов в поездках считается «часто»).

В датасете перечислены несколько должностей: руководитель отдела продаж, научный сотрудник, лаборант и т.д. (см. рис. 9)

Лет в компании и с момента последнего повышения

Среднее количество лет, проведенных в компании, для работающих и уволившихся сотрудников составляет 7,37 и 5,13 лет соответственно (см. рис. 10).

Лет с текущим руководителем

Среднее количество лет под руководством нынешнего начальника для работающих и уволившихся сотрудников составляет 4,37 и 2,85 лет соответственно (см. рис. 11).

Переработки

У некоторых сотрудников есть сверхурочная работа. Данные ясно показывают, что среди перерабатывающих сотрудников доля уволившихся значительно выше (см. рис. 12).

Месячный доход

Доход варьируется в диапазоне от $1009 до $19999 (см. рис. 13).

Целевая переменная: Усталость

Признак «Усталость» - это то, что необходимо предсказать, используя другие связанные признаки из персональных данных сотрудника и его профессиональных данных. В приложенном наборе данных 83,9% работающих сотрудников и 16,1% бывших сотрудников (см. рис. 14). Следовательно, на лицо проблема **несбалансированного класса**. Модели машинного обучения обычно работают лучше, когда число представителей каждого класса примерно одинаково. Нам придется устранить этот несбалансированность целевого признака до реализации алгоритма обучения.

Корреляция

Рассмотрим некоторые из наиболее значимых корреляций. Следует помнить, что коэффициенты корреляции измеряют только линейные корреляции.

Как показано на рисунке 15, признаки «Ставка», «Кол-во компаний, в которых работал» и «Удаленность от дома» имеют положительную корреляцию с признаком «Усталость». В то время как «Общий стаж», «Грейд» и «Лет в текущей должности имеют отрицательную корреляцию с признаком «Усталость».

РЕЗУЛЬТАТЫ АНАЛИЗА ДАННЫХ

- Нет отсутствующих или ошибочных значений, корректный тип данных у всех признаков.
- Сильные положительные корреляции с целевой переменной у признаков: «Рейтинг производительности», «Ставка», «Кол-во компаний, в которых работал» и «Удаленность от дома».
- Сильные отрицательные корреляции с целевой переменной у признаков: «Общий стаж», «Грейд», «Лет в текущей должности» и «Месячный доход».
- Датасет не сбалансирован по большинству наблюдений, описывающих работающих сотрудников.
- Неженатые сотрудники увольняются чаще.
- Около 10% уволившихся ушли после двух лет работы.
- Те, кто живут дальше от места работы, увольняются чаще.
- Те, кто чаще ездят в командировки, увольняются чаще.
- Те, кто работают сверхурочно, увольняются чаще.
- Те, кто ранее уже сменил несколько мест работы, увольняются чаще.

ПРЕПРОЦЕССИНГ ДАННЫХ

Ниже описан препроцессинг данных для подготовки датасета к реализации алгоритмов машинного обучения.

Кодирование

Сначала нам необходимо закодировать категориальные данные числовыми значениями, потому что модели не умеют работать с категориальными. Чтобы избежать повышения важности признаков с большим количеством уникальных значений, мы будем использовать **Label Encoding** и **One-Hot Encoding** (см. рис. 16 и 17).

Шкалирование

Шкалирование признаков с помощью **MinMaxScaler** существенно сужает диапазон значений. Модели машинного обучения работают лучше с входными числовыми значениями, находящимися в одном диапазоне. Зададим диапазон (0,5) (см. рис 18).

Разбивка данных на обучающую и тестовую выборки

Перед реализацией модели машинного обучения разделим исходный датасет на выборки для обучения и для тестирования (см. рис. 19).

ПОСТРОЕНИЕ МОДЕЛЕЙ МАШИННОГО ОБУЧЕНИЯ

Базовые модели

Ниже представлены базовые модели с использованием готовых гиперпараметров:

- Логистическая регрессия
- Случайный лес
- Метод опорных векторов
- Метод k-ближайших соседей
- Классификатор на базе дерева решений
- Гауссовский наивный байесовский классификатор

Оценим каждую модель по очереди и предоставим оценки точности и стандартного отклонения (см. рис. 20).

Так как в датасете данные не сбалансированы, то будет ошибкой использовать метрику «Точность классификации». Воспользуемся метрикой «Площадь под кривой ROC». Это метрика производительности для задач бинарной классификации. AUC показывает способность модели различать положительные и отрицательные классы и лучше подходит для текущего набора данных. Значение площади равное 1,0 соответствует идеальной модели, 0,5 - случайной модели (см. рис. 21).

Основываясь на сравнении метрики ROC AUC были выбраны логистическая регрессия и случайный лес для дальнейшей проработки, так как они показали наибольшие средние значения AUC.

Логистическая регрессия

Воспользуемся GridSearchCV для точной настройки гиперпараметров модели, перебирая определенные значения и используя метрику ROC AUC (см. рис. 22).

Матрица ошибок дает гораздо более подробное представление об оценке точности и о том, что происходит с метками – точно известно какие метки были спрогнозированы правильно, а какие нет. Точность классификатора на базе логистической регрессии составила на тестовой выборке 75,54% (см. рис. 23).

Вместо получения двоичных оценочных значений целевой переменной (0 или 1) можно предсказать вероятность. На выходе получились значения, в которых первое число - вероятность того, что сотрудник относится к классу «не увольняется» (0), а второе - вероятность того, что сотрудник относится к классу «готов уволиться» (1). Предсказывание вероятностей определенных меток позволяет измерить насколько сотрудник близок к уходу из компании (см. рис. 24).

Случайный лес

Так же настроим гиперпараметры с помощью GridSearchCV и метрики ROC AUC (см. рис. 25).

Случайный лес позволяет определить наиболее важные признаки для прогнозирования целевой переменной («Усталость). На рисунке 26 представлена диаграмма с признаками, отсортированными по их важности в порядке убывания.

Топ 10 самых важных признаков:

1. Месячный доход
2. Переработки
3. Возраст
4. Ставка
5. Расстояние от дома
6. Дневная ставка
7. Общий стаж

8. Лет в компании
9. Часовая ставка
10. Лет под руководством нынешнего начальника

Точность полученного классификатора Random Forest на тестовом наборе составила 86,14%. Матрица ошибок приведена на рисунке 27.

Предсказание вероятностей меток позволяет оценить склонность сотрудников к уходу из компании. Метрика ROC AUC для предсказаний вероятностей классификатором Случайный лес дает значение 0,818 (см. рис. 28).

Сравнение кривых ROC

Кривая ROC - это измерение производительности для задачи классификации при различных настройках пороговых значений. ROC - это кривая вероятности, а AUC - степень или мера разделимости классов. Она говорит о том, насколько модель способна различать классы. Зеленая линия на графике представляет собой кривую ROC для чисто случайного классификатора. Хороший классификатор находится как можно дальше от этой линии в сторону верхнего левого угла.

Из рисунка 29 видно, что у настроенной модели логистической регрессии значение AUC выше, чем у классификатора Случайный лес.

ЗАКЛЮЧЕНИЕ

Оценка риска

По мере того, как компания генерирует больше данных о своих сотрудниках - о новых сотрудниках и недавно уволенных - модель может быть переобучена с использованием этих дополнительных данных. И теоретически может давать более точные прогнозы для выявления сотрудников **с** высоким риском увольнения на основе вероятностной метки, присвоенной каждому сотруднику.

