ФГБОУ ВО НОВОСИБИРСКИЙ ГАУ

Информационных технологий и моделирования

Рег. № ПИ.03-56
«05» 10 2022г.

УТВЕРЖДЕН
на заседании кафедры
Протокол от «23» 09 2022г. № 2
Заведующий кафедрой
информационных технологий и
моделирования
_____ О.В. Агафонова
(подпись)

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Б1.В.ДВ.03.01 Визуальный анализ данных
Шифр и наименование дисциплины

09.03.03 Прикладная информатика
Код и наименование направления подготовки

Прикладная информатика
Направленность (профиль)

Новосибирск 2022

Паспорт фонда оценочных средств

№ п/п	Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
1	2	7	
1.	Библиотека NumPy. Работа с массивами NumPy.	ПК-2	Тест
2.	Библиотека Pandas для обработки и анализа данных. Обработка данных в Pandas.	ПК-2	Индивидуальное задание Тест
3.	Визуализация данных. Визуализация с помощью библиотеки Matplotlib и Seaborn. Возможности библиотеки Pandasдля визуализации.	ПК-2	Индивидуальное задание Тест
4.	Элементы статистики. Подготовка и исследование данных.	ПК-2	Индивидуальное задание
	Контрольная работа, зачет с оценкой	ПК-2	Темы контрольной работы, вопросы к зачету с оценкой

Тест

Тема 1. Библиотека NumPy. Работа с массивами NumPy.

1. Что такое NumPy?

а. Python библиотека для работы с большими многомерными массивами и матрицами, имеющая большой набор математических функций для операций с этими массивами.

б. Библиотека Python для работы с числами, строками и другими типами данных.

в. Функция Python для анализа больших данных.

2. Как объявить массив numpy, если этот модуль импортирован как np? *
   а. array
   б. np.array
   в. list
   г. np.list
3. Какой тип будет у элементов np.array([1, '2', 3.5])? *
   а. np.int
   б. np.float
   в. U (unicode)
   г. возникнет ошибка при создании array
4. Какое максимальное число можно хранить в np.array с типом np.int16? (проверьте себя в Colab) *
5. Можно ли задать n-мерный массив в numpy? *
   а. Да
   б. Нет
6. Какая команда задаст массив размером 5*5 заполненный единицами? *
   а. np.full((5,5), 1)
   б. np.full((1,1), 5)
   в. np.ones((5,5))
   г. np.eye(5)
7. Сколько элементов содержится в np.zeros((4, 3, 12, 2)) *
8. Какой атрибут позволяет узнать размерность np.array? *
   а. size
   б. len
   в. shape
   г. reshape

Критерии оценки:

- оценка «зачтено» выставляется студенту, если 80 и более % правильных ответов;

- оценка «не зачтено» выставляется студенту, если правильных ответов > 20%.

Индивидуальное задание

Тема 2. Библиотека Pandas для обработки и анализа данных. Обработка данных в Pandas.

Обработка данных в Pandas.

1. Загрузите данные из файла в объект *DataFrame*, добавьте заголовки к столбцам: «index», «year», «month», «day», «min_t», «average_t», «max_t», «rainfall».

Расшифровка:

- index – индекс ВМО,
- year – год,
- month – месяц,
- day – день,
- min_t – минимальная температура воздуха,
- average_t – средняя температура воздуха,
- max_t – максимальная температура воздуха,
- rainfall – количество осадков.

2. Удалите столбец index.
3. Используя метод info(), ответьте на вопросы:

- Есть ли в данных пропущенные значения?
- В каком столбце данных больше всего пропущенных значений?

4. В данных за какой год больше всего пропусков?
5. Объедините столбцы «Год», «Месяц» и «День» в один столбец «Дата» в формате гггг-мм-дд (2000-01-20). Данные в новом столбце должны иметь формат *datetime*;
6. Для каждого наблюдения рассчитайте размах температур (разность максимальной и минимальной суточных температур) и количество предшествующих ему дней без осадков (используйте циклы Python и условный оператор):

Мин. темп. воздуха	Сред. темп. воздуха	Макс. темп. воздуха	Кол-во осадков	Размах темп.	Кол-во дней без осадков
-12.4	-11.0	-9.9	3.9	2.5	0
-28.1	-14.8	-9.8	3.8	18.3	0
-38.5	-34.6	-26.6	0.0	11.9	0
-34.6	-30.1	-23.4	0.0	11.2	1
-26.8	-21.4	-16.6	1.1	10.2	2
-28.6	-24.2	-17.4	0.8	11.2	0
-31.0	-27.0	-24.0	0.0	7.0	0
-33.3	-30.3	-24.6	0.0	8.7	1

7. Определите самый длинный период засухи.
8. Для каждого года вычислите среднегодовую температуру и общее количество осадков. Запишите результаты в объекты Series.

- Какой год можно считать самым теплым? Какой самым холодным?

- В какой год выпало больше всего осадков? В какой меньше всего?

Используя запись имя_серии.plot() постройте график и посмотрите как изменялась температура. С помощью имя_серии. .plot.bar() отобразите на столбиковой диаграмме количество осадков, выпавших в каждый год.

9. Выведете наблюдения, удовлетворяющие условиям:

- Средняя температура воздуха ниже -30 оС (для некоторых регионов можно использовать -10 ^{o}C, -35^{o}C, -40 ^{o}C).
- Средняя температура воздуха выше 27^{o}C и количество дней без осадков больше 3.

