

Предикативная аналитика

Задача 1

Фреймворк

Фреймворк - 'каркас', предназначенный для облегчения процесса объединения компонентов при написании программы.

Фреймворк машинного обучения

Фреймворк - 'каркас', предназначенный для облегчения процесса объединения компонентов при создании алгоритмов, кодов, программ в области машинного обучения.

Фреймворк квантового машинного обучения

Фреймворк - 'каркас', предназначенный для облегчения процесса объединения компонентов при создании алгоритмов, кодов, программ в области машинного обучения с использованием параллелизма квантовых компьютеров.

Обосновать почему задачи машинного обучения хорошо подходят для решения на квантовых компьютерах

Главное преимущество вычислений на квантовых машинах - скорость. Но ускорение происходит не по причине увеличения вычислительных мощностей на каждый процесс, а благодаря взаимодействию кубитов, за счет чего обеспечивается возможность одновременной работы нескольких процессов. Параллелизм очень хорошо себя зарекомендовал в задачах тренировки глубоких нейронных сетей. Поэтому задачи машинного обучения хорошо подходят для решения на квантовых компьютерах.

Задача 2

Задачи:

- Построить математические модели для вибрации трёх подшипников, используя очевидные и скрытые зависимости.
- Сравнить и проанализировать показания разных датчиков, подумать, как лучше их использовать.
- Обосновать использование или неиспользование каждого параметра в математической модели

Схема компрессора



Загрузка данных

In [1]:

```
import pandas as pd
```

In [2]:

```
df = pd.read_excel('Data.xlsx')
```

Просмотр данных

In [3]:

df.head(5)

Out[3]:

	Дата и время	Обороты 1	Обороты 2	Обороты 3	Вибрация Подшипник 1	Вибрация Подшипник 2	Вибрация Подшипник 3	Сдвиг Подшипник 1	Сдвиг Подшипник 2	Сдвиг Подшипник 3	Давле мас (к
0	2016-09-02 00:10:00	5258.662931	5258.780949	5255.826839	53.798579	28.936931	45.781854	-0.041063	-0.031354	0.004203	0.344
1	2016-09-02 00:20:00	5277.254228	5277.860518	5274.922677	54.173565	28.835001	45.525913	-0.043171	-0.031549	0.000496	0.344
2	2016-09-02 00:30:00	5331.892912	5332.584639	5329.195142	55.400938	28.720065	45.859280	-0.040885	-0.032023	-0.000016	0.344
3	2016-09-02 00:40:00	5330.082210	5329.813639	5327.209481	56.015260	28.387437	46.192262	-0.039666	-0.031140	0.000731	0.344
4	2016-09-02 00:50:00	5296.671570	5296.561670	5294.262691	54.256911	28.309788	46.751371	-0.037788	-0.031344	0.000329	0.344

In [4]:

df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11004 entries, 0 to 11003
Data columns (total 17 columns):
#   Column                                     Non-Null Count  Dtype
---  -
0   Дата и время                             11004 non-null  datetime64[ns]
1   Обороты 1                               11004 non-null  float64
2   Обороты 2                               11004 non-null  float64
3   Обороты 3                               11004 non-null  float64
4   Вибрация Подшипник 1                    11004 non-null  float64
5   Вибрация Подшипник 2                    11004 non-null  float64
6   Вибрация Подшипник 3                    11004 non-null  float64
7   Сдвиг Подшипник 1                       11004 non-null  float64
8   Сдвиг Подшипник 2                       11004 non-null  float64
9   Сдвиг Подшипник 3                       11004 non-null  float64
10  Давление масла 1 (кПа)                   11004 non-null  float64
11  Давление масла 2 (кПа)                   11004 non-null  float64
12  Температура масла (С)                    11004 non-null  float64
13  Температура газа На входе компрессора    11004 non-null  float64
14  Температура газа На выходе компрессора   11004 non-null  float64
15  Давление газа На входе компрессора (кПа) 11004 non-null  float64
16  Давление газа На выходе компрессора (МПа) 11004 non-null  float64
dtypes: datetime64[ns](1), float64(16)
memory usage: 1.4 MB
```

Проверка на пропуски в данных

In [5]:

```
df.isnull().sum()
```

Out[5]:

```
Дата и время          0
Обороты 1             0
Обороты 2             0
Обороты 3             0
Вибрация Подшипник 1  0
Вибрация Подшипник 2  0
Вибрация Подшипник 3  0
Сдвиг Подшипник 1     0
Сдвиг Подшипник 2     0
Сдвиг Подшипник 3     0
Давление масла 1 (кПа) 0
Давление масла 2 (кПа) 0
Температура масла (С)  0
Температура газа На входе компрессора 0
Температура газа На выходе компрессора 0
Давление газа На входе компрессора (кПа) 0
Давление газа На выходе компрессора (МПа) 0
dtype: int64
```

Пропусков в данных нет

Описательная статистика

In [6]:

```
df.describe()
```

Out[6]:

	Обороты 1	Обороты 2	Обороты 3	Вибрация Подшипник 1	Вибрация Подшипник 2	Вибрация Подшипник 3	Сдвиг Подшипник 1	Сдвиг Подшипник 2	Сдвиг Подшипник 3
count	11004.000000	11004.000000	11004.000000	11004.000000	11004.000000	11004.000000	11004.000000	11004.000000	11004.000000
mean	5552.085572	5551.951317	5549.390565	65.103002	29.033050	46.199555	-0.020231	-0.030812	-0.009298
std	178.874828	178.841944	178.830382	6.275282	0.617651	0.655033	0.018524	0.002764	0.006698
min	5000.158690	5000.626625	4998.185228	47.533153	27.287129	44.711577	-0.082733	-0.038288	-0.026793
25%	5457.742330	5457.586679	5455.275482	61.234763	28.612512	45.693822	-0.032853	-0.032643	-0.013994
50%	5595.420369	5595.210789	5592.671716	65.490397	29.057773	46.186144	-0.021274	-0.030934	-0.008969
75%	5693.970754	5693.800836	5691.240462	70.270891	29.443051	46.599198	-0.006993	-0.028844	-0.004460
max	5852.693693	5852.791303	5849.808789	77.482260	30.973721	49.719864	0.023616	-0.022938	0.009361

Списиок названий столбцов

In [7]:

```
list(df)
```

Out[7]:

```
['Дата и время',
 'Обороты 1',
 'Обороты 2',
 'Обороты 3',
 'Вибрация Подшипник 1',
 'Вибрация Подшипник 2',
 'Вибрация Подшипник 3',
 'Сдвиг Подшипник 1',
 'Сдвиг Подшипник 2',
 'Сдвиг Подшипник 3',
 'Давление масла 1 (кПа)',
 'Давление масла 2 (кПа)',
 'Температура масла (С)',
 'Температура газа На входе компрессора',
 'Температура газа На выходе компрессора',
 'Давление газа На входе компрессора (кПа)',
 'Давление газа На выходе компрессора (МПа)']
```

Выделение пространства ответов и признаков в отдельные data frame

In [8]:

```
df_Y = df[['Вибрация Подшипник 1',
            'Вибрация Подшипник 2',
            'Вибрация Подшипник 3']]
df_Y.head(3)
```

Out[8]:

	Вибрация Подшипник 1	Вибрация Подшипник 2	Вибрация Подшипник 3
0	53.798579	28.936931	45.781854
1	54.173565	28.835001	45.525913
2	55.400938	28.720065	45.859280

In [9]:

```
df_X = df[['Дата и время',
            'Обороты 1',
            'Обороты 2',
            'Обороты 3',
            'Сдвиг Подшипник 1',
            'Сдвиг Подшипник 2',
            'Сдвиг Подшипник 3',
            'Давление масла 1 (кПа)',
            'Давление масла 2 (кПа)',
            'Температура масла (C)',
            'Температура газа На входе компрессора',
            'Температура газа На выходе компрессора',
            'Давление газа На входе компрессора (кПа)',
            'Давление газа На выходе компрессора (МПа)']]
df_X.head(3)
```

Out[9]:

	Дата и время	Обороты 1	Обороты 2	Обороты 3	Сдвиг Подшипник 1	Сдвиг Подшипник 2	Сдвиг Подшипник 3	Давление масла 1 (кПа)	Давление масла 2 (кПа)	Температура масла (C)	Температура газа На входе компрессора	Температура газа На выходе компрессора	Давление газа На входе компрессора (кПа)	Давление газа На выходе компрессора (МПа)
0	2016-09-02 00:10:00	5258.662931	5258.780949	5255.826839	-0.041063	-0.031354	0.004203	0.344716	0.340869	35.467580	23.1	23.2	0.344712	0.344712
1	2016-09-02 00:20:00	5277.254228	5277.860518	5274.922677	-0.043171	-0.031549	0.000496	0.344712	0.340852	35.456034	23.1	23.2	0.344712	0.344712
2	2016-09-02 00:30:00	5331.892912	5332.584639	5329.195142	-0.040885	-0.032023	-0.000016	0.344361	0.340497	35.528870	23.1	23.2	0.344361	0.344361

Построение графиков Вибрации от времени



Матрица корреляций

In [11]:

```
import seaborn as sns
import matplotlib.pyplot as plt
```

In [12]:

```
feature = list(df_X)
feature = feature[1:]
feature
```

Out[12]:

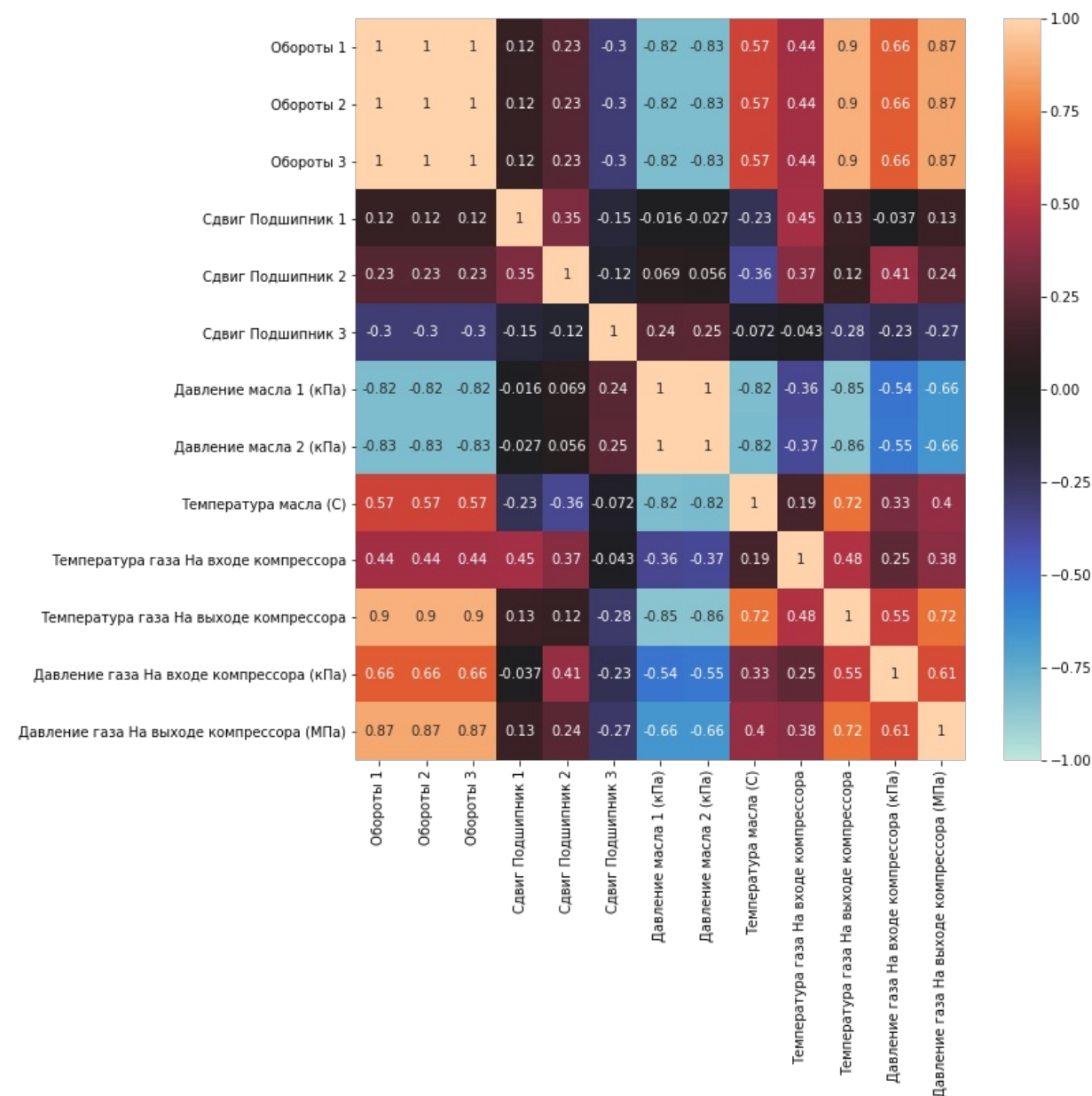
```
['Обороты 1',
 'Обороты 2',
 'Обороты 3',
 'Сдвиг Подшипник 1',
 'Сдвиг Подшипник 2',
 'Сдвиг Подшипник 3',
 'Давление масла 1 (кПа)',
 'Давление масла 2 (кПа)',
 'Температура масла (C)',
 'Температура газа На входе компрессора',
 'Температура газа На выходе компрессора',
 'Давление газа На входе компрессора (кПа)',
 'Давление газа На выходе компрессора (МПа)']
```

In [13]:

```
fig, ax = plt.subplots(figsize=(10,10))
sns.heatmap(df_X[feature].corr(), annot = True, vmin=-1, vmax=1, center= 0)
```

Out[13]:

<AxesSubplot:>



Уберем обороты 2,3 и давление масла 2

In [14]:

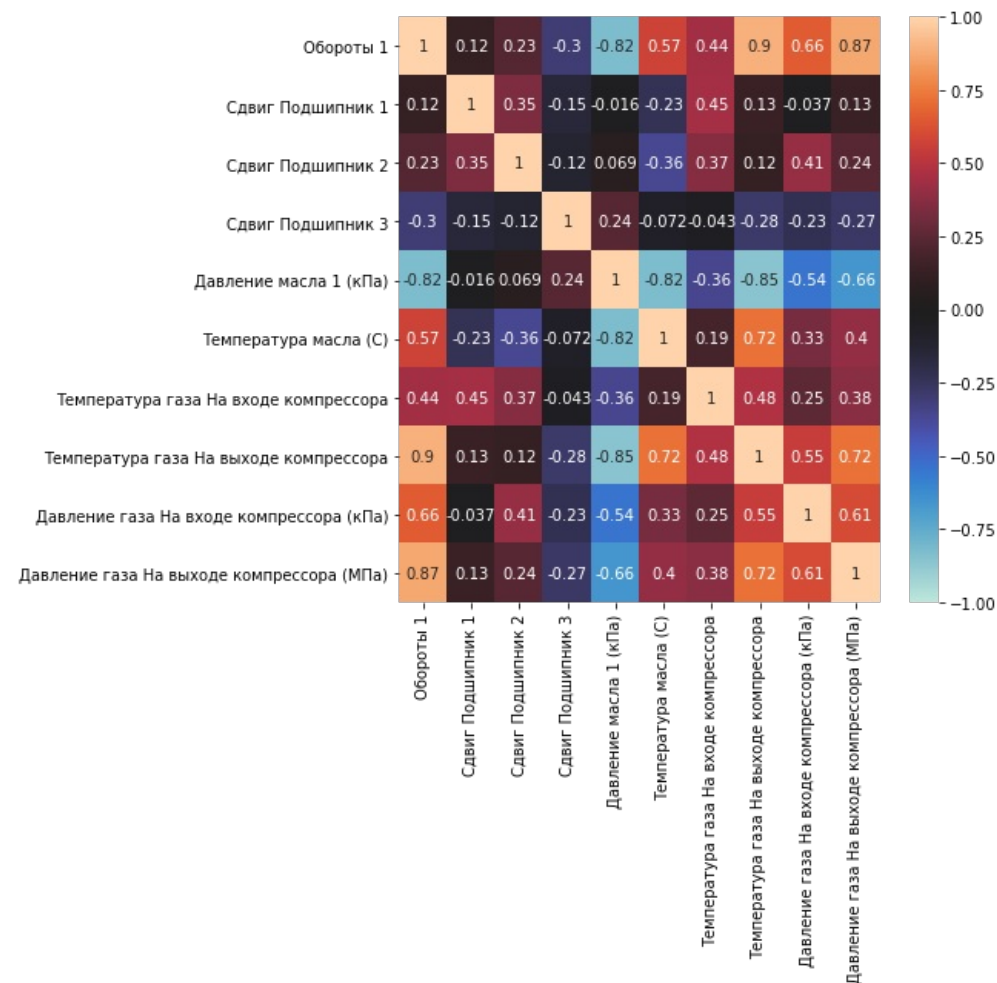
```
df_X = df_X[['Дата и время',  
            'Обороты 1',  
            'Сдвиг Подшипник 1',  
            'Сдвиг Подшипник 2',  
            'Сдвиг Подшипник 3',  
            'Давление масла 1 (кПа)',  
            'Температура масла (C)',  
            'Температура газа На входе компрессора',  
            'Температура газа На выходе компрессора',  
            'Давление газа На входе компрессора (кПа)',  
            'Давление газа На выходе компрессора (МПа)']]
```

In [15]:

```
fig, ax = plt.subplots(figsize=(7,7))  
sns.heatmap(df_X.corr(), annot = True, vmin=-1, vmax=1, center= 0, )
```

Out[15]:

<AxesSubplot:>



In [16]:

```
feature = list(df_X)  
feature = feature[1:]  
feature
```

Out[16]:

```
['Обороты 1',  
 'Сдвиг Подшипник 1',  
 'Сдвиг Подшипник 2',  
 'Сдвиг Подшипник 3',  
 'Давление масла 1 (кПа)',  
 'Температура масла (C)',  
 'Температура газа На входе компрессора',  
 'Температура газа На выходе компрессора',  
 'Давление газа На входе компрессора (кПа)',  
 'Давление газа На выходе компрессора (МПа)']
```

Статистический подход

Вибрация 1-ого подшипника

```
In [17]:
train_data = df_X.drop(labels = 'Дата и время', axis = 1)
```

```
In [18]:
train_data.head(3)
```

Out[18]:

	Обороты 1	Сдвиг Подшипник 1	Сдвиг Подшипник 2	Сдвиг Подшипник 3	Давление масла 1 (кПа)	Температура масла (С)	Температура газа На входе компрессора	Температура газа На выходе компрессора	Давление газа На входе компрессора (кПа)	Давление газа На выходе компрессора (кПа)
0	5258.662931	-0.041063	-0.031354	0.004203	0.344716	35.467580	23.744987	105.907962	38.161112	3.557
1	5277.254228	-0.043171	-0.031549	0.000496	0.344712	35.456034	23.127557	104.946009	39.077081	3.512
2	5331.892912	-0.040885	-0.032023	-0.000016	0.344361	35.528870	23.599861	106.309756	39.040205	3.559

```
In [19]:
train_labels = df_Y['Вибрация Подшипник 1']
```

```
In [20]:
train_labels.head(3)
```

Out[20]:

```
0    53.798579
1    54.173565
2    55.400938
Name: Вибрация Подшипник 1, dtype: float64
```

```
In [21]:
import statsmodels.api as sm
import statsmodels as statsmodels
```

```
In [22]:
train_data = sm.add_constant(train_data)
```

```
In [23]:
train_data.head(3)
```

Out[23]:

	const	Обороты 1	Сдвиг Подшипник 1	Сдвиг Подшипник 2	Сдвиг Подшипник 3	Давление масла 1 (кПа)	Температура масла (С)	Температура газа На входе компрессора	Температура газа На выходе компрессора	Давление газа На входе компрессора (кПа)	Давление газа На выходе компрессора (кПа)
0	1.0	5258.662931	-0.041063	-0.031354	0.004203	0.344716	35.467580	23.744987	105.907962	38.161112	3.557
1	1.0	5277.254228	-0.043171	-0.031549	0.000496	0.344712	35.456034	23.127557	104.946009	39.077081	3.512
2	1.0	5331.892912	-0.040885	-0.032023	-0.000016	0.344361	35.528870	23.599861	106.309756	39.040205	3.559

```
In [24]:
import statsmodels.api as sm
import statsmodels as statsmodels
model = sm.OLS(train_labels, train_data)
```

```
In [25]:
fit = model.fit()
```

In [26]:

```
g = fit.summary()
g
```

Out[26]:

OLS Regression Results

Dep. Variable:	Вибрация Подшипник 1	R-squared:	0.730
Model:	OLS	Adj. R-squared:	0.729
Method:	Least Squares	F-statistic:	2966.
Date:	Sun, 27 Sep 2020	Prob (F-statistic):	0.00
Time:	15:37:21	Log-Likelihood:	-28627.
No. Observations:	11004	AIC:	5.728e+04
Df Residuals:	10993	BIC:	5.736e+04
Df Model:	10		
Covariance Type:	nonrobust		

	coef	std err	t	P> t	[0.025	0.975]
const	153.2894	13.065	11.733	0.000	127.680	178.899
Обороты 1	0.0190	0.001	28.377	0.000	0.018	0.020
Сдвиг Подшипник 1	88.3437	2.209	40.001	0.000	84.014	92.673
Сдвиг Подшипник 2	-419.2766	17.930	-23.384	0.000	-454.423	-384.130
Сдвиг Подшипник 3	-37.0475	5.089	-7.280	0.000	-47.023	-27.072
Давление масла 1 (кПа)	-574.7973	23.128	-24.853	0.000	-620.132	-529.463
Температура масла (С)	0.1841	0.072	2.574	0.010	0.044	0.324
Температура газа На входе компрессора	-0.0359	0.013	-2.843	0.004	-0.061	-0.011
Температура газа На выходе компрессора	-0.1423	0.040	-3.590	0.000	-0.220	-0.065
Давление газа На входе компрессора (кПа)	-0.1286	0.025	-5.069	0.000	-0.178	-0.079
Давление газа На выходе компрессора (МПа)	1.4583	2.663	0.548	0.584	-3.761	6.677

Omnibus:	276.311	Durbin-Watson:	0.012
Prob(Omnibus):	0.000	Jarque-Bera (JB):	158.303
Skew:	-0.127	Prob(JB):	4.22e-35
Kurtosis:	2.470	Cond. No.	4.57e+06

Warnings:

- [1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
- [2] The condition number is large, 4.57e+06. This might indicate that there are strong multicollinearity or other numerical problems.

Reset-рест

In [27]:

```
import warnings
warnings.filterwarnings('ignore')
```

In [28]:

```
statsmodels.stats.diagnostic.linear_reset(fit, power=2)
```

Out[28]:

```
<class 'statsmodels.stats.contrast.ContrastResults'>
<Wald test (chi2): statistic=[[593.8079765]], p-value=3.7207027650545315e-131, df_denom=1>
```

Нулевая гипотеза – спецификация модели адекватна (квадраты). Р-значение = 3.7207027650545315e-131. Значит нулевая гипотеза отвергается на всех стандартных уровнях значимости. Спецификация модели неадекватна

Feature mining

In [29]:

```
df['N_gas_in'] = df_X['Давление газа На входе компрессора (кПа)']/df_X['Температура газа На входе компрессора']
df['N_gas_out'] = df_X['Давление газа На выходе компрессора (МПа)']/df_X['Температура газа На выходе компрессора']
df['N_oil'] = df_X['Давление масла 1 (кПа)']/df_X['Температура масла (C)']

df_X['N_gas_in'] = df['N_gas_in']
df_X['N_gas_out'] = df['N_gas_out']
df_X['N_oil'] = df['N_oil']
```

In [30]:

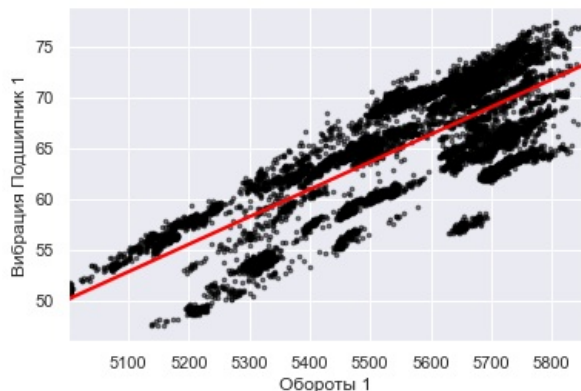
```
feature = [
    'Обороты 1',
    'Сдвиг Подшипник 1',
    'Сдвиг Подшипник 2',
    'Сдвиг Подшипник 3',
    'N_gas_in',
    'N_gas_out',
    'N_oil']
```

In [31]:

```
import seaborn as sns
```

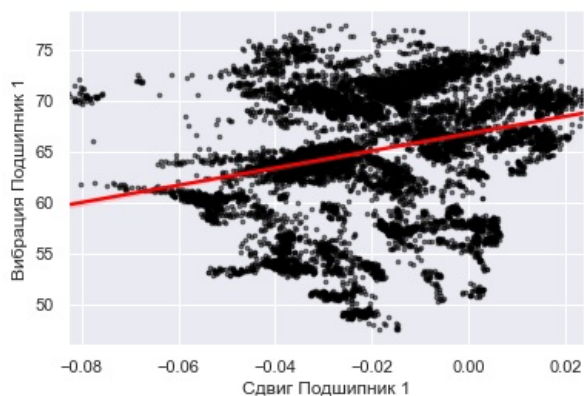
In [32]:

```
import seaborn as sns; sns.set_theme(color_codes=True)
#sns.set(rc={'axes.facecolor':'black',
#            #'figure.facecolor':'black',
#            #'axes.labelcolor':'white',
#            #'xtick.color':'white',
#            #'ytick.color':'white'})
ax = sns.regplot(x=feature[0],
                 y="Вибрация Подшипник 1",
                 data=df,
                 scatter_kws={'s':7, "color": "black", 'alpha':0.5},
                 line_kws={"color": "red"}, order = 1)
```



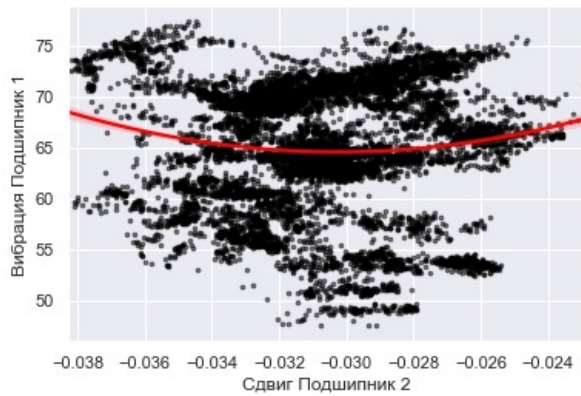
In [33]:

```
ax = sns.regplot(x=feature[1],
                 y="Вибрация Подшипник 1",
                 data=df,
                 scatter_kws={'s':7, "color": "black", 'alpha':0.5},
                 line_kws={"color": "red"}, order =1)
```



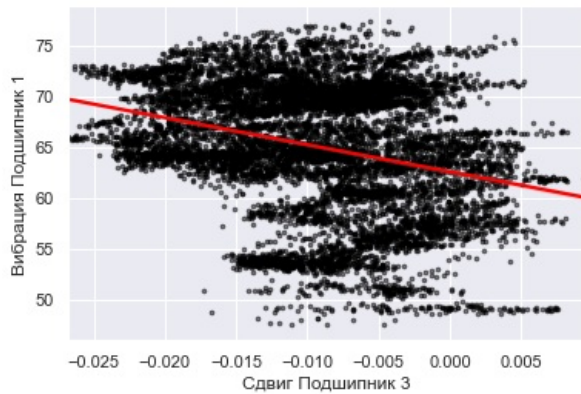
In [34]:

```
ax = sns.regplot(x=feature[2],
                 y="Вибрация Подшипник 1",
                 data=df,
                 scatter_kws={'s':7, "color": "black", 'alpha':0.5},
                 line_kws={"color": "red"}, order = 2)
```



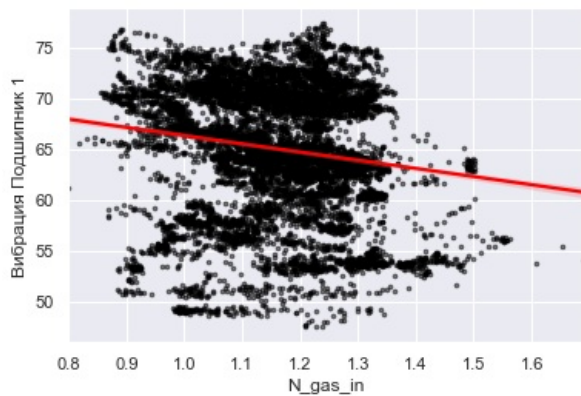
In [35]:

```
ax = sns.regplot(x=feature[3],
                 y="Вибрация Подшипник 1",
                 data=df,
                 scatter_kws={'s':7, "color": "black", 'alpha':0.5},
                 line_kws={"color": "red"}, order = 1)
```



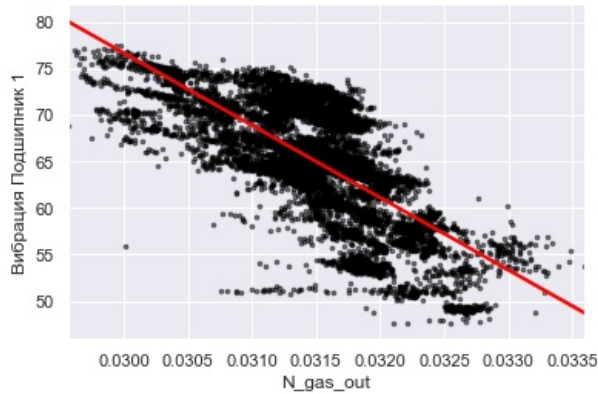
In [36]:

```
ax = sns.regplot(x=feature[4],
                 y="Вибрация Подшипник 1",
                 data=df,
                 scatter_kws={'s':7, "color": "black", 'alpha':0.5},
                 line_kws={"color": "red"}, order = 1)
```



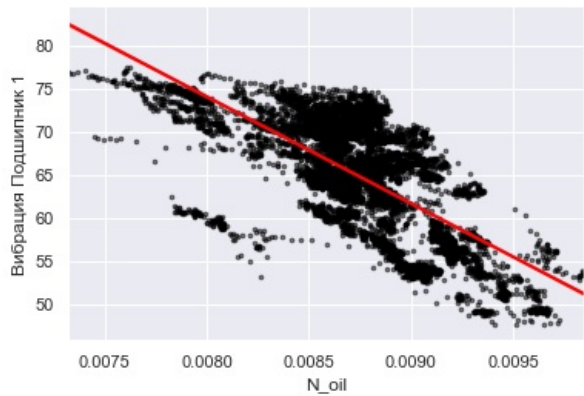
In [37]:

```
ax = sns.regplot(x=feature[5],
                 y="Вибрация Подшипник 1",
                 data=df,
                 scatter_kws={'s':7, "color": "black", 'alpha':0.5},
                 line_kws={"color": "red"}, order = 1)
```



In [38]:

```
ax = sns.regplot(x=feature[6],
                 y="Вибрация Подшипник 1",
                 data=df,
                 scatter_kws={'s':7, "color": "black", 'alpha':0.5},
                 line_kws={"color": "red"}, order = 1)
```



In [39]:

```
train_data = df_X[feature]
train_labels.head(3)
train_data = sm.add_constant(train_data)
```

In [40]:

```
train_data.head(3)
```

Out[40]:

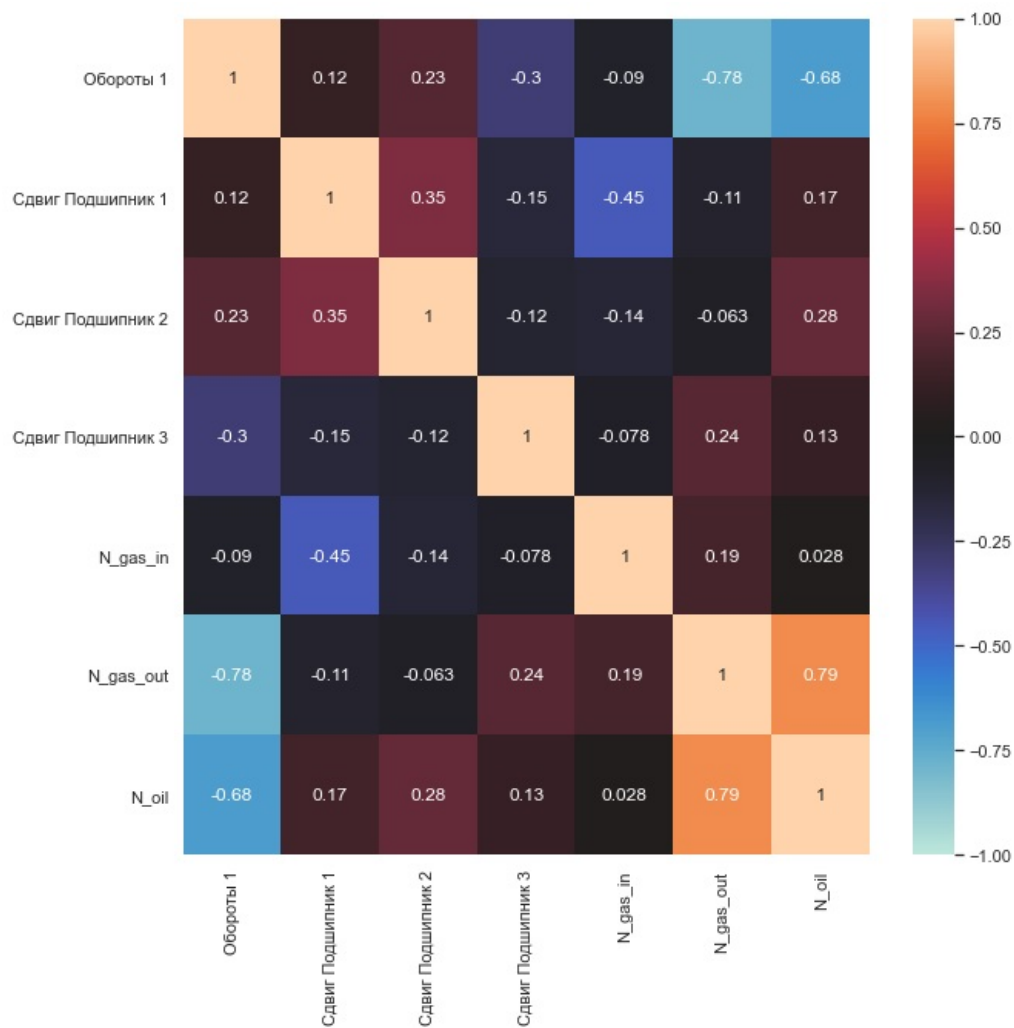
	const	Обороты 1	Сдвиг Подшипник 1	Сдвиг Подшипник 2	Сдвиг Подшипник 3	N_gas_in	N_gas_out	N_oil
0	1.0	5258.662931	-0.041063	-0.031354	0.004203	1.607123	0.033591	0.009719
1	1.0	5277.254228	-0.043171	-0.031549	0.000496	1.689633	0.033472	0.009722
2	1.0	5331.892912	-0.040885	-0.032023	-0.000016	1.654256	0.033482	0.009692

In [41]:

```
fig, ax = plt.subplots(figsize=(10,10))
sns.heatmap(df_X[feature].corr(), annot = True, vmin=-1, vmax=1, center= 0)
```

Out[41]:

<AxesSubplot:>



In [42]:

```
train_labels.head(3)
```

Out[42]:

```
0    53.798579
1    54.173565
2    55.400938
Name: Вибрация Подшипник 1, dtype: float64
```

In [43]:

```
model = sm.OLS(train_labels, train_data)
fit = model.fit()
g = fit.summary()
g
```

Out[43]:

OLS Regression Results

Dep. Variable:	Вибрация Подшипник 1	R-squared:	0.722
Model:	OLS	Adj. R-squared:	0.722
Method:	Least Squares	F-statistic:	4074.
Date:	Sun, 27 Sep 2020	Prob (F-statistic):	0.00
Time:	15:37:25	Log-Likelihood:	-28786.
No. Observations:	11004	AIC:	5.759e+04
Df Residuals:	10996	BIC:	5.765e+04
Df Model:	7		
Covariance Type:	nonrobust		

	coef	std err	t	P> t	[0.025	0.975]
const	-38.6879	4.560	-8.484	0.000	-47.626	-29.750
Обороты 1	0.0213	0.000	64.112	0.000	0.021	0.022
Сдвиг Подшипник 1	99.6554	2.119	47.026	0.000	95.502	103.809
Сдвиг Подшипник 2	-433.9120	15.358	-28.252	0.000	-464.017	-403.807
Сдвиг Подшипник 3	-51.4501	5.104	-10.080	0.000	-61.455	-41.445
N_gas_in	0.0174	0.310	0.056	0.955	-0.590	0.625
N_gas_out	691.4791	120.116	5.757	0.000	456.030	926.928
N_oil	-5498.9860	205.560	-26.751	0.000	-5901.920	-5096.052

Omnibus:	539.308	Durbin-Watson:	0.011
Prob(Omnibus):	0.000	Jarque-Bera (JB):	215.154
Skew:	-0.003	Prob(JB):	1.91e-47
Kurtosis:	2.315	Cond. No.	3.88e+07

Warnings:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 3.88e+07. This might indicate that there are strong multicollinearity or other numerical problems.

In [44]:

```
statsmodels.stats.diagnostic.linear_reset(fit, power=2)
```

Out[44]:

```
<class 'statsmodels.stats.contrast.ContrastResults'>
<Wald test (chi2): statistic=[[280.31891921]], p-value=6.398795886845253e-63, df_denom=1>
```

Нулевая гипотеза – спецификация модели адекватна (квадраты). Р-значение = 6.4e-63. Значит нулевая гипотеза отвергается на всех стандартных уровнях значимости. Спецификация модели неадекватна. Стоит отметить значительное увеличение на десятки порядков результата.

Добавим квадраты переменных

In [45]:

```
feature
```

Out[45]:

```
['Обороты 1',
 'Сдвиг Подшипник 1',
 'Сдвиг Подшипник 2',
 'Сдвиг Подшипник 3',
 'N_gas_in',
 'N_gas_out',
 'N_oil']
```

In [46]:

```
df['N_gas_in^2'] = df['N_gas_in']*df['N_gas_in']
df['N_gas_out^2'] = df['N_gas_out']*df['N_gas_out']
df['N_oil^2'] = df['N_oil']*df['N_oil']
df['Обороты 1^2'] = df['Обороты 1']*df['Обороты 1']
df['Сдвиг Подшипник 1^2'] = df['Сдвиг Подшипник 1']*df['Сдвиг Подшипник 1']
df['Сдвиг Подшипник 2^2'] = df['Сдвиг Подшипник 2']*df['Сдвиг Подшипник 2']
df['Сдвиг Подшипник 3^2'] = df['Сдвиг Подшипник 3']*df['Сдвиг Подшипник 3']

df_X['N_gas_in^2'] = df['N_gas_in^2']
df_X['N_gas_out^2'] = df['N_gas_out^2']
df_X['N_oil^2'] = df['N_oil^2']
df_X['Обороты 1^2'] = df['Обороты 1^2']
df_X['Сдвиг Подшипник 1^2'] = df['Сдвиг Подшипник 1^2']
df_X['Сдвиг Подшипник 2^2'] = df['Сдвиг Подшипник 2^2']
df_X['Сдвиг Подшипник 3^2'] = df['Сдвиг Подшипник 3^2']
```

In [47]:

```
df.head(3)
```

Out[47]:

	Дата и время	Обороты 1	Обороты 2	Обороты 3	Вибрация Подшипник 1	Вибрация Подшипник 2	Вибрация Подшипник 3	Сдвиг Подшипник 1	Сдвиг Подшипник 2	Сдвиг Подшипник 3	...
0	2016-09-02 00:10:00	5258.662931	5258.780949	5255.826839	53.798579	28.936931	45.781854	-0.041063	-0.031354	0.004203	...
1	2016-09-02 00:20:00	5277.254228	5277.860518	5274.922677	54.173565	28.835001	45.525913	-0.043171	-0.031549	0.000496	...
2	2016-09-02 00:30:00	5331.892912	5332.584639	5329.195142	55.400938	28.720065	45.859280	-0.040885	-0.032023	-0.000016	...

3 rows x 27 columns

In [48]:

```
df_X.head(3)
```

Out[48]:

	Дата и время	Обороты 1	Сдвиг Подшипник 1	Сдвиг Подшипник 2	Сдвиг Подшипник 3	Давление масла 1 (кПа)	Температура масла (C)	Температура газа На входе компрессора	Температура газа На выходе компрессора	Давление газа На входе компрессора (кПа)
0	2016-09-02 00:10:00	5258.662931	-0.041063	-0.031354	0.004203	0.344716	35.467580	23.744987	105.907962	38.161112
1	2016-09-02 00:20:00	5277.254228	-0.043171	-0.031549	0.000496	0.344712	35.456034	23.127557	104.946009	39.077081
2	2016-09-02 00:30:00	5331.892912	-0.040885	-0.032023	-0.000016	0.344361	35.528870	23.599861	106.309756	39.040205

3 rows x 21 columns

In [49]:

```
feature = [
    'Обороты 1',
    'Сдвиг Подшипник 1',
    'Сдвиг Подшипник 2',
    'Сдвиг Подшипник 3',
    'N_gas_in',
    'N_gas_out',
    'N_oil',
    'N_gas_in^2',
    'N_gas_out^2',
    'N_oil^2',
    'Обороты 1^2',
    'Сдвиг Подшипник 1^2',
    'Сдвиг Подшипник 2^2',
    'Сдвиг Подшипник 3^2']
```

```
In [50]:
train_data = df_X[feature]
train_labels.head(3)
train_data = sm.add_constant(train_data)
```

