

# Кольца Илья Вячеславович

## Лабораторная работа № 4

### Вариант 4

In [12]:

```
import pandas as pd
import warnings
import numpy as np
import scipy.stats as sts
import matplotlib.pyplot as plt
'''import rpy2.robjtypes.numpy2ri
from rpy2.robjtypes.packages import importr
rpy2.robjtypes.numpy2ri.activate()
stats = importr('stats')'''
warnings.filterwarnings('ignore')
%matplotlib inline
```

In [33]:

```
df = pd.read_csv('Data.csv')
df.head()
```

Out[33]:

	Age	AttendedBootcamp	BootcampFinish	BootcampFullJobAfter	BootcampLoanYesNo	BootcampMonthsAgo	BootcampNa
0	28.0	0.0	NaN	NaN	NaN	NaN	M
1	22.0	0.0	NaN	NaN	NaN	NaN	M
2	19.0	0.0	NaN	NaN	NaN	NaN	M
3	26.0	0.0	NaN	NaN	NaN	NaN	M
4	20.0	0.0	NaN	NaN	NaN	NaN	M

5 rows x 113 columns



In [34]:

```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15620 entries, 0 to 15619
Columns: 113 entries, Age to StudentDebtOwe
dtypes: float64(85), object(28)
memory usage: 13.5+ MB
```

In [14]:

```
df = df[['CityPopulation', 'EmploymentStatus', 'Gender', 'HasDebt', 'JobPref', 'JobWhere
Pref', 'MaritalStatus', 'Income', 'SchoolDegree']]
```

df.head()

Out[14]:

	CityPopulation	EmploymentStatus	Gender	HasDebt	JobPref	JobWherePref	MaritalStatus	Income	SchoolDegree
0	between 100,000 and 1 million	Employed for wages	male	1.0	freelance	NaN	married or domestic partnership	32000.0	some college credit, no degree
1	between 100,000 and 1 million	Employed for wages	male	0.0	work for a startup	in an office with other developers	NaN	15000.0	some college credit, no degree
2	more than 1 million	Employed for wages	male	0.0	start your own business	NaN	NaN	48000.0	high school diploma or equivalent (GED)
3	more than 1 million	Employed for wages	female	1.0	work for a startup	from home	NaN	43000.0	bachelor's degree
4	between 100,000 and 1 million	Employed for wages	female	1.0	work for a medium-sized company	in an office with other developers	NaN	6000.0	some college credit, no degree

In [15]:

```
df = df.dropna()
df = df[(df.Gender == 'male') | (df.Gender == 'female')]
df.head()
```

Out[15]:

	CityPopulation	EmploymentStatus	Gender	HasDebt	JobPref	JobWherePref	MaritalStatus	Income	SchoolDegree
20	more than 1 million	Employed for wages	female	1.0	work for a medium-sized company	in an office with other developers	married or domestic partnership	200000.0	master's degree (non-professional)
36	between 100,000 and 1 million	Self-employed freelancer	male	1.0	work for a medium-sized company	in an office with other developers	married or domestic partnership	60000.0	high school diploma or equivalent (GED)
59	more than 1 million	Employed for wages	male	1.0	work for a medium-sized company	in an office with other developers	married or domestic partnership	35000.0	some college credit, no degree
71	more than 1 million	Employed for wages	male	1.0	work for a multinational corporation	from home	married or domestic partnership	56000.0	some college credit, no degree
72	less than 100,000	Employed for wages	male	1.0	work for a medium-sized company	from home	married or domestic partnership	35000.0	bachelor's degree

In [16]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 931 entries, 20 to 15616
Data columns (total 9 columns):
#   Column              Non-Null Count  Dtype
---  -
0   CityPopulation      931 non-null   object
1   EmploymentStatus    931 non-null   object
2   Gender              931 non-null   object
3   HasDebt             931 non-null   float64
4   JobPref             931 non-null   object
```

```
5 JobWherePref      931 non-null    object
6 MaritalStatus     931 non-null    object
7 Income            931 non-null    float64
8 SchoolDegree      931 non-null    object
```

dtypes: float64(2), object(7)

memory usage: 72.7+ KB

In [17]:

```
print(f"Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для уровня до  
хода:")
z, p = sts.shapiro(df.Income)
print(f"Статистика критерия {z}, p_value {p}")
```

Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для уровня дохода:  
Статистика критерия 0.8556973934173584, p\_value 2.449250369630941e-28

In [18]:

```
df['log_income'] = np.log(df.Income + 1)
print(f"Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для уровня до  
хода после лог-трансформации:")
z, p = sts.shapiro(df.log_income)
print(f"Статистика критерия {z}, p_value {p}")
```

Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для уровня дохода пос  
ле лог-трансформации:

Статистика критерия 0.9735249876976013, p\_value 5.58358003929782e-12

In [19]:

```
df['sqrt_income'] = np.sqrt(df.Income)
print(f"Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для уровня до  
хода после вычисления квадратного корня:")
z, p = sts.shapiro(df.sqrt_income)
print(f"Статистика критерия {z}, p_value {p}")
```

Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для уровня дохода пос  
ле вычисления квадратного корня:

Статистика критерия 0.9709372520446777, p\_value 1.1025697499097986e-12

In [20]:

```
df['inv_income'] = 1 / (df.Income)
print(f"Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для уровня до  
хода после вычисления квадратного корня:")
z, p = sts.shapiro(df.inv_income)
print(f"Статистика критерия {z}, p_value {p}")
```

Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для уровня дохода пос  
ле вычисления квадратного корня:

Статистика критерия 0.7213217616081238, p\_value 1.0208053496571024e-36

In [21]:

```
df['max_minus_income'] = (df.Income.max() + 1 - df.Income)
print(f"Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для уровня до  
хода после вычисления квадратного корня:")
z, p = sts.shapiro(df.max_minus_income)
print(f"Статистика критерия {z}, p_value {p}")
```

Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для уровня дохода пос  
ле вычисления квадратного корня:

Статистика критерия 0.8556959629058838, p\_value 2.448604220134599e-28

In [22]:

```
df['sqrt4_income'] = df.Income ** (1 / 4)
print(f"Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для уровня до  
хода после вычисления квадратного корня:")
z, p = sts.shapiro(df.sqrt4_income)
```

```
print(f"Статистика критерия {z}, p_value {p}")
```

Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для уровня дохода после вычисления квадратного корня:  
Статистика критерия 0.9876828789710999, p\_value 4.7637308853154536e-07

In [23]:

```
print(f"Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для уровня дохода (первые 100):")
z, p = sts.shapiro(df.Income[:100])
print(f"Статистика критерия {z}, p_value {p}")
```

Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для уровня дохода (первые 100):  
Статистика критерия 0.8403542041778564, p\_value 5.3046971260073406e-09

In [24]:

```
import statsmodels.api as sm
from statsmodels.formula.api import ols

df_short = df.iloc[:100, :]

