**Московский государственный технический**

**университет им. Н. Э. Баумана**

**Факультет «Информатика и системы управления»**

Кафедра «Системы обработки информации и управления»

Курс «Технологии машинного обучения»

Отчет по лабораторной работе №3

Обработка пропусков в данных, кодирование категориальных признаков, масштабирование данных

Группа: ИУ5-62Б

Студент: Селедкина А.С.

Преподаватель: Гапанюк Ю.Е.

Москва, 2020 г.

**Цель лабораторной работы:** изучение способов предварительной обработки данных для дальнейшего формирования моделей.

**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 примеры выполнения**

**import** numpy **as** np

**import** pandas **as** pd

pd.set\_option('display.max.columns', 100)

**import** seaborn **as** sns

**import** matplotlib.pyplot **as** plt

%matplotlib inline

sns.set(style="ticks")

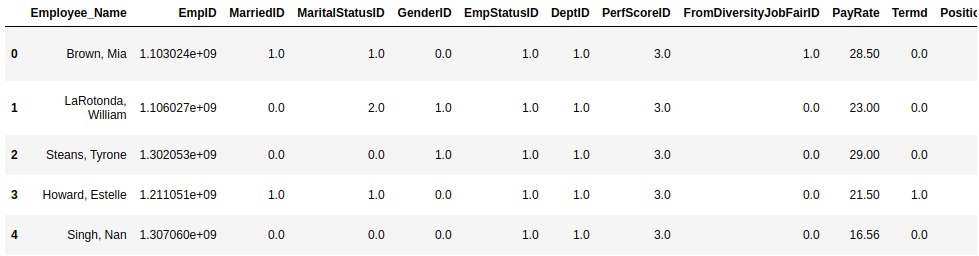
**from** sklearn.datasets **import** \*

data = pd.read\_csv('data/HRDataset\_v13.csv')

data.shape

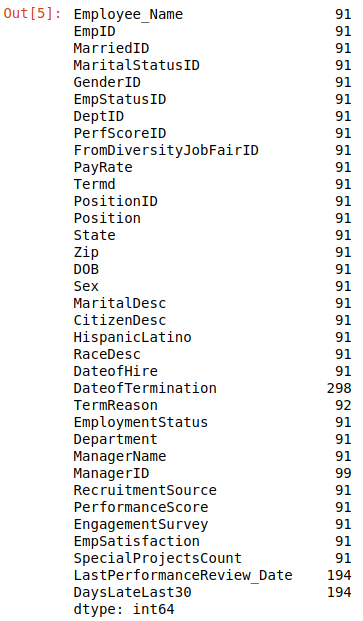


data.head()



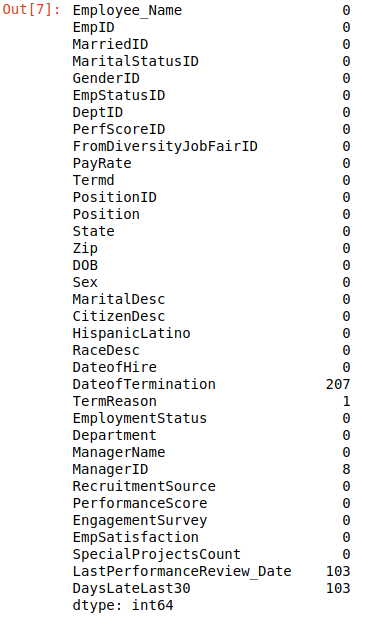
**Обработка пропусков в данных**

data.isnull().sum()



data1 = data.dropna(axis=0, how='all')

data1.isnull().sum()



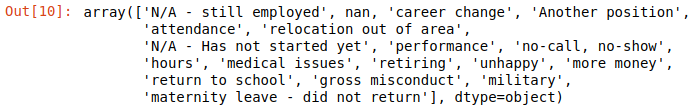
data1.shape



data1['TermReason'].dtype



data1['TermReason'].unique()

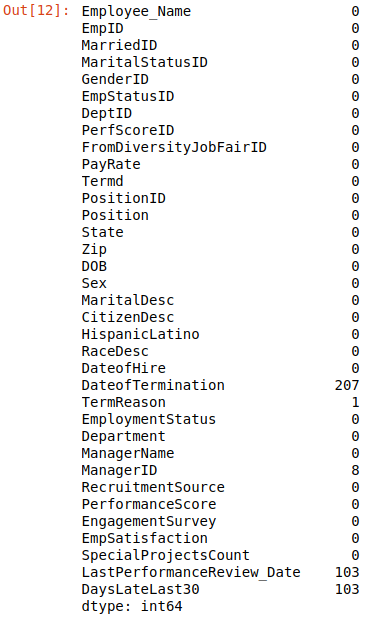


**from** sklearn.impute **import** SimpleImputer

imp = SimpleImputer(missing\_values=np.nan, strategy='most\_frequent')

