EECS 16B Designing Information Devices and Systems II
Fall 2021

Note 14: Upper Triangulation

1 Motivation

When studying systems of linear differential equations, we have written them in the form

$$\frac{\mathrm{d}}{\mathrm{d}t}\vec{x} = A\vec{x}, \tag{1}$$

where A is a matrix of scalar real coefficients, and $\vec{x}$ is the state vector. To solve such systems, we have developed a technique that involves diagonalizing A, solving for the state vector in the eigenbasis, and then making a change of basis back to the identity basis to obtain the full solution.

While the above technique is very effective, it relies on A being diagonalizable. Unfortunately, this assumption is not always true. For instance, consider

$$D = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}. \tag{2}$$

Computing its characteristic polynomial, we find that

$$\det(D - \lambda I) = (\lambda - 1)^2, \tag{3}$$

so its only root is $\lambda = 1$. But

$$\mathrm{Null}(D - \lambda I) = \mathrm{Null}\left(\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\right) = \mathrm{span}\left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}\right), \tag{4}$$

which is only one dimensional. Thus, D only has one eigenvector despite being 2×2, and so is not diagonalizable. We call such matrices *defective*.

In this note, we will develop a new change of basis that has similar properties to diagonalization, but that works for *all* matrices, allowing us to solve arbitrary systems of differential equations!

2 Upper-Triangular Form

In particular, we will aim to show that any square matrix A can be transformed, by a change of basis, into the matrix

$$T = \begin{bmatrix} \lambda_1 & ? & ? & \cdots & ? \\ 0 & \lambda_2 & ? & \cdots & ? \\ 0 & 0 & \lambda_3 & \cdots & ? \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda_n \end{bmatrix} \tag{5}$$

where the λ_i are eigenvalues of T and T is in upper-triangular form. Notice that, since for defective matrices there are fewer than n distinct eigenvalues, here we will repeat each eigenvalue in the diagonal in accordance with its *multiplicity*. The multiplicity of an eigenvalue λ_i of a matrix A represents the number of times the linear factor $(\lambda - \lambda_i)$ appears in the characteristic polynomial of A. For instance, consider the defective matrix

$$D = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}. \tag{6}$$

whose characteristic polynomial was shown above to be

$$P_D(\lambda) = (\lambda - 1)^2. \tag{7}$$

Thus, we say that the eigenvalue $\lambda = 1$ of D has a multiplicity of 2 (even though the corresponding eigenspace of D is only one-dimensional).

But does this construction make sense? We need to fill n spots on the diagonal of T with eigenvalues repeated according to their multiplicity, so we need to ensure that the eigenvalues' multiplicities sum to n. The degree of the characteristic polynomial of an $n \times n$ matrix (such as A) is the highest power of λ in the expression $P_A(\lambda) = \det(A - \lambda I)$. This determinant will have one factor of λ for each entry in the diagonal of $A - \lambda I$, and since $A - \lambda I$ is $n \times n$, it will have n factors of λ. Thus the degree of $P_A(\lambda)$ is n. Therefore the sum of the multiplicities of all distinct eigenvalues of A, and in general any $n \times n$ matrix, will be n, so we can indeed produce the λ_i that we need for our desired form to make sense.

3 Computing Upper-Triangular Form

We wish to find a change of basis that converts an arbitrary $n \times n$ square matrix A into the form T. Let this change of basis be represented by the columns $\vec{v}_i$ of the matrix U, such that

$$A = UTU^{-1} = \begin{bmatrix} | & & | \\ \vec{v}_1 & \cdots & \vec{v}_n \\ | & & | \end{bmatrix} \begin{bmatrix} \lambda_1 & ? & ? & \cdots & ? \\ 0 & \lambda_2 & ? & \cdots & ? \\ 0 & 0 & \lambda_3 & \cdots & ? \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda_n \end{bmatrix} \begin{bmatrix} | & & | \\ \vec{v}_1 & \cdots & \vec{v}_n \\ | & & | \end{bmatrix}^{-1}. \tag{8}$$

