

Matrices, Geometry & Mathematica

Authors: Bruce Carpenter, Bill Davis and Jerry Uhl ©2001

Producer: Bruce Carpenter

Publisher: Math Everywhere, Inc.

MGM.10 The Spectral Theorem for Symmetric Matrices and the Holy Grail of Matrix Theory GIVE IT A TRY!

G.1) Symmetric matrices and the SVD

G.1.a.i) Setting the parameters

Here's a matrix with three cleared entries r, s, t, x, y and z

```
Clear[r, s, t, x, y, z];
A = {
  {2.1, 5., -2.3, 0.0},
  {r, -4.2, 1.7, 0.9},
  {s, t, 4.2, -2.7},
  {x, y, z, -1.8}
};
MatrixForm[A]
{
  {2.1, 5., -2.3, 0}
  {r, -4.2, 1.7, 0.9}
  {s, t, 4.2, -2.7}
  {x, y, z, -1.8}
}
```

Set r, s, t, x, y and z so that the resulting matrix is symmetric.

G.1.a.ii) Hessians are symmetric

The hessian of a function of three variables f[x,y,z] is given by

$$H_f[x, y, z] = \begin{pmatrix} f^{(2,0,0)}[x, y, z] & f^{(1,1,0)}[x, y, z] & f^{(1,0,1)}[x, y, z] \\ f^{(1,1,0)}[x, y, z] & f^{(2,0,0)}[x, y, z] & f^{(0,1,1)}[x, y, z] \\ f^{(1,0,1)}[x, y, z] & f^{(0,1,1)}[x, y, z] & f^{(0,0,2)}[x, y, z] \end{pmatrix}$$

You make the call.

Is this matrix symmetric?

G.1.a.iii) Symmetric matrices are square matrices

Does a matrix A that hits on 4D and hangs in 3D have any chance of being symmetric?

Why or why not?

G.1.b.i) The transpose manipulation

Here is a random matrix A:

```
hitdim = Random[Integer, {2, 6}];
hangdim = Random[Integer, {2, 9}];
Clear[i, j];
A = Table[Random[Real, {-4, 4}], {i, 1, hangdim}, {j, 1, hitdim}];
MatrixForm[A]
{
  {-1.19105, 3.35461}
  {3.82957, 0.227172}
  {-3.04369, -0.565715}
  {0.447953, 3.51817}
  {1.43288, -0.877713}
}
```

Here's a random X in hitdimD:

```
X = Table[Random[Real, {-4, 4}], {j, 1, hitdim}];
{-2.88637, 2.13237}
```

Here's a random Y in hangdimD:

```
Y = Table[Random[Real, {-4, 4}], {i, 1, hangdim}];
{3.05618, -3.58059, 0.400993, 2.95193, 3.88463}
```

Here's a calculation of Y*(A.X)

```
Y.(A.X)
68.2432
```

And here's a calculation of (A^t.Y).X

```
(Transpose[A].Y).X
68.2432
```

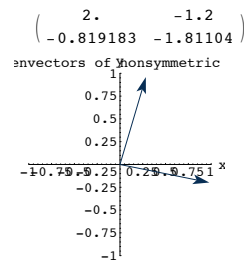
Say why you are not at all surprised with the outcome.

G.1.c.i) Eigenvectors of non-symmetric matrices

Here's a non-symmetric 2D matrix A shown with a plot of the unit eigenvectors of A:

```
A = {
  {2.0, -1.2},
  {Random[Real, {-0.9, -0.1}], Random[Real, {-3, 3}]}
};
MatrixForm[A]
{
  {2.0, -1.2}
  {Random[Real, {-0.9, -0.1}], Random[Real, {-3, 3}]}
}
Clear[eigenvector];
{eigenvector[1], eigenvector[2]} = Eigenvectors[A];
eigenplot = Table[Arrow[eigenvector[k],
  Tail -> {0, 0}, VectorColor -> Indigo], {k, 1, 2}];
Show[eigenplot, Axes -> True,
  AxesLabel -> {"x", "y"}, PlotRange -> {{-1, 1}, {-1, 1}},
  PlotLabel -> "Unit eigenvectors of nonsymmetric matrix A"];

```



Here is the dot product of the two plotted eigenvectors:

```
eigenvector[1].eigenvector[2]
0.0882971
```

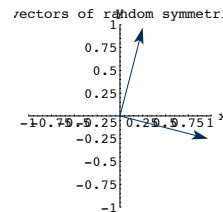
Does the outcome surprise you?

Why or why not?

G.1.c.ii) Eigenvectors of symmetric matrices

Now do the same thing for a random symmetric 2D matrix A:

```
B = {
  {Random[Real, {-3, 3}], Random[Real, {-3, 3}]},
  {Random[Real, {-3, 3}], Random[Real, {-3, 3}]}
};
A = (B + Transpose[B]) / 2;
MatrixForm[A]
{
  {2.3358, -0.452732}
  {-0.452732, 0.698261}
}
Clear[eigenvector];
{eigenvector[1], eigenvector[2]} = Eigenvectors[A];
eigenplot = Table[Arrow[eigenvector[k],
  Tail -> {0, 0}, VectorColor -> Indigo], {k, 1, 2}];
Show[eigenplot, Axes -> True,
  AxesLabel -> {"x", "y"}, PlotRange -> {{-1, 1}, {-1, 1}},
  PlotLabel -> "Unit eigenvectors of random symmetric matrix B"];
{
  {2.3358, -0.452732}
  {-0.452732, 0.698261}
}
```



Here is the dot product of the two plotted eigenvectors:

```
eigenvector[1].eigenvector[2]
0
```

Does the outcome surprise you?

Why or why not?

How is the Spectral Theorem for symmetric matrices related to your response?

G.1.d.i) SVD aligner frame from the eigenview

Here is random 2D matrix A:

```
A = {
  {Random[Real, {-3, 3}], Random[Real, {-3, 3}]},
  {Random[Real, {-3, 3}], Random[Real, {-3, 3}]}
};
MatrixForm[A]
{
  {-1.88455, 1.41846}
  {2.0534, 0.826086}
}
```

Here is A^t.A:

```
MatrixForm[Transpose[A].A]
{
  {7.76801, -0.976875}
  {-0.976875, 2.69444}
}
```

The matrix A^t.A is automatically symmetric because

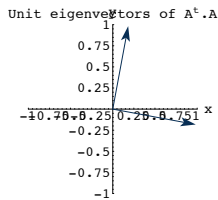
$$(A^t.A)^t = A^t.(A^t)^t = A^t.A$$

The Spectral theorem guarantees a perpendicular frame of eigenvectors of A^t.A.

Here they are:

```
Clear[eigenvector];
{eigenvector[1], eigenvector[2]} = Eigenvectors[Transpose[A].A];
eigenplot = Table[Arrow[eigenvector[k],
  Tail -> {0, 0}, VectorColor -> Indigo], {k, 1, 2}];
Show[eigenplot, Axes -> True, AxesLabel -> {"x", "y"},
  PlotRange -> {{-1, 1}, {-1, 1}},
  PlotLabel -> "Unit eigenvectors of \!\(A^t\)\.A"];

```



How do you get an SVD aligner frame for A from these two vectors?

