

**Липецкий государственный технический университет**

**Факультет автоматизации и информатики**

**Кафедра автоматизированных систем управления**

**ЛАБОРАТОРНАЯ РАБОТА №1**

**по дисциплине**

**«Прикладные интеллектуальные системы и экспертные системы»**

Студент

Мамедов Р. В.

Группа М-ИАП-23-1

Руководитель

Кургасов В. В.

доцент, канд. пед. наук

Липецк 2023 г.

### Задание кафедры

Задать значения количества продаж по 10 товарам в течение 12 месяцев (помесячно). Для каждого из товаров спрогнозировать количество продаж на следующий, 13-ый месяц и провести анализ достоверности планирования продаж.

## Ход работы

### Импортируем необходимые модули и библиотеки

```
import numpy as np
import pandas as pd
import seaborn
import random
```

Напишем на языке Python функцию для генерации последовательности заданного количество целых чисел из заданного диапазона.

```
def random_sales(min, max):
    li = []
    for i in range(0, 12):
        element = random.randint(min, max)
        li.append(element)
    return li
```

Сгенерируем и выведем количество продаж за год по месяцам для 10-ти товаров.

```
sales = pd.DataFrame({
    'iPhone': random_sales(5, 15),
    'Samsung Galaxy': random_sales(8, 20),
    'Google Pixel': random_sales(3, 10),
    'OnePlus': random_sales(5, 15),
    'Xiaomi Redmi': random_sales(10, 25),
    'Huawei P Series': random_sales(7, 18),
    'Sony Xperia': random_sales(3, 12),
    'LG G Series': random_sales(2, 8),
    'Motorola Moto': random_sales(4, 12),
    'Nokia Lumia': random_sales(1, 5)
})
sales
```

12 rows × 10 columns <code>pd.DataFrame</code>							SQL Inserts			
	iPhone	Samsung Galaxy	Google Pixel	OnePlus	Xiaomi Redmi	Huawei P				
0	7	15	7	8	19					
1	5	12	10	7	15					
2	6	14	3	6	21					
3	9	8	10	13	25					
4	11	14	8	6	11					
5	11	20	9	14	11					
6	12	12	7	14	19					
7	12	10	4	5	19					
8	12	16	3	9	25					
9	9	11	5	9	18					

Рисунок 1 — Количество продаж продуктов по месяцам

Построим график, отображающий нашу выборку

```
lp = seaborn.lineplot(sales)
seaborn.move_legend(lp, 'upper left', bbox_to_anchor=(1, 1))
```

