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:** Классификация мошеннических операций с банковскими картами на основе набора данных «Credit Card Fraud Detection» (kaggle.com)

приказ университета о направлении на практику
490 – С от 09.02.2021 г.

Отчет представлен к рассмотрению:
Студент группы КМБО-01-20 _____ Малов И.М.
(расшифровка подписи)
«8» июня 2021 г.

Отчет утвержден.
Допущен к защите:

Руководитель практики от кафедры _____ Петрусевич Д.А.
(расшифровка подписи)
«3» июня 2021 г.

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-01-20 института кибернетики Малову Илье Максимовичу**

(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:** классификация мошеннических операций с банковскими картами на основе набора данных «Credit Card Fraud Detection» (kaggle.com).

**4. ОГРАНИЗАЦИОННО-МЕТОДИЧЕСКИЕ УКАЗАНИЯ:** Построить несколько бинарных классификаторов. Какие параметры вносят наибольший вклад при определении мошеннических операций? Являются ли мошеннические операции выбросами?

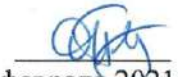
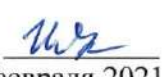
Заведующий кафедрой
высшей математики _____ Ю.И.Худак

«09» февраля 2021 г.

СОГЛАСОВАНО
Руководитель практики от кафедры:
«09» февраля 2021 г. _____ (Петрусевич Д.А.)
(подпись) (фамилия и инициалы)

Задание получил:
«09»февраля 2021 г. _____ (Малов И.М.)
(подпись) (фамилия и инициалы)

**ИНСТРУКТАЖ ПРОВЕДЕН:**

| Вид мероприятия | ФИО ответственного, подпись, дата | ФИО студента, подпись, дата |
|---|---|---|
| Охрана труда | Петрусевич Д.А.<br>  <br>«09» февраля 2021 г. | Малов И.М.<br>  <br>«09» февраля 2021 г. |
| Техника безопасности | Петрусевич Д.А.<br>  <br>«09» февраля 2021 г. | Малов И.М.<br>  <br>«09» февраля 2021 г. |
| Пожарная безопасность | Петрусевич Д.А.<br>  <br>«09» февраля 2021 г. | Малов И.М.<br>  <br>«09» февраля 2021 г. |
| Правила внутреннего распорядка | Петрусевич Д.А.<br>  <br>«09» февраля 2021 г. | Малов И.М.<br>  <br>«09» февраля 2021 г. |

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«МИРЭА - Российский технологический университет»**
**РТУ МИРЭА**

**РАБОЧИЙ ГРАФИК ПРОВЕДЕНИЯ** Научно-Исследовательской Работы

студента Малова И.М. 1 курса группы КМБО-01-20 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 / Малов И.М.

# Оглавление

ЗАДАЧА 1 ..... 2
ЗАДАЧА 2.1 ..... 4
ЗАДАЧА 2.2 ..... 6
ЗАДАЧА 3 ..... 8
ЗАДАЧА 4 ..... 13
ЗАДАЧА 5 ..... 16
ЗАКЛЮЧЕНИЕ ..... 19
СПИСОК ЛИТЕРАТУРЫ ..... 20
ПРИЛОЖЕНИЯ ..... 21

# Задача 1

Условие

Набор данных: swiss.
Объясняемая переменная: *Education.*
Регрессоры: *Fertility , Examination.*

1. Оцените среднее значение, дисперсию и СКО переменных, указанных во втором и третьем столбце.
2. Постройте зависимости вида y = a + bx, где y – объясняемая переменная, x – регрессор (для каждого варианта по две зависимости).
3. Оцените, насколько «хороша» модель по коэффициенту детерминации $R^2$?
4. Оцените, есть ли взаимосвязь между объясняемой переменной и объясняющей переменной (по значению p-статистики, «количеству звездочек» у регрессора в модели).

Решение

1. Для оценок среднего значения ,дисперсии и СКО будем использовать команды mean, var и sd соответственно .В результате выполнения команд имеем:
   - Среднее значение *Education* = 10.98
   - Среднее значение *Fertility* = 70.14
   - Среднее значение *Examination* = 16.49
   - Дисперсия *Education* = 92.46
   - Дисперсия *Fertility* = 156.04
   - Дисперсия *Examination* = 63.65
   - СКО *Education* = 9.62
   - СКО *Fertility* = 12.49
   - СКО *Examination* = 7.98

2. Для построения линейной зависимости используем команду lm. В результате выполнения команды для первой и второй модели имеем:
   1. y = 46.8179 - 0.5109x для *Education ~ Fertility*
   2. y = -2.9015 + 0.8418x для *Education ~ Examination*
3. Чтобы посмотреть $R^2$ воспользуемся командой summary. В результате её выполнения видим, что $R^2$ у первой модели = 0.44 – это значит, что модель 1 объясняет 44% колебаний переменной *Education* – меньше половины, но довольно неплохо для одного регрессора. Для второй модели $R^2$ = 0.49 –аналогично в сравнении с первой моделью.
4. P-статистика также показывается при выполнении команды summary. Для первой модели имеем очень низкие показатели p-статистики (3 звёздочки у каждого из параметров), что означает наличие сильной взаимосвязи между параметрами и объясняемой переменной. У второй модели регрессор *Examination* не имеет звезд у первого параметра ,то есть P-статистика показывает относительно большие значения, и имеет 3 звезды у второго параметра- P-статистика имеет относительно малые значения

Код решения задачи и сведения о проверенных моделях приведены в Приложении к задаче 1.

Выводы

В первой модели (*Education ~ Fertility*) есть причинно-следственная связь между поведением объясняемой переменной *Education* и регрессором *Fertility*(связь отрицательная), но она нелинейная и/или требует дополнительных регрессоров. Во второй модели (*Education ~ Examination*) причинно-следственная связь между объясняемой переменной *Education* и регрессором *Examination* прослеживается лучше, но она также не линейна и требует дополнительных регрессоров(связь положительная).

## Задача 2.1

Условие

Набор данных: swiss.
Объясняемая переменная: *Examination*.
Регрессоры: *Fertility, Catholic, Agriculture.*

1. Проверьте, что в наборе данных нет линейной зависимости (построить зависимости между переменными, указанными в варианте, и проверить, что $R^2$ в каждой из них невысокий). В случае, если $R^2$ большой, один из таких столбцов можно исключить из рассмотрения.
2. Постройте линейную модель зависимой переменной от указанных в варианте регрессоров по методу наименьших квадратов (команда lm пакета lmtest в языке R). Оценить, насколько хороша модель, согласно: 1) $R^2$, 2) p-значениям каждого коэффициента.
3. Введите в модель логарифмы регрессоров (если возможно). Сравнить модели и выбрать наилучшую.
4. Введите в модель всевозможные произведения пар регрессоров, в том числе квадраты регрессоров. Найдите одну или несколько наилучших моделей по доле объяснённого разброса в данных R2.

Решение

1. Проверим отсутствие зависимости между регрессорами с помощью команды lm,и рассмотрим значения $R^2$
   - *Fertility~Catholic* – $R^2 < 0.22$ => зависимости нет
   - *Fertility~Agriculture* – $R^2 < 0.13$ => зависимости нет
   - *Catholic~Agriculture* – $R^2 < 0.17$ => зависимости нет

Во всех случаях видно,что $R^2 < 0.25$ => регрессоры можно использовать вместе.

2. Построим модель, используя команду lm и воспользуемся командой summary.В результате её выполнения видим:
   1. $R^2 = 0.69$
   2. У *Catholic* ненадёжное значение p-статистики (1 звезда)

   Уберём из модели регрессор *Catholic*, как наименее значимый, и проверим, как изменится $R^2$:
   - $R^2 = 0.66$ – изменился на 0.03
   - $R^2 = 0.42$ – изменение на 0.24 => регрессор *Agriculture* лучше не исключать
   - $R^2 = 0.47$ – изменение на 0.19 => регрессор *Fertility* лучше не исключать

   Остановимся на *model = lm(Examination ~ Fertility + Agriculture + Catholic, data)*. Она имеет достаточно высокий $R^2$, и почти отличные показатели по p-статистике.
3. Введем в модель логарифмы для поиска наиболее хорошей комбинации регрессоров,не забывая проверять отсутствие линейной зависимости командой vif.Подробный код поиска наилучшей модели приведён в Приложении 1.