Right-multiplying by U to get rid of the inverse, we obtain

$$A \begin{bmatrix} | & & | \\ \vec{v}_1 & \cdots & \vec{v}_n \\ | & & | \end{bmatrix} = \begin{bmatrix} | & & | \\ \vec{v}_1 & \cdots & \vec{v}_n \\ | & & | \end{bmatrix} \begin{bmatrix} \lambda_1 & ? & ? & \cdots & ? \\ 0 & \lambda_2 & ? & \cdots & ? \\ 0 & 0 & \lambda_3 & \cdots & ? \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda_n \end{bmatrix}. \tag{9}$$

Now, breaking this matrix down into a series of vector equations, we obtain the system

$$A\vec{v}_1 = \lambda_1 \vec{v}_1 \tag{10}$$
$$A\vec{v}_2 = (?)\vec{v}_1 + \lambda_2 \vec{v}_2 \tag{11}$$
$$A\vec{v}_3 = (?)\vec{v}_1 + (?)\vec{v}_2 + \lambda_3 \vec{v}_3 \tag{12}$$

$$\vdots \tag{13}$$

$$A\vec{v}_n = (?)\vec{v}_1 + (?)\vec{v}_2 + \ldots + (?)\vec{v}_{n-1} + \lambda_n\vec{v}_n. \tag{14}$$

Note that we use the symbol ? to represent *different* unknown quantities each time it is written, not always the same value. Let's also temporarily forget that the λ_i are meant to be the eigenvalues of our system, and instead just treat them as arbitrary scalar coefficients.

Thus, we see that a change of basis that puts a matrix A into upper-triangular form is equivalent to constructing a basis $\{\vec{v}_i\}$ such that $A\vec{v}_i$ can be written as a linear combination of the vectors $\{\vec{v}_1, \vec{v}_2, \ldots, \vec{v}_i\}$, for all i.

One of these equations should immediately stand out – specifically, $A\vec{v}_1 = \lambda_1\vec{v}_1$, since this is the equation defining $\vec{v}_1$ to be an eigenvector of A with eigenvalue λ_1. So a necessary condition to write any square matrix in upper-triangular form is that it must have at least one eigenvector, even if the matrix isn't diagonalizable. Since we are attempting to define our upper-triangularization procedure for every square matrix, we need to prove that every square matrix has at least one eigenvalue. Otherwise, we would not have the necessary condition for some matrices.

4 Existence of at least one eigenvector

Let's try to prove that any square matrix A has at least one eigenvector. Recall that we solve for eigenvalues and eigenvectors by considering the matrix $A - \lambda I$, and searching for eigenvalues λ that caused $A - \lambda I$ to have a nontrivial nullspace. To do so, we viewed the determinant $\det(A - \lambda I)$ as a polynomial $P_A(\lambda)$ in λ, and searched for its roots.

However, the Fundamental Theorem of Algebra tells us that every polynomial must have at least one distinct (possibly complex) root![1] Thus, we will obtain at least one eigenvalue λ such that $A - \lambda I$ has a nontrivial nullspace. By considering an element $\vec{v} \in \text{Null}(A - \lambda I)$, we obtain

$$(A - \lambda I)\vec{v} = \vec{0} \tag{15}$$

$$\implies A\vec{v} - \lambda I\vec{v} =, \vec{0} \tag{16}$$

$$\implies A\vec{v} - \lambda\vec{v} =, \vec{0} \tag{17}$$

$$\implies A\vec{v} = \lambda\vec{v}, \tag{18}$$

so we have obtained an eigenvalue-eigenvector pair $(\lambda, \vec{v})$ for our matrix A, even if A were not diagonalizable.

5 Guessing a basis

So we know how to compute some $\vec{v}_1$ such that $A\vec{v}_1 = \lambda_1\vec{v}_1$. But what about the remaining $\vec{v}_i$ for $i \geq 2$? To determine these $\vec{v}_i$, we will make a guess. We will make a guess that the $\vec{v}_i$ not only form a basis, but in fact form an *orthonormal* basis - in other words, that $\vec{v}_i \perp \vec{v}_j$ for all $i \neq j$, and that $\|\vec{v}_i\| = 1$ for all i. Consider an arbitrary such orthonormal basis, starting with the known eigenvector $\vec{v}_1$ (normalized to be of magnitude 1) and constructing the remaining vectors using the Gram-Schmidt process. First, let's place the

[1] The Fundamental Theorem of Algebra actually tells us that the polynomial has n complex-valued roots, but crucially *some of them may be the same*. In particular they can *all* be the same, in which case we get one distinct eigenvalue.

