



MÜQAVİLƏ ÜZRƏ TƏMİZLİK TƏKLİFİNİN HAZIRLANMASI



Bina xidməti şirkəti FMG MMC , ofis kompleksi boyunca yerləşən korporativ ofislərin təmizlənməsi üçün müqavilə bağlamaq üçün müraciət etməyi planlaşdırır. Xidmət şirkətinin çəkəcəyi xərclər bu işi yerinə yetirmək üçün lazım olan təmizlik qruplarının sayına mütənasibdir. Müxtəlif sayda heyət tərəfindən çıxarılan otaqların sayı barədə son məlumatlar mövcuddur. 53 gün ərzində iştirak edən briqadaların sayı və bu briqadalar tərəfindən təmizlənmiş otaqların sayı barədə qeydlər aparıldı.

Təmizlənmiş otaqların sayı ilə ekipajların sayı arasındakı əlaqəni modelləşdirmək və 4 və 16 ekipaj tərəfindən təmizlənmə bilən otaqların sayını proqnozlaşdırmaq üçün regressiya tənliyini inkişaf etdirmək istəyirik.

Hallar	Ekipajların sayı	Təmizlənmiş otaqlar	Hallar	Ekipajların sayı	Təmizlənmiş otaqlar
1	16	51	28	4	18
2	10	37	29	16	72
3	12	37	30	8	22
4	16	46	31	10	55
5	16	45	32	16	65
6	4	11	33	6	26
7	2	6	34	10	52
8	4	19	35	12	55
9	6	29	36	8	33
10	2	14	37	10	38
11	12	47	38	8	23
12	8	37	39	8	38
13	16	60	40	2	10
14	2	6	41	16	65
15	2	11	42	8	31
16	2	10	43	8	33
17	6	19	44	12	47
18	10	33	45	10	42
19	16	46	46	16	78
20	16	69	47	2	6
21	10	41	48	2	6
22	6	19	49	8	40
23	2	6	50	12	39
24	6	27	51	4	9
25	10	35	52	4	22
26	12	55	53	12	41
27	4	15			

Regressiya modelini nəzərdən keçirməyə başlayaq

$$Y = \beta_0 + \beta_1 x + e$$

burada Y = təmizlənmiş otaqların sayı və x = təmizlik briqadalarının sayı. R-dən regressiya çıxışı müvafiq olaraq $x = 4$ və $x = 16$ olduqda 95% proqnoz intervalları daxil olmaqla aşağıda verilmişdir.

MÜƏLLİF: İBRAHİM İSMAYİL

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import statsmodels.api as sm

# Data preprocessing
data = {
    'ekipajların_sayı': [16, 10, 12, 16, 16, 4, 2, 4, 6, 2, 12, 8, 16, 2, 2, 2, 6, 10, 16, 16, 10, 6, 2, 6, 10, 12, 4, 4, 16, 8,
    'təmizlənmiş_otaqların_sayı': [51, 37, 37, 46, 45, 11, 6, 19, 29, 14, 47, 37, 60, 6, 11, 10, 19, 33, 46, 69, 41, 19, 6, 27, 3
}
df = pd.DataFrame(data)

# Linear regression
X = df['ekipajların_sayı']
y = df['təmizlənmiş_otaqların_sayı']
X = sm.add_constant(X) # Adding intercept term
model = sm.OLS(y, X).fit()

