



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА - Российский технологический университет»

РТУ МИРЭА

Институт кибернетики

Кафедра высшей математики

ОТЧЁТ ПО Научно-Исследовательской Работе
(указать вид практики)

Тема практики: Анализ употребления алкоголя учащимися (kaggle.com)
приказ университета о направлении на практику
490 – С от 09.02.2021 г.

Отчет представлен к
рассмотрению:
Студент группы КМБО-03-
20

Балашов Д.С.
(расшифровка подписи)
«11» июля 2021г.

Отчет утвержден.
Допущен к защите:

Руководитель практики от
кафедры

Петрусевич Д.А.
(расшифровка подписи)
«3» июля 2021г.

Москва 2021



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА - Российский технологический университет»

РТУ МИРЭА

ЗАДАНИЕ НА Научно-Исследовательскую Работу

Студенту 1 курса учебной группы КМБО-03-20 института кибернетики Балашову
Дмитрию Сергеевичу

(фамилия, имя и отчество)

Место и время практики: Институт кибернетики, кафедра высшей математики

Время практики: с «09» февраля 2021 по «31» мая 2021

Должность на практике: практикант

1. ЦЕЛЕВАЯ УСТАНОВКА: изучение основ анализа данных и машинного обучения

2. СОДЕРЖАНИЕ ПРАКТИКИ:

2.1 Изучить: литературу и практические примеры по темам: 1) построение линейной регрессии, 2) использование метода главных компонент, 3) поиск и устранение линейной зависимости в данных, 4) основы нормализации данных, 5) методы классификации и кластеризации («решающее дерево», «случайный лес», «k ближайших соседей»).

2.2 Практически выполнить: 1) снижение размерности исходных задач при помощи метода главных компонент при возможности; построение линейной регрессии для некоторого параметра, исключение регрессоров, не коррелирующих с объясняемой переменной; решение задачи классификации или кластеризации на основе открытого набора данных с ресурса kaggle.com

2.3 Ознакомиться: с применением метода главных компонент; методов классификации («решающего дерева», «случайного леса»); методов кластеризации («k ближайших соседей»); построением модели линейной регрессии

3. ДОПОЛНИТЕЛЬНОЕ ЗАДАНИЕ: анализ употребления алкоголя учащимися (kaggle.com).

4. ОГРАНИЗАЦИОННО-МЕТОДИЧЕСКИЕ УКАЗАНИЯ: построить прогноз итогового бала учащегося. Насколько сильно влияет употребление алкоголя? Какие параметры вносят наибольший вклад в предсказание? Построить графики статистической оценки параметров. Применить алгоритмы кластеризации. Что общего между объектами в каждом кластере?

Заведующий кафедрой
высшей математики

Ю.И.Худак

«09» февраля 2021г.

СОГЛАСОВАНО

Руководитель практики от кафедры:

«09» февраля 2021 г.

(подпись)

(Петрусеvич Д.А.)

(фамилия и инициалы)

Задание получил:

«09» февраля 2021 г.

(подпись)

(Балашов Д.С.)

(фамилия и инициалы)



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА - Российский технологический университет»
РТУ МИРЭА

РАБОЧИЙ ГРАФИК ПРОВЕДЕНИЯ Научно-Исследовательской Работы

студента Балашова Д.С. 1 курса группы КМБО-03-20 очной формы обучения,
обучающегося по направлению подготовки 01.03.02 «Прикладная математика и
информатика»,
профиль «Математическое моделирование и вычислительная математика»

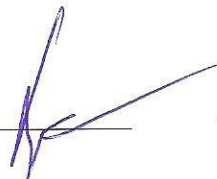
Неделя	Сроки выполнения	Этап	Отметка о выполнении
1	09.02.2021	Выбор темы НИР. Пройти инструктаж по технике безопасности	✓
1	09.02.2021	Вводная установочная лекция	✓
1	13.02.2021	Построение и оценка парной регрессии с помощью языка R	✓
2	20.02.2021	Построение и оценка множественной регрессии с помощью языка R	✓
3	27.02.2021	Построение доверительных интервалов. Обработка факторных переменных. Мультиколлинеарность	✓
4	06.03.2021	Гетероскедастичность	✓
5	13.03.2021	Классификация	✓
7	27.03.2021	Кластеризация. Предобработка данных	✓
9	10.04.2021	Метод главных компонент	✓
17	05.06.2021	Представление отчётных материалов по НИР и их защита. Передача обобщённых	✓

		материалов на кафедру для архивного хранения	
		Зачётная аттестация	

Содержание практики и планируемые результаты согласованы с руководителем практики от профильной организации.

Согласовано:

Заведующий кафедрой



/ ФИО /

Худак Ю.И.

Руководитель практики
от кафедры



/ ФИО /

Петрусевич Д.А.

Обучающийся



/ ФИО /

Балашов Д.С.

ИНСТРУКТАЖ ПРОВЕДЕН:

Вид мероприятия	ФИО ответственного, подпись, дата	ФИО студента, подпись, дата
Охрана труда	Петрусеви́ч Д.А.  «09» февраля 2021 г.	Балашов Д.С.  «09» февраля 2021 г.
Техника безопасности	Петрусеви́ч Д.А.  «09» февраля 2021 г.	Балашов Д.С.  «09» февраля 2021 г.
Пожарная безопасность	Петрусеви́ч Д.А.  «09» февраля 2021 г.	Балашов Д.С.  «09» февраля 2021 г.
Правила внутреннего распорядка	Петрусеви́ч Д.А.  «09» февраля 2021 г.	Балашов Д.С.  «09» февраля 2021 г.

Оглавление:

Задача 1.....	3
Задача 2.1.....	5
Задача 2.2.....	10
Задача 3	12
Задача 4	20
Задача 5	24
Задача 6	32
Заключение.....	40
Список литературы	41
Приложение.....	42

Задача №1

Необходимо загрузить данные из указанного набора и произвести следующие действия.

Набор данных: Swiss.

Объясняемая переменная: *Agriculture*.

Регрессоры: *Fertility*, *Examination*.

№1. Оценить среднее значение, дисперсию и СКО переменных, указанных во втором и в третьем столбце.

➤ *Agriculture*:

- Дисперсия = 515.799 (большой разброс)
- СКО = 22.71
- Среднее арифметическое = 50.66

➤ *Fertility*:

- Дисперсия = 156.04 (средний разброс)
- СКО = 12.49
- Среднее арифметическое = 70.14

➤ *Examination*:

- Дисперсия = 63.65 (маленький разброс)
- СКО = 7.98
- Среднее арифметическое = 16.49

№2. Построить зависимости вида $y = a + bx$, где y – объясняемая переменная, x – регрессор.

Таблица 1. Характеристики модели зависимости параметра *Agriculture* от параметра *Examination* в наборе данных Swiss.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	82.88	5.64	14.69	< 2e-16	***
<i>Examination</i>	-1.95	0.31	-6.33	9.95e - 08	***

$$\# Agriculture = -1.95 * Examination + 82.88$$

Таблица 2. Характеристики модели зависимости параметра *Agriculture* от параметра *Fertility* в наборе данных Swiss.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	5.63	18.06	0.31	0.76	
<i>Fertility</i>	0.64	0.25	2.53	0.01	*

$$\# Agriculture = 0.64 * Fertility + 5.63$$

№3. Оценить, насколько «хороша» модель по коэффициенту детерминации R^2 .

➤ $Agriculture = -1.95 * Examination + 82.88$:

$R^2 = 47\%$, что довольно неплохой показатель, так что можно сказать, что *Сельское Хозяйство* отрицательно зависит от *Оценок на Экзамене* (т.к. коэффициент отрицательный). Однако данная модель нуждается в обработке, поскольку она станет по-настоящему “хорошей” (то есть, по которой можно сделать однозначные прогнозы) только, когда $R^2 > 70\%$.

➤ $Agriculture = 0.64 * Fertility + 5.63$:

Поскольку $R^2 = 0.12\%$ (что $<< 30\%$), то можно с уверенностью сказать, что данная модель плоха, и делать по ней какие-либо выводы нельзя.

№4. Оценить, есть ли взаимосвязь между объясняемой переменной и объясняющей переменной.

➤ $Agriculture = -1.95 * Examination + 82.88$:

Поскольку при обоих коэффициентах значение р-статистики очень хорошее (3 звезды, см. *Таблица 1*), то можно сделать вывод, что существует взаимосвязь между Объясняемой Переменной (*Agriculture*) и Регрессом (*Examination*). Объяснить данную зависимость чисто логически довольно просто: «Чем выше оценки на экзамене, тем на более престижную вакансию претендует человек, вследствие чего, количество людей заинтересованных в работе в сфере сельского хозяйства снижается».

➤ $Agriculture = 0.64 * Fertility + 5.63$:

Поскольку при обоих коэффициентах значение р-статистики плохое (0 звезд / 1 звезда, см. *Таблица 2*), то взаимосвязь между Объясняемой Переменной (*Agriculture*) и Регрессом (*Fertility*) практически отсутствует, поэтому делать какие-либо выводы по этой зависимости нельзя.

Вывод: Были построены две зависимости с одинаковой Объясняемой Переменной (*Agriculture*) и разными Объясняющими Переменными (*Examination / Fertility*). Модель “*Agriculture ~ Fertility*” получилась плохой, поскольку у нее очень низкий R^2 и практически отсутствует зависимость между объясняемой переменной (*Agriculture*) и регрессором (*Fertility*). С моделью “*Agriculture ~ Examination*” всё не так однозначно, так как у неё довольно высокое значение R^2 и хорошее значение р-статистики. Исходя из этого, делаем вывод, что существует отрицательная зависимость между объясняемой переменной (*Agriculture*) и регрессором (*Examination*), однако она сложнее, чем линейная зависимость.

Таким образом, можно сделать вывод, что обе исследуемые модели нуждаются в доработке, и ни по одной из них нельзя сделать однозначных прогнозов.

Код решения задач приведён в “Приложение 1”.

Задача №2.1

Необходимо загрузить данные из указанного набора и произвести следующие действия.

Набор данных: Swiss.

Объясняемая переменная: *Fertility*.

Регрессоры: *Agriculture*, *Examination*, *Infant.Mortality*.

№1. Проверить, что в наборе данных нет линейной зависимости (построить зависимости между переменными, и проверить, что R^2 в каждой из них не высокий). В случае если R^2 большой, один из таких столбцов можно исключить из рассмотрения.

Проверим линейную регрессию $Agriculture \sim Examination, Infant.Mortality$. $R^2 = 49.1\%$. $VIF = 1/(1-R^2) = 1.96$. Хотя VIF и меньше 5, но если учесть, что при регрессоре (*Examination*) хорошее значение р-статистики (3 звезды, см. [Таблица 3](#)), и что $R^2 = 49.1\%$ - это довольно неплохой показатель, то можно сказать, что существует небольшая зависимость между регрессором (*Examination*) и регрессором (*Agriculture*).

Таблица 3. Характеристики модели зависимости параметра *Agriculture* от параметров *Examination* и *Infant.Mortality* в наборе данных Swiss.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	105.56	18.29	5.77	7.32e-07	***
<i>Examination</i>	-2	0.3	-6.49	6.42e-08	***
<i>Infant.Mortality</i>	-1.1	0.84	-1.3	0.2	

Зависимость $Examination \sim Agriculture, Infant.Mortality$. $R^2 = 49.6\%$. $VIF = 1/(1-R^2) = 1.98$. Хотя VIF и меньше 5, но если учесть, что при регрессоре (*Agriculture*) хорошее значение р-статистики (3 звезды, см. [Таблица 4](#)), и что $R^2 = 49.1\%$ - это довольно неплохой показатель, то можно сказать, что существует небольшая зависимость между регрессором (*Agriculture*) и регрессором (*Examination*).

Таблица 4. Характеристики модели зависимости параметра *Examination* от параметров *Agriculture* и *Infant.Mortality* в наборе данных Swiss.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	37.42	6.33	5.91	4.53e-07	***
<i>Agriculture</i>	-0.24	0.04	-6.49	6.42e-08	***
<i>Infant.Mortality</i>	-0.43	0.29	-1.46	0.15	

В регрессии $Infant.Mortality \sim Agriculture, Examination$ $R^2 = 4.9\%$. $VIF = 1/(1-R^2) = 1.05$. Т.к. VIF значительно меньше 5, и значение р-статистики плохое (0 звезд / 0 звезд, см. [Таблица 5](#)), то можно с уверенностью сказать, что регрессор (*Infant.Mortality*) не зависит от регрессоров (*Agriculture*, *Examination*).

Таблица 5. Характеристики модели зависимости параметра *Infant.Mortality* от параметров *Agriculture* и *Examination* в наборе данных Swiss.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	23.43	2.36	9.92	8.6e-13	***
<i>Agriculture</i>	-0.03	0.03	-1.3	0.2	
<i>Examination</i>	-0.1	0.07	-1.46	0.15	

Хоть между регрессором(*Agriculture*) и регрессором(*Examination*) и существует небольшая зависимость, но она слишком незначительна, чтобы их исключить. Таким образом, заключаем, что для построения моделей, можно использовать все регрессоры из условия.

№2. Построить линейную модель зависимой переменной (*Fertility*) от регрессоров (*Agriculture/Examination/Infant.Mortality*) по методу наименьших квадратов. Оценить, насколько хороша модель, согласно: 1) R^2 , 2) р-значениям каждого коэффициента.

2.1) В регрессии $Fertility \sim Agriculture, Examination, Infant.Mortality$ $R^2 = 53.98\%$. Для трех регрессоров значение R^2 довольно низкое, следовательно, сделать какие-либо выводы по данной зависимости нельзя.

2.2) Рассмотрим р-статистику у регрессоров (*Agriculture, Examination, Infant.Mortality*).

- Значение р-статистики при регрессоре (*Examination*) низкое (***, см. Таблица 6);
- Значение р-статистики при регрессоре (*Infant.Mortality*) довольно низкое (**, см. Таблица 6);

➤ Значение р-статистики при регрессоре (*Agriculture*) высокое (0 звезд, см. Таблица 6);

Обратим внимание, что регрессор (*Agriculture*) не значим. Р-статистика достаточно велика (см. Таблица 6), так что можно провести эксперимент по его исключению:

- $R^2 = 53.64\%$, следовательно R^2 изменился всего на 0.35%, что $\ll 5\%$. Таким образом, мы можем исключить из рассмотрения регрессор (*Agriculture*).

