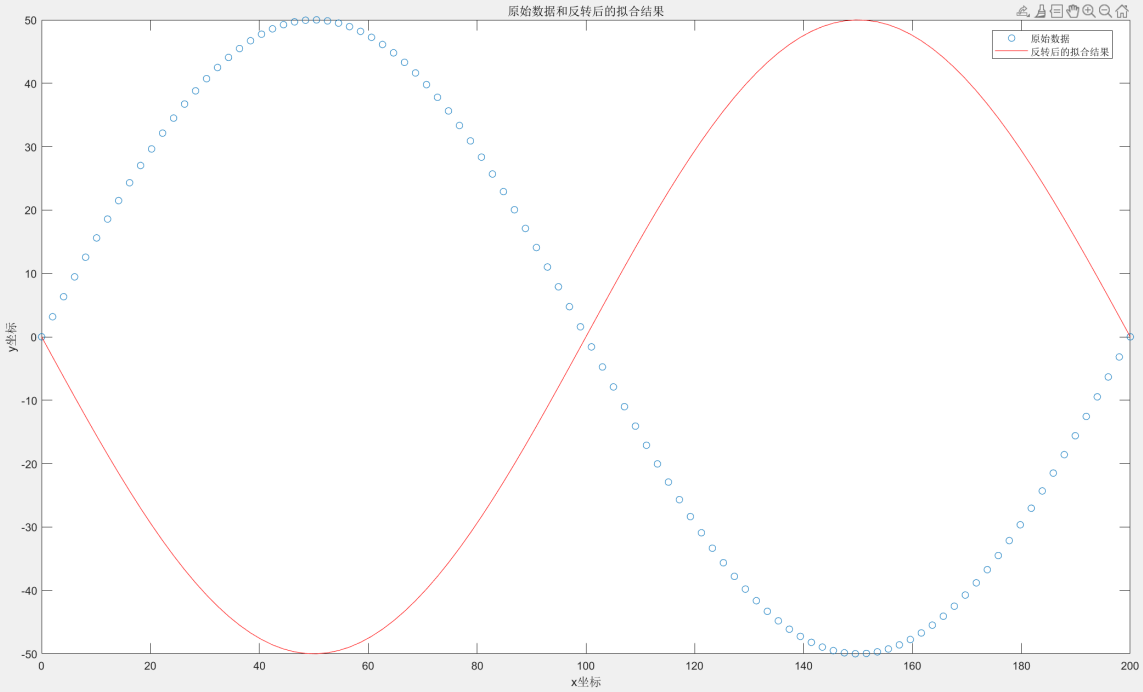
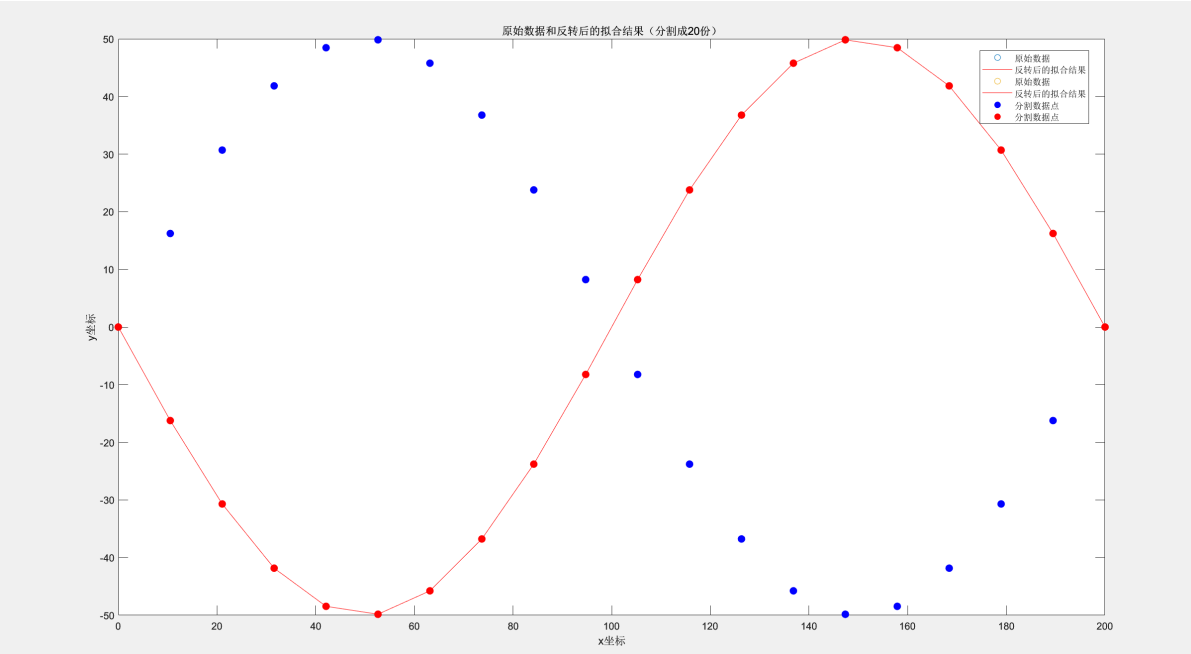
```
% 创建示例数据点
x = linspace(0, 200, 100); % 创建一个周期为200的x范围
y = 50 * sin(2*pi*x/200); % 正弦函数，波峰在50

% 定义正弦函数模型
sinusoidal_model = fittype('a * sin(2*pi*x/b)', 'independent', 'x',
'coefficients', {'a', 'b'});

% 初始化拟合参数的起始值
initial_guess = [50, 200];

% 使用fit函数进行非线性拟合
fit_obj = fit(x', y', sinusoidal_model, 'StartPoint', initial_guess);

% 反转y轴上的数据
y_flipped = -y;

% 绘制原始数据和反转后的拟合结果在同一张图上
plot(x, y, 'o', 'DisplayName', '原始数据');
hold on;
plot(x, y_flipped, 'r', 'DisplayName', '反转后的拟合结果');
legend('Location', 'Best');
xlabel('x坐标');
ylabel('y坐标');
title('原始数据和反转后的拟合结果');
```



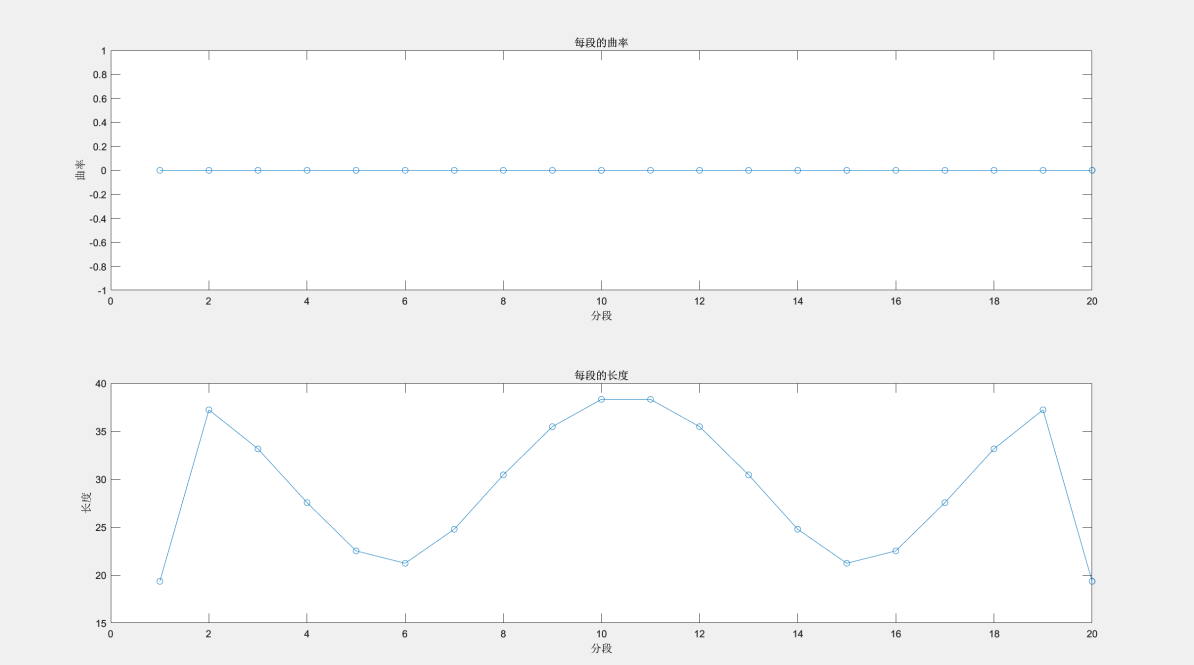

```
% 创建示例数据点
x_original = linspace(0, 200, 100); % 创建一个周期为200的x范围
y_original = 50 * sin(2*pi*x_original/200); % 正弦函数，波峰在50

% 定义正弦函数模型
```

```
sinusoidal_model = fittype('a * sin(2*pi*x/b)', 'independent', 'x',
'coefficients', {'a', 'b'});

% 初始化拟合参数的起始值
initial_guess = [50, 200];

% 使用fit函数进行非线性拟合
fit_obj = fit(x_original', y_original', sinusoidal_model, 'StartPoint',
initial_guess);

% 反转y轴上的数据
y_flipped = -y_original;

% 将曲线分割成20份
num_segments = 20;
x_segments = linspace(0, 200, num_segments);
y_original_segments = interp1(x_original, y_original, x_segments, 'linear');
y_flipped_segments = interp1(x_original, y_flipped, x_segments, 'linear');

% 绘制原始数据和反转后的拟合结果在同一张图上
plot(x_segments, y_original_segments, 'o', 'DisplayName', '原始数据');
hold on;
plot(x_segments, y_flipped_segments, 'r', 'DisplayName', '反转后的拟合结果');
legend('Location', 'Best');
