

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

Бахмутский М.В.

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

```
data = {  
    'Сушеный кальмар': np.random.normal(190, 4, 12),  
    'Вяленый кальмар': np.random.normal(190, 5, 12),  
    'Селедка': np.random.normal(188, 3, 12),  
    'Пиво': np.random.normal(133, 5, 12),  
    'Сухарики': np.random.normal(111, 35, 12),  
    'Чипсы': np.random.normal(111, 40, 12),  
    'Водка': np.random.normal(144, 8, 12),  
    'Сигареты': np.random.normal(166, 11, 12),  
    'Виски': np.random.normal(177, 3, 12),  
    'Ром': np.random.normal(111, 16, 12),  
}
```

Рисунок 1 – Генерация данных

	Сушеный кальмар	Вяленый кальмар	Селедка	Пиво	Сухарики	Чипсы	Водка	Сигареты	Виски	Ром
0	192.977075	189.182517	195.784630	130.619091	121.565718	84.464060	145.814833	175.190704	170.606651	106.888548
1	183.268848	186.277584	188.805616	140.128320	169.464316	80.605433	164.062096	189.849969	181.102368	130.418002
2	181.501220	192.783910	187.012461	134.595003	127.626878	146.887975	142.610997	190.802109	181.375708	127.424494
3	187.287445	185.228579	195.898556	132.846191	99.902754	101.511970	151.179238	165.317161	179.161254	106.951196
4	187.969004	194.223064	183.046447	142.738640	95.566572	132.704965	149.979312	162.379812	176.639319	114.512661
5	197.493188	197.053642	187.707292	138.405767	116.821518	116.571029	125.738495	168.983381	178.028151	126.296306
6	197.175155	194.268907	182.019944	135.092717	112.272964	36.528186	153.239454	163.805651	178.678467	72.280743
7	195.823756	195.610200	186.673747	131.214106	114.464756	132.208785	135.092561	181.031914	174.685756	110.091365
8	188.080650	192.663751	183.650398	135.049684	133.348015	127.291229	133.859765	161.733720	178.405153	115.549015
9	189.519383	182.989128	190.553499	125.617902	126.542580	125.266875	138.098048	147.127938	177.111407	120.854630
10	196.164600	188.361572	185.474794	128.202269	37.455582	119.083023	134.450188	190.874479	182.189369	141.292996
11	192.740357	197.286563	187.687342	132.332710	97.951735	39.371501	152.384721	167.924557	180.989172	85.212997

Рисунок 2 – Сформированные данные

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

Сушеный кальмар	75.576653
Вяленый кальмар	66.913839
Селедка	99.293785
Пиво	44.339525
Сухарики	85.750611
Чипсы	100.094518
Водка	47.267104
Сигареты	74.476405
Виски	14.550422
Ром	70.042663

dtype: float64

Рисунок 3 – Расчёт среднего арифметического (\hat{p}_0)

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

Сушеный кальмар	3.002659
Вяленый кальмар	5.284587
Селедка	2.508850
Пиво	5.513270
Сухарики	59.829138
Чипсы	27.798075
Водка	9.087451
Сигареты	9.735382
Виски	3.546933
Ром	15.713276

dtype: float64

Рисунок 4 – Расчет среднеквадратичного отклонения

```
x_extrapol = p0 + np.random.normal(0, std, len(p0))
pd.concat([df, pd.DataFrame([x_extrapol], columns=x_extrapol.index)]).reset_index(drop=True)
```