□G.1.d.ii) SVD stretch factors and hanger frame from the eigenview

Stay with the same matrix A and use the eigenvectors you found in part i) to generate the SVD stretch factors and the corresponding SVD hanger frame for A. Plot the hangerframe.

□G.1.e) Aligner frame for A versus aligner frame for A^T.A

Here's a new random 2D matrix A:

```
A = { Random[Real, {-3, 3}] Random[Real, {-3, 3}]
      Random[Real, {-3, 3}] Random[Real, {-3, 3}] };
MatrixForm[A]
```

$$\begin{pmatrix} 0.945066 & 0.701224 \\ -0.52231 & -2.50988 \end{pmatrix}$$

Here is A^T.A:

```
MatrixForm[Transpose[A].A]
```

$$\begin{pmatrix} 1.16596 & 1.97364 \\ 1.97364 & 6.79121 \end{pmatrix}$$

Let Mathematica come up with an SVD aligner frame for A:

```
SingularValues[A][[3]]
{{0.301185, 0.953566}, {0.953566, -0.301185}}
```

Let Mathematica come up with an SVD aligner frame for A^T.A::

```
SingularValues[Transpose[A].A][[3]]
{{-0.301185, -0.953566}, {-0.953566, 0.301185}}
```

Speculate on why these two turned out the way they did.

G.2) The Spectral Theorem is the Holy Grail of Matrix Theory

□G.2.a) Using the Spectral Theorem to do SVD analysis

Here is a random matrix A

```
hitdim = Random[Integer, {3, 7}];
hangdim = Random[Integer, {3, 7}];
```

```
A = Table[Random[Real, {-10, 10}], {i, 1, hangdim}, {j, 1, hitdim}];
MatrixForm[A]
```

$$\begin{pmatrix} 4.35479 & 8.75522 & -9.54675 \\ 8.84966 & 9.02385 & -8.88522 \\ 9.40472 & -2.15282 & -8.35597 \\ -8.59679 & 7.3843 & -6.11601 \\ -6.14196 & 9.65264 & -7.84692 \end{pmatrix}$$

The Spectral theorem guarantees an orthonormal basis of hitdimD consisting of eigenvectors of A^T.A. Here is one such:

```
Clear[eigenvector];
eigenvector[k_] := Eigenvectors[Transpose[A].A][[k]];
Table[eigenvector[k],
      {k, 1, Length[Eigenvectors[Transpose[A].A]}]]
{{0.182161, 0.655408, -0.732979},
 {0.93316, -0.350169, -0.0812}, {0.309886, 0.669195, 0.675388}}
```

- Use these eigenvectors of A^T.A to come up with an SVD aligner frame for A. Put answer here.
- Hit A on your aligner frame vectors to come up with the SVD stretch factors for A. Put answer here.
- Hit A on your aligner frame vectors to come up with the SVD hangerframe corresponding to the aligner frame you came up with earlier. Put answer here.

□G.2.b) Rank, orthonormal bases of null space, column space and row space

Here is a random matrix A

```
hitdim = Random[Integer, {5, 8}];
hangdim = Random[Integer, {2, 4}];
A = Table[Random[Real, {-10, 10}], {i, 1, hangdim}, {j, 1, hitdim}];
MatrixForm[A]
```

$$\begin{pmatrix} 1.55645 & -9.86011 & -5.07556 & -4.69159 & 8.80283 & -3.01033 \\ 2.58703 & 7.04944 & 7.16909 & 2.63488 & 3.83181 & 6.59619 \\ 8.31944 & 3.61103 & 2.71703 & 7.19147 & 0.47226 & 1.96699 \end{pmatrix}$$

Here is a look at A^T.A:

```
MatrixForm[Transpose[A].A]
```

$$\begin{pmatrix} 78.3283 & 32.932 & 33.251 & 59.3432 & 27.5431 & 28.7434 \\ 32.932 & 159.956 & 110.395 & 90.8027 & -58.0794 & 83.2846 \\ 33.251 & 110.395 & 84.5394 & 62.2416 & -15.9255 & 67.9122 \\ 59.3432 & 90.8027 & 62.2416 & 80.6709 & -27.8067 & 45.649 \\ 27.5431 & -58.0794 & -15.9255 & -27.8067 & 92.3957 & -0.295162 \\ 28.7434 & 83.2846 & 67.9122 & 45.649 & -0.295162 & 56.4409 \end{pmatrix}$$

The Spectral theorem guarantees an orthonormal basis of hitdimD consisting of eigenvectors of A^T.A. Here is one such:

```
Clear[eigenvector];
eigenvector[k_] := Eigenvectors[Transpose[A].A][[k]];
Table[eigenvector[k],
      {k, 1, Length[Eigenvectors[Transpose[A].A]}]]
{{-0.22616, -0.641392, -0.457035, -0.417066, 0.179495, -0.349891},
 {-0.597849, 0.212747, -0.0637351, -0.134356, -0.744949, -0.142314},
 {0.51084, -0.173419, -0.33973, 0.49495, -0.445732, -0.387169},
 {-0.574865, -0.123024, -0.0558049, 0.743633, 0.308028, -0.0583927},
 {-0.435267 - 0.00823909 i, -0.304208 + 0.116666 i,
 0.464264 - 0.135974 i, 0.56209, 0.186622 + 0.0523418 i,
 -0.341705 - 0.00407327 i}, {-0.435267 + 0.00823909 i,
 -0.304208 - 0.116666 i, 0.464264 + 0.135974 i, 0.56209,
 0.186622 - 0.0523418 i, -0.341705 + 0.00407327 i}}
```

Now hit each of these eigenvectors with A:

```
Table[A.eigenvector[k],
      {k, 1, Length[Eigenvectors[Transpose[A].A]}]]
{{12.882, -11.1021, -9.04217}, {-8.20364, -4.65108, -5.97666},
 {-0.850947, -5.29417, 5.28798}, {0, 0, 0}, {0, 0, 0}, {0, 0, 0}}
```

Use the information above to come up with quick calculations of:

- The rank of A. Put answer here.
- An orthonormal basis the row space of A and the dimension of the row space of A. Put answer here.
- An orthonormal basis of the null space of A and the dimension of the null space of A. Put answer here.
- An orthonormal basis of the column space of A and the dimension of the column space of A. Put answer here.
- PseudoInverse[A].Y for any Y in hangdimD.

□G.2.c) Facts about dimensions

Given a matrix A hitting on hitdimD and hanging in hangdim D, you already know that A^T.A is a symmetric matrix which hits on hitdimD and hangs in hitdimD.

The Spectral theorem gives you an orthonormal basis {X₁, X₂, X₃, X₄, ..., X_{hitdim}} of hitdimD consisting of unit eigenvectors of A^T.A. Group and (if necessary relabel) these vectors into two groups.