$n-1$ arbitrarily chosen vectors, which we will denote as $\vec{r}_1, \ldots, \vec{r}_{n-1}$ in a matrix R_{n-1} defined as

$$R_{n-1} = \begin{bmatrix} \vec{r}_1 & \vec{r}_2 & \cdots & \vec{r}_{n-1} \end{bmatrix}, \tag{19}$$

so our full basis which turns A into an upper-triangular matrix will look like

$$U_n = \begin{bmatrix} \vec{v}_1 & R_{n-1} \end{bmatrix}, \tag{20}$$

using block matrix notation. Notice that as the $\vec{r}_i$ are orthonormal, $R_{n-1}^\top R_{n-1} = I_{n-1}$, where I_k represents the k-dimensional identity matrix. And since $\vec{v}_1$ is orthogonal to each $\vec{r}_i$, the whole matrix U_n obeys $U_n^\top U_n = I_n$. This means that $U_n^{-1} = U_n^\top$, a fact we will use in just a bit.

Does this basis U turn A into an upper-triangular matrix? Let's find out, by computing the change of basis and using the fact we just derived:

$$\begin{aligned}
U_n^{-1} A U_n &= U_n^\top A U_n && (21)\\
&= \begin{bmatrix} \vec{v}_1 & R_{n-1} \end{bmatrix}^\top A \begin{bmatrix} \vec{v}_1 & R_{n-1} \end{bmatrix} && (22)\\
&= \begin{bmatrix} \vec{v}_1 & R_{n-1} \end{bmatrix}^\top \begin{bmatrix} A\vec{v}_1 & AR_{n-1} \end{bmatrix} && (23)\\
&= \begin{bmatrix} \vec{v}_1 & R_{n-1} \end{bmatrix}^\top \begin{bmatrix} \lambda_1\vec{v}_1 & AR_{n-1} \end{bmatrix} && (24)\\
&= \begin{bmatrix} \vec{v}_1^\top \\ R_{n-1}^\top \end{bmatrix} \begin{bmatrix} \lambda_1\vec{v}_1 & AR_{n-1} \end{bmatrix} && (25)\\
&= \begin{bmatrix} \vec{v}_1^\top(\lambda_1\vec{v}_1) & \vec{v}_1^\top(AR_{n-1}) \\ R_{n-1}^\top(\lambda_1\vec{v}_1) & R_{n-1}^\top(AR_{n-1}) \end{bmatrix} && (26)\\
&= \begin{bmatrix} \lambda_1\vec{v}_1^\top\vec{v}_1 & \vec{v}_1^\top AR_{n-1} \\ \lambda_1 R_{n-1}^\top\vec{v}_1 & R_{n-1}^\top AR_{n-1} \end{bmatrix} && (27)\\
&= \begin{bmatrix} \lambda_1 & \vec{v}_1^\top AR_{n-1} \\ \lambda_1 R_{n-1}^\top\vec{v}_1 & R_{n-1}^\top AR_{n-1} \end{bmatrix}. && (28)
\end{aligned}$$

What does this matrix look like? Ideally, we'd like it to be upper-triangular, and so of the form

$$\begin{bmatrix} \lambda_1 & ? & ? & \cdots & ? \\ 0 & \lambda_2 & ? & \cdots & ? \\ 0 & 0 & \lambda_3 & \cdots & ? \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda_n \end{bmatrix}. \tag{29}$$

The top two blocks of $U_n^{-1}AU_n$ look alright, since the top row of an upper triangular matrix does not have to contain any zeros.

The bottom two blocks, however, might pose more of an issue. Specifically, comparing the two matrices above, for $U_n^{-1}AU_n$ to be upper triangular, $\lambda_1 R_{n-1}^\top\vec{v}_1 = \vec{0}$, and $R_{n-1}^\top AR_{n-1}$ must itself be an $n-1$-dimensional square upper triangular matrix.

