Министерство науки и высшего образования Российской Федерации
Российский химико-технологический университет имени Д. И. Менделеева

Факультет цифровых технологий и химического инжиниринга
Кафедра информационных компьютерных технологий

ОТЧЕТ

ПО РАСЧЕТНО-ГРАФИЧЕСКОЙ РАБОТЕ

«Получение математической модели для расчёта энергетической ценности продукта в зависимости от содержания в нём макронутриентов»

Вариант №43

ВЫПОЛНИЛА: Студентка группы Кс-24 Мосолова В.Г.

ПРОВЕРИЛ: к.т.н., доцент Дударов С. П.

30 ноября 2022 г.
Москва

СОДЕРЖАНИЕ

1. Расчетно-графическая работа. Получение математической модели для расчёта энергетической ценности продукта в зависимости от содержания в нём макронутриентов

1.1. Задание: номер варианта, содержание задания

№ вар.	Математическая модель	Метод	Объём выборки
43	$Q_{расч} = a_0 + a_1 x_Б + a_2 x_Ж + a_3 x_У + a_4 x_Б^2 + a_5 x_Ж^2$	2	7

Используемые обозначения

a_j – коэффициент математической модели;

i – номер продукта в выборке;

j – номер коэффициента в модели;

N – количество продуктов в выборке;

$Q_{расч}$ – энергетическая ценность продукта, рассчитанная по математической модели;

$Q_{эксп}$ – энергетическая ценность продукта, взятого в экспериментальной выборке;

$x_{Бi}$ – содержание белков в продукте i;

$x_{Жi}$ – содержание жиров в продукте i;

$x_{Уi}$ – содержание углеводов в продукте i;

Метод нахождения коэффициентов математической модели:

2 – аппроксимация экспериментальных данных, МНК.

Задача. Найти аппроксимирующее соотношение заданной структуры на основе исходной выборки по продуктам.

1.2. Теоретическая часть

Аппроксимация – процесс нахождения функциональной зависимости, приближённо, но наилучшим образом во всех имеющихся экспериментальных точках описывающей явление или процесс в области её определения.

Метод наименьших квадратов(МНК) заключается в использовании для решения задачи аппроксимации квадратичного критерия рассогласования.

$$R_2 = \sum_{i=1}^{n} \left(y*_i - y_i \right)^2$$

Рисунок 1.2.1. Сумма квадратов рассогласований.

Матричная форма представления МНК:

1) Введём характеристическую матрицу X для многомерной полиномиальной зависимости:

$$y* = a_0 + a_{11}x_1 + a_{12}x_2 + a_{21}x_1^2 + a_{22}x_2^2 + a_2 x_1 x_2 + \dots$$

Рисунок 1.2.2. Многомерная полиномиальная зависимость, линейная относительно ее параметров.

$$\overline{\overline{X}} = \begin{bmatrix} 1 & x_{11} & x_{21} & x_{11}^2 & x_{21}^2 & x_{11}x_{21} & \dots \\ 1 & x_{12} & x_{22} & x_{12}^2 & x_{22}^2 & x_{12}x_{22} & \dots \\ 1 & x_{13} & x_{23} & x_{13}^2 & x_{23}^2 & x_{13}x_{23} & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & x_{1n} & x_{2n} & x_{1n}^2 & x_{2n}^2 & x_{1n}x_{2n} & \dots \end{bmatrix}$$

Рисунок 1.2.3. Характеристическая матрица.

2) Тогда мы можем получить выражение для нахождения вектора неизвестных коэффициентов аппроксимирующей зависимости в векторно-матричной форме с использованием характеристической матрицы X:

$$\vec{a} = \left(\overline{\overline{X}}^T \cdot \overline{\overline{X}} \right)^{-1} \cdot \overline{\overline{X}}^T \cdot \vec{y}$$

Рисунок 1.2.4. Вектор неизвестных коэффициентов аппроксимирующей зависимости.