xlabel('x坐标');
ylabel('y坐标');
title('原始数据和反转后的拟合结果（分割成20份）');

% 绘制分割后的数据点
scatter(x_segments, y_original_segments, 50, 'filled', 'MarkerEdgeColor', 'b',
'MarkerFaceColor', 'b', 'DisplayName', '分割数据点');
scatter(x_segments, y_flipped_segments, 50, 'filled', 'MarkerEdgeColor', 'r',
'MarkerFaceColor', 'r', 'DisplayName', '分割数据点');
```



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	
y	0	16.22934607	30.69549499	41.83878112	48.45708373	49.82423002	45.77086576	36.76774549	23.78686614	8.228012624	-8.228012624	-23.78686614	-36.76774549	-45.77086576	-49.82423002	-48.45708373	-41.83878112	-30.69549499	-16.22934607	0	
x	0	10.52631579	21.05263158	31.57894737	42.10526316	52.63157895	63.15789474	73.68421053	84.21052632	94.73684211	105.2631579	115.7894737	126.3157895	136.8421053	147.3684211	157.8947368	168.4210526	178.9473684	189.4736842	200	

```
% 创建示例数据点
x_original = linspace(0, 200, 100); % 创建一个周期为200的x范围
y_original = 50 * sin(2*pi*x_original/200); % 正弦函数，波峰在50

% 定义分割的数量
num_segments = 20;

% 将曲线分割成20份
x_segments = linspace(0, 200, num_segments);
y_segments = interp1(x_original, y_original, x_segments, 'linear');

% 初始化曲率和长度的数组
curvature = zeros(1, num_segments);
segment_lengths = zeros(1, num_segments);

% 计算每段的曲率和长度
for i = 1:num_segments
    if i == 1
        x1 = x_segments(1);
        y1 = y_segments(1);
        x2 = x_segments(2);
        y2 = y_segments(2);
    elseif i == num_segments
        x1 = x_segments(end - 1);
        y1 = y_segments(end - 1);
        x2 = x_segments(end);
        y2 = y_segments(end);
    else