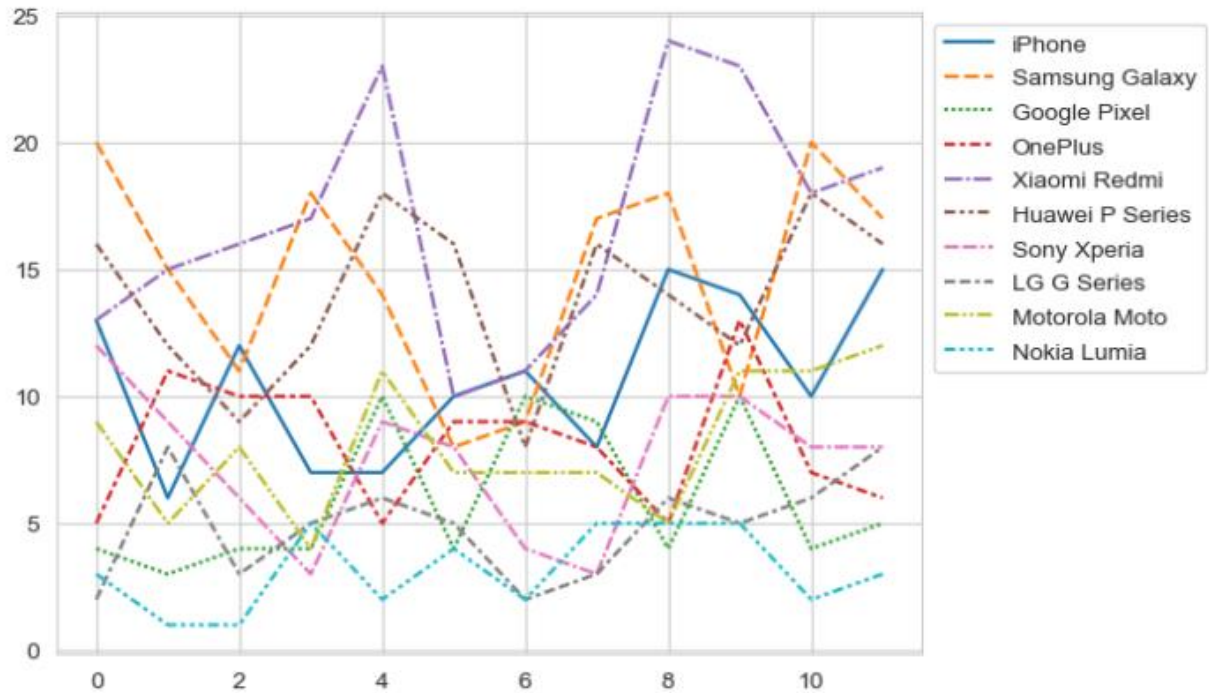


Рисунок 2 — Выборка данных на графике

Рассчитаем  $\hat{p}_0$

```
p0 = sales.sum() / sales.shape[0]
print(p0)
```

iPhone	10.666667
Samsung Galaxy	14.750000
Google Pixel	5.916667
OnePlus	8.166667
Xiaomi Redmi	16.916667
Huawei P Series	13.916667
Sony Xperia	7.500000
LG G Series	4.916667
Motorola Moto	8.083333
Nokia Lumia	3.166667
dtype: float64	

Рисунок 3 — Расчёт  $\hat{p}_0$

Рассчитаем планируемые показатели продаж на следующий месяц.

```
predict = pd.DataFrame([p0 + np.random.normal(0, std, len(p0))])
print(predict)
```

	iPhone	Samsung Galaxy	Google Pixel	OnePlus	Xiaomi Redmi \
0	10.071394	15.466663	12.167735	6.772711	14.401207

	Huawei P Series	Sony Xperia	LG G Series	Motorola Moto	Nokia Lumia
0	13.580086	4.916197	4.142667	7.789315	2.43823

Рисунок 4 — Планируемые показатели продаж

Проведем расчёт отношения  $\text{std} / p_0$ .

```
square_std = ((sales - p0) ** 2).sum() / (sales.shape[0] - 1)
std = square_std ** 0.5
reliability = std / p0
print(reliability)
```

iPhone	0.308462
Samsung Galaxy	0.233156
Google Pixel	0.416151
OnePlus	0.374547
Xiaomi Redmi	0.266414
Huawei P Series	0.272560
Sony Xperia	0.358183
LG G Series	0.523776
Motorola Moto	0.287352
Nokia Lumia	0.400201
dtype:	float64

Рисунок 5 — Расчёт отношения (оценки)

Введем условия, требующиеся для «раскраски» показателей.

```
condition1 = ((sales - p0) < 2 * std).all()
condition2 = p0 > 2 * std
condition3 = (sales > 0).all()
```

```
print('Условие 1: ')
print(condition1)
print('\n')
print('Условие 2: ')
print(condition2)
print('\n')
print('Условие 3: ')
print(condition3)
```

Условие 1:		Условие 2:		Условие 3:	
iPhone	True	iPhone	True	iPhone	True
Samsung Galaxy	True	Samsung Galaxy	True	Samsung Galaxy	True
Google Pixel	True	Google Pixel	True	Google Pixel	True
OnePlus	True	OnePlus	True	OnePlus	True
Xiaomi Redmi	True	Xiaomi Redmi	True	Xiaomi Redmi	True
Huawei P Series	True	Huawei P Series	True	Huawei P Series	True
Sony Xperia	True	Sony Xperia	True	Sony Xperia	True
LG G Series	True	LG G Series	True	LG G Series	True
Motorola Moto	True	Motorola Moto	True	Motorola Moto	True
Nokia Lumia	True	Nokia Lumia	False	Nokia Lumia	True
dtype: bool		dtype: bool		dtype: bool	

Рисунок 6 — Условия

Добавим цвет товарам в зависимости от условий, перечисленных выше.

```
product_color = pd.Series(dtype='string')
for product in sales.columns:
```

```

if not condition3[product]:
    product_color[product] = 'Red'
elif not condition1[product] and not condition2[product]:
    product_color[product] = 'Orange'
elif not condition1[product] or not condition2[product]:
    product_color[product] = 'Yellow'
else:
    product_color[product] = 'Green'

print(product_color)

```

iPhone	Green
Samsung Galaxy	Green
Google Pixel	Green
OnePlus	Green
Xiaomi Redmi	Green
Huawei P Series	Green
Sony Xperia	Green
LG G Series	Green
Motorola Moto	Green
Nokia Lumia	Yellow
dtype: object	

Рисунок 7 — Цветовые метки товаров

#### Вывод

В ходе лабораторной работы мы спрогнозировали объем продаж товаров на 13-ый месяц и провели анализ достоверности планирования продаж.

## Вывод

В ходе лабораторной работы мы спрогнозировали объем продаж товаров на 13-ый месяц и провели анализ достоверности планирования продаж.