

**Московский государственный технический
университет им. Н.Э. Баумана.**

Факультет «Информатика и управление»

Кафедра «Системы обработки информации и управления»

Курс «Теория машинного обучения»

Отчет по лабораторной работе №2

Выполнил:
студент группы ИУ5-64
Кузьмин Роман

Подпись и дата:

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю.Е.

Подпись и дата:

Москва, 2022 г.

Описание задания

1. Выбрать набор данных (датасет), содержащий категориальные 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.)
2. Для выбранного датасета (датасетов) на основе материалов лекции решить следующие задачи:
 - обработку пропусков в данных;
 - кодирование категориальных признаков;
 - масштабирование данных.

Текст программы и её результаты

```
import matplotlib.pyplot as plt
from matplotlib import pyplot
import missingno
import seaborn as sns
import pandas as pd
import numpy as np
```

```
[ ] data = pd.read_csv('/content/hotel_bookings.csv')
data.head()
```

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arrival_date_week
0	Resort Hotel	0	342	2015	July	
1	Resort Hotel	0	737	2015	July	
2	Resort Hotel	0	7	2015	July	
3	Resort Hotel	0	13	2015	July	
4	Resort Hotel	0	14	2015	July	

5 rows x 32 columns

```

del data['assigned_room_type']
del data['reserved_room_type']
del data['customer_type']
del data['reservation_status']
del data['reservation_status_date']
del data['distribution_channel']

```

```
data.isna().sum()
```

```

hotel                                0
is_canceled                          0
lead_time                            0
arrival_date_year                    0
arrival_date_month                   0
arrival_date_week_number             0
arrival_date_day_of_month            0
stays_in_weekend_nights              0
stays_in_week_nights                0
adults                              0
children                             4
babies                              0
meal                                 0
country                             488
market_segment                       0
is_repeated_guest                    0
previous_cancellations               0
previous_bookings_not_canceled       0
booking_changes                      0
deposit_type                         0
agent                               16340
company                             112593
days_in_waiting_list                0
adr                                  0
required_car_parking_spaces          0
total_of_special_requests            0
dtype: int64

```

```
for col in data.columns:
    pct_missing = np.mean(data[col].isnull())
    print('{} - {}'.format(col, round(pct_missing*100)))
```

```
hotel - 0%
is_canceled - 0%
lead_time - 0%
arrival_date_year - 0%
arrival_date_month - 0%
arrival_date_week_number - 0%
arrival_date_day_of_month - 0%
stays_in_weekend_nights - 0%
stays_in_week_nights - 0%
adults - 0%
children - 0%
babies - 0%
meal - 0%
country - 0%
market_segment - 0%
is_repeated_guest - 0%
previous_cancellations - 0%
previous_bookings_not_canceled - 0%
booking_changes - 0%
deposit_type - 0%
agent - 14%
company - 94%
days_in_waiting_list - 0%
adr - 0%
required_car_parking_spaces - 0%
total_of_special_requests - 0%
```

```
del data['company']
```

```
from sklearn.impute import SimpleImputer  
imp = SimpleImputer(missing_values=np.nan, strategy='most_frequent')  
data['country'] = imp.fit_transform(data[['country']])
```

```
data['agent'] = data['agent'].replace(np.nan, 0)  
data['children'] = data['children'].replace(np.nan, 0)
```

```
data.isna().sum()
```

hotel	0
is_canceled	0
lead_time	0
arrival_date_year	0
arrival_date_month	0
arrival_date_week_number	0
arrival_date_day_of_month	0
stays_in_weekend_nights	0
stays_in_week_nights	0
adults	0
children	0
babies	0
meal	0
country	0
market_segment	0
is_repeated_guest	0
previous_cancellations	0
previous_bookings_not_canceled	0
booking_changes	0
deposit_type	0
agent	0
days_in_waiting_list	0
adr	0
required_car_parking_spaces	0

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 119390 entries, 0 to 119389
Data columns (total 25 columns):
#   Column                                  Non-Null Count  Dtype
---  -
0   hotel                                  119390 non-null  object
1   is_canceled                           119390 non-null  int64
2   lead_time                             119390 non-null  int64
3   arrival_date_year                     119390 non-null  int64
4   arrival_date_month                    119390 non-null  object
5   arrival_date_week_number              119390 non-null  int64
6   arrival_date_day_of_month             119390 non-null  int64
7   stays_in_weekend_nights               119390 non-null  int64
8   stays_in_week_nights                  119390 non-null  int64
9   adults                                119390 non-null  int64
10  children                              119390 non-null  float64
11  babies                                119390 non-null  int64
12  meal                                   119390 non-null  object
13  country                               119390 non-null  object
14  market_segment                        119390 non-null  object
15  is_repeated_guest                     119390 non-null  int64
16  previous_cancellations                 119390 non-null  int64
17  previous_bookings_not_canceled         119390 non-null  int64
18  booking_changes                        119390 non-null  int64
19  deposit_type                           119390 non-null  object
20  agent                                  119390 non-null  float64
21  days_in_waiting_list                   119390 non-null  int64
22  adr                                     119390 non-null  float64
23  required_car_parking_spaces            119390 non-null  int64
24  total_of_special_requests              119390 non-null  int64
dtypes: float64(3), int64(16), object(6)
memory usage: 22.8+ MB
```

```
[ ] data = pd.get_dummies(data, columns=['hotel','arrival_date_month', 'meal',
                                         'country', 'market_segment', 'deposit_type'])
```

```
[ ] data.head()
```

	is_canceled	lead_time	arrival_date_year	arrival_date_week_number	arrival_date_day_of_month	stays_in_weekend_nights
0	0	342	2015	27		1
1	0	737	2015	27		1
2	0	7	2015	27		1
3	0	13	2015	27		1
4	0	14	2015	27		1

5 rows x 226 columns

```
from sklearn.preprocessing import MinMaxScaler, StandardScaler, Normalizer
sc1 = StandardScaler()
sc1_data = sc1.fit_transform(data[['lead_time']])
plt.hist(sc1_data, 50)
plt.show()
```