3) Вывод формулы:

$$\overline{\overline{F}} \vec{a} = \vec{b}$$

$$\overline{\overline{F}} = \overline{\overline{X}}^T \cdot \overline{\overline{X}}$$

$$\vec{b} = \overline{\overline{X}}^T \cdot \vec{y}$$

$$\overline{\overline{X}}^T \cdot \overline{\overline{X}} \cdot \vec{a} = \overline{\overline{X}}^T \cdot \vec{y}$$

$$\left(\overline{\overline{X}}^T \cdot \overline{\overline{X}} \right)^{-1} \cdot \overline{\overline{X}}^T \cdot \overline{\overline{X}} \cdot \vec{a} = \left(\overline{\overline{X}}^T \cdot \overline{\overline{X}} \right)^{-1} \cdot \overline{\overline{X}}^T \cdot \vec{y}$$

$$\overline{\overline{E}} \cdot \vec{a} = \left(\overline{\overline{X}}^T \cdot \overline{\overline{X}} \right)^{-1} \cdot \overline{\overline{X}}^T \cdot \vec{y}$$

$$\vec{a} = \left(\overline{\overline{X}}^T \cdot \overline{\overline{X}} \right)^{-1} \cdot \overline{\overline{X}}^T \cdot \vec{y}$$

Рисунок 1.2.5. Вывод формулы вектора неизвестных коэффициентов аппроксимирующей зависимости.

Где матрица F – умножение транспонированной матрицы X на ее исходную форму, а b – вектор столбец свободных членов, полученный умножением транспонированной матрицы X на вектор столбец y.

$$E = \frac{1}{N} \sqrt{\sum_{i=1}^{N} \left(Q_{\text{эксп}} - Q_{\text{расч}} \right)^2}$$

Рисунок 1.2.6. Среднеквадратичная ошибка найденной математической модели на выборке

1.3. Практическая часть

Выбрала продукты своего постоянного рациона питания в количестве, равном объёму выборки – 7.

Таблица 1.3.1

Таблица содержания макронутриентов и энергетической ценности продуктов из экспериментальной выборки.

№ п/п.	Наименование продукта	Содержание на 100 г			Энергетическая ценность, ккал
		Белков	Жиров	Углеводов	
1	Хлебцы	8.7	0.9	82.4	372.5
2	Крупа гречневая	13	2.5	68	346.5
3	Крем творожный	8.3	15	11.9	215.8
4	Льняная каша	36.6	11	10	348.4
5	Смесь хлопьев	11	3.5	66	339.5
6	Кускус	12	1	76	360
7	Хлеб тостовый	6.8	2.6	48.3	243.8

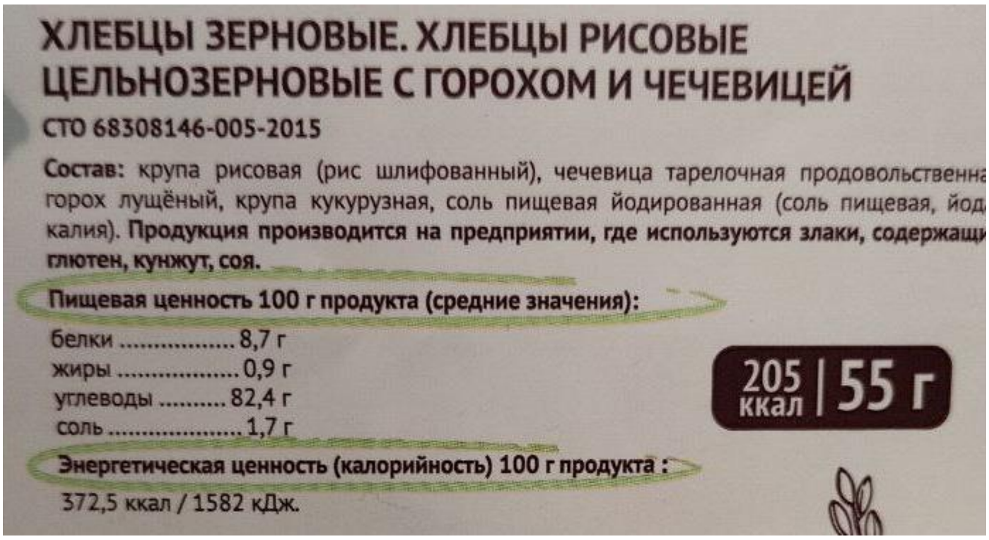

Рисунок 1.3.2. Хлебцы.