Ссылка на данные для задания
https://public.edu.asu.ru/mod/url/view.php?id=49629

Критерии оценки:

Оценочная шкала для итоговой проверки заключается в следующем:

1. Для отметки «Зачтено» необходимо набрать свыше 6 баллов.
2. Для отметки «Не зачтено» – количество балов от 0 до 6

Шкала распределения баллов для оценки работы

Количество баллов	Оценка в баллах			
	Задание выполнено полностью, без ошибок	Задание в целом выполнено, имеются незначительные замечания	Задание выполнено наполовину, есть серьезные замечания	Задание практически не выполнено
	10	7-9	4-7	0-4

Тест

Тема 2. Библиотека Pandas для обработки и анализа данных. Обработка данных в Pandas.

1. У нас есть следующий объект DataFrame:

```
df = pd.DataFrame({
    'country': ['Kazakhstan', 'Russia', 'Belarus', 'Ukraine'],
    'population': [17.04, 143.5, 9.5, 45.5],
    'square': [2724902, 17125191, 207600, 603628]
})
```

При этом мы заменяем индексы следующим способом:
df.index = ['KZ', 'RU', 'BY', 'UA']
Что нам надо сделать, чтобы добавить наименование индексации?

а. df.index.name = 'Country Code'
б. df.add(name='Country Code')
в. df.append.name('Country Code')

2. Имеется следующий объект типа Series:

```
my_series = pd.Series([5, 6, 7, 8, 9, 10])
```

При этом мы хотим заменить индексы следующим образом:
my_series.index = ['A', 'B', 'C', 'D', 'E']
Что мы получим в результате?

а. Ошибку, т.к. количество заменяемых индексов не совпадает с количеством исходных элементов
б. Замену всех индексов, кроме последнего. Он останется без изменений
в. Замену всех индексов, последний будет продублирован
г. Ошибку, т.к. таким методом нельзя производить замену индексов

3. Что будет содержать объект Series?

а. my_series = pd.Series([5, 6, 7, 8, 9, 10])
б. 0 5
   1 6
   2 7
   3 8
   4 9
   5 10
в. 1 5
   2 6
   3 7
   4 8

5 9
6 10

г. 5
6
7
8
9
10

4. С помощью какого метода мы можем обратиться к строкам по индексу объекта DataFrame?

а. .loc
б. .get
в. .index

5. У нас имеется файл с данными в формате "csv". Какой тип данных лучше всего подойдет, если мы хотим работать со всей информацией, хранящейся в данном файле?

а. Series
б. DataFrame
в. Словарь
г. Список

6. Что из себя представляет объект DataFrame?

а. Объект библиотеки Pandas, представляющий из себя табличную структуру данных.
б. Объект библиотеки Pandas, являющуюся словарем.
в. Python объект, описывающий формат данных типа "Дата"

7. С помощью какого метода можно построить сводную таблицу в Pandas?

а. .pivot_table
б. .groupby
в. .read_csv

8. Можно ли представить объект Series следующим образом?

```
my_series = pd.Series({'a': 5, 'b': 6, 'c': 7, 'd': 8})
```

а. Да, можно
б. Нет, нельзя

9. У нас есть объект DataFrame:

```
df = pd.DataFrame({
    'country': ['Kazakhstan', 'Russia', 'Belarus', 'Ukraine'],
    'population': [17.04, 143.5, 9.5, 45.5],
    'square': [2724902, 17125191, 207600, 603628]
})
```

В каком виде будут храниться данные?

а. country population square
0 Kazakhstan 17.04 2724902
1 Russia 143.50 17125191
2 Belarus 9.50 207600
3 Ukraine 45.50 603628
б. Id country population square
0 Kazakhstan 17.04 2724902
1 Russia 143.50 17125191
2 Belarus 9.50 207600
3 Ukraine 45.50 603628
в. country population square
Kazakhstan 17.04 2724902
Russia 143.50 17125191
Belarus 9.50 207600
Ukraine 45.50 603628

10.Что такое Pandas?

а. Библиотека Python для анализа данных.
б. Библиотека Python для работы с графикой.
в. Встроенная Python функция для анализа больших данных.

11.Имеется следующий код:

```
my_series = pd.Series([5, 6, 7, 8, 9, 10], index=['a', 'b', 'c', 'd', 'e', 'f'])
my_series[['a', 'b', 'f']] = 0
my_series[my_series > 0] * 2
```

Какой будет результат выполнения данного кода?

а. c 14
d 16
e 18
б. a 0
b 0
c 14
d 16

e 18
f 0

в. c 7
d 8
e 9

12.Что из себя представляет объект Series?

а. Структура из библиотеки Pandas, похожая на одномерный массив с использованием ассоциированных меток.
б. Структура из библиотеки Pandas, похожая на табличную структуру данных.
в. Объект библиотки NumPy, являющийся питоновским списком

13.Нужно ли обязательно указывать свой индекс при создании Pandas Series или Dataframe? *

а. Да
б. Нет

14.Какие типы данных можно считать pd.read? *

а. excel
б. jpeg
в. csv
г. json
д. mp3

15.Что будет, если сложить два массива pd.Series (пример: s1+s2), у которых не полностью совпадают индексы? *

а. Будет ошибка
б. Некоторые элементы будут nan
в. Получится конкатенация массивов

16.Какой метод выводит первые 10 строк объекта pd.DataFrame? *

а. .first(10)
б. .get(10)
в. .top(10)
г. .head(10)

17.Какой метод сбрасывает индекс у DataFrame? *

а. .drop_index()
б. .alter_index()
в. .reset_index()
г. .reindex()

18. Пусть df является объектом класса pd.DataFrame. Какой из следующих методов вставит вместо пропущенных значений строку 'unknown' и вернет новую таблицу? *

а. pd.fillna(df, 'unknown')
б. df.fillna('unknown')
в. df[df == np.nan] = 'unknown'
г. df.fillna('unknown', inplace=True)

19. Для чего нужны сводные (pivot tables) таблицы? *

а. Функция, которая нужна для совместимости pandas и других программ для работы с таблицами
б. Для компактного представления агрегированных данных
в. Для более эффективного хранения данных

Критерии оценки:

- оценка «зачтено» выставляется студенту, если 80 и более % правильных ответов;

- оценка «не зачтено» выставляется студенту, если правильных ответов > 20%.

