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

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Системы обработки информации и управления

Рубежный контроль №1

По курсу «Технологии машинного обучения»

Вариант 11

Подготовила:

Студентка группы ИУ5-65Б.

Очеретная С.В.

08.04.2022

Проверил:

Преподаватель кафедры ИУ5

Гапанюк Ю.Е.

Москва, 2022 г.

Тема: Технологии разведочного анализа и обработки данных.

Номер варианта	Номер задачи	Номер набора данных, указанного в задаче
11	2	3

Дополнительные требования по группам:

- Для студентов группы ИУ5-65Б - для набора данных построить "парные диаграммы".

Задача №2.

Для заданного набора данных проведите обработку пропусков в данных для одного категориального и одного количественного признака. Какие способы обработки пропусков в данных для категориальных и количественных признаков Вы использовали? Какие признаки Вы будете использовать для дальнейшего построения моделей машинного обучения и почему?

Загрузка библиотек и датасета

Набор данных: датасет 3

```
import numpy as np
import pandas as pd
import seaborn as sns
from sklearn.impute import SimpleImputer
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
```

```
filename = '../datasets/marvel-wikia-data.csv'
data = pd.read_csv(filename)
data.head()
```

```
   page_id                                name  \
0     1678           Spider-Man (Peter Parker)
1     7139     Captain America (Steven Rogers)
2    64786  Wolverine (James \"Logan\" Howlett)
3     1868    Iron Man (Anthony \"Tony\" Stark)
4     2460                  Thor (Thor Odinson)

                                urlslug                ID  \
0             \/Spider-Man_(Peter_Parker)   Secret Identity
1       \/Captain_America_(Steven_Rogers)   Public Identity
2  \/Wolverine_(James_%22Logan%22_Howlett)   Public Identity
3    \/Iron_Man_(Anthony_%22Tony%22_Stark)   Public Identity
4                  \/Thor_(Thor_Odinson)  No Dual Identity

               ALIGN         EYE        HAIR              SEX  GSM  \
0     Good Characters  Hazel Eyes  Brown Hair  Male Characters  NaN
1     Good Characters   Blue Eyes  White Hair  Male Characters  NaN
2  Neutral Characters   Blue Eyes  Black Hair  Male Characters  NaN
```

```
3     Good Characters   Blue Eyes  Black Hair  Male Characters  NaN
4     Good Characters   Blue Eyes  Blond Hair  Male Characters  NaN
```

	ALIVE	APPEARANCES	FIRST APPEARANCE	Year
0	Living Characters	4043.0	Aug-62	1962.0
1	Living Characters	3360.0	Mar-41	1941.0
2	Living Characters	3061.0	Oct-74	1974.0
3	Living Characters	2961.0	Mar-63	1963.0
4	Living Characters	2258.0	Nov-50	1950.0

Обработка пропусков

```
data.shape
```

```
(16376, 13)
```

```
# пропущенные значения
data.isnull().sum()
```

```
page_id                 0
name                    0
urlslug                 0
ID                   3770
ALIGN                2812
EYE                  9767
HAIR                 4264
SEX                   854
GSM                 16286
ALIVE                   3
APPEARANCES          1096
FIRST APPEARANCE      815
Year                  815
dtype: int64
```

```
data.dtypes
```

```
page_id               int64
name                 object
urlslug              object
ID                   object
ALIGN                object
EYE                  object
HAIR                 object
SEX                  object
GSM                  object
ALIVE                object
APPEARANCES         float64
FIRST APPEARANCE     object
Year                float64
dtype: object
```

```
data_clean = data
```

Категориальные признаки:

```
# Импьютация константой NA колонки ID
imp_id = SimpleImputer(missing_values=np.nan, strategy='constant', fill_value
```

```
='NA')
data_clean[['ID']] = imp_id.fit_transform(data_clean[['ID']])

# Импьютация константой NA колонки ALIGN
imp_ALIGN = SimpleImputer(missing_values=np.nan, strategy='constant', fill_va
lue='NA')
data_clean[['ALIGN']] = imp_ALIGN.fit_transform(data_clean[['ALIGN']])

# Импьютация самым частым колонки EYE
imp_EYE = SimpleImputer(missing_values=np.nan, strategy='most_frequent')
data_clean[['EYE']] = imp_EYE.fit_transform(data_clean[['EYE']])

# Импьютация самым частым колонки HAIR
imp_HAIR = SimpleImputer(missing_values=np.nan, strategy='most_frequent')
data_clean[['HAIR']] = imp_HAIR.fit_transform(data_clean[['HAIR']])

# Импьютация константой NA колонки SEX
imp_SEX = SimpleImputer(missing_values=np.nan, strategy='constant', fill_valu
e='NA')
data_clean[['SEX']] = imp_SEX.fit_transform(data_clean[['SEX']])

# Удалим колонку GSM, т.к. она почти пустая
data_clean = data_clean.drop(columns = ['GSM'], axis = 1)

# Импьютация константой NA колонки ALIVE
imp_ALIVE = SimpleImputer(missing_values=np.nan, strategy='constant', fill_va
lue='NA')
data_clean[['ALIVE']] = imp_ALIVE.fit_transform(data_clean[['ALIVE']])

# Импьютация константой NA колонки FIRST APPEARANCE
imp_FIRST_APPEARANCE = SimpleImputer(missing_values=np.nan, strategy='constan
t', fill_value='NA')
data_clean[['FIRST APPEARANCE']] = imp_FIRST_APPEARANCE.fit_transform(data_cl
ean[['FIRST APPEARANCE']])
```

Количественные признаки:

```
strategies=['median', 'most_frequent']

# импьютация нужной колонки с помощью нужной стратегии
def func_impute_col(dataset, column, strategy_param):
    temp_data = dataset[[column]]

    imp_num = SimpleImputer(strategy=strategy_param)
    data_num_imp = imp_num.fit_transform(temp_data)

    return data_num_imp

# замена медианой APPEARANCES
col_APPEARANCES_imp = func_impute_col(data_clean, 'APPEARANCES', strategies[0
])
data_clean[['APPEARANCES']] = col_APPEARANCES_imp

# замена часто встречаемым Year
col_Year_imp = func_impute_col(data_clean, 'Year', strategies[1])
data_clean[['Year']] = col_Year_imp

data_clean.isnull().sum()
```

```
page_id             0
name                0
urlslug             0
ID                  0
ALIGN               0
EYE                 0
HAIR                0
SEX                 0
ALIVE               0
APPEARANCES         0
FIRST APPEARANCE    0
Year                0
dtype: int64
```

Ответы на вопросы

Какие способы обработки пропусков в данных для категориальных и количественных признаков Вы использовали?

- замена константой (constant) - для категориальных
- замена самым часто встречаемым значением (most_frequent) - для категориальных и количественных
- замена медианой - для количественных

Какие признаки Вы будете использовать для дальнейшего построения моделей машинного обучения и почему?

- колонка GSM использоваться не будет, т.к. содержит очень много пропущенных значений
- столбец EYE тоже бы удалила из-за большого количества пропусков
- остальные признаки необходимы для описания датасета, поэтому их бы я отставила
- большинство признаков являются строковыми, а не числовыми, поэтому по корреляционной матрице отсеивать не будем