        x1 = x_segments(i - 1);
        y1 = y_segments(i - 1);
        x2 = x_segments(i + 1);
        y2 = y_segments(i + 1);
    end

    % 曲线段的长度
    segment_lengths(i) = sqrt((x2 - x1)^2 + (y2 - y1)^2);

    % 计算曲率
    curvature(i) = 2 * abs((x1 - x2) * (y2 - y1) - (y1 - y2) * (x2 - x1)) / ((x1
- x2)^2 + (y1 - y2)^2)^1.5;
end

% 绘制曲率和长度
subplot(2, 1, 1); % 第一个子图，曲率
plot(curvature, 'o-');
xlabel('分段');
ylabel('曲率');
title('每段的曲率');

subplot(2, 1, 2); % 第二个子图，长度
plot(segment_lengths, 'o-');
xlabel('分段');
ylabel('长度');
title('每段的长度');
```



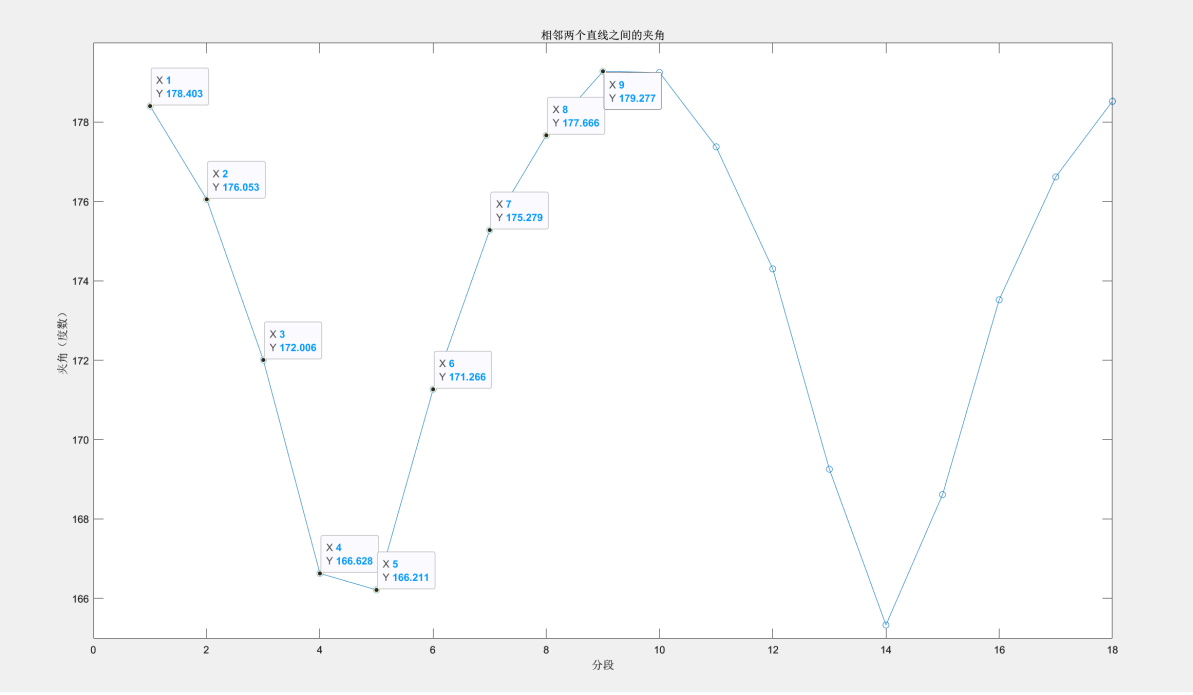
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
长度	19.3441205	37.22132063	33.15202044	27.54427946	22.51623172	21.22331415	24.7726681	30.43861915	35.46448432	38.31665119	38.31665119	35.46448432	30.43861915	24.7726681	21.22331415	22.51623172	27.54427946	33.15202044	37.22132063	19.3441205
时间	0.5	0.98	0.88	0.72	0.6	0.55	0.66	0.8	0.93	1	1	0.93	0.8	0.66	0.56	0.58	0.71	0.88	0.98	0.5

50% v=37.6 cm/s

两个轮子间距：47cm

```
% 创建示例数据点
x_original = linspace(0, 200, 100); % 创建一个周期为200的x范围
y_original = 50 * sin(2*pi*x_original/200); % 正弦函数，波峰在50

% 定义分割的数量
num_segments = 20;

% 将曲线分割成20份
x_segments = linspace(0, 200, num_segments);
y_segments = interp1(x_original, y_original, x_segments, 'linear');

% 初始化夹角数组
angles_deg = zeros(1, num_segments - 1); % 存储夹角（度数）

% 计算相邻两个直线之间的夹角
for i = 1:num_segments - 1
    if i == num_segments - 1
        i_next = i + 1;
    else
        i_next = i + 2;
    end

    % 计算第i段的方向向量
    v1 = [x_segments(i_next) - x_segments(i), y_segments(i_next) -