Индивидуальное задание

Тема 3. Визуализация данных. Визуализация с помощью библиотек Matplotlib и Seaborn. Возможности библиотеки Pandas для визуализации.

В файле «vgsale_1.csv» содержатся данные о видеоиграх, выпущенных с 1980 по 2020 гг. В файле представлены 16598 наблюдений, каждое из которых имеет 10 характеристик:

- Name – название игры,
- Platform – игровая платформа (PC, PSP, X360 и др.),
- Year – год выпуска игры,
- Genre – жанр игры,
- Publisher – издатель игры,
- Other_Sales – продажи в странах мира (в миллионах),
- Global_Sales – объем продаж по всему миру.

Загрузите файл «vgsales_1.csv» в объект DataFrame, рассчитайте необходимые показатели и визуализируйте информацию, используя различные инструменты pandas.

Проанализируйте полученные графики и сделайте выводы:

1. Игры каких жанров были наиболее популярны до 2000 года, а какие после?

2. Оцените популярность жанров по количеству выпущенных игр и по объему продаж по всему миру.

Для визуализации полученных результатов используйте столбиковые диаграммы.

Замечание. Одна и та же игра может встречаться в выборке несколько раз, т.к. она может быть выпущена на нескольких платформах.

3. Отобразите на графике общее число видеоигр, выпущенных в каждом году.

4. Определите трех издателей, выпустивших наибольшее количество видеоигр.

5. Изобразите количество выпущенных издателями видеоигр для каждой платформы на столбиковой диаграмме (можно использовать диаграмму с накоплением).

6. Отобразите на круговых диаграммах доли суммарного объема продаж с 1980г. до 2000г. и с 2000г. до 2020г. по всему миру.

Критерии оценки:

Оценочная шкала для итоговой проверки заключается в следующем:

1. Для отметки «Зачтено» необходимо набрать свыше 6 баллов.
2. Для отметки «Не зачтено» – количество балов от 0 до 6

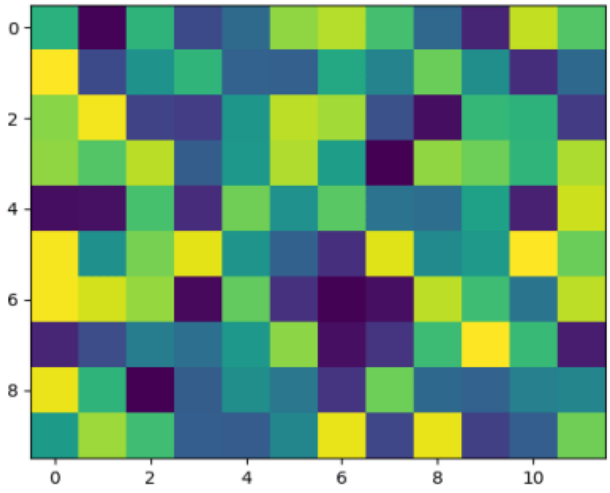
Шкала распределения баллов для оценки работы

Количество баллов	Оценка в баллах			
	Задание выполнено полностью, без ошибок	Задание в целом выполнено, имеются незначительные замечания	Задание выполнено наполовину, есть серьезные замечания	Задание практически не выполнено
	10	7-9	4-7	0-4

Тест

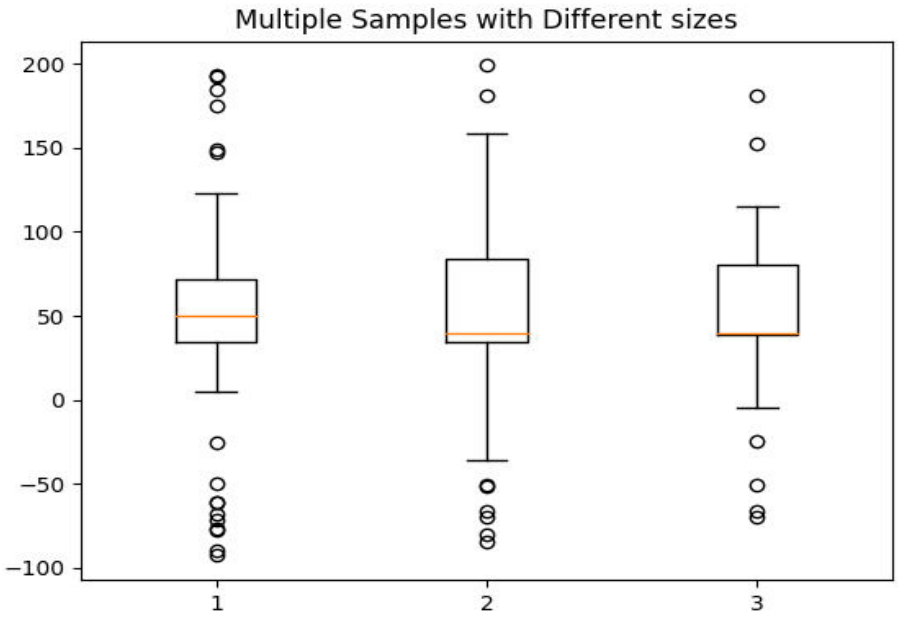
Тема 3. Визуализация данных. Визуализация с помощью библиотек Matplotlib и Seaborn. Возможности библиотеки Pandas для визуализации.

