

## 2D and 3D Measurements

©1999 Bill Davis, Horacio Porta and Jerry Uhl

Produced by Bruce Carpenter

Published by Math Everywhere, Inc.

www.matheverywhere.com

### VC.09 Spherical Coordinates Basics

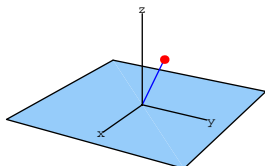
#### B.1) Spherical coordinates

You can specify a point in three dimensions as usual by its three coordinates  $\{x, y, z\}$ .

Another way to specify a point is to plant the  $x, y,$  and  $z$ -axes at  $\{0, 0, 0\}$  and then run a stick from the  $\{0, 0, 0\}$  to the point:

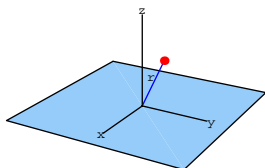
These instructions will not run unless you have run the initializations.

```
Show[stickplot, DisplayFunction -> $DisplayFunction];
```



Then you measure the length  $r$  of this stick:

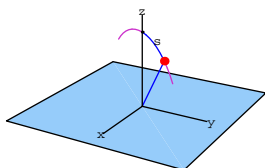
```
Show[rlabel, DisplayFunction -> $DisplayFunction];
```



Grab both plots, animate and run at speed 2 or any other speed you like.

Next measure the angle  $s$  that this stick makes with the  $z$ -axis:

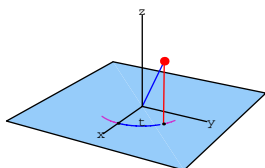
```
Show[slabel, DisplayFunction -> $DisplayFunction];
```



Grab all three plots, animate and run at speed 2 or any other speed you like.

Finally measure the angle  $t$  that the shadow of this ray on the  $xy$ -plane makes with the  $x$ -axis:

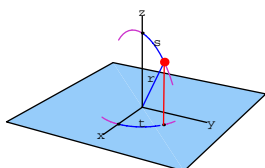
```
Show[tlabel, DisplayFunction -> $DisplayFunction];
```



Grab all four plots, animate and run at speed 2 or any other speed you like.

Here are all three measurements  $r, s,$  and  $t$  together:

```
Show[all, DisplayFunction -> $DisplayFunction];
```



Grab all five plots, animate and run at speed 2 or any other speed you like.

If you know the  $r, s,$  and  $t$  measurements for a point, then you can recover the  $\{x, y, z\}$  coordinates via:

$$z = r \cos[s],$$

$$x = r \sin[s] \cos[t], \text{ and}$$

$$y = r \sin[s] \sin[t].$$

Most folks call the measurements  $\{r, s, t\}$  the spherical coordinates of the point  $\{x, y, z\}$ .

#### □B.1.a.i)

Describe the sphere

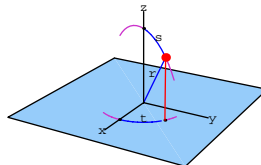
$$x^2 + y^2 + z^2 = 9$$

in spherical coordinates.

□Answer:

Remember that the spherical coordinates  $r, s,$  and  $t$  reflect the following measurements:

```
Show[all, DisplayFunction -> $DisplayFunction];
```



You get the whole sphere

$$x^2 + y^2 + z^2 = 9$$

by setting

$$r = 3,$$

running  $s$  from 0 to  $\pi,$  and

running  $t$  from 0 to  $2\pi.$

So the description of  $x^2 + y^2 + z^2 = 9$  in spherical coordinates is

$$r = 3,$$

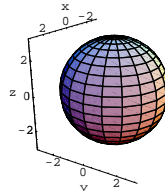
$$0 \leq s \leq \pi, \text{ and}$$

$$0 \leq t \leq 2\pi.$$

Check by plotting in  $xyz$ -space:

```
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
{x Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
{{slow, shigh}, {tlow, thigh}} = {{0, pi}, {0, 2 pi}};
Clear[sphereplotter];
sphereplotter[s_, t_] = {x[3, s, t], y[3, s, t], z[3, s, t]};
```

```
sphereplot =
ParametricPlot3D[Evaluate[sphereplotter[s, t]], {s, slow, shigh},
{t, tlow, thigh}, Boxed -> False, BoxRatios -> Automatic,
ViewPoint -> CMView, AxesLabel -> {"x", "y", "z"}];
```



Looks good and feels great.

#### □B.1.a.ii)

Describe the part of the sphere

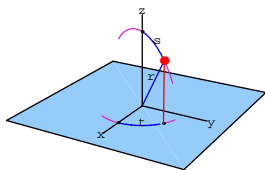
$$x^2 + y^2 + z^2 = 16$$

consisting of those points with  $z \geq 0$  in spherical coordinates.

□Answer:

Remember that the spherical coordinates  $r, s,$  and  $t$  reflect the following measurements:

```
Show[all, DisplayFunction -> $DisplayFunction];
```



To get the description of the part of the sphere

$$x^2 + y^2 + z^2 = 16$$

consisting of those points with  $z \geq 0$  in spherical coordinates, you set  $r = 4$ , run  $s$  from 0 to  $\frac{\pi}{2}$ , and run  $t$  from 0 to  $2\pi$ . This results in:

$$r = 4,$$

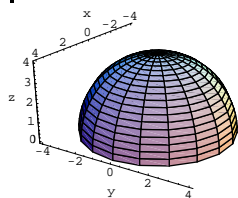
$$0 \leq s \leq \frac{\pi}{2}, \text{ and}$$

$$0 \leq t \leq 2\pi.$$

Check:

```
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
{r Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
{{slow, shigh}, {tlow, thigh}} = {{0,  $\frac{\pi}{2}$ }, {0, 2  $\pi$ }};
Clear[sphereplotter];
sphereplotter[s_, t_] = {x[4, s, t], y[4, s, t], z[4, s, t]};

hemisphereplot =
ParametricPlot3D[Evaluate[sphereplotter[s, t]], {s, slow, shigh},
{t, tlow, thigh}, Boxed -> False, BoxRatios -> Automatic,
ViewPoint -> CMView, AxesLabel -> {"x", "y", "z"}];
```



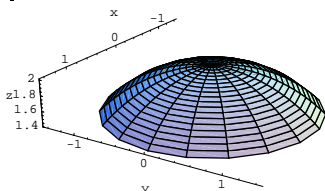
No problem-o.

□B.1.b.i)

Look at:

```
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
{r Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
{{slow, shigh}, {tlow, thigh}} = {{0,  $\frac{\pi}{4}$ }, {0, 2  $\pi$ }};
Clear[plotter];
plotter[s_, t_] = {x[2, s, t], y[2, s, t], z[2, s, t]};

ParametricPlot3D[Evaluate[plotter[s, t]], {s, slow, shigh},
{t, tlow, thigh}, Boxed -> False, BoxRatios -> Automatic,
ViewPoint -> CMView, AxesLabel -> {"x", "y", "z"}];
```



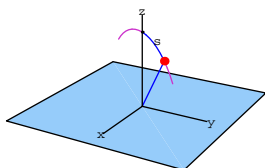
Explain why the plot turned out the way it did.

□Answer:

Note that  $s$  is allowed to run only from 0 to  $\frac{\pi}{4}$ .

Remember that  $s$  is the indicated angle:

```
Show[slabel, DisplayFunction -> $DisplayFunction];
```



Limiting  $s$  to the interval  $0 \leq s \leq \frac{\pi}{4}$  in the plot results in a plot of the top cap of the sphere

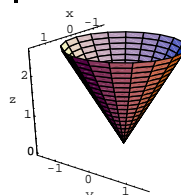
$$x^2 + y^2 + z^2 = 2^2.$$

Play with this by plotting with other intervals like  $\frac{\pi}{4} \leq s \leq \frac{\pi}{2}$ .

□B.1.b.ii)

Look at:

```
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
{r Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
{{rlow, rhigh}, {tlow, thigh}} = {{0, 3}, {0, 2  $\pi$ }};
Clear[plotter];
plotter[r_, t_] = {x[r,  $\frac{\pi}{6}$ , t], y[r,  $\frac{\pi}{6}$ , t], z[r,  $\frac{\pi}{6}$ , t]};
ParametricPlot3D[Evaluate[plotter[r, t]], {r, rlow, rhigh},
{t, tlow, thigh}, Boxed -> False, BoxRatios -> Automatic,
ViewPoint -> CMView, AxesLabel -> {"x", "y", "z"}];
```

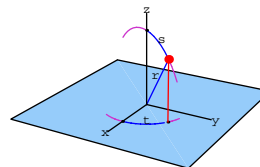


Pack in some good Baskin Robbins Ice Cream. Explain why the plot turned out the way it did.

□Answer:

This time  $s$  is held at  $\frac{\pi}{6}$ , but  $t$  and  $r$  are allowed to roam.

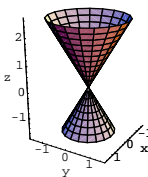
```
Show[all, DisplayFunction -> $DisplayFunction];
```



Holding  $s = \frac{\pi}{6}$  results in a cone whose sides make an angle of  $\frac{\pi}{6}$  radians with the  $z$ -axis.