Таблица 6. Характеристики модели зависимости параметра *Fertility* от параметров *Agriculture, Examination* и *Infant.Mortality* в наборе данных Swiss.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	60.87	12.83	4.75	2.32e-05	***
<i>Examination</i>	-1.04	0.23	-4.56	4.22e-05	***
<i>Infant.Mortality</i>	1.44	0.46	3.16	0.003	**
<i>Agriculture</i>	-0.05	0.08	-0.57	0.57	

№3. Ввести в модель логарифмы регрессоров. Сравнить модели и выбрать наилучшую.

При решении этой задачи были проверены модели:

1. $Fertility \sim \log(Examination), \log(Infant.Mortality)$ – Таблица 7
2. $\log(Fertility) \sim \log(Examination), \log(Infant.Mortality)$ – Таблица 8
3. $Fertility \sim \log(Examination), Infant.Mortality$ – Таблица 9
4. $Fertility \sim Examination, \log(Infant.Mortality)$ – Таблица 10

Таблица 7. Характеристики модели зависимости параметра $Fertility$ от параметров $\log(Examination)$, и $\log(Infant.Mortality)$ в наборе данных Swiss.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	6.35	25.49	0.25	0.8	
$\log(Examination)$	-12.76	2.22	-5.75	7.87e-07	***
$\log(Infant.Mortality)$	32.79	8.33	3.94	0.0003	***

Таблица 8. Характеристики модели зависимости параметра $\log(Fertility)$ от параметров $\log(Examination)$, и $\log(Infant.Mortality)$ в наборе данных Swiss.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	3.26	0.41	7.85	6.85e-10	***
$\log(Examination)$	-0.19	0.04	-5.34	3.13e-06	***
$\log(Infant.Mortality)$	0.5	0.14	3.68	0.0006	***

Таблица 9. Характеристики модели зависимости параметра $Fertility$ от параметров $\log(Examination)$, и $Infant.Mortality$ в наборе данных Swiss.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	67.48	10.45	6.46	7.11e-08	***
$\log(Examination)$	-12.89	2.18	-5.92	4.44e-07	***
$Infant.Mortality$	1.85	0.44	4.22	0.000119	***

Таблица 10. Характеристики модели зависимости параметра *Fertility* от параметров *Examination*, и $\log(\text{Infant.Mortality})$ в наборе данных Swiss.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	8.53	25.61	0.33	0.74	
<i>Examination</i>	-0.95	0.16	-5.73	8.33e-07	***
$\log(\text{Infant.Mortality})$	2.89	8.41	3.08	0.004	**

Значения R^2 для проверенных моделей:

1. 52.16% - показатели ухудшились;
2. 48.58% - показатели ухудшились;
3. 54% - показатели немного улучшились;
4. 52% - показатели ухудшились.

Наилучшей оказалась модель: $Fertility \sim \log(\text{Examination}), \text{Infant.Mortality}$.

$$\# Fertility = -12.89 * \log(\text{Examination}) + 1.85 * \text{Infant.Mortality} + 67.48$$

Вывод: Хотя введение в модель логарифмы регрессоров и дало прирост, но слишком незначительный.

№4. Ввести в модель всевозможные произведения пар регрессоров, в том числе квадраты регрессоров. Найти одну или несколько наилучших моделей по доле объяснённого разброса в данных R^2 .

4.1) При решении этой задачи была проверена модель ($Fertility \sim Examination, \text{Infant.Mortality}$), в которую были добавлены параметры: $I(\text{Examination}^2)$ / $I(\text{Infant.Mortality}^2)$ / $I(\text{Examination} * \text{Infant.Mortality})$. $\#R^2 = 54.69\%$

Таблица 11. Проверка на линейную зависимость между регрессоров (*Examination*, *Infant.Mortality*, $I(\text{Examination}^2)$, $I(\text{Infant.Mortality}^2)$, $I(\text{Examination} * \text{Infant.Mortality})$) с помощью команды VIF.

Параметр \ Характеристики	<i>Examination</i>	<i>Infant.Mortality</i>	$I(\text{Examination}^2)$	$I(\text{Infant.Mortality}^2)$	$I(\text{Examination} * \text{Infant.Mortality})$
VIF	57.84	159.2	15.68	114.72	64.08

Поскольку у многих объясняющих переменных значения VIF довольно большое, то можно сделать вывод, что в модели присутствует линейная зависимость между регрессорами.

4.2) Будем избавляться от регрессоров с максимальным VIF, пока все значения VIF не будут меньше 10.

1. $Fertility \sim Examination, I(Examination^2), I(Infant.Mortality^2), I(Examination*Infant.Mortality)$. – Таблица 12;

2. $Fertility \sim I(Examination^2), I(Infant.Mortality^2), I(Examination*Infant.Mortality)$. - Таблица 13;

3. $Fertility \sim I(Examination^2), I(Infant.Mortality^2)$ - Таблица 14;

Таблица 12. Проверка на линейную зависимость между регрессоров ($Examination$, $I(Examination^2)$, $I(Infant.Mortality^2)$, $I(Examination*Infant.Mortality)$) с помощью команды VIF.

Параметр \ Характеристики	$Examination$	$I(Examination^2)$	$I(Infant.Mortality^2)$	$I(Examination*Infant.Mortality)$
VIF	35.28	14	3.77	27.15

Таблица 13. Проверка на линейную зависимость между регрессоров ($I(Examination^2)$, $I(Infant.Mortality^2)$, $I(Examination*Infant.Mortality)$) с помощью команды VIF.

Параметр \ Характеристики	$I(Examination^2)$	$I(Infant.Mortality^2)$	$(Examination*Infant.Mortality)$
VIF	10.29	2.42	10.49

Таблица 14. Проверка на линейную зависимость между регрессоров ($I(Examination^2)$, $I(Infant.Mortality^2)$) с помощью команды VIF.

Параметр \ Характеристики	$I(Examination^2)$	$I(Infant.Mortality^2)$
VIF	1.027	1.027

Таблица 15. Характеристики модели зависимости параметра $Fertility$ от параметров $I(Examination^2)$, и $I(Infant.Mortality^2)$ в наборе данных Swiss.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	63.49	5.39	11.78	3.35e-15	***
$I(Examination^2)$	-0.02	0.004	5.38	2.76e-06	***
$I(Infant.Mortality^2)$	0.04	0.01	3.054	0.004	**

$$\# Fertility = -0.02 * I(Examination^2) + 0.04 * I(Infant.Mortality^2) + 63.49$$

#Значение R^2 для данной модели = 50.43%

Вывод: Наилучшая модель - это « $Fertility \sim I(Examination^2), I(Infant.Mortality^2)$ », поскольку все значения VIF регрессоров меньше 5.

Задача №2.2

№1. Оценить доверительные интервалы для всех коэффициентов в модели «*Fertility ~ Agriculture, Examination, Infant.Mortality*».

Всего проводилось 47 наблюдений, в данной модели оценивалось 4 коэффициента. Следовательно, количество свободных коэффициентов = $47 - 4 = 43$.

- *Agriculture*:
 $СКО(se) = 0.08$ (см. Таблица 16)
Критерий Стьюдента(t) = 2.02
Доверительный интервал: [-0.21, 0.12]
- *Examination*:
 $СКО(se) = 0.23$ (см. Таблица 16)
Критерий Стьюдента(t) = 2.02
Доверительный интервал: [-1.5, -0.58]
- *Infant.Mortality*:
 $СКО(se) = 0.46$ (см. Таблица 16)
Критерий Стьюдента(t) = 2.02
Доверительный интервал: [0.52, 2.36]
- *Intercept*:
 $СКО(se) = 12.83$ (см. Таблица 16)
Критерий Стьюдента(t) = 2.02
Доверительный интервал: [35, 86.74]

№2. Построим таблицу с доверительными интервалами для всех коэффициентов в модели и сделаем вывод о том, может ли коэффициент быть равен 0.

Таблица 16. Характеристики модели зависимости параметра *Fertility* от параметров *Agriculture*, *Examination* и *Infant.Mortality* в наборе данных Swiss.

Параметр \ Характеристики	Значение	СКО	Доверительный интервал	“Может ли коэффициент быть равен 0?”
(Intercept)	60.87	12.83	[35, 86.74]	Нет
<i>Examination</i>	-1.04	0.23	[-1.5, -0.58]	Нет
<i>Infant.Mortality</i>	1.44	0.46	[0.52, 2.36]	Нет
<i>Agriculture</i>	-0.05	0.08	[-0.21, 0.12]	Да

Вывод: Поскольку 0 попадает в доверительный интервал регрессора (*Agriculture*), то значение коэффициента перед этим регрессором может быть равно 0. Следовательно, объясняющая переменная (*Agriculture*) практически не связана с объясняемой переменной (*Fertility*).

№3. Оценить доверительный интервал для одного прогноза для модели «*Fertility ~ Agriculture, Examination, Infant.Mortality*».

Зададим следующий набор значений для регрессоров: (*Agriculture* = 10, *Examination* = 30, *Infant.Mortality* = 20).

Применим функцию *predict()* для оцениваемой модели, что вычислить прогноз модели и доверительный интервал:

Таблица 17. Результат выполнения функции *predict()*

Прогноз модели	Нижняя граница интервала	Верхняя граница интервала
58.01866	52.43242	63.60491

Вывод: Прогноз модели «*Fertility ~ Agriculture, Examination, Infant.Mortality*» оценивается как 58.02. Доверительный интервал для свободного коэффициента имеет вид: [52.43, 63.6].

Вывод:

Задача №2.1. Рассматриваемая модель была проверена на наличие линейной зависимости между регрессорами. Хотя и была обнаружена небольшая зависимость между двумя объясняющими переменными, но в виду её незначительности, было принято решение не исключать ни одну из переменных в модели.

Однако при исследовании самой модели было выявлено, что объясняемая переменная (*Fertility*) почти не зависит от одного из регрессоров (*Agriculture*), поэтому было всё же решено исключить из рассмотрения одну из объясняющих переменных.

В пункте №3 была попытка улучшить рассматриваемую модель, путём введения логарифмов регрессоров. Однако, это не дало видимых результатов.

В пункте №4 в модель были введены всевозможные произведения пар регрессоров, и была выявлена одна наилучшая модель по доле объяснённого разброса в данных R^2 .

Задача №2.2. Были найдены доверительные интервалы для всех коэффициентов в рассматриваемой модели (при $p=95\%$), и было выявлено, что т.к. значение коэффициента перед регрессором (*Agriculture*) может быть равно 0, то объясняющая переменная (*Agriculture*) практически не связана с объясняемой переменной (*Fertility*).

В пункте №3 для оценивания доверительного интервала для одного прогноза, были выбраны следующие значения: «*Agriculture* = 10, *Examination* = 30, *Infant.Mortality* = 20». Затем, с помощью функции *predict()* был вычислен прогноз и доверительный интервал для рассматриваемой модели.

Код решения задач приведён в “Приложение 2”.

Задача №3

Необходимо загрузить данные из указанного набора и произвести следующие действия.

Набор данных: Данные обследования РМЭЗ НИУ ВШЭ

Объясняемая переменная: *salary*

Регрессоры: *age, sex, higher_educ, status2, dur, wed1, wed2, wed3*.

№1. Построить линейную регрессию зарплаты на все параметры, и оценить коэффициент вздутия дисперсии VIF.

Построим линейную регрессию зарплаты(*salary*) на все параметры, предварительно исключив из рассмотрения все строки с отсутствующими значениями(NA). После этого также исключим из рассмотрения все регрессоры с отсутствующими значениями, и все незначительные регрессоры (см. Таблица 18).

Таблица 18. Характеристики модели зависимости параметра *salary* от параметров *age, sex, higher_educ, status2, dur, wed1, wed2, wed3* в наборе данных обследования РМЭЗ НИУ ВШЭ.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости	Можно ли исключить регрессор?
(Intercept)	-0.74	0.05	-15.85	<2e-16	***	Да
<i>age</i>	-0.08	0.02	-5.09	3.79e-07	***	Нет
<i>sex</i>	0.5	0.03	17.17	<2e-16	***	Нет
<i>higher_educ</i>	0.56	0.03	18.6	<2e-16	***	Да
<i>status2</i>	0.36	0.03	11.53	<2e-16	***	Да
<i>dur</i>	0.11	0.01	7.94	2.60e-15	***	Нет
<i>wed1</i>	0.11	0.04	2.56	0.01	*	Да
<i>wed2</i>	0.13	0.05	2.38	0.02	*	Да
<i>wed3</i>	-0.1	0.05	-1.82	0.07	.	Да

Построим линейную регрессию зарплаты(*salary*) на все оставшиеся параметры (*age, sex, dur*). $R^2 = 0.06639$. Значение р-статистики хорошее у всех переменных(3 звезды, см. Таблица 19). $VIF = 1 / (1-R^2) = 1.004$. $VIF \ll 5$, следовательно линейной зависимости между переменными нет.

Таблица 19. Характеристики модели зависимости параметра *salary* от параметров *age, sex, dur* в наборе данных обследования РМЭЗ НИУ ВШЭ.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	-0.19	0.02	-9.5	<2e-16	***
<i>age</i>	-0.08	0.01	-5.38	7.93e-08	***
<i>sex</i>	0.42	0.03	14.08	<2e-16	***
<i>dur</i>	0.08	0.02	5.52	3.59e-08	***

№2. Введём в модель логарифмы и степени переменных:

№2.1) Введём в модель логарифмы регрессоров, сравним модели и выберем наилучшую.

Рассмотрим модель ($salary \sim age, sex, dur, \log(age), \log(dur)$). $R^2 = 0.09061$, следовательно, показатель увеличился на 0.024, по сравнению с моделью: " $salary \sim age, sex, dur$ ". Значение р-статистики хорошее только у переменных $\log(age)$ и sex (3 звезды, см. Таблица 20).