Group 1 = {X₁, X₂, X₃, ..., X_p} for which A.X_j ≠ {0, 0, ..., 0} for all j = 1, 2, ..., p. Group 1 gives an SVD aligner frame for A.

Group 2 = {X_{p+1}, X_{p+2}, X_{p+3}, ..., X_{hitdim}} for which A.X_j = {0, 0, ..., 0} for all j = p+1, p+2, ..., hitdimD.

The facts of the matter are:

- The rank of A is p.
- To get an orthonormal basis of the column space R[A], you just go with $\left\{ \frac{A.X_1}{\|A.X_1\|}, \frac{A.X_2}{\|A.X_2\|}, \frac{A.X_3}{\|A.X_3\|}, \dots, \frac{A.X_p}{\|A.X_p\|} \right\}$. The upshot: SVD hangerframe for A is an orthonormal basis for R[A].
- To get an orthonormal basis of the row space R[A^T], you just go with {X₁, X₂, X₃, ..., X_p}. The upshot: SVD alignerframe (corresponding to non-zero stretch factors) for A is an orthonormal basis for R[A^T].
- To get an orthonormal basis of N[A], you just go with {X_{p+1}, X_{p+2}, X₃, ..., X_{hitdim}}. These are the eigenvectors of A^T.A that are hit into {0, 0, ..., 0} by A.

Given this:

- What is the dimension of the column space R[A]? Put answer here.
- What is the dimension of the row space R[A^T]? Put answer here.
- What is the dimension of the null space N[A]? Put answer here.
- When you add (dimension of R[A]) + (dimension of N[A]), what do you get? Put answer here.

G.3) How SVD stretch factors of A are related to eigenvalues of A^T.A

□G.3.a.i) How SVD stretch factors of A are related to eigenvalues of A^T.A

Here's a random 2D matrix:

```
A = { Random[Real, {-3, 3}] Random[Real, {-3, 3}]
      Random[Real, {-3, 3}] Random[Real, {-3, 3}] };
MatrixForm[A]
```

```
( 0.394145  2.94215 )
(-1.02352  -0.567313 )
```

The SVD stretch factors of A are:

```
| stretches = SingularValues[A][[2]]
{3.05775, 0.911696}
```

Compare with the square roots of the eigenvalues of A^t.A

```
| f[x_] = Sqrt[x];
Map[f, Eigenvalues[Transpose[A].A]]
{3.05775, 0.911696}
```

Try the same thing for other random matrices:

```
hitdim = Random[Integer, {3, 7}];
hangdim = Random[Integer, {3, 7}];
A =
Table[Random[Real, {-10, 10}], {i, 1, hangdim}, {j, 1, hitdim}];
MatrixForm[A]
```

```
( 1.66118  7.65409  5.03182  -2.03093 )
(-3.26326  2.34568  6.22899  -9.0206 )
( 4.14971  5.29624  9.0599  -1.65548 )
```

The SVD stretch factors of A are:

```
| stretches = SingularValues[A][[2]]
{16.5255, 8.49759, 3.44979}
```

Compare with the square roots of the eigenvalues of A^t.A

```
| f[x_] = Sqrt[x];
Map[f, Eigenvalues[Transpose[A].A]]
{16.5255, 8.49759, 3.44979, 8.12808 × 10-8}
```

The strong suggestion is that the SVD stretch factors of any matrix A are the square roots of the eigenvalues of A^t.A. Explain why this is guaranteed for any matrix A.

Click on the right for a tip on how to get started.

For any matrix A, the Spectral theorem gives you an orthonormal basis

{X₁, X₂, X₃, X₄, ..., X_{hitdim}}

of hitdimD consisting of **unit** eigenvectors of A^t.A.

Group and (if necessary relabel) these vectors into two groups:

Group 1 = {X₁, X₂, X₃, ..., X_p} for which A.X_j ≠ {0,0,...,0} for all j = 1,2,...,p.

Group 2 = {X_{p+1}, X_{p+2}, X_{p+3}, ..., X_{hitdim}} for which A.X_j = {0,0,...} for all j = p+1, p+2, ..., hitdimD.

Facts of the matter:

- An SVD alignerframe for A, {alignerframe[1], alignerframe[2], alignerframe[3], ..., alignerframe[p]}

is {X₁, X₂, X₃, ..., X_p}.

- The corresponding nonzero SVD stretch factors for A, {stretch[1], stretch[2], stretch[3], ..., stretch[p]},

are {||A.X₁||, ||A.X₂||, ||A.X₃||, ..., ||A.X_p||}.

So stretch[k] = ||A.X_k|| = $\sqrt{A.X_k.A.X_k} = \sqrt{X_k.A^t.A.X_k}$

□G.3.a.ii) The stretch factors of A^t.A are the eigenvalues of A^t.A

Here's a random matrix A:

```
hitdim = Random[Integer, {3, 7}];
hangdim = Random[Integer, {2, 4}];
A = Table[Random[Real, {-4, 4}], {i, 1, hangdim}, {j, 1, hitdim}];
MatrixForm[A]
```

```
(-3.87284  3.48002  -3.70382  1.8934  -0.959655  -3.39657 )
( 0.107281  -2.8934  2.51482  -3.31944  -2.52803  1.86302 )
```

Here is a look at A^t.A:

```
| MatrixForm[Transpose[A].A]
( 15.0104  -13.788  14.6141  -7.68895  3.44538  13.3542 )
(-13.788  20.4823  -20.1657  16.1935  3.97498  -17.2106 )
( 14.6141  -20.1657  20.0426  -15.3606  -2.80315  17.2654 )
(-7.68895  16.1935  -15.3606  14.6036  6.57462  -12.6152 )
( 3.44538  3.97498  -2.80315  6.57462  7.31187  -1.45023 )
( 13.3542  -17.2106  17.2654  -12.6152  -1.45023  15.0075 )
```

Here are the non-zero SVD stretch factors of A^t.A

```
| stretches = SingularValues[Transpose[A].A][[2]]
{78.0471, 14.4111}
```

Here are the eigenvalues of A^t.A

```
| eigenvalues = Eigenvalues[Transpose[A].A]
{78.0471, 14.4111, 0, 0, 0, 0}
```

Try it again for another random matrix A:

```
hitdim = Random[Integer, {3, 7}];
hangdim = Random[Integer, {2, 4}];
A = Table[Random[Real, {-4, 4}], {i, 1, hangdim}, {j, 1, hitdim}];
MatrixForm[A]
```

```
(-2.14965  -2.38107  -0.540758  -1.32461  3.15565  0.680655  0.967645 )
(-1.71637  -2.50423  2.56216  1.34369  2.94582  -2.63139  3.08214 )
( 1.0475  -2.94758  2.32827  2.47871  -3.05978  3.94582  3.81345 )
( 1.79814  3.46825  -1.9172  1.9631  0.179213  0.00900722  3.40741 )
```

Here are the non-zero SVD stretch factors of A^t.A

```
| stretches = SingularValues[Transpose[A].A][[2]]
{63.805, 54.8974, 31.8292, 11.1111}
```

Here are the eigenvalues of A^t.A

```
| eigenvalues = Eigenvalues[Transpose[A].A]
{63.805, 54.8974, 31.8292, 11.1111, 0, 0, 0}
```

Why did that happen?