For some variations, try:

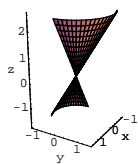
```
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
{r Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
{{rlow, rhigh}, {tlow, thigh}} = {{-2, 3}, {0, 2  $\pi$ }};
Clear[plotter];
plotter[r_, t_] = {x[r,  $\frac{\pi}{6}$ , t], y[r,  $\frac{\pi}{6}$ , t], z[r,  $\frac{\pi}{6}$ , t]};
ParametricPlot3D[Evaluate[plotter[r, t]], {r, rlow, rhigh},
{t, tlow, thigh}, Boxed -> False, BoxRatios -> Automatic,
ViewPoint -> CMView, AxesLabel -> {"x", "y", "z"}];
```



And:

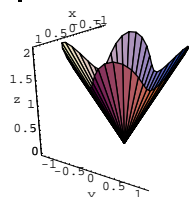
```
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
{r Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
{{rlow, rhigh}, {tlow, thigh}} = {{-2, 3}, {0,  $\frac{2\pi}{3}$ }};
Clear[plotter];
plotter[r_, t_] = {x[r,  $\frac{\pi}{6}$ , t], y[r,  $\frac{\pi}{6}$ , t], z[r,  $\frac{\pi}{6}$ , t]};
```

```
ParametricPlot3D[Evaluate[plotter[r, t]], {r, rlow, rhigh},
{t, tlow, thigh}, Boxed → False, BoxRatios → Automatic,
ViewPoint → CMView, AxesLabel → {"x", "y", "z"}];
```



And you can go a little nuts:

```
Clear[x, y, z, r, s, t, rad]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
{r Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
{tlow, thigh} = {0, 2 π};
rad[t_] = 2 + 0.4 Sin[4 t];
Clear[plotter];
plotter[r_, t_] =
{x[r rad[t], π/6, t], y[r rad[t], π/6, t], z[r rad[t], π/6, t]};
ParametricPlot3D[Evaluate[plotter[r, t]], {t, tlow, thigh}, {r, 0, 1},
PlotPoints → {50, 2}, Boxed → False, BoxRatios → Automatic,
ViewPoint → CMView, AxesLabel → {"x", "y", "z"}];
```



Play.

## B.2) Integrating with spherical coordinates

Here is the volume conversion factor  $V_{xyz}[r, s, t]$  for converting spherical coordinate  $r$ - $s$ - $t$  volume measurements into standard  $xyz$ -volume measurements:

```
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
{r Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
Clear[gradx, grady, gradz, Vxyz]
gradx[r_, s_, t_] =
{D[x[r, s, t], r], D[x[r, s, t], s], D[x[r, s, t], t]};
grady[r_, s_, t_] =
{D[y[r, s, t], r], D[y[r, s, t], s], D[y[r, s, t], t]};
gradz[r_, s_, t_] =
{D[z[r, s, t], r], D[z[r, s, t], s], D[z[r, s, t], t]};
Vxyz[r_, s_, t_] =
Abs[TrigExpand[Det[{gradx[r, s, t], grady[r, s, t], gradz[r, s, t]}]]]
```

$Abs[r^2 Sin[s]]$

In most situations you can get by with  $0 \leq s \leq \pi$ , so you usually don't have to multiply by  $-1$ , because

$$\sin[s] \geq 0 \text{ for } 0 \leq s \leq \pi.$$

With these ground rules,

$$V_{xyz}[r, s, t]$$

is given by:

```
Clear[Vxyz]
Vxyz[r_, s_, t_] = r^2 Sin[s]
r^2 Sin[s]
```

### B.2.a)

Use spherical coordinates to get a quick evaluation of

$$\iiint_{R_{xyz}} x^2 dx dy dz$$

where  $R_{xyz}$  is everything inside and on the sphere

$$x^2 + y^2 + z^2 = a^2.$$

□ Answer:

If you have an integral

$$\iiint_{R_{xyz}} f[x, y, z] dx dy dz$$

and  $R_{xyz}$  is a solid sphere, then spherical coordinates are the way to go.

Enter  $f[x, y, z] = x^2$ :

```
Clear[a, x, y, z, r, s, t, f, Vxyz]
f[x_, y_, z_] = x^2
x^2
```

Remember that  $R_{xyz}$  can be described by

$$0 \leq r \leq a$$

$$0 \leq s \leq \pi$$

$$0 \leq t \leq 2\pi.$$

When you set  $r = a$ , all you get is the skin of the sphere.

But when you run  $r$  from  $0$  to  $a$ , then you get the sphere and all that good stuff inside.

So

$$\begin{aligned} & \iiint_{R_{xyz}} f[x, y, z] dx dy dz \\ &= \int_0^{2\pi} \int_0^\pi \int_0^a f[x[r, s, t], y[r, s, t], z[r, s, t]] \\ & \quad V_{xyz}[r, s, t] dr ds dt \end{aligned}$$

Put in the spherical coordinate change of variables:

```
{x[r_, s_, t_] = r Sin[s] Cos[t];
y[r_, s_, t_] = r Sin[s] Sin[t];
z[r_, s_, t_] = r Cos[s];
Vxyz[r_, s_, t_] = r^2 Sin[s]
r^2 Sin[s]
```

Calculate

$$\begin{aligned} & \iiint_{R_{xyz}} f[x, y, z] dx dy dz \\ &= \int_0^{2\pi} \int_0^\pi \int_0^a f[x[r, s, t], y[r, s, t], z[r, s, t]] \\ & \quad V_{xyz}[r, s, t] dr ds dt \end{aligned}$$

```
Clear[a]
Integrate[Integrate[Integrate[f[x[r, s, t], y[r, s, t], z[r, s, t]] Vxyz[r, s, t] dr ds dt,
{r, 0, a}], {s, 0, π}], {t, 0, 2 π}]
4 a^5 π / 15
```

The computer did almost all the work, just the way it should. After all, what's a computer for?

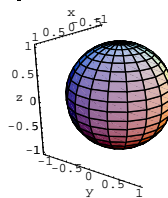
# VC.09 Spherical Coordinates Tutorials

## T.1) Balls and eggs

### □ T.1.a) Centering a sphere at a given point

Here's a plot of the sphere of radius 1 centered at  $\{0, 0, 0\}$ :

```
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
{r Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
{{slow, shigh}, {tlow, thigh}} = {{0, π}, {0, 2 π}};
Clear[firstsphereplotter];
firstsphereplotter[s_, t_] = {x[1, s, t], y[1, s, t], z[1, s, t]};
firstsphereplot = ParametricPlot3D[
Evaluate[firstsphereplotter[s, t]], {s, slow, shigh},
{t, tlow, thigh}, Boxed → False, BoxRatios → Automatic,
ViewPoint → CMView, AxesLabel → {"x", "y", "z"}];
```



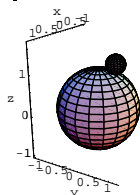
Add the plot of the sphere of radius 0.2 centered at  $\{1, 1, 1.5\}$  to this plot.

□ Answer:

That's easy. Just copy, paste, and edit the code that made the original plot:

```
center = {1, 1, 1.5};
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
center + {r Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
r = 0.2;
{{slow, shigh}, {tlow, thigh}} = {{0, π}, {0, 2 π}};
```

```
Clear[secondsphereplotter];
secondsphereplotter[s_, t_] = {x[r, s, t], y[r, s, t], z[r, s, t]};
secondsphereplot =
  ParametricPlot3D[Evaluate[secondsphereplotter[s, t]],
    {s, slow, shigh}, {t, tlow, thigh}, DisplayFunction -> Identity];
Show[firstsphereplot, secondsphereplot, Boxed -> False,
  BoxRatios -> Automatic, ViewPoint -> CMView, PlotRange -> All,
  AxesLabel -> {"x", "y", "z"}, DisplayFunction -> $DisplayFunction];
```



This plants the seeds for some interesting movies.

□T.1.b)

Use spherical coordinates to get a quick evaluation of

$$\iiint_{R_{xyz}} (x^2 + y^2) dx dy dz$$

where  $R_{xyz}$  is the sphere of radius 0.1 centered on {2, 5, 0}.

□Answer:

Go with spherical coordinates based at {2, 5, 0} and calculate the volume conversion factor:

```
center = {2, 5, 0};
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
  center + {r Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
{{rlow, rhigh}, {slow, shigh}, {tlow, thigh}} =
  {{0, 0.1}, {0, π}, {0, 2π}};
Clear[gradx, grady, gradz, Vxyz]
gradx[r_, s_, t_] =
  {D[x[r, s, t], r], D[x[r, s, t], s], D[x[r, s, t], t]};
grady[r_, s_, t_] =
  {D[y[r, s, t], r], D[y[r, s, t], s], D[y[r, s, t], t]};
gradz[r_, s_, t_] =
  {D[z[r, s, t], r], D[z[r, s, t], s], D[z[r, s, t], t]};
Vxyz[r_, s_, t_] =
  TrigExpand[Det[{gradx[r, s, t], grady[r, s, t], gradz[r, s, t]}]]
r^2 Sin[s]
```

Because you are going with  $0 \leq s \leq \pi$ , you know that  $r^2 \sin[s]$  can never go negative, so you don't have to multiply by  $-1$ .

Now you are ready to calculate

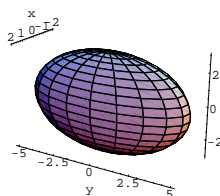
$$\begin{aligned} & \iiint_{R_{xyz}} (x^2 + y^2) dx dy dz \\ &= \int_{tlow}^{thigh} \int_{slow}^{shigh} \int_{rlow}^{rhigh} (x[r, s, t]^2 + y[r, s, t]^2) V_{xyz}[r, s, t] dr ds dt \\ &= \int_{tlow}^{thigh} \int_{slow}^{shigh} \int_{rlow}^{rhigh} (x[r, s, t]^2 + y[r, s, t]^2) V_{xyz}[r, s, t] dr ds dt \\ &= \int_{tlow}^{thigh} \int_{slow}^{shigh} \int_{rlow}^{rhigh} (x[r, s, t]^2 + y[r, s, t]^2) V_{xyz}[r, s, t] dr ds dt \\ &= 0.121492 + 0. \end{aligned}$$

Routine stuff.

□T.1.c.i)

Look at this:

```
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
  {2 r Sin[s] Cos[t], 5 r Sin[s] Sin[t], 3 r Cos[s]};
{{slow, shigh}, {tlow, thigh}} = {{0, π}, {0, 2π}};
Clear[plotter];
plotter[s_, t_] = {x[1, s, t], y[1, s, t], z[1, s, t]};
ParametricPlot3D[Evaluate[plotter[s, t]], {s, slow, shigh},
  {t, tlow, thigh}, Boxed -> False, BoxRatios -> Automatic,
  ViewPoint -> CMView, AxesLabel -> {"x", "y", "z"}];
```



What do folks call this skin?

