Московский государственный технический университет им. Н.Э. Баумана
Кафедра «Системы обработки информации и управления»

Лабораторная работа №1
по дисциплине
«Технологии машинного обучения»
на тему
«Разведочный анализ данных. Исследование и визуализация данных»

Выполнил:
студент группы ИУ5-63Б
Елизаров О. О.

Москва — 2020 г.

1. Цель лабораторной работы

Изучить различные методы визуализации данных

2. Задание

Требуется выполнить следующие действия

- Выбрать набор данных
- Создать ноутбук, который содержит следующие разделы:
    1. Текстовое описание выбранного наборы данных
    2. Основные характеристики датасета
    3. Визуальное исследование датасета
    4. Информация о корелляции признаков
- Сформировать отчет и разместить его на своем репозитории GitHub

3. Ход выполнения лабораторной работы

3.1. Текстовое описание набора данных

Данный набор данных представляет собой перечень всех переходов футболистов в период с сезона 2007\2008 посезон 2016\2017.Используются такие данные,как Имя,Фамилия,Позиция,Гражданство,Клуб откуда,Клуб куда и прочие.На основе этих данных постараемся определить зависимости,такие как Самая богатая лига или самый активный сезонпо трансферам. Так же постараемся понять,когда переходят более дорогие игроки,в середине сезона или междусезонами.

Основные характеристики датасета

Подключим необходимые библиотеки

[1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import string
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(style="ticks")
```

[2]:
```
data = pd.read_csv('ansel/transfer_data.csv', sep=",")
```

[3]:
```
print(data.head())
```

```
            PLAYER      WINDOW    POSITION    COUNTRY              FROM  \
0       Paul Pogba  Pre-Season  Midfielder     France          Juventus
1     Gareth Bale  Pre-Season  Midfielder      Wales         Tottenham
2     Luis Suarez  Pre-Season    Attacker    Uruguay         Liverpool
3         Ronaldo  Pre-Season         NaN        NaN  Manchester United
4  Gonzalo Higuain  Pre-Season    Attacker  Argentina            Napoli
```

```
                    TO DESCRIPTION        PRICE   LEAGUE SEASON
0  Manchester United        Sold  105000000.0      EPL  16/17
1        Real Madrid        Sold  100000000.0  La Liga  13/14
2          Barcelona        Sold   95000000.0  La Liga  14/15
3        Real Madrid        Sold   93900000.0  La Liga  09/10
4           Juventus        Sold   90000000.0  Serie A  16/17
```

[4]:
```
data.shape
```

[4]:
```
(6237, 10)
```

[5]:
```
data.columns
```

[5]:
```
Index(['PLAYER', 'WINDOW', 'POSITION', 'COUNTRY', 'FROM', 'TO',␣
 ↪'DESCRIPTION',
       'PRICE', 'LEAGUE', 'SEASON'],
      dtype='object')
```

[6]:
```
del data["POSITION"]
del data["COUNTRY"]
```

[7]:
```
for col in data.columns:
    #                   -
    temp_null_count = data[data[col].isnull()].shape[0]
    print('{} - {}'.format(col, temp_null_count))
data = data.drop(data[data["SEASON"] == 'nan'].index)
data=data.sort_values('SEASON')
```

```
PLAYER - 1
WINDOW - 1
FROM - 2
TO - 1
DESCRIPTION - 1
PRICE - 1
LEAGUE - 1
SEASON - 1
```

[ ]:
```

```

[8]:
```
data.dropna()
```

[8]:
```
                PLAYER      WINDOW               FROM                 TO  \
1122          Pongolle  Pre-Season          Liverpool         Recreativo
4727        L. Vigiani  Pre-Season            Livorno            Reggina
4726      C. Puggioni   Pre-Season        Pisa Calcio            Reggina
4725            Bianco  Pre-Season            Catania            Reggina
4724     N. Novakovic   Pre-Season             Odense            Reggina
…                …          …                   …                  …
2838  Ashley Fletcher   Pre-Season  Manchester United           West Ham
```