   Лучшая модель,даже по сравнению с исходной *model = lm(Examination ~ Fertility + I(log(Agriculture)) + Catholic , data)*

4. Попробуем тогда ввести в модель всевозможные произведения пар регрессоров, не забывая проверять отсутствие линейной зависимости. Подробный код поиска наилучшей модели приведён в Приложении 2.1.

Наилучшей среди моделей оказалась *model = lm(Examination ~ Fertility + I(Agriculture^2) + Catholic + I(log(Agriculture)), data),* у которой R~0.71 , но которая имеет посредственную p-статистику .

## Задача 2.2

Условие

Набор данных: swiss.
Объясняемая переменная: *Examination*.
Регрессоры: *Fertility, Catholic, Agriculture*.
Для зависимости, построенной при решении практического задания №2, оцените:

1. Доверительные интервалы для всех коэффициентов в модели, p = 95%.
2. Сделайте вывод о отвержении или невозможности отвергнуть статистическую гипотезу о том, что коэффициент равен 0.
3. Доверительный интервал для одного прогноза (p = 95%, набор значений регрессоров выбираете сами).

Решение

Имеем следующую модель: `model = lm(Examination ~ Fertility + Catholic + Agriculture, data)`

Таблица 1. Характеристики модели зависимости параметра: *Examination* от параметров *Fertility, Catholic, Agriculture* в наборе данных Swiss.

```
                Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 43.68234    4.10586  10.639 1.27e-13 ***
Fertility   -0.24582    0.06274  -3.918 0.000315 ***
 Catholic    -0.03953    0.01919  -2.060 0.045506 *
Agriculture -0.16431    0.03337  -4.923 1.30e-05 ***
```

5. Оценим доверительные интервалы для всех коэффициентов в модели (для p=95%):

Число степеней свободы в модели df = 43 - 4 = 39, и t-критерий Стьюдента тогда равен 2.022691.
Так как нам известны стандартные ошибки мы можем найти доверительные интервалы по формуле [x-at;x+at],где x-значения Estimate, a-значения Std.Error, и t=2.022691.

- Доверительный интервал свободного коэффициента: [35.3774 , 51.9872]
- Доверительный интервал *Fertility*: [-0.3727 , -0.1189]
- Доверительный интервал *Catholic*: [-0.0783 , -0.0007]
- Доверительный интервал *Agriculture*: [-0.2318 , -0.0968]

6. По доверительным интервалам сделаем вывод об отвержении или невозможности отвергнуть статистическую гипотезу о том, что коэффициент равен 0:

   Так как у всех коэффициентов доверительный интервал не включает в себя 0,можно относительно них же отвергать статистическую гипотезу о том , что коэффициенты могут быть = 0.

7. Оценим доверительный интервал для одного прогноза (p = 95%, *Fertility* = 20, *Catholic* = 10, *Agriculture* = 10), используя команду predict:

```
       fit      lwr      upr
1 36.72747 30.70748 42.74746
```

Рисунок 1. Оценка доверительного интервала с помощью команды predict

Имеем доверительный интервал [30.70748 42.74746]

Полный код решения задачи приведён в Приложении 2.2.

Выводы

Интервалы всех регрессоров не включают в себя 0,это значит то,что взаимосвязь с объясняющей переменной-есть.

Интервалы небольшие,из этого следует взаимосвязь между регрессорами и объясняемой переменной *Examination*-небольшая .

Доверительный интервал со значениями *Fertility* = 20, *Catholic* = 10, *Agriculture* = 10 получился достаточно большой, модель – не хорошая.

# Задача 3

Условие

Набор данных: r12i_os26b.sav – данные исследования RLMS-HSE
Объясняемая переменная: заработная плата за 30 дней - *salary*
Регрессоры: пол, возраст, семейное положение (состоит ли в зарегистрированном браке / разведён или вдовец / никогда не состоял в браке), наличие высшего образования, место проживания, среднее число рабочих часов в неделю – *sex, age, wed1, wed2, wed3, higher_educ, city_status, working_hours.*

1. Постройте линейную регрессию зарплаты на все параметры, которые Вы выделили из данных мониторинга. Не забудьте оценить коэффициент вздутия дисперсии VIF.
2. Поэкспериментируйте с функциями вещественных параметров: используйте логарифм и степени (хотя бы от 0.1 до 2 с шагом 0.1).
3. Выделите наилучшие модели из построенных: по значимости параметров, включённых в зависимости, и по объяснённому с помощью построенных зависимостей разбросу adjusted $R^2$ – $R^2_{adj}$.
4. Сделайте вывод о том, какие индивиды получают наибольшую зарплату.
5. Оцените регрессии для подмножества индивидов: а) городские жители, не состоявшие в браке; б) разведенные женщины, без высшего образования

Решение

Представим NA в удобном виде,после того как считали данные:

- Переменная *sex* :1- мужчина, 0 – женщина
- *age* – переменная с нормализованным возрастом (формула для нормализации значения: (age - mean(age)) / sqrt(var(age)) ),где команда mean-среднее арифмитическое,а команда var-дисперсия.
- Семейное положение:
  - *wed1* = 1, если человек состоит в зарегистрированном браке, иначе 0
  - *wed2* = 1, если человек разведён или вдовец, иначе 0
  - *wed3* = 1, если человек никогда не был в браке, иначе 0
  - Проверим, что между *wed1, wed2, wed3* нет линейной зависимости
- *higher_edu*c = 1, если у человека есть высшее образование, иначе 0(остальные 5 значений)
- *city_status* = 1, если человек живёт в городе, иначе 0
- *working_hours* – переменная с нормализованным числом рабочих часов в неделю (формула для нормализации значения: (working_hours - mean(working_hours)) / sqrt(var(working_hours)) )
- *salary* – переменная с нормализованной зарплатой (формула для нормализации значения: ((salary - mean(salary)) / sqrt(var(salary)) )

1. Построим линейную регрессию зарплаты на все параметры, оценим vif:

Модель строим командой model1 = lm(data = data2, salary ~ sex + age + wed1 + wed2 + wed3 + higher_educ + city_status + working_hours)

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   -0.59335    0.06029  -9.842  < 2e-16 ***
sex            0.44784    0.03486  12.848  < 2e-16 ***
age           -0.06912    0.01837  -3.762 0.000172 ***
wed1          -0.05903    0.05590  -1.056 0.291021
wed2          -0.04722    0.06885  -0.686 0.492874
wed3          -0.23496    0.06990  -3.361 0.000785 ***
higher_educ    0.47544    0.03876  12.268  < 2e-16 ***
city_status    0.48293    0.03742  12.906  < 2e-16 ***
working_hours  0.15475    0.01697   9.117  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9097 on 3048 degrees of freedom
Multiple R-squared:  0.1747,    Adjusted R-squared:  0.1725
F-statistic: 80.65 on 8 and 3048 DF,  p-value: < 2.2e-16
```

Рисунок 1. Характеристики model1, где model1 = lm(data = data2, salary ~ sex + age + wed1 + wed2 + wed3 + higher_educ + city_status + working_hours)

Из рисунка 1 видим, что переменные *wed1* и *wed2* имеют плохую p-статистику. Уберём их и посмотрим, как изменится $R^2$:

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   -0.64210    0.03642 -17.631  < 2e-16 ***
sex            0.44678    0.03404  13.127  < 2e-16 ***
age           -0.07107    0.01813  -3.921 9.03e-05 ***
wed3          -0.18721    0.05266  -3.555 0.000384 ***
higher_educ    0.47232    0.03864  12.225  < 2e-16 ***
city_status    0.48333    0.03739  12.928  < 2e-16 ***
working_hours  0.15547    0.01695   9.175  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9095 on 3050 degrees of freedom
Multiple R-squared:  0.1744,    Adjusted R-squared:  0.1728
F-statistic: 107.4 on 6 and 3050 DF,  p-value: < 2.2e-16
```

Рисунок 2. Результат работы команды summary(model1), где model1 = lm(data = data2, salary ~ sex + age + wed3 + higher_educ + city_status + working_hours)

Из рисунка 2 видим, что $R^2$ изменился незначительно, зато p-статистика теперь хорошая для всех регрессоров. В дальнейшем будем работать с этой моделью.

Оценим vif у модели 1:

```
> vif(model1)#зависимость между регрессорами-отсутствует
       sex          age         wed3   higher_educ   city_status working_hours
  1.054764     1.214019     1.202465     1.031371     1.015514     1.060804
```

Рисунок 3. Результат работы команды vif(model1)

Из рисунка 3 видим, что vif низкий – линейной зависимости между регрессорами нет.