□Answer:

It all depends on who you are talking to.

Common-sense folks call it an egg.

Fancy folks call it an ellipsoid.

□T.1.c.ii)

Agree that  $R_{xyz}$  is everything inside and on the skin plotted in part i) above. Calculate

$$\iiint_{R_{xyz}} (x^2 + y^2 + z^2) dx dy dz.$$

□Answer:

You plotted this skin by going with

$$\begin{aligned} & \{x[r, s, t], y[r, s, t], z[r, s, t]\} \\ &= \{2 r \sin[s] \cos[t], 5 r \sin[s] \sin[t], 3 r \cos[s]\} \end{aligned}$$

and keeping  $r = 1$ , running  $s$  from slow to shigh and running  $t$  from tlow to thigh.

Remembering that  $R_{xyz}$  is everything inside and on this skin, you see that you can describe  $R_{xyz}$  with

$$\begin{aligned} & \{x[r, s, t], y[r, s, t], z[r, s, t]\} \\ &= \{2 r \sin[s] \cos[t], 5 r \sin[s] \sin[t], 3 r \cos[s]\} \end{aligned}$$

by running  $r$  from 0 to 1, running  $s$  from slow to shigh, and running  $t$

from tlow to thigh.

In short,

$$\begin{aligned} & \iiint_{R_{xyz}} (x^2 + y^2 + z^2) dx dy dz \\ &= \iiint_{R_{int}} (x[r, s, t]^2 + y[r, s, t]^2 + z[r, s, t]^2) \\ & \quad V_{xyz}[r, s, t] dr ds dt \\ &= \int_{tlow}^{thigh} \int_{slow}^{shigh} \int_0^1 (x[r, s, t]^2 + y[r, s, t]^2 + z[r, s, t]^2) \\ & \quad V_{xyz}[r, s, t] dr ds dt \end{aligned}$$

Calculate the volume conversion factor:

```
Clear[gradx, grady, gradz, Vxyz]
gradx[r_, s_, t_] =
  {D[x[r, s, t], r], D[x[r, s, t], s], D[x[r, s, t], t]};
grady[r_, s_, t_] =
  {D[y[r, s, t], r], D[y[r, s, t], s], D[y[r, s, t], t]};
gradz[r_, s_, t_] = {D[z[r, s, t], r],
  D[z[r, s, t], s], D[z[r, s, t], t]};
Vxyz[r_, s_, t_] =
  TrigExpand[Det[{gradx[r, s, t], grady[r, s, t], gradz[r, s, t]}]]
30 r^2 Sin[s]
```

Here comes the calculation of

$$\begin{aligned} & \iiint_{R_{xyz}} (x^2 + y^2 + z^2) dx dy dz \\ &= \int_{tlow}^{thigh} \int_{slow}^{shigh} \int_0^1 (x[r, s, t]^2 + y[r, s, t]^2 + z[r, s, t]^2) \\ & \quad V_{xyz}[r, s, t] dr ds dt \end{aligned}$$

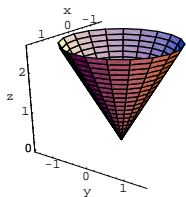
$$\int_{tlow}^{thigh} \int_{slow}^{shigh} \int_0^1 (x[r, s, t]^2 + y[r, s, t]^2 + z[r, s, t]^2) V_{xyz}[r, s, t] dr ds dt = 304 \pi$$

T.2) Ice cream cones and tops

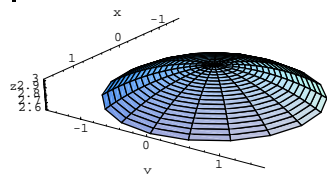
□T.2.a.i) Ice cream cones

Here's an ice cream cone plotted in two steps:

```
Clear[x, y, z, r, s, t, radial]
xx[r_, s_, t_] = r Sin[s] Cos[t];
yy[r_, s_, t_] = r Sin[s] Sin[t];
zz[r_, s_, t_] = r Cos[s];
Clear[sideplotter]
sideplotter[r_, t_] = {xx[r, Pi/6, t], yy[r, Pi/6, t], zz[r, Pi/6, t]};
side = ParametricPlot3D[Evaluate[sideplotter[r, t]],
  {r, 0, 3}, {t, 0, 2 Pi}, Boxed -> False, BoxRatios -> Automatic,
  ViewPoint -> CMView, AxesLabel -> {"x", "y", "z"}];
```

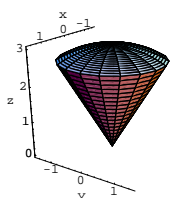


```
Clear[capplotter]
capplotter[s_, t_] = {xx[3, s, t], yy[3, s, t], zz[3, s, t]};
cap = ParametricPlot3D[Evaluate[capplotter[s, t]],
  {s, 0, Pi/6}, {t, 0, 2 Pi}, Boxed -> False, BoxRatios -> Automatic,
  ViewPoint -> CMView, AxesLabel -> {"x", "y", "z"}];
```



Here's the whole ice cream cone:

```
Show[cap, side, Boxed -> False, BoxRatios -> Automatic,
  ViewPoint -> CMView, AxesLabel -> {"x", "y", "z"}];
```



Stingy serving.

Imagine that the ice cream is packed solidly all the way to the bottom, and measure the volume of the ice cream.

□ Answer:

When you review the plotting instructions above, you find that you plotted the side by plotting

$$\{x[r, \frac{\pi}{6}, t], y[r, \frac{\pi}{6}, t], z[r, \frac{\pi}{6}, t]\}$$

with  $0 \leq r \leq 3$ , and  $0 \leq t \leq 2\pi$ .

You also find that you plotted the cap by plotting

$$\{x[3, s, t], y[3, s, t], z[3, s, t]\}$$

with  $0 \leq s \leq \frac{\pi}{6}$ , and  $0 \leq t \leq 2\pi$ .

When you go with

$$0 \leq r \leq 3, 0 \leq s \leq \frac{\pi}{6}, \text{ and } 0 \leq t \leq 2\pi,$$

the points

$$\{x[r, s, t], y[r, s, t], z[r, s, t]\}$$

will sweep out the skin of the top and everything inside.

Call the skin of the top and everything inside by the name  $R_{xyz}$ .

To measure the volume of  $R_{xyz}$ , all you gotta do is calculate

$$\text{volume} = \iiint_R dx dy dz$$

$$= \iiint_{R_{xyz}} V_{xyz}[r, s, t] dr ds dt$$

$$= \int_0^{2\pi} \int_0^{\frac{\pi}{6}} \int_0^3 V_{xyz}[r, s, t] dr ds dt$$

```
Clear[gradx, grady, gradz, Vxyz]
gradx[r_, s_, t_] =
  {D[x[r, s, t], r], D[x[r, s, t], s], D[x[r, s, t], t]};
grady[r_, s_, t_] =
  {D[y[r, s, t], r], D[y[r, s, t], s], D[y[r, s, t], t]};
gradz[r_, s_, t_] =
  {D[z[r, s, t], r], D[z[r, s, t], s], D[z[r, s, t], t]};
Vxyz[r_, s_, t_] =
  TrigExpand[Det[{gradx[r, s, t], grady[r, s, t], gradz[r, s, t]}]]
r^2 Sin[s]
```

Because  $0 \leq s \leq \frac{\pi}{6}$ , this can never be negative; so

$$\text{volume} = \iiint_R dx dy dz$$

$$= \iiint_{R_{xyz}} V_{xyz}[r, s, t] dr ds dt$$

$$= \int_0^{2\pi} \int_0^{\frac{\pi}{6}} \int_0^3 V_{xyz}[r, s, t] dr ds dt$$

measures out to:

$$\text{volume} = \int_0^{2\pi} \int_0^{\frac{\pi}{6}} \int_0^3 V_{xyz}[r, s, t] dr ds dt$$

$$18\pi - 9\sqrt{3}\pi$$

That measurement is a bit too esoteric; check it out in plain decimal cubic units:

```
N[volume]
7.57608
```

Very, very easy.

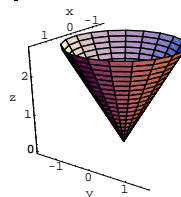
Once again, if you can plot something, you've got a really good chance of setting up the integral to measure its volume.

### □T.2.a.ii) Tops

You can use spherical coordinates to come up with your own custom-made coordinate systems.

Here's a sample involving a top. It's only a small variant of what you did in part a.i) above.

```
Clear[x, xx, y, yy, z, zz, r, s, t, radial]
xx[r_, s_, t_] = r Sin[s] Cos[t];
yy[r_, s_, t_] = r Sin[s] Sin[t];
zz[r_, s_, t_] = r Cos[s];
radial[s_] = 3 + E^-5s;
{xx[r_, s_, t_], yy[r_, s_, t_], zz[r_, s_, t_]} =
  {xx[r radial[s], s, t], yy[r radial[s], s, t], zz[r radial[s], s, t]};
Clear[sideplotter, radial]
radial[s_] = 3 + E^-5s;
sideplotter[r_, t_] = {xx[r, Pi/6, t], yy[r, Pi/6, t], zz[r, Pi/6, t]};
side = ParametricPlot3D[Evaluate[sideplotter[r, t]],
  {r, 0, 1}, {t, 0, 2 Pi}, Boxed -> False, BoxRatios -> Automatic,
  ViewPoint -> CMView, AxesLabel -> {"x", "y", "z"}];
```

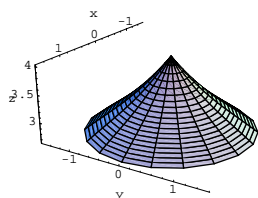


```
Clear[capplotter]
capplotter[s_, t_] = {xx[1, s, t], yy[1, s, t], zz[1, s, t]};
cap = ParametricPlot3D[Evaluate[capplotter[s, t]],
  {s, 0, Pi/6}, {t, 0, 2 Pi}, Boxed -> False, BoxRatios -> Automatic,
  ViewPoint -> CMView, AxesLabel -> {"x", "y", "z"}];
```

9.40702

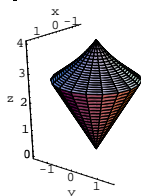
Easy.

