Клепов Дмитрий (М3238)
Обратная связь: dimkakirov43@mail.ru

Вариант 9

Задание.

Для заданной квадратичной функции $y = a_3 + a_2x + a_1x^2$

- Смоделировать квадратичную функцию, наблюдаемую в нормальных шумах
- Оценить коэффициенты квадратичной зависимости, уровень шумов и квадратичную функцию по зашумленным данным
- Сравнить полученные результаты с исходными данными.
- Проделать аналогичное с прямой $y = c_2 + c_1x$

Исходные данные

- $x_{min} = -0.5$
- $x_{max} = 1.5$
- $n = 50$
- $y = 3.5 - 2.2x + 1.2x^2$, $y = 2.8 + 1.3x$
- $s = 2.3$

Графики.

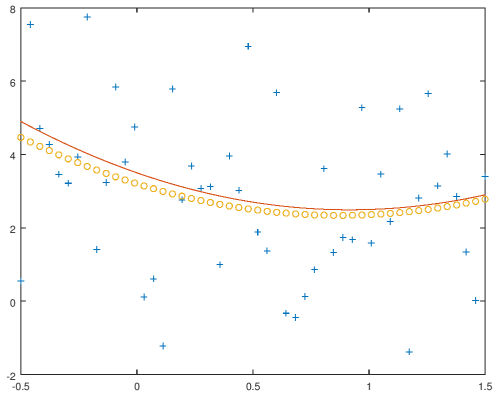

Рис. 1: Квадратичная

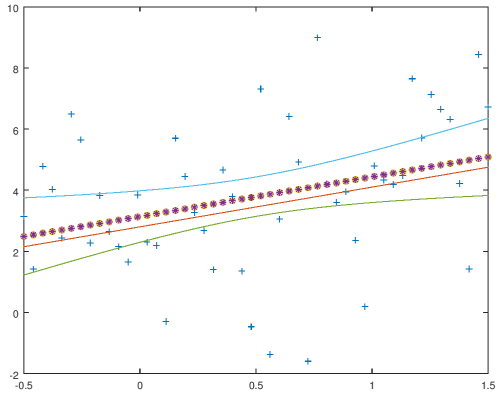

Рис. 2: Линейная

- Квадратичная:

```
pkg load statistics;

xmin = -0.5;
xmax = 1.5;
n = 50;
a = [1.2, -2.2, 3.5];
s = 2.3;

X = xmin : (xmax - xmin) / (n - 1) : xmax;
origin_y = polyval(a, X);
noised_y = origin_y + s * randn(1, n);
polynom = polyfit(X, noised_y, 2);
computed_y = polyval(polynom, X);

plot(X, noised_y, '+', X, origin_y, X, computed_y, 'o');

diff = computed_y - noised_y;
```

```
printf("Noise: %f\n", sqrt(diff / (n - 3) * diff'));
printf("Ort check: %f\n", computed_y * diff');
```

Выход:

```
    Noise: 2.473507
    Ort check: -0.000000
```

- Линейная:

```
pkg load statistics;

xmin = -0.5;
xmax = 1.5;
n = 50;
c = [1.3, 2.8];
s = 2.3;

X = xmin : (xmax - xmin) / (n - 1) : xmax;
origin_y = polyval(c, X);
noised_y = origin_y + s * randn(1, n);
polinom = polyfit(X, noised_y, 1);
printf("Diff: %f\n", polinom(1) - b);

Yp2 = polyval(polinom, X);
avg_x = mean(X);
avg_noised_y = mean(noised_y);
cov = (X - avg_x) * (noised_y - avg_noised_y)' / (n - 1);
b = cov / (std(X)^2);
Yp1 = avg_noised_y + b * (X - avg_x);

diff = Yp1 - noised_y;
printf("Ort check: %f\n", Yp1 * diff');
sn = sqrt(diff / (n - 2) * diff');
printf("Noise: %f\n", sn);

t = 1.96;
h = t * (sn / sqrt(n));
d = h * (1 + (X - avg_x).^2 / (std(X)^2)).^(1 / 2);
left = Yp1 - d;
right = Yp1 + d;

plot(X, noised_y, '+', X, origin_y, X, Yp1, 'o', X, Yp2, '*', X, left, X, right);
```

Выход:

```
    Diff: 1.369099
    Ort check: -0.000000
    Noise: 2.617631
```

Вывод: (для обеих)

- Полученная функция почти совпадает с исходной
- Полученный уровень шумов близок к заданному
- Вектор несвязок ортогонален вектору зашумленной функции

Для линейной: функция попала в доверительный интервал.