1. Какого типа следующий график?



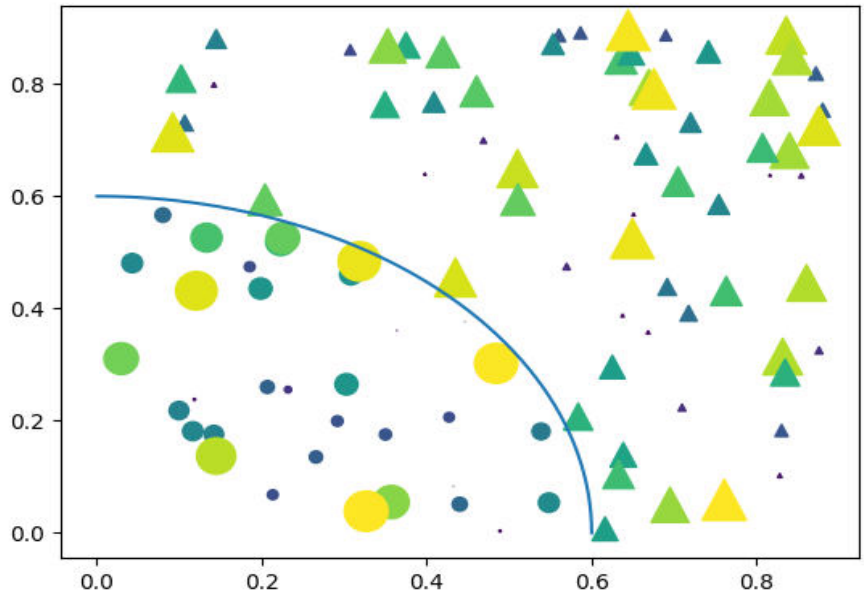
а. bar
б. hist
в. box
г. scatter
д. heat map

2. Какого типа следующий график?



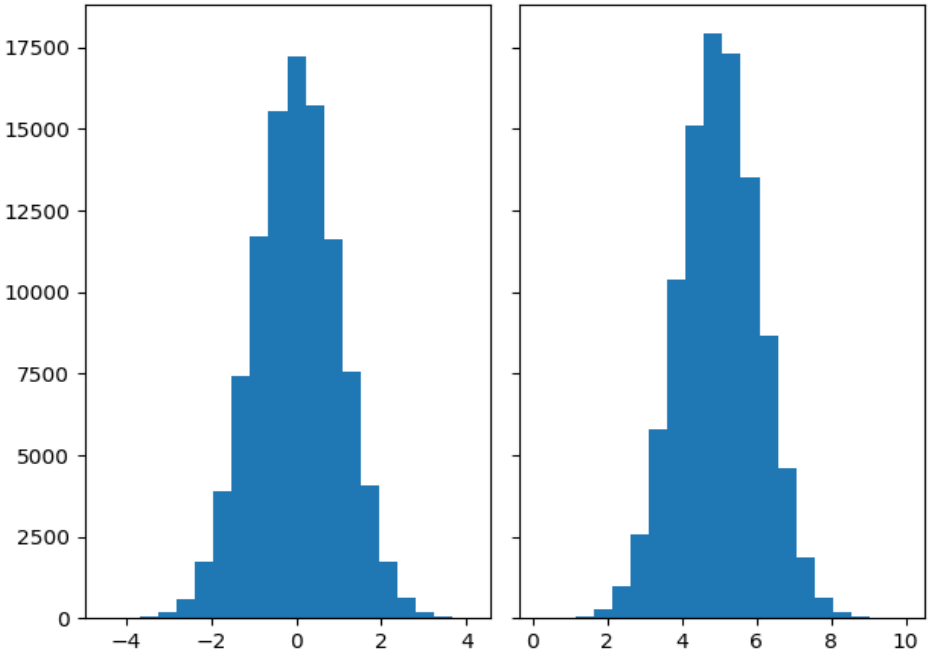
a. bar
b. hist
c. box
d. scatter
e. heat map

3. Какого типа следующий график?



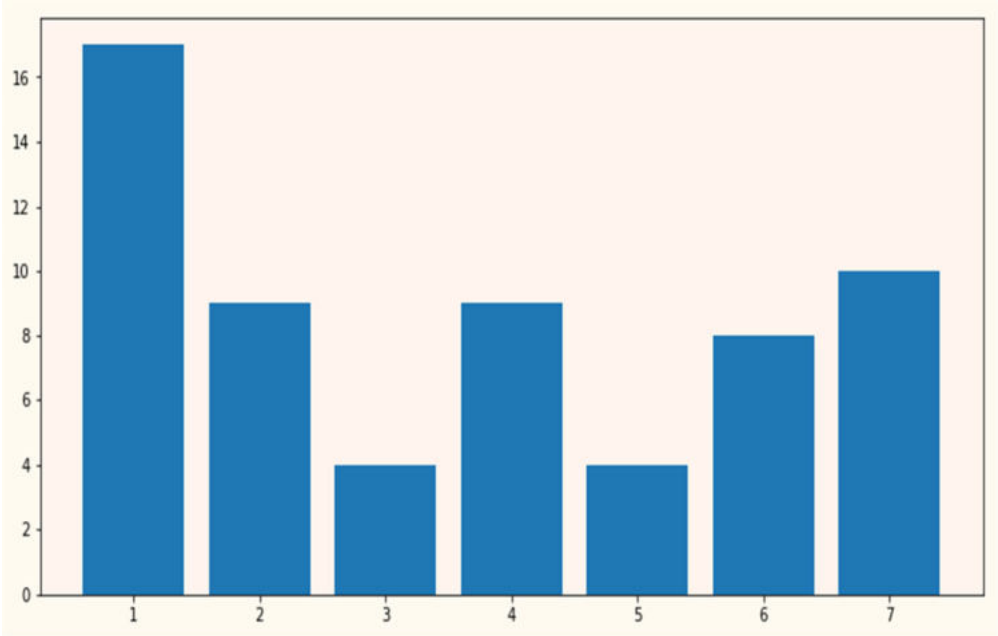
а. bar
б. hist
в. box
г. scatter
д. heat map

4. Какого типа следующий график?