Let's try to verify the first of our requirements. Recall that we chose the $\vec{r}_i$ to be orthonormal to $\vec{v}_1$, so $\vec{r}_i^\top \vec{v}_1 = 0$ for each i. So then

$$\lambda_1 R_{n-1}^\top \vec{v}_1 = \lambda_1 \begin{bmatrix} - & \vec{r}_1^\top & - \\ - & \vec{r}_2^\top & - \\ & \vdots & \\ - & \vec{r}_{n-1}^\top & - \end{bmatrix} \vec{v}_1 = \lambda_1 \begin{bmatrix} \vec{r}_1^\top \vec{v}_1 \\ \vec{r}_2^\top \vec{v}_1 \\ \vdots \\ \vec{r}_{n-1}^\top \vec{v}_1 \end{bmatrix} = \lambda_1 \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \vec{0}, \tag{30}$$

as desired! Therefore, we can rewrite

$$U_n^{-1} A U_n = \begin{bmatrix} \lambda_1 & \vec{v}_1^\top A R_{n-1} \\ \vec{0} & R_{n-1}^\top A R_{n-1} \end{bmatrix}. \tag{31}$$

Unfortunately, recall that we chose the columns of R_{n-1} essentially arbitrarily, so long as together with $\vec{v}_1$ they formed an orthonormal basis for all of n-dimensional space. So there is no guarantee that $R_{n-1}^\top A R_{n-1}$ is upper-triangular, meaning that we aren't quite done yet.

Next, we'll learn how to make a *specific* choice of R_{n-1}, and thus U_n, so that we end up with an upper-triangular matrix. We'll study small-n cases, which will show us how the general case works.

6 Low-dimensional cases

The above approach doesn't always take us to an upper-triangular form, but it certainly takes us closer to one. When *does* it take us to upper-triangular form? Exactly when

$$R_{n-1}^\top A R_{n-1} \tag{32}$$

is itself upper-triangular. And in what case is that matrix upper-triangular? Well, if it's 1×1, then it is upper-triangular by definition, so we'd be done. In other words, we know that when $n = 2$, the above approach yields an upper-triangulation of the original matrix.

Let's take the time to work this out algebraically. Let M_2 be a 2×2 matrix. The above approach lets us construct an orthonormal basis

$$U_2 = \begin{bmatrix} \vec{v}_1 & R_1 \end{bmatrix} \tag{33}$$

such that

$$U_2^{-1} M_2 U_2 = \begin{bmatrix} \lambda_1 & \vec{v}_1^\top M_2 R_1 \\ \vec{0} & R_1^\top M_2 R_1 \end{bmatrix}. \tag{34}$$

But in the 2×2 case, R_1 is simply another unit vector, orthogonal to $\vec{v}_1$. Let this vector be $\vec{v}_2$, so

$$U_2 = \begin{bmatrix} \vec{v}_1 & \vec{v}_2 \end{bmatrix}. \tag{35}$$

Then we see that

$$U_2^{-1} M_2 U_2 = \begin{bmatrix} \lambda_1 & \vec{v}_1^\top M_2 \vec{v}_2 \\ \vec{0} & \vec{v}_2^\top M_2 \vec{v}_2 \end{bmatrix}. \tag{36}$$

Since all the components of the above matrix are in fact 1×1, we have expressed M_2 in upper-triangular form!

This step resolved quickly, on account of *all* 1×1 matrices being upper triangular. This stops being the case for 2×2 matrices. So, let's look at the next case: when $n = 3$. Consider some 3×3 matrix M_3, and construct an orthonormal basis

$$U_3 = \begin{bmatrix} \vec{v}_1 & R_2 \end{bmatrix} \tag{37}$$

such that

$$U_3^{-1} M_3 U_3 = \begin{bmatrix} \lambda_1 & \vec{v}_1^\top M_3 R_2 \\ \vec{0} & R_2^\top M_3 R_2 \end{bmatrix}. \tag{38}$$

As mentioned before, since $R_2^\top M_3 R_2$ is not necessarily upper-triangular, we run into a problem with our solution.

But wait! $R_2^\top M_3 R_2$ is a 2×2 matrix. And we just saw how to upper-triangularize arbitrary 2×2 matrices! Let

$$M_2 = R_2^\top M_3 R_2 \tag{39}$$

and upper-triangularize it as $T_2 = U_2^{-1} M_2 U_2$ for some orthonormal basis U_2 and upper-triangular matrix T_2.