Once again, if you can plot something, you've got a damn good chance of setting up the integral that measures its volume.



Here's the whole top:

```
Show[cap, side, Boxed → False, BoxRatios → Automatic,
ViewPoint → CMView, AxesLabel → {"x", "y", "z"}];
```



Spiffy.

Measure the volume inside this spinner.

□ Answer:

When you review the plotting instructions above, you find that you plotted the side by plotting

$$\{x[r, \frac{\pi}{6}, t], y[r, \frac{\pi}{6}, t], z[r, \frac{\pi}{6}, t]\}$$

with  $0 \leq r \leq 1$  and  $0 \leq t \leq 2\pi$ .

You also find that you plotted the cap by plotting

$$\{x[1, s, t], y[1, s, t], z[1, s, t]\}$$

with  $0 \leq s \leq \frac{\pi}{6}$  and  $0 \leq t \leq 2\pi$ .

When you go with

$$0 \leq r \leq 1, 0 \leq s \leq \frac{\pi}{6} \text{ and } 0 \leq t \leq 2\pi,$$

the points

$$\{x[r, s, t], y[r, s, t], z[r, s, t]\}$$

sweep out the skin of the top and everything inside.

Call the skin of the top and everything inside  $R_{xyz}$ .

To measure the volume of  $R_{xyz}$ , all you have to do is calculate

$$\begin{aligned} \text{volume} &= \iiint_R dx dy dz \\ &= \iiint_{R_{xyz}} V_{xyz}[r, s, t] dr ds dt \\ &= \int_0^{2\pi} \int_0^{\frac{\pi}{6}} \int_0^1 V_{xyz}[r, s, t] dr ds dt \end{aligned}$$

```
Clear[gradx, grady, gradz, Vxyz]
gradx[r_, s_, t_] =
{D[x[r, s, t], r], D[x[r, s, t], s], D[x[r, s, t], t]};
grady[r_, s_, t_] =
{D[y[r, s, t], r], D[y[r, s, t], s], D[y[r, s, t], t]};
gradz[r_, s_, t_] =
{D[z[r, s, t], r], D[z[r, s, t], s], D[z[r, s, t], t]};
Vxyz[r_, s_, t_] =
TrigExpand[Det[{gradx[r, s, t], grady[r, s, t], gradz[r, s, t]}]]
27 r^2 Sin[s] + E^-15 s r^2 Sin[s] + 9 E^-10 s r^2 Sin[s] + 27 E^-5 s r^2 Sin[s]
```

Because  $0 \leq s \leq \frac{\pi}{6}$ , this can never be negative, so

$$\begin{aligned} \text{volume} &= \iiint_R dx dy dz \\ &= \iiint_{R_{xyz}} V_{xyz}[r, s, t] dr ds dt \\ &= \int_0^{2\pi} \int_0^{\frac{\pi}{6}} \int_0^1 V_{xyz}[r, s, t] dr ds dt \end{aligned}$$

measures out to:

```
volume = \int_0^{2\pi} \int_0^{\frac{\pi}{6}} \int_0^1 V_{xyz}[r, s, t] dr ds dt
2 \left( \frac{4173916}{445107} - \frac{9\sqrt{3}}{2} - \frac{5}{452} E^{-5\pi/2} - \frac{E^{-5\pi/2}}{452\sqrt{3}} - \frac{15}{101} E^{-5\pi/3} - \frac{3}{202} \sqrt{3} E^{-5\pi/3} - \frac{45}{52} E^{-5\pi/6} - \frac{9}{52} \sqrt{3} E^{-5\pi/6} \right) \pi
```

That measurement is way too abstruse; check it out in plain decimal cubic units:

```
N[volume]
```

### T.3) Jennifer Phillips's flower

□ T.3.a)

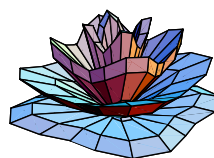
When she was asked to do some art with spherical coordinates, 1992 Calculus&Mathematica student Jennifer Phillips responded with the flower you are about to see. She commented, "I only did one, but I like it a lot."

So will you.

Activate, enjoy, read her code, and think about what you too can do with spherical coordinates and tubes.

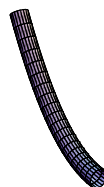
□ The bloom:

```
Clear[x, y, z, r, s, t]
x[r_, s_, t_] = r Sin[s] Cos[t];
y[r_, s_, t_] = r Sin[s] Sin[t];
z[r_, s_, t_] = r Cos[s];
r[s_, t_] = 1.5 (1 - Sin[16 s]) + 0.5 (1 - Cos[4 t]);
bloom = ParametricPlot3D[{x[r[s, t], s, t], y[r[s, t], s, t],
z[r[s, t], s, t]}, {s, 0, \frac{\pi}{2}}, {t, 0, 2\pi}, Axes → None,
ViewPoint → CMView, Boxed → False, PlotRange → All];
```



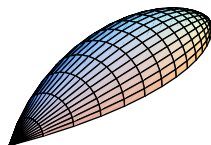
□ The stem:

```
Clear[s, P]
P[s_] = {-s, -2 s, s^2};
Clear[
unittan, mainnormal, mainunitnormal, binormal, secondunitnormal]
unittan[s_] = \frac{P'[s]}{\sqrt{Expand[P'[s] . P'[s]}}];
mainnormal[s_] = Together[D[unittan[s], s]];
mainunitnormal[s_] = Simplify[\frac{mainnormal[s]}{\sqrt{mainnormal[s] . mainnormal[s]}}];
binormal[s_] = Simplify[unittan[s] x mainunitnormal[s]];
r = 0.3;
Clear[stem, t]
stem[s_, t_] = Together[
P[s] + r Cos[t] mainunitnormal[s] + r Sin[t] binormal[s]] + {2, 4, -4};
stemplot = ParametricPlot3D[stem[s, t], {s, 0, 2}, {t, 0, 2\pi},
Axes → None, ViewPoint → CMView, Boxed → False, PlotRange → All];
```



□ A leaf:

```
Clear[x, y, z, r, s, t]
x[r_, s_, t_] = r Sin[s] Cos[t] + 2;
y[r_, s_, t_] = r Sin[s] Sin[t] - 2;
z[r_, s_, t_] = r Cos[s] + 2;
r[s_] = 3.3 (1 - Sin[2 s]);
leaf = ParametricPlot3D[{z[r[s], s, t], x[r[s], s, t], y[r[s], s, t]},
{0, \frac{\pi}{4}}, {0, \pi}, Axes → None, ViewPoint → CMView,
Boxed → False, PlotRange → All];
```



□All together:

```
Show[bloom, stemplot, leaf, Axes → None, ViewPoint → CMView,
      Boxed → False, PlotRange → All];
```



Exquisite.  
Way to go, Jennifer!

**VC.09 Spherical Coordinates  
Give Them a Try!**

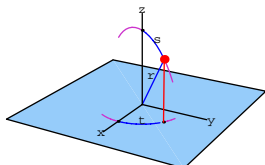
Experience with the starred problems will be useful for understanding developments later in the course.

**G.1) Plotting with spherical coordinates\***

□G.1.a)

Here is the physical meaning of the spherical coordinates  $r$ ,  $s$ , and  $t$ :

```
Show[all, DisplayFunction → $DisplayFunction];
```

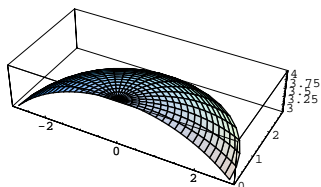


Here are the formulas for spherical coordinates:

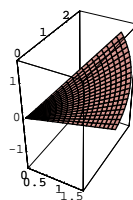
```
Clear[x, y, z, r, s, t]
x[r_, s_, t_] = r Sin[s] Cos[t]
y[r_, s_, t_] = r Sin[s] Sin[t]
z[r_, s_, t_] = r Cos[s]
r Cos[t] Sin[s]
r Sin[s] Sin[t]
r Cos[s]
```

Account for why the following plots turn out the way they do:

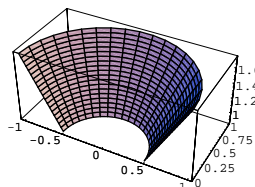
```
ParametricPlot3D[{x[4, s, t], y[4, s, t], z[4, s, t]}, {s, 0, π/4},
                 {t, 0, π}, Axes → Automatic];
```



```
ParametricPlot3D[{x[x, s, π/3], y[x, s, π/3], z[x, s, π/3]},
                 {s, π/3, 2π/3}, {r, 0, 3}, Axes → Automatic];
```



```
ParametricPlot3D[{x[r, π/6, t], y[r, π/6, t], z[r, π/6, t]},
                 {r, 1, 2}, {t, 0, π}, Axes → Automatic];
```



□G.1.b.i)

Describe the part of the sphere

$$x^2 + y^2 + z^2 = 4$$

consisting of those points with  $y \geq 0$  in spherical coordinates.

□G.1.b.ii)

Describe the cone

$$z = \sqrt{x^2 + y^2}$$

in spherical coordinates.

□Tip:

You get this cone by rotating the line  $z = y$  (with  $y \geq 0$ ) in the  $yz$ -plane about the  $z$ -axis.

**G.2) Integrating\***

□G.2.a)

A sphere of radius  $\text{rad}$  contains a volume of

$$\frac{4\pi \text{rad}^3}{3}$$

cubic units.