а. bar
б. hist
в. box
г. scatter
д. heat map

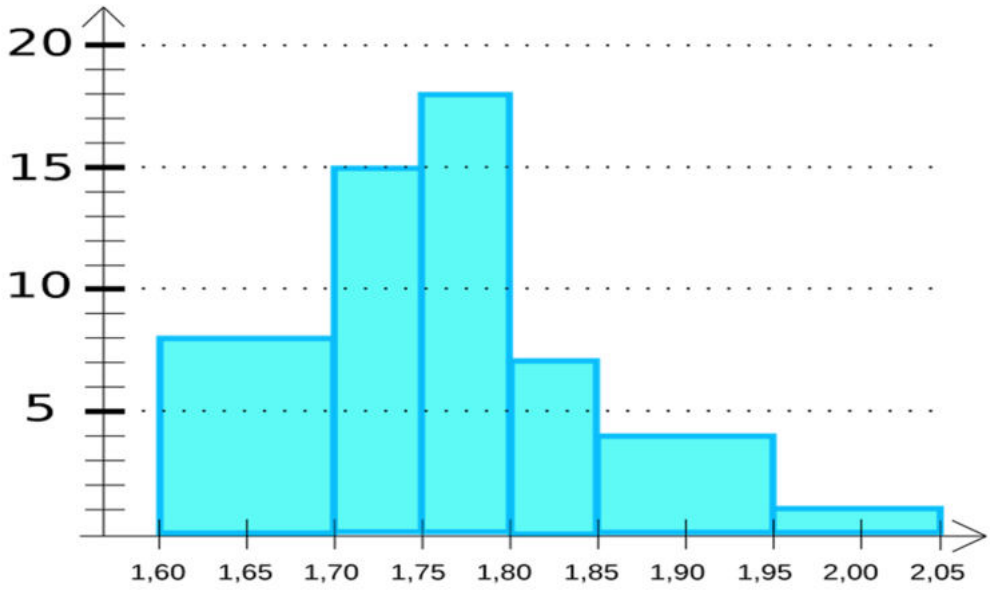
5. Какого типа следующий график?



а. bar
б. hist
в. box
г. scatter

д. heat map

6. Какой способ представления данных изображен на рисунке? *

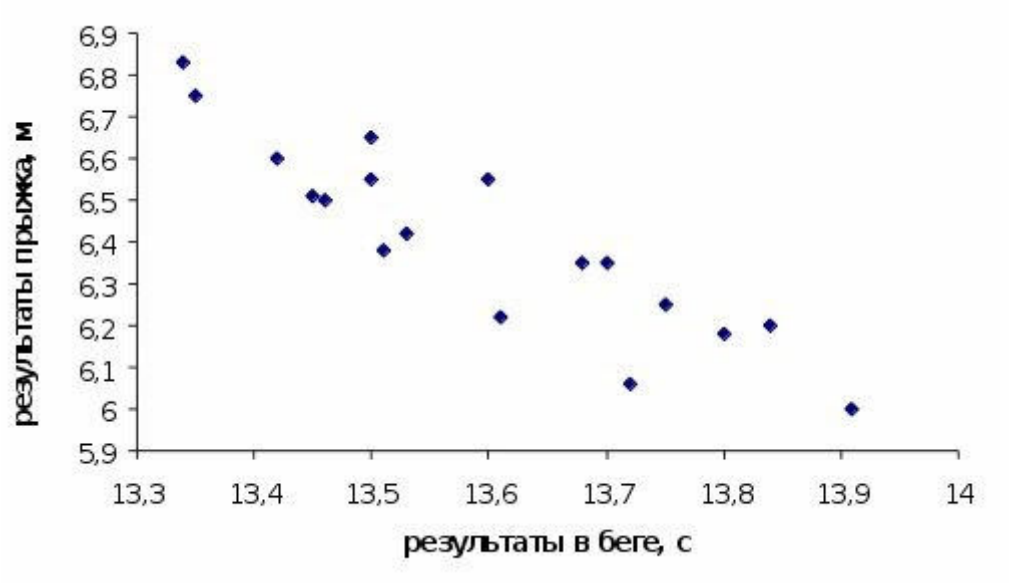


а. Дендрограмма
б. График функций
в. Гистограмма

7. Какой размерности графики можно создавать в matplotlib? *

а. 2D
б. 3D
в. 4D

8. Какая зависимость наблюдается в данных? *



а. Чем дольше бег, тем короче прыжок (отрицательная корреляция)
б. Чем дольше бег, тем длиннее прыжок (положительная корреляция)
в. Заметной зависимости не наблюдается (нулевая корреляция)

Критерии оценки:

- оценка «зачтено» выставляется студенту, если 80 и более % правильных ответов;

- оценка «не зачтено» выставляется студенту, если правильных ответов > 20%.

