Оценка погрешностей параметров моделей при оптимизации

Метод наименьших квадратов
Одномерная линейная регрессия

$$\sigma=\sum_i \left(f\left(\bar{\beta},x_i\right)-y_i\right)^2$$

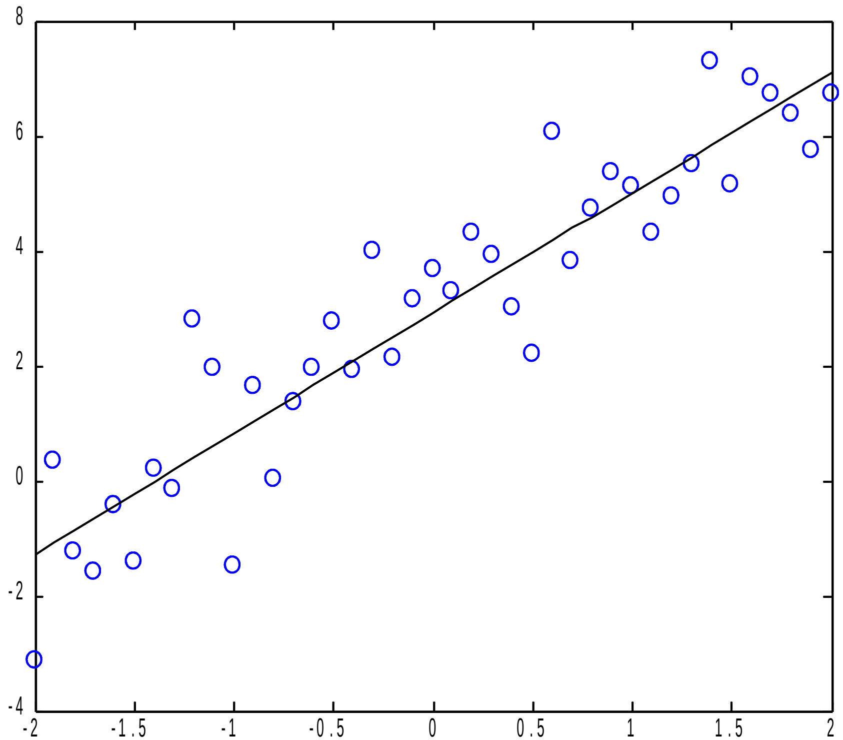
$$\frac{\partial\sigma}{\partial\beta_j}=2\sum_i\left(\left(f\left(\bar{\beta},x_i\right)-y_i\right)\frac{\partial f}{\partial\beta_j}\left(\bar{\beta},x_i\right)\right)=0$$

Уравнение прямой: $y=\beta_1 x+\beta_2$

$$\begin{cases}\dfrac{\partial\sigma}{\partial\beta_1}=\sum\limits_i x_i\left(\beta_1 x_i+\beta_2-y_i\right)=0\\ \dfrac{\partial\sigma}{\partial\beta_2}=\sum\limits_i\left(\beta_1 x_i+\beta_2-y_i\right)=0\end{cases}$$

$$\begin{cases}\beta_1\overline{x^2}+\beta_2\bar{x}-\overline{xy}=0\\ \beta_1\bar{x}+\beta_2-\bar{y}=0\end{cases}$$

$$\begin{cases}\beta_1=\dfrac{\overline{xy}-\bar{x}\,\bar{y}}{\overline{x^2}-\bar{x}^2}\\ \beta_2=\bar{y}-\beta_1\bar{x}\end{cases}$$