Таблица 20. Характеристики модели зависимости параметра $salary$ от параметров $age, sex, dur, \log(age), \log(dur)$ в наборе данных обследования РМЭЗ НИУ ВШЭ.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	0.19	0.19	1.02	0.3	
age	-0.52	0.13	-3.93	9.31e-05	***
$\log(age)$	0.09	0.07	1.29	0.2	
sex	0.4	0.08	5.15	3.51e-07	***
dur	0.07	0.07	1.06	0.29	
$\log(dur)$	0.01	0.09	0.13	0.9	

Попробуем улучшить модель, убирая из неё параметры с наибольшими коэффициентами в VIF.

Таблица 21. Проверка на линейную зависимость между регрессорами ($age, sex, dur, \log(age), \log(dur)$) с помощью команды VIF.

Параметр \ Характеристики	age	sex	dur	$\log(age)$	$\log(dur)$
VIF	3.79	1.02	4.14	3.79	4.16

Рассмотрим модель ($salary \sim age, sex, dur, \log(age)$). $R^2 = 0.06722$, следовательно, показатель увеличился на 0.008, по сравнению с моделью: " $salary \sim age, sex, dur$ ". Значение р-статистики очень хорошая (3 звезды, см. Таблица 22) у всех переменных, кроме $\log(age)$ и "Свободного Коэффициента".

Таблица 22. Характеристики модели зависимости параметра $salary$ от параметров $age, sex, dur, \log(age)$ в наборе данных обследования РМЭЗ НИУ ВШЭ.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	0.05	0.08	0.61	0.54	
age	-0.29	0.07	-4.43	1.01e-05	***
$\log(age)$	0.032	0.04	0.89	0.37	
sex	0.32	0.04	7.73	1.68e-14	***
dur	0.09	0.02	4.76	2.05e-06	***

Таблица 23. Проверка на линейную зависимость между регрессорами (*age*, *sex*, *dur*, $\log(\text{age})$) с помощью команды VIF.

Параметр \ Характеристики	<i>age</i>	<i>sex</i>	<i>dur</i>	$\log(\text{age})$
VIF	3.64	1.04	1.05	3.61

Рассмотрим модель ($\text{salary} \sim \text{sex}, \text{dur}, \log(\text{age})$). $R^2 = 0.05907$, следовательно, показатель уменьшился на 0.007, по сравнению с моделью: " $\text{salary} \sim \text{age}, \text{sex}, \text{dur}$ ". Значение р-статистики очень хорошее (3 звезды, см. Таблица 24) у всех переменных.

Таблица 24. Характеристики модели зависимости параметра *salary* от параметров *sex*, *dur*, $\log(\text{age})$ в наборе данных обследования РМЭЗ НИУ ВШЭ.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	-0.27	0.03	-9.53	<2e-16	***
$\log(\text{age})$	-0.1	0.02	-5.4	7.18e-08	***
<i>sex</i>	0.32	0.04	9.64	3.33-14	***
<i>dur</i>	0.1	0.02	5.07	4.29e-07	***

№2.2) Введём в модель степени регрессоров, сравним модели и выберем наилучшую.

Рассмотрим модель ($\text{salary} \sim \text{age}, \text{sex}, \text{dur}, I(\text{age}^{0.1}), I(\text{dur}^{0.1})$). $R^2 = 0.09058$ - следовательно, показатель увеличился на 0.024, по сравнению с моделью: " $\text{salary} \sim \text{age}, \text{sex}, \text{dur}$ ". Значение р-статистики очень хорошее (3 звезды, см. Таблица 25) только у переменных *age* и *sex*.

Таблица 25. Характеристики модели зависимости параметра *salary* от параметров *age*, *sex*, *dur*, $I(\text{age}^{0.1})$, $I(\text{dur}^{0.1})$ в наборе данных обследования РМЭЗ НИУ ВШЭ.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	-1.14	1.1	-1.03	0.3	
<i>age</i>	0.054	0.15	-362	0.0003	***
$I(\text{age}^{0.1})$	1.1	0.88	1.26	0.2	
<i>sex</i>	0.4	0.08	5.1	3.68e-07	***
<i>dur</i>	0.06	0.07	0.8	0.5	
$I(\text{dur}^{0.1})$	0.26	0.96	0.27	0.79	

Таблица 26. Проверка на линейную зависимость между регрессорами (age , sex , dur , $I(age^{0.1})$, $I(dur^{0.1})$) с помощью команды VIF.

Параметр \ Характеристики	age	sex	dur	$I(age^{0.1})$	$I(dur^{0.1})$
VIF	4.835	1.02	5.01	4.829	5.03

Попробуем улучшить модель, убирая из неё параметры с наибольшими коэффициентами в VIF.

Рассмотрим модель ($salary \sim age, sex, dur, I(age^{0.1})$). $R^2 = 0.09058$ - следовательно, показатель увеличился на 0.024, по сравнению с моделью: " $salary \sim age, sex, dur$ ". Значение р-статистики очень хорошее (3 звезды) только у переменных age и sex .

Таблица 27. Проверка на линейную зависимость между регрессорами (age , sex , dur , $I(age^{0.1})$) с помощью команды VIF.

Параметр \ Характеристики	age	sex	dur	$I(age^{0.1})$
VIF	4.6	4.58	1.24	1.05

Рассмотрим модель ($salary \sim sex, dur, I(age^{0.1})$). $R^2 = 0.06045$ - следовательно, показатель увеличился на 0.0059, по сравнению с моделью: " $salary \sim age, sex, dur$ ". Значение р-статистики очень хорошее (3 звезды, см. Таблица 28) у всех переменных.

Таблица 28. Характеристики модели зависимости параметра $salary$ от параметров sex , dur , $I(age^{0.1})$ в наборе данных обследования РМЭЗ НИУ ВШЭ.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	0.9	0.2	4.57	5.29e-06	***
$I(age^{0.1})$	-1.2	0.21	-5.69	1.44e-08	***
sex	0.32	0.04	7.66	2.91e-14	***
dur	0.1	0.02	5.02	5.67e-07	***

Попробуем улучшить модель, повышая степень (до степени "2", с шагом 0.1):

При решении этой задачи были проверены модели:

1. $salary \sim sex, dur, I(age^{0.2})$
2. $salary \sim sex, dur, I(age^{0.3})$
3. $salary \sim sex, dur, I(age^{0.4})$
4. $salary \sim sex, dur, I(age^{0.5})$

Поскольку при повышении степени от "0.1" до "0.5"(см. Таблица 29) R^2 возрастает, то попробуем взять сразу степень равную "0.9":

5. $salary \sim sex, dur, I(age^{0.9})$
6. $salary \sim sex, dur, I(age^{1.1})$
7. $salary \sim sex, dur, I(age^{1.2})$
8. $salary \sim sex, dur, I(age^{1.3})$
9. $salary \sim sex, dur, I(age^{1.4})$

Поскольку при степени "1.4" значение R^2 меньше, чем при степени "1.3"(см. Таблица 29), то попробуем сразу взять степень равную "1.9":

10. $salary \sim sex, dur, I(age^{1.9})$
11. $salary \sim sex, dur, I(age^2)$

Построим сравнительную таблицу для всех проверенных моделей:

Таблица 29. Сравнение уровня значимости и значения R^2 для одиннадцати рассматриваемых моделей.

Рассматриваемая модель	Уровень значимости	Значение R^2	Насколько изменилось значение R^2 по сравнению с моделью: "salary ~ age, sex, dur"
$salary \sim sex, dur, I(age^{0.2})$	Значение р-статистики очень хорошее (***) у всех переменных	0.06174	уменьшился на 0.005
$salary \sim sex, dur, I(age^{0.3})$	Значение р-статистики очень хорошее (***) у всех переменных	0.06291	уменьшился на 0.003
$salary \sim sex, dur, I(age^{0.4})$	Значение р-статистики очень хорошее (***) у всех переменных, кроме "Свободного Коэффициента"	0.06394	уменьшился на 0.002
$salary \sim sex, dur, I(age^{0.5})$	Значение р-статистики очень хорошее (***) у всех переменных, кроме "Свободного Коэффициента"	0.06394	уменьшился на 0.001
$salary \sim sex, dur, I(age^{0.9})$	Значение р-статистики очень хорошее (***) у всех переменных, кроме "Свободного Коэффициента"	0.06704	увеличился на 0.0007
$salary \sim sex, dur, I(age^{1.1})$	Значение р-статистики очень хорошее (***) у всех переменных, кроме "Свободного Коэффициента"	0.06749	увеличился на 0.0011
$salary \sim sex, dur, I(age^{1.2})$	Значение р-статистики очень хорошее (***) у всех переменных, кроме "Свободного Коэффициента"	0.06758	увеличился на 0.00119
$salary \sim sex, dur, I(age^{1.3})$	Значение р-статистики очень хорошее (***) у всех переменных, кроме "Свободного Коэффициента"	0.06759	увеличился на 0.0012
$salary \sim sex, dur, I(age^{1.4})$	Значение р-статистики очень хорошее (***) у всех переменных, кроме "Свободного Коэффициента"	0.06754	увеличился на 0.00115
$salary \sim sex, dur, I(age^{1.9})$	Значение р-статистики очень хорошее (***) у всех переменных	0.06643	увеличился на 0.00004
$salary \sim sex, dur, I(age^2)$	Значение р-статистики очень хорошее (***) у всех переменных	0.07877	увеличился на 0.012

№3. Выберем наилучшую модель среди построенных.

3.1) Вывод: Среди моделей с логарифмами лучшей можно считать: " salary ~ sex , dur, log(age)", однако эта модель хуже первоначальной: "salary ~ age, sex, dur" (т.к. р-статистика хорошая в обоих случаях, а R^2 в модели с логарифмом меньше, чем в первоначальной модели).

Следовательно, $\log()$ не дал ощутимых результатов.

3.2) Вывод: Среди моделей со степенями лучшая - это модель с квадратом, поскольку в остальных случаях у нас ниже R^2 и больше параметров с плохим значением р-статистики.

№4. Сделай вывод о том, какие индивиды получают наибольшую зарплату.

Вывод: Наибольшую зарплату получают мужчины более молодого возраста, живущие в городе, с высшим образованием, с большой продолжительностью рабочей недели и никогда не состоящие в браке.

№5. Оценить регрессии для подмножества индивидов:

5.1) Городские жители, состоящие в браке;

5.2) Разведенные, без высшего образования.

5.1) Найдем подмножество городских жителей, состоящих в браке:

```
data_3 = subset(data_2, status2 == 1)
data_3
```

```
data_4 = subset(data_3, wed1 == 1)
data_4
```

Рассмотрим модель ($salary \sim sex, dur, I(age^2)$). $R^2 = 0.1173$, Значение р-статистики хорошее у всех переменных, кроме "Свободного Коэффициента» (см. [Таблица 30](#)). Следовательно, можно сделать вывод, что модель довольно хорошая.

[Таблица 30](#). Характеристики модели зависимости параметра *salary* от параметров *sex*, *dur*, $I(age^2)$ для подмножества индивидов: "Городские жители, состоящие в браке" в наборе данных обследования РМЭЗ НИУ ВШЭ.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	0.05	0.04	1.17	0.24	
$I(age^2)$	0.1	0.03	3.53	0.0004	***
<i>sex</i>	0.65	0.05	12.43	<2e-16	***
<i>dur</i>	-0.18	0.03	-7.03	1.97e-12	***

Вывод: Наибольшую зарплату получают мужчины, с высшим образованием.

5.2) Найдем подмножество разведенных, без высшего образования:

```
data_5 = subset(data_2, wed2=1)
data_5
```

```
data_6 = subset(data_5, higher_educ==0)
data_6
```

Рассмотрим модель ($salary \sim sex, dur, I(age^2)$). $R^2 = 0.09543$, Значение р-статистики хорошее у всех переменных (см. [Таблица 31](#)). Следовательно, можно сделать вывод, что модель довольно хорошая.

Таблица 31. Характеристики модели зависимости параметра *salary* от параметров *sex*, *dur*, $I(age^2)$ для подмножества индивидов: “Разведенные, без высшего образования” в наборе данных обследования РМЭЗ НИУ ВШЭ.

Параметр \ Характеристики	Значение	Std. Error	t value	Pr(> t)	Уровень значимости
(Intercept)	-0.26	0.04	-5.9	7.22e-09	***
$I(age^2)$	0.03	0.03	2.78	0.0006	**
<i>sex</i>	0.08	0.08	3.2	0.002	**
<i>dur</i>	-0.1	0.02	-4.44	1.13e-15	***

Вывод: Наибольшую зарплату получают мужчины более молодого возраста, с большой продолжительностью рабочей недели.

Вывод:

В пункте №1 была построена линейная регрессия зарплаты на все параметры. Оценив коэффициент вздутия дисперсии VIF, мы выявили, что линейной зависимости между переменными нет.

В пункте №2.1 была попытка улучшить рассматриваемую модель, путём введения логарифмов регрессоров. Однако это не дало видимых результатов.

В пункте №2.2 в модель были введены всевозможные произведения пар регрессоров, и была выявлена одна наилучшая модель по доле объяснённого разброса в данных R^2 .

Исходя из предоставленных данных, было выявлено, что наибольшую зарплату получают мужчины более молодого возраста, живущие в городе, с высшим образованием, с большой продолжительностью рабочей недели и никогда не состоящие в браке.

В пункте №5.1 была построена зависимость параметра *salary* от параметров *sex*, *dur*, $I(age^2)$ для подмножества индивидов: “Городские жители, состоящие в браке”, и было выявлено, что наибольшую зарплату получают мужчины, с высшим образованием.

В пункте №5.2 была построена зависимость параметра *salary* от параметров *sex*, *dur*, $I(age^2)$ для подмножества индивидов: “Разведенные, без высшего образования”, и было выявлено, что наибольшую зарплату получают мужчины более молодого возраста, с большой продолжительностью рабочей недели.

Код решения задач приведён в “Приложение 3”.