2. Введём в модель логарифмы и степени.

Логарифмы и степени имеет смысл вводить только для параметров *age* и *working_hours*, так как

остальные принимают только значения 0 или 1.
Модель с логарифмами: *model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ + city_status + I(log(working_hours)) + I(log(age)))* –у модели достаточно хороший vif, рассмортим остальные(model2,model3):
Поиск наилучшей модели
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ + city_status + I(log(working_hours)) + I(log(age)))
model2 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ + city_status + I(log(age)))
model3 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ + city_status + I(log(working_hours)))

Из них лучший $R^2$ имеет первая модель, но у неё плохая p-статистика для обоих логарифмов. У модели 3 $R^2$ немного ниже чем у первой и второй модели, p-статистика – неплохая,*I(log(working_hours))*-имеет плохую p-статистику.У модели 2 $R^2$-между model1 и model3(достаточно рядом),p-статистика хорошая,кроме *wed3* и *I(log(age))*,у этих переменных нет звёзд.
Наилучшей моделью будем считать: model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ + city_status + I(log(working_hours)) + I(log(age)))
Построим модели со степенями в которых степень будет задаваться переменной *power*, меняющий значение от 0.1 до 2 с шагом 0.1:
power = 0.1
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ + city_status + I(working_hours^power) + I(age^power))
Модель имеет $R^2$~0.2155 и плохую p-статистику у переменных со словами *age* и *working_hours*.
Сравнивая остальные модели с отличием в степени,можно заметить что $R^2$-понижается,не считая power=2.

3. Выделим наилучшие модели из построенных: по значимости параметров, включённых в зависимости, и по объяснённому с помощью построенных зависимостей разбросу adjusted $R^2$ – $R^2$_adj

Наилучшими по значению $R^2$ из всех моделей без линейной зависимости регрессоров являются модели для степеней 0.1, 0.2, 2,0. Разброс $R^2$ - $R^2$_adj одинаковый у power=0.1 и power=0.2.Разброс у power=2 меньше чем у моделей с меньшей степенью. Лучшей моделью будет считаться модель при power=0.1,так как у неё наибольший $R^2$,даже учитывая,что p-статистиканемного хуже чем у модели при power=2. Из этих трёх моделей лучшей является модель *model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ + city_status + I(working_hours^power) + I(age^power))* для power = 0.1, которая имеет наивысший $R^2$ = 0.2155.

4. Согласно наилучшей модели больше всего зарабатывают молодые мужчины с высшим образованием, проживающие в городах, работающие большое число часов в неделю.
5. Оценим регрессии для подмножества индивидов: а) Не вступавшие в брак, без высшего образования;б) Городские жители, состоящие в браке

а) Не вступавшие в брак, без высшего образования:
data3 = subset(data2, higher_educ == 0)
data3 = subset(data3, wed3 == 1)

Тогда имеем следующую модель:

```
Coefficients:
                       Estimate Std. Error t value Pr(>|t|)
(Intercept)             2.41999    1.37507   1.760 0.079042 .
sex                     0.44772    0.09777   4.579 5.91e-06 ***
working_hours           0.32291    0.09485   3.404 0.000717 ***
age                    -0.16038    0.19618  -0.818 0.414008
city_status             0.67296    0.10257   6.561 1.35e-10 ***
I(working_hours^power) -1.98224    1.08348  -1.830 0.067924 .
I(age^power)           -1.42960    1.11297  -1.284 0.199574
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.069 on 495 degrees of freedom
  (2555 observations deleted due to missingness)
Multiple R-squared:  0.1688,    Adjusted R-squared:  0.1588
F-statistic: 16.76 on 6 and 495 DF,  p-value: < 2.2e-16
```

Рисунок 4. Результат работы команды summary(model1), где model1 = lm(data = data2, salary ~ sex + working_hours + age + city_status + I(working_hours^power) + I(age^power))

$R^2$~0.1688.Параметры *sex,working_hours* и *city_status* имеют достаточно хорошую p-статистику.Согласно модели: больше всего зарабатывают молодые(ненадёжная p-статистика) мужчины,работающие много,проживающие в городе.

б) Городские жители, состоящие в браке:

data3 = subset(data2, city_status  == 1)

data3 = subset(data3, wed2 == 1)

Тогда имеем следующую модель:

```
Coefficients:
                       Estimate Std. Error t value Pr(>|t|)
(Intercept)             4.01213    1.35489   2.961 0.003211 **
sex                     0.38276    0.09828   3.895 0.000112 ***
working_hours           0.39119    0.09485   4.124 4.36e-05 ***
age                    -0.07877    0.19648  -0.401 0.688671
higher_educ             0.71901    0.11761   6.113 1.98e-09 ***
I(working_hours^power) -2.85850    1.07902  -2.649 0.008327 **
I(age^power)           -2.02103    1.11521  -1.812 0.070556 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.074 on 495 degrees of freedom
  (2555 observations deleted due to missingness)
Multiple R-squared:   0.16,     Adjusted R-squared:  0.1498
F-statistic: 15.71 on 6 and 495 DF,  p-value: < 2.2e-16
```

Рисунок 5. Результат работы команды summary(*model1*), где *model1 = lm(data = data2, salary ~ sex + working_hours + age + higher_educ + I(working_hours^power) + I(age^power))*

Все параметры, кроме *age*, значимые, $R^2$ ~ 0.16

Согласно этой модели наибольшая зарплата у мужчин с высшим образованием молодого(ненадёжная p-статистика) возраста, работающих много.

Полный код решения задачи приведён в Приложении 3.

## Выводы

Из всей выборки больше всего зарабатывают молодые мужчины с высшим образованием, проживающие в городах, работающие большое число часов в неделю.
Среди людей,которые не вступали в браки;не имеющих высшего образования, больше всего зарабатывают молодые мужчины,работающие много,проживающие в городе.
Среди городских жителей, состоящих в браке наибольшая зарплата у мужчин с высшим образованием молодого возраста, работающих много.

# Задача 4

Условие

Набор данных: StudentsPerformance – данные исследований с сайта https://www.kaggle.com/spscientist/students-performance-in-exams

Регрессоры: пол , этническая принадлежность , уровень образования , баллы по математике , баллы по письму , баллы по чтению , обучение,курс подготовки к тестированию – *gender , race/ethnicity , parental level of education , math score , writing score , reading score , lunch , test preparation course.*

1 Обработайте набор данных 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?

Решение

1. Обработаем данные:
   - Этническая принадлежность:
     - *race/ethnicity* = 0, если человек относится к group A.
     - *race/ethnicity* = 1, если человек относится к group B.
     - *race/ethnicity* = 2, если человек относится к group C.
     - *race/ethnicity* = 3, если человек относится к group D.
     - *race/ethnicity* = 4, если человек относится к group E.
   - Уровень образования:
     - *parental level of education*= 0, если человек закончил some college.
     - *parental level of education*= 1, если человек закончил some high school.
     - *parental level of education*= 2, если человек закончил high school.
     - *parental level of education*= 3, если человек имеет bachelor's degree.
     - *parental level of education*= 4, если человек имеет associate's degree.
     - *parental level of education*= 5, если человек имеет master's degree.
   - Обучение-0 если обучение бюджетное/сокращённое,1 если обучение-стандартное.
   - Курс подготовки к тестированию-0 если не пройден,1 если закончен.

Выделим целевой признак , и удалим его из данных:

```
data_sel = data.loc[:, data.columns.isin(['gender', 'race/ethnicity', 'parental level of education',
                                          'lunch', 'test preparation course', 'math score', 'reading score','writing score'])]
data_sel = data_sel.dropna()
data_sel['writing score'] = np.where(data_sel['writing score'] > np.average(data_sel['writing score']) , 0, 1)
writing_score = data_sel.loc[:, data_sel.columns.isin(['writing score'])]
X = data_sel.loc[:, data_sel.columns.isin(['gender', 'race/ethnicity', 'parental level of education', 'lunch',
                                          'test preparation course', 'math score', 'reading score'])]
```

Рисунок 1. Столбец writing score-отделяется от data_sel ,X-таблица ,в которой отсутствует writing score.