```
2837  Hal Robson-Kanu  Pre-Season           Reading          West Brom
2836  Adrian Mariappa  Pre-Season     Crystal Palace            Watford
3084    Oliver Torres  Pre-Season    Atletico Madrid             Porto
0          Paul Pogba  Pre-Season           Juventus  Manchester United

     DESCRIPTION        PRICE   LEAGUE SEASON
1122        Sold    4000000.0  La Liga  07/08
4727        Free          0.0  Serie A  07/08
4726        Free          0.0  Serie A  07/08
4725        Free          0.0  Serie A  07/08
4724        Free          0.0  Serie A  07/08
…           …           …       …     …
2838        Free          0.0      EPL  16/17
2837        Free          0.0      EPL  16/17
2836        Free          0.0      EPL  16/17
3084        Loan          0.0      ROE  16/17
0           Sold  105000000.0      EPL  16/17

[6235 rows x 8 columns]
```

[9]:
```
data.describe()
```

[9]:
```
              PRICE
count  6.236000e+03
mean   2.758248e+06
std    7.312051e+06
min    0.000000e+00
25%    0.000000e+00
50%    0.000000e+00
75%    2.000000e+06
max    1.050000e+08
```

[10]:
```
dataseas=data.groupby("LEAGUE")
```

[11]:
```
seasons=data["SEASON"].unique()[:len(data["SEASON"].unique())-1]
# for elem,ind in (seasons):
#     elem.replace("/","!")
#     print(type(elem))
seasonL=sorted(seasons, key=lambda x:(x==0, x))
print("res:",seasonL)
```

```
res: ['07/08', '08/09', '09/10', '10/11', '11/12', '12/13', '13/14', '14/15',
'15', '15/16', '16', '16/17']
```

[12]:
```
i=0
sum=[]
names=[]
while(i<len(data["LEAGUE"].unique())-1):
    sum.append(dataseas.get_group(data["LEAGUE"].unique()[i])["PRICE"].
 ↪max())
```