□G.3.a.iii) The condition number of A in terms of the condition number of A^t.A

Given any matrix A, you can express the condition number of A in terms of the condition number of A^t.A.

Do it.

If A is ill-conditioned, do you think that A^t.A even more ill-conditioned? Why?

G.4) The guts of Spectral Theorem

□G.4.a) Importance of Spectral Theorem

All through the basics, there was lots of raving about the importance of the Spectral Theorem.

What does the Spectral Theorem say and why is it so important?

□G.4.b) Guts of the proof of the Spectral Theorem

Here's a random symmetric 2D matrix A:

```
Clear[i, j];
B = Table[Random[Real, {-4, 4}], {i, 1, 2}, {j, 1, 2}];
A = 1/2 (B + Transpose[B]);
MatrixForm[A]
```

```
( 2.80745  3.26996 )
( 3.26996  1.12378 )
```

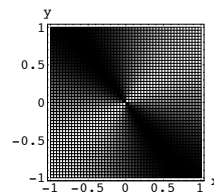
Make this function:

```
Clear[g, x, y];
g[x_, y_] = Expand[(x, y).(A.(x, y)) / (x, y).(x, y)]
```

$$\frac{2.80745 x^2}{x^2 + y^2} + \frac{6.53992 x y}{x^2 + y^2} + \frac{1.12378 y^2}{x^2 + y^2}$$

Look at this graphic:

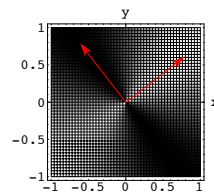
```
glevels = DensityPlot[Evaluate[g[x, y]], {x, -1, 1}, {y, -1, 1},
Axes -> True, AxesLabel -> {"x", "y"}, PlotPoints -> 50];
```



The lighter the square, the higher $g[x,y] = \frac{(x,y).(A.(x,y))}{(x,y).(x,y)}$ is.

Throw in the unit eigenvectors of A:

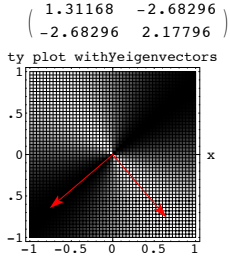
```
Clear[eigenvector];
{eigenvector[1], eigenvector[2]} = Eigenvectors[A];
eigenplot = Table[Arrow[eigenvector[k],
Tail -> {0, 0}, VectorColor -> Red], {k, 1, 2}];
Show[glevels, eigenplot];
```



Here's another:

```
Clear[i, j];
B = Table[Random[Real, {-4, 4}], {i, 1, 2}, {j, 1, 2}];
A = 1/2 (B + Transpose[B]);
MatrixForm[A]
Clear[g, x, y];
g[x_, y_] = Expand[(x, y).(A.(x, y)) / (x, y).(x, y)];
glevels = DensityPlot[Evaluate[g[x, y]],
{x, -1, 1}, {y, -1, 1}, Axes -> True, AxesLabel -> {"x", "y"},
PlotPoints -> 50, DisplayFunction -> Identity];
```

```
Clear[eigenvector];
{eigenvector[1], eigenvector[2]} = Eigenvectors[A];
eigenplot = Table[Arrow[eigenvector[k],
  Tail -> {0, 0}, VectorColor -> Red], {k, 1, 2}];
Show[glevels, eigenplot, PlotLabel ->
  "Density plot with eigenvectors of A",
  DisplayFunction -> $DisplayFunction];
```



The lighter the square, the higher $g(x,y) = \frac{(x,y) \cdot A \cdot (x,y)}{(x,y) \cdot (x,y)}$ is.

Rerun the last cell a couple of times.

How does the proof of the Spectral theorem tell you why all these graphics turned out the way they did?

G.4.c) The eigenvalues of symmetric matrices are all real

How does the proof of the spectral theorem reveal why the eigenvalues of a symmetric matrix are all real?

G.5) Positive semidefinite and positive definite symmetric matrices

G.5.a.i) The square root of a positive semidefinite matrix

If someone hands you a symmetric matrix A with all nonnegative eigenvalues, then it's pretty easy to come up with another symmetric matrix B so that $B \cdot B = B^T \cdot B = A$

Explain in general terms how you can do this.

G.5.a.ii) If $A = B^T \cdot B$, then all the eigenvalues of A are non-negative

When someone hands you any matrix B and you put $A = B^T \cdot B$, then the transpose manipulation yields

$$\begin{aligned} X \cdot (A \cdot X) &= X \cdot ((B^T \cdot B) \cdot X) \\ &= X \cdot (B^T \cdot (B \cdot X)) \\ &= (B \cdot X) \cdot (B \cdot X) = \|B \cdot X\|^2. \end{aligned}$$

Because $\|B \cdot X\|^2$ cannot be negative, this tells you $X \cdot (A \cdot X) \geq 0$

no matter what X is.

At this point you take over and explain why all the eigenvalues of $A = B^T \cdot B$ are non-negative

G.5.a.iii) FYI: Positive semi definite and positive definite matrices

No response from you is asked for here.

At this point you have enough to settle on this factoid:

Saying that all the eigenvalues of a symmetric matrix A are nonnegative is the same as saying

$$X \cdot (A \cdot X) \geq 0$$

no matter what X is.

Most folks call these symmetric matrices by the name "positive semi-definite matrices"

It turns out that

Saying that all the eigenvalues of a symmetric matrix A are positive is the same as saying

$$X \cdot (A \cdot X) > 0$$

unless $X = \{0, 0, \dots, 0\}$

Most folks call these symmetric matrices by the name "positive definite matrices."

G.5.a.iv) Positive definite Hessians

When you go with a function $f(x,y,z)$ and find a point $\{a,b,c\}$ at which $\text{grad}f[a,b,c] = 0$ and you find that the Hessian $H_f[a, b, c]$ is positive definite, how does $f[a,b,c]$ compare to $f[x,y,z]$ when $\{x,y,z\}$ is near $\{a,b,c\}$?

G.6) Hessians for function Max-Min

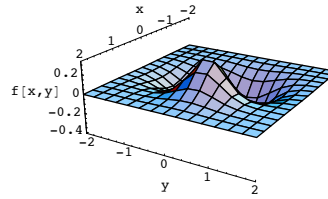
G.6.a) Local maximizers and local minimizers of $f[x,y] = e^{-1.8x^2-1.7y^2} (x - y^2 + \text{Sin}[xy])$

Here's a function $f[x,y]$:

```
Clear[f, x, y];
f[x_, y_] = e^{-1.8 x^2 - 1.7 y^2} (x - y^2 + Sin[x y])
e^{-1.8 x^2 - 1.7 y^2} (x - y^2 + Sin[x y])
```

Here's a plot of $f[x,y]$ in the region $-2 \leq x \leq 2$ and $-2 \leq y \leq 2$:

```
{xlow, xhigh} = {-2, 2};
{ylo, yhi} = {-2, 2};
Plot3D[f[x, y], {x, xlow, xhigh},
  {y, ylo, yhi}, AxesLabel -> {"x", "y", "f[x,y]"},
  Boxed -> False, PlotRange -> All, ViewPoint -> CMView];
```



You can see that this function has a local maximizer and possibly a couple of local minimizers in the plotted region.