Метод наименьших квадратов
Многомерная линейная регрессия

$$y=\beta_1 x^{(1)}+\beta_2 x^{(2)}+\ldots+\beta_m x^{(m)}$$

Система уравнений в явном виде

$$\begin{cases} \beta_1 x_1^{(1)}+\beta_2 x_1^{(2)}+\ldots+\beta_m x_1^{(m)}+\epsilon_1=y_1 \\ \beta_1 x_2^{(1)}+\beta_2 x_2^{(2)}+\ldots+\beta_m x_2^{(m)}+\epsilon_2=y_2 \\ \ldots \\ \beta_1 x_n^{(1)}+\beta_2 x_n^{(2)}+\ldots+\beta_m x_n^{(m)}+\epsilon_n=y_k \end{cases}$$

m — число параметров модели
n — число точек

Система уравнений в матричном виде

$$X\beta+\epsilon=y$$

Матрица регрессоров

$$X=\begin{pmatrix} x_1^{(1)} & x_1^{(2)} & \ldots & x_1^{(k)} \\ x_2^{(1)} & x_2^{(2)} & \ldots & x_2^{(k)} \\ \ddots & \vdots & \vdots & \vdots \\ x_n^{(1)} & x_n^{(2)} & \ldots & x_n^{(k)} \end{pmatrix}$$

Вектор отклонений

$$\epsilon=\begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix}$$

Коэффициенты регрессии

$$\beta=\begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{pmatrix}$$

Метод наименьших квадратов

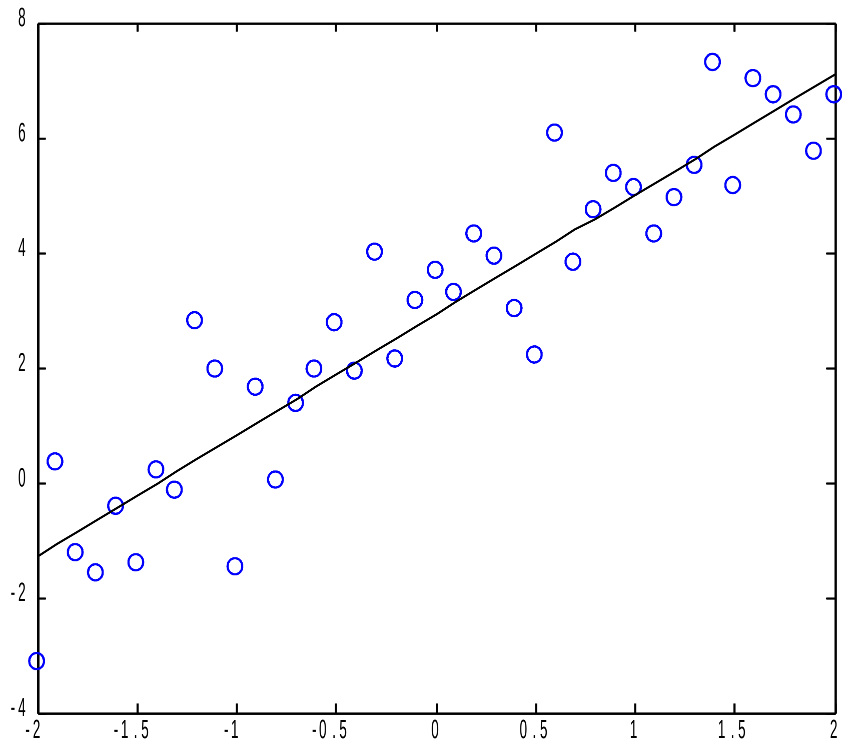
Рассматриваемые методики:
1. Операторы языка MATLAB (только линейная регрессия)
2. Statistics Toolbox (nlinfit + nlparci)
3. Optimization Toolbox + Statistics Toolbox (lsqnonlin + nlparci)

Рассматриваемые модели:
1. Одномерная линейная регрессия
2. Многомерная линейная регрессия
3. Нелинейная регрессия, поддающаяся линеаризации
4. Нелинейная регрессия без линеаризации
5. Термодинамическая модель системы ацетон-гексан

Одномерная линейная регрессия

0. Исходные данные

```
x = (-2:0.1:2)';
y = 2*x + 3 + randn(size(x));
```



1. Построить матрицу X

$$X = \begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{pmatrix}$$

```
X = [x ones(size(x))];
```