Рисунок 1.3.3. Крупа гречневая.

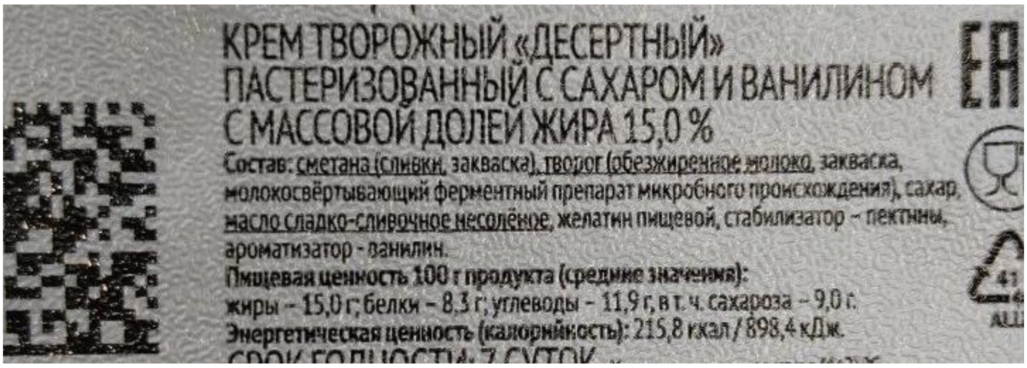

Рисунок 1.3.4. Крем творожный.

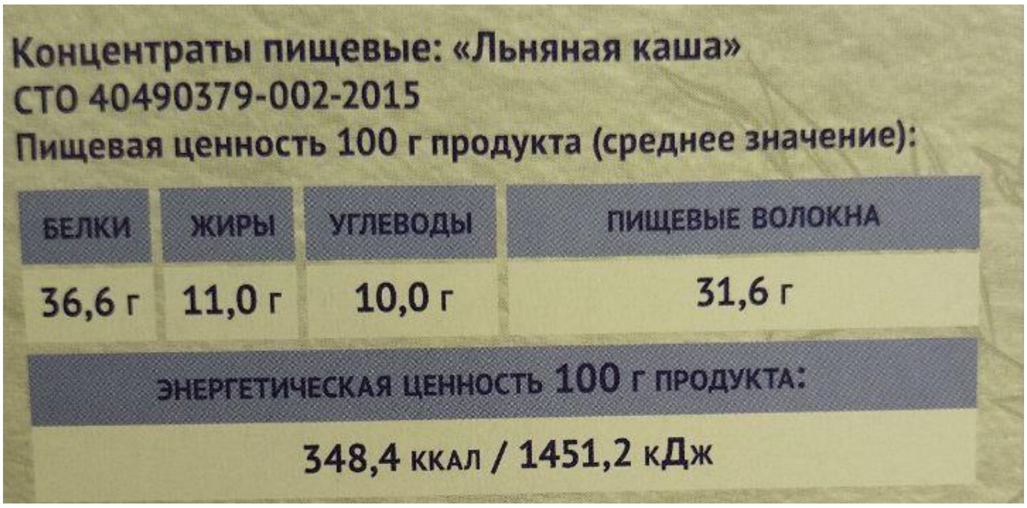

Рисунок 1.3.5. Льняная каша.