Задача №4

Набор данных: https://www.kaggle.com/arashnic/hr-analytics-job-change-of-data-scientists?select=aug_test.csv

Тип классификатора – DecisionTreeClassifier (решающее дерево)
Классификация по столбцу - Education level (Graduate – класс 0, остальные уровни – класс 1)

№1. Обработайте набор данных, указанный во втором столбце таблицы 4.1, подготовив его к решению задачи классификации. Выделите целевой признак, указанный в последнем столбце таблицы, и удалите его из данных, на основе которых будет обучаться классификатор. Разделите набор данных на тестовую и обучающую выборку. Постройте классификатор типа, указанного в третьем столбце, для задачи классификации по параметру, указанному в последнем столбце. Оцените точность построенного классификатора с помощью метрик precision, recall и F1 на тестовой выборке.

№2. Постройте классификатор типа Случайный Лес (Random Forest) для решения той же задачи классификации. Оцените его качество с помощью метрик precision, recall и F1 на тестовой выборке. Какой из классификаторов оказывается лучше?

Опишем столбцы нашего Датасета:

0. city_development_index : индекс развития города (в масштабе)
1. gender: Пол кандидата
2. relevent_experience: Соответствующий опыт кандидата
3. enrolled_university: Тип зачисленных университетских курсов, если таковые имеются
4. education_level: Уровень образования кандидата
5. major_discipline: Обучение основной дисциплине кандидата
6. experience: Кандидатский общий стаж в годах
7. company_size: Количество сотрудников в компании текущего работодателя
8. company_type: Тип текущего работодателя
9. last_new_job: разница в годах между предыдущей работой и текущей работой
10. training_hours: завершённые часы обучения

Применим функцию `info()`, чтобы посмотреть информацию о Датасете:

```
data.info()

<class 'pandas.core.frame.DataFrame'>
Index: 2129 entries, city_41 to city_102
Data columns (total 11 columns):
#   Column                      Non-Null Count  Dtype
---  -
0   city_development_index      2129 non-null   float64
1   gender                      1621 non-null   object
2   relevent_experience          2129 non-null   object
3   enrolled_university         2098 non-null   object
4   education_level             2077 non-null   object
5   major_discipline            1817 non-null   object
6   experience                   2124 non-null   object
7   company_size                1507 non-null   object
8   company_type                1495 non-null   object
9   last_new_job                2089 non-null   object
10  training_hours              2129 non-null   int64
dtypes: float64(1), int64(1), object(9)
memory usage: 199.6+ KB
```

Рисунок 1. Результат работы команды `info()` в наборе данных `aug_test.csv`.

Некоторые признаки имеют пропуски и являются типами “object”, а не численными (см. Рисунок 1). Обработаем данные для решения задачи классификации:

- 1) Кодлируем столбцы из категориального признака в численный (см. Рисунок 2).
- 2) Будем заполнять пропуски в данных значением по умолчанию – индексом 0 (см. Рисунок 2).

```
array = ['enrolled_university', 'last_new_job', 'major_discipline', 'company_type', 'company_size', 'experience']
for column in array:
    Set = set(data[column])
    i = 0
    for item in Set:
        data[column] = data[column].replace(item, i)
        i = i + 1

data.loc[data.experience == 'NaN', 'experience'] = 0
data.loc[data.last_new_job == 'never', 'last_new_job'] = 0
data.loc[data.last_new_job == 'NaN', 'major_discipline'] = 0
data.loc[data.last_new_job == 'NaN', 'company_size'] = 0
```

Рисунок 2. Преобразуем текстовые данные в числовые.

Выделим целевой признак(`education_level`) и удалим его из данных, на основе которых будет обучаться классификатор:

```
Data2= data.education_level
train = data.drop('education_level', axis = 1)
```

Разделим данные на обучающую и тестовую выборку, а затем построим классификатор типа - "Решающее дерево" для задачи классификации по параметру(`education_level`) (см. Рисунок 3).

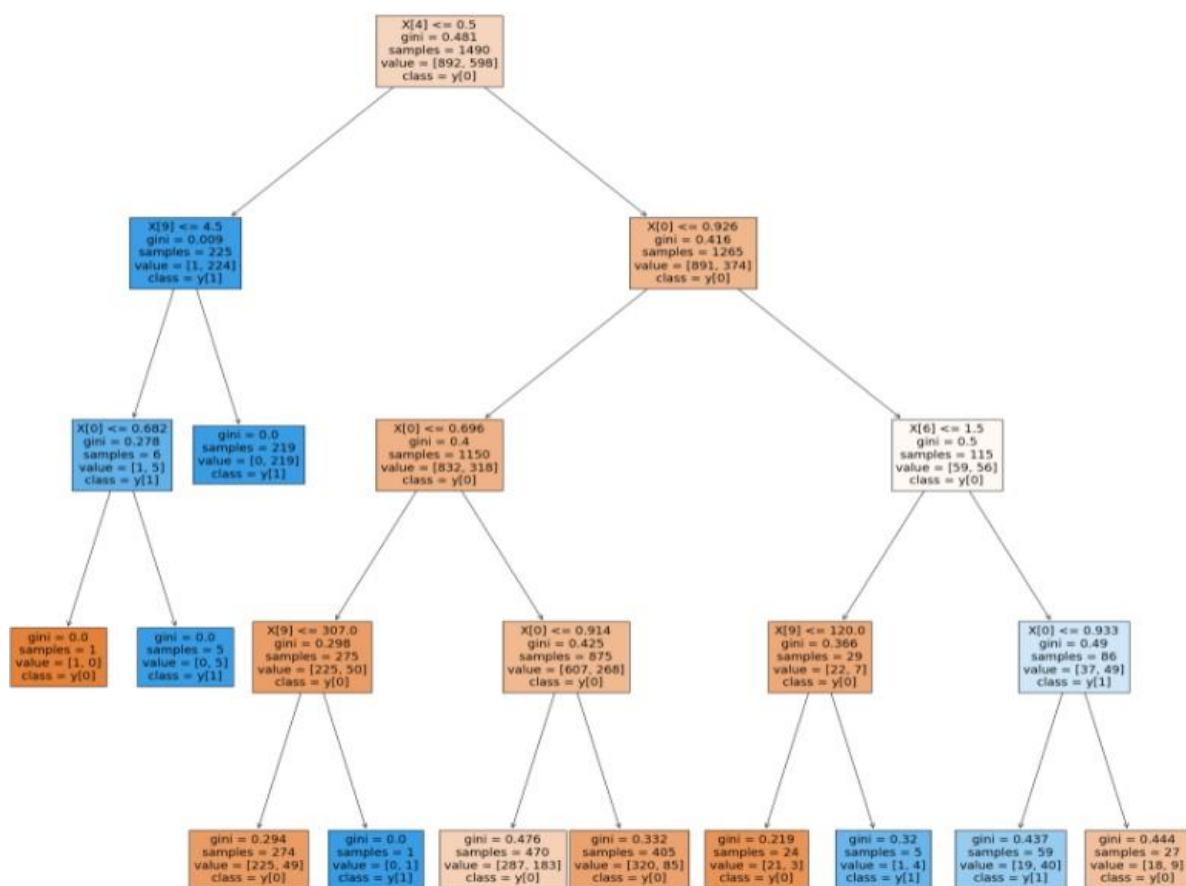


Рисунок 3. Классификатор типа - "Решающее дерево" для задачи классификации по параметру(education_level).

Оценим точность построенного классификатора с помощью метрик precision, recall и F1, а после попробуем улучшить данную модель, путём подбора гиперпараметров, с помощью GridSearchCV (см. Таблица 32).

Таблица 32. Сравнительная таблица результата работы метрик precision, recall и F1 в наборе данных aug_test.csv.

Метрики	Тип выборки	Обучающаяся выборка		Тестовая выборка		“Сравним результаты работы метрик на тестовой выборке”
		Используя оценщик GridSearchCV	Не используя, оценщик GridSearchCV	Используя оценщик GridSearchCV	Не используя, оценщик GridSearchCV	
F1		0.6366	0.5689	0.533	0.5023	Лучше после подбора гиперпараметров
precision		0.7512	0.8944	0.827	0.8059	
recall		0.5602	0.4181	0.397	0.3702	

Построим, классификатор типа Случайный Лес (Random Forest) для решения той же задачи классификации и оценим его качество с помощью метрик precision, recall и F1 на тестовой выборке (см. [Таблица 33](#)).

[Таблица 33](#). Работа метрик precision, recall и F1 в наборе данных aug_test.csv.

Метрики	Обучающаяся выборка	Тестовая выборка
F1	0.6366	0.533
precision	0.7512	0.827
recall	0.5602	0.397

Вывод:

Сравнивая результаты метрик классификаторов: Решающего дерева и Рандомного леса, видим, что классификатор Рандомного леса лучше, чем классификатор Решающего дерева, без использования оценщика GridSearchCV (см. [Таблица 34](#)).

[Таблица 34](#). Сравнение метрик классификаторов Решающего дерева и Рандомного леса.

Метрики	Решающего дерева	Рандомного леса
F1	50.23%	53.3%
precision	80.59%	82.7%
recall	37.02%	39.7%

На основе всех полученных данных можно сделать вывод, что классификатор DecisionTreeClassifier (решающее дерево) не подходит для данного Датасета, поскольку хоть он и имеет довольно хорошую полноту в 80.59%, но при этом имеет плохую точность в 50.23%, и f-меру в 37.02%.

Код решения задач приведён в “Приложение 4”.

Задача №5

Балашов Д. С. - Тема: "Употребление алкоголя учащимися"

Данные были получены в ходе опроса учащихся, изучающих математику и португальский язык в средней школе.

Наборы данных student-mat.csv (курс математики) и student-por.csv (курс португальского языка):

Набор данных №1:

<https://www.kaggle.com/uciml/student-alcohol-consumption?select=student-mat.csv>

Набор данных №2:

<https://www.kaggle.com/uciml/student-alcohol-consumption?select=student-por.csv>

Необходимо провести анализ Датасета и сделать обработку данных по предложенному алгоритму.

1. Сколько в Датасете объектов и признаков? Дать описание каждому признаку, если оно есть.

Объектов - 1044, Признаков – 33

Признаки:

1. school(школа) - бинарный признак: "GP" - Gabriel Pereira / "MS" - Mousinho da Silveira
2. sex(пол) - бинарный признак: "F" - female / "M" – male
3. age(возраст) - числовой признак: $15 \leq \text{age} \leq 22$
4. address(тип населённого пункта) - бинарный признак: "U" - urban / "R" – rural
5. famsize(количество человек в семье) - бинарный признак: "LE3" - ≤ 3 / "GT3" - > 3
6. Pstatus(состоят ли родители в браке) - бинарный признак: "T" - living together / "A" – Apart
7. Medu(образование матери) - категориальный признак: "0" - нет образования, "1" - начальное образование(закончила 4 класса), "2" - неполное среднее(закончила 9 классов), "3" - среднее образование(закончила 11 классов), "4" - высшее образование
8. Fedu(образование отца) - категориальный признак: "0" - нет образования, "1" - начальное образование(закончил 4 класса), "2" - неполное среднее(закончил 9 классов), "3" - среднее образование(закончил 11 классов), "4" - высшее образование
9. Mjob (работа матери) - категориальный признак: "teacher", "health" - care related, "services" - administrative or police, "at_home", "other"

10. Fjob(работа отца) - категориальный признак: "teacher", "health" - care related, "services" - administrative or police, "at_home", "other"
11. reason(причина выбора школы) - категориальный признак: "close to home", "school reputation", "course preference", "other"
12. guardian(опекун) - категориальный признак: "mother", "father", "other"
13. traveltime(сколько добираться до школы) - категориальный признак: "1" - < 15 минут, "2" - 15 минут <= traveltime <= 30 минут, "3" - 30 минут <= traveltime <= 1 час, "4" - > 1 час
14. studytime(сколько часов в неделю уделяется учёбе) - категориальный признак: "1" - < 2 часов, "2" - 2 часа <= studytime <= 5 часов, "3" - 5 часов <= traveltime <= 10 часов, "4" - > 10 часов
15. failures(сколько раз ученик оставался на второй год) - категориальный признак: "1" - 1 раз, "2" - 2 раза, "3" - 3 раза, "4" - > 3 раз
16. schoolsup(ученик имеет дополнительную образовательную поддержку) - бинарный признак: "yes" / "no"
17. famsup(ученик получает помощь в образовании со стороны семьи) - бинарный признак: "yes" / "no"
18. paid(ученик посещает дополнительные платные занятия по предмету курса "Math / Portuguese") - бинарный признак: "yes" / "no"
19. activities(участвует во внеклассной деятельности) - бинарный признак: "yes" / "no"
20. nursery(посещал детский сад) - бинарный признак: "yes" / "no"
21. higher(заинтересован в получении высшего образования) - бинарный признак: "yes" / "no"
22. internet(дома есть подключение к интернету) - бинарный признак: "yes" / "no"
23. romantic(состоит в романтических отношениях) - бинарный признак: "yes" / "no"
24. famrel(взаимоотношения в семье) - категориальный признак: шкала от 1 до 5, где "1" - very bad, "5" - excellent
25. freetime(свободное время после школы) - категориальный признак: шкала от 1 до 5, где "1" - very low, "5" - very high
26. goout(прогулки на улице) - категориальный признак: шкала от 1 до 5, где "1" - very low, "5" - very high
27. Dalc(употребление алкоголя в будний день) - категориальный признак: шкала от 1 до 5, где "1" - very low, "5" - very high

28. Walc(употребление алкоголя в выходные дни) - категориальный признак: шкала от 1 до 5, где "1" - very low, "5" - very high
29. health(состояние здоровья) - категориальный признак: шкала от 1 до 5, где "1" - very bad, "5" - very good
- 30.absences(количество прогулов занятий в школе) - числовой признак: $0 \leq \text{absences} \leq 93$
- 31.G1(оценка за первый период) - числовой признак: $0 \leq G1 \leq 20$
- 32.G2(оценка за второй период) - числовой признак: $0 \leq G2 \leq 20$
- 33.G3(итоговая оценка) - числовой признак: $0 \leq G3 \leq 20$

2. Сколько категориальных признаков, какие?