Сотрудникам можно присвоить атрибут «Оценка риска», полученный по результатам предсказанной метки:

- Низкий уровень риска для сотрудников с меткой < 0,6
- Средний уровень риска для сотрудников с меткой от 0,6 до 0,8
- Высокий уровень риска для сотрудников с меткой выше 0,8

Индикаторы и стратегия удержания

Самые важные индикаторы увольняющихся сотрудников включают в себя:

- **Месячный доход**. Люди с более высокой заработной платой реже уходят из компании. Следовательно, необходимо приложить усилия для сбора информации об отраслевых ориентирах на местном рынке труда, чтобы определить, обеспечивает ли компания конкурентоспособную заработную плату.
- **Переработки**. Люди, которые работают сверхурочно, чаще уходят из компании. Следовательно, должны быть предприняты усилия для заблаговременного определения масштабов проектов с соответствующей поддержкой и человеческими ресурсами, чтобы сократить переработки.
- **Возраст**. Чаще увольняются сотрудники относительно молодой возрастной группы 25–35 лет. Следовательно, необходимо приложить усилия, чтобы четко сформулировать долгосрочное видение компании и

молодых сотрудников, соответствующих этому видению, а также предоставить стимулы в виде, например, четких путей продвижения по службе.

- **Расстояние от дома**. Сотрудники, которым дальше добираться на место работы от дома, с большей вероятностью покинут компанию. Следовательно, можно оказать поддержку в виде трансфера за счет компании для групп сотрудников, выезжающих из одного района, или в форме транспортных надбавок. Первоначальная проверка сотрудников на предмет удаленности их места проживания, вероятно, не рекомендуется, так как это будет рассматриваться как форма дискриминации, тем более если сотрудники приходят на работу вовремя.
- **Общий стаж**. Более опытные сотрудники с меньшей вероятностью уйдут. Сотрудники со стажем от 5 до 8 лет потенциально подвержены более высокому риску увольнения.
- **Лет в компании**. Сотрудники, отработавшие в компании два года, подвержены более высокому риску увольнения.
- **Лет под руководством нынешнего начальника**. Много сотрудников ушли в течение шести месяцев после перехода под руководство нынешнего начальника. По данным о руководителе каждого сотрудника можно определить, у каких руководителей происходит больше всего увольнений.

Рассмотрим некоторые метрики, определяющие, следует ли предпринимать действия с линейным руководителем.

- Кол-во лет, в течение которых линейный руководитель остается на одной должности. Это может сигнализировать о том, что сотрудникам требуется тренинг по менеджменту или ментор, в идеале из топ-менеджмента.
- Паттерны уволившихся сотрудников: это может указывать на повторяющиеся закономерности увольнения, и в этом случае могут быть приняты соответствующие меры.

Заключительные мысли

Для каждой группы риска можно сформировать свой план удержания сотрудников. В дополнение к предлагаемым шагам для перечисленных выше индикаторов можно инициировать личные встречи сотрудника HR и сотрудников со средним и высоким уровнем риска для обсуждения условий работы. Кроме того, встреча с непосредственным руководителем этого сотрудника позволит обсудить рабочую среду в команде и можно ли предпринять шаги для ее улучшения.

ВЫВОДЫ

В ходе научно-исследовательской работы была построена модель оттока сотрудников для разработки стратегии удержания.

Были сформулированы главные вопросы, ответы на которые позволяют приблизиться к пониманию того, когда и почему сотрудники более склонны к уходу из компании.

Ответы на поставленные вопросы были получены с использованием машинного обучения.

ПРИЛОЖЕНИЕ 1

```
# Read Excel file
df_sourcefile = pd.read_excel(
    'Data/WA_Fn-UseC_-HR-Employee-Attrition.xlsx', sheet_name=0)
print("Shape of dataframe is: {}".format(df_sourcefile.shape))
```

```
Shape of dataframe is: (1470, 35)
```

```
# Make a copy of the original sourcefile
df_HR = df_sourcefile.copy()
```

Рис. 1 Набор данных

```
# Dataset columns
df_HR.columns
```

```
Index(['Age', 'Attrition', 'BusinessTravel', 'DailyRate', 'Department',
       'DistanceFromHome', 'Education', 'EducationField', 'EmployeeCount',
       'EmployeeNumber', 'EnvironmentSatisfaction', 'Gender', 'HourlyRate',
       'JobInvolvement', 'JobLevel', 'JobRole', 'JobSatisfaction',
       'MaritalStatus', 'MonthlyIncome', 'MonthlyRate', 'NumCompaniesWorked',
       'Over18', 'OverTime', 'PercentSalaryHike', 'PerformanceRating',
       'RelationshipSatisfaction', 'StandardHours', 'StockOptionLevel',
       'TotalWorkingYears', 'TrainingTimesLastYear', 'WorkLifeBalance',
       'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion',
       'YearsWithCurrManager'],
      dtype='object')
```

```
# Dataset header
df_HR.head()
```

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField
0	41	Yes	Travel_Rarely	1102	Sales	1	2	Life Sciences
1	49	No	Travel_Frequently	279	Research & Development	8	1	Life Sciences
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2	Other
3	33	No	Travel_Frequently	1392	Research & Development	3	4	Life Sciences
4	27	No	Travel_Rarely	591	Research & Development	2	1	Medical

Рис. 2 Колонки датасета

ПРИЛОЖЕНИЕ 2

```
df_HR.columns.to_series().groupby(df_HR.dtypes).groups
```

```
{dtype('int64'): Index(['Age', 'DailyRate', 'DistanceFromHome', 'Education', 'EmployeeCount',
       'EmployeeNumber', 'EnvironmentSatisfaction', 'HourlyRate',
       'JobInvolvement', 'JobLevel', 'JobSatisfaction', 'MonthlyIncome',
       'MonthlyRate', 'NumCompaniesWorked', 'PercentSalaryHike',
       'PerformanceRating', 'RelationshipSatisfaction', 'StandardHours',
       'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear',
       'WorkLifeBalance', 'YearsAtCompany', 'YearsInCurrentRole',
       'YearsSinceLastPromotion', 'YearsWithCurrManager'],
      dtype='object'),
 dtype('O'): Index(['Attrition', 'BusinessTravel', 'Department', 'EducationField', 'Gender',
       'JobRole', 'MaritalStatus', 'Over18', 'OverTime'],
      dtype='object')}
```