Индивидуальное задание

Тема 4. Элементы статистики. Подготовка и исследование данных. Практическая статистика и визуализация с Python

Подготовьтесь к выполнению практического задания:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
fromsklearn.preprocessing import StandardScaler,
OneHotEncoder
from sklearn.model_selection import train_test_split
import seaborn as sns

# можноитак: import seaborn as sb
fromscipy.stats import norm
fromscipy import stats
from pandas import DataFrame
%matplotlib inline
```

1) *Данные*

Данные (house_train.csv) представляют собой набор данных о ценах на жилье. Подготовьте информацию о датасете:

```
pd.set_option('display.max_columns', 100)
df = pd.read_csv('house_train.csv')
df.drop('Id', axis=1, inplace=True)
df.head()
```

Приведите описание датасета:

– Сколько данных в датасете?

– Сколько параметров? Выведите список всех параметров.

– Есть ли категориальные признаки? Перечислите / выведите их.

– Выведите первые пять строчек DataFrame.

По умолчанию pandas, ради экономии времени, указывает приблизительные сведения об использовании памяти объектом DataFrame. Если нас интересуют точные сведения, то нужно установить параметр memory_usage в значение *'deep'*:

Пусть Имя DataFrame – df, тогда исполняем инструкцию:

```
df.info(memory_usage='deep')
```

2) Просмотреть основную информацию по датасету можно, выполнив инструкции (выполнить ниже)