В данном Датасете 15 категориальных признаков:

- 1.Medu(образование матери)
2. Fedu(образование отца)
- 3.Mjob(работа матери)
- 4.Fjob(работа отца)
- 5.reason(причина выбора школы)
- 6.guardian(опекун)
- 7.traveltime(сколько добираться до школы)
- 8.studytime(сколько часов в неделю уделяется учёбе)
- 9.failures(сколько раз ученик оставался на второй год)
- 10.famrel(взаимоотношения в семье)
- 11.freetime(свободное время после школы)
- 12.goout(прогулки на улице)
- 13.Dalc(употребление алкоголя в будний день)
- 14.Walc(употребление алкоголя в выходные дни)
15. health(состояние здоровья)

3. Столбец с максимальным количеством уникальных значений категориального признака?

В данном Датасете 10 столбцов с максимальным количеством уникальных значений категориального признака (см. Рисунок 4):

1)Medu = 5; 2)Fedu = 5; 3)Mjob = 5; 4)Fjob = 5; 5)famrel = 5; 6)freetime = 5; 7)goout = 5; 8)Dalc = 5; 9)Walc = 5; 10)health = 5;



```
: Data_test = ['Medu', 'Fedu', 'Mjob', 'Fjob', 'reason', 'guardian', 'traveltime', 'studytime', 'failures', 'famrel', 'freetime',
for column in Data.columns:
    if column in Data_test:
        print(column, Data[column].nunique())
```

Medu 5
Fedu 5
Mjob 5
Fjob 5
reason 4
guardian 3
traveltime 4
studytime 4
failures 4
famrel 5
freetime 5
goout 5
Dalc 5
Walc 5
health 5

Рисунок 4. Результат применения функции nunique().

4. Есть ли бинарные признаки?

В данном Датасете 13 бинарных признаков:

- 1.school(школа)
- 2.sex(пол)
- 3.address(тип населённого пункта)
- 4.famsize(количество человек в семье)
- 5.Pstatus(состоят ли родители в браке)
- 6.schoolsup(ученик имеет дополнительную образовательную поддержку)
- 7.famsup(ученик получает помощь в образовании со стороны семьи)
- 8.paid(ученик посещает дополнительные платные занятия по предмету курса "Math / Portuguese")
- 9.activities(участвует во внеклассной деятельности)
- 10.nursery(посещал детский сад)

11.higher(заинтересован в получении высшего образования)

12.internett(дома есть подключение к интернету)

13.romantic(состоит в романтических отношениях)

5.Какие числовые признаки?

1.age(возраст)

2.absences(количество прогулов занятий в школе)

3.G1(оценка за первый период)

4.G2(оценка за второй период)

5.G3(итоговая оценка)

6.Есть ли пропуски?

Основываясь на результате info(), мы видим, что пропусков в Датасете нет (3-ий столбец) (см. Рисунок 5):

```
Data.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1044 entries, 0 to 1043
Data columns (total 33 columns):
#   Column      Non-Null Count  Dtype
---  -
0   school      1044 non-null   object
1   sex         1044 non-null   object
2   age         1044 non-null   int64
3   address     1044 non-null   object
4   famsize     1044 non-null   object
5   Pstatus     1044 non-null   object
6   Medu        1044 non-null   int64
7   Fedu        1044 non-null   int64
8   Mjob        1044 non-null   object
9   Fjob        1044 non-null   object
10  reason      1044 non-null   object
11  guardian    1044 non-null   object
12  traveltime  1044 non-null   int64
13  studytime   1044 non-null   int64
14  failures    1044 non-null   int64
15  schoolsup    1044 non-null   object
16  famsup      1044 non-null   object
17  paid        1044 non-null   object
18  activities  1044 non-null   object
19  nursery     1044 non-null   object
20  higher      1044 non-null   object
21  internet    1044 non-null   object
22  romantic    1044 non-null   object
23  famrel      1044 non-null   int64
24  freetime    1044 non-null   int64
25  goout       1044 non-null   int64
26  Dalc        1044 non-null   int64
27  Walc        1044 non-null   int64
28  health      1044 non-null   int64
29  absences    1044 non-null   int64
30  G1          1044 non-null   int64
31  G2          1044 non-null   int64
32  G3          1044 non-null   int64
dtypes: int64(16), object(17)
memory usage: 269.3+ KB
```

Рисунок 5. Результат применения функции info().

7. Сколько объектов с пропусками?

Ответ: ни одного.

8. Столбец с максимальным количеством пропусков?

Ответ: ни одного.

9. Есть ли на ваш взгляд выбросы, аномальные значения?

Ответ: аномальных значений и выбросов не наблюдается.

10. Столбец с максимальным средним значением после нормировки признаков через стандартное отклонение?

- 1) Про-нормируем числовые признаки через стандартные отклонения (см. Рисунок 6).
- 2) Преобразуем текстовые данные в числовые (см. Рисунок 6).

Нормируем признак для числовых признаков через стандартное отклонение:

```
Data['age'] = (Data['age'] - Data['age'].mean())/(math.sqrt(Data['age'].var()))
```

```
Data['absences'] = (Data['absences'] - Data['absences'].mean())/(math.sqrt(Data['absences'].var()))
```

```
Data['G1'] = (Data['G1'] - Data['G1'].mean())/(math.sqrt(Data['G1'].var()))
```

```
Data['G2'] = (Data['G2'] - Data['G2'].mean())/(math.sqrt(Data['G2'].var()))
```

```
Data['G3'] = (Data['G3'] - Data['G3'].mean())/(math.sqrt(Data['G3'].var()))
```

Преобразуем текстовые данные в числовые:

```
array = ['school', 'sex', 'address', 'famsize', 'Pstatus', 'Mjob', 'Fjob', 'reason', 'guardian', 'schoolsup', 'famsup', 'paid',  
for column in array:  
    Set = set(Data[column])  
    i = 0  
    for item in Set:  
        Data[column] = Data[column].replace(item, i)  
        i = i + 1
```

Рисунок 6. Нормируем числовые признаки через стандартные отклонения и преобразовываем текстовые данные в числовые.

- 3) Найдем максимальное среднее значение:

```
max_average = -math.inf  
for column in array:  
    if Data[column].mean() > max_average:  
        max_average = Data[column].mean()  
print(column,max_average)
```

Ответ: столбец с максимальным средним значением после нормировки признаков - romantic

11. Столбец с целевым признаком?

Ответ: поскольку нам будет необходимо построить прогноз итогового бала учащегося, то целевой признак - G3.

12. Сколько объектов попадает в тренировочную выборку при использовании train_test_split с параметрами test_size = 0.3, random_state = 42?

- 1) Выделим целевой признак (G3) и удалим его из Датасета (см. Рисунок 7).

- 2) Разделим данные на обучающую и тестовую выборку (см. Рисунок 7).

Выделим целевой признак(G3) и удалим его из Датасета

```
Data2 = Data.G3  
train = Data.drop('G3', axis = 1)
```

Разделяем данные на обучающую и тестовую выборку

```
X_train, X_test, Y_train, Y_test = train_test_split(train, Data2, test_size = 0.3, random_state = 42)  
train.shape
```

(1044, 32)

X_train

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	...	romantic	famrel	freetime	goout	Dalc	Walc	health	absences	G
447	0	1	0.220929	1	0	1	2	1	2	2	...	1	5	5	5	5	5	3	0.574094	-1.07716
694	1	0	-1.392007	0	1	0	4	3	2	2	...	1	5	2	2	1	1	5	0.574094	-1.07716
448	0	0	-0.585539	1	0	1	4	4	3	3	...	1	4	2	2	1	1	4	0.252034	1.60434
934	1	1	0.220929	0	0	1	1	1	2	2	...	0	4	3	3	1	2	4	-0.392087	0.26356
582	0	0	1.833865	0	1	0	1	1	1	2	...	0	1	4	4	1	1	5	-0.714147	-1.74754
...
87	1	0	-1.392007	0	0	1	4	2	2	2	...	0	5	3	3	1	3	1	-0.070027	1.26916
330	1	0	1.027397	0	0	1	2	1	2	2	...	1	4	3	1	1	1	5	0.898154	0.26356
466	0	1	-1.392007	0	0	1	3	1	2	0	...	0	3	2	3	1	3	4	-0.714147	-0.40678
121	1	1	-1.392007	0	0	1	2	2	0	0	...	0	5	5	4	1	2	5	0.252034	0.93396
860	1	1	0.220929	0	1	1	4	4	0	2	...	1	5	3	5	4	5	3	1.379245	0.26356

730 rows × 32 columns

Рисунок 7. Количество объектов в тестовой выборке, после удаления целевого признака (G3) из Датасета и разделения данных на обучающую и тестовую выборку.

13. Между какими признаками наблюдается линейная зависимость (корреляция)?

Проверим коэффициент VIF для всех признаков, и если он больше 10, то существует линейная зависимость (см. Рисунок 8).

```
vif.loc[vif['VIF'] > 10.0]
```

	column_name	VIF
6	Medu	13.219419
7	Fedu	10.361689
20	higher	13.300180
23	famrel	17.835771
24	freetime	12.320756
25	goout	11.915192

Рисунок 8. Результат применения VIF.

Ответ: Линейная зависимость наблюдается у следующий признаков: Medu, Fedu, higher, famrel, freetime, goout.

14. Сколько признаков достаточно для объяснения 90% дисперсии после применения метода PCA?

Ответ: Для объяснения 90% дисперсии - достаточно 24 признаков.

15. Какой признак вносит наибольший вклад в первую компоненту?

```
pca = PCA(n_components=24)
X = pca.fit(New_Data)

First_component = pca.components_[0]
i = 0
result = First_component[i]

for j in range(0, len(First_component)):
    if abs(First_component[j]) > abs(result):
        result = First_component[j]
        i = j
print(i, result)
```

31 -0.33494913434254237

Рисунок 9. Поиск признака, вносящего наибольший вклад в первую компоненту.

Наибольший вклад в первую компоненту вносит 31-ый признак.

Ответ: это признак G2

Код решения задач приведён в “Приложение 5”.

Задача №6

Балашов Д. С. - Тема: "Употребление алкоголя учащимися"

Данные были получены в ходе опроса учащихся, изучающих математику и португальский язык в средней школе.

Наборы данных student-mat.csv (курс математики) и student-por.csv (курс португальского языка):

Набор данных №1:

<https://www.kaggle.com/uciml/student-alcohol-consumption?select=student-mat.csv>

Набор данных №2:

<https://www.kaggle.com/uciml/student-alcohol-consumption?select=student-por.csv>

№1. Построить прогноз итогового бала учащегося.

Построим модель линейной регрессии (см. Рисунок 10).

```
model = LinearRegression().fit(X_train,Y_train)
print (model.score(X_test,Y_test))
```

0.8354351069850131

Рисунок 10. Построение линейной регрессии и результат выполнения функции score(). $R^2 = 0.8354351069850131$

Найдем значения коэффициентов в нашей модели (см. Рисунок 11).

```
print('intercept:', model.intercept_)
print('slope:', model.coef_)
```

```
intercept: 0.16987376643217875
slope: [ 0.02563716  0.01169818 -0.01950017 -0.01566681 -0.01634197  0.0074082
 -0.01024713  0.0065342  -0.00298944 -0.02094183 -0.05491112  0.04814894
 -0.01929104 -0.11166943  0.07740339  0.04593801 -0.07089452 -0.03005844
 -0.00711277 -0.00707154 -0.03755929 -0.00923924 -0.01344943  0.00997318
 -0.00732745  0.05021778  0.12701637  0.7973484 ]
```

Рисунок 11. Нахождение значения коэффициентов в модели.

Тогда, линейная регрессия итогового бала (G3) на все параметры выглядит следующим образом:

$$\begin{aligned} G3 = & 0.16987376643217875 + 0.02563716033429331 * \text{school} + \\ & 0.01169817521405268 * \text{sex} + (-0.019500166793284434) * \text{age} + \\ & (-0.015666814055066567) * \text{address} + (-0.016341973485239178) * \text{famsize} + \\ & 0.007408199119350586 * \text{Pstatus} + (-0.010247132598149907) * \text{Fedu} + \end{aligned}$$

0.006534203835179059 * Mjob + (-0.002989437286947814) * Fjob +
 (-0.020941826441747944) * reason + (-0.0549111209231241) * guardian +
 0.04814894291335772 * traveltime + (-0.019291040061392817) * studytime +
 (-0.11166943202709809) * failures + 0.0774033856006938 * schoolsup +
 0.045938010732101374 * famsup + (-0.07089451684886684) * paid +
 (-0.03005844014350527) * activities + (-0.00711277454691548) * nursery +
 (-0.007071539538965607) * internet + (-0.03755928669946984) * romantic +
 (-0.009239242193155701) * freetime + (-0.01344942615766682) * Dalc +
 0.00997318440366965 * Walc + (-0.007327450384538271) * health +
 0.050217781870780465 * absences + 0.12701636592953505 * G1 +
 0.7973484032613066 * G2

№2. Насколько сильно влияет употребление алкоголя?

Чтобы выявить, насколько сильно употребление алкоголя влияет на итоговый бал учащегося, построим линейную регрессию G3(итоговый бал) на параметры Dalc(употребление алкоголя в будний день) и Walc(употребление алкоголя в выходные дни).

Построим модель линейной регрессии (см. Рисунок 12).

```
Alc_Data = Data.loc[:, Data.columns.isin(['Dalc', 'Walc'])]
data = Data.loc[:, Data.columns.isin(['G3'])]
```

```
model_Alc = LinearRegression().fit(Alc_Data, Data2)
print(model_Alc.score(Alc_Data, Data2))
```

```
0.018754503525254318
```

Рисунок 12. Построение линейной регрессии и результат выполнения функции score(). $R^2 = 0.018754503525254318$

Найдем значения коэффициентов в нашей модели (см. Рисунок 13).

```
print('intercept:', model_Alc.intercept_)
print('slope:', model_Alc.coef_)
```

```
intercept: 0.25492506596432607
slope: [-0.1031551 -0.04411732]
```

Рисунок 13. Нахождение значения коэффициентов в модели.

Тогда, линейная регрессия итогового бала (G3) на параметры Dalc и Walc выглядит следующим образом:

$$G3 = 0.25492506596432607 + (-0.10315510322966899) * Dalc + (-0.044117319844223485) * Walc$$

Вывод: Как видно из данной модели, употребление алкоголя негативно сказывается на итоговом бале учащегося, причем употребление алкоголя в будние дни сильнее влияет на итоговую оценку, чем употребление алкоголя в выходные дни.

