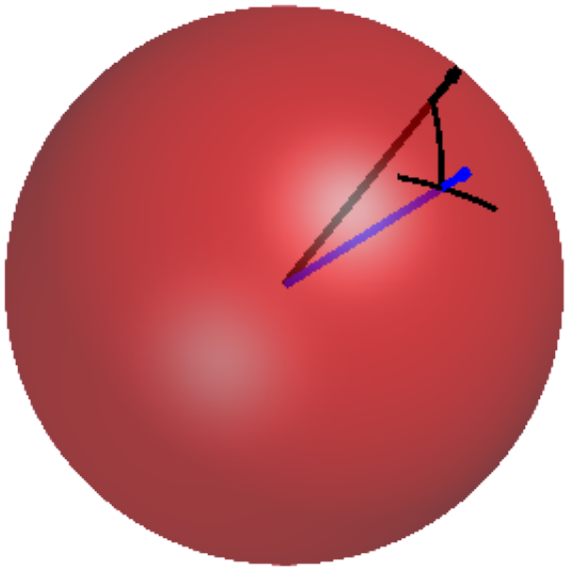
```
% gda12_09
% depicts vector relationships in 3D space
% Supports Figure 12.13

clear all;

% independent variable x
Nx = 51;
xmin = -1;
xmax = 1;
Dx = (xmax-xmin)/(Nx-1);
x = xmin + Dx*[0:Nx-1]';

% independent variable y
Ny = 51;
ymin = -1;
ymax = 1;
Dy = (ymax-ymin)/(Ny-1);
y = ymin + Dy*[0:Ny-1]';

% independent variable z
Nz = 51;
zmin = -1;
zmax = 1;
Dz = (zmax-zmin)/(Nz-1);
z = zmin + Dz*[0:Nz-1]';

figure(1);
clf;
set(gca,'LineWidth',2);
hold on;
axis( [xmin, xmax, ymin, ymax, zmin, zmax]');
axis off;

% central direction
q1=45;
f1=45;
b1=1;
a1= [b1*cos(pi*f1/180)*sin(pi*q1/180), b1*sin(pi*f1/180)*sin(pi*q1/180), b1*cos(pi*q1/180)]';
arrow3( a1, 'k-', 3);

% another vector
q2=65;
f2=35;
b2=1;
a2= [b2*cos(pi*f2/180)*sin(pi*q2/180), b2*sin(pi*f2/180)*sin(pi*q2/180), b2*cos(pi*q2/180)]';
arrow3( a2, 'b-', 3);

% co-latitudinal arc from central vector to the other vector
lena1 = sqrt(a1'*a1);
lena2 = sqrt(a2'*a2);
ua1 = a1/lena1;
ua2 = a2/lena2;
th=180*acos((ua1'*ua2))/pi;
Nth = floor(th)+1;
alpha = [0:Nth-1]'/(Nth-1);
arc1=zeros(Nth,3);
b3 = 0.85;
arc1 = alpha*ua1' + (1-alpha)*ua2';
```

```
lenarc1 = sqrt( arc1(:,1).^2 + arc1(:,2).^2 + arc1(:,3).^2 );
arc1 = b3 * arc1 ./ (lenarc1*ones(1,3));
plot3( arc1(:,1), arc1(:,2), arc1(:,3), 'k-', 'LineWidth', 2 );

% longitudinal arc
a3 = cross(a1/lena1,a2/lena2);
lena3 = sqrt(a3'*a3);
ua3 = a3/lena3;
Nth=20;
alpha = [-Nth:Nth-1]'/90;
arc2=zeros(Nth,3);
arc2 = alpha*ua3' + (1-alpha)*ua2';
lenarc2 = sqrt( arc2(:,1).^2 + arc2(:,2).^2 + arc2(:,3).^2 );
arc2 = b3 * arc2 ./ (lenarc2*ones(1,3));
plot3( arc2(:,1), arc2(:,2), arc2(:,3), 'k-', 'LineWidth', 2 );

% depict sphere in 3D space by contouring a Normal distribution

% 3D grid
[XX, YY, ZZ] = meshgrid( x, y, z );

% parameters for Normal distribution
rbar = [0, 0, 0]';
sd=1;
C = (sd^2)*eye(3,3);
CI = inv(C);
DC = det(C);
norm = ( ((2*pi)^(3/2)) * sqrt(DC) );

% tabulate Normal distribution
PP = zeros(Nx,Ny,Nz);
for i=[1:Nx]
for j=[1:Ny]
for k=[1:Nz]
    r = [XX(i,j,k), YY(i,j,k), ZZ(i,j,k)]';
    PP(i,j,k) = exp(-0.5*(r-rbar)'*CI*(r-rbar))/norm;
end
end
end

% contour Normal distribution in 3D space
maxP=max(max(max(PP)));
p=patch(isosurface( XX, YY, ZZ, PP, 0.7*maxP ));
isonormals(XX,YY,ZZ,PP, p)
set(p, 'FaceColor', 'red', 'FaceAlpha', 0.5, 'EdgeColor', 'none');

% set view angle, etc
daspect([1 1 1])
view(87,6)
camlight; lighting phong
```



```
% Figure 12.13 Unit sphere, showing coordinate system used in Fisher distribution. MatLab scri
```