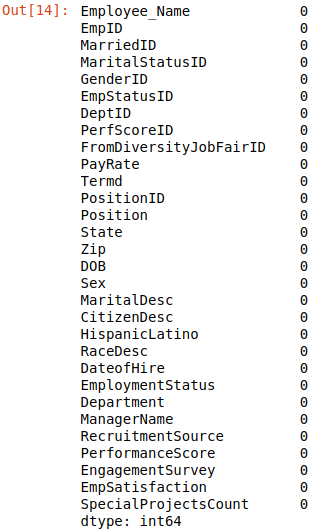
data\_imp = imp.fit(data1[['TermReason']])

data1.isnull().sum()



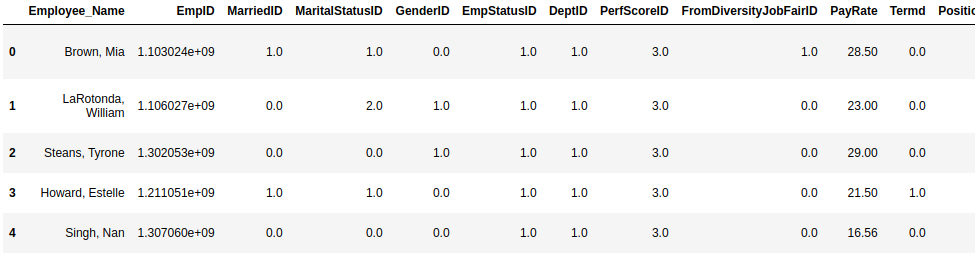
data2 = data1.dropna(axis=1, how='any')

data2.isnull().sum()



**Кодирование категориальных признаков**

data2.head()



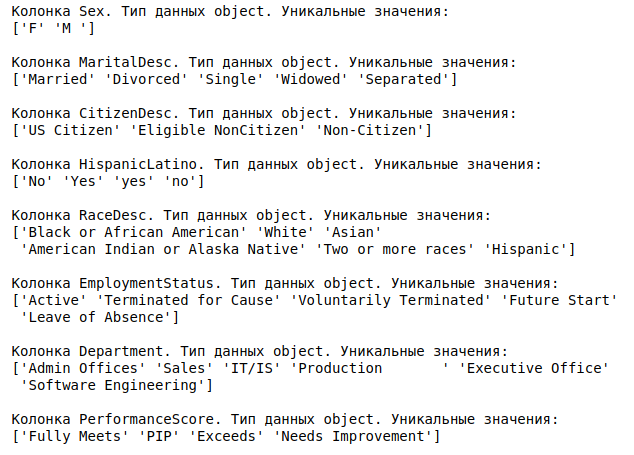
for col in data2.columns:

dt = str(data2[col].dtype)

if dt=='object' and len(data2[col].unique()) < 10:

print('Колонка {}. Тип данных {}. Уникальные

значения:\n{}\n'.format(col, dt, data2[col].unique()))



**from** sklearn.preprocessing **import** LabelEncoder

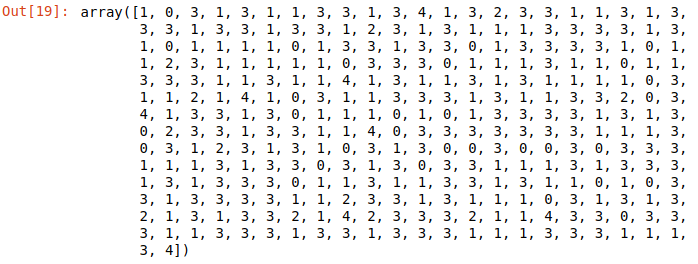
le = LabelEncoder()

data\_le = le.fit\_transform(data2['MaritalDesc'])

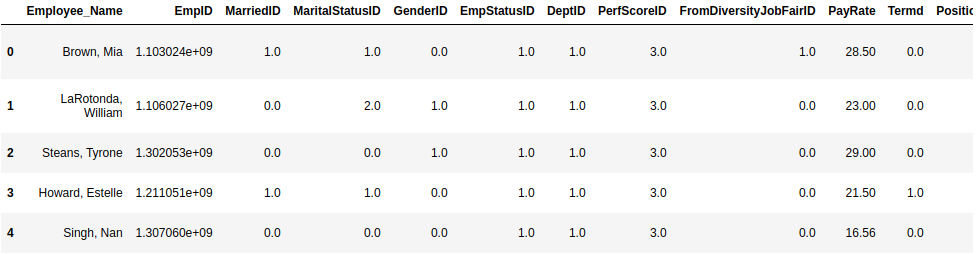
np.unique(data\_le)



data\_le



data2.head()



**Масштабирование данных**

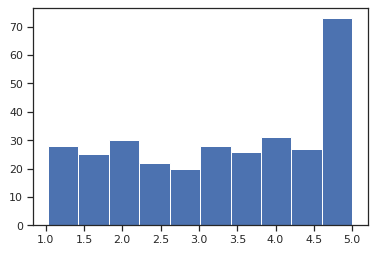
**from** sklearn.preprocessing **import** MinMaxScaler

sc = MinMaxScaler()

sc\_data = sc.fit\_transform(data2[['EngagementSurvey']])

plt.hist(data2['EngagementSurvey'], 10)

plt.show()



plt.hist(sc\_data, 10)

plt.show()