```
df.columns#Просмотр имен столбцов функций
df.shape#Просмотр количества строк и столбцов
```

df.describe() #Просмотр основной статистики

Статистическая сводка для числовых данных включает в себя среднее, минимальное и максимальное значения данных, которые могут быть полезны для определения размера некоторых переменных и оценивания того, какие переменные могут быть наиболее важными.

С помощью метода *describe()* выведите описательную статистику числовых данных:

df.describe()

А что будет, если выполнить такую инструкцию: df.describe().T ?

– получим статистику отдельного показателя, например, *"SalePrice": df['SalePrice'].describe()*

Укажите:

а) чему равно среднее значение?

б) чему равно стандартное отклонение?

3) Проверьте, есть ли пропуски и повторы в данных.

Пропущенные и неопределённые значения выявляет метод isna(), а суммарное количество таких значений – метод sum().

Вызов обоих методов можно записать в одну строку, разделив их точкой. Pythonсначала вызовет метод isna(), а затем результаты его работы передаст методу

sum():

print(df.isna().sum())

Количество пустых значений в наборе данных можно сохранить. Результат сохраним в переменной *na_number:*

na_number=(df.isna().sum())

print(na_number)

4) Повторяющиеся строки – дубликаты – выявляются методом

duplicated() и подсчитываются тем же *sum().* Если возвращаются нули, то данные пригодны для исследования:

print(df.duplicated().sum()) 0

Количество дубликатов в наборе данных можно сохранить, например, в переменной *duplicated_number:*

duplicated_number=df.duplicated().sum()

print(duplicated_number)

5) Получите список названий столбцов, запросив атрибут *columns.*

print(df.columns)

6) Анализ пропущенных значений и удаление

а) Выясним, в каких параметрах отсутствует статистика (данные):

na_count = df.isnull().sum().sort_values(ascending=False) #

Вычисляем, сколько пропущенных значений в параметрах *na_rate = na_count / len(df) #* Вычисляем частоту или вероятность, с которой

пропущенное значение встречается в каждом параметре. Если вероятность большая (>0.5), столбцы параметры можно смело удалять).

 # формируем массив для печати

na_data = pd.concat([na_count,

na_rate],axis=1,keys=['count','ratio'])

print(na_data)

Есть два способа обработать отсутствующие значения. Один – проанализировать, полезны ли параметры (признаки) с отсутствующими значениями для задачи. Бесполезные параметры удаляются.

Полезные же признаки зависят от количества отсутствующих значений. Если количество отсутствующих значений больше определенного количества (%) – удалите образцы, а если меньше определенного количества (%) – используйте среднее значение, медианное значение или моду для их восстановления.

Второй способ – проанализировать причины, по которым эти отсутствующие значения отсутствуют, и использовать определенный метод для их преобразования в тип данных (тип переменной).

Первые четыре параметра можно смело удалять.

Если количество отсутствующих данных для определенного признака достигает более 15%, то этот признак следует удалить, и считается, что такого признака нет в наборе данных – то есть мы не будем пытаться заполнить отсутствующие значения этих признаков).

Проследим за количеством оставшихся столбцов. Зафиксируем первоначальное количество столбцов с помощью функции:

df.shape (1460, 80)

Удалите столбцы с максимальным количеством отсутствующих данных:

'PoolQC', 'MiscFeature' и 'Alley'. Это не должно привести к уменьшению эффективного объема информации в данных, поскольку буквальные значения этих признаков, похоже, не имеют ничего общего с интересующим нас признаком – цена на жилье.

*df = df.drop(['PoolQC', 'MiscFeature', 'Alley'], axis=1)*или лучше создайте новый файл и выведите столбцы*df_new=df.drop(['PoolQC', 'MiscFeature', 'Alley'], axis=1)*

print(df_new.isna().sum())

Выведем отдельно количество оставшихся столбцов:

df_new.shape (1460, 77)

В оставшихся переменных с пропущенными значениями несколько функций GarageX. *'GarageQual' и 'GarageCond'* имеют одинаковое количество пропущенных значений. На основании этого делаем вывод, что они могут представлять один и тот же набор наблюдений, поэтому удалим эти функции. Ту же операцию можно выполнить для *Fence* – изгородь – 1179 нулей.

df_new = df_new.drop(['GarageQual', 'GarageCond', 'Fence'],

axis=1)

print(df_new.isna().sum()) 74 столбца осталось

Что касается *MasVnrArea и MasVnrType*, в соответствии с их буквальным значением, мы думаем, что они не важны, и у них есть сильная корреляция (как мы увидим дальше) с *YearBuilt и GeneralQual*. Таким образом, мы не потеряем никакой информации, если удалим эти две функции.

df_new=df_new.drop(['MasVnrArea', 'MasVnrType'], axis=1)

print(df_new.isna().sum()) 72 столбца осталось

Выведем отдельно количество оставшихся столбцов:

df_new.shape # Размер данных после обработки пропущенногозначения

В общем, мы удалили почти все переменные с пропущенными значениями.

7) Однофакторный анализ данных

– Переименуем опять файл:

df = df_new

Проверим размер данных:

df.shape

Считаем заново исходный файл с данными:

df = pd.read_csv('house_train.csv')

df.head()

7.1. Гистограмма

Построим гистограмму параметра *SalePrice* в библиотеке *seaborn:*

sns.distplot(df['SalePrice'])

Результат:

По рисунку: цена дома подчиняется нормальному распределению?

Можем рассчитать его асимметрию и эксцесс:

print("Skewness: %f" % df['SalePrice'].skew())

print("Kurtosis: %f" % df['SalePrice'].kurt())

Задание: Постройте гистограмму параметра SalePrice всех домов с заголовком

'Histogram of Sale Price', заголовокосих – 'price', заголовокосиу – 'count'.

7.2. Boxplot

Построить коробочную диаграмму (ящик с усами) признака *SalePrice* всех домов в данных. Боксплоты не показывают форму распределения, но они могут дать нам лучшее представление о центре и распространении распределения, а также о любых возможных выбросах, которые могут существовать. Боксплоты и гистограммы часто дополняют друг друга и помогают нам лучше понять данные.

Заголовок рисунка – *title='Box plot of Sale Price'*. Результат:

7.3. Гистограммы и Боксплоты по группам

На графиках по группам, мы можем видеть, как переменная меняется в ответ на изменение другой, например, как меняется стоимость дома SalePrice в зависимости от того, есть ли кондиционер или нет (параметр 'CentralAir'). Или, как цена дома SalePrice зависит от размера гаража и т.д.

а) Для построения Boxplot и гистограмм цены дома сгруппируем данные сондиционером (name = 'Withairconditioning') ибезкондиционера(name = 'Noairconditioning'); дляBoxplottitle = "BoxplotofSalePricebyairconditioning".

б) Длягистограммы – заголовок title='Histogram of House Sale Price for both with and with no Central air conditioning'

в) Выведем описательную статистику 'CentralAir' и 'SalePrice' спомощью инструкции:

df.groupby('CentralAir')['SalePrice'].describe()

Выход: 'N' – стоимость дома без кондиционера; 'Y' – стоимость дома сондиционером. Очевидно, что средняя цена продажи домов без кондиционеранамного ниже, чем у домов с кондиционером.

г) Постройте Boxplot и гистограмму цены продажи домов(параметр'SalePrice'), сгруппированные по размеру гаража (параметр'GarageCars'): *title = "BoxplotofSalePricebygaragesize".*

Используйте при группировке *name = 'no garage' и name = '1-car garage'* – если гараж для одной машины; *name = '2-car garage'* –если гараж для двух машин; name = *'3-car garage'* – если гараж для трехмашин; *name = '4-car garage'* – если гараж для четырех машин.

Судить о средней цене дома можно по черте внутри каждого блока.

Прокомментируйте среднюю цену дома в зависимости от размеров гаража.

Постройте:

д) Гистограмму цены продажи дома без гаража
