Task3

October 13, 2018

```
In [29]: import numpy as np
         import pandas as pd
```

Считаем данные из файла "Weibull.csv"

```
In [30]: weibull = pd.read_csv('Weibull.csv', header=None, names=["data"])
```

Посмотрим на первые несколько из них

```
In [31]: weibull.head(10)
```

```
Out[31]:    data
         0  0.86
         1  0.02
         2  3.06
         3  0.09
         4  0.08
         5  1.12
         6  0.69
         7  0.06
         8  0.91
         9  2.39
```

Выведем также общую статистику по данным

```
In [32]: weibull.describe()
```

```
Out[32]:               data
         count  3652.000000
         mean      1.074285
         std       1.291072
         min       0.000000
         25%       0.220000
         50%       0.630000
         75%       1.450000
         max      14.300000
```

Посчитаем оценку максимального правдоподобия. Для этого напишем функцию плотности распределения Вейбулла p_γ, которая равная $\gamma x^{\gamma-1} e^{-x^\gamma}$. В нее также будем передавать параметр γ.

```
In [33]: def p(x, gamma):
             return gamma * (x ** (gamma - 1)) * np.exp(-x ** gamma)
```

```
In [34]: sample = np.array(weibull.values)
```

Как можно заметить, в нашей выборке существуют нулевые значения, поэтому увеличим их на величину 0.001, чтобы функция правдоподобия не обращалась всегда в ноль. Так, как точность наших измерений равна 10^{-2}, то при увеличении значения на 10^{-3} мы не получим значения, которое уже было в нашей выборке.

```
In [35]: sample = np.array([data if data > 0 else data + 0.001 for data in sample])
```

Оценим параметр формы методом максимального правдоподобия, то есть найдем: $\hat{\gamma}(X1,\ldots,X_n) = \underset{\gamma}{argmax}\ f_\gamma(X_1,\ldots,X_n) = \underset{\gamma}{argmax}\ \prod_{i=0}^{n} p_\gamma(X_i)$

Для начала возьмем первые 4 года. Будем искать $\hat{\gamma}$ по логарифмической сетке $log_{10}\gamma \in [-2,2]$ с шагом 10^{-3}.

```
In [36]: gammas = np.logspace(-2, 2, 4 * 1000 + 1)
         max_estimator = gammas[0]
         max_likelihood_function = np.sum(p(sample[:4 * 365], gammas[0]))
         for gamma in gammas[1:]:
             current_likelihood_function = np.sum(p(sample[:4 * 365], gamma))
             if current_likelihood_function > max_likelihood_function:
                 max_likelihood_function = current_likelihood_function
                 max_estimator = gamma
         print("Оценка максимального правдоподобия -", max_estimator)
```

```
Оценка максимального правдоподобия - 0.26853444456585074
```

И возьмем всю выборку

```
In [37]: gammas = np.logspace(-2, 2, 4 * 1000 + 1)
         max_estimator = gammas[0]
         max_likelihood_function = np.sum(p(sample, gammas[0]))
         for gamma in gammas[1:]:
             current_likelihood_function = np.sum(p(sample, gamma))
             if current_likelihood_function > max_likelihood_function:
                 max_likelihood_function = current_likelihood_function
                 max_estimator = gamma
         print("Оценка максимального правдоподобия -", max_estimator)
```

```
Оценка максимального правдоподобия - 0.27415741719278824
```

Вывод: Как видно из результатов, оценка максимального правдоподобия параметра γ почти не отличается для половины и всей выборки. Таким образом мы смогли оценить параметр формы в распределении Вейбулла методом максимального правдоподобия.