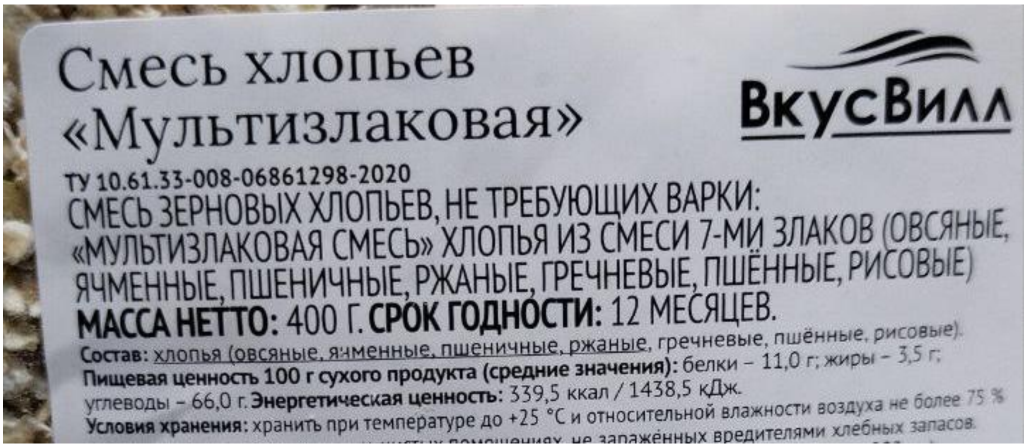

Рисунок 1.3.6. Смесь хлопьев.

Рисунок 1.3.7. Кускус.

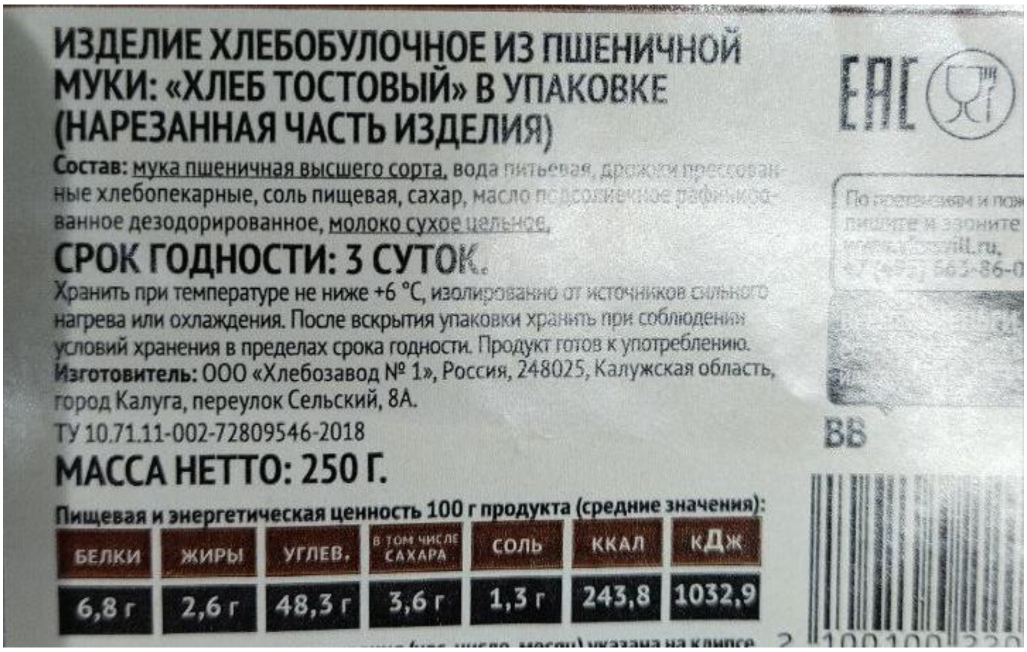

Рисунок 1.3.8. Хлеб тостовый.

Выбрала 3 продукта для тестирования модели.

Таблица 1.3.9

Таблица содержания макронутриентов и энергетической ценности продуктов из тестовой выборки.