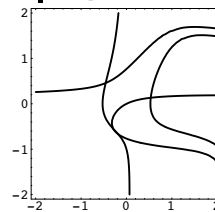
```
Clear[gradf];
gradf[x_, y_] = {Dx f[x, y], Dy f[x, y]}
{e^{-1.8 x^2 - 1.7 y^2} (1 + y Cos[x y]) - 3.6 e^{-1.8 x^2 - 1.7 y^2} x (x - y^2 + Sin[x y]),
  e^{-1.8 x^2 - 1.7 y^2} (-2 y + x Cos[x y]) - 3.4 e^{-1.8 x^2 - 1.7 y^2} y (x - y^2 + Sin[x y])}
```

There is no chance of solving $\text{grad}f[x,y] = \{0,0\}$ with the solve command.

```
xderivcontourplot = ContourPlot[gradf[x, y][[1]], {x, xlow, xhigh},
  {y, ylo, yhi}, Contours -> {0}, ContourStyle -> Thickness[0.01],
  PlotPoints -> 50, ContourSmoothing -> Automatic,
  ContourShading -> False, DisplayFunction -> Identity];
```

```
yderivcontourplot = ContourPlot[gradf[x, y][[2]], {x, xlow, xhigh},
  {y, ylo, yhi}, Contours -> {0}, ContourStyle -> Thickness[0.01],
  PlotPoints -> 50, ContourSmoothing -> Automatic,
  ContourShading -> False, DisplayFunction -> Identity];
```

```
zeroderiv = Show[xderivcontourplot,
  yderivcontourplot, DisplayFunction -> $DisplayFunction];
```



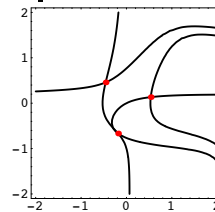
The curves are plots of $\partial_x f[x, y] = 0$ and $\partial_y f[x, y] = 0$.

Let Mathematica come up with the points of intersection:

```
{x1, y1} = {x, y} /. FindRoot[gradf[x, y] == {0, 0}, {x, 0}, {y, -0.5}]
{-0.184622, -0.667741}
{x2, y2} =
{x, y} /. FindRoot[gradf[x, y] == {0, 0}, {x, -0.5}, {y, 0.5}]
{-0.458731, 0.45765}
{x3, y3} = {x, y} /. FindRoot[gradf[x, y] == {0, 0}, {x, 0.5}, {y, 0}]
{0.5349, 0.133367}
```

See them:

```
pointplots = {Graphics[Red, PointSize[0.03], Point[{x1, y1}]],
  Graphics[Red, PointSize[0.03], Point[{x2, y2}]],
  Graphics[Red, PointSize[0.03], Point[{x3, y3}]]};
Show[zeroderiv, pointplots];
```



Move in with the Hessian to determine what's happening at these points.

G.6.b) Local maximizers and local minimizers and saddle points of

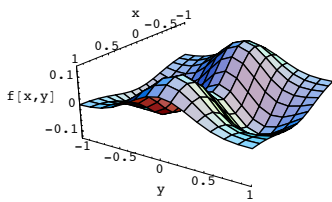
$$f[x,y] = e^{-2x^2-3y^2} (3 \text{Sin}[0.5x]^2 - y^2)$$

Here's a function $f[x,y]$:

```
Clear[f, x, y];
f[x_, y_] = e^{-2 x^2 - 3 y^2} (3 Sin[0.5 x]^2 - y^2)
e^{-2 x^2 - 3 y^2} (-y^2 + 3 Sin[0.5 x]^2)
```

Here's a plot of $f[x,y]$ in the region $-1 \leq x \leq 1$ and $-1 \leq y \leq 1$:

```
{xlow, xhigh} = {-1, 1};
{ylo, yhi} = {-1, 1};
Plot3D[f[x, y], {x, xlow, xhigh},
  {y, ylo, yhi}, AxesLabel -> {"x", "y", "f[x,y]"},
  Boxed -> False, PlotRange -> All, ViewPoint -> CMView];
```



You can see that this function has a local maximizers, a local minimizer and maybe a saddle point in the plotted region:
Locate the points and use the Hessian to determine which are local maximizers, which are local minimizers and which are saddle points.

□ G.6.c) Kicking up the dimension

Here's a function of three variables:

```
Clear[f, gradf, x, y, z];
f[x_, y_, z_] = 5.1 x + 6.5 y + 8.1 z +  $\frac{1.3}{x y z}$ 
5.1 x + 6.5 y +  $\frac{1.3}{x y z}$  + 8.1 z
```

You can't plot this function, but you can calculate its gradient:

```
gradf[x_, y_, z_] = {D[f[x, y, z], x], D[f[x, y, z], y], D[f[x, y, z], z]}
{5.1 -  $\frac{1.3}{x^2 y z}$ , 6.5 -  $\frac{1.3}{x y^2 z}$ , 8.1 -  $\frac{1.3}{x y z^2}$ }
```

The hessian

$$H_f[x, y, z] = \begin{pmatrix} f^{(2,0,0)}[x, y, z] & f^{(1,1,0)}[x, y, z] & f^{(1,0,1)}[x, y, z] \\ f^{(1,1,0)}[x, y, z] & f^{(0,2,0)}[x, y, z] & f^{(0,1,1)}[x, y, z] \\ f^{(1,0,1)}[x, y, z] & f^{(0,1,1)}[x, y, z] & f^{(0,0,2)}[x, y, z] \end{pmatrix}$$

is given by

```
Hf[x_, y_, z_] = {
  {D[{x,2} f[x, y, z], D[{x,y} f[x, y, z], D[{x,z} f[x, y, z],
  {D[{y,x} f[x, y, z], D[{y,2} f[x, y, z], D[{y,z} f[x, y, z],
  {D[{z,x} f[x, y, z], D[{z,y} f[x, y, z], D[{z,2} f[x, y, z]
}];
MatrixForm[Hf[x, y, z]]
```

$$\begin{pmatrix} -\frac{2.6}{x^2 y z} & -\frac{1.3}{x^2 y^2 z} & -\frac{1.3}{x^2 y z^2} \\ -\frac{1.3}{x^2 y^2 z} & -\frac{2.6}{x y^2 z} & -\frac{1.3}{x y^2 z^2} \\ -\frac{1.3}{x^2 y z^2} & -\frac{1.3}{x y^2 z^2} & -\frac{2.6}{x y z^3} \end{pmatrix}$$

Find the point {a,b,c} with a > 0, b > 0 and c > 0 at which gradf[a,b,c] = {0,0,0}.