Рис. 3 Группировка колонок

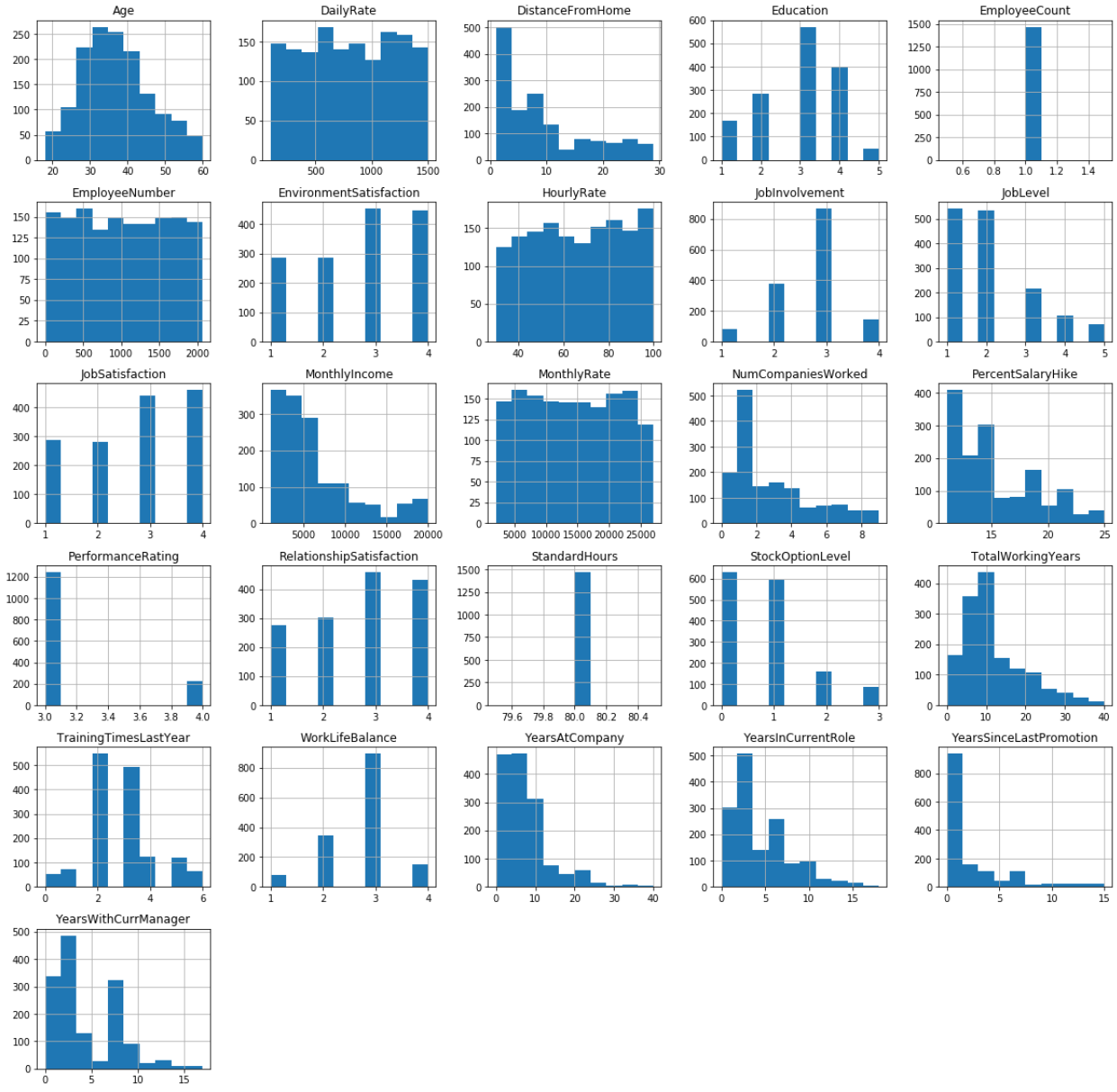

Рис. 4 Гистограммы для числовых признаков

ПРИЛОЖЕНИЕ 3

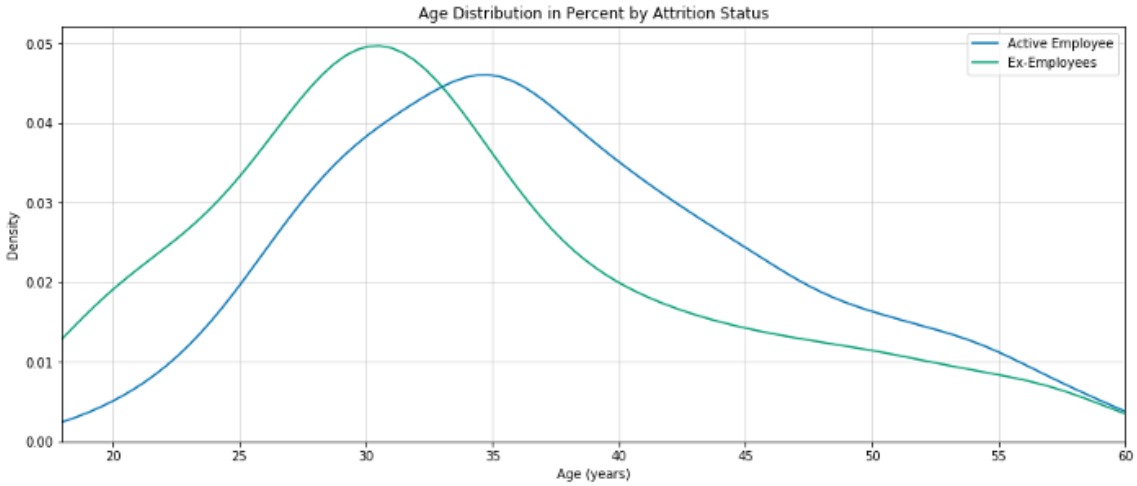

Рис. 5 График ядерной оценки плотности возраста с разбивкой по целевому признаку

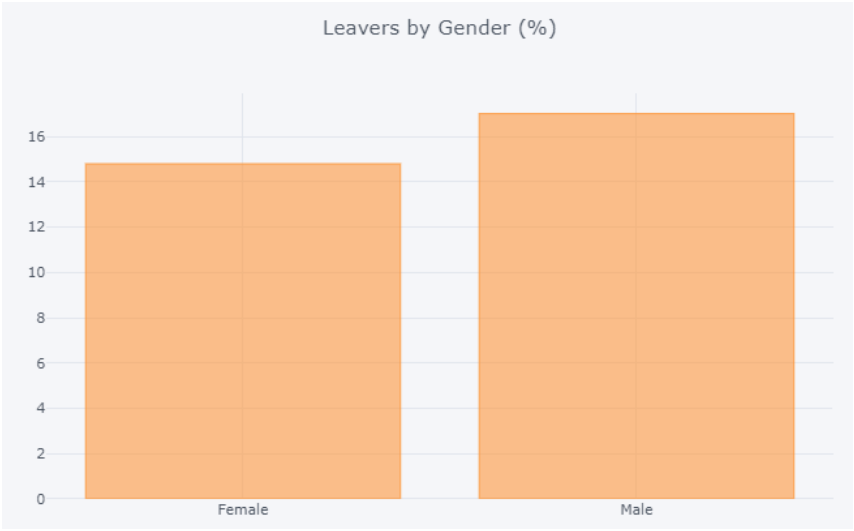

Рис. 6 Гендерное распределение бывших сотрудников

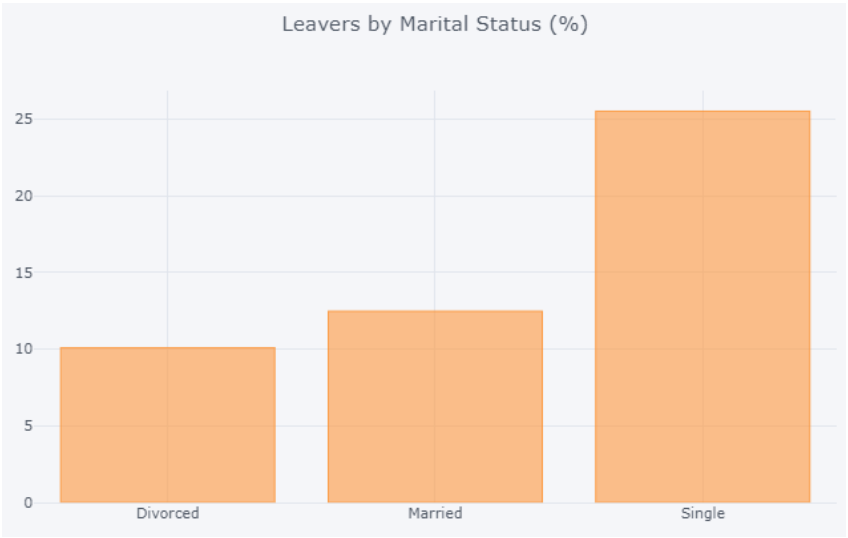

Рис. 7 Распределение бывших сотрудников по семейному положению

ПРИЛОЖЕНИЕ 4