	Сушеный кальмар	Вяленый кальмар	Селедка	Пиво	Сухарики	Чипсы	Водка	Сигареты	Виски	Ром
0	186.300929	193.362182	187.319873	122.745054	105.098508	50.369142	137.422456	151.856827	175.330397	94.921659
1	189.110015	188.669715	182.856752	139.533085	67.112086	106.460248	146.946792	171.683786	178.901241	102.402816
2	186.475017	187.575813	184.282767	122.354394	128.186119	131.305779	146.683029	177.538850	185.066702	129.535335
3	187.670357	190.441440	187.286318	127.547523	187.086279	128.480450	148.354214	167.178498	176.313425	104.854637
4	190.124374	196.285643	182.262370	127.998534	146.581777	117.636711	140.491348	150.314123	184.980763	146.923611
5	190.130943	185.793807	182.091626	137.570762	55.918319	85.918569	134.122406	159.026864	181.116073	109.704645
6	197.264878	184.798722	187.739398	131.863185	94.298470	103.396332	155.987016	179.355583	175.719338	96.406407
7	186.846894	187.389566	187.874471	133.800104	65.714120	145.221156	157.388006	174.178099	177.215695	103.009796
8	189.370849	201.636011	188.011447	133.159201	73.686513	101.232000	144.809666	158.912740	176.122398	108.940809
9	191.036625	195.620874	184.045346	135.956551	6.174320	69.258406	155.589561	169.691700	177.319164	123.427707
10	188.907525	195.307641	183.308541	135.321953	211.640596	133.868021	143.995251	171.554189	178.069953	116.443085
11	191.854063	186.034953	181.994883	133.214239	158.692666	100.844819	127.474584	159.584410	174.634497	95.350276
12	192.617681	204.198906	181.694657	144.219919	158.139825	76.604602	144.987573	158.980912	175.371188	119.182452

Рисунок 5 – Расчёт планируемого показателя на 13-ый месяц

```
reliability = std / p0
reliability
```

Сушеный кальмар	0.015838
Вяленый кальмар	0.027657
Селедка	0.013567
Пиво	0.041845
Сухарики	0.552188
Чипсы	0.261836
Водка	0.062699
Сигареты	0.058680
Виски	0.019882
Ром	0.141569

```
dtype: float64
```

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

Уловие 1 $X_i - \hat{p}_0 < 2\hat{\sigma} : \forall i$

```
: cond1 = ((df - p0) < 2 * std).all()
cond1
```

```
: Сушеный кальмар    False
   Вяленый кальмар    True
   Селедка            True
   Пиво              True
   Сухарики           True
   Чипсы             True
   Водка             True
   Сигареты          True
   Виски             True
   Ром              False
dtype: bool
```

Рисунок 7 – Проверка первого условия

Условие 2 $\hat{p}_0 > 2\hat{\sigma}$

```
cond2 = p0 > 2 * std
cond2
```

Сушеный кальмар	True
Вяленый кальмар	True
Селедка	True
Пиво	True
Сухарики	False
Чипсы	True
Водка	True
Сигареты	True
Виски	True
Ром	True

dtype: bool

Рисунок 8 – Проверка второго условия

Условие 3 $X_i > 0 : \forall i$

```
: cond3 = (df > 0).all()
cond3
```

```
: Сушеный кальмар    True
: Вяленый кальмар    True
: Селедка             True
: Пиво                True
: Сухарики            True
: Чипсы               True
: Водка               True
: Сигареты            True
: Виски               True
: Ром                 True
dtype: bool
```

Рисунок 9 – Проверка третьего условия

```

product_color = pd.Series(dtype='string')
for name in df.columns:
    if (not cond3[name]):
        product_color[name] = 'Красный'
    elif (not cond1[name] and not cond2[name]):
        product_color[name] = 'Оранжевый'
    elif (not cond1[name] or not cond2[name]):
        product_color[name] = 'Желтый'
    else:
        product_color[name] = 'Зеленый'
product_color

```

Сушеный кальмар	Желтый
Вяленый кальмар	Зеленый
Селедка	Зеленый
Пиво	Зеленый
Сухарики	Желтый
Чипсы	Зеленый
Водка	Зеленый
Сигареты	Зеленый
Виски	Зеленый
Ром	Желтый

dtype: object

Рисунок 10 – Сведем итоги в таблицу

Таким образом, автоматизированное планирование продаж достоверно у всех, кроме позиций «сушеный кальмар, Сухарики, Ром», для них нужна корректировка полученных данных.

Вывод

В ходе выполнения данной лабораторной работы мы получили базовые навыки работы с языком python и набором функций для анализа и обработки данных. Прогнозировали количество продаж по десяти товарам на тринадцатый месяц и провели анализ достоверности планирования продаж.