МГТУ им. Н. Э. Баумана

кафедра ИУ5

курс «Технологии машинного обучения»

Лабораторная работа №1

«Разведочный анализ данных. Исследование и визуализация данных»

ВЫПОЛНИЛ:

Болгова А. В.

Группа ИУ5-61Б

ПРОВЕРИЛ:

Гапанюк Ю. Е.

Москва, 2020 г.

Цель лабораторной работы: изучение различных методов визуализации данных.

Краткое описание: Построение основных графиков, входящих в этап разведочного анализа данных. Корреляционный анализ данных. Формирование выводов о возможности построения моделей машинного обучения и о возможном вкладе признаков в модель.

Задание:

Выбрать набор данных (датасет). Создать ноутбук, который содержит следующие разделы:

1. Текстовое описание выбранного набора данных
2. Основные характеристики датасета
3. Визуальное исследование датасета. Необходимо использовать не менее 2 различных библиотек и не менее 5 графиков.
4. Информация о корреляции признаков.

Сформировать отчет и разместить его в репозитории на github.

Выбор набора данных:

Источник:
https://www.kaggle.com/hugoncosta/price-of-flats-in-moscow

Следующий набор данных дает небольшую выборку цен на квартиры в Москве.

Показатели:

price
цена квартиры в $1000
totsp
общая площадь квартиры, кв.м.
livesp
жилая площадь квартиры, кв.м.
kitsp
площадь кухни, кв.м.
dist
расстояние от центра в км.
metrdist
расстояние до метро в минутах
walk
1 – пешком от метро, 0 – на транспорте
brick
1 – кирпичный, монолит ж/б, 0 – другой
floor

1 – этаж кроме первого и последнего, 0 – иначе.
code
число от 1 до 8, при помощи которого мы группируем наблюдения по подвыборкам: 1. Наблюдения сгруппированы на севере, вокруг Калужско-Рижской линии метрополитена 2. Север, вокруг Серпуховско-Тимирязевской линии метрополитена 3. Северо-запад, вокруг Замоскворецкой линии метрополитена 4. Северо-запад, вокруг Таганско-Краснопресненской линии метрополитена 5. Юго-восток, вокруг Люблинской линии метрополитена 6. Юго-восток, вокруг Таганско-Краснопресненской линии метрополитена 7. Восток, вокруг Калининской линии метрополитена 8. Восток, вокруг Арбатско-Покровской линии метрополитена

Выполнение работы

1. Загрузка и первичная обработка данных

In [171]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
```

In [172]:
```
data = pd.read_csv('../data/flats_moscow.csv', sep=",", engine='python')
```

In [173]:
```
# Список колонок с типами данных
data.dtypes
```

Out[173]:
```
Unnamed: 0      int64
price           int64
totsp           int64
livesp          int64
kitsp         float64
dist          float64
metrdist        int64
walk            int64
brick           int64
floor           int64
code            int64
dtype: object
```

In [174]:
```
data.head()
```

Out[174]:

	Unnamed: 0	price	totsp	livesp	kitsp	dist	metrdist	walk	brick	floor	code
0	1	81	58	40	6.0	12.5	7	1	1	1	3
1	2	75	44	28	6.0	13.5	7	1	0	1	6
2	3	128	70	42	6.0	14.5	3	1	1	1	3
3	4	95	61	37	6.0	13.5	7	1	0	1	1
4	5	330	104	60	11.0	10.5	7	0	1	1	3

In [175]:
```
# Проверка на пропуски
data.isnull().sum()
```

Out[175]:
```
Unnamed: 0    0
price         0
totsp         0
livesp        0
kitsp         0
dist          0
metrdist      0
walk          0
brick         0
floor         0
code          0
dtype: int64
```

=> Пустых значений в наборе данных нет.

In [176]:
```
# Размер набора данных: (строки, колонки)
data.shape
```

Out[176]:
```
(2040, 11)
```

In [177]:
```
total_count = data.shape[0]
print('Всего строк: {}'.format(total_count))
```

