Министерство образования и науки 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**

Отчет по лабораторной работе № 2
«Обработка признаков»
по курсу “Методы машинного обучения”

Исполнитель:
Студент группы ИУ5-22М
Желанкина А.С.
05.03.2021

Москва, 2021

Задание лабораторной работы

Выбрать набор данных (датасет), содержащий категориальные и числовые признаки и пропуски в данных. Для выполнения следующих пунктов можно использовать несколько различных наборов данных (один для обработки пропусков, другой для категориальных признаков и т.д.)

Для выбранного датасета (датасетов) на основе материалов лекций решить следующие задачи:

- устранение пропусков в данных;
- кодирование категориальных признаков;
- нормализацию числовых признаков.

Описание датасета

Рассмотрим статистику стартапов, которые были созданы в промежутке между 2011 и 2012 годами. Выбор этого периода объясняется тем, что за это время часть исследуемых стартапов с большой вероятностью достигла поставленных целей. В то время как стартапы основанные после 2013 года рассматривать рано, так как многие из них еще не успели достичь правильно интерпретируемых результатов. Для построения модели была использована база стартапов Crunchbase. Из неё был сформирован датасет, состоящий из 3987 строк и 19 столбцов.

Экранные формы с текстом программы и примерами её выполнения

Такая информация выводится о датасете.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3999 entries, 0 to 3998
Data columns (total 41 columns):
 #   Column                                        Non-Null Count  Dtype
---  ------                                        --------------  -----
 0   Organization Name                             3999 non-null   object
 1   Organization Name URL                         3999 non-null   object
 2   Headquarters Location                         3994 non-null   object
 3   Total Equity Funding Amount                   281 non-null    float6
4
 4   Total Equity Funding Amount Currency          281 non-null    object
 5   Total Equity Funding Amount Currency (in USD) 281 non-null    float6
4
 6   Exit Date                                     922 non-null    object
 7   Exit Date Precision                           922 non-null    object
 8   Founded Date                                  3994 non-null   object
 9   Founded Date Precision                        3994 non-null   object
 10  Investor Type                                 25 non-null     object
 11  Industry Groups                               3964 non-null   object
 12  Number of Employees                           3881 non-null   object
```

```
 13  Last Equity Funding Amount                        2931 non-null   float6
4
 14  Last Equity Funding Amount Currency               2931 non-null   object
 15  Last Equity Funding Amount Currency (in USD)      2931 non-null   float6
4
 16  Funding Status                                    2675 non-null   object
 17  Total Funding Amount                              3297 non-null   float6
4
 18  Total Funding Amount Currency                     3297 non-null   object
 19  Total Funding Amount Currency (in USD)            3297 non-null   float6
4
 20  Last Funding Date                                 3369 non-null   object
 21  Last Funding Amount                               2960 non-null   float6
4
 22  Last Funding Amount Currency                      2960 non-null   object
 23  Last Funding Amount Currency (in USD)             2960 non-null   float6
4
 24  Number of Funding Rounds                          3369 non-null   float6
4
 25  Number of Investors                               3125 non-null   float6
4
 26  Acquisition Status                                1333 non-null   object
 27  Acquired by                                       726 non-null    object
 28  Acquired by URL                                   726 non-null    object
 29  Announced Date                                    726 non-null    object
 30  Announced Date Precision                          726 non-null    object
 31  IPO Date                                          223 non-null    object
 32  IPO Status                                        3994 non-null   object
 33  Delisted Date                                     16 non-null     object
 34  Delisted Date Precision                           17 non-null     object
 35  SimilarWeb - Monthly Visits                       3354 non-null   object
 36  IPqwery - Patents Granted                         2573 non-null   object
 37  IPqwery - Trademarks Registered                   2573 non-null   object
 38  Website                                           2996 non-null   object
 39  Description                                       2000 non-null   object
 40  Industries                                        2992 non-null   object
dtypes: float64(10), object(31)
```

Так как присутствует большое количество колонок, в которых достаточно сложно заполнить пропуски (больше 50%), или коррелирующих между собой колонок, то такие данные было решено удалить.

In [6]:
```
data = data.drop('Organization Name URL', 1)
data = data.drop('Total Equity Funding Amount', 1)
data = data.drop('Total Equity Funding Amount Currency', 1)
data = data.drop('Total Equity Funding Amount Currency (in USD)', 1)
data = data.drop('Exit Date', 1)
data = data.drop('Exit Date Precision', 1)
data = data.drop('Founded Date Precision', 1)
data = data.drop('Investor Type', 1)
data = data.drop('Last Equity Funding Amount Currency', 1)
data = data.drop('Last Equity Funding Amount Currency (in USD)', 1)
data = data.drop('Last Funding Amount Currency', 1)
data = data.drop('Last Funding Amount Currency (in USD)', 1)
data = data.drop('Acquired by', 1)
data = data.drop('Acquired by URL', 1)
data = data.drop('Announced Date', 1)
data = data.drop('Announced Date Precision', 1)
data = data.drop('IPO Date', 1)
data = data.drop('Delisted Date', 1)
data = data.drop('Delisted Date Precision', 1)
data = data.drop('Description', 1)
data = data.drop('Total Funding Amount Currency', 1)
data = data.drop('Total Funding Amount Currency (in USD)', 1)
data = data.drop('Last Equity Funding Amount', 1)
data = data.drop('Total Funding Amount', 1)
```

In [7]:
```
data = data.dropna(subset=['IPO Status'])
```

Новый датасет имеет такие данные:

In [8]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 3994 entries, 0 to 3998
Data columns (total 17 columns):
 #   Column                          Non-Null Count  Dtype  
---  ------                          --------------  -----  
 0   Organization Name               3994 non-null   object 
 1   Headquarters Location           3994 non-null   object 
 2   Founded Date                    3994 non-null   object 
 3   Industry Groups                 3964 non-null   object 
 4   Number of Employees             3881 non-null   object 
 5   Funding Status                  2675 non-null   object 
 6   Last Funding Date               3369 non-null   object 
 7   Last Funding Amount             2960 non-null   float64
 8   Number of Funding Rounds        3369 non-null   float64
 9   Number of Investors             3125 non-null   float64
 10  Acquisition Status              1333 non-null   object 
 11  IPO Status                      3994 non-null   object 
 12  SimilarWeb - Monthly Visits     3354 non-null   object 
 13  IPqwery - Patents Granted       2573 non-null   object 
 14  IPqwery - Trademarks Registered 2573 non-null   object 
 15  Website                         2991 non-null   object 
 16  Industries                      2992 non-null   object 
dtypes: float64(3), object(14)
memory usage: 561.7+ KB
```