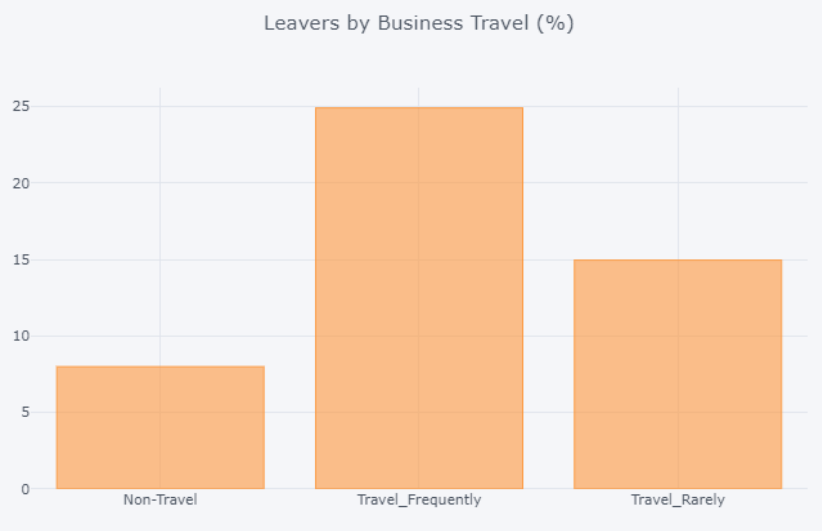

Рис. 8 Распределение бывших сотрудников по командировкам

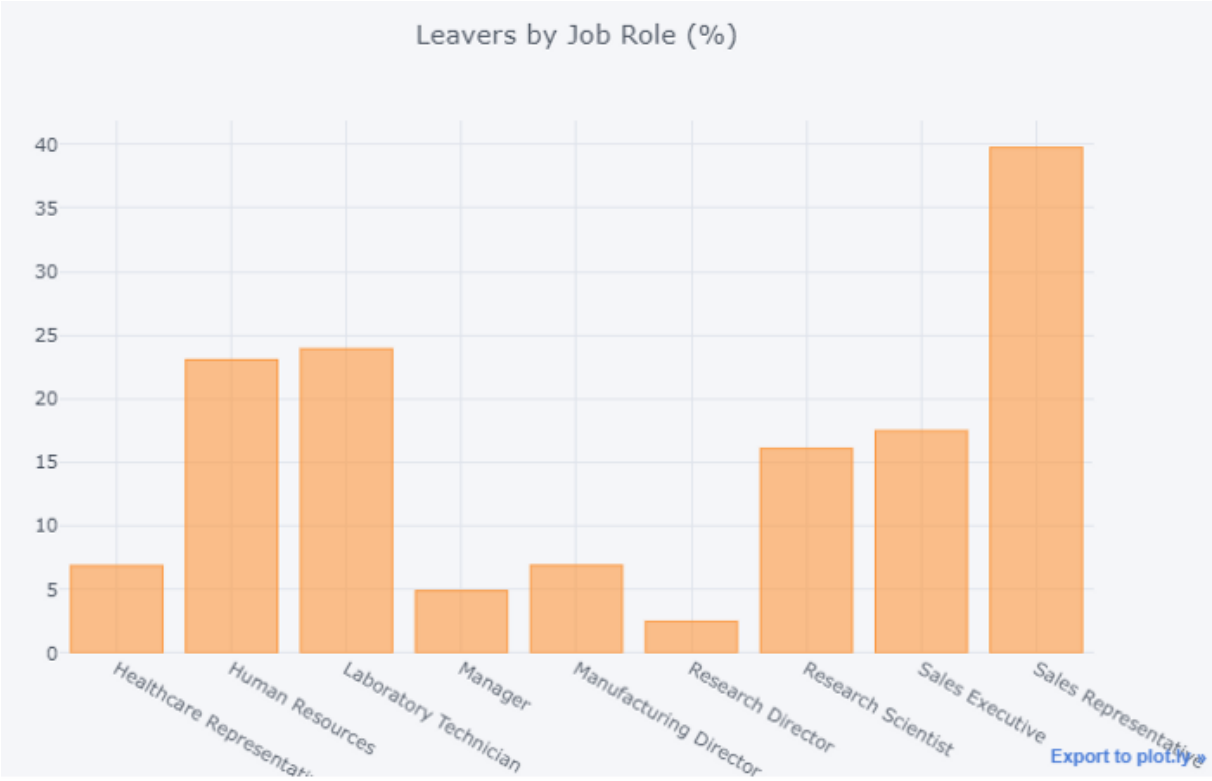

Рис. 9 Распределение бывших сотрудников по должности

ПРИЛОЖЕНИЕ 5

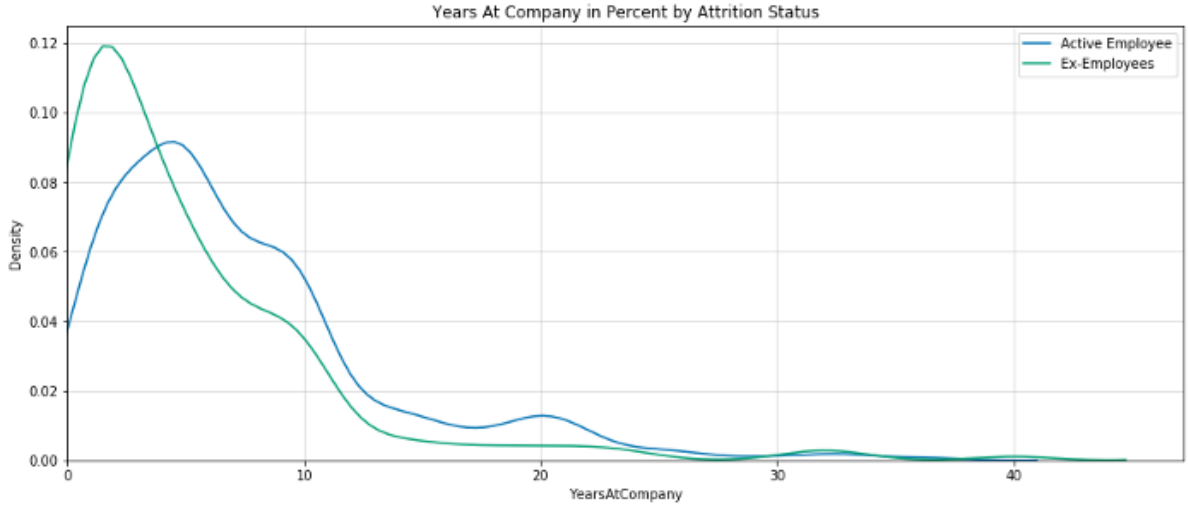

Рис. 10 График ядерной оценки плотности лет в компании с разбивкой по целевому признаку

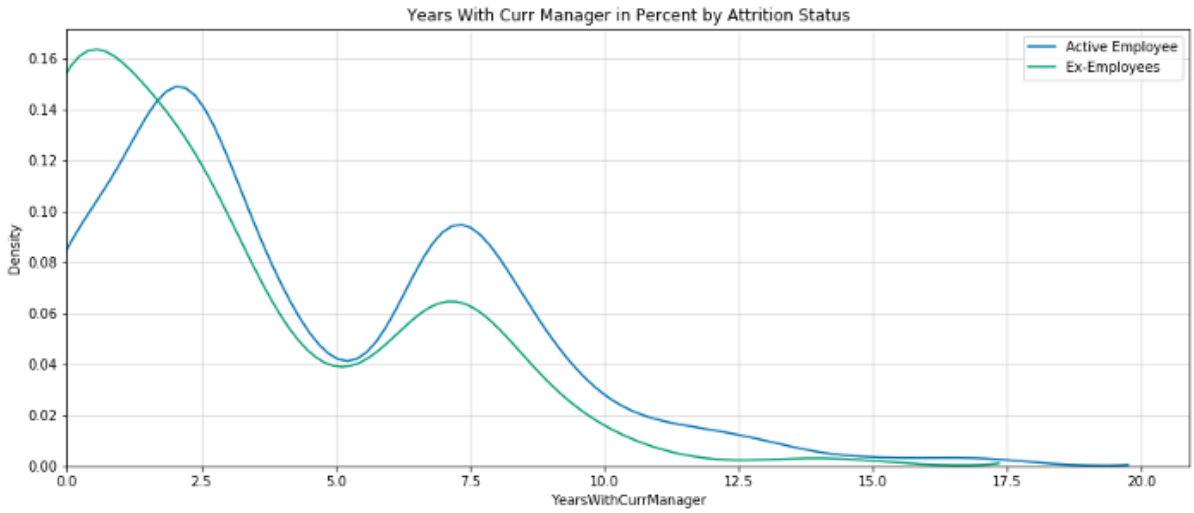

Рис. 11 График ядерной оценки плотности лет с текущим руководителем с разбивкой по целевому признаку

