

Измерения волокна

1 декабря 2021 г.

Нужные штуки: 1) Средневзвешенный диаметр по длине

2) Среднеквадратичное квадратичное отклонение по длине

3) График средних по проекциям от длины

4) График отклонения по проекция от длины

5) Относительное среднее значение некруглости по длине (в процентах)

6) График средней некруглости от длины

```
[1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
[2]: path = "Герборт"
csvfile = "2021-11-24 15-56-27-2932.csv"
DF = pd.read_csv(path+"\\ "+csvfile, sep = ";", decimal = ",").drop(columns =
→["UpPointY_PX", "DownPointY_PX"])
```

```
[3]: DF
```

```
[3]:
```

| | X_MM | Fi | Diameter |
|-----|------|-------|----------|
| 0 | 0 | 0.0 | 151.800 |
| 1 | 1 | 0.0 | 151.423 |
| 2 | 2 | 0.0 | 146.302 |
| 3 | 3 | 0.0 | 153.255 |
| 4 | 4 | 0.0 | 150.452 |
| ... | ... | ... | ... |
| 499 | 23 | 170.0 | 155.466 |
| 500 | 24 | 170.0 | 142.581 |
| 501 | 25 | 170.0 | 150.129 |
| 502 | 26 | 170.0 | 151.153 |
| 503 | 27 | 170.0 | 172.500 |

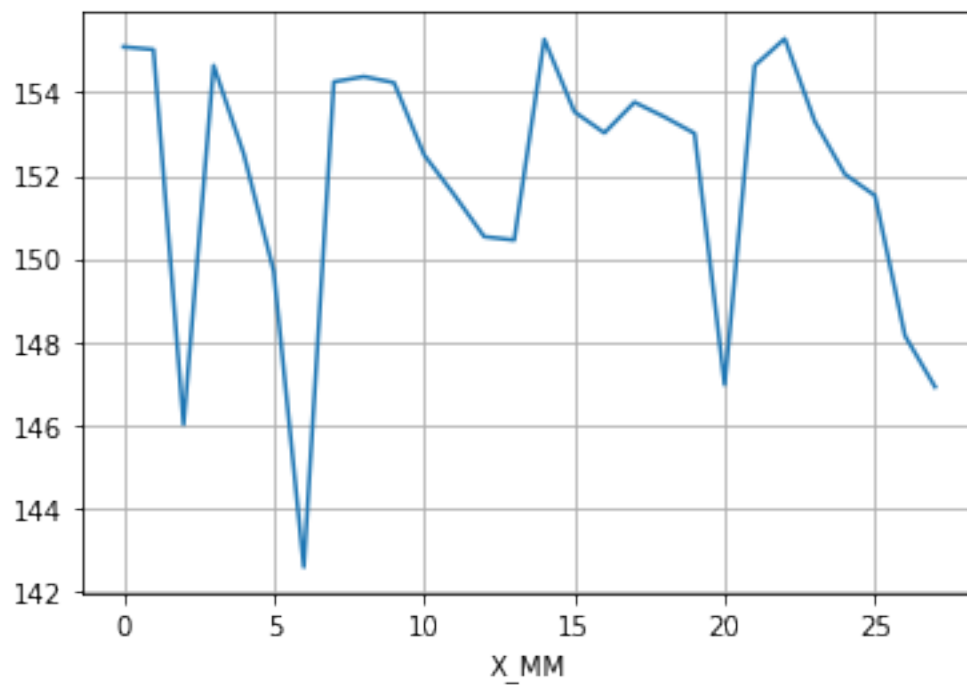
[504 rows x 3 columns]

```
[4]: means = DF.groupby("X_MM")["Diameter"].mean() # Группируем по длине, считаем
→среднее значение по всем проекциям
```