```
    names.append(data["LEAGUE"].unique()[i])
#     print(data["LEAGUE"].unique()[i])
    i+=1
print(sum)
print(names)
```

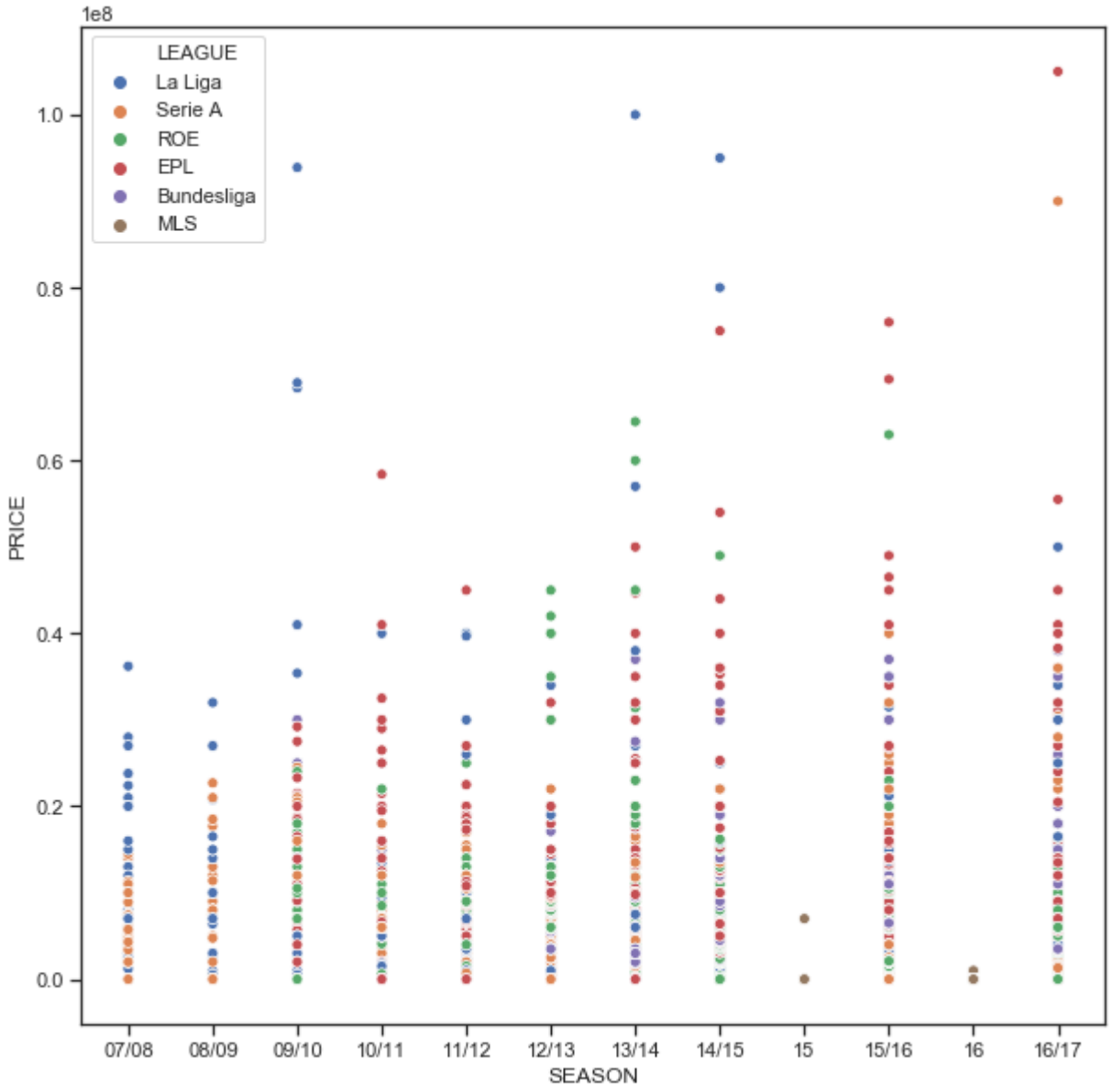
```
[100000000.0, 90000000.0, 64500000.0, 105000000.0, 40000000.0, 7000000.0]