№ п/п.	Наименование продукта	Содержание на 100 г			Энергетическая ценность, ккал
		Белков	Жиров	Углеводов	
1	Кекс шоколадный	5.8	29.1	44.2	461.9
2	Крупа рисовая	7	1	74	333
3	Печенье затяжное	13.4	18	62.9	467.2

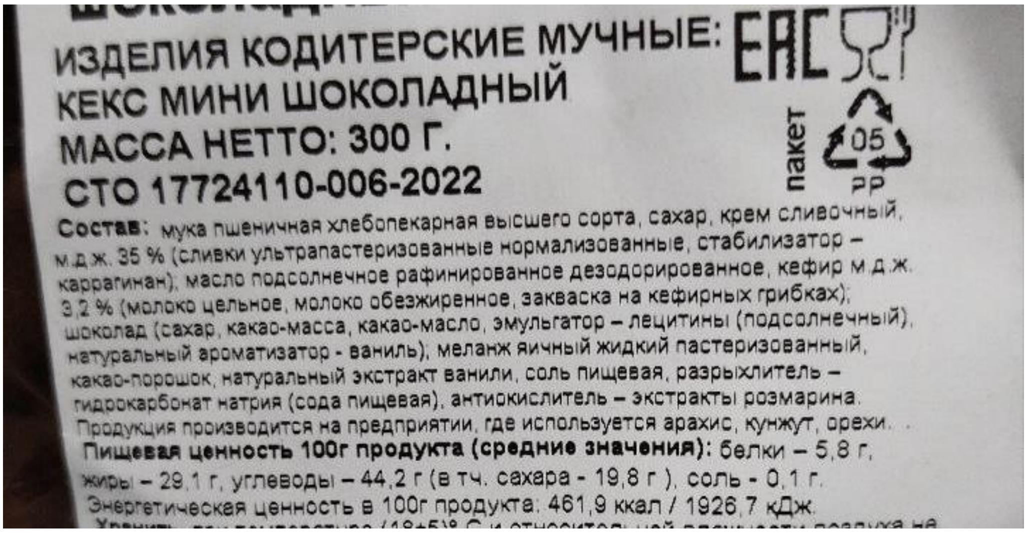

Рисунок 1.3.10. Кекс шоколадный.

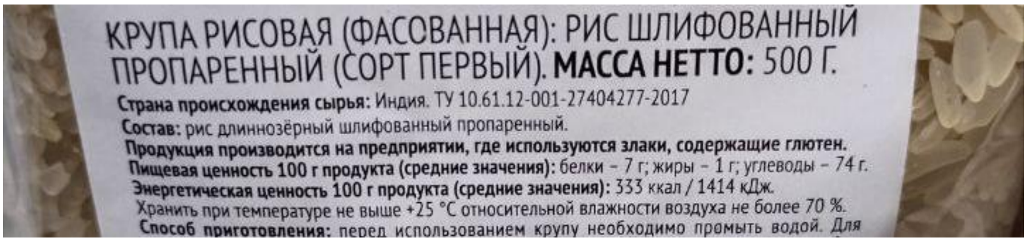

Рисунок 1.3.11. Крупа рисовая.

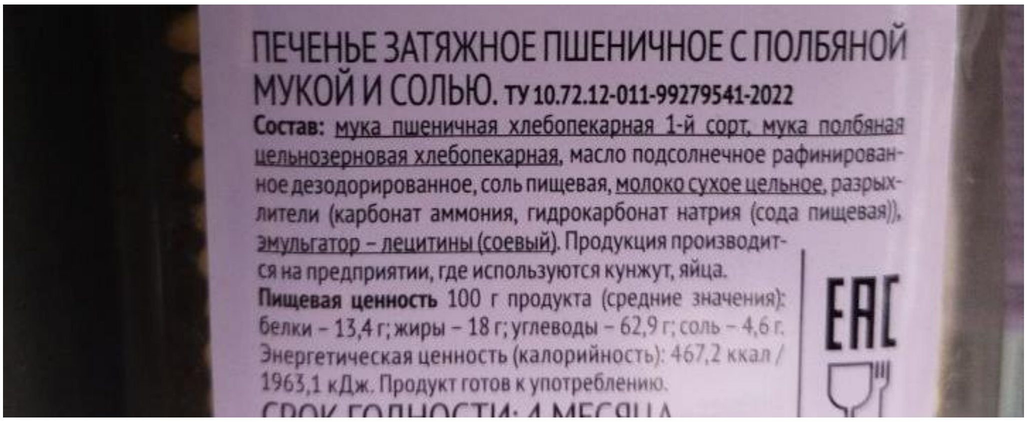

Рисунок 1.3.12. Печенье затяжное.