Use the Hessian to determine whether the point {a,b,c} is a local maximizer, local minimizer or saddle point for f[x,y,z].
Here's a start:

```
Solve[gradf[x, y, z] == {0, 0, 0}]
{{x -> -0.847535, z -> -0.533633, y -> -0.664989},
 {x -> -0.847535 i, z -> -0.533633 i, y -> -0.664989 i},
 {x -> 0.847535 i, z -> 0.533633 i, y -> 0.664989 i},
 {x -> 0.847535, z -> 0.533633, y -> 0.664989}}
```

G.7) Quadratic forms

$$f[x, y] = a x^2 + b x y + c y^2 + d x + e y + g$$

set equal to a constant

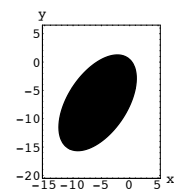
□ G.7.a) Setting a quadratic form equal to a constant and getting an ellipse

Here's an example of something folks call a quadratic form:

```
Clear[f, x, y];
f[x_, y_] = 2.3 x^2 - 1.9 x y + 1.5 y^2 + 12.4 x + 10.8 y
12.4 x + 2.3 x^2 + 10.8 y - 1.9 x y + 1.5 y^2
```

And look at this plot:

```
constant = 5;
curveplot = ContourPlot[f[x, y], {x, -15, 5}, {y, -20, 6},
  Contours -> {constant}, Axes -> True, AxesLabel -> {"x", "y"},
  ContourSmoothing -> Automatic, PlotPoints -> 50,
  AspectRatio -> Automatic, ColorFunction -> Automatic];
```



The border of the black region is a plot of the curve consisting of all the points {x,y} for which

$$f[x, y] = \text{constant} = 5.$$

Parameterize and plot this curve.

Give the perpendicular frame on which it is hung and measure the length of the long axis and the short axis.

The border of the black region is a plot of the curve consisting of all the points {x,y} for which

$$f[x, y] = \text{constant} = 5.$$

Parameterize and plot this curve.

Give the perpendicular frame on which it is hung and measure the length of the long axis and the short axis.

How is the shape of the ellipse related to the ratio of the eigenvalues of the Hessian?

□ G.7.b.i) Setting a quadratic form equal to a constant and getting a hyperbola

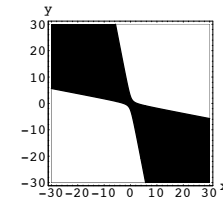
Here's another example of something folks call a quadratic form:

```
Clear[f, x, y];
f[x_, y_] = x^2 + 5.4 x y + y^2 + 2.1 x + 1.7
1.7 + 0.2 x + x^2 + 5.4 x y + y^2
```

And look at this plot:

```
constant = 8;
ranger = 30;

curveplot =
  ContourPlot[f[x, y], {x, -ranger, ranger}, {y, -ranger, ranger},
  Contours -> {constant}, Axes -> True, AxesLabel -> {"x", "y"},
  ContourSmoothing -> Automatic, PlotPoints -> 100,
  AspectRatio -> Automatic, ColorFunction -> Automatic];
```



The border of the black region is a plot of the curve consisting of all the points {x,y} for which

$$f[x, y] = \text{constant} = 8.$$

You decide to look at the Hessian of f[x,y]:

```
Clear[H];
Hf[x_, y_] = {
  {D[{x,2} f[x, y], D[{x,y} f[x, y],
  {D[{x,y} f[x, y], D[{y,2} f[x, y]
}];
MatrixForm[Hf[x, y]]
```

$$\begin{pmatrix} 2 & 5.4 \\ 5.4 & 2 \end{pmatrix}$$

And you look at the eigenvalues of the Hessian:

```
Eigenvalues[Hf[x, y]]
```

$$\{7.4, -3.4\}$$

And now you say: "I can explain why the plot came out the way it did."
What did you mean?

□ G.7.b.ii) Setting a quadratic form equal to a constant and getting an ellipse

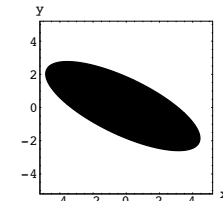
Here's another example of something folks call a quadratic form:

```
Clear[f, x, y];
f[x_, y_] = x^2 + 2.5 x y + 3 y^2 + 2.1 x + 1.7
1.7 + 0.2 x + x^2 + 2.5 x y + 3 y^2
```

And look at this plot:

```
constant = 12;
ranger = 5;

curveplot =
  ContourPlot[f[x, y], {x, -ranger, ranger}, {y, -ranger, ranger},
  Contours -> {constant}, Axes -> True, AxesLabel -> {"x", "y"},
  ContourSmoothing -> Automatic, PlotPoints -> 100,
  AspectRatio -> Automatic, ColorFunction -> Automatic];
```



The border of the black region is a plot of the curve consisting of all the points {x,y} for which

$$f[x, y] = \text{constant} = 12.$$

You decide to look at the Hessian of f[x,y]:

```
Clear[H];
Hf[x_, y_] = {
  {D[{x,2} f[x, y], D[{x,y} f[x, y],
  {D[{x,y} f[x, y], D[{y,2} f[x, y]
}];
MatrixForm[Hf[x, y]]
```

$$\begin{pmatrix} 2 & 2.5 \\ 2.5 & 6 \end{pmatrix}$$

And you look at the eigenvalues of the Hessian:

```
Eigenvalues[Hf[x, y]]
```

{7.20156, 0.798438}

And now you say: "I can explain why the plot came out the way it did." What did you mean?

□G.7.b.iii) Setting a quadratic form equal to a constant and getting a line

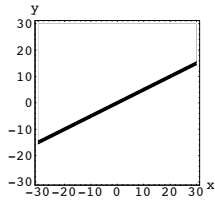
Here's another example of something folks call a quadratic form:

```
Clear[f, x, y];
f[x_, y_] = x^2 - 4 x y + 4 y^2 + 1
1 + x^2 - 4 x y + 4 y^2
```

And look at this plot:

```
constant = 2;
ranger = 30;

curveplot =
ContourPlot[f[x, y], {x, -ranger, ranger}, {y, -ranger, ranger},
Contours -> {constant}, Axes -> True, AxesLabel -> {"x", "y"},
ContourSmoothing -> Automatic, PlotPoints -> 100,
AspectRatio -> Automatic, ColorFunction -> Automatic];
```



This is a plot of the curve defined by setting f[x,y] = constant = 2. This puzzles you so you look at the gradient f[x,y]:

```
Clear[gradf];
gradf[x_, y_] = {D[f[x, y], x], D[f[x, y], y]}
{2 x - 4 y, -4 x + 8 y}
Solve[gradf[x, y] == {0, 0}]
Solve::svars: Equations may not give solutions for all "solve" variables.
{{x -> 2 y}}
```

And you look at the Hessian of f[x,y]:

```
Clear[H];
Hf[x_, y_] = {D[{D[f[x, y], x], D[f[x, y], y]}, x],
D[{D[f[x, y], x], D[f[x, y], y]}, y]};
MatrixForm[Hf[x, y]]
```

$$\begin{pmatrix} 2 & -4 \\ -4 & 8 \end{pmatrix}$$

And you look at the eigenvalues of the Hessian:

```
Eigenvalues[Hf[x, y]]
{0, 10}
```

And now you exclaim: "I can explain why the plot came out the way it did!" What did you mean?

