Министерство науки и высшего образования Российской Федерации
САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Математическая статистика
Расчетно-графическая работа №4
Вариант 3, 1

Работу выполнил:
Рангулов Д.Э
Проверил:
___________ Береговенко. И.И.
«__»______________2024 г.

Санкт-Петербург 2024

Задание 1

1. Линейная модель

   Пусть имеется m независимых количественных признаков $x_1, ..., x_m$

   y - зависимая переменная.

   Имеется набор обучающих данных $(y_i, x_{i_1}, ..., x_{i_m})_{i=1}^n$

   $$y = f(x_1, ..., x_m)$$

   Пусть $Y = (y_1, ..., y_n)$, $X = (x_{i,j})_{i \leq n, j \leq m}$ - вектор зависимых и матрица независмых переменных.

   $$Y = Xc + \varepsilon$$

   где $\varepsilon = (\varepsilon, ...\varepsilon)^T$ - вектор ошибок(случайный), $c = (c_1, ..., c_m)$ - вектор коэффициентов.

   Помимо этого, из условия рассмотрения модели "вместе со свободным коэффициентом добавим к X единичный вектор размерности n, чтобы его учесть.

   $$X = \begin{bmatrix} 1 & x_{1,1} & ... & x_{1,m} \\ 1 & x_{2,1} & ... & x_{2,m} \\ ... & ... & ... & ... \\ 1 & x_{n,1} & ... & x_{n,m} \end{bmatrix}$$

2. Найдем оценку параметров $\hat{c}$ и $S^2(\hat{c})$:

   Оценки, полученные по методу МНК, гарантируют несмещенность, состоятельность.

   По методу наименьших квадратов, имеем:

   $A = X^T X$

   $\hat{c} = A^{-1} X^T Y$

   $S^2(\hat{c}) = (Y - Xc)^T (Y - Xc)$

$$\hat{\sigma}^2 = \frac{S^2(\hat{c})}{n-m}$$

Напишем реализацию этого метода на Python:

Листинг 1: Подготовка данных

```
file_path = 'MEN_SHOES.csv'
data = pd.read_csv(file_path)
data = data.dropna()
Y = np.array(data['RATING']).T
n = len(Y)
X1 = pd.to_numeric(data['How_Many_Sold'].replace({',': ''}, regex=True).
    values)
X2 = pd.to_numeric(data['Current_Price'].replace({' ': '', ',': ''}, regex=
    True).values)
X1 = X1.astype(int)
X2 = X2.astype(int)
X = np.column_stack((np.ones(n), X1, X2))
```

Листинг 2: Метод наименьших квадратов

```
A = X.T @ X
c_hat = np.linalg.inv(A) @ X.T @ Y
print(c_hat)
S2 = (Y - X @ c_hat).T @ (Y - X @ c_hat)
sigma2_hat = S2 / (n - m)
print(sigma2_hat)
```

По результатам выполнения программы, получаем:

$$\hat{c_0} = 3.34; \quad \hat{c_1} = 8.263e-06; \quad \hat{c_2} = 5.26e-04; \quad \hat{\sigma}^2 = 0.119;$$

3. Построение доверительных интервалов:

По следствию из тоеремы о Линейной регрессии, имеем:

Пусть $\alpha = 0.05$

$$(\hat{c}_i - c_i)\sqrt{\frac{n-m}{A_{ii}^{-1}S^2(\hat{c})}} \sim T_{(n-m)}$$

$$\frac{S^2(\hat{c})}{\sigma^2} \sim \chi^2_{n-m}$$

$$\hat{c}_i - \sqrt{\frac{A_{ii}^{-1}S^2(\hat{c})}{n-m}}t_{n-m,\frac{\alpha}{2}} \leq c_i \leq \hat{c}_i + \sqrt{\frac{A_{ii}^{-1}S^2(\hat{c})}{n-m}}t_{n-m,\frac{\alpha}{2}}$$

$$\frac{S^2(\hat{c})}{\chi^2_{n-m,1-\frac{\alpha}{2}}} \leq \sigma^2 \leq \frac{S^2(\hat{c})}{\chi^2_{n-m,\frac{\alpha}{2}}}$$

Листинг 3: Построение доверительных интервалов

```
df = n - m
alpha = 0.05
t = t.ppf(1 - alpha/2, df)

#C_i
for i in range(0, m):
    down = c_hat[i] - np.sqrt((A_inv[i, i] * S2) / (n - m)) * t
    up = c_hat[i] + np.sqrt((A_inv[i, i] * S2) / (n - m)) * t
    print(down, up)

# SIGMA
down_sigma = S2 / chi2.ppf(alpha/2, df)
up_sigma = S2 / chi2.ppf(1 - alpha/2, df)
```