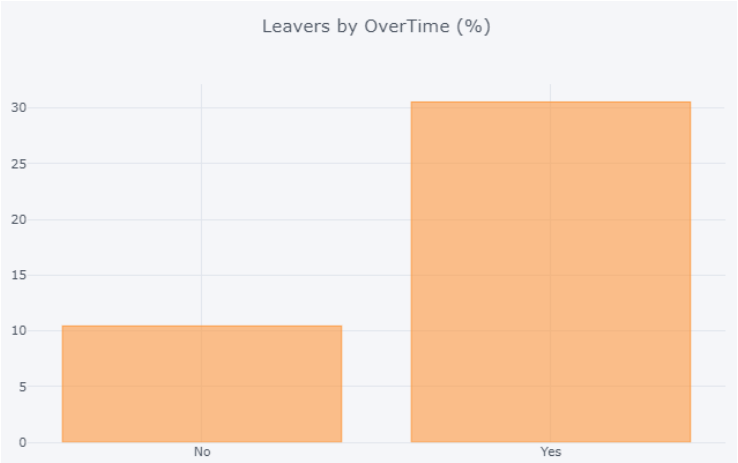

Рис. 12 Распределение бывших сотрудников по переработкам

ПРИЛОЖЕНИЕ 6

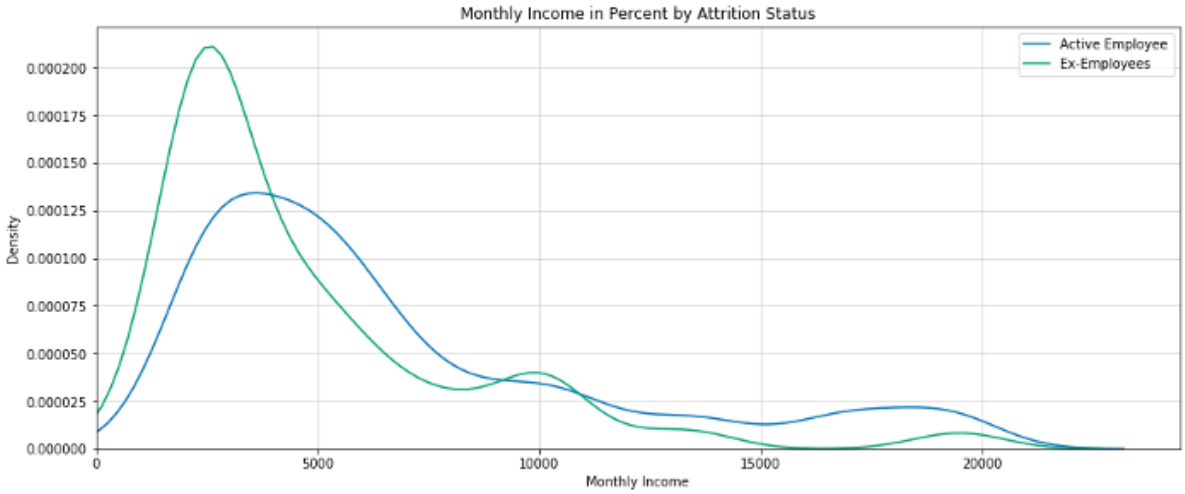

Рис. 13 График ядерной оценки плотности месячного дохода с разбивкой по целевому признаку