modell = ols("Income ~ SchoolDegree", df_short).fit()
modell.summary()
```

Out[24]:

OLS Regression Results

Dep. Variable:	Income	R-squared:	0.044
Model:	OLS	Adj. R-squared:	-0.052
Method:	Least Squares	F-statistic:	0.4561
Date:	Tue, 07 Dec 2021	Prob (F-statistic):	0.900
Time:	14:41:18	Log-Likelihood:	-1182.7
No. Observations:	100	AIC:	2385.
Df Residuals:	90	BIC:	2411.
Df Model:	9		
Covariance Type:	nonrobust		

	coef	std err	t	P> t	[0.025	0.975]
Intercept	5.85e+04	1.75e+04	3.350	0.001	2.38e+04	9.32e+04
SchoolDegree[T.associate's degree]	-3.164e+04	2.19e+04	-1.446	0.152	-7.51e+04	1.18e+04
SchoolDegree[T.bachelor's degree]	-7465.7143	1.84e+04	-0.405	0.686	-4.41e+04	2.92e+04
SchoolDegree[T.high school diploma or equivalent (GED)]	-1.369e+04	2.14e+04	-0.640	0.524	-5.62e+04	2.88e+04
SchoolDegree[T.master's degree (non-professional)]	-5777.7778	1.93e+04	-0.299	0.765	-4.41e+04	3.26e+04
SchoolDegree[T.no high school (secondary school)]	6500.0000	3.9e+04	0.166	0.868	-7.11e+04	8.41e+04
SchoolDegree[T.professional degree (MBA, MD, JD, etc.)]	-1.056e+04	2.14e+04	-0.494	0.623	-5.31e+04	3.19e+04
SchoolDegree[T.some college credit, no degree]	-8783.3333	2.02e+04	-0.436	0.664	-4.88e+04	3.13e+04
SchoolDegree[T.some high school]	-1.95e+04	2.67e+04	-0.731	0.467	-7.25e+04	3.35e+04
SchoolDegree[T.trade, technical, or vocational training]	-1.525e+04	2.47e+04	-0.618	0.538	-6.43e+04	3.38e+04

Omnibus:	48.916	Durbin-Watson:	1.740
Prob(Omnibus):	0.000	Jarque-Bera (JB):	138.918
Skew:	1.774	Prob(JB):	6.83e-31
Kurtosis:	7.556	Cond. No.	18.2

## Notes:

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

Статистика Дарбина-Уотсона стремится к 2, следовательно автокорреляция отсутствует. \ Omnibus test, **p-value=0 < 0.05** - данные гомоскедастичны. \ **Cond. No. 18.2 < 20**, мультиколлинеарность не наблюдается. \ **p\_value > 0.05**, гипотеза об отсутствии связи между образованием и уровнем дохода принимается. Взаимосвязь отсутствует, тест Тьюки не применяется.

In [25]:

```
model2 = ols("Income ~ C(Gender) + C(MaritalStatus)", df_short).fit()
print(model2.summary())

anova = sm.stats.anova_lm(model2, typ=1)
print(anova)
```

### OLS Regression Results

Dep. Variable:	Income	R-squared:	0.003
Model:	OLS	Adj. R-squared:	-0.028
Method:	Least Squares	F-statistic:	0.08715
Date:	Tue, 07 Dec 2021	Prob (F-statistic):	0.967
Time:	14:41:18	Log-Likelihood:	-1184.8
No. Observations:	100	AIC:	2378.
Df Residuals:	96	BIC:	2388.
Df Model:	3		
Covariance Type:	nonrobust		

  

			coef	std err	t
P> t	[0.025	0.975]			
-----					
Intercept			4.608e+04	1.7e+04	2.715
0.008	1.24e+04	7.98e+04			
C(Gender) [T.male]			-2976.7607	8811.301	-0.338
0.736	-2.05e+04	1.45e+04			
C(MaritalStatus) [T.married or domestic partnership]			5337.5571	1.59e+04	0.336
0.738	-2.62e+04	3.69e+04			
C(MaritalStatus) [T.single, never married]			2426.1620	1.97e+04	0.123
0.902	-3.67e+04	4.15e+04			
-----					
Omnibus:	50.101	Durbin-Watson:		1.760	
Prob(Omnibus):	0.000	Jarque-Bera (JB):		142.275	
Skew:	1.827	Prob(JB):		1.27e-31	
Kurtosis:	7.561	Cond. No.		12.9	

## Notes:

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

	df	sum_sq	mean_sq	F	PR(>F)
C(Gender)	1.0	1.266437e+08	1.266437e+08	0.106214	0.745205
C(MaritalStatus)	2.0	1.850984e+08	9.254920e+07	0.077620	0.925374
Residual	96.0	1.144646e+11	1.192339e+09	NaN	NaN

Статистика Дарбина-Уотсона стремится к 2, следовательно автокорреляция отсутствует. \ Omnibus test, **p-value=0 < 0.05** - данные гомоскедастичны. \ **Cond. No. 12.9 < 20**, мультиколлинеарность не наблюдается. \ **p\_value > 0.05**, гипотеза об отсутствии связи между уровнем дохода и параметрами принимается.

In [26]:

```
model3 = ols("Income ~ C(Gender) * C(MaritalStatus)", df_short).fit()
print(model3.summary())

anova = sm.stats.anova_lm(model3, typ=1)
print(anova)
```

### OLS Regression Results

```

=====
Dep. Variable:          Income      R-squared:          0.013
Model:                  OLS         Adj. R-squared:     -0.039
Method:                 Least Squares   F-statistic:       0.2518
Date:                  Tue, 07 Dec 2021   Prob (F-statistic): 0.938
Time:                  14:41:18         Log-Likelihood:    -1184.3
No. Observations:      100            AIC:              2381.
Df Residuals:          94             BIC:              2396.
Df Model:              5
Covariance Type:       nonrobust
=====

```

```

=====
                                coef      std
err      t      P>|t|      [0.025      0.975]
-----
Intercept                                4e+04      3.47
e+04      1.152      0.252      -2.89e+04      1.09e+05
C(Gender) [T.male]                                4625.0000      3.88e
+04      0.119      0.905      -7.24e+04      8.17e+04
C(MaritalStatus) [T.married or domestic partnership] 1.423e+04      3.58e+
04      0.398      0.692      -5.68e+04      8.53e+04
C(MaritalStatus) [T.single, never married] -1.09e+04      4.25e
+04      -0.256      0.798      -9.53e+04      7.35e+04
C(Gender) [T.male]:C(MaritalStatus) [T.married or domestic partnership] -1.104e+04      4e+
04      -0.276      0.783      -9.04e+04      6.83e+04
C(Gender) [T.male]:C(MaritalStatus) [T.single, never married] 1.827e+04      4.81e+
04      0.380      0.705      -7.71e+04      1.14e+05
=====
Omnibus:          49.257      Durbin-Watson:          1.806
Prob(Omnibus):    0.000      Jarque-Bera (JB):    136.824
Skew:             1.804      Prob(JB):           1.95e-30
Kurtosis:         7.453      Cond. No.           46.2
=====

```

## Notes:

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

```

                                df      sum_sq      mean_sq      F      \
C(Gender)          1.0  1.266437e+08  1.266437e+08  0.105108
C(MaritalStatus)   2.0  1.850984e+08  9.254920e+07  0.076811
C(Gender):C(MaritalStatus) 2.0  1.205011e+09  6.025055e+08  0.500051
Residual          94.0  1.132595e+11  1.204889e+09      NaN

                                PR(>F)
C(Gender)          0.746504
C(MaritalStatus)   0.926123
C(Gender):C(MaritalStatus) 0.608104
Residual          NaN

```

Статистика Дарбина-Уотсона стремится к **2**, следовательно автокорреляция отсутствует. \ Омнибус тест, **p-value=0 < 0.05** - данные гомоскедастичны. \ **Cond. No. 46.2 > 20**, мультиколлинеарность наблюдается. \ **p\_value > 0.05**, гипотеза об отсутствии связи между уровнем дохода и параметрами принимается.

In [27]:

```

data = pd.read_excel('countries.xlsx', sheet_name='russia')
data = data.iloc[:, :10]
data.head()

```

Out[27]:

	date	seed	brent	petrol_ru	usd	interest_ru	inflation_ru	petrol_br	interest_br	inflation_br
0	2021-12-01	59585.375500	5240.182184	145.462335	74.0191	7.50	1.11	58.198200	9.25	0.9
1	2021-11-01	58974.278400	5932.841316	169.926867	72.2724	7.50	1.11	58.717200	9.25	0.9
2	2021-10-01	53242.145160	5969.471025	156.704642	71.4372	6.75	1.11	58.042000	9.25	0.9
3	2021-09-01	51859.870920	5413.151835	155.521319	72.8676	6.75	0.60	57.960143	9.25	1.3

	date	seed	brent	petrol_ru	usd	interest_ru	inflation_ru	petrol_br	interest_br	inflation_br
4	2021-08-01	53016.130615	5195.676437	174.052402	73.5671	6.50	0.17	57.260970	9.25	0.2

In [28]:

```
for i in ['seed', 'brent', 'usd', 'petrol_ru', 'petrol_br']:
    data[i] = pd.cut(data[i], bins=[data[i].quantile(0) - 1,
                                     data[i].quantile(0.2),
                                     data[i].quantile(0.4),
                                     data[i].quantile(0.60),
                                     data[i].quantile(0.80),
                                     data[i].quantile(1) + 1],
                     labels=['very low', 'low', 'medium', 'high', 'very high'])
data.head()
```

Out[28]:

	date	seed	brent	petrol_ru	usd	interest_ru	inflation_ru	petrol_br	interest_br	inflation_br
0	2021-12-01	very high	very high	high	high	7.50	1.11	very high	9.25	0.9
1	2021-11-01	very high	very high	very high	medium	7.50	1.11	very high	9.25	0.9
2	2021-10-01	very high	very high	very high	medium	6.75	1.11	very high	9.25	0.9
3	2021-09-01	very high	very high	very high	medium	6.75	0.60	very high	9.25	1.3
4	2021-08-01	very high	very high	very high	medium	6.50	0.17	very high	9.25	0.2

In [29]:

```
print(f"Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для инфляции в России:")
z, p = sts.shapiro(data.inflation_ru)
print(f"Статистика критерия {z}, p_value {p}")
```

Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для инфляции в России :  
Статистика критерия 0.9729399681091309, p\_value 0.5111347436904907

In [30]:

```
print(f"Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для инфляции в Белоруссии:")
z, p = sts.shapiro(data.inflation_br)
print(f"Статистика критерия {z}, p_value {p}")
```

Проверка гипотезы на нормальность распределения тестом Шапиро-Уилка для инфляции в Белоруссии :  
Статистика критерия 0.9658294916152954, p\_value 0.32278209924697876

In [31]:

```
model4 = ols("inflation_ru ~ C(seed) + C(brent)+ C(petrol_ru)", data).fit()
print(model4.summary())

anova = sm.stats.anova_lm(model4, typ=1)
print(anova)
```

#### OLS Regression Results

```
=====
Dep. Variable:          inflation_ru    R-squared:                0.536
Model:                  OLS            Adj. R-squared:           0.293
Method:                 Least Squares   F-statistic:              2.211
Date:                  Tue, 07 Dec 2021 Prob (F-statistic):       0.0493
Time:                  14:41:21         Log-Likelihood:          0.71463
No. Observations:      36              AIC:                    24.57
Df Residuals:          23              BIC:                    45.16
Df Model:              12
Covariance Type:       nonrobust
=====
```

```
=====
coef      std err          t      P>|t|      [0.025      0.975]
```

```
-----
Intercept                -0.0315      0.243      -0.130      0.898      -0.534
0.471
C(seed) [T.low]           0.1515      0.222      0.682      0.502      -0.308
0.611
C(seed) [T.medium]        0.4533      0.235      1.928      0.066      -0.033
0.940
C(seed) [T.high]          0.4689      0.173      2.714      0.012      0.111
0.826
C(seed) [T.very high]     0.6221      0.286      2.178      0.040      0.031
1.213
C(brent) [T.low]          0.3155      0.212      1.487      0.151      -0.124
0.755
C(brent) [T.medium]       0.4952      0.238      2.082      0.049      0.003
0.987
C(brent) [T.high]         0.6548      0.262      2.498      0.020      0.112
1.197
C(brent) [T.very high]    0.7581      0.356      2.129      0.044      0.022
1.495
C(petrol_ru) [T.low]      -0.1321      0.169      -0.780      0.443      -0.482
0.218
C(petrol_ru) [T.medium]   -0.2834      0.187      -1.517      0.143      -0.670
0.103
C(petrol_ru) [T.high]     -0.3696      0.246      -1.505      0.146      -0.878
0.138
C(petrol_ru) [T.very high] -0.6363      0.336      -1.893      0.071      -1.332
0.059
=====
```

```
Omnibus:                0.192      Durbin-Watson:          1.414
Prob(Omnibus) :          0.909      Jarque-Bera (JB) :      0.170
Skew:                   -0.141      Prob(JB) :              0.918
Kurtosis:                2.814      Cond. No.               13.2
=====
```

Notes:

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