2. Найти коэффициенты регрессии β

$$\beta = \left(X^T X\right)^{-1} X^T y$$

```
beta = X\y;
```

3. Найти доверительные интервалы

$$C = \left(X^T X\right)^{-1}$$

$$\Delta\beta_j = s_{\beta,j}\left(X^T X\right)^{-1} t_{\alpha,n}$$

$$s^2_{\beta,j} = C_{jj} \frac{\sum_i^n \left(y_i - F(x_i)\right)^2}{f}$$

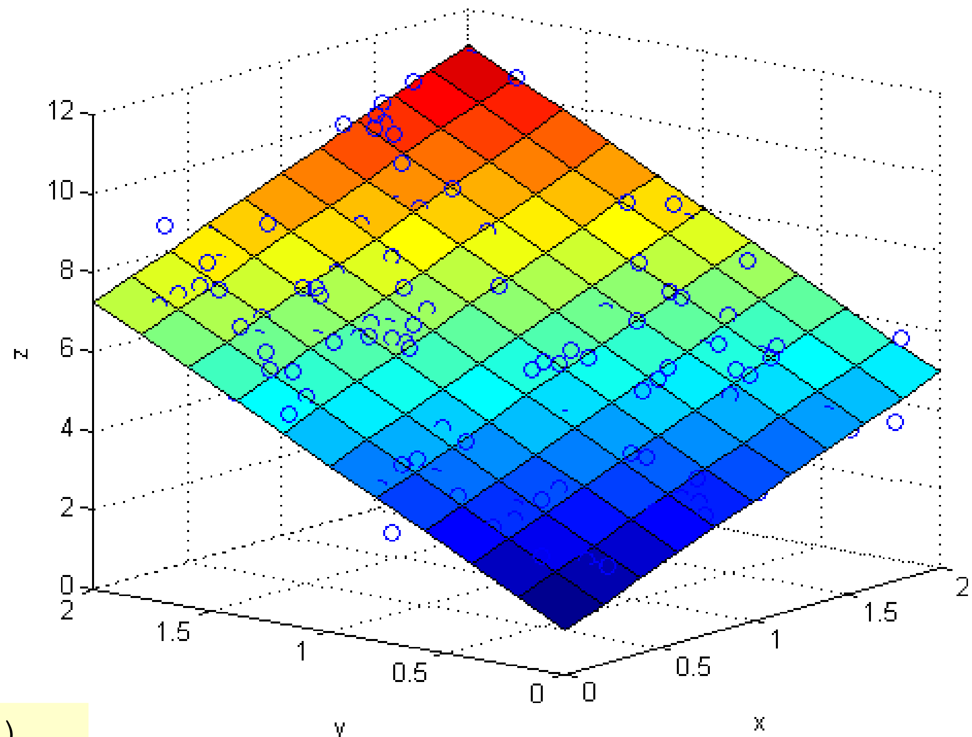
```
resid=y-(beta(1)*x+beta(2));
covar=inv(X'*X);
sbeta=sqrt(diag(covar)/(numel(x)-2))*norm(resid);
dbeta=sbeta*tinv(1-0.05/2,numel(x)-2);
```

Многомерная линейная регрессия

$$z = ax + by + c$$

0. Исходные данные

```
x = rand(150,1)*2;
y = rand(150,1)*2;
z = 2*x + 3*y + 1 + randn(size(x));
```



1. Построить матрицу X

$$X = \begin{pmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \ddots & \vdots \\ x_n & y_n & 1 \end{pmatrix}$$

```
X = [x y ones(size(x))];
```

2. Найти коэффициенты регрессии β

$$\beta = (X^T X)^{-1} X^T y$$

```
b = X\z;
```

3. Найти доверительные интервалы

```
ci=nlparci(b,b(1)*x+b(2)*y+b(3)-z,'J',X)
```