['La Liga', 'Serie A', 'ROE', 'EPL', 'Bundesliga', 'MLS']
```

[13]:
```
len(data["LEAGUE"].unique())
```

[13]: 7

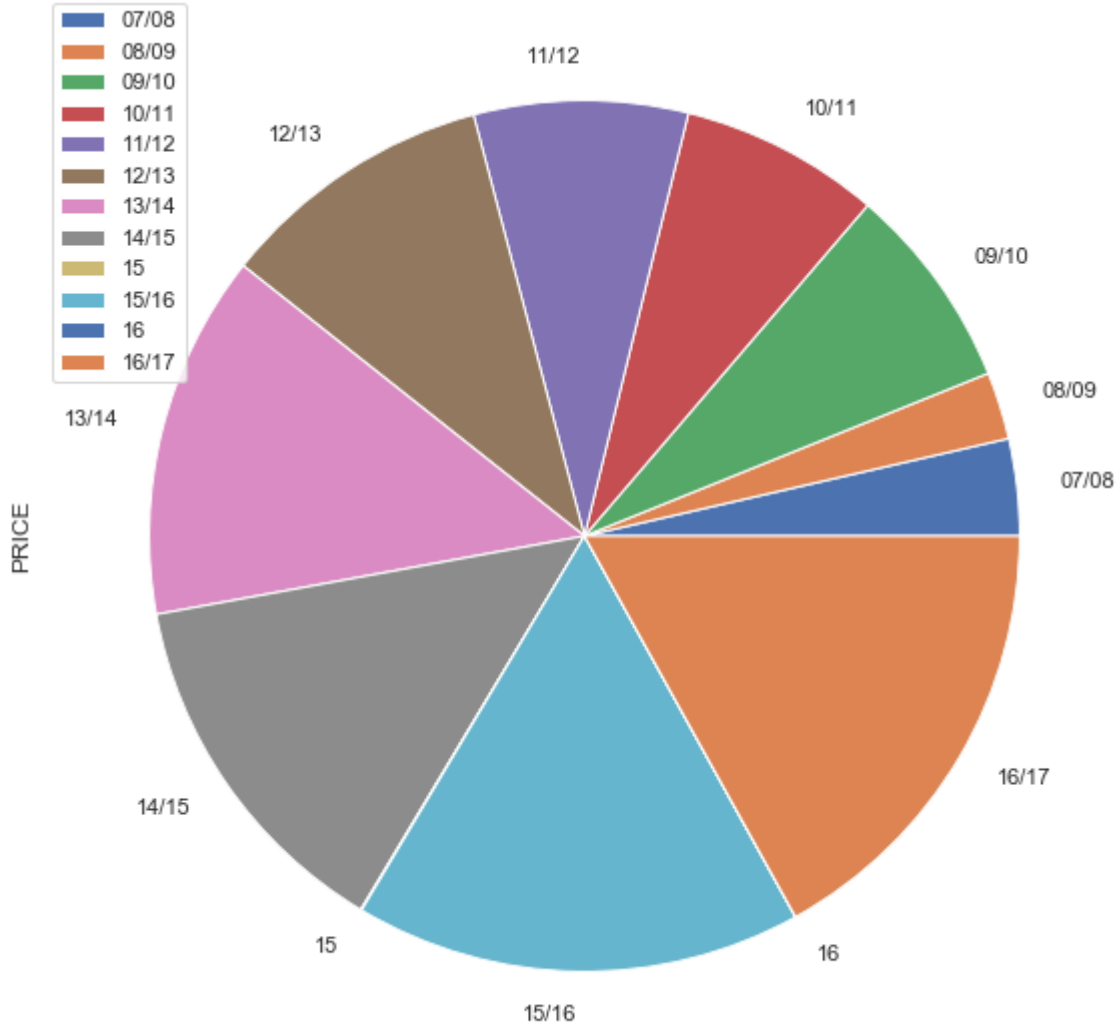
[14]:
```
fig, ax = plt.subplots(figsize=(10,10))
sns.scatterplot(ax=ax, x="SEASON", y="PRICE" ,data=data, hue="LEAGUE")