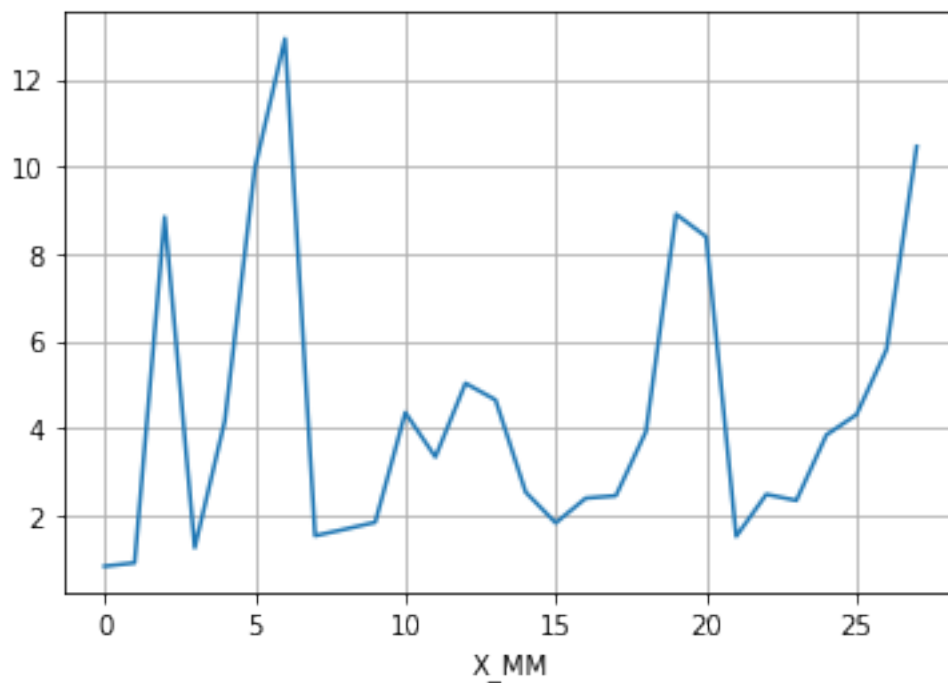
```
[5]: means
```

```
[5]: X_MM
0      155.091333
1      155.026167
2      146.023278
3      154.645000
4      152.524667
5      149.703667
6      142.600056
7      154.252000
8      154.377000
9      154.231833
10     152.500833
11     151.542389
12     150.533167
13     150.452278
14     155.272722
15     153.533278
16     153.024833
17     153.761611
18     153.408167
19     153.012833
20     146.978389
21     154.640722
22     155.287500
23     153.297389
24     152.033500
25     151.521444
26     148.152333
27     146.924444
Name: Diameter, dtype: float64
```

```
[6]: means.plot() # График средних по длине
plt.grid()
```



```
[7]: DF.groupby("X_MM")["Diameter"].std().plot() # График среднеквадратичного
      ↪ отклонения по длине
      plt.grid()
```



```
[8]: means.mean() # Средневзвешенный диаметр по длине волокна
```

```
[8]: 151.94117261904765
```

```
[9]: means.std() # Среднеквадратичное отклонение по длине
```

```
[9]: 3.229794845783249
```

Значение некруглости по формуле $(D_{\max} - D_{\min}) / 2$

(Источник: http://ng.sibstrin.ru/wolchin/umm/eskd/eskd/GOST/2_308/24642/0012.gif
http://ng.sibstrin.ru/wolchin/umm/eskd/eskd/GOST/2_308/001.htm)

```
[10]: nekr = (DF.groupby("X_MM")["Diameter"].max() - DF.groupby("X_MM")["Diameter"] .  
→min()) / 2
```

```
[11]: nekr
```

```
[11]: X_MM  
0      1.8060  
1      1.9675  
2     17.5560  
3      2.5605  
4      7.7630  
5     18.0055  
6     20.9700  
7      1.9740  
8      2.6885  
9      3.2345  
10     7.2775  
11     6.2265  
12    10.5925  
13     8.8410  
14     3.8330  
15     2.5200  
16     3.2345  
17     4.2050  
18     7.8700  
19    19.6760  
20    16.8730  
21     2.0240  
22     4.1105  
23     3.6790  
24     6.5500  
25     8.0860  
26    10.4040  
27    23.3955
```

Name: Diameter, dtype: float64

```
[12]: nekr.mean() # Абсолютное значение некруглости по длине
```

```
[12]: 8.140125000000001
```

```
[13]: nekr.mean()/means.mean() * 100 # Значение некруглости обычно берется в процентах,
      ↪ от среднего диаметра. Обычно допуск - 6%.
```

```
[13]: 5.357418834991628
```

```
[14]: (nekr / means.mean() * 100).plot() # График некруглостей в процентах
      plt.grid()
```