Пропуски в числовых значениях заменяли на значение среднего в данном столбце.

In [9]:
```
def repl(col):
    new_col = []
    for n in col:
        if type(n) == str:
            #print('{} = {}, {}'.format(n, n.replace(',', ''), i))
            n = n.replace(',', '')
            #print(n)
            n = float(n)
        new_col.append(n)
    new_col = pd.Series(new_col)
    return new_col
```

In [10]:
```
array = ['Last Funding Amount',
         'Number of Funding Rounds',
         'Number of Investors',
         'SimilarWeb - Monthly Visits',
         'IPqwery - Patents Granted',
         'IPqwery - Trademarks Registered']
for i in array:
    data[i] = repl(data[i])
    data[i] = data[i].fillna(data[i].mean())
```

Заполнение категориальных пропусков зависит от столбца, в котором есть пропуски. Заполнялось либо наиболее вероятным значением, либо наиболее часто встречающимся, либо ничего не значащим.

In [11]:
```
data['Acquisition Status'] = data['Acquisition Status'].fillna('Was not Acquired')
```

In [12]:
```
simp = SimpleImputer(missing_values=np.nan, strategy='most_frequent')
data[['Industry Groups']] = simp.fit_transform(data[['Industry Groups']])
data[['Number of Employees']] = simp.fit_transform(data[['Number of Employees']])
data[['Industries']] = simp.fit_transform(data[['Industries']])
```

In [13]:
```
simp2 = SimpleImputer(missing_values=np.nan, strategy='constant', fill_value='!!!')
data[['Funding Status']] = simp.fit_transform(data[['Funding Status']])
data[['Last Funding Date']] = simp.fit_transform(data[['Last Funding Date']])
data[['Website']] = simp.fit_transform(data[['Website']])
```

Кодирование категориальных значений с помощью LabelEncoder.

In [15]:
```
label_enc = LabelEncoder()
obj_columns = ['Organization Name', 'Headquarters Location', 'Founded Date',
               'Industry Groups', 'Number of Employees', 'Funding Status',
               'Last Funding Date', 'Website', 'Industries']

for obj in obj_columns:
    data[obj] = label_enc.fit_transform(data[obj])
```

Нормализация обучающей выборки из датасета производилась с помощью MinMaxScaler.

In [37]:
```
min_max_sc = MinMaxScaler()

x_train = min_max_sc.fit_transform(x_train)
X_test = min_max_sc.transform(X_test)
```