```
Всего строк: 2040
```

In [178]:
```
# Основные статистические характеристики набора данных
data.describe()
```

Out[178]:

	Unnamed: 0	price	totsp	livesp	kitsp	dist	metrdist	walk	brick	floor	code
count	2040.000000	2040.000000	2040.000000	2040.000000	2040.000000	2040.000000	2040.000000	2040.000000	2040.000000	2040.000000	2040.000000
mean	1020.500000	127.496569	73.084314	46.337255	8.898529	11.015686	8.117157	0.685784	0.323039	0.790686	4.322059
std	589.041594	51.878220	15.123450	7.894348	2.787073	3.375539	3.815574	0.464317	0.467752	0.406918	2.183289
min	1.000000	50.000000	44.000000	28.000000	5.000000	3.000000	1.000000	0.000000	0.000000	0.000000	1.000000
25%	510.750000	95.000000	62.000000	42.000000	7.000000	9.000000	5.000000	0.000000	0.000000	1.000000	3.000000
50%	1020.500000	115.000000	73.500000	45.000000	9.000000	12.000000	7.000000	1.000000	0.000000	1.000000	4.000000
75%	1530.250000	142.000000	79.000000	50.000000	10.000000	13.500000	10.000000	1.000000	1.000000	1.000000	6.000000
max	2040.000000	730.000000	192.000000	102.000000	25.000000	17.000000	20.000000	1.000000	1.000000	1.000000	8.000000

Выделим индексы основных регионов:

In [179]:
```
data['code'].unique()
```

Out[179]:
```
array([3, 6, 1, 8, 4, 5, 7, 2], dtype=int64)
```

2. Визуальное исследование датасета

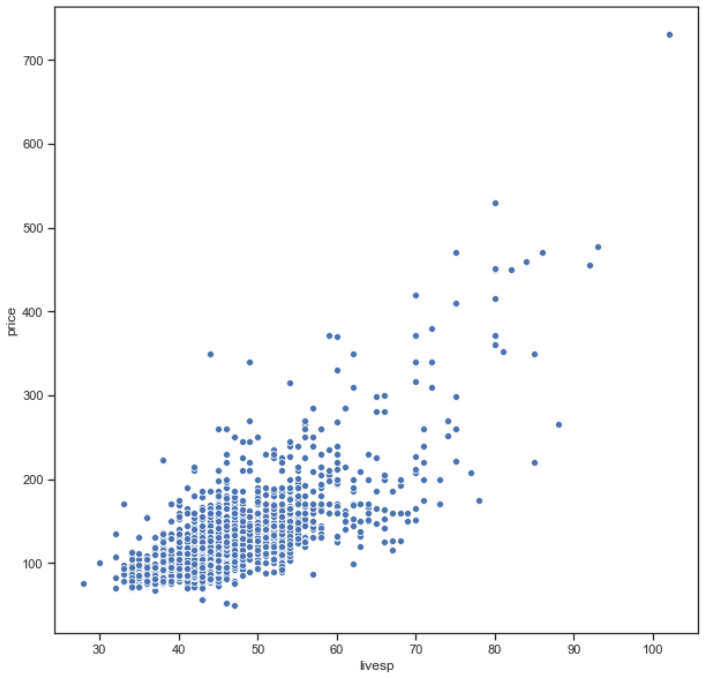
Исследование набора данных будет проводиться при помощи диаграмм рассеивания и гистограмм.

С помощью диаграммы рассеивания можно оценить, коррелируют ли между собой две выбранные переменные.

Исследуем зависимость цены (price) от жилой площади квартиры (livesp)

In [180]:
```
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x='livesp', y='price', data=data)
```

Out[180]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x131980d0>
```

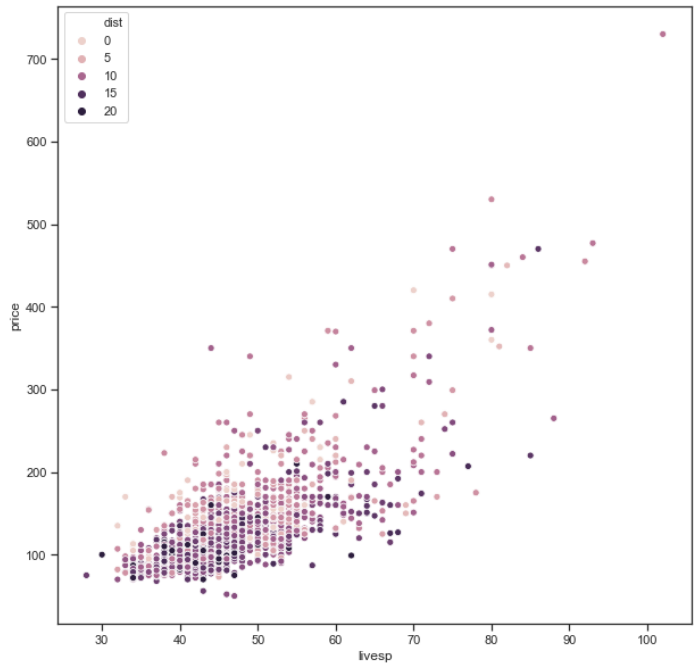


Как видно, зависимость имеет характер, близкий к экспоненциальному.