# Display regression results
print(model.summary())

# Plot the data with the Least squares line
plt.scatter(df['ekipajların_sayı'], df['təmizlənmiş_otaqların_sayı'])
plt.plot(df['ekipajların_sayı'], model.predict(X), color='red')
plt.xlabel('ekipajların_sayı')
plt.ylabel('təmizlənmiş_otaqların_sayı')
plt.title('Ən kiçik kvadratlar xətti ilə otaq təmizləmə məlumatları')
plt.show()
```

OLS Regression Results

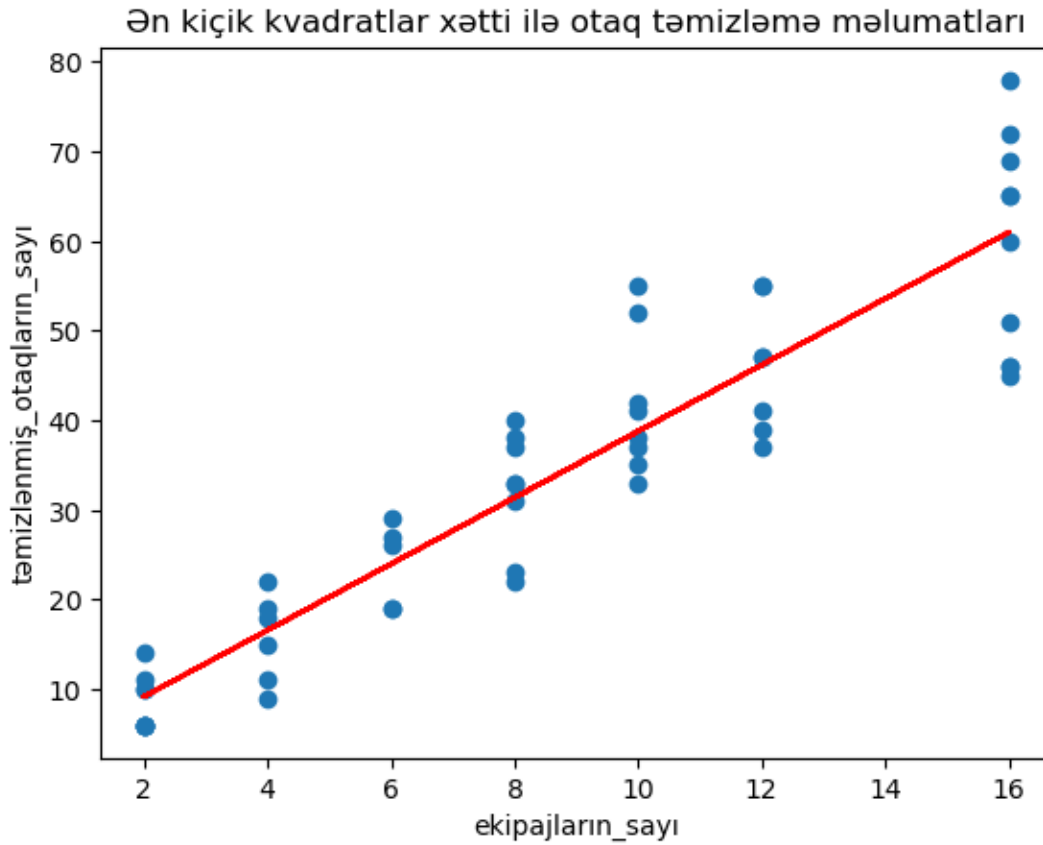
```
=====
Dep. Variable:    təmizlənmiş_otaqların_sayı    R-squared:        0.857
Model:            OLS                          Adj. R-squared:    0.854
Method:           Least Squares                F-statistic:      305.3
Date:             Thu, 11 Jul 2024              Prob (F-statistic): 3.55e-23
Time:             23:00:07                     Log-Likelihood:   -179.80
No. Observations: 53                          AIC:              363.6
Df Residuals:     51                          BIC:              367.5
Df Model:         1
Covariance Type:  nonrobust
=====
```

	coef	std err	t	P> t	[0.025	0.975]
const	1.7847	2.096	0.851	0.399	-2.424	5.994
ekipajların_sayı	3.7009	0.212	17.472	0.000	3.276	4.126