y_segments(i)];

    % 计算第i+1段的方向向量
    v2 = [x_segments(i + 1) - x_segments(i_next), y_segments(i + 1) -
y_segments(i_next)];
```

```
    % 计算夹角的余弦值
    cos_theta = dot(v1, v2) / (norm(v1) * norm(v2));

    % 计算夹角的度数
    angle_deg = acosd(cos_theta);

    angles_deg(i) = angle_deg;
end

% 绘制夹角
plot(1:num_segments-1, angles_deg, 'o-');
xlabel('分段');
ylabel('夹角（度数）');
title('相邻两个直线之间的夹角');
```



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
角度	0.027875159	0.068882739	0.13952753	0.233386994	0.240659862	0.152432972	0.082402609	0.04074433	0.012617388	0.013120869	0.04580075	0.099475185	0.187558454	0.256065511	0.19874043	0.113051303	0.0590074	0.025780101	0
L	36.94493376	35.98125563	34.32110305	32.11540564	31.94449324	34.01782516	35.66353869	36.64250825	37.30349138	37.29165958	36.52368238	35.26233315	33.19237633	31.58246049	32.9295999	34.94329438	36.2133261	36.99416763	37.5
V_L	49.25991168	47.97500751	45.76147073	42.82054085	42.59265765	45.35710021	47.55138492	48.85667767	49.73798851	49.72221277	48.69824317	47.0164442	44.25650177	42.10994732	43.90613324	46.59105917	48.28443484	49.32555684	50
R	38.25506624	39.21874437	40.87889696	43.08459436	43.25550676	41.18217484	39.53646131	38.55749176	37.89650862	37.90834042	38.67631763	39.93766685	42.00762367	43.61753951	42.27040011	40.25670562	38.9866739	38.20583237	37.5
V_R	51.00675499	52.29165916	54.50519595	57.44612581	57.67400901	54.90956645	52.71528175	51.40998901	50.52867816	50.54445389	51.56842351	53.25022247	56.01016489	58.15671935	56.36053348	53.67560749	51.98223187	50.94110983	50