Рассмотрим еще и влияние третьего параметра – расстояния от центра до квартиры (dist).

```
In [181]: fighue, ax = plt.subplots(figsize=(10,10))
          sns.scatterplot(ax=ax, x='livesp', y='price', data=data, hue='dist')
```

```
Out[181]: <matplotlib.axes._subplots.AxesSubplot at 0x131a29f0>
```



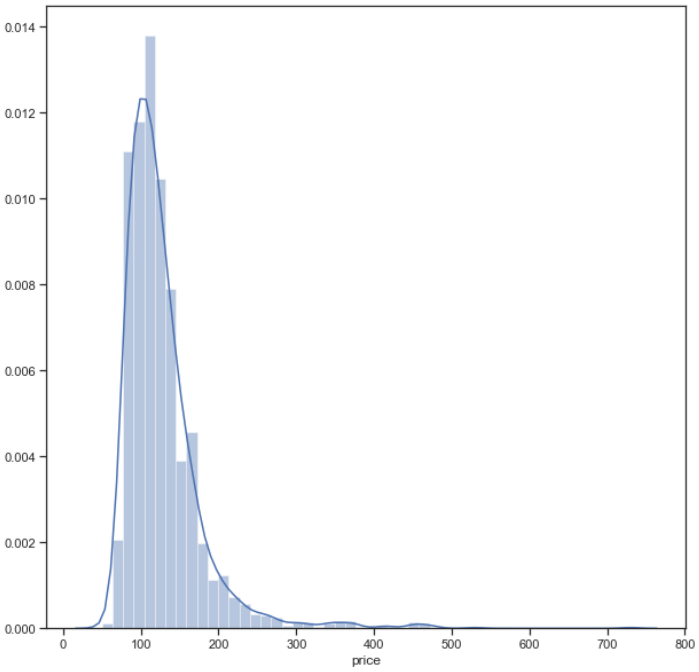
Из графика видно, что в целом наблюдается слабая, неявная зависимость цены от расстояния от центра.

Еще одним способом визуального исследования данных является представление показателей в виде гистограммы. С помощью гистограммы можно оценить плотность вероятности распределения данных.

Исследуем вероятность для поля цена (price):

```
In [182]: figdist, ax = plt.subplots(figsize=(10,10))
          sns.distplot(data['price'])
```

```
Out[182]: <matplotlib.axes._subplots.AxesSubplot at 0x139b9410>
```

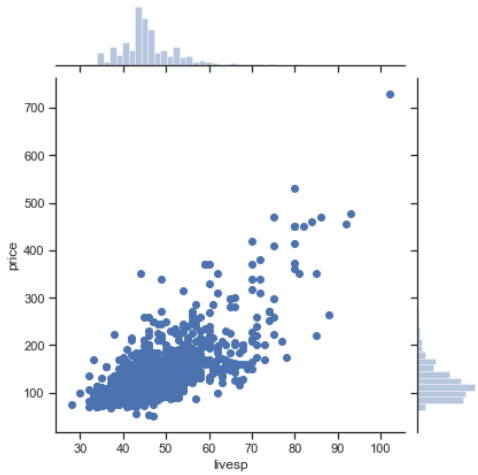


Полученные результаты практически совпадают с нормальным распределением (распределением Гаусса).

Для наглядности можно построить комбинированный график с гистограммой и диаграммой рассеяния:

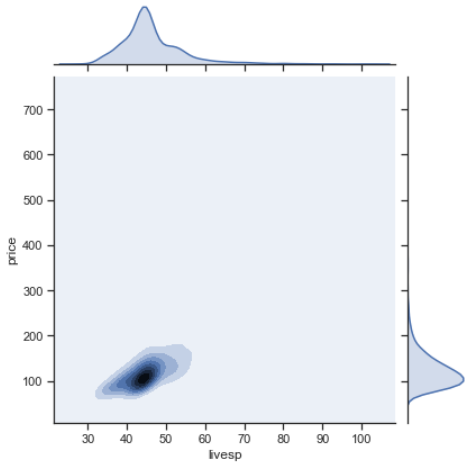
```
In [183]: sns.jointplot(x='livesp', y='price', data=data)
```

```
Out[183]: <seaborn.axisgrid.JointGrid at 0x13392910>
```



```
In [188]: sns.jointplot(x='livesp', y='price', data=data, kind="kde")

Out[188]: <seaborn.axisgrid.JointGrid at 0x18616d70>
```



3. Исследование корреляции признаков

Проверка корреляции признаков позволяет решить две задачи:

1. Понять, какие признаки наиболее сильно коррелируют с целевым признаком. Именно эти признаки будут наиболее информативными для моделей машинного обучения. Признаки, которые слабо коррелируют с целевым признаком, можно попробовать исключить из построения модели, иногда это повышает качество модели. Некоторые алгоритмы машинного обучения автоматически определяют ценность того или иного признака для построения модели.