Таблица с данными:

| | race/ethnicity | parental level of education | lunch | test preparation course | math score | reading score |
|---|---|---|---|---|---|---|
| **gender** | | | | | | |
| **female** | 1 | 3 | 1 | 0 | 72 | 72 |
| **female** | 2 | 0 | 1 | 1 | 69 | 90 |
| **female** | 1 | 5 | 1 | 0 | 90 | 95 |
| **male** | 0 | 4 | 0 | 0 | 47 | 57 |
| **male** | 2 | 0 | 1 | 0 | 76 | 78 |
| **...** | ... | ... | ... | ... | ... | ... |
| **female** | 4 | 5 | 1 | 1 | 88 | 99 |
| **male** | 2 | 2 | 0 | 0 | 62 | 55 |
| **female** | 2 | 2 | 0 | 1 | 59 | 71 |
| **female** | 3 | 0 | 1 | 1 | 68 | 78 |
| **female** | 3 | 0 | 0 | 0 | 77 | 86 |

1000 rows × 6 columns

Рисунок 2. Результат работы кода на рисунке 1.

Построим классификатор типа :метод опорных векторов:

```
GridSearchCV(estimator=SVC(),
             param_grid={'C': (0.25, 0.5, 0.75, 1),
                         'decision_function_shape': ('ovo', 'ovr'),
                         'gamma': (1, 2, 3, 'auto'),
                         'kernel': ('linear', 'rbf'),
                         'shrinking': (True, False)})
```

Рисунок 3. Создаётся классификатор опорных векторов с тестовой выборкой.

Оценим точность построенного классификатора с помощью метрик precision,recall и F1:

```
f1:0.9144850613243941
precision:0.920411613960001
recall:0.9142857142857143
```

Рисунок 4. Показатели метрик достаточно большие(наибольшая у метрики-precision).

2. Построим классификатор типа Случайный Лес(Random Forest) для решения той же задачи:

```
GridSearchCV(cv=5, estimator=RandomForestClassifier(),
             param_grid={'criterion': ['gini'],
                         'max_depth': [1, 2, 3, 4, 5, 6, 7, 8, 9],
                         'max_features': ['auto'],
                         'n_estimators': [50, 100, 150]})
```

Рисунок 5. . Создаётся классификатор Случайного Леса с тестовой выборкой.

. Оценим точность построенного классификатора с помощью метрик precision,recall и F1:

```
f1:0.9189274665824458
precision:0.9351134277471808
recall:0.9142857142857143
```

Рисунок 6. Показатели метрик также как и у опорных векторов- достаточно большие(наибольшая у метрики-precision).

Таким образом,сравнивая 3 и 5 рисунки-видно,что в классификаторе Случайный лес показатели метрик чуть больше,чем в классификаторе опорных векторов,из этого следует, что классификатор Случайный Лес -лучше.

Код решения задачи и сведения о проверенных моделях приведены в Приложении к задаче 4.

<u>Выводы</u>

Метрики F1,precision и recall-выдают у обоих классификаторов высокие показатели.
В SVM объекты разделяются на класс с помощью гиперплоскости.
В этой задаче для SVM важны такие параметры как C-доп.ограничения(штрафы),kernel-тип ядра(в данном случае используются линейный и радиальный),decision_function_shape- форма функции принятия решений(в данном случае ovo-“one vs one” и ovr-“one vs rest”),gamma(коэффициент ядра для rbf),shrinking-сжатие.
В RFC объекты разделяются на класс с помощью множества решающих деревьев.
В этой задаче для RFC важны такие параметры как criterion-функция измерения качества раскола(gini-мера,показывающая насколько часто элемент неверно помечается),max_depth-максимальная глубина деревьев,max_features-количество функций,которые следует учитывать при поиске лучшего разделения,n_estimators-кол-во деревьев.
Наибольшие показатели у классификаторов выдаёт precision(точность)
Случайный Лес – лучше делит данные на классы,чем Метод Опорных Векторов.

## ЗАДАЧА 5

Предобработка данных и PCA

В данной задаче мне необходимо провести анализ датасета (в моём случае набор данных Credit Card Fraud Detection) с помощью языка Python. Также взадаче требуется ответить на вопросы:

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

2. Сколько категориальных признаков, какие?

3. Столбец с максимальным количеством уникальных значений категориальногопризнака?

4. Есть ли бинарные признаки?

5. Какие числовые признаки?

6. Есть ли пропуски?

7. Сколько объектов с пропусками?

8. Столбец с максимальным количеством пропусков?

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

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

11. Столбец с целевым признаком?

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

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

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

15. Какой признак вносит наибольший вклад в первую компоненту?Решение:

Описание переменных и набора данных.

Рассмотрим данный датасет. С помощью функции data.shape смотрим размертаблицы. Она состоит из 284807 строк (объектов) и 31 столбцов (признаков).

Описание столбцов:

Time- Количество секунд, прошедших между этой транзакцией и первойтранзакцией в наборе данных

V1-V28- может быть результатом уменьшения размерности PCA для защитыпользовательских идентификаторов и чувствительных функций

Amount-сумма сделки

Class- 1 для мошеннических операций, 0 в противном случае

В данном случае категориальных признаков - нет, поэтому обрабатывать их не нужно.

Однако в наборе данных имеется 1 бинарный признак (Class) и 30 числовых признаков. Также в данном наборе отсутствуют пропуски. В данном примере нет аномальных значений.

После нормировки признаков через стандартное отклонение в столбце ‘Class’ можно увидеть максимальное среднее значение. Также столбец ‘Class’ является целевым признаком. После выделения тренировочной и тестовой выборки мы получаем, что в тренировочную выборку попадает 199364 объектов, а в тестовую – 85443.

Посмотрев на рисунок 20, можно увидеть, что линейная зависимость(корреляция) не наблюдается.

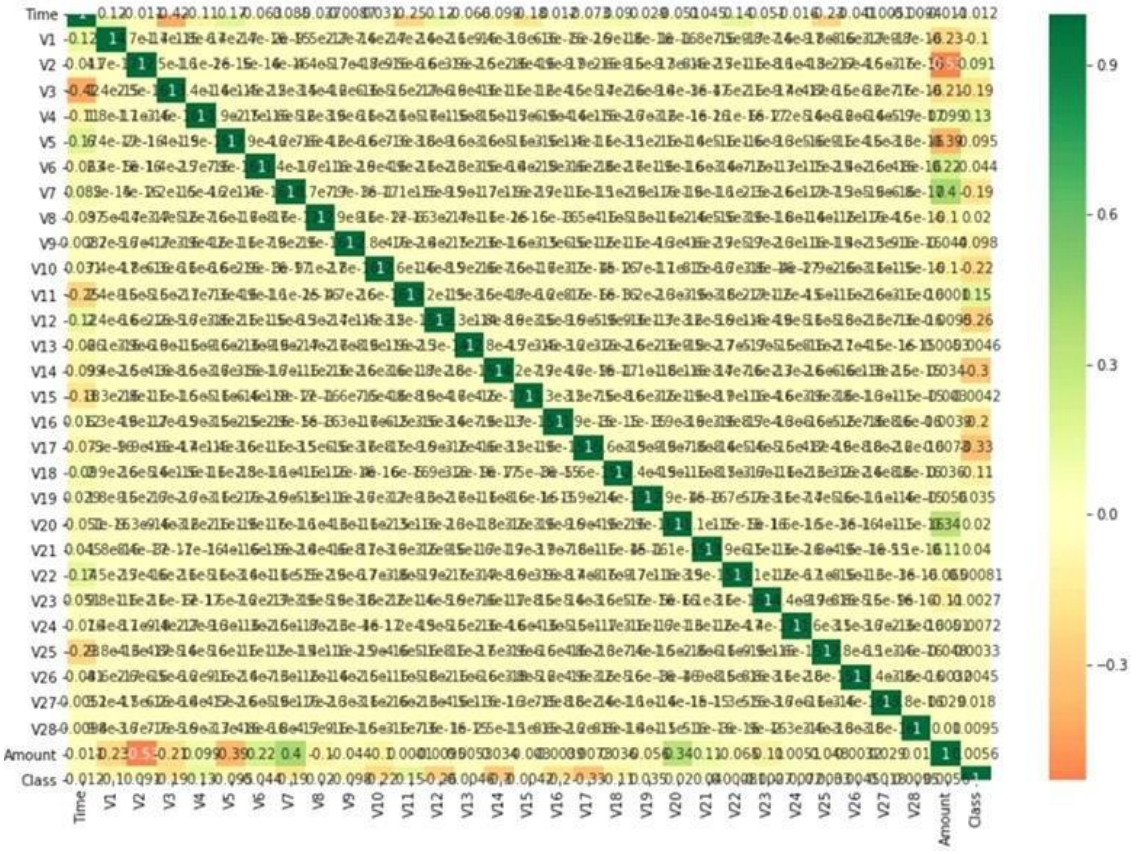



Рисунок 20. Результат визуального анализа данных.