```
In [51]:
train_data.head(3)
```

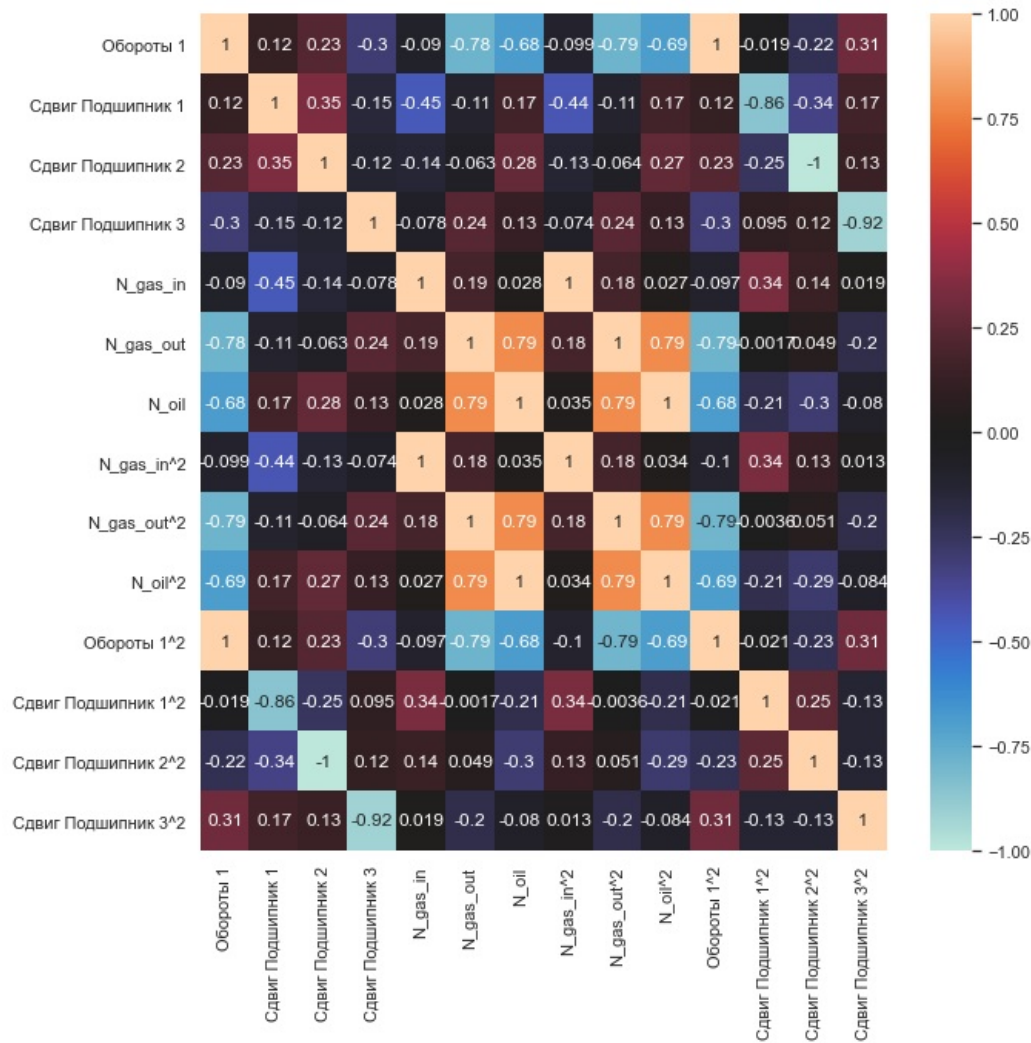
Out[51]:

	const	Обороты 1	Сдвиг Подшипник 1	Сдвиг Подшипник 2	Сдвиг Подшипник 3	N_gas_in	N_gas_out	N_oil	N_gas_in^2	N_gas_out^2	N_oil^2	
0	1.0	5258.662931	-0.041063	-0.031354	0.004203	1.607123	0.033591	0.009719	2.582844	0.001128	0.000094	2.7
1	1.0	5277.254228	-0.043171	-0.031549	0.000496	1.689633	0.033472	0.009722	2.854859	0.001120	0.000095	2.7
2	1.0	5331.892912	-0.040885	-0.032023	-0.000016	1.654256	0.033482	0.009692	2.736562	0.001121	0.000094	2.8

```
In [52]:
fig, ax = plt.subplots(figsize=(10,10))
sns.heatmap(df_X[feature].corr(), annot = True, vmin=-1, vmax=1, center= 0)
```

Out[52]:

<AxesSubplot:>



In [53]:

```
feature = [  
    'Обороты 1',  
    'Сдвиг Подшипник 1',  
    'Сдвиг Подшипник 2',  
    'Сдвиг Подшипник 3',  
    'N_gas_in',  
    'N_gas_out',  
    'N_oil',  
    'Сдвиг Подшипник 1^2',  
    'Сдвиг Подшипник 3^2',]
```

In [54]:

```
train_data = df_X[feature]  
train_labels.head(3)  
train_data = sm.add_constant(train_data)
```

In [55]:

```
train_data.head(3)
```

Out[55]:

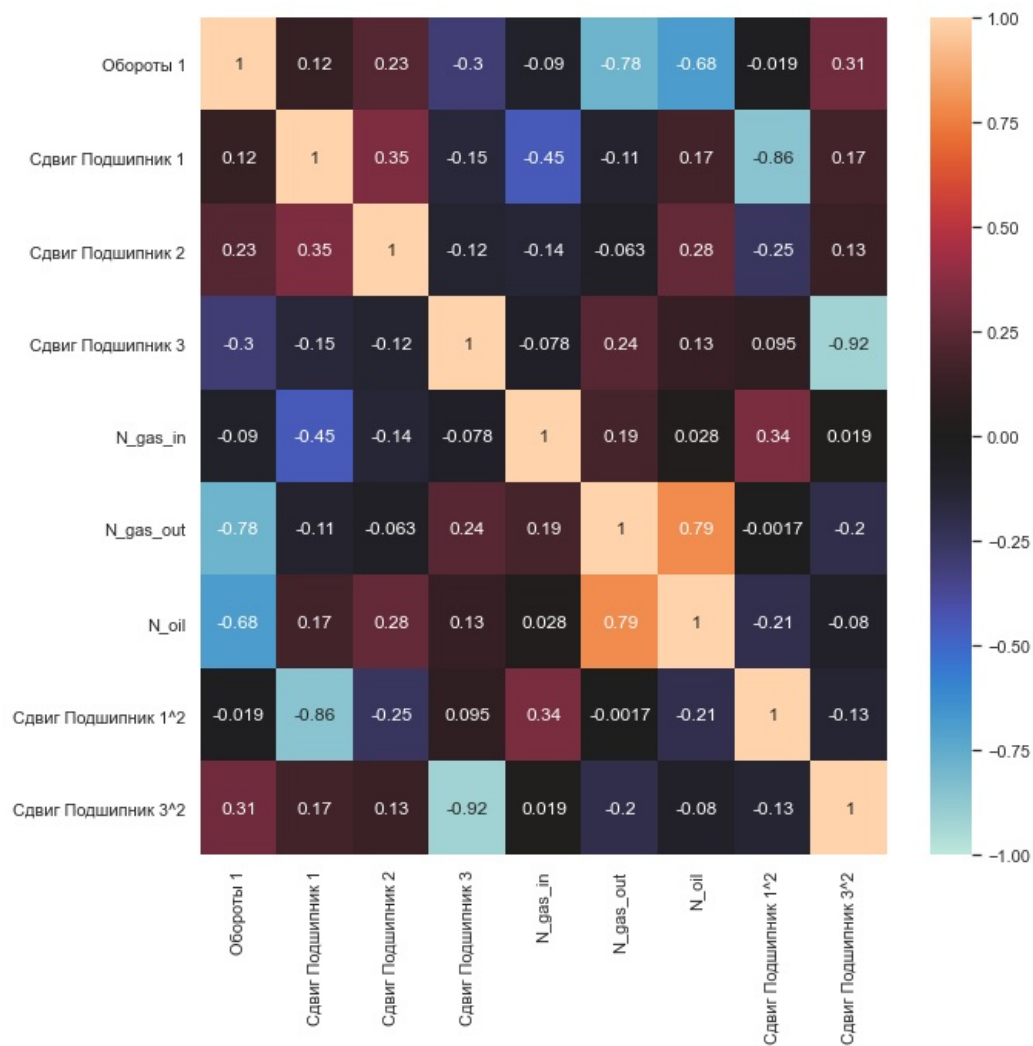
	const	Обороты 1	Сдвиг Подшипник 1	Сдвиг Подшипник 2	Сдвиг Подшипник 3	N_gas_in	N_gas_out	N_oil	Сдвиг Подшипник 1^2	Сдвиг Подшипник 3^2
0	1.0	5258.662931	-0.041063	-0.031354	0.004203	1.607123	0.033591	0.009719	0.001686	1.766763e-05
1	1.0	5277.254228	-0.043171	-0.031549	0.000496	1.689633	0.033472	0.009722	0.001864	2.464321e-07
2	1.0	5331.892912	-0.040885	-0.032023	-0.000016	1.654256	0.033482	0.009692	0.001672	2.649095e-10

In [56]:

```
fig, ax = plt.subplots(figsize=(10,10))
sns.heatmap(df_X[feature].corr(), annot = True, vmin=-1, vmax=1, center= 0)
```

Out[56]:

<AxesSubplot:>



In [57]:

```
model = sm.OLS(train_labels, train_data)
fit = model.fit()
g = fit.summary()
g
```

Out[57]:

OLS Regression Results

Dep. Variable:	Вибрация Подшипник 1	R-squared:	0.734			
Model:	OLS	Adj. R-squared:	0.734			
Method:	Least Squares	F-statistic:	3371.			
Date:	Sun, 27 Sep 2020	Prob (F-statistic):	0.00			
Time:	15:37:27	Log-Likelihood:	-28537.			
No. Observations:	11004	AIC:	5.709e+04			
Df Residuals:	10994	BIC:	5.717e+04			
Df Model:	9					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
const	-45.0124	4.594	-9.798	0.000	-54.017	-36.007
Обороты 1	0.0223	0.000	66.612	0.000	0.022	0.023
Сдвиг Подшипник 1	39.5433	3.648	10.839	0.000	32.392	46.694
Сдвиг Подшипник 2	-431.5898	15.113	-28.557	0.000	-461.214	-401.966
Сдвиг Подшипник 3	-186.3863	12.321	-15.127	0.000	-210.538	-162.234
N_gas_in	-0.9231	0.307	-3.006	0.003	-1.525	-0.321
N_gas_out	637.1606	118.363	5.383	0.000	405.148	869.173
N_oil	-5147.4613	202.791	-25.383	0.000	-5544.969	-4749.954
Сдвиг Подшипник 1^2	-1403.9336	70.253	-19.984	0.000	-1541.642	-1266.225
Сдвиг Подшипник 3^2	-6960.0728	593.400	-11.729	0.000	-8123.243	-5796.903
Omnibus:	23.509	Durbin-Watson:	0.013			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	24.798			
Skew:	0.087	Prob(JB):	4.12e-06			
Kurtosis:	3.154	Cond. No.	1.07e+08			

Warnings:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

[2] The condition number is large, 1.07e+08. This might indicate that there are strong multicollinearity or other numerical problems.

In [58]:

```
statsmodels.stats.diagnostic.linear_reset(fit, power=2)
```

Out[58]:

```
<class 'statsmodels.stats.contrast.ContrastResults'>
<Wald test (chi2): statistic=[[151.86374637]], p-value=6.785866850028105e-35, df_denom=1>
```

Нулевая гипотеза – спецификация модели адекватна (квадраты). Р-значение = 6.8e-35. Значит нулевая гипотеза отвергается на всех стандартных уровнях значимости. Спецификация модели неадекватна. Стоит отметить значительное увеличение на десятки порядков результата.

Добавим обратные величины

In [59]:

```
df['Обрат Обороты 1'] = 1/df['Обороты 1']
df['Обрат Сдвиг Подшипник 1'] = 1/df['Сдвиг Подшипник 1']
df['Обрат Сдвиг Подшипник 2'] = 1/df['Сдвиг Подшипник 2']
df['Обрат Сдвиг Подшипник 3'] = 1/df['Сдвиг Подшипник 3']

df_X['Обрат Обороты 1'] = df['Обрат Обороты 1']
df_X['Обрат Сдвиг Подшипник 1'] = df['Обрат Сдвиг Подшипник 1']
df_X['Обрат Сдвиг Подшипник 2'] = df['Обрат Сдвиг Подшипник 2']
df_X['Обрат Сдвиг Подшипник 3'] = df['Обрат Сдвиг Подшипник 3']
```

In [60]:

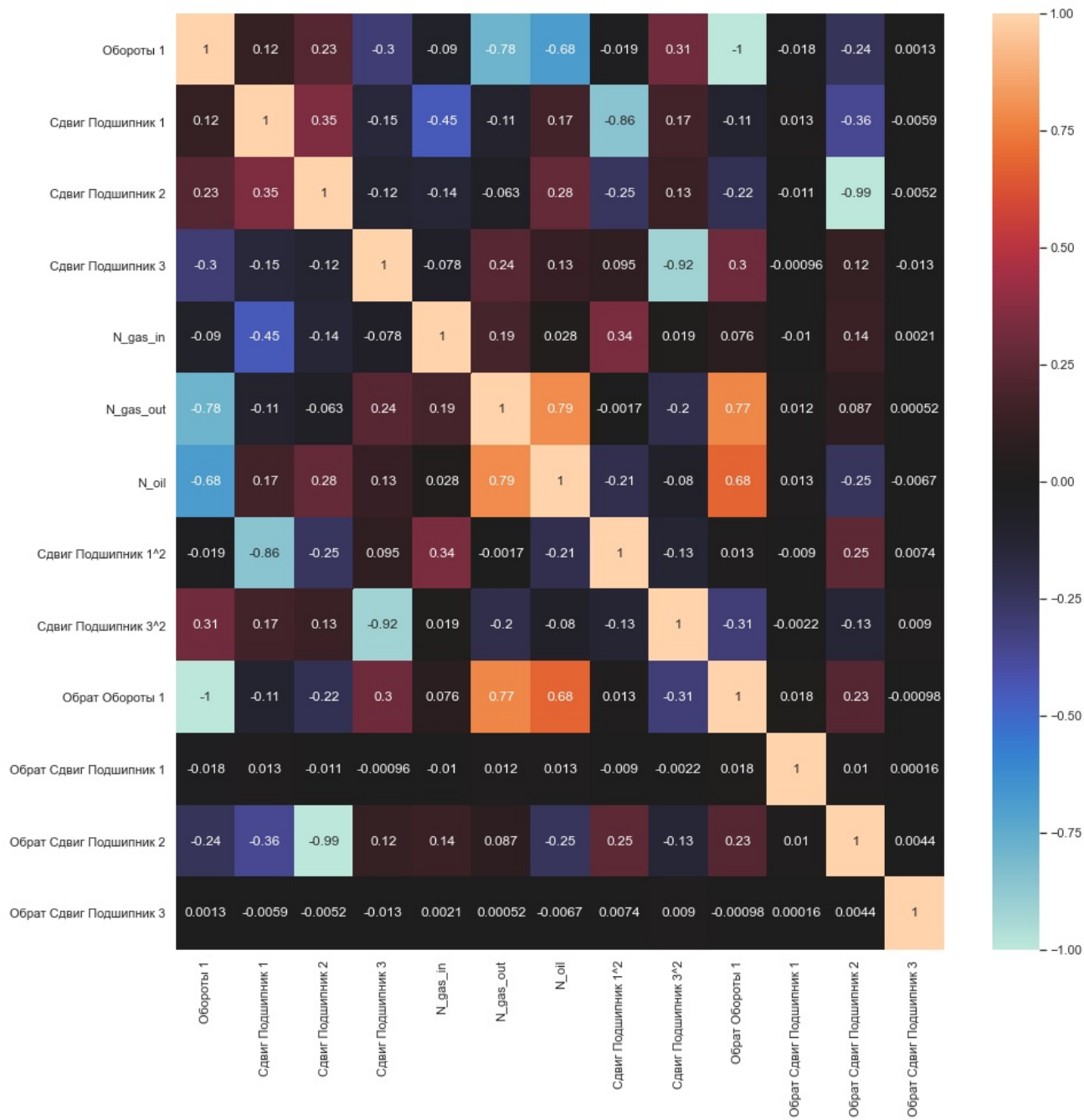
```
feature = [  
    'Обороты 1',  
    'Сдвиг Подшипник 1',  
    'Сдвиг Подшипник 2',  
    'Сдвиг Подшипник 3',  
    'N_gas_in',  
    'N_gas_out',  
    'N_oil',  
    'Сдвиг Подшипник 1^2',  
    'Сдвиг Подшипник 3^2',  
    'Обрат Обороты 1',  
    'Обрат Сдвиг Подшипник 1',  
    'Обрат Сдвиг Подшипник 2',  
    'Обрат Сдвиг Подшипник 3']
```

In [61]:

```
fig, ax = plt.subplots(figsize=(15,15))  
sns.heatmap(df_X[feature].corr(), annot = True, vmin=-1, vmax=1, center= 0)
```

Out[61]:

<AxesSubplot:>

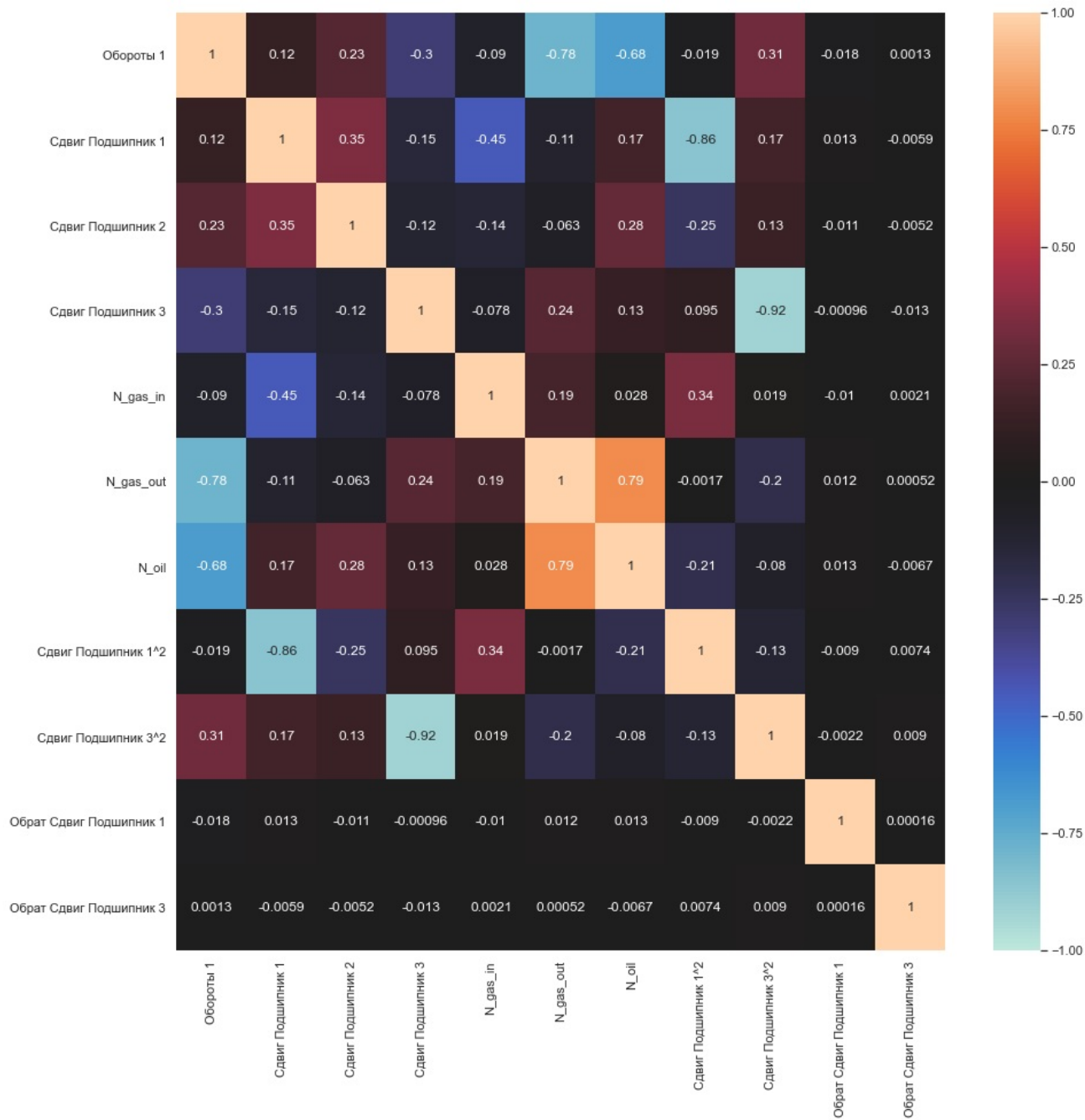


In [62]:

```
feature = [  
    'Обороты 1',  
    'Сдвиг Подшипник 1',  
    'Сдвиг Подшипник 2',  
    'Сдвиг Подшипник 3',  
    'N_gas_in',  
    'N_gas_out',  
    'N_oil',  
    'Сдвиг Подшипник 1^2',  
    'Сдвиг Подшипник 3^2',  
    'Обрат Сдвиг Подшипник 1',  
    'Обрат Сдвиг Подшипник 3']  
  
fig, ax = plt.subplots(figsize=(15,15))  
sns.heatmap(df_X[feature].corr(), annot = True, vmin=-1, vmax=1, center= 0)
```

Out[62]:

<AxesSubplot:>



In [63]:

```
train_data = df_X[feature]
train_labels.head(3)
train_data = sm.add_constant(train_data)

model = sm.OLS(train_labels, train_data)
fit = model.fit()
g = fit.summary()
g
```

Out[63]:

OLS Regression Results

Dep. Variable:	Вибрация Подшипник 1	R-squared:	0.734			
Model:	OLS	Adj. R-squared:	0.734			
Method:	Least Squares	F-statistic:	2758.			
Date:	Sun, 27 Sep 2020	Prob (F-statistic):	0.00			
Time:	15:37:28	Log-Likelihood:	-28537.			
No. Observations:	11004	AIC:	5.710e+04			
Df Residuals:	10992	BIC:	5.719e+04			
Df Model:	11					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
const	-45.0313	4.594	-9.802	0.000	-54.037	-36.026
Обороты 1	0.0223	0.000	66.605	0.000	0.022	0.023
Сдвиг Подшипник 1	39.5734	3.649	10.846	0.000	32.422	46.725
Сдвиг Подшипник 2	-431.7071	15.115	-28.561	0.000	-461.336	-402.078
Сдвиг Подшипник 3	-186.5746	12.324	-15.140	0.000	-210.731	-162.418
N_gas_in	-0.9253	0.307	-3.013	0.003	-1.527	-0.323
N_gas_out	638.1105	118.375	5.391	0.000	406.074	870.148
N_oil	-5148.2528	202.813	-25.384	0.000	-5545.803	-4750.702
Сдвиг Подшипник 1^2	-1403.4312	70.259	-19.975	0.000	-1541.152	-1265.710
Сдвиг Подшипник 3^2	-6966.1483	593.468	-11.738	0.000	-8129.453	-5802.844
Обрат Сдвиг Подшипник 1	-1.202e-06	2.07e-06	-0.582	0.561	-5.25e-06	2.85e-06
Обрат Сдвиг Подшипник 3	-7.123e-09	9.84e-09	-0.724	0.469	-2.64e-08	1.22e-08
Omnibus:	23.580	Durbin-Watson:	0.013			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	24.869			
Skew:	0.087	Prob(JB):	3.98e-06			
Kurtosis:	3.154	Cond. No.	6.04e+10			

Warnings:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

[2] The condition number is large, 6.04e+10. This might indicate that there are strong multicollinearity or other numerical problems.

In [64]:

```
statsmodels.stats.diagnostic.linear_reset(fit, power=2)
```

Out[64]:

```
<class 'statsmodels.stats.contrast.ContrastResults'>
<Wald test (chi2): statistic=[[151.86694162]], p-value=6.774963888289924e-35, df_denom=1>
```

Нулевая гипотеза – спецификация модели адекватна (квадраты). Р-значение = 6.8e-35. Значит нулевая гипотеза отвергается на всех стандартных уровнях значимости. Спецификация модели неадекватна. Для подбора спецификации необходимо более детальное понимание работы устройства.

ML подход

Вибрация 1-ого подшипника.

In [65]:

```
feature = ['Дата и время',
'Обороты 1',
'Сдвиг Подшипник 1',
'Сдвиг Подшипник 2',
'Сдвиг Подшипник 3',
'Давление масла 1 (кПа)',
'Температура масла (C)',
'Температура газа На входе компрессора',
'Температура газа На выходе компрессора',
'Давление газа На входе компрессора (кПа)',
'Давление газа На выходе компрессора (МПа)']
```

In [66]:

```
from sklearn import model_selection
from sklearn import metrics
from sklearn import linear_model

train_data_with_time, test_data_with_time, train_labels, test_labels = model_selection.train_test_split(
    df[feature], df_Y[['Вибрация Подшипник 1']],
    test_size = 0.3, random_state = 0)
```

In [67]:

```
train_data = train_data_with_time.drop(labels = 'Дата и время', axis = 1)
test_data = test_data_with_time.drop(labels = 'Дата и время', axis = 1)
```

In [68]:

```
train_data.head(3)
```

Out[68]:

	Обороты 1	Сдвиг Подшипник 1	Сдвиг Подшипник 2	Сдвиг Подшипник 3	Давление масла 1 (кПа)	Температура масла (C)	Температура газа На входе компрессора	Температура газа На выходе компрессора	Давление газа На входе компрессора (кПа)	Давление газа На выходе компрессора (МПа)
326	5690.409310	-0.044005	-0.037722	-0.013086	0.333918	41.436768	32.200254	116.979284	37.157404	3.157404
9624	5507.845046	-0.026793	-0.031441	-0.005843	0.341639	39.529826	34.848592	114.352496	31.444880	3.144880
463	5735.913114	-0.033573	-0.036878	-0.018636	0.332840	42.050476	30.245932	118.368103	36.292547	3.292547

In [69]:

```
from sklearn.linear_model import LinearRegression

model_LR = LinearRegression()
model_LR.fit(train_data, train_labels)
model_predictions = model_LR.predict(test_data)
```

MSE

In [70]:

```
metrics.mean_squared_error(test_labels, model_predictions)
```

Out[70]:

10.490947392829261

In [71]:

```
df_model_predictions = pd.DataFrame(data=model_predictions)
df_model_predictions.head(3)
```

Out[71]:

	0
0	61.016653
1	61.910919
2	73.736708

In [72]:

```
train_data.head(3)
```

Out[72]:

	Обороты 1	Сдвиг Подшипник 1	Сдвиг Подшипник 2	Сдвиг Подшипник 3	Давление масла 1 (кПа)	Температура масла (C)	Температура газа На входе компрессора	Температура газа На выходе компрессора	Давление газа На входе компрессора (кПа)	Давление газа На выходе компрессора (кПа)
326	5690.409310	-0.044005	-0.037722	-0.013086	0.333918	41.436768	32.200254	116.979284	37.157404	3.591
9624	5507.845046	-0.026793	-0.031441	-0.005843	0.341639	39.529826	34.848592	114.352496	31.444880	3.571
463	5735.913114	-0.033573	-0.036878	-0.018636	0.332840	42.050476	30.245932	118.368103	36.292547	3.611

In [73]:

```
feature = ['Обороты 1',
           'Сдвиг Подшипник 1',
           'Сдвиг Подшипник 2',
           'Сдвиг Подшипник 3',
           'Давление масла 1 (кПа)',
           'Температура масла (C)',
           'Температура газа На входе компрессора',
           'Температура газа На выходе компрессора',
           'Давление газа На входе компрессора (кПа)',
           'Давление газа На выходе компрессора (МПа)']
```

In [74]:

```
for i in range(len(feature)):
    print(feature[i], ' = ', model_LR.coef_[0][i])
```

Обороты 1 = 0.018609838918690665
Сдвиг Подшипник 1 = 88.43072913696372
Сдвиг Подшипник 2 = -405.4839357921208
Сдвиг Подшипник 3 = -34.916594557964174
Давление масла 1 (кПа) = -575.357728619246
Температура масла (C) = 0.21420283975863322
Температура газа На входе компрессора = -0.043155062583072466
Температура газа На выходе компрессора = -0.12536324110748437
Давление газа На входе компрессора (кПа) = -0.13006008563293178
Давление газа На выходе компрессора (МПа) = 2.1438121126145915

Другая модель

In [75]:

```
feature = ['Дата и время',
           'Обороты 1',
           'Сдвиг Подшипник 1',
           'Сдвиг Подшипник 2',
           'Сдвиг Подшипник 3',
           'N_gas_in',
           'N_gas_out',
           'N_oil',
           'Сдвиг Подшипник 1^2',
           'Сдвиг Подшипник 3^2']
```

In [76]:

```
df[feature].head(3)
```

Out[76]:

	Дата и время	Обороты 1	Сдвиг Подшипник 1	Сдвиг Подшипник 2	Сдвиг Подшипник 3	N_gas_in	N_gas_out	N_oil	Сдвиг Подшипник 1^2	Сдвиг Подшипник 3^2
0	2016-09-02 00:10:00	5258.662931	-0.041063	-0.031354	0.004203	1.607123	0.033591	0.009719	0.001686	1.766763e-05
1	2016-09-02 00:20:00	5277.254228	-0.043171	-0.031549	0.000496	1.689633	0.033472	0.009722	0.001864	2.464321e-07
2	2016-09-02 00:30:00	5331.892912	-0.040885	-0.032023	-0.000016	1.654256	0.033482	0.009692	0.001672	2.649095e-10

In [77]:

```
train_data_with_time, test_data_with_time, train_labels, test_labels = model_selection.train_test_split(
    df[feature], df_Y[['Вибрация Подшипник 1']],
    test_size = 0.3, random_state = 0)
```

In [78]:

```
train_data = train_data_with_time.drop(labels = 'Дата и время', axis = 1)
test_data = test_data_with_time.drop(labels = 'Дата и время', axis = 1)
```

In [79]:

```
model_LR = LinearRegression()
model_LR.fit(train_data, train_labels)
model_predictions = model_LR.predict(test_data)
```

MSE

In [80]:

```
metrics.mean_squared_error(test_labels, model_predictions)
```

Out[80]:

10.312747516817867

In [81]:

```
df_model_predictions = pd.DataFrame(data=model_predictions)
df_model_predictions.head(3)
```

Out[81]:

	0
0	62.148483
1	63.234531
2	74.264729

In [82]:

```
feature = feature[1:]
feature
```

Out[82]:

```
['Обороты 1',
 'Сдвиг Подшипник 1',
 'Сдвиг Подшипник 2',
 'Сдвиг Подшипник 3',
 'N_gas_in',
 'N_gas_out',
 'N_oil',
 'Сдвиг Подшипник 1^2',
 'Сдвиг Подшипник 3^2']
```

In [83]:

```
for i in range(len(feature)):
    print(feature[i], ' = ', model_LR.coef_[0][i])
```

```
Обороты 1 = 0.022082801618752146
Сдвиг Подшипник 1 = 39.88838900898378
Сдвиг Подшипник 2 = -420.0004624886363
Сдвиг Подшипник 3 = -184.4072024641636
N_gas_in = -0.7578438979205707
N_gas_out = 591.4407631363326
N_oil = -5198.487829915386
Сдвиг Подшипник 1^2 = -1393.275403920896
Сдвиг Подшипник 3^2 = -6955.629332322441
```

График наблюдений и расчетных значений

In [84]:

```
df_plot = pd.DataFrame({'prediction' : [], 'Дата и время' : [], 'Вибрация Подшипник 1' : []})
df_plot
```

Out[84]:

prediction	Дата и время	Вибрация Подшипник 1
------------	--------------	----------------------

In [85]:

```
test_labels['Вибрация Подшипник 1']
df_model_predictions[0]
test_data_with_time['Дата и время']
```

Out[85]:

```
7165    2016-10-21 18:20:00.001
10235    2016-11-12 02:00:00.001
1121     2016-09-09 19:00:00.000
1328     2016-09-11 05:30:00.000
3806     2016-09-28 10:30:00.000
...
3359     2016-09-25 08:00:00.000
3277     2016-09-24 18:20:00.000
8844     2016-11-02 10:10:00.001
1589     2016-09-13 01:00:00.000
7913     2016-10-26 23:00:00.001
Name: Дата и время, Length: 3302, dtype: datetime64[ns]
```

In [86]:

```
plt.figure(figsize=(15,5), dpi=80, facecolor='w', edgecolor='k')
ax = plt.gca()
plt.scatter(test_data_with_time['Дата и время'], test_labels['Вибрация Подшипник 1'], color = 'red', label = 'Real', s = 1)

plt.scatter(test_data_with_time['Дата и время'], df_model_predictions[0], color = 'blue', label = 'Predicted', s = 1)
plt.title('Вибрация Подшипник 1', fontsize=14)
plt.legend(loc=2, prop={'size': 14})
plt.show()
```



Вибрация 2-ого подшипника

In [87]:

```
feature = ['Дата и время',
           'Обороты 1',
           'Сдвиг Подшипник 1',
           'Сдвиг Подшипник 2',
           'Сдвиг Подшипник 3',
           'N_gas_in',
           'N_gas_out',
           'N_oil',
           'Сдвиг Подшипник 1^2',
           'Сдвиг Подшипник 3^2']
```

In [88]:

```
from sklearn import model_selection
from sklearn import metrics
from sklearn import linear_model

train_data_with_time, test_data_with_time, train_labels, test_labels = model_selection.train_test_split(
    df_X[feature], df_Y[['Вибрация Подшипник 2']],
    test_size = 0.3, random_state = 0)
```

In [89]:

```
train_data = train_data_with_time.drop(labels = 'Дата и время', axis = 1)
test_data = test_data_with_time.drop(labels = 'Дата и время', axis = 1)
```

In [90]:

```
from sklearn.linear_model import LinearRegression

model_LR = LinearRegression()
model_LR.fit(train_data, train_labels)
model_predictions = model_LR.predict(test_data)
```

MSE

In [91]:

```
metrics.mean_squared_error(test_labels, model_predictions)
```

Out[91]:

0.24068057869475804

In [92]:

```
df_model_predictions = pd.DataFrame(data=model_predictions)
df_model_predictions.head(3)
```

Out[92]:

	0
0	28.942057
1	28.857692
2	29.595040

In [93]:

```
feature = ['Обороты 1',
           'Сдвиг Подшипник 1',
           'Сдвиг Подшипник 2',
           'Сдвиг Подшипник 3',
           'N_gas_in',
           'N_gas_out',
           'N_oil',
           'Сдвиг Подшипник 1^2',
           'Сдвиг Подшипник 3^2']
```

In [94]:

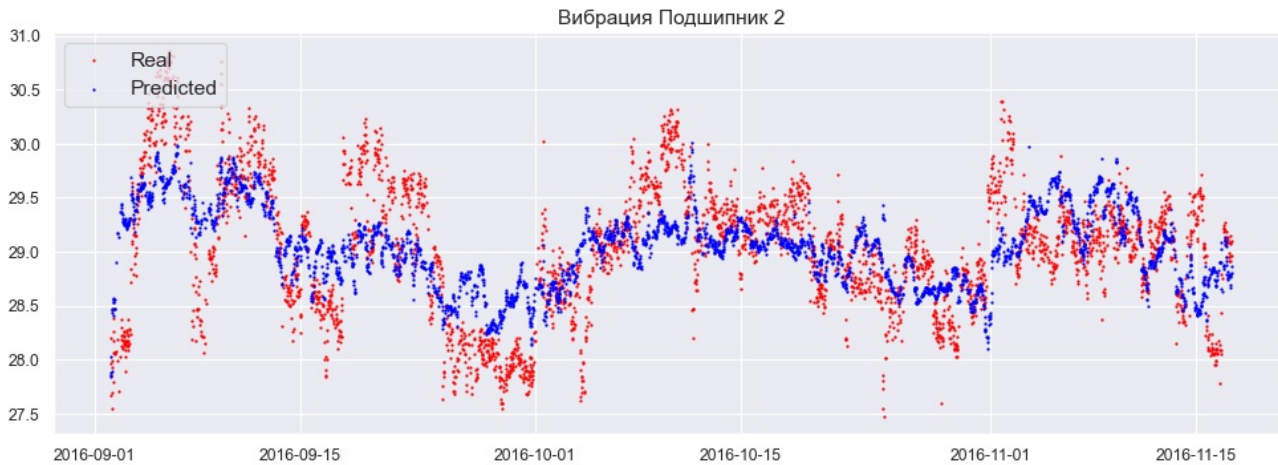
```
for i in range(len(feature)):
    print(feature[i], ' = ', model_LR.coef_[0][i])
```

```
Обороты 1 = -0.0010071441756197346
Сдвиг Подшипник 1 = 7.235292202916836
Сдвиг Подшипник 2 = -79.4681106771712
Сдвиг Подшипник 3 = -28.382408301455758
N_gas_in = -0.3653171053537295
N_gas_out = -275.47117767562423
N_oil = -189.6960229404625
Сдвиг Подшипник 1^2 = -80.51539897512389
Сдвиг Подшипник 3^2 = -69.62301514245112
```

In [95]:

```
plt.figure(figsize=(15,5), dpi=80, facecolor='w', edgecolor='k')
ax = plt.gca()
plt.scatter(test_data_with_time['Дата и время'], test_labels['Вибрация Подшипник 2'], color = 'red', label = 'Real', s = 1)

plt.scatter(test_data_with_time['Дата и время'], df_model_predictions[0], color = 'blue', label = 'Predicted', s = 1)
plt.title('Вибрация Подшипник 2', fontsize=14)
plt.legend(loc=2, prop={'size': 14})
plt.show()
```



Вибрация 3-ого подшипника

In [96]:

```
feature = ['Дата и время',
           'Обороты 1',
           'Сдвиг Подшипник 1',
           'Сдвиг Подшипник 2',
           'Сдвиг Подшипник 3',
           'N_gas_in',
           'N_gas_out',
           'N_oil',
           'Сдвиг Подшипник 1^2',
           'Сдвиг Подшипник 3^2']
```

In [97]:

```
from sklearn import model_selection
from sklearn import metrics
from sklearn import linear_model

train_data_with_time, test_data_with_time, train_labels, test_labels = model_selection.train_test_split(
    df_X, df_Y[['Вибрация Подшипник 3']],
    test_size = 0.3, random_state = 0)
```

In [98]:

```
train_data = train_data_with_time.drop(labels = 'Дата и время', axis = 1)
test_data = test_data_with_time.drop(labels = 'Дата и время', axis = 1)
```

In [99]:

```
model_LR = LinearRegression()
model_LR.fit(train_data, train_labels)
model_predictions = model_LR.predict(test_data)
```

MSE

In [100]:

```
metrics.mean_squared_error(test_labels, model_predictions)
```

Out[100]:

0.10263938377695851

In [101]:

```
df_model_predictions = pd.DataFrame(data=model_predictions)
df_model_predictions.head(3)
```

Out[101]:

	0
0	46.437460
1	45.535597
2	47.089597

In [102]:

```
feature = ['Обороты 1',
           'Сдвиг Подшипник 1',
           'Сдвиг Подшипник 2',
           'Сдвиг Подшипник 3',
           'N_gas_in',
           'N_gas_out',
           'N_oil',
           'Сдвиг Подшипник 1^2',
           'Сдвиг Подшипник 3^2']

for i in range(len(feature)):
    print(feature[i], ' = ', model_LR.coef_[0][i])
```

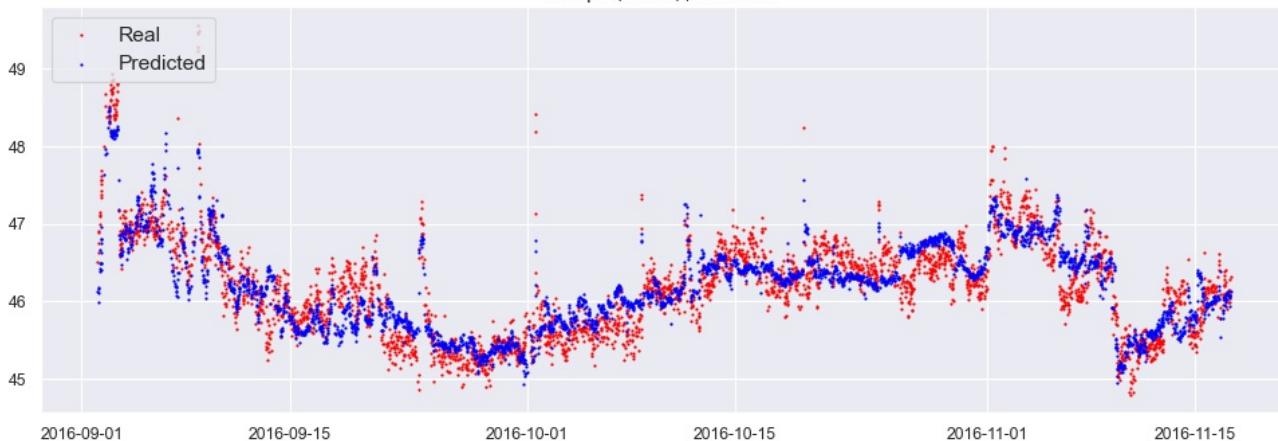
```
Обороты 1 = 0.20041093264245208
Сдвиг Подшипник 1 = 7.0543534667370125
Сдвиг Подшипник 2 = 1073.2405310118274
Сдвиг Подшипник 3 = -4.000263781187274
N_gas_in = -1084.6581104674299
N_gas_out = 8.724051514563145
N_oil = -0.09779370989217187
Сдвиг Подшипник 1^2 = -0.6904467062134189
Сдвиг Подшипник 3^2 = 0.15424350110833873
```

In [103]:

```
plt.figure(figsize=(15,5), dpi=80, facecolor='w', edgecolor='k')
ax = plt.gca()
plt.scatter(test_data_with_time['Дата и время'], test_labels['Вибрация Подшипник 3'], color = 'red', label = 'Real', s = 1)

plt.scatter(test_data_with_time['Дата и время'], df_model_predictions[0], color = 'blue', label = 'Predicted', s = 1)
plt.title('Вибрация Подшипник 3', fontsize=14)
plt.legend(loc=2, prop={'size': 14})
plt.show()
```

Вибрация Подшипник 3



Выводы

Построены математические модели для вибрации трёх подшипников.

В качестве регрессоров использовались не температуры и давления, а их отношение, что пропорционально концентрации.

Из-за сильной корреляции оборотов в рамках модели был оставлен только один регрессор.