□G.7.b.iv) Setting a quadratic form equal to a constant and getting a parabola

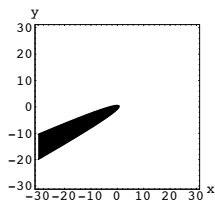
Here's another example of something folks call a quadratic form:

```
Clear[f, x, y];
f[x_, y_] = x^2 - 4.0 x y + 4.0 y^2 + 2.3 x + 1.3 y
2.3 x + x^2 + 1.3 y - 4. x y + 4. y^2
```

And look at this plot:

```
constant = 2;
ranger = 30;

curveplot =
ContourPlot[f[x, y], {x, -ranger, ranger}, {y, -ranger, ranger},
Contours -> {constant}, Axes -> True, AxesLabel -> {"x", "y"},
ContourSmoothing -> Automatic, PlotPoints -> 100,
AspectRatio -> Automatic, ColorFunction -> Automatic];
```



This is a plot of the curve defined by setting f[x,y] = constant = 2. It's not totally clear whether this is a parabola or part of an ellipse. You look at the gradient f[x,y]:

```
Clear[gradf];
gradf[x_, y_] = {D[f[x, y], x], D[f[x, y], y]}
{2.3 + 2 x - 4. y, 1.3 - 4. x + 8. y}
Solve[gradf[x, y] == {0, 0}]
{}
```

And you look at the Hessian of f[x,y]:

```
Clear[H];
Hf[x_, y_] = {D[{D[f[x, y], x], D[f[x, y], y]}, x],
D[{D[f[x, y], x], D[f[x, y], y]}, y]};
MatrixForm[Hf[x, y]]
```

$$\begin{pmatrix} 2 & -4. \\ -4. & 8. \end{pmatrix}$$

And you look at the eigenvalues of the Hessian:

```
Eigenvalues[Hf[x, y]]
{10., 0}
```

And now you exclaim: "I can explain why the plot came out the way it did! And I know that this plot is a parabola!"

What did you mean? And how did you come up with your decision?

G.8) Quadratic forms defined by a symmetric matrix A via

$$f[x, y] = \{x, y\} \cdot A \cdot \{x, y\}$$

□G.8.a) Quadratic forms f[x,y]={x,y}.(A.{x,y}) don't give rise to parabolas

Here's a symmetric matrix A:

```
Clear[f, x, y, a, b, c, d];
A = {{a, b/2}, {b/2, c}};
MatrixForm[A]
```

$$\begin{pmatrix} a & \frac{b}{2} \\ \frac{b}{2} & c \end{pmatrix}$$

Use A to define this function f[x,y]={x,y}.(A.{x,y}):

```
f[x_, y_] = Expand[{x, y} \cdot (A \cdot {x, y})]
a x^2 + b x y + c y^2
```

This is a quadratic form.

Look at

```
gradf[x_, y_] = {D[f[x, y], x], D[f[x, y], y]}
{2 a x + b y, b x + 2 c y}
```

Notice that grad[0,0] = {0,0}:

```
gradf[0, 0]
{0, 0}
```

Say why this gives away the fact that when you set f[x,y] = constant, you will not get a parabola.

□G.8.a.ii) The Hessian

Stay with the same function f[x,y] and calculate the Hessian Hf[x, y].

How is Hf[x, y] related to A?

```
Hf[x_, y_] =
{{D[{D[f[x, y], x], D[f[x, y], y]}, x],
D[{D[f[x, y], x], D[f[x, y], y]}, y]};
MatrixForm[Hf[x, y]]
```

$$\begin{pmatrix} 2 a & b \\ b & 2 c \end{pmatrix}$$

□G.8.a.iii)

How do the eigenvalues and eigenvectors of a given symmetric matrix A tell you how how curves defined by setting

$$f[x,y]=\{x,y\} \cdot (A \cdot \{x,y\}) = \text{constant}$$

plot out?

□G.8.a.iv)

Explain this:

What do you look for in symmetric matrix A do you to be sure that it is impossible for ||{x,y}|| to go to infinity while maintaining the equality {x,y}.(A.{x,y}) = 20?

G.9) A^t.A and the Grammian test for linear independence

□G.9.a.i) The rank of A^t.A is the same as the rank of A

Explain this:

Given any matrix A, the rank of A^t.A is the same as the rank of A.

□G.9.a.ii) If A is of full rank, then A^t.A is invertible

Explain this:

If A is of full rank, then A^t.A is invertible.

□G.9.b.i) The Grammian matrix

Given a set of vectors

$$\{X[1], X[2], \dots, X[n]\}$$

in kD, the corresponding Grammian matrix is the matrix with the entry

$$X[i] \cdot X[j]$$

in its ith horizontal row and jth vertical column. The resulting Grammian matrix is a nD (square) symmetric matrix.

Clued in folks know that the Grammian matrix is the same as A^t.A for a certain matrix A. What matrix A are they thinking about?

□ G.9.b.ii) The Grammian test for linear independence

Given a set vectors
 $\{X[1], X[2], \dots, X[n]\}$,
 clued in folks know that saying that
 $\{X[1], X[2], \dots, X[n]\}$ is a linearly independent set
 is the same as saying that the determinant of the corresponding Grammian matrix is not 0.
 How do they know this?

□ G.9.b.iii) A case in which the determinant of the Grammian matrix is 0

Here is a random set consisting of at least seven vectors in 6D:

```
number = Random[Integer, {7, 12}];
Clear[X, i, j];
Table[X[i_] = Table[Random[Real, {-10, 10}], {j, 1, 6}],
      {i, 1, number}]

{{-0.142322, -5.36435, 1.62544, 2.81385, 4.03701, -1.56111},
 {-0.725115, 2.9493, 4.5034, 3.94354, 0.604262, -2.2577},
 {9.59564, -6.5045, -9.41826, -0.776227, -7.42298, -5.25089},
 {-7.02166, 6.41432, -0.702186, -7.59189, 8.73415, -9.03057},
 {9.44014, 7.77246, -2.89129, -1.84442, -4.59688, -0.666434},
 {7.83383, 5.20629, 0.899724, 5.39003, -2.77044, -2.53601},
 {1.30408, 1.89453, -3.35218, 8.24022, -1.27294, -2.85458},
 {-6.33052, -8.17411, 9.42925, -5.26269, -5.06467, -9.14353}}
```

The determinant of the corresponding Grammian matrix is:

```
grammian = Table[X[i].X[j], {i, 1, number}, {j, 1, number}];
Det[grammian]
0
```

Explain how you could have known this in advance.