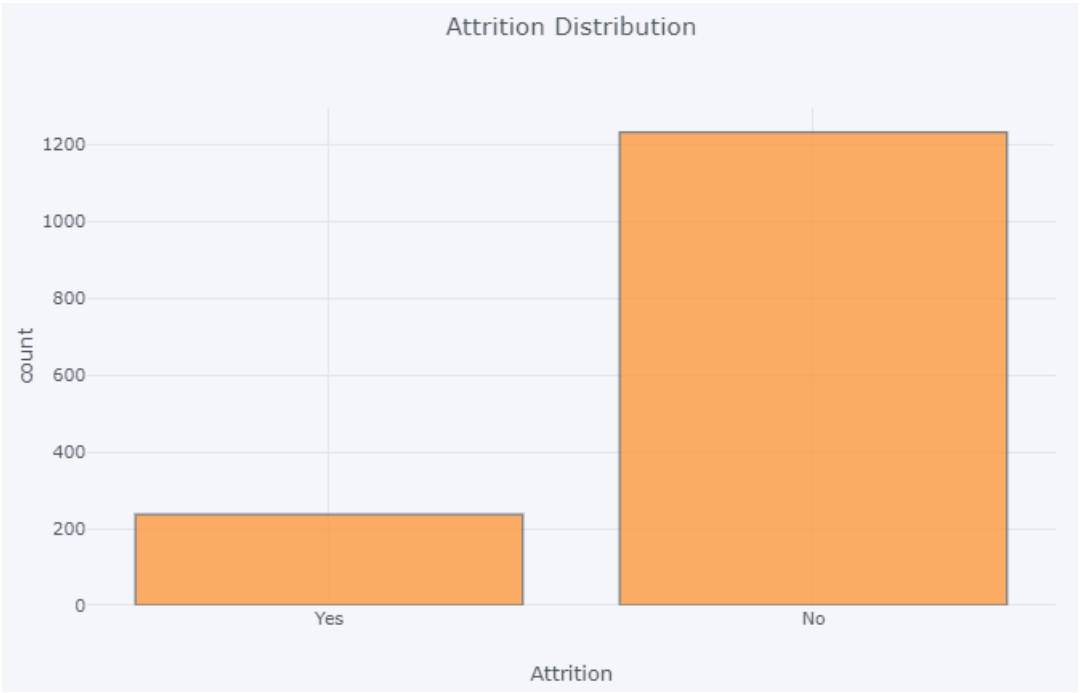

Рис. 14 Целевая переменная - усталость

ПРИЛОЖЕНИЕ 7

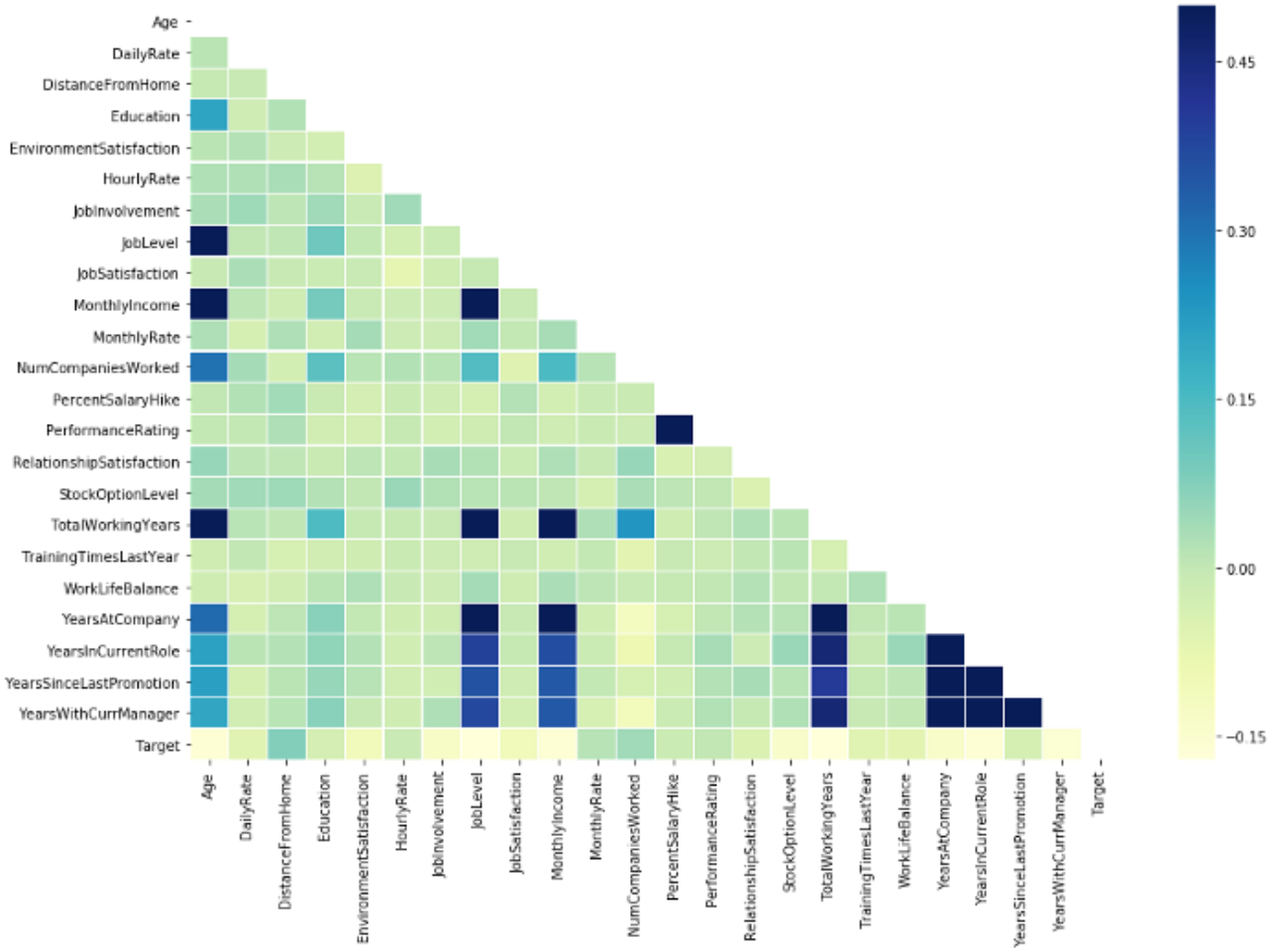

Рис. 15 Корреляции признаков

```
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
# Create a label encoder object
le = LabelEncoder()
```

```
print(df_HR.shape)
df_HR.head()
```

```
(1470, 35)
```

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	EmployeeCount	Emplo
0	41	Yes	Travel_Rarely	1102	Sales	1	2	Life Sciences	1	1
1	49	No	Travel_Frequently	279	Research & Development	8	1	Life Sciences	1	2
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2	Other	1	4
3	33	No	Travel_Frequently	1392	Research & Development	3	4	Life Sciences	1	5
4	27	No	Travel_Rarely	591	Research & Development	2	1	Medical	1	7

Рис. 16 Кодирование категориальных признаков

ПРИЛОЖЕНИЕ 8

```
# Label Encoding will be used for columns with 2 or less unique values
le_count = 0
for col in df_HR.columns[1:]:
    if df_HR[col].dtype == 'object':
        if len(list(df_HR[col].unique())) <= 2:
            le.fit(df_HR[col])
            df_HR[col] = le.transform(df_HR[col])
            le_count += 1
print('{} columns were label encoded.'.format(le_count))
```

```
4 columns were label encoded.
```

```
# convert rest of categorical variable into dummy
df_HR = pd.get_dummies(df_HR, drop_first=True)
```

> The resulting dataframe has **49 columns** for 1,470 employees.

```
print(df_HR.shape)
df_HR.head()
```

```
(1470, 49)
```

	Age	Attrition	DailyRate	DistanceFromHome	Education	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	Gender	Ho
0	41	1	1102	1	2	1	1	2	0	94
1	49	0	279	8	1	1	2	3	1	61
2	37	1	1373	2	2	1	4	4	1	92

Рис. 17 LabelEncoder и OneHotEncoder

```
# import MinMaxScaler
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler(feature_range=(0, 5))
HR_col = list(df_HR.columns)
HR_col.remove('Attrition')
for col in HR_col:
    df_HR[col] = df_HR[col].astype(float)
    df_HR[[col]] = scaler.fit_transform(df_HR[[col]])
df_HR['Attrition'] = pd.to_numeric(df_HR['Attrition'], downcast='float')
df_HR.head()
```

	Age	Attrition	DailyRate	DistanceFromHome	Education	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	Gender
0	2.738095	1.0	3.579098	0.000000	1.25	0.0	0.000000	1.666667	0.0
1	3.690476	0.0	0.633500	1.250000	0.00	0.0	0.002419	3.333333	5.0
2	2.261905	1.0	4.549034	0.178571	1.25	0.0	0.007257	5.000000	5.0
3	1.785714	0.0	4.617037	0.357143	3.75	0.0	0.009676	5.000000	0.0
4	1.071429	0.0	1.750179	0.178571	0.00	0.0	0.014514	0.000000	5.0

```
print('Size of Full Encoded Dataset: {}'. format(df_HR.shape))
```

```
Size of Full Encoded Dataset: (1470, 49)
```

Рис. 18 Шкалирование через MinMaxScaler

ПРИЛОЖЕНИЕ 9

```
# assign the target to a new dataframe and convert it to a numerical feature
#df_target = df_HR[['Attrition']].copy()
target = df_HR['Attrition'].copy()
```

```
# Since we have class imbalance (i.e. more employees with turnover=0 than turnover=1)
# let's use stratify=y to maintain the same ratio as in the training dataset when splitting the d
ataset
X_train, X_test, y_train, y_test = train_test_split(df_HR,
                                                    target,
                                                    test_size=0.25,
                                                    random_state=7,
                                                    stratify=target)
print("Number transactions X_train dataset: ", X_train.shape)
print("Number transactions y_train dataset: ", y_train.shape)
print("Number transactions X_test dataset: ", X_test.shape)
print("Number transactions y_test dataset: ", y_test.shape)
```

```
Number transactions X_train dataset:  (1102, 44)
Number transactions y_train dataset:  (1102,)
Number transactions X_test dataset:  (368, 44)
Number transactions y_test dataset:  (368,)
```

