

Липецкий государственный технический университет

Факультет автоматизации и информатики

Кафедра автоматизированных систем управления

ЛАБОРАТОРНАЯ РАБОТА №1

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

Экспертные системы. Оценка планирования продаж.

Студент

Осипов А.А.

Группа М-ИАП-22

Руководитель

Кургасов В.В.

Липецк 2022 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 месяц и провести анализ достоверности планирования продаж.

Ход работы

Используемые библиотеки:

- numpy библиотека для языка python предназначена для поддержки многомерных массивов (включая матрицы), поддержки высокоуровневых математических функций;
- pandas библиотека предназначена для обработки и анализа массивов данных.

```
import numpy as numpy
import pandas as pandas
```

Рисунок 1 – Импорт библиотек

Выборка

```
data = {
    'Песок': numpy.random.normal(69, 7, 12),
    'Цемент': numpy.random.normal(123, 17, 12),
    'Щебенка': numpy.random.normal(77, 7, 12),
    'Земля': numpy.random.normal(50, 5, 12),
    'Вода': numpy.random.normal(100, 15, 12),
    'Гравий': numpy.random.normal(95, 9, 12),
    'Кирпичи': numpy.random.normal(33, 4, 12),
    'Газоблоки': numpy.random.normal(111, 21, 12),
    'Доски': numpy.random.normal(112, 22, 12),
    'Сваи': numpy.random.normal(70, 15, 12),
}
```

Рисунок 2 – Генерация данных

Получение p_0

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

Песок	69.464207
Цемент	117.338473
Щебенка	75.796389
Земля	48.139080
Вода	100.484333
Гравий	98.118221
Кирпичи	32.746996
Газоблоки	108.526260
Доски	106.719839
Сваи	72.487965

dtype: float64

Рисунок 3 – Расчёт \hat{p}_0

Получение среднеквадратичного отклонения

```
sr = ((table - p0) ** 2).sum() / (table.shape[0] - 1)  
sr = sr ** (1 / 2)
```

Рисунок 4 – Расчет среднеквадратичного отклонения

Значение планируемого показателя на 13-ый месяц

```
table2 = p0 + numpy.random.normal(0, sr, len(p0))  
columns = table2.index  
pandas.concat([table, pandas.DataFrame([table2], columns=columns)]).reset_index(drop=True)  
table2
```

Песок	65.753872
Цемент	121.578611
Щебенка	74.231374
Земля	42.359353
Вода	93.491863
Гравий	91.255555
Кирпичи	30.882569
Газоблоки	120.937035
Доски	89.848868
Сваи	73.359246

dtype: float64

Рисунок 5 – Расчёт планируемого показателя

Оценки

```
reliability = sr / p0  
reliability
```

```
Песок      0.053797  
Цемент     0.131384  
Щебенка    0.075674  
Земля      0.049780  
Вода       0.116477  
Гравий     0.107105  
Кирпичи    0.090368  
Газоблоки  0.230648  
Доски      0.225880  
Сваи       0.157859  
dtype: float64
```

Рисунок 6 – Расчёт $\frac{\hat{\sigma}_p}{\hat{p}_0}$

Уловие 1

```
cond1 = ((table - p0) < 2 * sr).all()  
cond1
```

```
Песок      False  
Цемент     True  
Щебенка    True  
Земля      False  
Вода       True  
Гравий     False  
Кирпичи    True  
Газоблоки  True  
Доски      True  
Сваи       True  
dtype: bool
```

Рисунок 7 – Проверка первого условия

Условие 2

```
cond2 = p0 > 2 * sr  
cond2
```

```
Песок      True  
Цемент     True  
Щебенка    True  
Земля      True  
Вода       True  
Гравий     True  
Кирпичи    True  
Газоблоки  True  
Доски      True  
Сваи       True  
dtype: bool
```

Рисунок 8 – Проверка второго условия

Условие 3

```
cond3 = (table > 0).all()
cond3
```

```
Песок      True
Цемент     True
Щебенка    True
Земля      True
Вода       True
Гравий     True
Кирпичи    True
Газоблоки  True
Доски      True
Сваи       True
dtype: bool
```

Рисунок 9 – Проверка третьего условия

```
colClas = pandas.DataFrame({'keyword':table.columns,
                             'color': 'green'})
i = 0
table.columns
for name in table.columns:
    if (not cond3[name]):
        colClas.color[i] = 'red'
    elif (not cond1[name] and not cond2[name]):
        colClas.color[i] = 'orange'
    elif (not cond1[name] or not cond2[name]):
        colClas.color[i] = 'yellow'
    i += 1

new_dict = pandas.Series(colClas.color.values,index=colClas.keyword).to_dict()
colClas.style.applymap(lambda v: f"background-color: {new_dict.get(v, 'None')}")
```

	keyword	color
0	Песок	yellow
1	Цемент	green
2	Щебенка	green
3	Земля	yellow
4	Вода	green
5	Гравий	yellow
6	Кирпичи	green
7	Газоблоки	green
8	Доски	green
9	Сваи	green

Рисунок 10 – Пример итогов работы программы

Таким образом, для товарных позиций «песок», «земля», «гравий» нужна корректировка полученных данных.

Код программы

```
import numpy as numpy
import pandas as pandas

## Выборка
data = {
    'Песок': numpy.random.normal(69, 7, 12),
    'Цемент': numpy.random.normal(123, 17, 12),
    'Щебенка': numpy.random.normal(77, 7, 12),
    'Земля': numpy.random.normal(50, 5, 12),
    'Вода': numpy.random.normal(100, 15, 12),
    'Травий': numpy.random.normal(95, 9, 12),
    'Кирпичи': numpy.random.normal(33, 4, 12),
    'Газоблоки': numpy.random.normal(111, 21, 12),
    'Доски': numpy.random.normal(112, 22, 12),
    'Сваи': numpy.random.normal(70, 15, 12),
}
table = pandas.DataFrame(data)
table
## Получение p0
p0 = table.sum() / table.shape[0]
p0
## Получение среднеквадратичного отклонения
sr = ((table - p0) ** 2).sum() / (table.shape[0] - 1)
sr = sr ** (1 / 2)
## Значение планируемого показателя на 13-ый месяц
table2 = p0 + numpy.random.normal(0, sr, len(p0))
columns = table2.index
pandas.concat([table, pandas.DataFrame([table2],
columns=columns)]).reset_index(drop=True)
table2
```

```

## Оценки
reliability = sr / p0
reliability
#### Уловие 1
cond1 = ((table - p0) < 2 * sr).all()
cond1
#### Условие 2
cond2 = p0 > 2 * sr
cond2
#### Условие 3
cond3 = (table > 0).all()
cond3
#### Цветовая классификация
colClas = pandas.DataFrame({'keyword':table.columns,
                             'color': 'green'})

i = 0
table.columns
for name in table.columns:
    if (not cond3[name]):
        colClas.color[i] = 'red'
    elif (not cond1[name] and not cond2[name]):
        colClas.color[i] = 'orange'
    elif (not cond1[name] or not cond2[name]):
        colClas.color[i] = 'yellow'
    i += 1

new_dict
=
pandas.Series(colClas.color.values,index=colClas.keyword).to_dict()
colClas.style.applymap(lambda v: f"background-color: {new_dict.get(v,
'None')}")

```


Вывод

В ходе выполнения данной лабораторной работы мы спрогнозировали количество продаж на тринадцатый месяц и провели анализ достоверности планирования продаж.