```

              df      sum_sq  mean_sq      F      PR(>F)
C(seed)       4.0    1.627046  0.406762  4.618297  0.006958
C(brent)       4.0    0.350369  0.087592  0.994507  0.430456
C(petrol_ru)   4.0    0.359509  0.089877  1.020449  0.417605
Residual      23.0    2.025751  0.088076      NaN      NaN
```

Статистика Дарбина-Уотсона стремится к **2**, следовательно автокорреляция отсутствует.\ Омнибус тест, **p-value=0.9 > 0.05** - данные гетероскедастичны.\ **Cond. No. 13.2 < 20**, мультиколлинеарность не наблюдается.\ **p\_value= 0.049 < 0.05**, гипотеза об отсутствии связи между уровнем инфляции и параметрами отвергается.

In [32]:

```
model5 = ols("inflation_br ~ C(seed) + C(brent) + C(petrol_br)", data).fit()
print(model5.summary())

anova = sm.stats.anova_lm(model5, typ=1)
print(anova)
```

#### OLS Regression Results

```
=====
Dep. Variable:          inflation_br      R-squared:                0.377
Model:                  OLS              Adj. R-squared:           0.092
Method:                 Least Squares    F-statistic:               1.323
Date:                  Tue, 07 Dec 2021  Prob (F-statistic):       0.272
Time:                  14:41:21          Log-Likelihood:           -14.103
No. Observations:      36              AIC:                     52.21
Df Residuals:          24              BIC:                     71.21
Df Model:              11
Covariance Type:       nonrobust
=====
```

```
=====
              coef      std err      t      P>|t|      [0.025
0.975]
```



```
-----
Intercept          -0.4135      0.410      -1.009      0.323      -1.260
0.433
C(seed) [T.low]      0.5451      0.292      1.866      0.074      -0.058
1.148
C(seed) [T.medium]    0.7891      0.363      2.176      0.040      0.041
1.537
C(seed) [T.high]     0.7959      0.440      1.809      0.083      -0.112
1.704
C(seed) [T.very high] 0.5843      0.565      1.035      0.311      -0.581
1.750
C(brent) [T.low]      0.5254      0.310      1.695      0.103      -0.114
1.165
C(brent) [T.medium]   0.5508      0.409      1.346      0.191      -0.293
1.395
C(brent) [T.high]     0.6379      0.384      1.662      0.109      -0.154
1.430
C(brent) [T.very high] 0.2566      0.391      0.657      0.517      -0.549
1.063
C(petrol_br) [T.low]  0.1820      0.359      0.506      0.617      -0.560
0.924
C(petrol_br) [T.medium] 0.1358      0.412      0.330      0.744      -0.714
0.985
C(petrol_br) [T.high] -0.0130      0.517      -0.025      0.980      -1.079
1.053
C(petrol_br) [T.very high] 0.2566      0.391      0.657      0.517      -0.549
1.063
=====
Omnibus:              0.131      Durbin-Watson:          1.755
Prob(Omnibus):        0.937      Jarque-Bera (JB):       0.338
Skew:                 -0.068      Prob(JB):               0.844
Kurtosis:             2.545      Cond. No.               3.94e+17
=====
```

#### Notes:

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

[2] The smallest eigenvalue is 3.4e-34. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

	df	sum_sq	mean_sq	F	PR(>F)
C(seed)	4.0	1.830850	0.457713	2.380746	0.079938
C(brent)	4.0	0.831339	0.207835	1.081032	0.387925
C(petrol_br)	4.0	0.151057	0.037764	0.196427	0.937830
Residual	24.0	4.614142	0.192256	NaN	NaN

Статистика Дарбина-Уотсона стремится к **2**, следовательно автокорреляция отсутствует.\ Омнибус тест, **p-value=0.9 > 0.05** - данные гетероскедастичны.\ **Cond. No. 3.94e+17 > 20**, мультиколлинеарность наблюдается.\ **p\_value 0.27 > 0.05**, гипотеза об отсутствии связи между уровнем инфляции и параметрами принимается.