С использованием метода МНК определила коэффициенты математической модели, указанного в варианте задания.

```
#define N 7
#define M 6

void input_data(double x[N][M], double Q[N]);
void transp(double x[N][M], double xT[M][N]);
void multi(double x[N][M], double xT[M][N], double F[M][M]);
void invert(double F[M][M], double E[M][M]);
void coefficient_a(double E[M][M], double xT[M][N], double a[M], double Q[N]);
void approxi(double a[M]);

int main() {
    double x[N][M], xT[M][N], Q[N];
    double a[M];
    double E[M][M];
    double F[M][M] = {0};
    input_data(x, Q);
    transp(x, xT);
    multi(x, xT, F);
    invert(F, E);
    coefficient_a(E, xT, a, Q);
    approxi(a);
    return 0;
}
```

Рисунок 1.3.13. Вызов функций для нахождения коэффициентов мат. модели.

```
void input_data(double x[N][M], double Q[N]) {
    double B, Zh, U;
    FILE*f=fopen("data.txt", "r");
    if(f==0) printf("error");
    for(int i = 0; i < N; i++) {
        fscanf(f, "%lf%lf%lf%lf", &B, &Zh, &U, &Q[i]);
        x[i][0]= 1;
        x[i][1]= B;
        x[i][2]= Zh;
        x[i][3]= U;
        x[i][4]= B*B;
        x[i][5]= Zh*Zh;
    }
    printf("x[][]:\n");
    for(int i=0; i<N; i++) {
        for(int j=0; j<M; j++) {
            printf("%-10.3lf", x[i][j]);
        }
        printf("\n");
    }
    fclose(f);
}
```

Рисунок 1.3.14. Функция для считывания данных БЖУ и Ккал исходной (экспериментальной) выборки. Составление характеристической матрицы и ее вывод.

```
void transp(double x[N][M], double xT[M][N]) {
    for(int i=0; i<N; i++) {
        for(int j=0; j<M; j++) {
            xT[j][i]=x[i][j];
        }
    }
    printf("xT[][]:\n");
    for(int i=0; i<M; i++) {
        for(int j=0; j<N; j++) {
            printf("%-10.3lf", xT[i][j]);
        }
        printf("\n");
    }
}
```

Рисунок 1.3.15. Функция составления транспонированной характеристической матрицы и ее вывод.

```
void multi(double x[N][M], double xT[M][N], double F[M][M]){
    for(int i = 0; i < M; i++) {
        for(int j = 0; j < N; j++) {
            F[i][j] = 0;
            for(int k = 0; k < N;k++){
                F[i][j]+=xT[i][k]*x[k][j];
            }
        }
    }
    printf("F[][]:\n");
    for(int i=0; i<M; i++) {
        for(int j=0; j<M; j++) {
            printf("%-13.3lf", F[i][j]);
        }
        printf("\n");
    }
}
```

Рисунок 1.3.16. Функция составления матрицы F и ее вывод.

```
void coefficient_a(double E[M][M], double xT[M][N], double a[M], double Q[N]) {
    double c[M][N];
    for(int i = 0; i < M; i++) {
        for(int j = 0; j < N; j++) {
            c[i][j] = 0;
            for(int k = 0; k < M;k++){
                c[i][j]+=E[i][k]*xT[k][j];
            }
        }
    }
    for (int i = 0; i < M; i++) {
        a[i] = 0;
        for (int j = 0; j < N; j++) {
            a[i]+= c[i][j]*Q[j];
        }
    }
    for(int i=0; i<M; i++) {
        printf("a[%d]=%-13.3lf\n", i, a[i]);
    }
    printf("\n");
}
```