Получившееся результаты:

$$3.33 \leq c_0 \leq 3.35$$

$$7.84e-06 \leq c_1 \leq 8.67e-06$$

$$0.000514 \leq c_2 \leq 0.000537$$

$$0.117 \leq \sigma^2 \leq 0.121$$

4. Расчет коэффициента детерминации:

$RSS = S^2(\hat{c}) = (Y - Xc)^T(Y - Xc) = 44713$ - квадратическая ошибка

$TSS = (Y - \overline{Y} I_n)^T \cdot (Y - \overline{Y} I_n)^T = 3788$ - сумма квадратов остатков

Листинг 4: Расчет RSS TSS R2

```
S2 = (Y - X @ c_hat).T @ (Y - X @ c_hat)
TSS = (Y - np.mean(Y) * np.ones(n).T).T @ (Y - np.mean(Y) * np.ones(n).T)
R2 = 1 - RSS / TSS
```

$$R^2 = 1 - \frac{RSS}{TSS} = 0.268$$

Чем ближе значение коэффициента R^2, тем сильнее зависимость. То есть, по сути соответствии модели данным.

Значение $R^2 = 0.268$ говорит нам о том, что модель слабо обьясняет вариацию данных, а если точнее, то она описывает 26.8% данных.

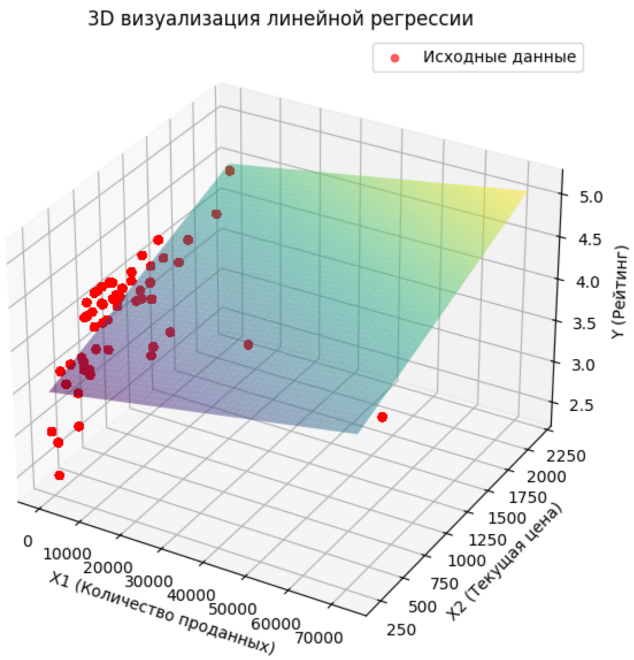

Рис. 1.1: 3D визуализация

Можно заметить сильный разброс точек относительно плоскости. Это показывает, что линейная модель плохо описывает данные. Помимо этого, это также подтверждает низкий коэффициент детерминации.

5. Проверка гипотезы ”Чем больше продажи, тем больше рейтинг”.

   Пусть $\alpha = 0.05$

   $H_0 : c_1 = 0$ - продажи не влияют на рейтинг

   $H_1 : c_1 > 0$ - чем больше продажи, тем больше рейтинг

   Для этого используем статистику:

$$(\hat{c_1} - c_1)\sqrt{\frac{n-m}{A_{ii}^{-1} S^2(\hat{c})}} \sim T_{(n-m)}$$

   Так как при $H_0 : c_1 = 0$, то получаем:

$$\hat{c_1}\sqrt{\frac{n-m}{A_{ii}^{-1} S^2(\hat{c})}} \sim T_{(n-m)}$$

   Считаем статистику:

Листинг 5: $t_{\text{набл}}$

```
t_nabl = c_hat[1] * np.sqrt((n - m) / (A_inv[1, 1] * S2))
```

$$t_{\text{набл}} = \hat{c_1}\sqrt{\frac{n-m}{A_{ii}^{-1} S^2(\hat{c})}} \sim T_{(n-m)} = 39.1;$$

Листинг 6: $t_{\text{крит}}$

```
df = n - m
alpha = 0.05
t_critical = t.ppf(1 - alpha/2, df)
```

   $t_{\text{крит}} = 1.96;$

   Так как $t_{\text{набл}} > t_{\text{крит}}$ мы отвергаем H_0 в пользу гипотезы H_1. Следовательно ”чем больше продажи, тем больше рейтинг”верно.

6. Проверка гипотезы ”Рейтинг зависит от цены”.

   Пусть $\alpha = 0.05$

$H_0 : c_2 = 0;$

$H_1 : c_2 \neq 0;$

Абсолютно аналогичным образом, как в прошлом пункте, все высчитываем:

$t_{\text{набл}} = \hat{c_1}\sqrt{\dfrac{n-m}{A_{ii}^{-1}S^2(\hat{c})}} \sim T_{(n-m)} = 88.5;$

$t_{\text{крит}} = 1.96;$

Так как $t_{\text{набл}} > t_{\text{крит}}$ мы отвергаем H_0 в пользу гипотезы H_1. Следовательно "Рейтинг зависит от цены"верно.

