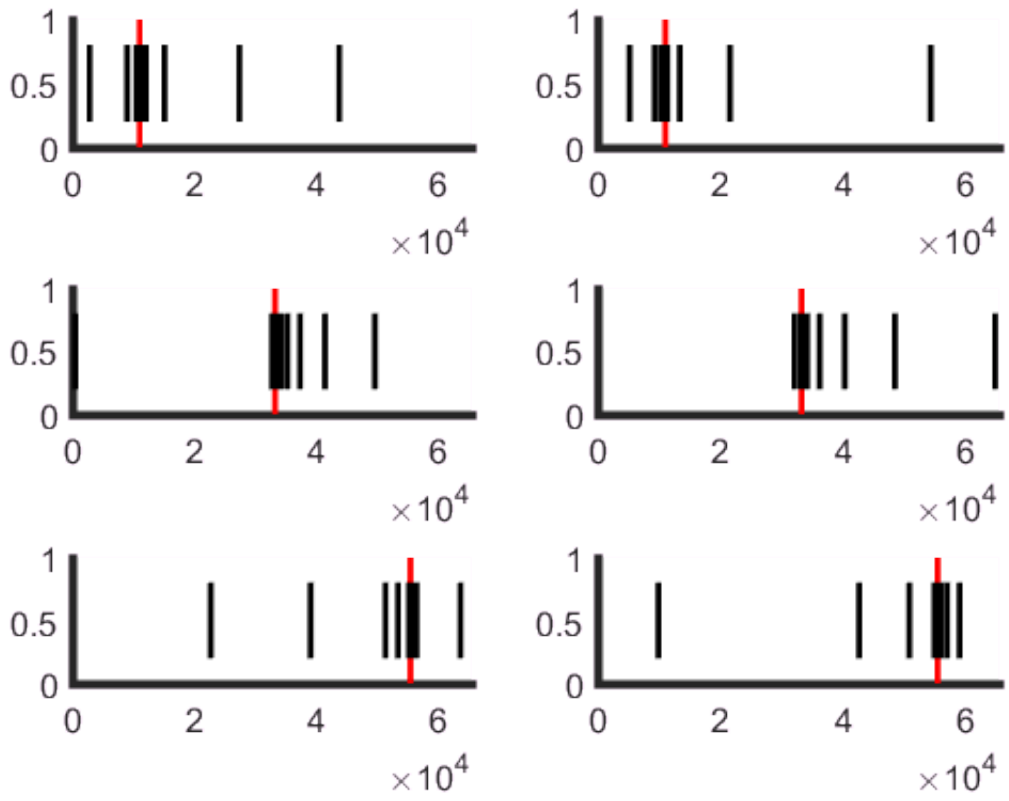
```
% gda09_17
% plots the distributions of one bit-mutations
% for both normal binary and Gray code representations
% of three exemplary 16-bit integers
% Supports Figure 9.15

clear all;

global gray1 gray1index pow2;
load('gda_gray1table.mat');

figure(1);
clf;

subplot(3,2,1);
hold on;
set(gca,'LineWidth',3);
set(gca,'FontSize',12);
axis( [0, 65535, 0, 1] );

i1 = 11111;
plot( [i1,i1]', [0, 0.2]', 'r-', 'LineWidth', 2 );
plot( [i1,i1]', [0.8, 1]', 'r-', 'LineWidth', 2 );
g1=gda_int2bin(i1);

for i=[1:16]
    g2 = g1;
    k=i;
    if( g1(k) == '0' );
        g2(k) = '1';
    else
        g2(k) = '0';
    end
    i2=gda_bin2int(g2);
    plot( [i2,i2]', [0.2, 0.8]', 'k-', 'LineWidth', 2 );
end

subplot(3,2,3);
hold on;
set(gca,'LineWidth',3);
set(gca,'FontSize',12);
axis( [0, 65535, 0, 1] );

i1 = 33333;
plot( [i1,i1]', [0, 0.2]', 'r-', 'LineWidth', 2 );
plot( [i1,i1]', [0.8, 1]', 'r-', 'LineWidth', 2 );
g1=gda_int2bin(i1);

for i=[1:16]
    g2 = g1;
    k=i;
    if( g1(k) == '0' );
        g2(k) = '1';
    else
        g2(k) = '0';
    end
    i2=gda_bin2int(g2);
    plot( [i2,i2]', [0.2, 0.8]', 'k-', 'LineWidth', 2 );
end
```

```
subplot(3,2,5);
hold on;
set(gca,'LineWidth',3);
set(gca,'FontSize',12);
axis( [0, 65535, 0, 1] );

i1 = 55555;
plot( [i1,i1]', [0, 0.2]', 'r-', 'LineWidth', 2 );
plot( [i1,i1]', [0.8, 1]', 'r-', 'LineWidth', 2 );
g1=gda_int2bin(i1);

for i=[1:16]
    g2 = g1;
    k=i;
    if( g1(k) == '0' );
        g2(k) = '1';
    else
        g2(k) = '0';
    end
    i2=gda_bin2int(g2);
    plot( [i2,i2]', [0.2, 0.8]', 'k-', 'LineWidth', 2 );
end

subplot(3,2,2);
hold on;
set(gca,'LineWidth',3);
set(gca,'FontSize',12);
axis( [0, 65535, 0, 1] );

i1 = 11111;
plot( [i1,i1]', [0, 0.2]', 'r-', 'LineWidth', 2 );
plot( [i1,i1]', [0.8, 1]', 'r-', 'LineWidth', 2 );
g1=gda_int2gray(i1);

for i=[1:16]
    g2 = g1;
    k=i;
    if( g1(k) == '0' );
        g2(k) = '1';
    else
        g2(k) = '0';
    end
    i2=gda_gray2int(g2);
    plot( [i2,i2]', [0.2, 0.8]', 'k-', 'LineWidth', 2 );
end

subplot(3,2,4);
hold on;
set(gca,'LineWidth',3);
set(gca,'FontSize',12);
axis( [0, 65535, 0, 1] );

i1 = 33333;
plot( [i1,i1]', [0, 0.2]', 'r-', 'LineWidth', 2 );
plot( [i1,i1]', [0.8, 1]', 'r-', 'LineWidth', 2 );
g1=gda_int2gray(i1);

for i=[1:16]
    g2 = g1;
    k=i;
```

```
    if( g1(k) == '0' );
        g2(k) = '1';
    else
        g2(k) = '0';
    end
    i2=gda_gray2int(g2);
    plot( [i2,i2]', [0.2, 0.8]', 'k-', 'LineWidth', 2 );
end

subplot(3,2,6);
hold on;
set(gca,'LineWidth',3);
set(gca,'FontSize',12);
axis( [0, 65535, 0, 1] );

i1 = 55555;
plot( [i1,i1]', [0, 0.2]', 'r-', 'LineWidth', 2 );
plot( [i1,i1]', [0.8, 1]', 'r-', 'LineWidth', 2 );
g1=gda_int2gray(i1);

for i=[1:16]
    g2 = g1;
    k=i;
    if( g1(k) == '0' );
        g2(k) = '1';
    else
        g2(k) = '0';
    end
    i2=gda_gray2int(g2);
    plot( [i2,i2]', [0.2, 0.8]', 'k-', 'LineWidth', 2 );
end
```



```
% Figure 9.15 Effect on one-bit mutations of the 16-bit standard binary code for the numbers
% (A) 11,111, (B) 33,333 and (C) 55,555. (C)-(E) Same as (A)-(C), except for the 16-bit Gray
% code. In each case, the original number is shown in red and the mutations in black. MatLab
% script gda09_17.
```