Наибольший вклад в первую компоненту вносит признак ‘*Amount*’.

Применив метод PCA для уменьшения количества описывающих компонент (Рисунок 21), я узнал, что для описания 90% дисперсии данных достаточно 3 компонентов.

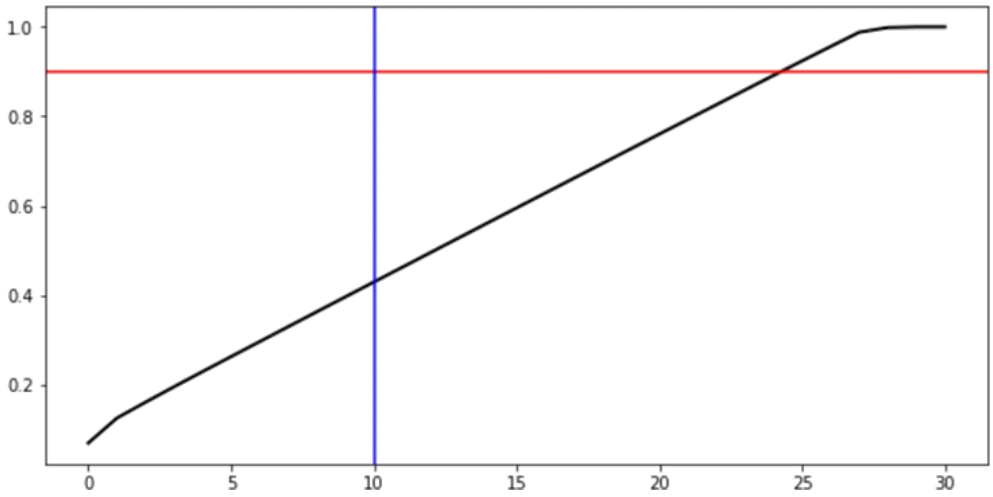



Рисунок 21. Результат применения метода PCA.

Вывод

Первично обработав данные, я подготовил их к применению алгоритмов классификации и регрессии. После, с помощью метода главных компонент (метод PCA) я узнал, что для описания целевого признака target достаточно всего 3 переменных.

Код решения задачи и сведения о проверенных моделях приведены в Приложении к задаче 5.

## ЗАКЛЮЧЕНИЕ

В ходе научно-исследовательской работы были выполнены 5 задач. В результате их выполнения я освоил основные принципы работы с наборами данных на языке R и Python. Также при выполнении заданий я познакомился с методами построения модели PCA.

В задаче 1 было проведено исследование данных по кантонам в Швейцарии в конце 19 века. В результате я выявил зависимость процента образования (объясняемой переменной) от различных факторов (регрессоров).

В задаче 2.1 и задаче 2.2 было проведено исследование данных по кантонам в Швейцарии в конце 19 века. В результате было выявлено,что существует нелинейная зависимость между объясняемой переменной (Examination) и различными регрессорами,а также с помощью доверительных интервалов было доказано что взаимосвязь с объясняемой переменной существует,но при этом,небольшая.

В задаче 3 я описал, как определённые параметры влияют на заработную плату различных слоёв населения, основываясь на данных российского мониторинга экономического положения и здоровья населения НИУ-ВШЭ в 2004 году.Можно судить, что молодые мужчины с высшим образованием, которые проживают и много работают в городах получают больше остальных.

В задаче 4 был проведен анадиз данных с исследований, взятых с сайта https://www.kaggle.com/spscientist/students-performance-in-exams В результате, по предварительно отсортированным данным(значениям-строчкам были выданы числа,а столбец *writing score* не учитывался,при обучении классификаторов) были обучены два классификатора:SVM(Метод опорных векторов) и RFC(Случайный лес).В ходе анализиа ыло выявлено,что RFC лучше делит на классы,чем SVM.

В задаче 5 я провел первичный анализ и предобработку данных предложенного датасета "Credit Card Fraud Detection" (Обнаружение мошенничества с кредитными картами) с помощью языка программирования Python. Проведя описание переменных и набора данных в целом, сделав его подготовку, выполнив визуальный анализ, а также применив метод главных компонент (метод PCA), были выявлены необходимые условия для описания целевого признака.

## СПИСОК ЛИТЕРАТУРЫ

1. Ершов Э.Б. Распространение коэффициента детерминации на общий случай линейной регрессии, оцениваемой с помощью различных версий метода наименьших квадратов (рус., англ.)/ЦЭМИ РАН Экономика и математические методы. — Москва: ЦЭМИ РАН, 2002.— Т. 38, вып. 3. — С. 107-120.

2. Демиденко Е.З. Линейная и нелинейная регрессия/М.: Финансы и статистика,1981. — 302 с.

3. Шведов.А.С. Теория вероятностей и математическая статистика: промежуточный уровень [Текст]: учеб.пособие/ А.С.Шведов ; Нац. исслед. ун-т «Высшая школа экономики». – М. : Изд. Дом Высшей школы экономики, 2016. – (Учебник Высшей школы экономики). – 280 с. – 600 экз. –ISBN 978-5-7598-1301-9(в пер.)

4. Николенко С.И., Тулупьев А.Л. Н63 Самообучающиеся системы. –М.: МЦНМО, 2009. – 288 с.: 24 илл.

5. Магнус Я.Р. Эконометрика. Начальный курс /Катышев П.К., Пересецкий А.А. —М.: Дело, 2004.—6-е изд., перераб. и доп. - 576 с.

## ПРИЛОЖЕНИЯ

Приложение к задаче 1

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

data = swiss
#выводим данные
data

#Пункт 1. Оцените среднее значение, дисперсию и СКО переменных, указанных во втором и третьем столбце.

#среднее значение:
print(paste(mean(data$Education)))
print(paste(mean(data$Fertility)))
print(paste(mean(data$Examination)))

#дисперсия
print(paste(var(data$Education)))
print(paste(var(data$Fertility)))
print(paste(var(data$Examination)))

#СКО
print(paste(sd(data$Education)))
print(paste(sd(data$Fertility)))
print(paste(sd(data$Examination)))

#Пункт 2. Постройте зависимости вида y = a + bx, где y - объясняемая переменная, x - регрессор.

model1 = lm(Education~Fertility, data)
model2 = lm(Education~Examination, data)

model1 # y = 46.8179 - 0.5109x
model2 # y = -2.9015 + 0.8418x

# Пункт 3. Оцените, насколько «хороша» модель по коэффициенту детерминации R^2
summary(model1) #R^2 = 0.4406 -R^2 неплохой, модель относительно хороша.
summary(model2) #R^2 = 0.4878 -R^2 неплохой, модель относительно хороша.

#Пункт 4. Оцените, есть ли взаимосвязь между объясняемой переменной и объясняющей переменной

# У model1 параметры имеют по 3 звезды ,из этого следует что взаимосвязь -очень сильная
# В model2 у первого параметра нет звёзд ,у второго-3.Коэфициент зависимости первой переменной - низкий , второй - высок .