Let  $R$  be the region inside and on the sphere

$$x^2 + y^2 + z^2 = \text{rad}^2,$$

and use spherical coordinates to measure

$$\text{volume of } R = \iiint_R dx dy dz$$

to give a quick explanation of how this formula comes to be.

□G.2.b)

Calculate

$$\iiint_{R_{xyz}} (x^2 + y^2 + z^2) dx dy dz$$

where  $R_{xyz}$  is the region consisting of everything between and on the spheres

$$x^2 + y^2 + z^2 = a^2 \text{ and } x^2 + y^2 + z^2 = b^2.$$

Assume  $0 < a < b$ .

□Tip:

You can save a lot of time if you go to spherical coordinates with  $a \leq r \leq b$ .

□G.2.c)

A spherical shell whose inner radius is 2 cm and whose outer radius is 8 cm is made of material of varying density.

At a point  $\{x, y, z\}$ , the density of the material is 0.39 times the square of the distance to the center of the sphere.

Assume all the air has been pumped out of the hollow interior.

Does the shell sink or float in water?

## □G.2.d)

Measure the volume of the solid region that is inside the sphere

$$x^2 + y^2 + z^2 = 4,$$

but is outside the cylinder

$$x^2 + y^2 = 1.$$

□Tip:

Cylindrical coordinates

$$x = r \cos[t],$$

$$y = r \sin[t], \text{ and}$$

$$z = s$$

will work better than spherical coordinates.

## □G.2.e)

Calculate

$$\int \int \int_{R_{xyz}} \left( \left( \frac{x}{a} \right)^2 + \left( \frac{y}{b} \right)^2 + \left( \frac{z}{c} \right)^2 \right) dx dy dz,$$

where  $R_{xyz}$  is the region consisting of everything inside and on the ellipsoid

$$\left( \frac{x}{a} \right)^2 + \left( \frac{y}{b} \right)^2 + \left( \frac{z}{c} \right)^2 = 1.$$

## G.3) Ice cream cones

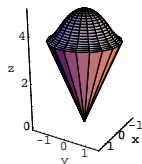
## □G.3.a)

Here's sugar waffle ice cream cone plotted with a serving of ice cream:

```
Clear[x, y, z, r, s, t]
x[r_, s_, t_] = r Sin[s] Cos[t];
y[r_, s_, t_] = r Sin[s] Sin[t];
z[r_, s_, t_] = r Cos[s];
Clear[coneplotter]
coneplotter[r_, t_] = {x[r, π/8, t], y[r, π/8, t], z[r, π/8, t]};
cone = ParametricPlot3D[Evaluate[coneplotter[r, t]], {r, 0, 4},
  {t, 0, 2 π}, PlotPoints → {2, Automatic}, DisplayFunction → Identity];
Clear[x, xx, y, yy, z, zz, r, s, t, radial, caplotter]
xx[r_, s_, t_] = r Sin[s] Cos[t];
```

```
yy[r_, s_, t_] = r Sin[s] Sin[t];
zz[r_, s_, t_] = r Cos[s];
radial[s_] = 4 + Cos[4 s]^2;
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
  {xx[r radial[s], s, t], yy[r radial[s], s, t], zz[r radial[s], s, t]};
caplotter[s_, t_] = {x[1, s, t], y[1, s, t], z[1, s, t]};
cap = ParametricPlot3D[Evaluate[caplotter[s, t]],
  {s, 0, π/8}, {t, 0, 2 π}, DisplayFunction → Identity];

Show[cone, cap, Boxed → False, BoxRatios → Automatic,
  ViewPoint → CMView, PlotRange → All, AxesLabel → {"x", "y", "z"},
  DisplayFunction → $DisplayFunction];
```



Tsk-tsk.

Any ice cream stand that tries to peddle ice cream cones like this will not stay in business very long.

Use the same sugar waffle base, but this time plot the ice cream the way you would like to see it, and measure the total volume of the ice cream in your fully stuffed cone.

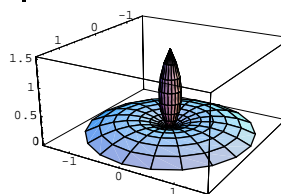
## G.4) Skins

## □G.4.a)

Here's the skin of a mutant solid:

```
Clear[x, xx, y, yy, z, zz, r, radial, s, t]
{xx[r_, s_, t_], yy[r_, s_, t_], zz[r_, s_, t_]} =
  {r Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
radial[s_] = 1.5 (1 - Sin[2 s]);
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
  {xx[r radial[s], s, t], yy[r radial[s], s, t], zz[r radial[s], s, t]};
```

```
mutant =
  ParametricPlot3D[{x[1, s, t], y[1, s, t], z[1, s, t]}, {s, 0, π/2},
  {t, 0, 2 π}, Axes → Automatic, ViewPoint → CMView, PlotRange → All];
```



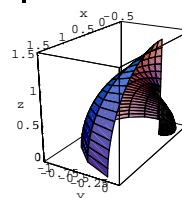
Measure the volume of the whole mutant solid.

Then measure the volume of the base and the centerpiece, and check your first measurement by adding the second and third measurements.

## □G.4.b)

Here is the skin of the shell of a young snail:

```
Clear[x, y, z, radial, s, t]
x[r_, s_, t_] = r Sin[s] Cos[t];
y[r_, s_, t_] = r Sin[s] Sin[t];
z[r_, s_, t_] = r Cos[s];
radial[t_] = 1.5 (1 - Cos[t/4]);
ParametricPlot3D[
  {x[radial[t], s, t], y[radial[t], s, t], z[radial[t], s, t]},
  {s, 0, π/2}, {t, 0, 2 π}, AxesLabel → {"x", "y", "z"},
  ViewPoint → CMView, PlotRange → All];
```

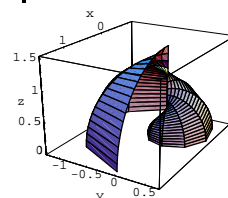


Measure the volume under this shell and over the xy-floor.

## □G.4.c)

Here's the skin of another snail shell:

```
Clear[x, y, z, r, s, t]
x[r_, s_, t_] = r Sin[s] Cos[t];
y[r_, s_, t_] = r Sin[s] Sin[t];
z[r_, s_, t_] = r Cos[s];
r[t_] = 1.5 (1 - Cos[t/8]);
ParametricPlot3D[
  {x[r[t], s, t], y[r[t], s, t], z[r[t], s, t]}, {s, 0, π/2}, {t, 0, 4 π},
  AxesLabel → {"x", "y", "z"}, ViewPoint → CMView, PlotRange → All];
```



Measure the volume under this shell and over the xy-floor.

□Tip:

Watch out, because as  $t$  runs from  $0$  to  $4\pi$ , this shell wraps around twice.

If you integrate with respect to  $t$  from  $t = 0$  to  $t = 4\pi$ , your answer will be off.



### G.5) Floaters

□G.5.a)

All measurements here are in centimeters.

A solid floater of radius  $r$  is to be made in the shape of the sphere

$$x^2 + y^2 + z^2 = r^2.$$

The density  $[x, y, z]$  of the floater is to be proportional to the distance of the point  $\{x, y, z\}$  from  $\{0, 0, 0\}$ ; this means

$$\text{density}[x, y, z] = k \sqrt{x^2 + y^2 + z^2}$$

Give a formula for the function  $k[r]$  such that this beast of radius  $r$  will float in water if  $0 < k < k[r]$  and will sink if  $k > k[r]$ .

Plot  $k[r]$  as a function of  $r$ .

Does  $k[r]$  have a limiting value  $\lim_{r \rightarrow \infty} k[r]$ ?

□G.5.b)

A cubic centimeter of platinum weighs 21.4 grams. A cubic centimeter of ethyl alcohol weighs 0.81 grams.

Your job is to build a hollow platinum spherical shell 100 centimeters in diameter with the outer wall  $h$  centimeters thick so that the hollow spherical shell will float in the ethyl alcohol. How small does  $h$  have to be to pull this off?

□Tip:

This problem is a ringer, once you remember that the volume of a sphere of radius  $r$  is  $\frac{4\pi r^3}{3}$  and that the floating spherical shell will displace the volume of water that weighs as much as the hollow sphere.

Neglect the weight of the air inside the spherical shell.

### G.6) Earth and moon

□G.6.a.i)

The radius of the earth is approximately 6400 km.

The radius of the moon is approximately 1750 km.

The moon rotates about the earth in an approximately circular orbit and the distance from the center of the earth to the center of the moon is approximately 384,000 km.

Make a true scale plot showing the earth and moon in position as appropriately sized spheres.

□Tip:

If you put the center of the earth at  $\{0, 0, 0\}$ , then you can take any unit vector, like

$$U = \frac{\{-1, 1, 1\}}{\sqrt{3}},$$

and put the center of the moon at 384,000  $U$ .

You are not obligated to use this choice of unit vector, put the moon where you think it looks best, but keep it in true scale.

□G.6.a.ii)

Some scientists believe that the moon formed as a result of a huge collision between the earth and a large interplanetary body.

See William K. Hartman's book, *Astronomy: The Cosmic Journey*, Wadsworth, Belmont, California, 1991, p 163.

Many of these same scientists believe that at some time long, long ago, the moon was in circular orbit about the earth, but the distance between the center of the earth and the center of the moon was only about 70,000 km.

Make a true scale plot showing the earth and moon in this position as appropriately sized spheres.

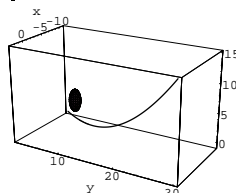
### G.7) Star Wars

Calculus&Mathematica thanks space engineer Lynn Purser of Teledyne-Brown Engineering, Huntsville, Alabama, for suggesting a problem like this in his talk at the First Annual Mathematica Conference in 1990.