Рис. 19 Разбивка на обучающую и тестовую выборки

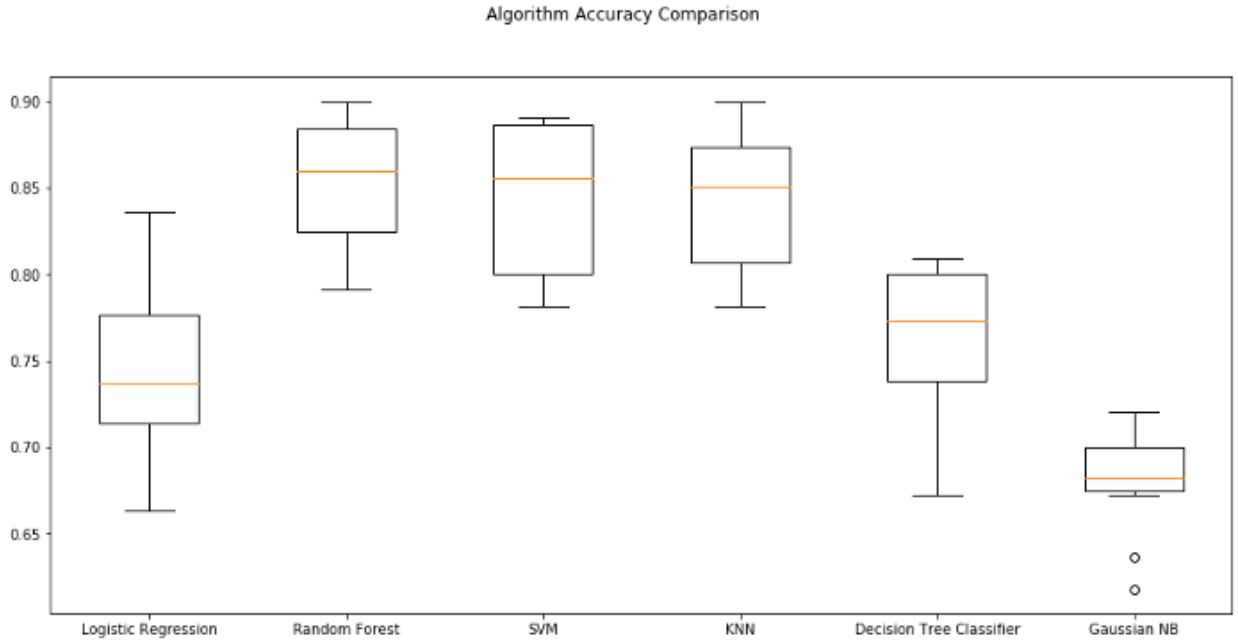

Рис. 20 Сравнение оценок точности и стандартных отклонений моделей

ПРИЛОЖЕНИЕ 10

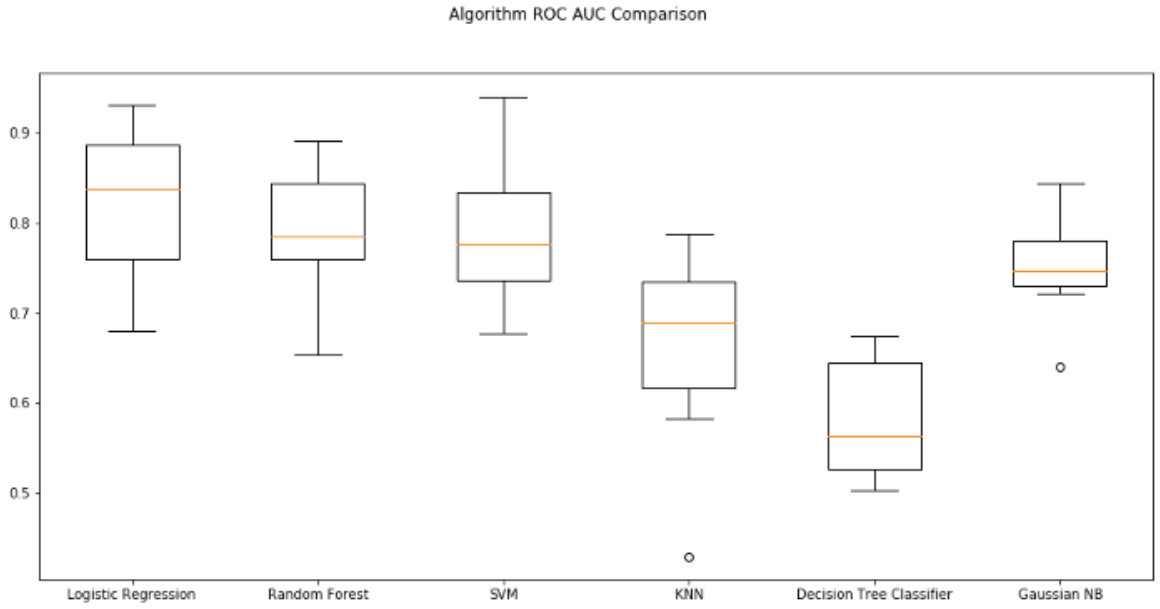

Рис. 21 Сравнение по метрике «Площадь под кривой ROC»

```
param_grid = {'C': np.arange(1e-03, 2, 0.01)} # hyper-parameter list to fine-tune
log_gs = GridSearchCV(LogisticRegression(solver='liblinear', # setting GridSearchCV
                                         class_weight="balanced",
                                         random_state=7),
                      iid=True,
                      return_train_score=True,
                      param_grid=param_grid,
                      scoring='roc_auc',
                      cv=10)

log_grid = log_gs.fit(X_train, y_train)
log_opt = log_grid.best_estimator_
results = log_gs.cv_results_

print('='*20)
print("best params: " + str(log_gs.best_estimator_))
print("best params: " + str(log_gs.best_params_))
print('best score:', log_gs.best_score_)
print('='*20)
```

```
====================
best params: LogisticRegression(C=0.05099999999999999, class_weight='balanced', dual=False,
          fit_intercept=True, intercept_scaling=1, max_iter=100,
          multi_class='warn', n_jobs=None, penalty='l2', random_state=7,
          solver='liblinear', tol=0.0001, verbose=0, warm_start=False)
best params: {'C': 0.05099999999999999}
best score: 0.8180815631000706
====================
```

Рис. 22 GridSearchCV для точной настройки гиперпараметров модели логистической регрессии

ПРИЛОЖЕНИЕ 11

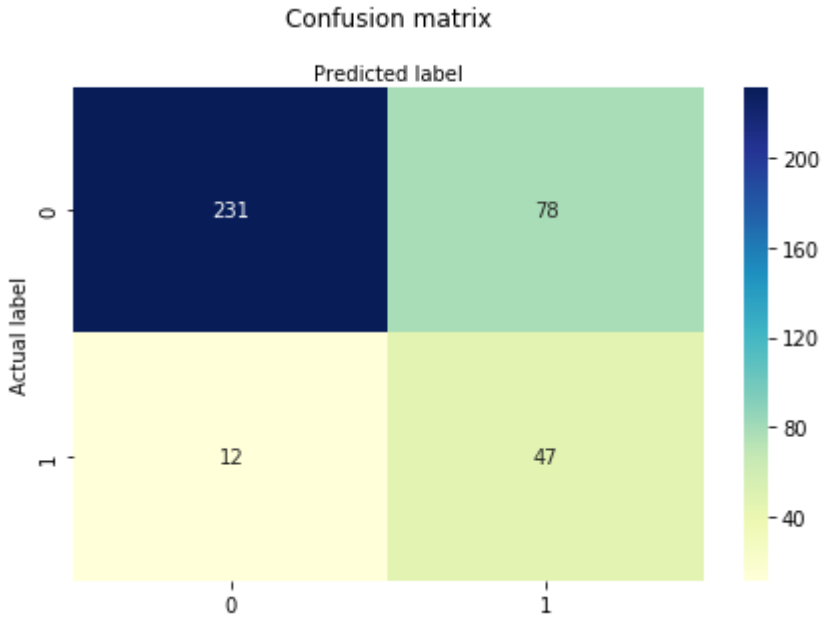

Рис. 23 Матрица ошибок для логистической регрессии

```
log_opt.fit(X_train, y_train) # fit optimised model to the training data
probs = log_opt.predict_proba(X_test) # predict probabilities
probs = probs[:, 1] # we will only keep probabilities associated with the employee leaving
logit_roc_auc = roc_auc_score(y_test, probs) # calculate AUC score using test dataset
print('AUC score: %.3f' % logit_roc_auc)
```

```
AUC score: 0.857
```

Рис. 24 Предсказывание вероятностей определенных меток

ПРИЛОЖЕНИЕ 12

```
rf_classifier = RandomForestClassifier(class_weight = "balanced",
                                       random_state=7)
param_grid = {'n_estimators': [50, 75, 100, 125, 150, 175],
              'min_samples_split':[2,4,6,8,10],
              'min_samples_leaf': [1, 2, 3, 4],
              'max_depth': [5, 10, 15, 20, 25]}

grid_obj = GridSearchCV(rf_classifier,
                        iid=True,
                        return_train_score=True,
                        param_grid=param_grid,
                        scoring='roc_auc',
                        cv=10)

grid_fit = grid_obj.fit(X_train, y_train)
rf_opt = grid_fit.best_estimator_

print('='*20)
print("best params: " + str(grid_obj.best_estimator_))
print("best params: " + str(grid_obj.best_params_))
print('best score:', grid_obj.best_score_)
print('='*20)
```

```
====================
best params: RandomForestClassifier(bootstrap=True, class_weight='balanced',
            criterion='gini', max_depth=15, max_features='auto',
            max_leaf_nodes=None, min_impurity_decrease=0.0,
            min_impurity_split=None, min_samples_leaf=1,
            min_samples_split=8, min_weight_fraction_leaf=0.0,
            n_estimators=75, n_jobs=None, oob_score=False, random_state=7,
            verbose=0, warm_start=False)
best params: {'max_depth': 15, 'min_samples_leaf': 1, 'min_samples_split': 8, 'n_estimators': 7
5}
best score: 0.7956083711198764
====================
```