#Вывод"
#В model1 взаимосвязи между переменными -есть, но зависимость -нелинейная ,нужны дополнительные регрессоры/регрессор.
#В model2 связь со второй переменной -сильная, с первой -слабая ,зависимость - нелинейная ,нужны дополнительные регрессоры/регрессор.
```

Приложение к задаче 2.1

```
library("lmtest")
library("GGally")
library("car")  # без этого не работает функция vif()

# При чтении избавляемся от записей с недостающими данными.
data = na.omit(swiss)

# Выводим данные
data

# Examination ~ Fertility, Catholic, Agriculture

# 1. Проверим отсутствие зависимости между регрессорами перед построением модели
linfunc_1 = lm(Fertility~Catholic, data)
summary(linfunc_1) # R^2 < 22% - зависимости нет

linfunc_1 = lm(Fertility~Agriculture, data)
summary(linfunc_1) # R^2 < 13% - зависимости нет

linfunc_1 = lm(Catholic~Agriculture, data)
summary(linfunc_1) # R^2 < 17% - зависимости нет

# Можно использовать регрессоры вместе



# 2. Построим линейную модель и оценим её
model = lm(Examination ~ Fertility + Catholic + Agriculture, data)
summary(model)
# R^2 ~ 0.69, p-значение у Catholic ненадёжно (одна звездочка) - модель достаточно
хороша(остальные p-значения имеют по 3 звезды)

#Уберём из модели регрессор Catholic, как наименее значимый, и проверим, как
изменится R^2
model = lm(Examination ~ Fertility + Agriculture, data)
summary(model)
# R^2 ~ 0.66 - R^2 практически не изменился(у всех параметров по 3 звезды)

# Попробуем убрать ещё один регрессор
model = lm(Examination ~ Fertility, data)
summary(model) #R^2 ~ 0.42 - изменился сильно, регрессор Agriculture лучше не убирать

# Попробуем убрать другой регрессор
model = lm(Examination ~ Agriculture, data)
summary(model) #R^2 ~ 0.47 - изменился сильно, регрессор Fertility лучше не убирать


# В дальнейшем будем работать с моделью:
model = lm(Examination ~ Fertility + Agriculture + Catholic , data) #R^2 ~ 0.69


# 3. Попробуем ввести в модель логарифмы регрессоров, предварительно проверяя, что
нет линейной зависимости
```

```
model = lm(Examination ~ I(log(Fertility)) + I(log(Agriculture)) + I(log(Catholic)) ,
data)
vif(model) # линейной зависимости нет.
summary(model) #R^2 ~0.67

model = lm(I(log(Examination)) ~ I(log(Fertility)) + I(log(Agriculture)) +
I(log(Catholic)) , data)
vif(model) # линейной зависимости нет.
summary(model) # R^2 ~ 0.54, p-статистика неплоха,при I(log(Examination))- R-заметно
снижается

model = lm(Examination ~ Fertility + Agriculture + I(log(Catholic)) , data)
vif(model) # линейной зависимости нет.
summary(model) #R^2 ~0.68

model = lm(Examination ~ I(log(Fertility)) + I(log(Agriculture)) + Catholic , data)
vif(model) # линейной зависимости нет.
summary(model) # R^2 ~ 0.69

model = lm(Examination ~ Fertility + I(log(Agriculture)) + Catholic , data)
vif(model) # линейной зависимости нет.
summary(model) # R^2 ~ 0.70, p-статистика достаточно хороша

model = lm(Examination ~ Fertility + I(log(Agriculture)) + I(log(Catholic)) , data)
vif(model) # линейной зависимости нет.
summary(model) # R^2 ~ 0.68

model = lm(Examination ~ I(log(Fertility)) + Agriculture + I(log(Catholic)) , data)
vif(model) # линейной зависимости нет.
summary(model) #R^2 ~0.68

model = lm(Examination ~ I(log(Fertility)) + Agriculture + Catholic , data)
vif(model) # линейной зависимости нет.
summary(model) #R^2 ~0.69, p-статистика плоха для Catholic

# Наилучшей из них будет следующая модель:
model = lm(Examination ~ Fertility + I(log(Agriculture)) + Catholic , data)  # R^2 ~
0.70


# 4. Попробуем ввести в модель всевозможные произведения пар регрессоров,
предварительно проверяя, что нет линейной зависимости

model = lm(Examination ~ Fertility + I(log(Agriculture)) + Catholic + I(Fertility^2)
+ I(Agriculture^2) + I(Fertility*Agriculture) + I(Fertility*Catholic) +
I(Catholic*Agriculture) + I(Catholic^2), data)
vif(model) # есть линейная зависимость, уберём регрессоры с максимальным VIF

model = lm(Examination ~ Fertility + I(log(Agriculture)) + Catholic +
I(Agriculture^2) + I(Fertility*Agriculture) + I(Fertility*Catholic) +
I(Catholic*Agriculture) + I(Catholic^2), data)
vif(model) # есть линейная зависимость, уберём регрессоры с максимальным VIF

model = lm(Examination ~ Fertility + I(log(Agriculture)) + Catholic +
I(Agriculture^2) + I(Fertility*Agriculture) + I(Catholic*Agriculture) +
I(Catholic^2), data)
vif(model) # есть линейная зависимость, уберём регрессоры с максимальным VIF

model = lm(Examination ~ Fertility + I(log(Agriculture)) + Catholic +
I(Agriculture^2) + I(Fertility*Agriculture) + I(Catholic*Agriculture), data)
vif(model) # есть линейная зависимость, уберём регрессоры с максимальным VIF
```

```r
model = lm(Examination ~ Fertility + I(log(Agriculture)) + Catholic +
I(Agriculture^2) + I(Catholic*Agriculture), data)
vif(model) # есть линейная зависимость, уберём регрессоры с максимальным VIF

model = lm(Examination ~ Fertility + I(log(Agriculture)) + Catholic +
I(Agriculture^2), data)
vif(model)#хорошие показатели,имеет смысл посмотреть на R^2
summary(model) #R^2~0.71, p-статистика не плохая

model = lm(Examination ~ Fertility + Catholic + I(Agriculture^2), data)
vif(model)
summary(model) # R^2 ~ 0.65, p-статистика крайне плоха для Catholic

model = lm(Examination ~ Fertility + I(Agriculture^2), data)
vif(model)
summary(model) # R^2~ 0.63

#Наилучшая модель:
model = lm(Examination ~ Fertility + I(log(Agriculture)) + Catholic , data)  # R^2 ~
0.70
```

## Приложение к задаче 2.2

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

data = na.omit(swiss)

# модель из практического задания 2:
model = lm(Examination ~ Fertility + Catholic + Agriculture, data)
summary(model)

# 1. Оценим доверительные интервалы для всех коэффициентов в модели (для p=95%)

#коэф. свободы = 43,всего параметров 4==> df = 43-4 = 39
t_critical = qt(0.975, df = 39)

# Стандартные ошибки коэффициентов (взяты из summary(model)):
Std_Error_Intercept = 4.10586
Std_Error_Fertility = 0.06274
Std_Error_Catholic = 0.01919
Std_Error_Agriculture = 0.03337

Estimate_Std._Intercept=43.68234
Estimate_Std._Fertility=-0.24582
Estimate_Std._Catholic=-0.03953
Estimate_Std._Agriculture=-0.16431

# Выведем соответствующие доверительные интервалы:[x-at,x+at]
#x-значение таблицы Estimate(коэф.),a-значение таблицы Std Error в строке элемента
x,t(одинаков для всех переменных)-критерий Стьюдента.

print(paste("Доверительный интервал Intercept: [", Estimate_Std._Intercept -
t_critical * Std_Error_Intercept,
            ",", Estimate_Std._Intercept + t_critical * Std_Error_Intercept, "]"))

print(paste("Доверительный интервал Fertility: [", Estimate_Std._Fertility -
t_critical * Std_Error_Fertility,
            ",", Estimate_Std._Fertility + t_critical * Std_Error_Fertility, "]"))
```

```r
print(paste("Доверительный интервал Catholic: [", Estimate_Std._Catholic - t_critical
* Std_Error_Catholic,
            ",", Estimate_Std._Catholic + t_critical * Std_Error_Catholic, "]"))

print(paste("Доверительный интервал Agriculture: [", Estimate_Std._Agriculture -
t_critical * Std_Error_Agriculture,
            ",", Estimate_Std._Agriculture + t_critical * Std_Error_Agriculture,
"]"))


# 2. Вывод о отвержении или невозможности отвергнуть статистическую гипотезу о том,
что коэффициент равен 0:

# Доверительный интервал свободного коэффициента: [ 35.3774542590579 ,
51.9872257409421 ]
# Доверительный интервал Fertility: [ -0.372723628323106 , -0.118916371676894 ]
# Доверительный интервал Catholic: [ -0.0783454387555054 , -0.000714561244494573 ]
# Доверительный интервал Agriculture: [ -0.231807196001627 , -0.0968128039983733 ]

# все интервалы не соприкасаются с 0->отвергаем статистическую гипотезу о том что
коэффициент может быть = 0


# 3. Доверительный интервал для одного прогноза (p = 95%, Fertility = 20, Catholic =
10, Agriculture = 10).

new.data = data.frame(Fertility = 20, Catholic = 10, Agriculture = 10)
predict(model, new.data, interval = "confidence")

# Доверительный интервал: [30.70748, 42.74746]
```