4. Вывести результаты регрессии

```
plot3(x,y,z,'bo');
hold on;
[xm,ym]=meshgrid(0:0.2:2,0:0.2:2);
zm=b(1)*xm+b(2)*ym+b(3);
surf(xm, ym, zm);
hold off;
xlabel('x');ylabel('y');zlabel('z');
```

beta =

1.9143
3.0841
1.1087

ci =

1.7780	2.0506
2.9582	3.2099
0.9107	1.3067

Линейная регрессия (линеаризация нелинейных функций)

$$y = ax^2 + bx + c$$

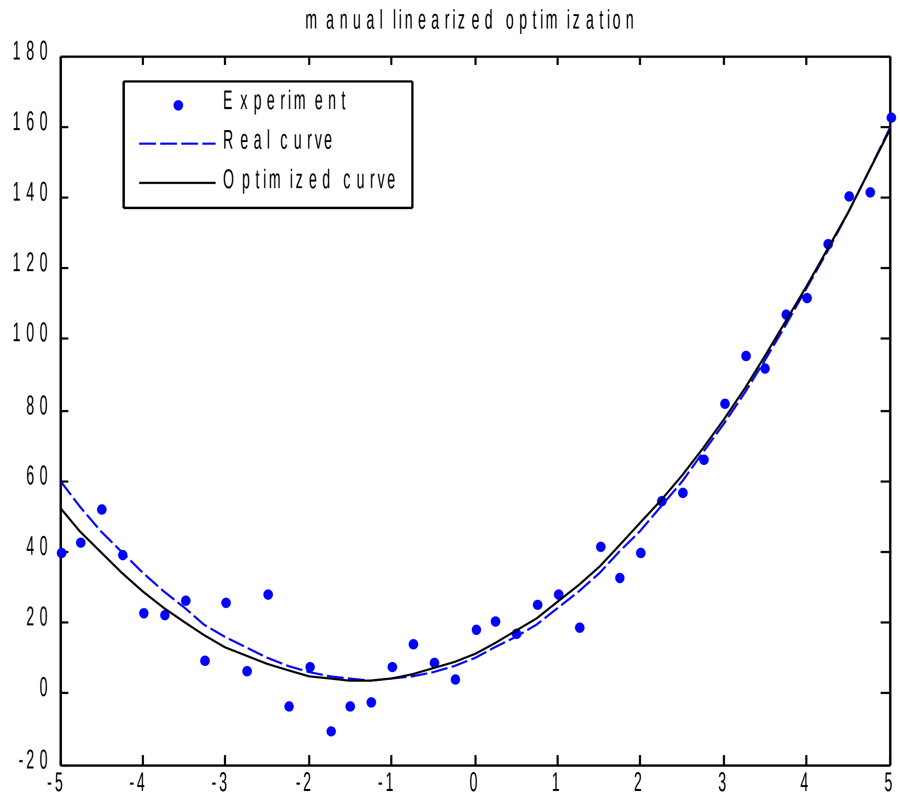
0. Исходные данные

```
x = (-5:0.25:5)';
yid = 4*x.^2 + 10*x + 10;
y = yid + randn(size(x))*10;
```

1. Построить матрицу X

$$X = \begin{pmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ \vdots & \vdots & \vdots \\ x_n^2 & x_n & 1 \end{pmatrix}$$

```
X = [x.^2 x ones(size(x))];
```



2. Найти коэффициенты регрессии β

3. Найти доверительные интервалы

```
manual linearized optimization
a = 3.782 +- 0.308
b = 10.702 +- 0.813
c = 11.321 +- 3.609
```

Можно использовать для полиномов, уравнения Аррениуса, уравнения Михаэлиса-Ментен и т. п.