7. Проверьте гипотезу H_0 о равенстве одновременно нулю коэффициентов при цене и количестве проданных экземпляров против альтернативы $\frac{H_1 =}{H_0}$

   $H_0 : c_1 = 0$ и $c_2 = 0$

   H_1 : хотя бы один из коэффициентов не равен нулю.

   $\dfrac{n-m}{m} \cdot \dfrac{S^2(c) - S^2(\hat{c})}{S^2(\hat{c})} \sim F_{m,n-m}$

   Рассчитаем $S^2(c)$ при истинности H_0:

   $S^2(c) = (Y - Xc)^T(Y - Xc)$, где $c = \begin{bmatrix} 3.34 & 0 & 0 \end{bmatrix}$

Листинг 7: $S^2(c)$

```
c_hat = [3.34, 0, 0]
S2_H0 = (Y - X @ c_hat).T @ (Y - X @ c_hat)
```

$S^2(c) = 9135$

Листинг 8: $f_{\text{набл}}$

```
c_hat = [3.34, 0, 0]
S2_H0 = (Y - X @ c_hat).T @ (Y - X @ c_hat)
```

$f_{\text{набл}} = 17721.09$

$f_{\text{крит}} = 2.6$

Так как $f_{\text{набл}} > f_{\text{крит}}$ мы отвергаем H_0 в пользу гипотезы H_1.

Задание 2

$$y_{i,j} = \mu_j + \varepsilon_{i,j},$$

где $1 \leq i \leq I_j$, $I = I_1 + ...I_j$ - номер наблюдения в j-ом факторе, $1 \leq j \leq J$ - уровень фактора, $\varepsilon_{i,j}$ - центрированные независимые гауссовские величины с одинаковой дисперсией.

$H_0 : \mu_1 = ... = \mu_J$

H_1 : не все μ равны

Число факторов $J = 3$;

Общее число наблюдений $n = 150$;

Найдем среднее по группам:

Листинг 9: Подсчет среднего и дисперсии

```
filtered_data = data[data['Species'] == 'setosa']
y1 = np.mean(filtered_data['Summary_width'])
filtered_data = data[data['Species'] == 'versicolor']
y2 = np.mean(filtered_data['Summary_width'])
filtered_data = data[data['Species'] == 'virginica']
y3 = np.mean(filtered_data['Summary_width'])
print(y1, y2, y3)
```

Таблица 1. Результат подсчета среднего по группам:

j	Factor	$\overline{y_{*,j}}$
1	setosa	17.62
2	versicolor	22.24
3	virginica	30.98

MS_B - оценка межгрупповой дисперсии. $MS_B = \dfrac{S_B}{J-1}$, $S_B = \sum\limits_{j=1}^{J} n_j(\overline{y_{*,j}} - \overline{y})^2$

Листинг 10: Подсчет MS_B

```
y_sr = np.array([y1, y2, y3])
Sb = 0
for j in range (0, 3):
    filtered_data = data[data['Species'] == unique_species[j]]
    Sb += len(filtered_data) * (y_sr[j] - np.mean(y_sr)) ** 2
MSb = Sb / (len(unique_species) - 1)
```

$MS_B = 2300.67$

MS_W - оценка внутригрупповой дисперсии, $MS_W = \dfrac{S_W}{I-J}$,

$S_W = \sum\limits_j \sum\limits_i (y_{i,j} - \overline{y_{*,j}})$

Листинг 11: Подсчет MS_W

```
Sw = 0
for j in range(0, 3):
    filtered_data = data[data['Species'] == unique_species[j]]
    filtered_data = filtered_data['Summary_width'].values
    for i in range(50):
        Sw += (filtered_data[i] - y_sr[j])**2
MSw = Sw / (len(data) - len(unique_species))
print(MSw)
```

$MS_W = 17.259$

Далее, имеем

$$F = \frac{MS_B}{MS_W} \sim F(J-1, I-J)$$

$F_{\text{набл}} = 133.3$

Находим $F_{\text{крит}}$ для $\alpha = 0.05$

Листинг 12: Подсчет $F_{\text{крит}}$

```
alpha = 0.05
dfn = len(unique_species)
dfd = len(data) - len(unique_species)
F_crit = f.ppf(1 - alpha, dfn, dfd)
```

$F_{\text{крит}} = 2.66$

Так как критическая область правосторонняя и $F_{\text{набл}} > F_{\text{крит}}$, то нулевая гипотеза отклоняется в пользу H_1, так как различия между группами(факторами) являются статистически значимыми.

Вывод по работе