Рис. 25 GridSearchCV и метрики ROC AUC для случайного леса

ПРИЛОЖЕНИЕ 13

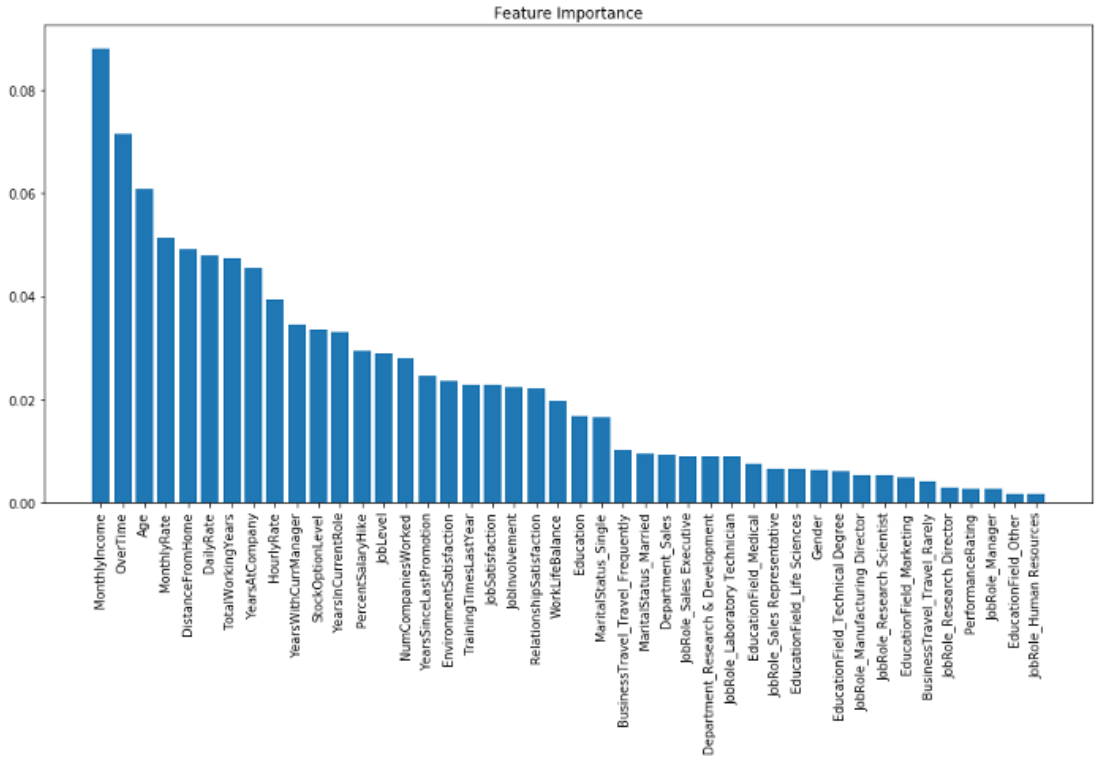

Рис. 26 диаграмма с признаками, отсортированными по их важности в порядке убывания

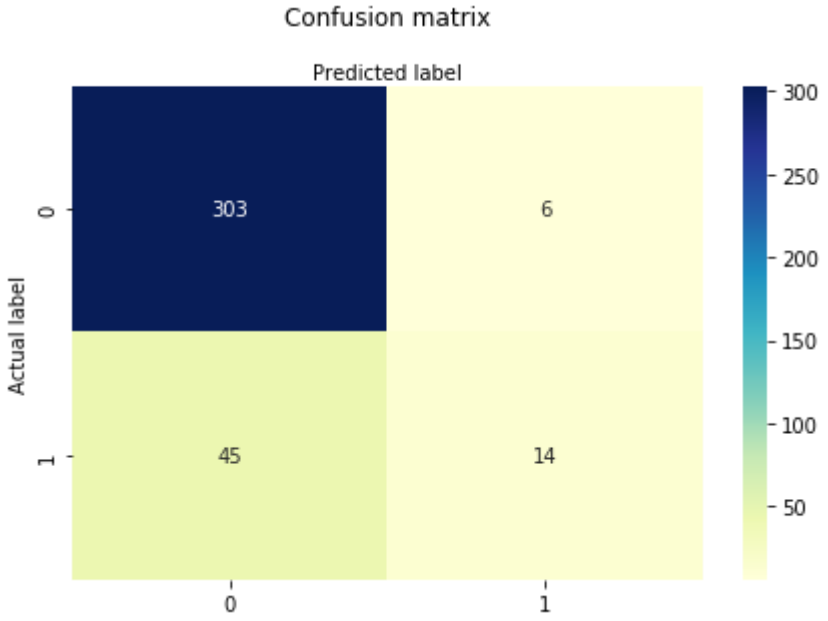

Рис. 27 Матрица ошибок для случайного леса

ПРИЛОЖЕНИЕ 13

```
rf_opt.fit(X_train, y_train) # fit optimised model to the training data
probs = rf_opt.predict_proba(X_test) # predict probabilities
probs = probs[:, 1] # we will only keep probabilities associated with the employee leaving
rf_opt_roc_auc = roc_auc_score(y_test, probs) # calculate AUC score using test dataset
print('AUC score: %.3f' % rf_opt_roc_auc)
```

```
AUC score: 0.818
```

Рис. 28 Метрика ROC AUC для предсказаний вероятностей классификатором Случайный лес

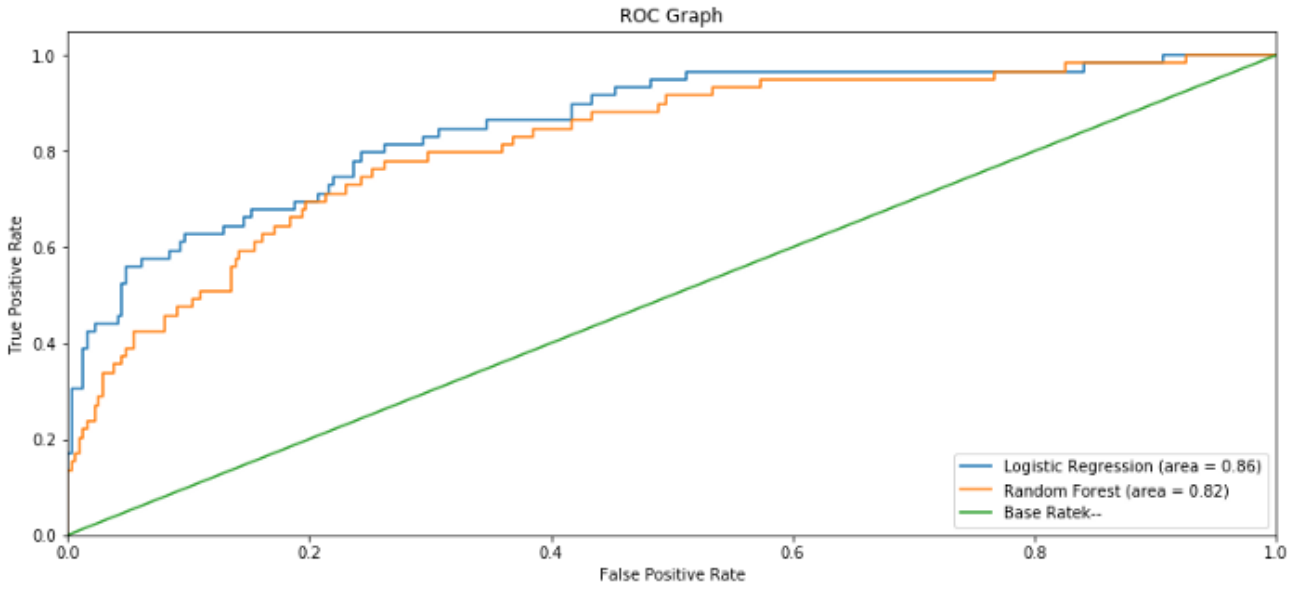

Рис. 29 Сравнение кривых ROC

СПИСОК ЛИТЕРАТУРЫ

1. Гапанюк Ю.Е. Методы машинного обучения; Лекционный материал [Электронный ресурс]. - https://github.com/ugapanyuk/ml_course_2021/wiki/COURSE_MMO, 10.12.2021

2. Себастьян Р. Python и машинное обучение, ЛитРес, 2015. – 356 с.

3. Гапанюк Ю.Е. Технологии машинного обучения; Лекционный материал [Электронный ресурс]. - https://github.com/ugapanyuk/ml_course_2021/wiki/COURSE_TMO, 10.12.2021

4. Маккини У. Python и анализ данных, ДМК Пресс, 2020. – 540с.

5. Элбон К. Машинное обучение с использованием Python. Сборник рецептов, BHV, 2019. – 384с.