title='Histogram of Sale Price of houses with no garage'
е) Гистограмму цены продажи дома с гаражом на 1 машину
title='Histogram of Sale Price of houses with 1-car garage'
ё) Гистограмму цены продажи дома с гаражом на 2 машины
title='Histogram of Sale Price of houses with 2-car garage')
ж) Гистограмму цены продажи дома с гаражом на 3 машины
title='Histogram of Sale Price of houses with 3-car garage'
з) Гистограмму цены продажи дома с гаражом на 4 машины
title='Histogram of Sale Price of houses with 4-car garage'

Критерии оценки:

Оценочная шкала для итоговой проверки заключается в следующем:

1. Для отметки «Зачтено» необходимо набрать свыше 6 баллов.
2. Для отметки «Не зачтено» – количество балов от 0 до 6

Шкала распределения баллов для оценки работы

Количество баллов	Оценка в баллах			
	Задание выполнено полностью, без ошибок	Задание в целом выполнено, имеются незначительные замечания	Задание выполнено наполовину, есть серьезные замечания	Задание практически не выполнено
	10	7-9	4-7	0-4

Темы контрольной работы

1. Оценка принадлежности объектов классам с помощью модели логистической регрессии.
2. Алгоритм k-ближайших соседей для классификации данных.
3. Сегментация клиентской базы с помощью иерархического кластерного анализа.
4. Классификация изображений с помощью CNN: Три марки молока.
5. Технология обработки, анализа и визуализация текстовых данных.
6. Недообучение и переобучение регрессионых моделей.
7. Анализ многомерных данных с помощью метода главных компонентов.
8. Предсказание кредитной платёжеспособности клиентов банка методом k-ближайших соседей.
9. Технологии обработки пропущенных значений в данных.
10. Прогнозирование стоимости недвижимости на основе линейной регрессии.

Критерии оценки:

– оценка «зачтено» выставляется студенту, если выполнены все требования к написанию и защите контрольной работы: обозначена проблема и обоснована её актуальность, сделан краткий анализ различных точек зрения на рассматриваемую проблему и логично изложена собственная позиция, сформулированы выводы, тема раскрыта полностью, выдержан объём, соблюдены требования к внешнему оформлению, даны правильные ответы на дополнительные вопросы. Работа может быть зачтена и в том случае, когда основные требования к контрольной работе и ее защите выполнены, но при этом допущены недочёты. В частности, имеются неточности в изложении материала; отсутствует логическая последовательность в суждениях; не выдержан объём контрольной работы; имеются упущения в оформлении; на дополнительные вопросы при защите даны неполные ответы;

– оценка «не зачтено» – тема контрольной работы не раскрыта, задания не выполнены, обнаруживается существенное непонимание проблемы.

Вопросы к зачету с оценкой

1. Что такое визуализация данных.
2. Классификация по цели представления данных.
3. Визуализация как этап анализа данных.
4. Характеристики средств визуализации данных.
5. Язык Python и особенности его стиля программирования.
6. Синтаксис и управляющие конструкции языка Python. Переменные, значения и их типы. Типы данных в Python.
7. Встроенные операции и функции. Основные алгоритмические конструкции.
8. Списки, кортежи и словари.
9. Наука о данных и Python. Библиотеки: NumPy, Pandas, Matplotlib.
10. Основы NumPy: многомерные массивы и векторные вычисления. Индексирование и вырезание. Универсальные функции: быстрые поэлементные операции над массивами.
11. Обработка данных с применением массивов.Типы данных в NumPy.
12. Визуализация данных в Python. Обзор библиотеки Matplotlib.
13. Возможности библиотеки Pandas для визуализации. Обзор библиотеки Pandas.
14. Возможности библиотеки Pandas для визуализации. Построение простых графиков, таких как гистограммы, графические графики, точечные графики.
15. Работа с данными в Pandas: типы данных, арифметические операторы, обработка данных, агрегирование, группировка, работа со строками, соединение данных.
16. Визуализация данных в Seaborn.
17. Основы математической статистики. Сбор, обработка и анализ данных с помощью Python.
18. Введение в анализ табличных данных в Python. Пакет pandas. Объекты Series (последовательность) и DataFrame (таблица).

Критерии оценки:

– отметка **«отлично»** выставляется студенту, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, использует в ответе материал монографической литературы, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.

– отметка **«хорошо»** выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения.

– отметка **«удовлетворительно»** выставляется обучающемуся, если он

имеет знания только основного материала, но не усвоил его деталей, демонстрирует недостаточно систематизированные теоретические знания программного материала, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ.

– отметка **«неудовлетворительно»** выставляется обучающемуся, который не знает значительной части программного материала, допускает существенные ошибки при его изложении, неуверенно, с большими затруднениями выполняет практические работы.

ЗАДАНИЯ
ДЛЯ ОЦЕНКИ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИИ

Задания для оценки сформированности компетенции ПК-2:

1. Что называется математической статистикой?
Ответ: …

2. Для чего используется библиотека Matplotlib?
Ответ: …

3. Сопоставьте имена объектов и функций в Pandas с их назначениями:

1. DataFrame	a) Хранит таблицы
2. Series	b) Применяет функции ко всем значениям
3. apply	c) Считает среднее
4. mean	d) Хранит ряд данных

Ответ 1a,2d,3b,4c

4. Напишите имя функции, используемой для группировок
Ответ: …

5. Сопоставьте функции с графиками, которые они строят:

1. .plot.bar()	a) Столбчатая диаграмма
2. .plot.pie()	b) Линейная диаграмма
3. .plot()	c) Круговая диаграмма

Ответ:1a, 2c,3b

6. Как называется выражение внутри квадратных скобок?
lst = [i for i in range (1,10)]
Ответ: …

7. Какой метод отвечает за объединение массивов по вертикали?
a) stack()
b) ver_stack()
c) vstack()
d)hstack()
Ответ: c)

8. Какой метод отвечает за создание массива, используя кортежи?

a) array_tuple();
b) numpy.array();
c) tuple_num_array();
d) arrays().
Ответ:b)

9. С помощью, какой функции можно отсортировать список?
Ответ: …

10. Numpy – это
a) математическая библиотека с поддержкой многомерных массивов;
b) библиотека для обработки и анализа данных;
c) библиотека для построения графиков;
d) библиотека для перевода текста.
Ответ: a)

Критерии оценки результатов тестирования:

– оценка «отлично» выставляется студенту, если он отвечает верно на 80-100% вопросов.

– оценка «хорошо», выставляется студенту, если он отвечает верно на 70-79% вопросов.

– оценка «удовлетворительно», выставляется студенту, если он отвечает верно на 60-69% вопросов.

– оценка «неудовлетворительно» выставляется студенту, если он не освоил материал темы, дает менее 60% правильных ответов.

МАТРИЦА СООТВЕТСТВИЯ КРИТЕРИЕВ ОЦЕНКИ УРОВНЮ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Критерии оценки	Уровень сформированности компетенций
Оценка по пятибалльной системе	
«*Отлично*»	«*Высокий уровень*»
«*Хорошо*»	«*Повышенный уровень*»
«*Удовлетворительно*»	«*Пороговый уровень*»
«*Неудовлетворительно*»	«*Не достаточный*»
Оценка по системе «зачет – незачет»	
«*Зачтено*»	«*Достаточный*»
«*Не зачтено*»	«*Не достаточный*»

Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

1. Положение «О балльно-рейтинговой системе аттестации студентов»: СМК ПНД 08-01-2022, введено приказом от 28.09.2011 №371-О (http://nsau.edu.ru/file/403: режим доступа свободный);

2. Положение «О проведении текущего контроля и промежуточной аттестации обучающихся в ФГБОУ ВО Новосибирский ГАУ»: СМК ПНД 77-01-2022, введено в действие приказом от 03.08.2015 №268а-О (http://nsau.edu.ru/file/104821: режим доступа свободный).