Here's a curve and an ellipsoid:

```
Clear[h, P]
P[h_] = {h - 12, 2 h, 0.1 h (h - 10) + h/2};
{hlow, hhigh} = {1, 15};
curveplot =
  ParametricPlot3D[P[h], {h, hlow, hhigh}, DisplayFunction -> Identity];
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
  {r Sin[s] Cos[t], r Sin[s] Sin[t], 2 r Cos[s]};
{{slow, shigh}, {tlow, thigh}} = {{0, Pi}, {0, 2 Pi}};
Clear[plotter];
plotter[s_, t_] = {-9, 5, 3} + {x[1, s, t], y[1, s, t], z[1, s, t]};
ellipseplot = ParametricPlot3D[Evaluate[plotter[s, t]],
  {s, slow, shigh}, {t, tlow, thigh}, DisplayFunction -> Identity];

Show[ellipseplot, curveplot, BoxRatios -> Automatic,
  ViewPoint -> CMView, PlotRange -> All, AxesLabel -> {"x", "y", "z"},
  DisplayFunction -> $DisplayFunction];
```



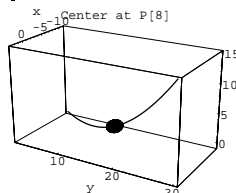
Here's how you can pick up the ellipsoid and move it so that the center of the ellipsoid is at  $P[h]$  on the curve, and aligned so that the long axis of the ellipsoid shares ink with the unit tangent vector to the curve at the point  $P[h]$ .

Here's a plot in the case  $h = 8$ :

```
Clear[unittan, mainnormal,
  mainunitnormal, binormal, secondunitnormal]
```

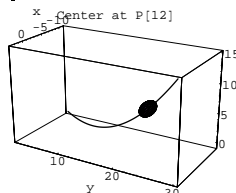
```
unittan[h_] = P'[h] /
  Sqrt[Expand[P'[h] . P'[h]]];
mainnormal[h_] = Together[D[unittan[h], h]];
mainunitnormal[h_] = Simplify[
  mainnormal[h] / Sqrt[mainnormal[h] . mainnormal[h]]];
binormal[h_] = Chop[Expand[Cross[unittan[h], mainunitnormal[h]]]];
Clear[egg]
egg[h_, r_, t_] = P[h] + N[x[1, s, t] mainunitnormal[h] +
  y[1, s, t] binormal[h] + z[1, s, t] unittan[h]];
eggplot = ParametricPlot3D[Evaluate[egg[8, r, t]], {s, slow, shigh},
  {t, tlow, thigh}, ViewPoint -> CMView, DisplayFunction -> Identity];

Show[eggplot, curveplot, BoxRatios -> Automatic,
  ViewPoint -> CMView, PlotRange -> All, AxesLabel -> {"x", "y", "z"},
  PlotLabel -> "Center at P[8]", DisplayFunction -> $DisplayFunction];
```



$h = 12$ :

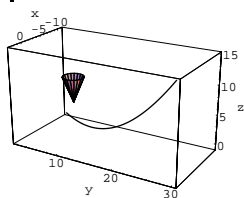
```
eggplot = ParametricPlot3D[Evaluate[egg[12, r, t]], {s, slow, shigh},
  {t, tlow, thigh}, ViewPoint -> CMView, DisplayFunction -> Identity];
Show[eggplot, curveplot, BoxRatios -> Automatic,
  ViewPoint -> CMView, PlotRange -> All, AxesLabel -> {"x", "y", "z"},
  PlotLabel -> "Center at P[12]", DisplayFunction -> $DisplayFunction];
```



Play with other values of  $h$ .

Now stay with the same curve, but look at this cone:

```
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
{r Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
{rlow, rhigh}, {tlow, thigh}} = {{0, 5}, {0, 2 π}};
Clear[plotter];
plotter[s_, t_] = {-9, 5, 3} + {x[r, π/8, t], y[r, π/8, t], z[r, π/8, t]};
coneplot = ParametricPlot3D[
  Evaluate[plotter[s, t]], {r, rlow, rhigh}, {t, tlow, thigh},
  PlotPoints → {2, Automatic}, DisplayFunction → Identity];
Show[coneplot, curveplot, BoxRatios → Automatic,
  ViewPoint → CMView, PlotRange → All, AxesLabel → {"x", "y", "z"},
  DisplayFunction → $DisplayFunction];
```



At the tip of this cone is a phaser (zapper). The cone itself represents the range of the phaser in the sense that any object that finds itself in the cone will be immediately disintegrated by the phaser. Any object outside the cone is immune to attack by the phaser.

□G.7.a.i)

Your mission is to take the tip of the cone, and move it to a point P[h] on the curve so that the central axis of the cone is aligned with the unit tangent to the curve at P[h], and directed so that the open end points in the direction of the parameterization of the curve. Plot in the cases h = 4, 9, and 15.

□G.7.a.ii)

Stay with the same curve as in part i) above. A hostile space station is anchored at the point {-3.2, 17.6, 3.5}. If you move your phaser as above along the given curve, will your phaser be able to zap this space station? Back up your answer with a decisive plot showing the cone eating the point or showing that the cone cannot eat the point.

□G.7.a.iii)

Stay with the same curve as in parts i) and ii) above. When you think about it, you see that there are two ways for an object to become vulnerable to the phaser. The most obvious way is for the object to find itself at a point sitting in the opening at the wide end of the cone. Give a plot of the skin of the solid consisting of all points that are vulnerable this way.

□G.7.a.iv)

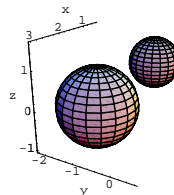
For some curves, there is another way for an object to become vulnerable to the phaser. Say what this way is and discuss the problems you might encounter in plotting these additional vulnerable points.

**G.8) Planes between spheres**

Here are two spheres:

```
center1 = {2, -1, 0};
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
center1 + {r Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
r1 = 1;
{{slow, shigh}, {tlow, thigh}} = {{0, π}, {0, 2 π}};
Clear[sphere1plotter];
sphere1plotter[s_, t_] = {x[r1, s, t], y[r1, s, t], z[r1, s, t]};
sphere1plot = ParametricPlot3D[Evaluate[sphere1plotter[s, t]],
  {s, slow, shigh}, {t, tlow, thigh}, DisplayFunction → Identity];
center2 = {1, 0, 1};
Clear[x, y, z, r, s, t]
{x[r_, s_, t_], y[r_, s_, t_], z[r_, s_, t_]} =
```

```
center2 + {r Sin[s] Cos[t], r Sin[s] Sin[t], r Cos[s]};
r2 = 0.6;
{{slow, shigh}, {tlow, thigh}} = {{0, π}, {0, 2 π}};
Clear[sphere2plotter];
sphere2plotter[s_, t_] = {x[r2, s, t], y[r2, s, t], z[r2, s, t]};
sphere2plot = ParametricPlot3D[Evaluate[sphere2plotter[s, t]],
  {s, slow, shigh}, {t, tlow, thigh}, DisplayFunction → Identity];
Show[sphere1plot, sphere2plot, Boxed → False, BoxRatios → Automatic,
  ViewPoint → CMView, PlotRange → All, AxesLabel → {"x", "y", "z"},
  DisplayFunction → $DisplayFunction];
```



□G.8.a)

As you can see, there is not a lot of room between these spheres, but there is room enough to insert a plane between them so that the two spheres are on different sides of the plane.

Insert such a plane into this plot.

□Tip:

Think about this vector:

```
center2 - center1
{-1, 1, 1}
```

**G.9) Art**

□G.9.a)

Use spherical coordinates, tubes, and anything else you want to mix in to make a gallery of eye-catching surfaces of your own design. Show off.

**G.10) Four dimensions: Is it just science fiction?**

□G.10.a.i)

The equation  $x^2 + y^2 + z^2 + t^2 = 3^2$  defines a (hyper)sphere of radius 3 in four dimensions. If you set  $t = 2$ , you get a cross-section of the four dimensional sphere. Identify this cross section and plot it.

□Tip:

Just the cross section of 3D spheres are circles, the cross sections of 4D spheres are 3D spheres.

□G.10.a.ii)

The equation  $x^2 + y^2 + z^2 + t^2 = \text{rad}^2$  defines a (hyper)sphere of radius rad in four dimensions. Use a system of double polar coordinates and the appropriate volume conversion factor to give a formula that measures the four dimensional volume of this (hyper)sphere of radius rad.

□Tip:

Agreeing that R stands for everything inside and on the four dimensional sphere of radius rad, you will also agree that its volume is

$$\iiint_{\mathbb{R}} dx dy dz dt.$$

Double polar coordinates mean:

```
Clear[x, y, z, t, r, s, u, v, rad]
{x[r_, s_, u_, v_]} = r Cos[s];
y[r_, s_, u_, v_] = r Sin[s];
z[r_, s_, u_, v_] = u Cos[v];
t[r_, s_, u_, v_] = u Sin[v];
```

You let  $s$  and  $v$  run from 0 to  $2\pi$ , and to see what  $r$  and  $u$  do, substitute into

$$x^2 + y^2 + z^2 + t^2 = \text{rad}^2:$$

```
TrigExpand[
  x[r, s, u, v]^2 + y[r, s, u, v]^2 + z[r, s, u, v]^2 + t[r, s, u, v]^2 ==
  rad^2
  r^2 + u^2 == rad^2
```

This tells you to put

$$r[p, q] = p \text{Cos}[q], \text{ and}$$

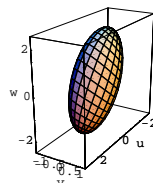
$$u[p, q] = p \text{Sin}[q],$$

to let  $p$  run from 0 to  $\text{rad}$ , and to let  $q$  run from 0 to  $\pi/2$ .

Watch to be sure that your volume conversion factor stays positive.