```
=====
Omnibus:            0.241    Durbin-Watson:           1.699
Prob(Omnibus):      0.886    Jarque-Bera (JB):           0.012
Skew:               0.030    Prob(JB):                  0.994
Kurtosis:           3.042    Cond. No.                   20.8
=====
```

Notes:

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



Aşağıdakı qrafikdə təmizlik briqadalarının sayına qarşı standartlaşdırılmış qalıqların qrafiki verilmişdir. Qrafikdən aydın olur ki, standartlaşdırılmış qalıqlardakı dəyişkənlik ekipajların sayı ilə artmağa meyllidir. Beləliklə, xətlərin dispersiyasının sabit olduğu fərziyyəsi bu halda pozulmuş kimi görünür.

MÜƏLLİF: İBRAHİM İSMAYİL

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import statsmodels.api as sm

# Data
data = {
    'Hallar': list(range(1, 54)),
    'Ekipajlar': [16, 10, 12, 16, 16, 4, 2, 4, 6, 2, 12, 8, 16, 2, 2, 2, 6, 10, 16, 16, 10, 6, 2, 6, 10, 12, 4, 4, 16, 8, 10, 16,
    'Otaqlar': [51, 37, 37, 46, 45, 11, 6, 19, 29, 14, 47, 37, 60, 6, 11, 10, 19, 33, 46, 69, 41, 19, 6, 27, 35, 55, 15, 18, 72,
}

df = pd.DataFrame(data)

# Regression
X = df['Ekipajlar']
Y = df['Otaqlar']
X = sm.add_constant(X) # adding a constant

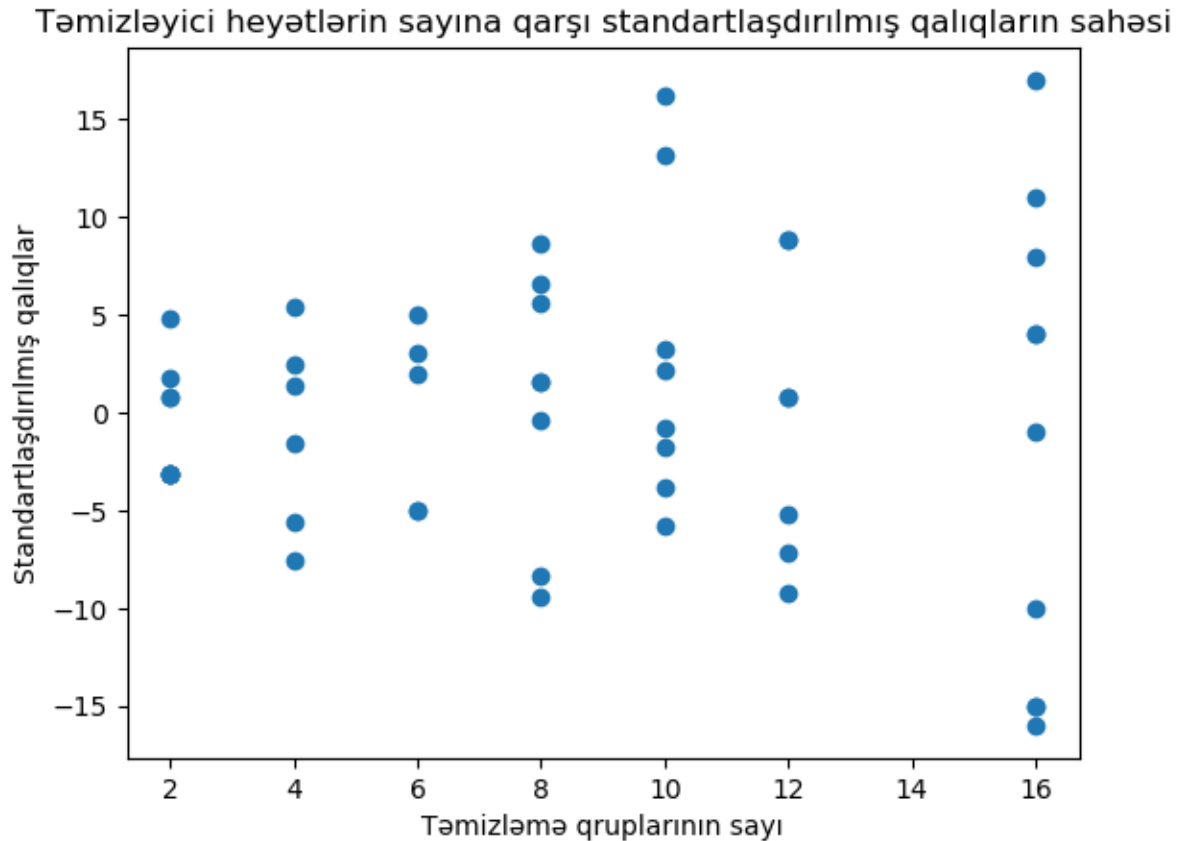
model = sm.OLS(Y, X).fit()
predictions = model.predict(X)

print_model = model.summary()
print(print_model)

# Plot residuals
residuals = model.resid
plt.scatter(df['Ekipajlar'], residuals)
plt.xlabel('Təmizləmə qruplarının sayı')
plt.ylabel('Standartlaşdırılmış qalıqlar')
plt.title('Təmizləyici heyətlərin sayına qarşı standartlaşdırılmış qalıqların sahəsi')
plt.show()
```