# plt.scatter(x="SEASON", y="PRICE" ,data=data, hue="LEAGUE")
```

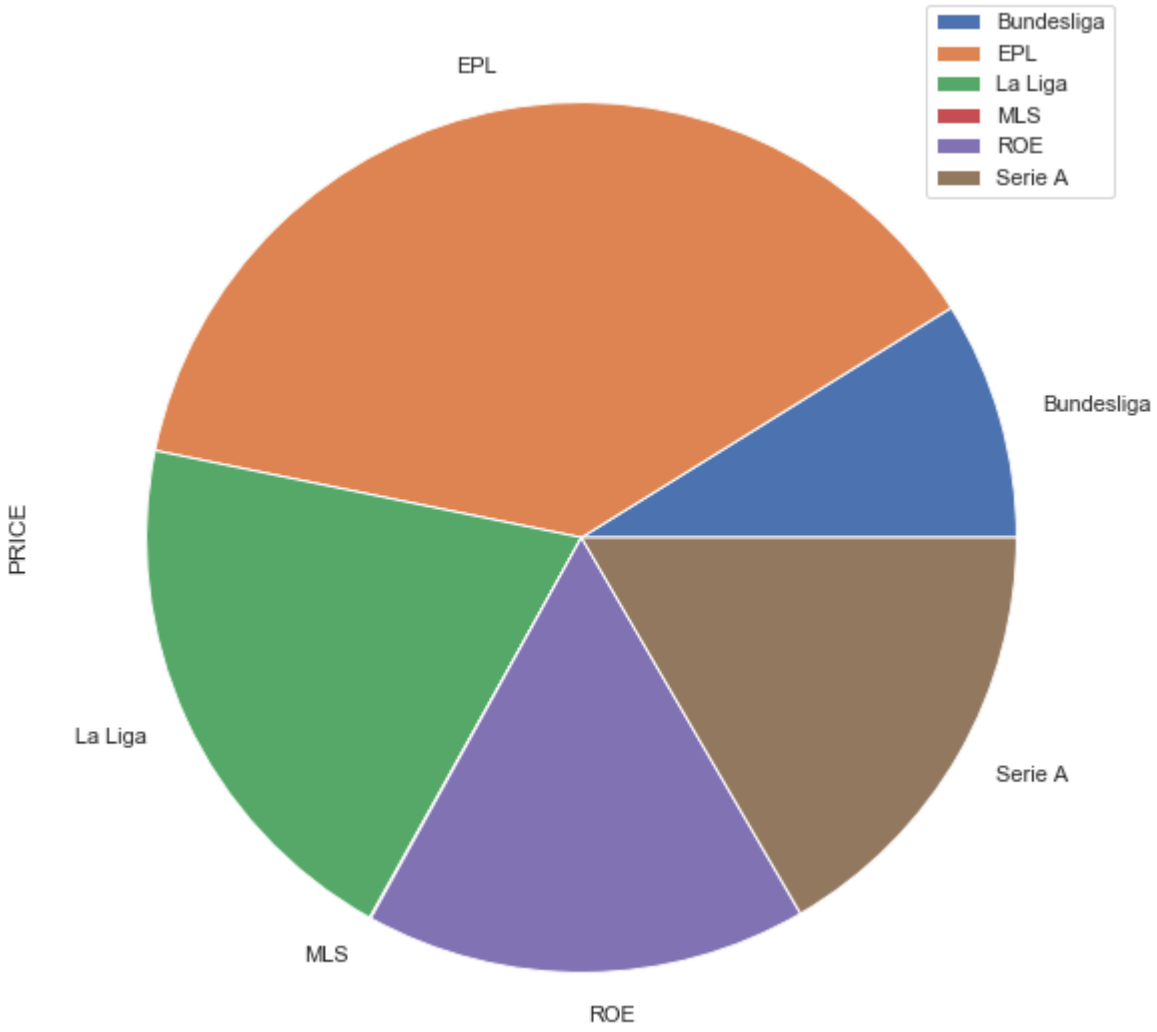
[14]: <matplotlib.axes._subplots.AxesSubplot at 0x1b20aa5e948>

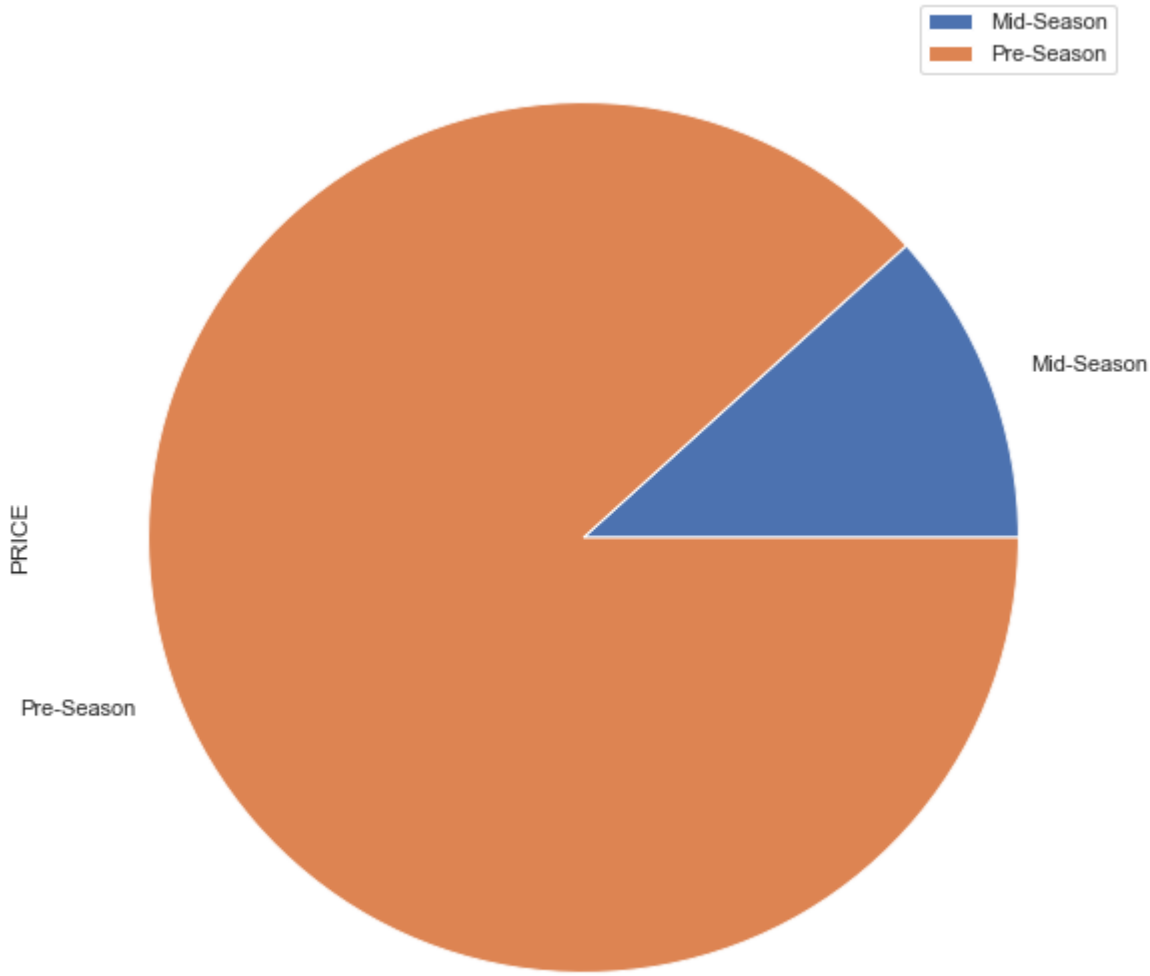


[37]:
```
data.groupby("SEASON").sum().plot.pie(y='PRICE',figsize = (10,10))
data.groupby("LEAGUE").sum().plot.pie(y='PRICE',figsize = (10,10))
data.groupby("WINDOW").sum().plot.pie(y='PRICE',figsize = (10,10))
```

[37]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1b20f610ac8>
```