2. Понять, какие нецелевые признаки линейно зависимы между собой. Линейно зависимые признаки, как правило, очень плохо влияют на качество моделей. Поэтому если несколько признаков линейно зависимы, то для построения модели из них выбирают какой-то один признак.

Существуют разные алгоритмы построения корреляционной матрицы:

```
In [184]: data.corr(method='pearson')

Out[184]:
```

	Unnamed: 0	price	totsp	livesp	kitsp	dist	metrdist	walk	brick	floor	code
Unnamed: 0	1.000000	-0.022761	-0.036063	-0.031432	-0.029838	0.029367	0.007293	-0.010263	-0.034944	0.016642	-0.020986
price	-0.022761	1.000000	0.756042	0.729614	0.597235	-0.331563	-0.152116	0.151996	0.260835	0.132564	-0.089623
totsp	-0.036063	0.756042	1.000000	0.862236	0.781536	-0.114698	-0.041426	0.011876	0.121287	0.113723	-0.021959
livesp	-0.031432	0.729614	0.862236	1.000000	0.573528	-0.197228	-0.052112	0.060367	0.254309	0.094658	-0.005480
kitsp	-0.029838	0.597235	0.781536	0.573528	1.000000	-0.061874	-0.028490	-0.009112	-0.019235	0.117050	-0.054632
dist	0.029367	-0.331563	-0.114698	-0.197228	-0.061874	1.000000	0.099185	-0.175277	-0.394742	0.020530	-0.191975
metrdist	0.007293	-0.152116	-0.041426	-0.052112	-0.028490	0.099185	1.000000	-0.040667	-0.066557	-0.021787	-0.001882
walk	-0.010263	0.151996	0.011876	0.060367	-0.009112	-0.175277	-0.040667	1.000000	0.153708	-0.021207	-0.035588
brick	-0.034944	0.260835	0.121287	0.254309	-0.019235	-0.394742	-0.066557	0.153708	1.000000	-0.051694	0.052712
floor	0.016642	0.132564	0.113723	0.094658	0.117050	0.020530	-0.021787	-0.021207	-0.051694	1.000000	-0.049948
code	-0.020986	-0.089623	-0.021959	-0.005480	-0.054632	-0.191975	-0.001882	-0.035588	0.052712	-0.049948	1.000000

In [185]:
```
data.corr(method='kendall')
```

Out[185]:

	Unnamed: 0	price	totsp	livesp	kitsp	dist	metrdist	walk	brick	floor	code
Unnamed: 0	1.000000	-0.016284	-0.028642	-0.027992	-0.019367	0.023070	0.005220	-0.008382	-0.028539	0.013592	-0.014651
price	-0.016284	1.000000	0.590742	0.506202	0.494208	-0.255946	-0.111954	0.144860	0.224930	0.129073	-0.062721
totsp	-0.028642	0.590742	1.000000	0.658063	0.649599	-0.079131	-0.008968	0.003467	0.082351	0.100711	-0.001891
livesp	-0.027992	0.506202	0.658063	1.000000	0.405436	-0.142573	-0.019362	0.046134	0.200027	0.075247	0.006572
kitsp	-0.019367	0.494208	0.649599	0.405436	1.000000	-0.019973	-0.001230	-0.026902	-0.055019	0.104325	-0.030473
dist	0.023070	-0.255946	-0.079131	-0.142573	-0.019973	1.000000	0.081346	-0.155314	-0.339736	0.025458	-0.184100
metrdist	0.005220	-0.111954	-0.008968	-0.019362	-0.001230	0.081346	1.000000	-0.042195	-0.060836	-0.024240	0.000939
walk	-0.008382	0.144860	0.003467	0.046134	-0.026902	-0.155314	-0.042195	1.000000	0.153708	-0.021207	-0.033269
brick	-0.028539	0.224930	0.082351	0.200027	-0.055019	-0.339736	-0.060836	0.153708	1.000000	-0.051694	0.040259
floor	0.013592	0.129073	0.100711	0.075247	0.104325	0.025458	-0.024240	-0.021207	-0.051694	1.000000	-0.042066
code	-0.014651	-0.062721	-0.001891	0.006572	-0.030473	-0.184100	0.000939	-0.033269	0.040259	-0.042066	1.000000