OLS Regression Results

```
=====
Dep. Variable:          Otaqlar    R-squared:            0.857
Model:                  OLS        Adj. R-squared:       0.854
Method:                 Least Squares    F-statistic:        305.3
Date:                  Thu, 11 Jul 2024    Prob (F-statistic):  3.55e-23
Time:                  23:09:39          Log-Likelihood:     -179.80
No. Observations:      53              AIC:               363.6
Df Residuals:          51              BIC:               367.5
Df Model:               1
Covariance Type:       nonrobust
=====
               coef    std err          t      P>|t|      [0.025     0.975]
-----
const          1.7847      2.096      0.851     0.399     -2.424      5.994
Ekipajlar      3.7009      0.212     17.472     0.000      3.276      4.126
=====
Omnibus:            0.241    Durbin-Watson:           1.699
Prob(Omnibus):      0.886    Jarque-Bera (JB):           0.012
Skew:               0.030    Prob(JB):                 0.994
Kurtosis:           3.042    Cond. No.                  20.8
=====
```



Aşağıdakı qrafikdə standartlaşdırılmış qalıqların x -ə qarşı mütləq qiymətinin kvadrat kökünün qrafiki verilmişdir. Qrafikdə ən kiçik kvadratlar reqressiya xətti əlavə edildi. Qrafikdə artan tendensiyanın aydın sübutu var ki, bu da xətlərin dispersiyasının x ilə artdığına dair sübutların olduğunu göstərir.

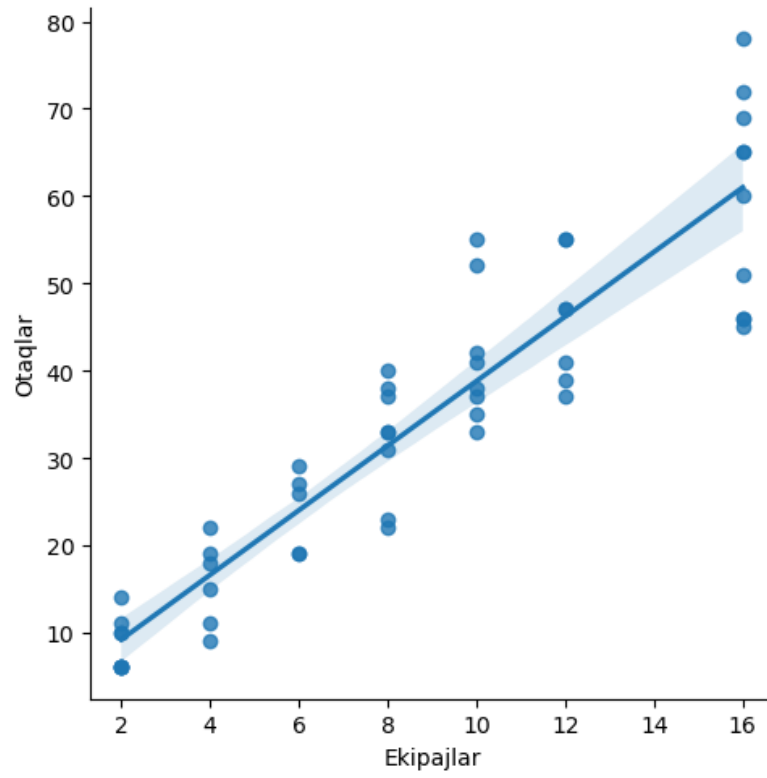
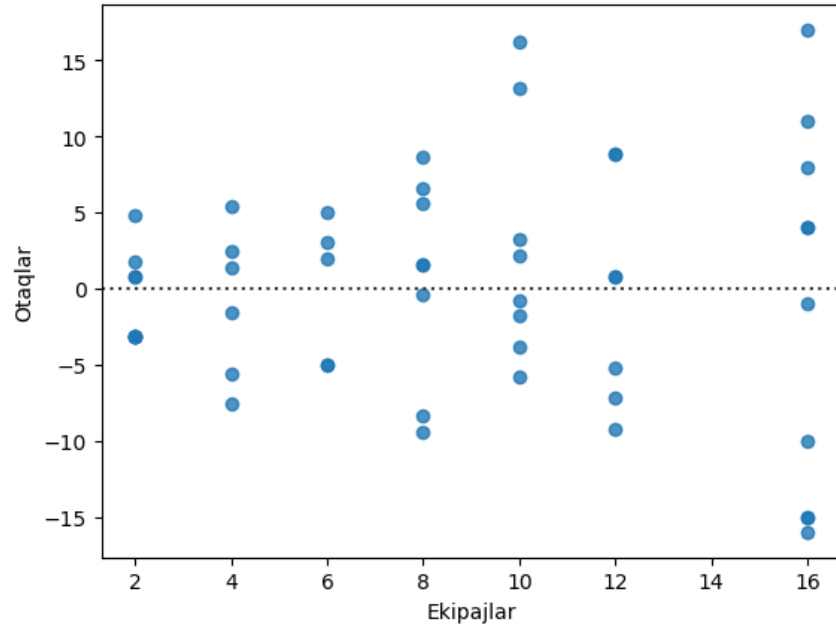
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Create a DataFrame from the provided data
data = {
    'Həllər': list(range(1, 54)),
    'Ekipajlar': [16, 10, 12, 16, 16, 4, 2, 4, 6, 2, 12, 8, 16, 2, 2, 2, 6, 10, 16, 16, 10, 6, 2, 6, 10, 12, 4, 4, 16, 8, 10, 16,
    'Otaqlar': [51, 37, 37, 46, 45, 11, 6, 19, 29, 14, 47, 37, 60, 6, 11, 10, 19, 33, 46, 69, 41, 19, 6, 27, 35, 55, 15, 18, 72,
}
df = pd.DataFrame(data)

