### Отчет по дисциплине “Математическая статистика”

##### Описательная статистика

Выполнил: Султанов Нурсиль

Ильнарович 09-012

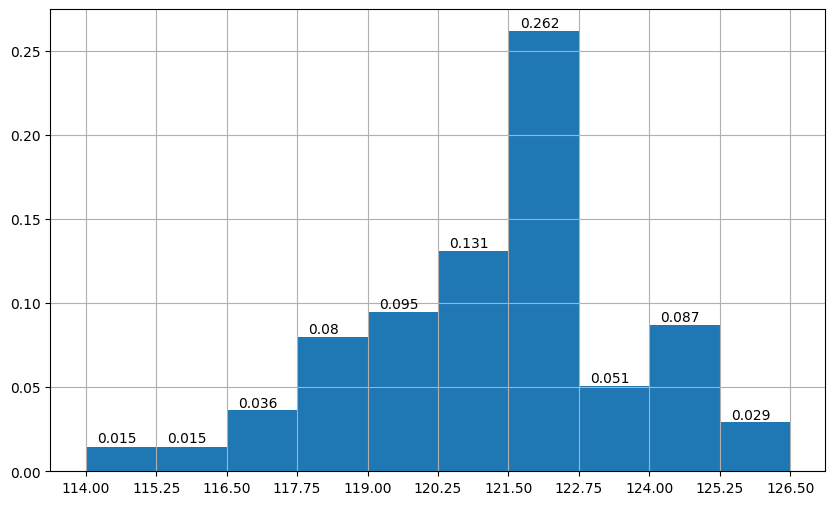
Задание 1. Раздел 1.

Цель работы: произвести базовое статистическое исследование предоставленных данных: моментные и базовые характеристики выборки, гистограмма, эмпирическая функция распределения.

1.1. Характеристики выборки

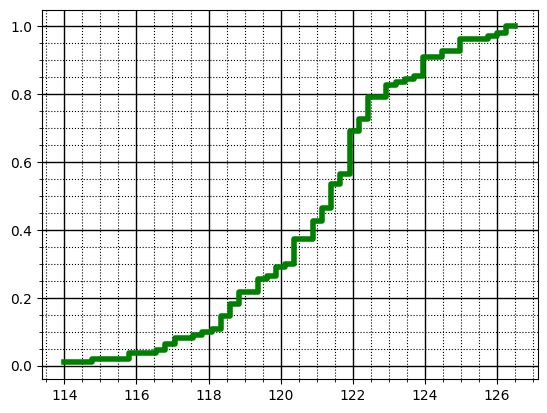
|  |  |
| --- | --- |
| **Характеристика** | **Значение** |
| Объём выборки | 110 |
| Минимум | 114 |
| Максимум | 126.5 |
| Размах | 12.5 |
| Среднее | 121.2 |
| Смещённая выборочная дисперсия | 5.8 |
| Несмещенная выборочная дисперсия | 5.9 |
| Стандартное отклонение | 2.4 |
| Асимметрия | -0.4 |
| Медиана | 121.5 |
| Интерквартильная широта | 3 |

1.2 Посторение гистограммы



Оценка моды распределения: 121.5

1.3. Построение эмпирической функции распределения.



1.4 Код программы.

import csv

import math

import numpy as np

import matplotlib.pylab as plt

import statsmodels.api as sm

a=[]

with open('r1z1.csv', newline='') as csvfile:

reader = csv.reader(csvfile,quoting=csv.QUOTE\_NONNUMERIC)

for row in reader:

a.append(row[0])

a.pop(0) #удаление название столбца типа str

a.sort()

print(a)

n=len(a)

print("Объем выборки: ", len(a))

print("Минимум: ", a[0])

print("Максимум: ", a[n-1])

print("Размах: ", round(a[n-1] - a[0],1))

s=0

for i in a:

s += i

mean\_value = round(s / n, 1)

print("Среднее: ", mean\_value)

s = 0

for i in a:

s += (i - mean\_value)\*\*2

sample\_variance = round(s / n, 1)

print("Выборочная дисперсия:", sample\_variance)

unbiased\_variance = round(n\*sample\_variance/(n-1), 1)

print("Несмещённая выборочная дисперсия:", unbiased\_variance)

sigma = round(math.sqrt(sample\_variance), 1)

print("Стандартное отклонение:", sigma)

s = 0

for i in a:

s += (i - mean\_value)\*\*3

asymmetry = round(s/(n \* sigma\*\*3), 1)

print("Ассиметрия:", asymmetry)

if n % 2 == 0:

print("Медиана:", (round((a[n//2-1] + a[n//2])/2, 1)))

else:

print("Медиана:", a[n // 2])

if n % 4 == 0:

nkvart = round((a[n//4-1] + a[n//4])/2, 1)

vkvart = round((a[3\*n//4-1] + a[3\*n//4])/2, 1)

else:

nkvart = a[n // 4]

vkvart = a[3 \* n // 4]

print("Интерквартильная широта:", vkvart - nkvart)

fig=plt.figure(figsize=(10,6))

ax=fig.add\_subplot()

ax.xaxis.set\_major\_locator(plt.IndexLocator(base=1.25, offset=0))

n, bins,pitches = ax.hist(a,bins=int(np.round(len(a)/10))-1, density=True)

for i in pitches:

height = round(i.get\_height(), 3)

ax.annotate(height, (i.get\_x() + 0.2, height+0.002))

ax.grid()

plt.show()

ecdf = sm.distributions.ECDF(a,)

x = np.linspace(min(a), max(a))

y = ecdf(x,)

plt.step(x, y, color = 'g',

linewidth = 4)

plt.minorticks\_on()

# Определяем внешний вид линий основной сетки:

plt.grid(which='major',

color = 'k',

linewidth = 1)

# Определяем внешний вид линий вспомогательной

# сетки:

plt.grid(which='minor',

color = 'k',

linestyle = ':')

plt.show()