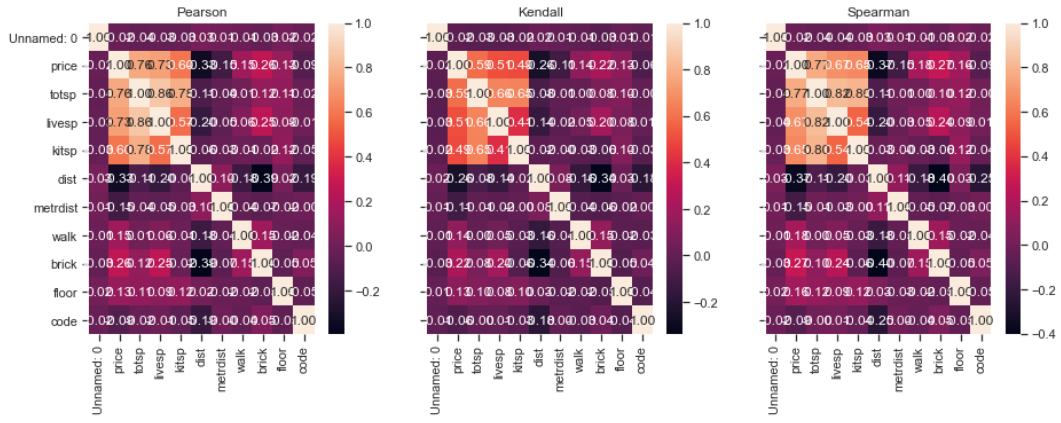
In [186]:
```
data.corr(method='spearman')
```

Out[186]:

	Unnamed: 0	price	totsp	livesp	kitsp	dist	metrdist	walk	brick	floor	code
Unnamed: 0	1.000000	-0.024281	-0.042338	-0.041216	-0.027364	0.033723	0.007006	-0.010263	-0.034944	0.016642	-0.020483
price	-0.024281	1.000000	0.768785	0.671953	0.651975	-0.366695	-0.153349	0.176057	0.273369	0.156869	-0.086993
totsp	-0.042338	0.768785	1.000000	0.819544	0.801094	-0.114953	-0.012387	0.004185	0.099418	0.121583	-0.002355
livesp	-0.041216	0.671953	0.819544	1.000000	0.542124	-0.199267	-0.026200	0.054960	0.238296	0.089643	0.010491
kitsp	-0.027364	0.651975	0.801094	0.542124	1.000000	-0.028054	-0.001287	-0.030900	-0.063193	0.119827	-0.040110
dist	0.033723	-0.366695	-0.114953	-0.199267	-0.028054	1.000000	0.108896	-0.184136	-0.402783	0.030183	-0.253626
metrdist	0.007006	-0.153349	-0.012387	-0.026200	-0.001287	0.108896	1.000000	-0.047204	-0.068059	-0.027118	0.000624
walk	-0.010263	0.176057	0.004185	0.054960	-0.030900	-0.184136	-0.047204	1.000000	0.153708	-0.021207	-0.038328
brick	-0.034944	0.273369	0.099418	0.238296	-0.063193	-0.402783	-0.068059	0.153708	1.000000	-0.051694	0.046382
floor	0.016642	0.156869	0.121583	0.089643	0.119827	0.030183	-0.027118	-0.021207	-0.051694	1.000000	-0.048464
code	-0.020483	-0.086993	-0.002355	0.010491	-0.040110	-0.253626	0.000624	-0.038328	0.046382	-0.048464	1.000000

Для наглядности можно составить тепловую карту:

In [187]:
```
figwarmmaps, ax = plt.subplots(1, 3, sharex='col', sharey='row', figsize=(15,5))
sns.heatmap(data.corr(method='pearson'), ax=ax[0], annot=True, fmt='.2f')
sns.heatmap(data.corr(method='kendall'), ax=ax[1], annot=True, fmt='.2f')
sns.heatmap(data.corr(method='spearman'), ax=ax[2], annot=True, fmt='.2f')
fig.suptitle('Корреляционные матрицы, построенные различными методами')
ax[0].title.set_text('Pearson')
ax[1].title.set_text('Kendall')
ax[2].title.set_text('Spearman')
```



Таким образом, можно выделить признаки, слабо влияющие на целевой признак – цену (price): регион (code), этаж (floor), brick, до метро пешком/на транспорте (walk), расстояние до метро (metrdist) и от центра (dist). В основном цена на квартиру в Москве зависит от площади комнат.