# Plot a diagnostic plot to detect nonconstant error variance
sns.residplot(x='Ekipajlar', y='Otaqlar', data=df)
plt.show()

# Regression analysis
result = df[['Otaqlar', 'Ekipajlar']].sort_values(by='Ekipajlar') # Sorting the data by Crews for analysis
sns.lmplot(x='Ekipajlar', y='Otaqlar', data=result) # Visualizing the linear regression model
plt.show()

# Additional statistical analysis can be performed using the statsmodels or scikit-Learn libraries in Python
```



MÜƏLLİF: İBRAHİM İSMAYİL

Aşağıdakı qrafik tərəfindən hazırlanmış diaqnostik qrafik var. Standartlaşdırılmış qalıqların mütləq dəyərinin uyğun dəyərlərə qarşı kvadrat kökünün qrafikidir. Bu qrafikə əlavə olunan xətt qeyri-parametrik hamarlama üsulu ilə əldə edilən xəttidir. Uyğun qiymətlər $\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$ ilə verildiyi üçün bu qrafikin forması ilə yuxarıdakı göstərilənlər eynidir.

```
import matplotlib.pyplot as plt
import pandas as pd
import statsmodels.api as sm

# Creating a DataFrame with the data provided
data = {
    'Hallar': [i for i in range(1, 54)],
    'Ekipajlar': [16, 10, 12, 16, 16, 4, 2, 4, 6, 2, 12, 8, 16, 2, 2, 2, 6, 10, 16, 16, 10, 6, 2, 6, 6, 10, 12, 4, 4, 16, 8, 10, 16,
    'Otaqlar': [51, 37, 37, 46, 45, 11, 6, 19, 29, 14, 47, 37, 60, 6, 11, 10, 19, 33, 46, 69, 41, 19, 6, 27, 35, 55, 15, 18, 72,
}

df = pd.DataFrame(data)

# Performing linear regression
X = df['Ekipajlar']
y = df['Otaqlar']
X = sm.add_constant(X) # adding a constant

model = sm.OLS(y, X).fit()
predictions = model.predict(X)