№3. Какие параметры вносят наибольший вклад в предсказание?

Хоть потребление алкоголя и негативно сказывается на итоговом бале учащегося, но коэффициенты перед параметрами Dalc(употребление алкоголя в будний день) и Walc(употребление алкоголя в выходные дни) довольно маленькие. Таким образом, эти параметры не вносят наибольший вклад в предсказание результата итоговой оценки.

Вычислим среднее, максимальное и минимальное значение среди коэффициентов модели (см. *Рисунок 14*).

```
middle = model.coef_.mean()
print("Среднее значение = ",middle)
max = model.coef_.max()
print("Максимальное значение = ",max)
min = model.coef_.min()
print("Минимальное значение = ",min)
```

```
Среднее значение = 0.026894721179613775
Максимальное значение = 0.7973484032613066
Минимальное значение = -0.11166943202709809
```

Рисунок 14. Вычисление среднего, максимального и минимального значения среди коэффициентов модели.

Найдем параметры, вносящие наибольший вклад в предсказание (см. *Рисунок 15*).

```
vail = model.coef_[0].mean()
for i in range(len(train.columns)):
    if abs(model.coef_[i].max()) > vail:
        print(train.columns[i], "=", model.coef_[i].max())
```

```
guardian = -0.0549111209231241
traveltime = 0.04814894291335772
failures = -0.11166943202709809
schoolsup = 0.0774033856006938
famsup = 0.045938010732101374
paid = -0.07089451684886684
activities = -0.03005844014350527
romantic = -0.03755928669946984
absences = 0.050217781870780465
G1 = 0.12701636592953505
G2 = 0.7973484032613066
```

Рисунок 15. Параметры, которые вносят наибольший вклад в предсказание.

Найдем параметры, который вносят далеко не основной вклад в предсказание результата итоговой оценки (см. *Рисунок 16*).

```
vail = model.coef_[0].max()
for i in range(len(train.columns)):
    if abs(model.coef_[i].max()) < vail:
        print(train.columns[i], "=", model.coef_[i].max())
```

```
sex = 0.01169817521405268
age = -0.019500166793284434
address = -0.015666814055066567
famsize = -0.016341973485239178
Pstatus = 0.007408199119350586
Fedu = -0.010247132598149907
Mjob = 0.006534203835179059
Fjob = -0.002989437286947814
reason = -0.020941826441747944
studytime = -0.019291040061392817
nursery = -0.00711277454691548
internet = -0.007071539538965607
freetime = -0.009239242193155701
Dalc = -0.01344942615766682
Walc = 0.00997318440366965
health = -0.007327450384538271
```

Рисунок 16. Параметры, который вносят далеко не основной вклад в предсказание результата итоговой оценки.

№4. Построить графики статистической оценки параметров.

Выведем статистику модели в виде таблицы:

OLS Regression Results						
Dep. Variable:	G3	R-squared (uncentered):	0.840			
Model:	OLS	Adj. R-squared (uncentered):	0.835			
Method:	Least Squares	F-statistic:	190.1			
Date:	Fri, 04 Jun 2021	Prob (F-statistic):	0.00			
Time:	23:51:26	Log-Likelihood:	-525.21			
No. Observations:	1044	AIC:	1106			
Df Residuals:	1016	BIC:	1245.			
Df Model:	28					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
school	0.0306	0.034	0.901	0.368	-0.036	0.097
sex	0.0057	0.028	0.203	0.839	-0.049	0.060
age	-0.0113	0.014	-0.779	0.436	-0.040	0.017
address	-0.0275	0.032	-0.864	0.388	-0.090	0.035
famsize	0.0040	0.029	0.139	0.889	-0.053	0.061
Pstatus	0.0524	0.042	1.257	0.209	-0.029	0.134
Fedu	-0.0115	0.013	-0.922	0.357	-0.036	0.013
Mjob	0.0046	0.008	0.547	0.584	-0.012	0.021
Fjob	0.0022	0.012	0.192	0.847	-0.021	0.025
reason	-0.0086	0.013	-0.657	0.511	-0.034	0.017
guardian	-0.0213	0.025	-0.855	0.393	-0.070	0.028
traveltime	0.0382	0.018	2.127	0.034	0.003	0.073
studytime	-0.0042	0.016	-0.269	0.788	-0.035	0.027
failures	-0.0706	0.022	-3.229	0.001	-0.113	-0.028
schoolsup	0.0249	0.042	0.587	0.557	-0.058	0.108
famsup	0.0525	0.027	1.916	0.056	-0.001	0.106
paid	-0.0887	0.032	-2.754	0.006	-0.152	-0.026
activities	-0.0268	0.026	-1.028	0.304	-0.078	0.024
nursery	-0.0205	0.031	-0.652	0.515	-0.082	0.041
internet	0.0108	0.032	0.334	0.739	-0.053	0.074
romantic	-0.0294	0.027	-1.072	0.284	-0.083	0.024
freetime	0.0058	0.012	0.498	0.619	-0.017	0.029
Dalc	-0.0187	0.018	-1.023	0.306	-0.055	0.017
Walc	0.0126	0.013	0.961	0.337	-0.013	0.038
health	0.0030	0.009	0.350	0.726	-0.014	0.020
absences	0.0465	0.014	3.427	0.001	0.020	0.073
G1	0.1026	0.025	4.025	0.000	0.053	0.153
G2	0.8168	0.025	32.614	0.000	0.768	0.866
Omnibus:	661.814	Durbin-Watson:	1.800			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	8158.941			
Skew:	-2.752	Prob(JB):	0.00			
Kurtosis:	15.540	Cond. No.	25.7			

Рисунок 17. Статистика линейной регрессии итогового бала (G3) на все параметры.

Построим диаграммы:

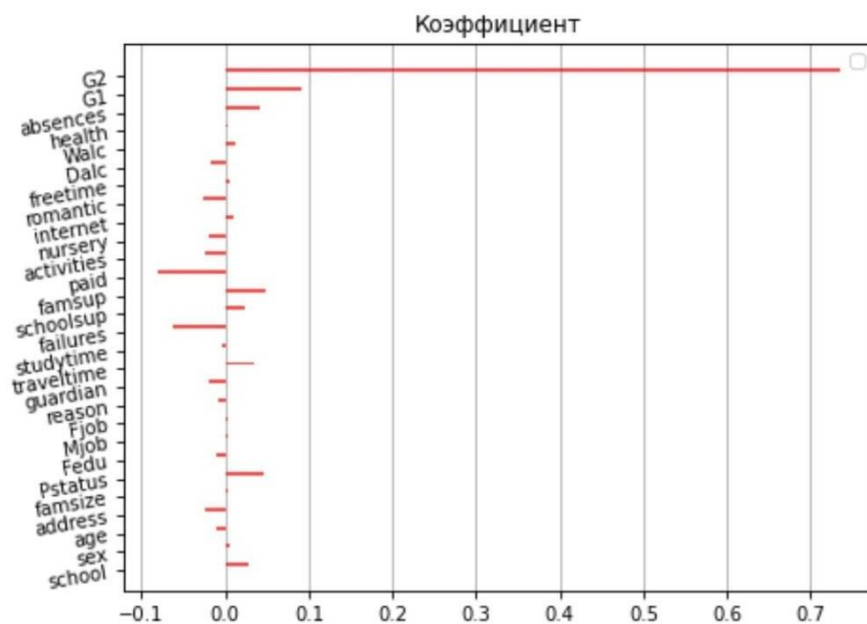


Рисунок 18. Графики статистической оценки параметра: “Коэффициент”.

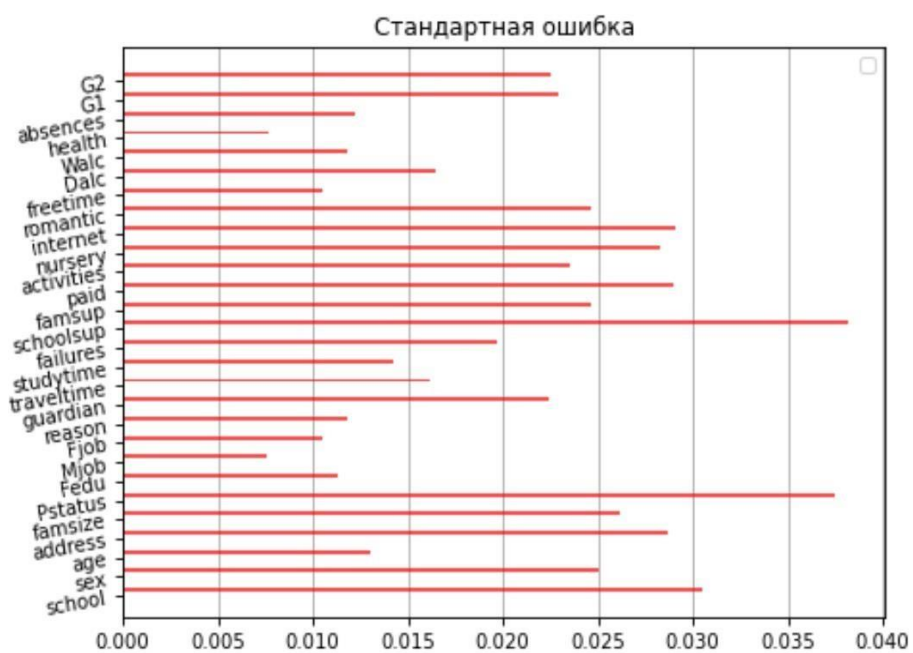


Рисунок 19. Графики статистической оценки параметра: “Стандартная ошибка”.

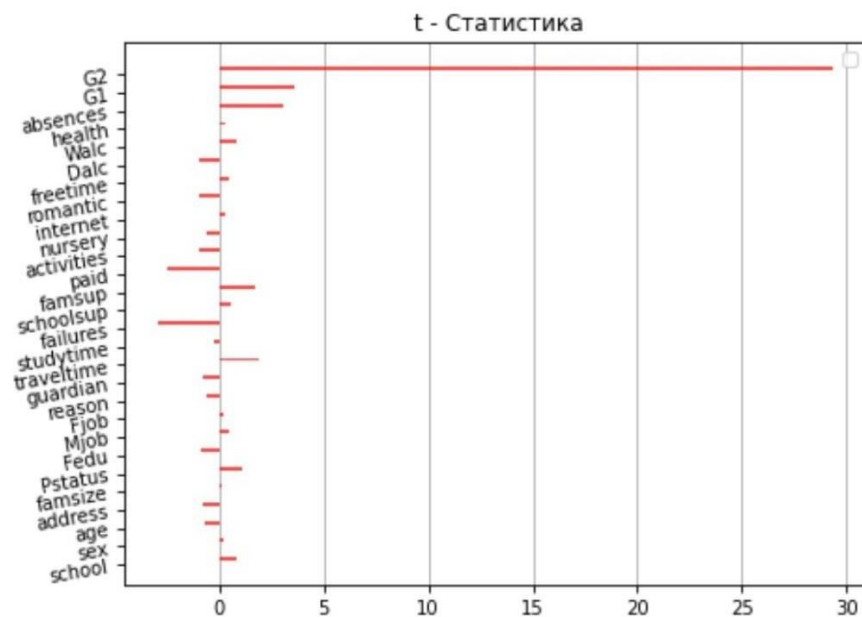


Рисунок 20. Графики статистической оценки параметра: “t - Статистика”.

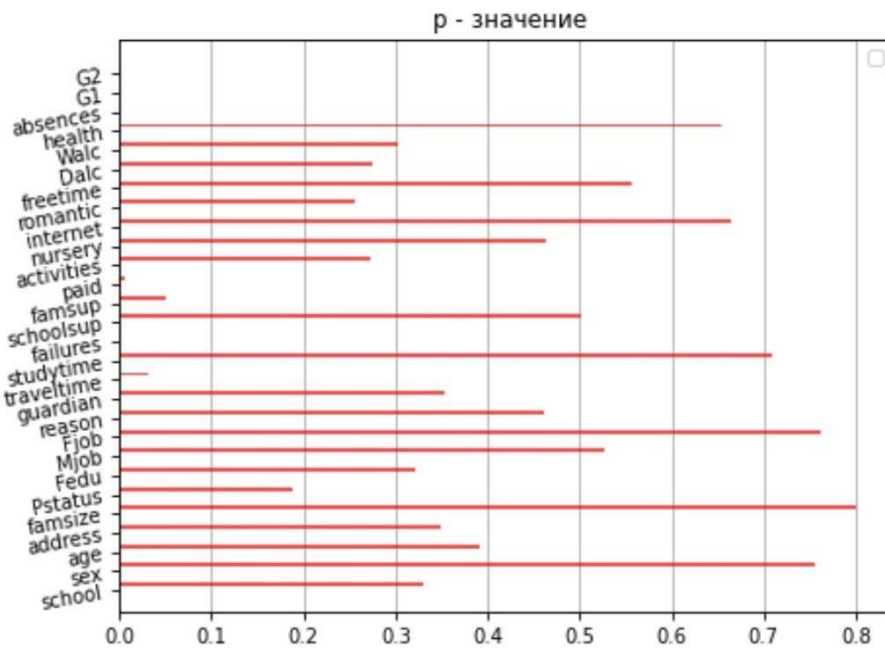


Рисунок 21. Графики статистической оценки параметра: “p - Значение”.

№5. Применить алгоритмы кластеризации.

```
dbscan = DBSCAN(eps=3.14, min_samples=2.5)

dbscan.fit(X_train)
pca = PCA(n_components=2).fit(X_train)
pca_2d = pca.transform(X_train)
print(set(dbscan.labels_))

{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, -1}

matplotlib.pyplot.figure(figsize=(10, 5))
colors = matplotlib.cm.rainbow(numpy.linspace(0, 1, len(set(dbscan.labels_))))
for i in range(0, pca_2d.shape[0]):
    if dbscan.labels_[i] == -1:
        c0 = matplotlib.pyplot.scatter(pca_2d[i, 0], pca_2d[i, 1], c = 'grey', marker='*')
    else:
        matplotlib.pyplot.scatter(pca_2d[i, 0], pca_2d[i, 1], c = [colors[dbscan.labels_[i]]], marker='o')
matplotlib.pyplot.title('результат работы DBSCAN')
matplotlib.pyplot.show()
```

Рисунок 22. Применяем алгоритмы кластеризации.