Рисунок 1.3.17. Функция нахождения вектора неизвестных коэффициентов аппроксимирующей зависимости и его вывод.

```
void invert(double F[M][M], double E[M][M]) {
    for (int i = 0; i < M; i++) {
        for (int j = 0; j < M; j++){
            if (i == j) E[i][j] = 1;
            else E[i][j] = 0;
        }
    }
    double temp;
    for (int k = 0; k < M; k++) {
        temp = F[k][k];
        for (int j = 0; j < M; j++) {
            F[k][j] /= temp;
            E[k][j] /= temp;
        }
        for (int i = k + 1; i < M; i++) {
            temp = F[i][k];
            for (int j = 0; j < M; j++) {
                F[i][j] -= F[k][j] * temp;
                E[i][j] -= E[k][j] * temp;
            }
        }
    }
    for (int k = M - 1; k > 0; k--) {
        for (int i = k - 1; i >= 0; i--) {
            temp = F[i][k];
            for (int j = 0; j < M; j++)
            {
                F[i][j] -= F[k][j] * temp;
                E[i][j] -= E[k][j] * temp;
            }
        }
    }
```

Рисунок 1.3.18. Нахождение обратной матрицы F.

```
void approxi(double a[M]) {
    double B, Zh, U, Q_real[N];
    double R[N] = {0};
    FILE*f=fopen("data.txt", "r");
    if(f==0) printf("error");
    int k = 0;
    printf("\n7 products:\n");
    while(!feof(f)) {
        fscanf(f, "%lf%lf%lf%lf", &B, &Zh, &U, &Q_real[k]);
        double Q = a[0]+a[1]*B+a[2]*Zh+a[3]*U+a[4]*B*B+a[5]*Zh*Zh;
        printf("Q[%d]=%-10.4lf\n", k, Q);
        R[k] = (Q - Q_real[k]) * (Q - Q_real[k]);
        k++;
    }
    double sum = 0;
    for(int i = 0; i < N; i++) {
        sum+=R[i];
        printf("\nR[%d]=%-4.7lf", i, R[i]);
    }
    printf("\n\nE=%-4.10lf\n", sqrt(sum)/N);
    fclose(f);
```

Рисунок 1.3.19. Функция нахождения (для исходной выборки – 7 продуктов) Q расчетных, квадратов рассогласований, среднеквадратичной ошибки и их вывод.

```
    printf("\n3 products:\n");
    double R_i[3], Q_i[3]={0};
    FILE*w=fopen("input.txt", "r");
    if(w==0) printf("error");
    int l = 0;
    while(!feof(f)) {
        fscanf(w, "%lf%lf%lf%lf", &B, &Zh, &U, &Q_i[l]);
        double Q = a[0]+a[1]*B+a[2]*Zh+a[3]*U+a[4]*B*B+a[5]*Zh*Zh;
        printf("Q[%d]=%-10.4lf\n", l, Q);
        R_i[l] = (Q - Q_i[l]) * (Q - Q_i[l]);
        l++;
    }
    sum = 0;
    for(int i = 0; i < 3; i++) {
        sum+=R_i[i];
        printf("\nR[%d]=%-4.7lf", i, R_i[i]);
    }
    printf("\n\nE=%-4.10lf\n\n", sqrt(sum)/3);
    fclose(w);
```

Рисунок 1.3.20. Функция нахождения (для тестовой выборки – 3 продукта) Q расчетных, квадратов рассогласований, среднеквадратичной ошибки и их вывод.

Вывод результатов программы:

```
x[][]:
1.000   8.700   0.900   82.400  75.690   0.810
1.000   13.000  2.500   68.000  169.000  6.250
1.000   8.300   15.000  11.900  68.890   225.000
1.000   36.600  11.000  10.000  1339.560 121.000
1.000   11.000  3.500   66.000  121.000  12.250
1.000   12.000  1.000   76.000  144.000  1.000
1.000   6.800   2.600   48.300  46.240   6.760
```