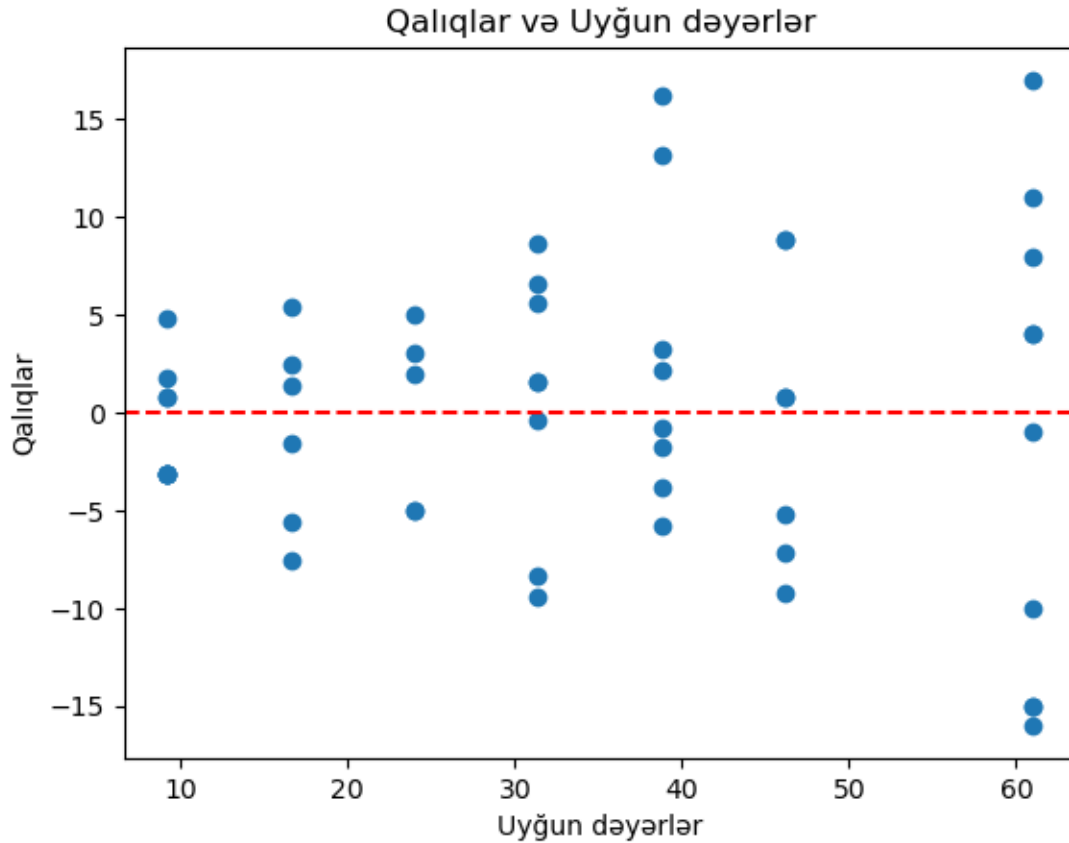
# Regression output
print(model.summary())

# Plotting Residuals vs Fitted
plt.scatter(predictions, model.resid)
plt.axhline(y=0, color='r', linestyle='--')
plt.xlabel('Uyğun dəyərlər')
plt.ylabel('Qalıqlar')
plt.title('Qalıqlar və Uyğun dəyərlər')
plt.show()
```

OLS Regression Results

Dep. Variable:	Otaqlar	R-squared:	0.857			
Model:	OLS	Adj. R-squared:	0.854			
Method:	Least Squares	F-statistic:	305.3			
Date:	Thu, 11 Jul 2024	Prob (F-statistic):	3.55e-23			
Time:	23:30:29	Log-Likelihood:	-179.80			
No. Observations:	53	AIC:	363.6			
Df Residuals:	51	BIC:	367.5			
Df Model:	1					
Covariance Type:	nonrobust					
=====						
	coef	std err	t	P> t	[0.025	0.975]

const	1.7847	2.096	0.851	0.399	-2.424	5.994
Ekipajlar	3.7009	0.212	17.472	0.000	3.276	4.126
=====						
Omnibus:	0.241	Durbin-Watson:	1.699			
Prob(Omnibus):	0.886	Jarque-Bera (JB):	0.012			
Skew:	0.030	Prob(JB):	0.994			
Kurtosis:	3.042	Cond. No.	20.8			



Bu nümunədə x -dəyişən (ekipajların sayı) 2, 4, 6, 8, 10, 12 və 16 dəyərləri ilə diskretdir. Diqqət yetirin ki, hər x dəyərində Y -dəyişəninin (təmizlənmiş otaqların sayı) çoxlu ölçüləri var. Bu xüsusi halda, x -in hər bir diskret qiymətində Y -nin standart kənarlaşmasını birbaşa hesablamaq mümkündür. Yuxarıdakı Cədvəldən aydın olur ki, təmizlənmiş otaqların sayının standart kənarlaşması x , briqadaların sayı artdıqca artır.