## Приложение к задаче 3

```r
# #install.packages("devtools")
#devtools::install_github("bdemeshev/rlms")

library("lmtest")
library("rlms")
library("dplyr")
library("GGally")
library("car")
library("sandwich")

"
hh5 Пол респондента
  1 МУЖСКОЙ
  2 ЖЕНСКИЙ

h_marst СЕМЕЙНОЕ ПОЛОЖЕНИЕ
 1 Никогда в браке не состояли
 2 Состоите в зарегистрированном браке
 3 Живете вместе, но не зарегистрированы
 4 Разведены и в браке не состоите
 5 Вдовец (вдова)
 6 ОФИЦИАЛЬНО ЗАРЕГИСТРИРОВАНЫ, НО ВМЕСТЕ НЕ ПРОЖИВАЮТ
h_diplom ЗАКОНЧЕННОЕ ОБРАЗОВАНИЕ (ГРУППА)
 1 окончил 0 - 6 классов
 2 незаконч среднее образование (7 - 8 кл)
 3 незаконч среднее образование (7 - 8 кл) + что-то еще
 4 законч среднее образование
 5 законч среднее специальное образование
```

```
 6 законч высшее образование и выше
status ТИП НАСЕЛЕННОГО ПУНКТА
 1 областной центр
 2 город
 3 ПГТ
 4 село
"
data <- rlms_read("C:\\Users\\Admin\\Documents\\R\\r12i_os26b.sav")


data = select(data, hh5, h_age, h_marst, h_diplom, status, hj13.2, hj6.2)
data = na.omit(data)
glimpse(data)

data2 = select(data,) #Новая база данных для нормализованных значений

#Возраст
age = data$h_age
data2["age"] = (age - mean(age)) / sqrt(var(age))
glimpse(data2["age"])

#Пол
data2["sex"] = 0
data2$sex[which(data$hh5 == 1)] <- 1
glimpse(data2["sex"])

#Семейное положение:

#Никогда не состоял/ла в браке?
data2$wed3 = 0
data2$wed3[which(data$h_marst==1)] <- 1
glimpse(data2["wed3"])

#Состоит ли в зарегестрированном браке?
data2$wed1 = 0
data2$wed1[which(data$h_marst==2)] <- 1
data2$wed1[which(data$h_marst==6)] <- 1
glimpse(data2["wed1"])

#Разведён или вдовец?
data2$wed2 = 0
data2$wed2[which(data$h_marst==4)] <- 1
data2$wed2[which(data$h_marst==5)] <- 1
glimpse(data2["wed2"])

# Проверка на отсутствие зависимости
vif(lm(data$hj13.2 ~ data2$wed1 + data2$wed2 + data2$wed3))

#Наличие высшего образования
data2$higher_educ = 0
data2$higher_educ[which(data$h_diplom==6)] <- 1
glimpse(data2["higher_educ"])

#Живёт в городе?
data2$city_status = 0
data2$city_status[which(data$status==1)] <- 1
data2$city_status[which(data$status==2)] <- 1
glimpse(data2["city_status"])

#Нормализованное среднее число рабочих часов в неделю
working_hours = data$hj6.2
data2$working_hours = (working_hours - mean(working_hours)) /
```

```
sqrt(var(working_hours))
glimpse(data2["working_hours"])

#Нормализованная средняя зарплата
salary = data$hj13.2
data2$salary = (salary - mean(salary)) / sqrt(var(salary))
glimpse(data2["salary"])


# 1. Постройте линейную регрессию зарплаты на все параметры, которые Вы выделили из
данных мониторинга. Не забудьте оценить коэффициент вздутия дисперсии VIF.

model1 = lm(data = data2, salary ~ sex + age + wed1 + wed2 + wed3 + higher_educ +
city_status + working_hours)
vif(model1)#зависимость между регрессорами-отсутствует
summary(model1)#R^2~0.1747,wed1 и wed2- не имеют звёзд(плохая p-статистика)

model1 = lm(data = data2, salary ~ sex + age + wed3 + higher_educ + city_status +
working_hours)
vif(model1)#зависимость между регрессорами-отсутствует
summary(model1)#p-статистика--отличная,R^2~0.1744(зависимость-нелинейная)

# 2. Поэкспериментируйте с функциями вещественных параметров: используйте логарифм и
степени (хотя бы от 0.1 до 2 с шагом 0.1).

#sex,wed3,higher_educ,city_status-имеют значения только 0 и 1->не имеет смысла
использовать с ними логарифмирования и возведение в степень

# с логарифмами:
model1 = lm(data = data2, salary ~ sex + working_hours + age +  wed3 + higher_educ +
city_status + I(log(working_hours)) + I(log(age)))
vif(model1)#vif<5 у всех регрессоров,age,wed3 и оба логарифма имеют плохую p-
статистику
summary(model1)#R^2~0.2164(зависимость нелинейная)

model2 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(log(age)))
vif(model2)#зависимость между регрессорами-отсутствует
summary(model2)#R^2~0.1958 p-статистика плохая у wed3 и I(log(age))

model3 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status +  I(log(working_hours)))
vif(model3)#зависимость между регрессорами-отсутствует
summary(model3)#R^2~0.1916 p-статистика плохая у I(log(working_hours))

#со степенями
power = 0.1
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)
summary(model1)#R^2~0.2155,плохая p-статистика у переменных со словами age и
working_hours

power = 0.2
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1) #есть переменные у которых vif>5
summary(model1)#R^2~0.2146

power = 0.3
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
```

```
vif(model1) #есть переменные у которых vif>5
summary(model1)#R^2~0.2138

power = 0.4
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)#есть переменные у которых vif>10
summary(model1)#R^2~0.2129

power = 0.5
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)#есть переменные у которых vif>15
summary(model1)#R^2~0.2122

power = 0.6
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)#есть переменные у которых vif>25
summary(model1)#R^2~0.2114

power = 0.7
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)#есть переменные у которых vif>45
summary(model1)#R^2~0.2108


power = 0.8
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)#есть переменные у которых vif>100
summary(model1)#R^2~0.2103

power = 0.9
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)#есть переменные у которых vif>480
summary(model1)#R^2~0.2098

power = 1.1
model1 = lm(data = data2, salary ~ sex + working_hours + wed3 + higher_educ +
city_status + I(age^power))
vif(model1)#есть переменные у которых vif>520
summary(model1)#R^2~0.209

power = 1.2
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)#есть переменные у которых vif>130
summary(model1)#R^2~0.2088

power = 1.3
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)#есть переменные у которых vif>60
summary(model1)#R^2~0.2085

power = 1.4
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)#есть переменные у которых vif>35
```

```r
summary(model1)#R^2~0.2084

power = 1.5
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)#есть переменные у которых vif>20
summary(model1)#R^2~0.2082

#R^2 изменяется очень медленно,перейдём сразу к power=1.9

power = 1.9
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)#есть переменные у которых vif>9
summary(model1)#R^2~0.208

power = 2
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)#vif у всех переменных<1,5
summary(model1)#R^2~0.183



# 3.Выделите наилучшие модели из построенных: по значимости параметров,включённых в
зависимости, и по объяснённому с помощью построенных зависимостей разбросу adjusted
R2 - R2adj.

#сравним лучшие модели из пункта 2
power = 2 #наилучшая p-статистика
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)
summary(model1)
#Multiple R-squared:  0.183,    Adjusted R-squared:  0.1809

power = 0.1 #наибольший R^2
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)
summary(model1)
#Multiple R-squared:  0.2155,   Adjusted R-squared:  0.2028

power = 0.2#p-статистика и R^2 схожая с model1 при power=0.1
model1 = lm(data = data2, salary ~ sex + working_hours + age + wed3 + higher_educ +
city_status + I(working_hours^power) + I(age^power))
vif(model1)
summary(model1)
#Multiple R-squared:  0.2146,   Adjusted R-squared:  0.2019

# Разброс R2 - R2_adj у model1 при power=2 - наименьший, а R^2 больше для степени 0.1

#Итог:среди моделей с наименьшей линейной зависимостью,с наилучшими по сравнению с
остальными показателями p-статистики у регрессоров,лучшей по R^2 оказалась модель для
степени 0.1



# 4. Сделайте вывод о том, какие индивиды получают наибольшую зарплату.

#Согласно этой модели больше всего зарабатывают молодые(ненадёжная p-статистика)
```

```
мужчины с высшим образованием, проживающие в городах, работающие много часов в
неделю.

# 5. Оцените регрессии для подмножества индивидов:

#1)Не вступавшие в брак, без высшего образования

power = 0.1
data3 = subset(data2, higher_educ  == 0)
data3 = subset(data3, wed3 == 1)
model1 = lm(data = data2, salary ~ sex + working_hours + age + city_status +
I(working_hours^power) + I(age^power))
summary(model1)#R^2~0.1688
#Больше всего зарабатывают молодые(ненадёжная p-статистика) мужчины,работающие
много,проживающие в городе

# 2) Городские жители, состоящие в браке
power = 0.1
data3 = subset(data2, city_status  == 1)
data3 = subset(data3, wed2 == 1)
model1 = lm(data = data2, salary ~ sex + working_hours + age + higher_educ +
I(working_hours^power) + I(age^power))
summary(model1)#R^2 ~ 0.16
# Наибольшая зарплата у мужчин с высшим образованием молодого(ненадёжная p-
статистика) возраста, работающих много
```