Нелинейная регрессия в Statistics Toolbox

$$y = ax^2 + bx + c$$

0. Исходные данные

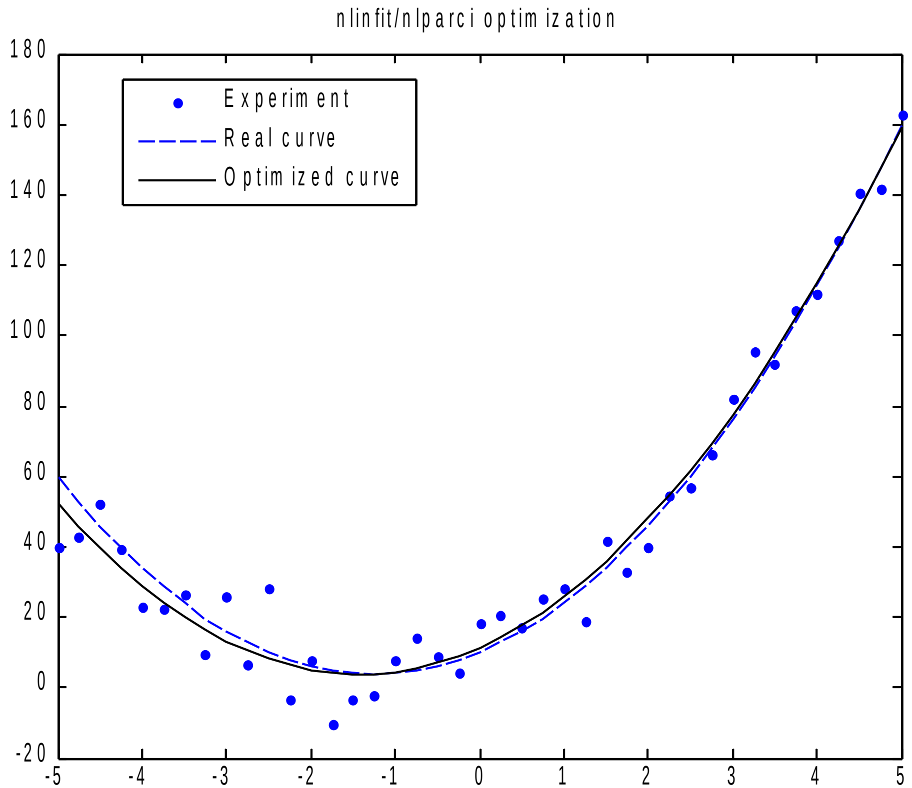
```
x=(-5:0.25:5)';
yid=4*x.^2+10*x+10;
y=yid+randn(size(x))*10;
func=@(b,x)(b(1)*x.^2+b(2)*x +b(3));
```

1. Найти коэффициенты регрессии β

```
[b, resid, J] = ...
  nlinfit(x, y, func,[0 0 0]);
```

2. Найти доверительные интервалы

```
ci = nlparci(b, resid, 'jacobian', J);
db = (ci(:,2) - ci(:,1))/2;
```




```
nlinfit/nlparci optimization
a = 3.782 +- 0.308
b = 10.702 +- 0.813
c = 11.321 +- 3.609
```

Якобиан J вместо матрицы X

$$J = \begin{pmatrix} \frac{\partial F_1(x_1)}{\partial \beta_1} & \frac{\partial F_1(x_1)}{\partial \beta_2} & \cdots & \frac{\partial F_1(x_1)}{\partial \beta_m} \\ \frac{\partial F_2(x_2)}{\partial \beta_1} & \frac{\partial F_2(x_2)}{\partial \beta_2} & \cdots & \frac{\partial F_2(x_2)}{\partial \beta_m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_n(x_n)}{\partial \beta_1} & \frac{\partial F_n(x_n)}{\partial \beta_2} & \cdots & \frac{\partial F_n(x_n)}{\partial \beta_m} \end{pmatrix}$$

Нелинейная регрессия в Optimization Toolbox

$$y = ax^2 + bx + c$$

0. Исходные данные

```
x=(-5:0.25:5)';
yid=4*x.^2+10*x+10;
y=yid+randn(size(x))*10;
func=@(b,x)(b(1)*x.^2+b(2)*x +b(3));
```

1. Найти коэффициенты регрессии β

```
[b,resnorm,resid,~,~,~,J] = ...
    lsqnonlin(@(b) func(b, x)-y,[0 0 0]);
```