Рисунок 1.3.21. Полученная характеристическая матрица X.

```
xT[][]:
1.000   1.000   1.000   1.000    1.000   1.000   1.000
8.700   13.000  8.300   36.600   11.000  12.000  6.800
0.900   2.500   15.000  11.000   3.500   1.000   2.600
82.400  68.000  11.900  10.000   66.000  76.000  48.300
75.690  169.000 68.890  1339.560 121.000 144.000 46.240
0.810   6.250   225.000 121.000  12.250  1.000   6.760
```

Рисунок 1.3.22. Полученная транспонированная характеристическая матрица X.

```
F[][]:
7.000     96.400    36.500    362.600   1964.380    373.070
96.400    1964.380  635.610   4032.090  55828.618   6577.115
36.500    635.610   373.070   965.240   16946.855   4783.805
362.600   4032.090  965.240   24120.260 53107.639   5590.252
1964.380  55828.618 16946.855 53107.639 1870971.939 180643.401
373.070   6577.115  4783.805  5590.252  180643.401  65502.479
```

Рисунок 1.3.23. Полученная матрица F.

```
invert F:
22.2391955   0.2693499   -3.1438876  -0.2659966  -0.0066502  0.1169378
0.2693499    0.2411842   -0.2195412  -0.0252136  -0.0055127  0.0076370
-3.1438876   -0.2195412  0.8144063   0.0497275   0.0045776   -0.0363961
-0.2659966   -0.0252136  0.0497275   0.0053519   0.0005906   -0.0016705
-0.0066502   -0.0055127  0.0045776   0.0005906   0.0001279   -0.0001460
0.1169378    0.0076370   -0.0363961  -0.0016705  -0.0001460  0.0017857
```

Рисунок 1.3.24. Полученная обратная матрица F.

```
a[0]=4.769
a[1]=1.812
a[2]=9.861
a[3]=4.073
a[4]=0.099
a[5]=-0.032
```

Рисунок 1.3.25. Полученные коэффициенты математической модели.

Получила формулу:

$Q_{расч} = 4.769 + 1.812x_{Б} + 9.861x_{Ж} + 4.073x_{У} + 0.099x_{Б}^2 - 0.032x_{Ж}^2$

Рассчитала значения Q расчетных, квадратов рассогласований для каждого продукта исходной выборки с 7 продуктами и тестовой выборки с 3 продуктами, а также значения среднеквадратичной ошибки для каждой выборки.

Q[i] – Q расчетное,

R[i] – квадрат рассогласования,

E – среднеквадратичная ошибка.

```
7 products:
Q[0]=372.4598
Q[1]=346.3954
Q[2]=215.7986
Q[3]=348.4009
Q[4]=339.5664
Q[5]=360.0809
Q[6]=243.7979

R[0]=0.0016159
R[1]=0.0109370
R[2]=0.0000018
R[3]=0.0000008
R[4]=0.0044110
R[5]=0.0065457
R[6]=0.0000044

E=0.0219073779
```

Рисунок 1.3.26. Результаты для исходной выборки.

```
3 products:
Q[0]=458.5471
Q[1]=333.5164
Q[2]=470.0927

R[0]=11.2422607
R[1]=0.2666626
R[2]=8.3678290

E=1.4861117159
```

Рисунок 1.3.27. Результаты для тестовой выборки.

1.4. Выводы по работе

С помощью метода МНК по экспериментальной выборке я получила значения неизвестных коэффициентов математической модели и составила аппроксимирующее соотношение. Для исходной выборки значение среднеквадратичной ошибки получилось небольшое E=0.0219, для тестовой выборки E=1.486.

Небольшое значение ошибки для исходной выборки могло получится благодаря тому, что мат. модель достаточно точно аппроксимировала исходную выборку. Ошибка в тестовой выборке могла получиться чуть больше, потому что их данные(значения БЖУ и Ккал) тестовой выборки менее похожи на данные исходной.

Минимизировать ошибку можно, подбирая мат. модель(меняя степень полинома), которая наиболее точно будет описывать выборку.

Полученную мат. модель можно использовать если необходимо приблизительно рассчитать значение Ккал продуктов. Мат. модель не получится использовать при больших значениях(около 100 и более) одного из макронутриентов и при этом очень малых других, так как возникает большая ошибка.