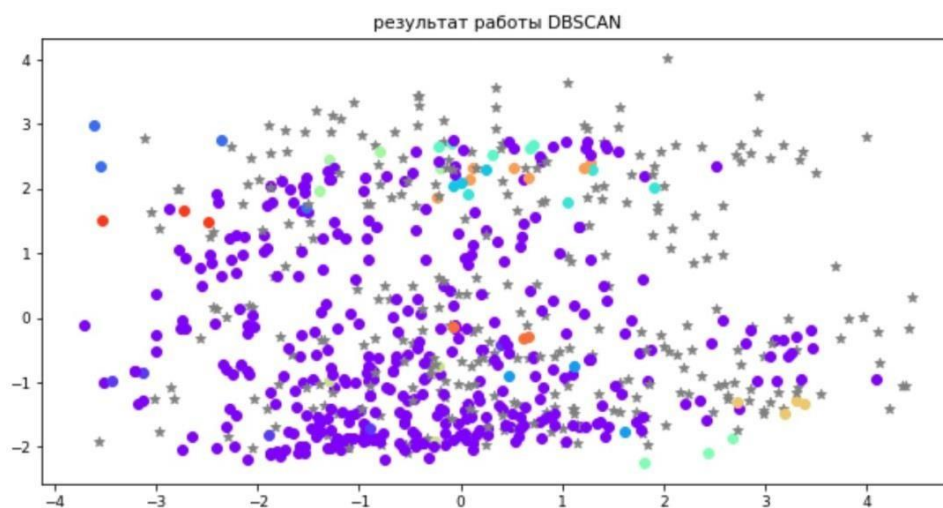


Рисунок 23. Результат работы DBSCAN.

Вывод: Как видно из графика, у данной модели нет явных кластеров.

№6. Что общего между объектами в каждом кластере?

Ответ: Нельзя найти общие признаки у кластеров, так как кластеров нет.

Вывод:

Мы построили прогноз итогового бала учащегося, и выявили, что хоть употребление алкоголя негативно влияет на итоговую оценку, но данный параметр не является определяющим фактором предсказания.

Так же, мы применили алгоритм кластеризации на данную модель, и обнаружили отсутствие явных кластеров. На основе всех полученных данных можно сделать вывод, что алгоритм кластеризации не подходит для данной модели.

Заключение

В первой задаче были рассмотрены две зависимости с одинаковой объясняемой переменной и разными объясняющими переменными. В первой части второй задаче, на наличие линейной зависимости, были проверены регрессоры рассматриваемой модели, была получена наилучшая модель по доле объяснённого разброса в данных R^2 . Во второй части этой задачи, были найдены доверительные интервалы для всех коэффициентов в рассматриваемой модели, а также был оценён доверительный интервал для одного прогноза рассматриваемой модели. В третьей задаче была построена модель линейной регрессии, на основе которой был сделан вывод, что наибольшую зарплату получают мужчины более молодого возраста, с большой продолжительностью рабочей недели. В четвертой задаче были построены классификаторы – случайный лес и решающие дерево. Основываясь на результате метрик классификаторов, был сделан вывод, что классификатор решающее дерево не подходит для данного Датасета. В пятой задаче был проведен анализ Датасета (из задачи 6), а также была проведена полная структуризация его данных. В шестой задаче мы построили прогноз итогового бала учащегося, и попробовали применить алгоритм кластеризации для получившейся модели. На основе всех полученных данных можно сделать вывод, что алгоритм кластеризации не подходит для рассматриваемых нами набора данных.

Список литературы

1. Грас Дж. Data Science. Наука о данных с нуля. [Текст]/ Дж. Грас – Санкт-Петербург: «БХВ-Петербург», 2017. - 63 – 65, 199 – 204 с. [1]
2. Florian O. Introduction to Econometrics with R. [Текст]/ O. Florian, V. Viers, J. M. Robin, P. Villedieu, G. Kenedi – Essen: University of Duisburg-Essen, 2020. - 63 – 65, 199 – 204 с. [2]
3. Stock J. H. Introduction to econometrics. [Текст]/ J. H. Stock, M. W. Watson – New York : Pearson, 2012. - 111 – 148, 186 – 220 с. [3]
4. Peng R. D. R programming for data science. [Текст]/ R. D. Peng – Leanpub, 2016. - 86 – 181 с. [4]
5. Орлов А. И. Прикладная статистика. [Текст]/ А. И. Орлов – Москва: «Экзамен», 2004. - 23 – 28 с. [5]
6. Орлов А. И. Эконометрика. Учебник для вузов. [Текст]/ А. И. Орлов – Москва: «Экзамен», 2003. - 67 – 74 с. [6]
7. Плошко Б. Г. История статистики: Учебное пособие. [Текст]/ Б. Г. Плошко, И. И. Елисеева – Москва: «Финансы и статистика», 1990. - 50 – 63, 151 – 160 с. [7]

Приложение

“Приложение 1”.

```
library("lmtest")  
library("GGally")  
library("car")
```

```
data = swiss  
help(swiss)
```

```
data  
summary(data)  
ggpairs(data)
```

```
model_agr_ex = lm(Agriculture~Examination, data)  
model_agr_ex  
summary(model_agr_ex)
```

```
plot(model_agr_ex) + abline(a = 82.88, b = -1.95, col = "red")
```

```
model_agr_F = lm(Agriculture~Fertility, data)  
model_agr_F  
summary(model_agr_F)
```

```
plot(model_agr_F) + abline(a = 5.63, b = 0.64, col = "red")
```

```
var(data$Agriculture)  
sd(data$Agriculture)  
mean(data$Agriculture)
```

```
var(data$Examination)  
sd(data$Examination)  
mean(data$Examination)
```

```
var(data$Fertility)  
sd(data$Fertility)  
mean(data$Fertility)
```

“Приложение 2”.

```
library("lmtest")  
library("GGally")  
library("car")
```

```
data = swiss  
help(swiss)
```

```
vif(model)  
model_test_1 = lm(Agriculture ~ Examination + Infant.Mortality, data) model_test_1  
summary(model_test_1)
```

```
model_test_2 = lm(Examination ~ Agriculture + Infant.Mortality, data) model_test_2  
summary(model_test_2)
```

```
model_test_3 = lm(Infant.Mortality ~ Agriculture + Examination , data)  
model_test_3  
summary(model_test_3)
```

```
model = lm(Fertility ~ Agriculture + Examination + Infant.Mortality, data)  
model  
summary(model)
```

```
model = lm(Fertility ~ Examination + Infant.Mortality, data)  
model  
summary(model)
```

```
model = lm(Fertility ~ log(Examination) + log(Infant.Mortality), data)  
model  
summary(model)  
vif(model)
```

```
model = lm(log(Fertility) ~ log(Examination) + log(Infant.Mortality), data)  
model  
summary(model)  
vif(model)
```

```
model = lm(Fertility ~ log(Examination) + Infant.Mortality, data)  
model  
summary(model)  
vif(model)
```

```

model = lm(Fertility ~ Examination + log(Infant.Mortality), data)
model
summary(model)
vif(model)

```

```

model_1 = lm(Fertility ~ Examination + Infant.Mortality + I(Examination^2) +
I(Infant.Mortality^2) + I(Examination*Infant.Mortality), data)
model_1
summary(model_1)
vif(model_1)

```

```

model_2 = lm(Fertility ~ Examination + I(Examination^2) + I(Infant.Mortality^2) +
I(Examination*Infant.Mortality), data)
model_2
summary(model_2)
vif(model_2)

```

```

model_3 = lm(Fertility ~ I(Examination^2) + I(Infant.Mortality^2) +
I(Examination*Infant.Mortality), data)
model_3
summary(model_3)
vif(model_3)

```

```

model_4 = lm(Fertility ~ I(Examination^2) + I(Infant.Mortality^2), data)
model_4
summary(model_4)
vif(model_4)

```

```

model = lm(Fertility ~ Agriculture + Examination + Infant.Mortality, data)
model
summary(model)

```

```

se = 0.07975
t = qt(0.975, df = 43)
model$coefficients[2] - t * se
model$coefficients[2] + t * se
confint(model, level = 0.95)

```

```

se = 0.22811
t = qt(0.975, df = 43)
model$coefficients[3] - t * se
model$coefficients[3] + t * se
confint(model, level = 0.95)

```



```
se = 0.45513
t = qt(0.975, df = 43)
model$coefficients[4] - t * se
```

```
model$coefficients[4] + t * se
confint(model, level = 0.95)
```

```
se = 12.82691
t = qt(0.975, df = 43)
model$coefficients[1] - t * se
model$coefficients[1] + t * se
confint(model, level = 0.95)
```

```
model = lm(Fertility ~ Agriculture + Examination + Infant.Mortality, data)
model
summary(model)
```

```
new.data = data.frame(Agriculture = 10, Examination = 30, Infant.Mortality = 20)
predict(model, new.data, interval = "confidence")
```

“Приложение 3”.

```
data_2 = select(data_1, salary, age, sex, higher_educ, status2, dur, wed1, wed2, wed3)
```

```
data_2 = na.omit(data_2)
```

```
glimpse(data_2)
```

```
model1 = lm(data = data_2, salary ~ age + sex + higher_educ + status2 + dur + wed1 + wed2 + wed3)
summary(model1)
vif(model1)
```

```
model2 = lm(data = data_2, salary ~ age + sex + dur)
```

```
summary(model2)
vif(model2)
```

```
model3 = lm(salary ~ age + log(age) + sex + dur + log(dur), data = data_2)
```

```
summary(model3)
vif(model3)
```

```
model4 = lm(salary ~ age + log(age) + sex + dur, data = data_2)
```

```
summary(model4)
vif(model4)
```

```

model5 = lm(salary~log(age) + sex + dur, data = data_2)

summary(model5)
vif(model5)

model6 = lm(salary~age + I(age^0.1) + sex + dur+ I(dur^0.1),data = data_2)
summary(model6)
vif(model6)

model7 = lm(salary~age + I(age^0.1) + sex + dur, data = data_2)

summary(model7)
vif(model7)

model8 = lm(salary~I(age^0.1) + sex + dur, data = data_2)

summary(model8)
vif(model8)

model9 = lm(salary~I(age^0.2) + sex + dur, data = data_2)

summary(model9)
vif(model9)

model10 = lm(salary~I(age^0.3) + sex + dur, data = data_2)

summary(model10)
vif(model10)

model11 = lm(salary~I(age^0.4) + sex + dur, data = data_2)

summary(model11)
vif(model11)

model12 = lm(salary~I(age^0.5) + sex + dur, data = data_2)

summary(model12)
vif(model12)

model13 = lm(salary~I(age^0.9) + sex + dur, data = data_2)

summary(model13)
vif(model13)

model14 = lm(salary~I(age^1.1) + sex + dur, data = data_2)

summary(model14)
vif(model14)

```

```

model15 = lm(salary~I(age^1.2) + sex + dur, data = data_2)

summary(model15)
vif(model15)

model16 = lm(salary~I(age^1.3) + sex + dur, data = data_2)

summary(model16)
vif(model16)

model17 = lm(salary~I(age^1.4) + sex + dur, data = data_2)

summary(model17)
vif(model17)

model18 = lm(salary~I(age^1.9) + sex + dur, data = data_2)

summary(model18)
vif(model18)

model19 = lm(salary~I(age^2) + sex + dur, data = data_2)

summary(model19)
vif(model19)

data_3 = subset(data_2, status2 == 1)

data_3
data_4 = subset(data_3, wed1 == 1)

data_4
data_5 = subset(data_2, wed2 == 1)

data_5
data_6 = subset(data_5, higher_educ == 0)

data_6

model_subset1 = lm(data = data_4, salary~dur + sex + I(age^2))

summary(model_subset1)

model_subset2 = lm(data = data_6, salary~dur + sex + I(age^2))
summary(model_subset2)