G.10) Another version of the Holy Grail: Working with $A \cdot A^t$ instead of $A^t \cdot A$

□ G.10.a) Working with $A \cdot A^t$ instead of $A^t \cdot A$

Given a matrix A, the matrix $A \cdot A^t$ is guaranteed to be symmetric.
 The Spectral theorem steps in to guarantee an orthonormal basis (perpendicular frame)
 $\{Y_1, Y_2, Y_3, Y_4, \dots, Y_{\text{handim}}\}$
 of eigenvectors of $A \cdot A^t$ spanning all of handdimD.

How are these vectors related to the SVD of A?

How can you use these vectors to find the SVD stretch factors of A?

G.11) Trying to get SVD to spit out eigenvectors for symmetric matrices

□ G.11.a.i) Eigenvectors and SVD for symmetric matrices

Here's 2D symmetric matrix:

```
A = {{1.6, 0.2},
      {0.2, 0.3}};
MatrixForm[A]
```

$$\begin{pmatrix} 1.6 & 0.2 \\ 0.2 & 0.3 \end{pmatrix}$$

Compare *Mathematica's* calculation of eigenvectors of A with *Mathematica's* calculation of an SVD alignerframe of A:

```
Eigenvalues[A]
{{0.988883, 0.148696}, {-0.148696, 0.988883}}
SingularValues[A][[3]]
{{-0.988883, -0.148696}, {-0.148696, 0.988883}}
```

For this matrix, the calculated SVD aligner frame consists of eigenvectors.

Here's another 2D symmetric matrix A:

```
s = 0.3 Pi;
perpframe = {{Cos[s], Sin[s]}, {Cos[s + Pi/2], Sin[s + Pi/2]}};
aligner = perpframe;
hanger = Transpose[aligner];
diagonal = DiagonalMatrix[{1.2, -0.6}];

A = hanger.diagonal.aligner;
MatrixForm[A]
```

$$\begin{pmatrix} 0.0218847 & 0.855951 \\ 0.855951 & 0.578115 \end{pmatrix}$$

Compare *Mathematica's* calculation of eigenvectors of A with *Mathematica's* calculation of an SVD alignerframe of A:

```
Eigenvalues[A]
{{-0.587785, -0.809017}, {-0.809017, 0.587785}}
SingularValues[A][[3]]
{{0.587785, 0.809017}, {-0.809017, 0.587785}}
```

Again for this matrix, the calculated SVD aligner frame consists of eigenvectors.

Here's another 2D symmetric matrix A:

```
s = 0.4 Pi;
aligner = {{Cos[s], Sin[s]}, {Cos[s + Pi/2], Sin[s + Pi/2]}};
hanger = Transpose[aligner];
diagonal = DiagonalMatrix[{2, -2}];
A = hanger.diagonal.aligner;
MatrixForm[A]
```

$$\begin{pmatrix} -1.61803 & 1.17557 \\ 1.17557 & 1.61803 \end{pmatrix}$$

Compare *Mathematica's* calculation of eigenvectors of A with *Mathematica's* calculation of an SVD alignerframe of A:

```
Eigenvalues[A]
{{-0.951057, 0.309017}, {-0.309017, -0.951057}}
SingularValues[A][[3]]
{{1., 0}, {0, 1.}}
```

Are the calculated alignerframe vectors also eigenvectors of A?

□ G.11.a.ii) Sometimes any perpendicular frame works for an aligner frame

Stay with the same matrix A as in the very last of part i).
 This the matrix A was made with this frame for both its aligner and hanger frames:

```
aligner
{{0.309017, 0.951057}, {-0.951057, 0.309017}}
```

Yet *Mathematica* comes up with these aligner and hangerframes:

```
Clear[alignerframe, hangerframe];
{alignerframe[1], alignerframe[2]} = SingularValues[A][[3]]
{hangerframe[1], hangerframe[2]} = SingularValues[A][[1]]
{{1., 0}, {0, 1.}}
{{-0.809017, 0.587785}, {0.587785, 0.809017}}
```

And these stretch factors:

```
Clear[stretch];
{stretch[1], stretch[2]} = SingularValues[A][[2]]
{2., 2.}
```

Check:

```
A.alignerframe[1] == stretch[1] hangerframe[1]
True
A.alignerframe[2] == stretch[2] hangerframe[2]
True
```

Mathematica's calculation checks out.

Now look at $A^t \cdot A$:

```
MatrixForm[Transpose[A].A]
```

$$\begin{pmatrix} 4. & 0 \\ 0 & 4. \end{pmatrix}$$

Remembering that an SVD alignerframe is any perpendicular frame that consists of eigenvectors of $A^t \cdot A$, say why any 2D perpendicular frame works as an SVD aligner frame for A.

□ G.11.a.iii) Key info about {stretch[1],stretch[2]} of a symmetric matrix

Take your best shot at this question:

Given a 2D symmetric matrix A, you look at the SVD stretch factors of A, {stretch[1],stretch[2]}.

When you look at

$$\{\text{stretch}[1], \text{stretch}[2]\}$$

and extract the key information, it is possible to decide whether any computed SVD alignerframe is guaranteed to consist of eigenvectors of A.

What is this key information?

□ G.11.a.iv) Key info about {eigenvalue[1],eigenvalue[2]} of a symmetric matrix

Take your best shot at this question:

Given a 2D symmetric matrix A, you look at the eigenvalues of A, {eigenvalue[1],eigenvalue[2]}.

When you look at

$$\{\text{eigenvalue}[1], \text{eigenvalue}[2]\}$$

and extract the key information, it is possible to decide whether any computed SVD alignerframe is guaranteed to consist of eigenvectors of A.

What is this key information?

□ G.11.b.i) Moving to higher dimensions

Take your best shot at this question:

Given a kD symmetric matrix A, you look at the eigenvalues of A, {eigenvalue[1],eigenvalue[2], ...eigenvalue[k]}.

as guaranteed by the Spectral Theorem.

When you look at

$$\{\text{eigenvalue}[1], \text{eigenvalue}[2], \dots, \text{eigenvalue}[k]\}$$

and extract the key information, it is possible to decide whether any computed SVD alignerframe is guaranteed to consist of eigenvectors of A.

What is this key information?

□ **G.11.b.ii) Moving to higher dimensions**

Take your best shot at this question:

Given a $k \times k$ symmetric matrix A , you look at the SVD stretch factors of A ,
 $\{\text{stretch}[1], \text{stretch}[2], \dots, \text{stretch}[k]\}$ (including the zero stretch factors)
as guaranteed by the Spectral Theorem.

When you look at

$\{\text{stretch}[1], \text{stretch}[2], \dots, \text{stretch}[k]\}$

and extract the key information, it is possible to decide whether any computed SVD
alignerframe is guaranteed to consist of eigenvectors of A .

What is this key information?