This hypersphere does not have 4D volume 0.

```
Clear[r, s, t]
x[r_, s_, t_] = r Sin[s] Cos[t];
y[r_, s_, t_] = r Sin[s] Sin[t];
z[r_, s_, t_] = r Cos[s];
r = 1;
{{slow, shigh}, {tlow, thigh}} = {{0, \pi}, {0, 2 \pi}};
uvwplot =
ParametricPlot3D[Evaluate[{u[x[r, s, t], y[r, s, t], z[r, s, t]],
  v[x[r, s, t], y[r, s, t], z[r, s, t]],
  w[x[r, s, t], y[r, s, t], z[r, s, t]]}],
  {s, slow, shigh}, {t, tlow, thigh}, BoxRatios -> Automatic,
  ViewPoint -> CMView, AxesLabel -> {"u", "v", "w"}];
```



You guessed right!

This is a tilted ellipsoid centered at  $\{0, 0, 0\}$ .

Now look at:

```
eigens = Eigensystem[N[matrix]]
{{3.03316, -2.3813, 0.848139}, {0.920753, 0.271721, 0.279968},
{-0.305077, 0.0541348, 0.950788}, {-0.243193, 0.960852, -0.13274}}
```

The eigenvalues of the matrix are:

```
{eigenvalue1, eigenvalue2, eigenvalue3} = eigens[[1]]
{3.03316, -2.3813, 0.848139}
```

The eigenvectors of the matrix are:

```
{eigenvector1, eigenvector2, eigenvector3} = eigens[[2]]
{{0.920753, 0.271721, 0.279968}, {-0.305077, 0.0541348, 0.950788},
{-0.243193, 0.960852, -0.13274}}
```

The unit eigenvectors of the matrix are:

$$\left\{ \frac{\text{eigenvector1}}{\sqrt{\text{eigenvector1} \cdot \text{eigenvector1}}}, \frac{\text{eigenvector2}}{\sqrt{\text{eigenvector2} \cdot \text{eigenvector2}}}, \frac{\text{eigenvector3}}{\sqrt{\text{eigenvector3} \cdot \text{eigenvector3}}} \right\}$$

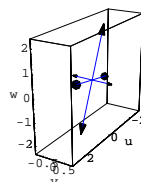
```
{0.920753, 0.271721, 0.279968}, {-0.305077, 0.0541348, 0.950788},
{-0.243193, 0.960852, -0.13274}}
```

Now put:

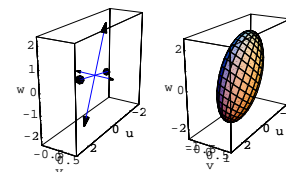
```
{amazing1, amazing2, amazing3} = {eigenvalue1 uniteigen1,
  eigenvalue2 uniteigen2, eigenvalue3 uniteigen3}
{{2.79279, 0.824172, 0.849189}, {0.726479, -0.128911, -2.26411},
{-0.206261, 0.814937, -0.112582}}
```

And look at this plot:

```
eigenplot = Show[Arrow[amazing1],
  Arrow[amazing2], Arrow[amazing3], Arrow[-amazing1],
  Arrow[-amazing2], Arrow[-amazing3], BoxRatios -> Automatic,
  Axes -> True, ViewPoint -> CMView, AxesLabel -> {"u", "v", "w"}];
```



```
Show[GraphicsArray[{eigenplot, uvwplot}]];
```



Feast your eyes on the ellipsoid built from the amazing vectors shown beside the uvw-plot of the original xyz-sphere  $x^2 + y^2 + z^2 = 1$ :

```
eigenellipsoid = ParametricPlot3D[Evaluate[x[r, s, t] amazing1 +
  y[r, s, t] amazing2 + z[r, s, t] amazing3], {s, slow, shigh},
  {t, tlow, thigh}, BoxRatios -> Automatic, ViewPoint -> CMView,
  AxesLabel -> {"u", "v", "w"}, DisplayFunction -> Identity];
Show[GraphicsArray[{eigenellipsoid, eigenplot, uvwplot}],
  DisplayFunction -> $DisplayFunction];
```

□G.10.b)

Conjure up the equation of a (hyper)sphere of radius  $r$  in five dimensions.

What is the five dimensional volume of this sphere?

□Tip:

Try using spherical coordinates on three dimensions and polar coordinates on the remaining two dimensions.

G.11) Eigenvalues and eigenvectors

One good way to extend your mathematical horizons is to look up some unfamiliar Mathematica instructions and try them out. This is what you will do here with a little coaching from your friends at Calculus&Mathematica.

A couple of lessons back, you might have looked at eigenvectors and eigenvalues in 2D. Now is your chance to see them in 3D.

Take nine numbers

$ax, ay, az, bx, by, bz, cx, cy,$  and  $cz$

with  $bx = ay, cy = bz,$  and  $cx = az,$  and make a matrix:

```
Clear[ax, ay, az, bx, by, bz, cx, cy, cz]
bx = ay; cy = bz; cx = az;
matrix = {{ax, ay, az}, {bx, by, bz}, {cx, cy, cz}};
MatrixForm[matrix]

$$\begin{pmatrix} ax & ay & az \\ ay & by & bz \\ az & bz & cz \end{pmatrix}$$

```

Note the pattern of the entries not on the diagonal.

Now do the same thing with specific numbers:

```
{ax, ay, az, by, bz, cz} = {2.4, 0.6, 1.5, 1, 0, -1.9};
{{ax, ay, az}, {bx, by, bz}, {cx, cy, cz}} =
  {{ax, ay, az}, {ay, by, bz}, {az, bz, cz}};
matrix = {{ax, ay, az}, {bx, by, bz}, {cx, cy, cz}};
MatrixForm[matrix]

$$\begin{pmatrix} 2.4 & 0.6 & 1.5 \\ 0.6 & 1 & 0 \\ 1.5 & 0 & -1.9 \end{pmatrix}$$

```

Note the pattern.

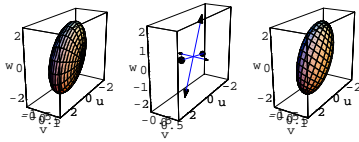
Go from xyz-space to uvw-space with:

```
Clear[u, v, w, x, y, z]
{u[x_, y_, z_], v[x_, y_, z_], w[x_, y_, z_]} =
  {ax x + ay y + az z, bx x + by y + bz z, cx x + cy y + cz z}
{2.4 x + 0.6 y + 1.5 z, 0.6 x + y, 1.5 x - 1.9 z}
```

Here is how the xyz-paper sphere

$$x^2 + y^2 + z^2 = 1$$

plots out in uvw-space in true scale:



Just for good measure, look at:

```
Clear[gradu, gradv, gradw, Vuvw]
gradu[x_, y_, z_] =
{D[u[x, y, z], x], D[u[x, y, z], y], D[u[x, y, z], z]};
gradv[x_, y_, z_] =
{D[v[x, y, z], x], D[v[x, y, z], y], D[v[x, y, z], z]};
gradw[x_, y_, z_] =
{D[w[x, y, z], x], D[w[x, y, z], y], D[w[x, y, z], z]};
Vuvw[x_, y_, z_] = Det[{gradu[x, y, z], gradv[x, y, z], gradw[x, y, z]}];
{eigenvalue1 eigenvalue2 eigenvalue3, Vuvw[x, y, z]}
{-6.126, -6.126}
```

Zingo.

The xyz-to-uvw volume conversion factor is the same as the product of the three eigenvalues.

□G.11.a)

Remembering that the ellipsoid above is the uvw-space plot of the xyz-space sphere circle  $x^2 + y^2 + z^2 = 1$ , write down the uvw-space measurement of the volume enclosed by the ellipsoid plotted above.

□Tip:

The volume inside the xyz-space sphere

$$x^2 + y^2 + z^2 = 1$$

measures out to  $\frac{4\pi}{3}$  cubic units.

□G.11.b)

Take nine new numbers

ax, ay, az, bx, by, bz, cx, cy, and cz,

with bx = ay, cy = bz, and cx = az, and try it again:

```
{ax, ay, az, by, bz, cz} = {2.1, 1.0, 1.0, -4.2, 3.1, 1.2};
{{ax, ay, az}, {bx, by, bz}, {cx, cy, cz}} =
```

```
{ax, ay, az}, {ay, by, bz}, {az, bz, cz};
matrix = {ax, ay, az}, {bx, by, bz}, {cx, cy, cz};
MatrixForm[matrix]

$$\begin{pmatrix} 2.1 & 1. & 1. \\ 1. & -4.2 & 3.1 \\ 1. & 3.1 & 1.2 \end{pmatrix}$$

```

Go from xyz-space to uvw-space with:

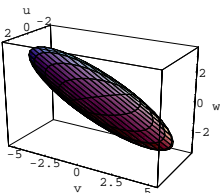
```
Clear[u, v, w, x, y, z]
{u[x_, y_, z_], v[x_, y_, z_], w[x_, y_, z_]} =
{ax x + ay y + az z, bx x + by y + bz z, cx x + cy y + cz z}
{2.1 x + 1. y + 1. z, 1. x - 4.2 y + 3.1 z, 1. x + 3.1 y + 1.2 z}
```

Here is how the xyz-paper sphere

$$x^2 + y^2 + z^2 = 1$$

plots out in uvw-space in true scale:

```
Clear[r, s, t]
x[r_, s_, t_] = r Sin[s] Cos[t];
y[r_, s_, t_] = r Sin[s] Sin[t];
z[r_, s_, t_] = r Cos[s];
r = 1;
{{slow, shigh}, {tlow, thigh}} = {{0, π}, {0, 2π}};
uvwplot =
ParametricPlot3D[Evaluate[{u[x[r, s, t], y[r, s, t], z[r, s, t]],
v[x[r, s, t], y[r, s, t], z[r, s, t]],
w[x[r, s, t], y[r, s, t], z[r, s, t]]}],
{s, slow, shigh}, {t, tlow, thigh}, BoxRatios -> Automatic,
ViewPoint -> CMView, AxesLabel -> {"u", "v", "w"}];
```



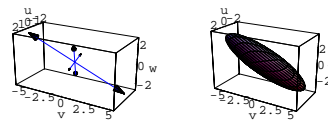
You guessed right again!