```

“Приложение 4”.

```
import pandas
import numpy
import math
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import GridSearchCV
from sklearn.tree import plot_tree
from matplotlib import pyplot as plt
data = pandas.read_csv('aug_test.csv', index_col='city')
data = data[data.columns.drop(['enrollee_id'])]

data.head()

data.info()

data['gender'] = numpy.where(data['gender'] == 'Male', 1, 0)
data['education_level'] = numpy.where(data['education_level'] == 'Graduate', 0, 1)
data['relevent_experience'] = numpy.where(data['relevent_experience'] == 'No relevent
experience',0,1)

array = ['enrolled_university', 'last_new_job', 'major_discipline', 'company_type',
'company_size', 'experience']

for column in array:
    Set = set(data[column])
    i = 0
    for item in
        Set:
            data[column] = data[column].replace(item, i)
            i = i + 1

data.loc[data.experience == 'NaN' , 'experience'] = 0
data.loc[data.last_new_job == 'never' , 'last_new_job'] = 0
data.loc[data.last_new_job == 'NaN' , 'major_discipline'] = 0
data.loc[data.last_new_job == 'NaN' , 'company_size'] = 0

data

data.info()

Data2 = data.education_level
train = data.drop('education_level', axis = 1)

train.info()
data.columns
```

```
X_train, X_test, Y_train, Y_test = train_test_split(train, Data2, test_size = 0.3, random_state = 42)
train.shape
```

```
Tree = DecisionTreeClassifier(random_state=42, max_depth = 4)
Tree = Tree.fit(X_train, Y_train)
Tree
```

```
fig = plt.figure(figsize=(30,30))
_ = plot_tree(Tree, filled=True, class_names=True)
```

```
print("f1:"+str(numpy.average(cross_val_score(Tree, X_train, Y_train, scoring='f1'))))
print("precision:"+str(numpy.average(cross_val_score(Tree, X_train, Y_train,
scoring='precision'))))
print("recall:"+str(numpy.average(cross_val_score(Tree, X_train, Y_train, scoring='recall'))))
print("f1:"+str(numpy.average(cross_val_score(Tree, X_test, Y_test, scoring='f1'))))
print("precision:"+str(numpy.average(cross_val_score(Tree, X_test, Y_test,
scoring='precision'))))
print("recall:"+str(numpy.average(cross_val_score(Tree, X_test, Y_test, scoring='recall'))))
tree = DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None,
max_features=None, max_leaf_nodes=None, min_impurity_decrease=0, min_samples_leaf=2,
min_samples_split=3, min_weight_fraction_leaf=0, random_state=35, splitter='best')
```

```
params =
{
'max_depth':
list(range(1, 25)),
'min_samples_split': [2, 3, 4, 5, 6, 7, 8, 9, 10]
}
```

```
gridsearch = GridSearchCV(cv=3, error_score='raise-deprecating', estimator=tree, n_jobs=-1,
param_grid=params, pre_dispatch='2*n_jobs', refit=True,
return_train_score='warn', verbose=1)
```

```
gridsearch.fit(X_train, Y_train)
Tree = gridsearch.best_estimator_
Tree
```

```
print("f1:"+str(numpy.average(cross_val_score(Tree, X_train, Y_train, scoring='f1'))))
print("precision:"+str(numpy.average(cross_val_score(Tree, X_train, Y_train,
scoring='precision'))))
print("recall:"+str(numpy.average(cross_val_score(Tree, X_train, Y_train, scoring='recall'))))
print("f1:"+str(numpy.average(cross_val_score(Tree, X_test, Y_test, scoring='f1'))))
print("precision:"+str(numpy.average(cross_val_score(Tree, X_test, Y_test,
scoring='precision'))))
print("recall:"+str(numpy.average(cross_val_score(Tree, X_test, Y_test, scoring='recall'))))
```

```
fig = plt.figure(figsize=(30,30))
_ = plot_tree(Tree, filled=True, class_names=True)
```

```

from sklearn.ensemble import RandomForestClassifier

param_grid = { 'n_estimators': [50, 100, 150], 'max_features': ['auto'], 'max_depth': list(range(1,
10)), 'criterion': ['gini']}
RandForCrit = GridSearchCV(estimator = RandomForestClassifier(), param_grid =
param_grid, cv = 5, refit = True)
RandForCrit.fit(X_train, Y_train)
RandForCrit.predict(X_test)
print("f1:" + str(numpy.average(cross_val_score(Tree, X_train, Y_train, scoring='f1'))))
print("precision:" + str(numpy.average(cross_val_score(Tree, X_train, Y_train,
scoring='precision'))))
print("recall:" + str(numpy.average(cross_val_score(Tree, X_train, Y_train, scoring='recall'))))
print("f1:" + str(numpy.average(cross_val_score(Tree, X_test, Y_test, scoring='f1'))))
print("precision:" + str(numpy.average(cross_val_score(Tree, X_test, Y_test,
scoring='precision'))))
print("recall:" + str(numpy.average(cross_val_score(Tree, X_test, Y_test, scoring='recall'))))

RFC = GridSearchCV(estimator=RandomForestClassifier(), param_grid=param_grid, cv= 5,
refit = True)
RFC.fit(X_train, Y_train)

len(RFC.best_estimator_.estimators_)

```

“Приложение 5”.

```

!pip install pandas
!pip install sklearn
!pip install statsmodels

import pandas
import numpy
import math
from sklearn.model_selection import train_test_split
from statsmodels.stats.outliers_influence import variance_inflation_factor
from sklearn import preprocessing
from sklearn.decomposition import PCA

data1 = pandas.read_csv('student-por.csv')
data2 = pandas.read_csv('student-mat.csv')
Data = [data1, data1]
Data = pandas.concat(Data)

Data

data1.info()
data2.info()

Data = data1.append(data2, ignore_index=True)

```

Data

Data.info()

Data.columns

```
Data_test = ['Medu', 'Fedu', 'Mjob', 'Fjob', 'reason', 'guardian', 'traveltime', 'studytime', 'failures',
'famrel', 'freetime', 'goout', 'Dalc', 'Walc', 'health']
for column in Data.columns:
    if column in Data_test:
        print(column, Data[column].nunique())
```

Data.info()

```
for column in Data.columns:
    print(Data[column].value_counts())
```

```
Data['age'] = (Data['age'] - Data['age'].mean())/(math.sqrt(Data['age'].var()))
```

```
Data['absences'] = (Data['absences'] -
Data['absences'].mean())/(math.sqrt(Data['absences'].var()))
```

```
Data['G1'] = (Data['G1'] - Data['G1'].mean())/(math.sqrt(Data['G1'].var()))
```

```
Data['G2'] = (Data['G2'] - Data['G2'].mean())/(math.sqrt(Data['G2'].var()))
```

```
Data['G3'] = (Data['G3'] - Data['G3'].mean())/(math.sqrt(Data['G3'].var()))
```

```
array = ['school', 'sex', 'address', 'famsize', 'Pstatus', 'Mjob', 'Fjob', 'reason', 'guardian',
'schoolsup', 'famsup', 'paid', 'activities', 'nursery', 'higher', 'internet', 'romantic']
for column in array:
    Set = set(Data[column])
    i = 0
    for item in Set:
        Data[column] = Data[column].replace(item, i)
    i = i + 1
```

Data

```
max_average = -math.inf
for column in array:
    if Data[column].mean() > max_average:
        max_average = Data[column].mean()
    print(column, max_average)
```

```
Data2 = Data.G3
train = Data.drop('G3', axis = 1)
```

```

X_train, X_test, Y_train, Y_test = train_test_split(train, Data2, test_size = 0.3, random_state =
42)
train.shape

X_train

vif = pandas.DataFrame()
vif["column_name"] = train.columns
vif["VIF"] = [variance_inflation_factor(train.values, i) for i in range(len(train.columns))]

vif.loc[vif['VIF'] > 10.0]

New_Data = pandas.DataFrame(preprocessing.scale(X_train), columns = X_train.columns)

for i in range(1, len(New_Data.columns)):
pca = PCA(n_components=i)
pca.fit(New_Data)
print(i, sum(pca.explained_variance_ratio_))

pca = PCA(n_components=24)
X = pca.fit(New_Data)

First_component = pca.components_[0]
i = 0
result = First_component[i]

for j in range(0, len(First_component)):
if abs(First_component[j]) > abs(result):
result = First_component[j]
i = j
print(i, result)

New_Data.iloc[:, 31]

```

“Приложение 6”.

```

!pip install pandas
!pip install sklearn
!pip install statsmodels

import pandas
import numpy
import math
from sklearn.cluster import DBSCAN
from sklearn.decomposition import PCA
import matplotlib.cm
from statsmodels.stats.outliers_influence import variance_inflation_factor
import matplotlib.pyplot
import statsmodels.api
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split

```



```

import matplotlib as mpl
import matplotlib.dates as mdates
import datetime as dt
import csv

data1 = pandas.read_csv('student-por.csv')
data2 = pandas.read_csv('student-mat.csv')
Data = [data1, data1]
Data = pandas.concat(Data)
Data = data1.append(data2, ignore_index=True)

Data

arr = ['school', 'sex', 'address', 'famsize', 'Pstatus', 'Mjob', 'Fjob', 'reason', 'guardian', 'schoolsup',
'famsup', 'paid', 'activities', 'nursery', 'higher', 'internet', 'romantic']
for column in arr:
    Set = set(Data[column])
    i = 0
    for item in Set:
        Data[column] = Data[column].replace(item, i)
        i = i + 1

Data['age'] = (Data['age'] - Data['age'].mean())/(math.sqrt(Data['age'].var()))
Data['absences'] = (Data['absences'] -
Data['absences'].mean())/(math.sqrt(Data['absences'].var()))
Data['G1'] = (Data['G1'] - Data['G1'].mean())/(math.sqrt(Data['G1'].var()))
Data['G2'] = (Data['G2'] - Data['G2'].mean())/(math.sqrt(Data['G2'].var()))
Data['G3'] = (Data['G3'] - Data['G3'].mean())/(math.sqrt(Data['G3'].var()))

Data2 = Data.G3
train = Data.drop('G3', axis = 1)

vif = pandas.DataFrame()
vif["column_name"] = train.columns
vif["VIF"] = [variance_inflation_factor(train.values, i) for i in range(len(train.columns))]
vif.loc[vif["VIF"] > 10.0]

while not (vif.empty):
    vif = pandas.DataFrame()
    vif["column_name"] = train.columns
    vif["VIF"] = [variance_inflation_factor(train.values, i) for i in range(len(train.columns))]
    vif = vif.loc[vif["VIF"] >= 10.0]
    a = vif[vif["VIF"] == vif["VIF"].max()]["column_name"]
    print(a.array[0])
    train = train.drop(a, axis=1)

train

```

```

X_train, X_test, Y_train, Y_test = train_test_split(train, Data2, test_size = 0.3, random_state =
42)
train.shape

X_train

model = LinearRegression().fit(X_train, Y_train)
print (model.score(X_test, Y_test))

print('intercept:', model.intercept_)
print('slope:', model.coef_)

print('G3 = ', model.intercept_, '+')
for i in range(len(train.columns)-1):
print(model.coef_[i], '*', train.columns[i], '+')
print(model.coef_[len(train.columns)-1], '*', train.columns[len(train.columns)-1])

Alc_Data = Data.loc[:, Data.columns.isin(['Dalc', 'Walc'])]
data = Data.loc[:, Data.columns.isin(['G3'])]

model_Alc = LinearRegression().fit(Alc_Data, Data2)
print(model_Alc.score(Alc_Data, Data2))

print('intercept:', model_Alc.intercept_)
print('slope:', model_Alc.coef_)

print('G3 = ', model_Alc.intercept_, '+')
for i in range(len(Alc_Data.columns)-1):
print(model_Alc.coef_[i], '*', Alc_Data.columns[i], '+')
print(model_Alc.coef_[len(Alc_Data.columns)-
1], '*', Alc_Data.columns[len(Alc_Data.columns)-1])

middle = model.coef_.mean()
print("Среднее значение = ", middle)
max = model.coef_.max()
print("Максимальное значение = ", max)
min = model.coef_.min()
print("Минимальное значение = ", min)

vail = model.coef_[0].mean()
for i in range(len(train.columns)):
if abs(model.coef_[i].max()) > vail:
print(train.columns[i], "=", model.coef_[i].max())

vail = model.coef_[0].max()
for i in range(len(train.columns)):
if abs(model.coef_[i].max()) < vail:
print(train.columns[i], "=", model.coef_[i].max())

ols = statsmodels.api.OLS(data, train)
res = ols.fit()
print(res.summary())

```

```

dpi = 80
fig = matplotlib.pyplot.figure(dpi = dpi, figsize = (512 / dpi, 384 / dpi) )
matplotlib.pyplot.rcParams.update({'font.size': 9})

matplotlib.pyplot.title('Коэффициент')

ax = matplotlib.pyplot.axes()
ax.xaxis.grid(True, zorder = 1)

xs = range(len(train.columns))

matplotlib.pyplot.barh([x + 0.3 for x in xs], [ d * 0.9 for d in res.params],
height = 0.2, color = 'red', alpha = 0.7,
zorder = 2)

matplotlib.pyplot.xticks(xs, train.columns, rotation = 10)

matplotlib.pyplot.legend(loc='upper right')
fig.savefig('barshoris.png')

dpi = 80
fig = matplotlib.pyplot.figure(dpi = dpi, figsize = (512 / dpi, 384 / dpi) )
matplotlib.pyplot.rcParams.update({'font.size': 9})

matplotlib.pyplot.title('Стандартная ошибка')

ax = matplotlib.pyplot.axes()
ax.xaxis.grid(True, zorder = 1)

xs = range(len(train.columns))

matplotlib.pyplot.barh([x + 0.3 for x in xs], [ d * 0.9 for d in res.bse],
height = 0.2, color = 'red', alpha = 0.7,
zorder = 2)

matplotlib.pyplot.xticks(xs, train.columns, rotation = 10)

matplotlib.pyplot.legend(loc='upper right')
fig.savefig('barshoris.png')
dpi = 80
fig = matplotlib.pyplot.figure(dpi = dpi, figsize = (512 / dpi, 384 / dpi) )
matplotlib.pyplot.rcParams.update({'font.size': 9})

matplotlib.pyplot.title('t - Статистика')

ax = matplotlib.pyplot.axes()
ax.xaxis.grid(True, zorder = 1)
xs = range(len(train.columns))

```

```

matplotlib.pyplot.barh([x + 0.3 for x in xs], [ d * 0.9 for d in res.tvalues],
height = 0.2, color = 'red', alpha = 0.7,
zorder = 2)

matplotlib.pyplot.yticks(xs, train.columns, rotation = 10)

matplotlib.pyplot.legend(loc='upper right')
fig.savefig('barshoris.png')

dpi = 80
fig = matplotlib.pyplot.figure(dpi = dpi, figsize = (512 / dpi, 384 / dpi) )
matplotlib.pyplot.rcParams.update({'font.size': 9})

matplotlib.pyplot.title(' p - значение')

ax = matplotlib.pyplot.axes()
ax.xaxis.grid(True, zorder = 1)

xs = range(len(train.columns))

matplotlib.pyplot.barh([x + 0.3 for x in xs], [ d * 0.9 for d in res.pvalues],
height = 0.2, color = 'red', alpha = 0.7,
zorder = 2)

matplotlib.pyplot.yticks(xs, train.columns, rotation = 10)

matplotlib.pyplot.legend(loc='upper right')
fig.savefig('barshoris.png')

dbscan = DBSCAN(eps=3.14, min_samples=2.5)

dbscan.fit(X_train)
pca = PCA(n_components=2).fit(X_train)
pca_2d = pca.transform(X_train)
print(set(dbscan.labels_))

matplotlib.pyplot.figure(figsize=(10, 5))
colors = matplotlib.cm .rainbow(numpy.linspace(0, 1, len(set(dbscan.labels_))))
for i in range(0, pca_2d.shape[0]):
    if dbscan.labels_[i] == -1:
        c0 = matplotlib.pyplot.scatter(pca_2d[i, 0], pca_2d[i, 1], c = 'grey', marker='*')
    else:
        matplotlib.pyplot.scatter(pca_2d[i, 0], pca_2d[i, 1], c = [colors[dbscan.labels_[i]]], marker='o')
matplotlib.pyplot.title('результат работы DBSCAN')
matplotlib.pyplot.show()

```