2. Найти доверительные интервалы

Вариант 1 (Statistics Toolbox)

```
ci = nlparci(b, resid, 'jacobian', J);
db = (ci(:,2) - ci(:,1))/2;
```

$$J = \begin{pmatrix} \frac{\partial F_1(x_1)}{\partial \beta_1} & \frac{\partial F_1(x_1)}{\partial \beta_2} & \cdots & \frac{\partial F_1(x_1)}{\partial \beta_m} \\ \frac{\partial F_2(x_2)}{\partial \beta_1} & \frac{\partial F_2(x_2)}{\partial \beta_2} & \cdots & \frac{\partial F_2(x_2)}{\partial \beta_m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_n(x_n)}{\partial \beta_1} & \frac{\partial F_n(x_n)}{\partial \beta_2} & \cdots & \frac{\partial F_n(x_n)}{\partial \beta_m} \end{pmatrix}$$

Вариант 2 (MATLAB)

```
C = inv(J'*J);
fnum = numel(x) - numel(b);
sb = sqrt(diag(C)/fnum) * norm(resid);
t = tinv(1 - 0.05/2, fnum);
db = full(sb)*t;
```

$$C = (J^T J)^{-1}$$

$$\Delta\beta_j = s_{\beta,j}(X^T X)^{-1} t_{\alpha,n}$$

$$s_{\beta,j}^2 = C_{jj} \frac{\sum_i^n (y_i - F(x_i))^2}{f}$$

Сравнение рассмотренных методов

Преимущества	Недостаки
1. Линейная регрессия (операторы MATLAB)	
• Не требует начального приближения • Отличная сходимость • Высокое быстродействие • Не нужны дополнительные Toolbox • Поддерживается MATLAB Coder	• Только линейные модели • Необходимо всё делать в явном виде (в т.ч. расчёт доверительных интервалов)
2. MATLAB Statistics Toolbox	
• Поддержка нелинейных моделей • Автоматический расчёт доверительных интервалов	• Нужно начальное приближение • Требует Statistics Toolbox • Только подбор кривой • Не поддерживается MATLAB Coder
3. MATLAB Optimization Toolbox	
• Поддержка нелинейных моделей • Не только подбор кривой, но и минимизация суммы квадратов отклонений • Широкий выбор алгоритмов	• Нужно начальное приближение • Требует Optimization Toolbox • Расчёт доверительных интервалов вручную (или с помощью Statistics Toolbox) • Не поддерживается MATLAB Coder

Система ацетон-гексан

```
% System  n-hexane(1) — acetone(2)
% x=x(acetone)
   Pxy_exp=... % p(mm Hg), x, y
   [187.20, 0.0913, 0.3966
    226.70, 0.2563, 0.5421
    232.30, 0.3019, 0.5595
    232.40, 0.3543, 0.5737
    237.00, 0.4035, 0.5827
    238.80, 0.5325, 0.6092
    237.70, 0.6609, 0.6362
    239.30, 0.7309, 0.6564
    237.90, 0.7679, 0.6722
    234.30, 0.7862, 0.6823
    234.10, 0.8219, 0.6975
    230.30, 0.8528, 0.7202
    220.60, 0.9105, 0.7778
    202.90, 0.9619, 0.8739
   ];
   R=1.987; % cal/(K*mol)
   t=20; % Temperature, C
   p10=119.60; % P(s) for (1)
   p20=181.50; % P(s) for (2)
```

```
param =

    1.8629
    0.1393


ci =

    1.7094    2.0163
   -0.0936    0.3723
```

Модель

$$\frac{G^{ex}}{RT}=x(1-x)(A+Bx)$$

$$\ln\gamma_1=x^2\left(A+B(2x-1)\right)$$

$$\ln\gamma_2=(1-x)^2(A+2Bx)$$

Линеаризация

$$\alpha=\frac{G^{ex}}{RTx(1-x)}=A+Bx$$

$$\frac{G^{ex}}{RT}=x_1\ln\gamma_1+x_2\ln\gamma_2$$

$$\gamma_1=\frac{py_1}{p_1^{(0)}x_1}\quad\gamma_2=\frac{py_2}{p_2^{(0)}x_2}$$