Another tilted ellipsoid centered at {0, 0, 0}.

Now look at:

```
eigens = Eigensystem[N[matrix]];
{eigenvalue1, eigenvalue2, eigenvalue3} = eigens[[1]];
{eigenvector1, eigenvector2, eigenvector3} = eigens[[2]];
```

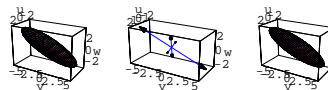
```
{uniteigen1, uniteigen2, uniteigen3} = {
eigenvector1 / Sqrt[eigenvector1.eigenvector1],
eigenvector2 / Sqrt[eigenvector2.eigenvector2],
eigenvector3 / Sqrt[eigenvector3.eigenvector3]};
{amazing1, amazing2, amazing3} = {eigenvalue1 uniteigen1,
eigenvalue2 uniteigen2, eigenvalue3 uniteigen3};
eigenplot = Show[Arrow[amazing1], Arrow[amazing2], Arrow[amazing3],
Arrow[-amazing1], Arrow[-amazing2], Arrow[-amazing3],
BoxRatios -> Automatic, Axes -> True, ViewPoint -> CMView,
AxesLabel -> {"u", "v", "w"}, DisplayFunction -> Identity];
Show[GraphicsArray[{eigenplot, uvwplot}],
DisplayFunction -> $DisplayFunction];
```



Feast your eyes on the ellipsoid built from the amazing vectors shown beside the uvw-plot of the original xyz-sphere

$$x^2 + y^2 + z^2 = 1:$$

```
eigenellipsoid = ParametricPlot3D[Evaluate[x[r, s, t] amazing1 +
y[r, s, t] amazing2 + z[r, s, t] amazing3], {s, slow, shigh},
{t, tlow, thigh}, BoxRatios -> Automatic, ViewPoint -> CMView,
AxesLabel -> {"u", "v", "w"}, DisplayFunction -> Identity];
Show[GraphicsArray[{eigenellipsoid, eigenplot, uvwplot}],
DisplayFunction -> $DisplayFunction];
```



Radical.

Again, just for good measure, look at:

```
Clear[gradu, gradv, gradw, Vuvw]
gradu[x_, y_, z_] =
{D[u[x, y, z], x], D[u[x, y, z], y], D[u[x, y, z], z]};
gradv[x_, y_, z_] =
{D[v[x, y, z], x], D[v[x, y, z], y], D[v[x, y, z], z]};
gradw[x_, y_, z_] =
{D[w[x, y, z], x], D[w[x, y, z], y], D[w[x, y, z], z]};
Vuvw[x_, y_, z_] = Det[{gradu[x, y, z], gradv[x, y, z], gradw[x, y, z]}];
{eigenvalue1 eigenvalue2 eigenvalue3, Vuvw[x, y, z]}
{-21.565, -21.565}
```

Double zingo.

Play with some more choices of nine new numbers

ax, ay, az, bx, by, bz, cx, cy, and cz,

with bx = ay, cy = bz, and cx = az, and try it again until you get enough experience to form your own opinions about the answers to the questions:

- What do you think unit eigenvectors are?
- What do you think eigenvalues are?
- Why are lots of folks delighted with eigenvectors and eigenvalues?

□Tip:

Don't look up the answer in a linear algebra book, because most linear algebra books don't look at eigenvector and eigenvalues from this visual geometric perspective. Instead, they make eigenvectors and eigenvalues the final products of some dreary algebra. Sadly decadent.

□G.11.b)

What do you think happens to the ellipsoid when one of the eigenvalues is 0?

What do you think happens to the ellipsoid when two of the eigenvalues are 0?

□G.11.c)

If what you've seen so far hasn't blown your mind, look at these:

```
{ax, ay, az, by, bz, cz} = {4.6, 2.1, -1.0, 5.8, 1.7, 3.9};
{{ax, ay, az}, {bx, by, bz}, {cx, cy, cz}} =
```

```

{{ax, ay, az}, {ay, by, bz}, {az, bz, cz}};
matrix = {{ax, ay, az}, {bx, by, bz}, {cx, cy, cz}};
MatrixForm[matrix]

```

$$\begin{pmatrix} 4.6 & 2.1 & -1. \\ 2.1 & 5.8 & 1.7 \\ -1. & 1.7 & 3.9 \end{pmatrix}$$

Go from xyz-space to uvw-space with:

```

Clear[u, v, w, x, y, z]
{u[x_, y_, z_], v[x_, y_, z_], w[x_, y_, z_]} =
{ax x + ay y + az z, bx x + by y + bz z, cx x + cy y + cz z}
{4.6 x + 2.1 y - 1. z, 2.1 x + 5.8 y + 1.7 z, -1. x + 1.7 y + 3.9 z}

```

Check this out:

```

eigens = Eigensystem[N[matrix]];
{eigenvalue1, eigenvalue2, eigenvalue3} = eigens[[1]];
{eigenvector1, eigenvector2, eigenvector3} = eigens[[2]];
{uniteigen1, uniteigen2, uniteigen3} = {
  eigenvector1 / Sqrt[eigenvector1 . eigenvector1],
  eigenvector2 / Sqrt[eigenvector2 . eigenvector2],
  eigenvector3 / Sqrt[eigenvector3 . eigenvector3]
};
{amazing1, amazing2, amazing3} = {eigenvalue1 uniteigen1,
  eigenvalue2 uniteigen2, eigenvalue3 uniteigen3}
{{3.80262, 6.28164, 1.86989}, {-3.36648, 0.897864, 3.82985},
{-0.881388, 0.821509, -0.967344}}
{u[x, y, z], v[x, y, z], w[x, y, z]} ==
ExpandAll[Together[{x, y, z} . uniteigen1 amazing1 +
{x, y, z} . uniteigen2 amazing2 + {x, y, z} . uniteigen3 amazing3}]
True

```

It works for any ax, ay, az, bx, by, bz, cx, cy, and cz you feed in as long as you go with bx = ay, cy = bz, and cx = az.

```

{ax, ay, az, by, bz, cz} =
{Random[Real, {-10, 10}], Random[Real, {-10, 10}],
Random[Real, {-10, 10}], Random[Real, {-10, 10}],
Random[Real, {-10, 10}], Random[Real, {-10, 10}]};
{{ax, ay, az}, {bx, by, bz}, {cx, cy, cz}} =
{{ax, ay, az}, {ay, by, bz}, {az, bz, cz}}
matrix = {{ax, ay, az}, {bx, by, bz}, {cx, cy, cz}};
MatrixForm[matrix]
{{-0.956152, -5.66264, 7.93974}, {-5.66264, -0.216274, -5.03065},
{7.93974, -5.03065, 5.48548}}

```

$$\begin{pmatrix} -0.956152 & -5.66264 & 7.93974 \\ -5.66264 & -0.216274 & -5.03065 \\ 7.93974 & -5.03065 & 5.48548 \end{pmatrix}$$

Go from xyz-space to uvw-space with:

```

Clear[u, v, w, x, y, z]
{u[x_, y_, z_], v[x_, y_, z_], w[x_, y_, z_]} =
{ax x + ay y + az z, bx x + by y + bz z, cx x + cy y + cz z}
{-0.956152 x - 5.66264 y + 7.93974 z, -5.66264 x - 0.216274 y - 5.03065 z,
7.93974 x - 5.03065 y + 5.48548 z}
eigens = Eigensystem[N[matrix]];
{eigenvalue1, eigenvalue2, eigenvalue3} = eigens[[1]];
{eigenvector1, eigenvector2, eigenvector3} = eigens[[2]];
{uniteigen1, uniteigen2, uniteigen3} = {
  eigenvector1 / Sqrt[eigenvector1 . eigenvector1],
  eigenvector2 / Sqrt[eigenvector2 . eigenvector2],
  eigenvector3 / Sqrt[eigenvector3 . eigenvector3]
};
{amazing1, amazing2, amazing3} = {eigenvalue1 uniteigen1,
  eigenvalue2 uniteigen2, eigenvalue3 uniteigen3};
{u[x, y, z], v[x, y, z], w[x, y, z]} ==
ExpandAll[Together[{x, y, z} . uniteigen1 amazing1 +
{x, y, z} . uniteigen2 amazing2 + {x, y, z} . uniteigen3 amazing3}]
True

```

In other words when you go with any numbers

ax, ay, az, bx, by, bz, cx, cy, and cz,

with

bx = ay, cy = bz, and cx = az,

and you put

$u[x, y, z] = ax x + ay y + az z,$

$v[x, y, z] = bx x + by y + bz z,$  and

$w[x, y, z] = cx x + cy y + cz z,$

then you can count on having

$$\begin{aligned} \{u[x, y, z], v[x, y, z], w[x, y, z]\} \\ = \{ \{x, y, z\} \cdot uniteigen1 \text{ amazing1} \\ + \{x, y, z\} \cdot uniteigen2 \text{ amazing2} \\ + \{x, y, z\} \cdot uniteigen3 \text{ amazing3} \} \end{aligned}$$

This resolves  $\{u[x, y, z], v[x, y, z], w[x, y, z]\}$  into three perpendicular components in the directions of the three unit eigenvectors because

amazing1 = eigenvalue1 uniteigen1,  
amazing2 = eigenvalue2 uniteigen2, and  
amazing3 = eigenvalue3 uniteigen3.

Fancy folks call this the "spectral theorem" and make this the centerpiece of a good linear algebra course. Some interesting thoughts about stretching, shrinking, and flipping may come to you mind. If so, write them down.

On the other hand, if you don't want to say anything, then don't. It's your choice and you won't be graded down for copping out.