Приложение к задаче 4

```
!pip install pandas
!pip install sklearn
import pandas
import numpy as np
import warnings
warnings.filterwarnings('ignore')
data = pandas.read_csv('StudentsPerformance.csv', index_col='gender')
data_sel = data.loc[:, data.columns.isin(['gender', 'race/ethnicity', 'parental level
of education',
                                          'lunch', 'test preparation course', 'math
score', 'reading score','writing score'])]
data_sel['test preparation course'] = np.where(data_sel['test preparation course'] ==
'none', 0, 1)

data_sel['lunch'] = np.where(data_sel['lunch'] == 'free/reduced', 0,1)

data_sel['race/ethnicity'] = np.where(data_sel['race/ethnicity'] == 'group A',  0,
data_sel['race/ethnicity'])
data_sel['race/ethnicity'] = np.where(data_sel['race/ethnicity'] == 'group B',  1,
data_sel['race/ethnicity'])
data_sel['race/ethnicity'] = np.where(data_sel['race/ethnicity'] == 'group C',  2,
data_sel['race/ethnicity'])
data_sel['race/ethnicity'] = np.where(data_sel['race/ethnicity'] == 'group D',  3,
data_sel['race/ethnicity'])
data_sel['race/ethnicity'] = np.where(data_sel['race/ethnicity'] == 'group E',  4,
data_sel['race/ethnicity'])

data_sel['parental level of education'] = np.where(data_sel['parental level of educa-
tion'] == 'some college',  0, data_sel['parental level of education'])
data_sel['parental level of education'] = np.where(data_sel['parental level of educa-
tion'] == 'some high school',  1, data_sel['parental level of education'])
data_sel['parental level of education'] = np.where(data_sel['parental level of educa-
tion'] == 'high school',  2, data_sel['parental level of education'])
data_sel['parental level of education'] = np.where(data_sel['parental level of educa-
```

```
tion'] == "bachelor's degree",  3, data_sel['parental level of education'])
data_sel['parental level of education'] = np.where(data_sel['parental level of educa-
tion'] == "associate's degree",  4, data_sel['parental level of education'])
data_sel['parental level of education'] = np.where(data_sel['parental level of educa-
tion'] == "master's degree",  5, data_sel['parental level of education'])

data_sel = data_sel.dropna()
data_sel['writing score'] = np.where(data_sel['writing score'] >
np.average(data_sel['writing score']) , 0, 1)
writing_score = data_sel.loc[:, data_sel.columns.isin(['writing score'])]
X = data_sel.loc[:, data_sel.columns.isin(['gender', 'race/ethnicity', 'parental lev-
el of education', 'lunch',
                                           'test preparation course', 'math score',
'reading score'])]

from sklearn.model_selection import train_test_split
x_train, x_validation, y_train, y_validation = train_test_split(X, writing_score,
test_size=.33, random_state=5)
from sklearn.model_selection import StratifiedShuffleSplit
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import GridSearchCV
from sklearn.preprocessing import StandardScaler
from sklearn.svm import SVC

#опорные векторы
svm = SVC()
parameters = {'kernel':('linear', 'rbf'), 'C':(0.25,0.5,0.75,1),'gamma':
(1,2,3,'auto'),'decision_function_shape':('ovo','ovr'),'shrinking':(True,False)}
clf = GridSearchCV(svm, parameters)
clf.fit(x_train,y_train)

print("f1:"+str(np.average(cross_val_score(clf, x_validation, y_validation, scor-
ing='f1'))))
print("precision:"+str(np.average(cross_val_score(clf, x_validation, y_validation,
scoring='precision'))))
print("recall:"+str(np.average(cross_val_score(clf, x_validation, y_validation, scor-
ing='recall'))))

#случайный лес
from sklearn.ensemble import RandomForestClassifier
param_grid = { 'n_estimators': [50,100,150],'max_features': ['auto'],'max_depth' :
list(range(1, 10)), 'criterion' :['gini']}
RFC = GridSearchCV(estimator=RandomForestClassifier(), param_grid=param_grid, cv= 5,
refit = True)
RFC.fit(x_train,y_train)

print("f1:"+str(np.average(cross_val_score(RFC.best_estimator_, x_validation,
y_validation, scoring='f1'))))
print("precision:"+str(np.average(cross_val_score(RFC.best_estimator_, x_validation,
y_validation,scoring='precision'))))
print("recall:"+str(np.average(cross_val_score(RFC.best_estimator_, x_validation,
y_validation, scoring='recall'))))
```

Приложение к задаче 5

```
import numpy as np # библиотека для эффективной работы с данными
import pandas as pd # библиотека для работы с наборами данных
import matplotlib.pyplot as plt # библиотека для визуализации
import seaborn as sns # еще одна библиотека для построения графиков
data = pd.read_csv('creditcard.csv')
```

```
data.shape

data.info() # выводим информацию о наборе данных
data.describe() # статистический анализ числовых столбцов
data.corr() # корреляция числовых столбцов
plt.figure(figsize=(15,10))
sns.heatmap(data.corr(), xticklabels=data.corr().columns, ytick-
labels=data.corr().columns, cmap='RdYlGn', center=0, annot=True)

# Нормализация факторных переменных
from sklearn.preprocessing import StandardScaler
scale_features_std = StandardScaler()
features_std = scale_features_std.fit_transform(data[['Time', 'V1', 'V2', 'V3', 'V4',
'V5', 'V6', 'V7','V8', 'V9', 'V10', 'V11',
'V12', 'V13', 'V14', 'V15', 'V16', 'V17', 'V18','V19', 'V20', 'V21', 'V21',
'V22', 'V23', 'V24', 'V25', 'V26', 'V27', 'V28', 'Amount']])
features_std
data[['Time', 'V1', 'V2', 'V3', 'V4', 'V5', 'V6', 'V7','V8', 'V9', 'V10', 'V11',
'V12', 'V13', 'V14', 'V15', 'V16', 'V17', 'V18',
'V19', 'V20', 'V21', 'V21', 'V22', 'V23', 'V24', 'V25', 'V26', 'V27', 'V28',
'Amount']]= features_std
data.head()
data.describe() #Целевой признак
target=data.Class
train=data
#Выделяем тренировочную и тестовую выборки
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(train, target, test_size = 0.3,
random_state = 42)
N_train, _ = X_train.shape N_test, _ = X_test.shape print (N_train, N_test)
#Метод главных компонентов
from sklearn.decomposition import PCA
%matplotlib inline
import matplotlib.pyplot as plt
pca = PCA()
pca.fit(X_train)
X_pca = pca.transform(X_train)

for i, component in enumerate(pca.components_):
print("{} component: {}% of initial variance".format(i + 1, round(100 *
pca.explained_variance_ratio_[i], 2)))
print(" + ".join("%.3f x %s" % (value, name)for value, name in
zip(component,train.columns)))
plt.figure(figsize=(10,5))
plt.plot(np.cumsum(pca.explained_variance_ratio_), color='k', lw=2)
plt.axhline(0.9, c='r')
plt.axvline(10, c='b')

less_dimensional_X = pca.transform(X_train)

from sklearn.manifold import TSNE
tsne = TSNE(n_components=2, random_state=0)
tsne_results = tsne.fit_transform(less_dimensional_X)

tsne_df = pd.DataFrame({'X':tsne_results[:,0],
                        'Y':tsne_results[:,1],
                        'real_ans':y_train})
import seaborn as sns
import matplotlib.pyplot as plt
plt.figure(figsize=(7, 7))
sns.scatterplot(x="X", y="Y",
              data=tsne_df);
```