Bundesliga
EPL
La Liga
MLS
ROE
Serie A
EPL
Bundesliga
PRICE
La Liga
Serie A
MLS
ROE



[26]:
```
data["PRICE"].unique()
```

[26]:
```
array([4.000e+06, 0.000e+00, 1.400e+07, 1.500e+06, 1.450e+07, 1.300e+07,
       1.200e+07, 1.700e+06, 3.000e+06, 3.200e+06, 1.200e+06, 6.000e+06,
       1.500e+07, 6.300e+06, 1.600e+07, 7.000e+06, 8.000e+06, 2.500e+06,
       7.500e+06, 9.000e+06, 2.000e+06, 8.900e+06, 1.130e+07, 7.400e+06,
       1.100e+07, 1.000e+07, 5.300e+06, 4.600e+06, 2.240e+07, 4.500e+06,
       3.620e+07, 3.700e+06, 2.800e+07, 2.100e+07, 2.000e+07, 5.000e+06,
       4.700e+06, 2.380e+07, 5.500e+06, 2.700e+07, 3.400e+06, 4.300e+06,
       5.750e+06, 3.200e+07, 1.140e+07, 4.750e+06, 2.400e+06, 9.500e+05,
       5.000e+05, 1.770e+07, 2.080e+07, 1.850e+07, 1.650e+07, 6.500e+06,
       6.400e+06, 2.270e+07, 2.900e+06, 2.850e+06, 2.800e+06, 2.300e+06,
       5.700e+06, 2.200e+06, 5.800e+06, 3.300e+06, 5.850e+06, 3.500e+06,
       4.100e+06, 3.100e+06, 6.800e+06, 4.000e+05, 1.700e+07, 3.000e+07,
       1.800e+06, 2.150e+07, 2.120e+07, 2.050e+07, 2.500e+07, 2.450e+07,
       2.400e+07, 2.330e+07, 3.000e+05, 1.870e+07, 1.800e+07, 2.000e+05,
       2.750e+07, 1.000e+06, 2.920e+07, 1.390e+07, 8.000e+05, 4.100e+07,
       9.000e+05, 7.500e+05, 9.400e+06, 9.300e+06, 9.200e+06, 9.100e+06,
```

```
6.000e+05, 6.840e+07, 6.900e+07, 7.000e+05, 9.390e+07, 1.050e+07,
1.030e+07, 1.120e+07, 3.540e+07, 5.900e+05, 3.600e+06, 3.950e+05,
2.750e+06, 2.400e+05, 1.300e+06, 3.500e+05, 1.900e+06, 9.000e+04,
3.000e+04, 1.350e+07, 1.550e+07, 7.300e+06, 7.200e+06, 1.250e+07,
9.700e+06, 9.500e+06, 8.500e+06, 3.250e+07, 2.200e+07, 2.650e+07,
2.900e+07, 4.900e+06, 4.000e+07, 4.800e+06, 4.200e+06, 5.840e+07,
6.600e+06, 1.950e+07, 1.500e+05, 2.800e+05, 2.500e+05, 1.250e+05,
7.900e+06, 7.800e+06, 5.100e+06, 8.700e+06, 8.800e+06, 5.900e+06,
2.600e+07, 3.750e+06, 2.250e+07, 1.900e+07, 4.400e+06, 4.500e+07,
3.970e+07, 1.400e+06, 1.100e+06, 1.730e+07, 8.500e+05, 1.080e+07,
4.250e+06, 6.300e+05, 3.400e+07, 3.500e+07, 4.200e+07, 1.710e+07,
6.100e+06, 6.750e+06, 7.600e+06, 7.700e+06, 8.600e+06, 3.700e+05,
3.800e+06, 7.400e+05, 1.320e+07, 8.100e+06, 8.200e+06, 2.600e+06,
2.700e+06, 1.180e+07, 9.800e+06, 2.350e+06, 1.000e+08, 2.900e+05,
6.450e+07, 6.000e+07, 5.000e+07, 1.750e+07, 1.760e+07, 2.100e+06,
2.550e+07, 3.140e+07, 1.750e+06, 3.700e+07, 3.800e+07, 1.600e+06,
4.470e+07, 2.300e+07, 5.700e+07, 5.600e+06, 3.650e+06, 1.010e+07,
1.170e+07, 1.260e+07, 1.330e+07, 2.530e+07, 3.100e+07, 3.530e+07,
3.600e+07, 4.400e+07, 4.900e+07, 5.400e+07, 7.500e+07, 8.000e+07,
9.500e+07, 1.520e+07, 1.575e+07, 1.620e+07, 2.250e+06, 3.300e+05,
9.350e+06, 8.250e+06, 5.250e+06, 6.200e+06, 2.680e+07, 3.150e+07,
4.650e+07, 6.300e+07, 6.940e+07, 7.600e+07, 1.150e+07, 1.280e+07,
1.125e+07, 1.020e+07, 1.570e+07, 1.370e+07, 2.600e+05, 2.750e+05,
6.500e+05, 4.500e+05, 1.250e+06, 2.150e+06, 1.425e+07, 1.440e+07,
1.640e+07, 3.040e+07, 3.830e+07, 5.550e+07, 9.000e+07, 2.580e+07,
4.850e+06, 1.050e+08,       nan])
```

Вывод:

К сожалению, в данном наборе данных слишком мало челочисленных параметров , чтоб строить полноценную матрицу корреляций. Однако,мы смогли обнаружить следующие зависимости,которые очень хорошо заметны на круговой диаграмме. 1)Лигой,которая потратила больше всех денег стала-английская 2)Больше всего трансферов происходит перед сезоном,а не в середине 3)С каждым годом тратится все больше и больше денег на покупку футболистов

[30]:
```
sns.heatmap(data.corr(), annot=True, fmt='.3f')
```

[30]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1b20bb7f948>
```