Ideally, we'd be able to combine our "partial" upper-triangularization of M_3 with this complete upper-triangularization of M_2 in order to obtain an upper-triangularization of M_3 itself. Making substitutions, we might conjecture that an upper-triangularization might look something like

$$\begin{bmatrix} \lambda_1 & \vec{?}^\top \\ \vec{0} & U_2^{-1} M_2 U_2 \end{bmatrix} = \begin{bmatrix} \lambda_1 & \vec{?}^\top \\ \vec{0} & U_2^\top R_2^\top M_3 R_2 U_2 \end{bmatrix}. \tag{40}$$

Notice that we don't really care about the values of the elements above the diagonal, since they don't affect whether or not our result is upper-triangular, so we just denote them as $?$.

Can we construct a change of basis U_3, in terms of U_2, to write M_3 in the above form? Well, observe that the above form can be further rewritten as

$$\begin{bmatrix} \lambda_1 & \vec{?}^\top \\ \vec{0} & (R_2 U_2)^\top M_3 (R_2 U_2) \end{bmatrix}. \tag{41}$$

In other words, it looks very much like how U_3 acted on M_3, except with $R_2 U_2$ instead of just R_2. Thus, based on what the above U_3 looked like, we can conjecture that the alternative change of basis

$$U_3 = \begin{bmatrix} \vec{v}_1 & R_2 U_2 \end{bmatrix} \tag{42}$$

will rewrite A in upper-triangular form. Let's check this out and see if it works. Recall that we constructed

$$U_2 = \begin{bmatrix} \vec{v}_2 & \vec{v}_3 \end{bmatrix}, \tag{43}$$

where $\vec{v}_2$ is an eigenvector of M_2 with eigenvalue λ_2 and $\vec{v}_3 \perp \vec{v}_2$. (Notice that we have incremented subscript indices to avoid ambiguity.) Thus, our new U_3 looks like

$$U_3 = \begin{bmatrix} \vec{v}_1 & R_2 U_2 \end{bmatrix} = \begin{bmatrix} \vec{v}_1 & R_2 \vec{v}_2 & R_2 \vec{v}_3 \end{bmatrix}. \tag{44}$$

We wish to compute $U_3^{-1} M_3 U_3$ and verify that it is upper-triangular. To do so, we need to compute U_3^{-1}.

How did we do this when we considered that $U_3 = \begin{bmatrix} \vec{v}_1 & R_2 \end{bmatrix}$? Well, we looked at each column, and showed (albeit briefly) that the set of columns is orthonormal, by appealing to the Gram-Schmidt construction. Since any orthonormal matrix U obeys $U^\top U = I$, we obtained $U^\top = U^{-1}$. So we were able to say $U_3^{-1} = U_3^\top$.

Let's do the same thing here. Observe that the second and third columns of U_3, $R_2\vec{v}_2$ and $R_2\vec{v}_3$, both lie in the column space of R_2, which by construction is orthogonal to the first column $\vec{v}_1$. Furthermore, we see that the inner product of the second and third columns is

$$\begin{aligned}(R_2\vec{v}_2)^\top(R_2\vec{v}_3) &= \vec{v}_2^\top(R_2^\top R_2)\vec{v}_3 && (45)\\ &= \vec{v}_2^\top I_2 \vec{v}_3 && (46)\\ &= \vec{v}_2^\top \vec{v}_3 && (47)\\ &= 0, && (48)\end{aligned}$$

relying on the fact that $R_2^\top R_2 = I_2$ as R_2 is orthonormal, and $\vec{v}_2^\top \vec{v}_3 = 0$ as the two vectors were constructed to be orthogonal. Thus, all the columns of U_3 are mutually orthogonal. To verify that they are of unit magnitude, we can simply compute their squared magnitude through inner products, where we see that

$$\begin{aligned}\vec{v}_1^\top \vec{v}_1 &= 1 && (49)\\ (R_2\vec{v}_2)^\top(R_2\vec{v}_2) = \vec{v}_2^\top R_2^\top R_2 \vec{v}_2 &= \vec{v}_2^\top \vec{v}_2 = 1 && (50)\\ (R_2\vec{v}_3)^\top(R_2\vec{v}_3) = \vec{v}_3^\top R_2^\top R_2 \vec{v}_3 &= \vec{v}_3^\top \vec{v}_3 = 1, && (51)\end{aligned}$$

since $\vec{v}_1$, $\vec{v}_2$, and $\vec{v}_3$ were all constructed to be of unit magnitude.

Therefore, we have shown that U_3 forms an orthonormal basis, so $U_3^{-1} = U_3^\top$. Thus, using similar techniques to what we did in the "partial" upper-triangularization, we can rewrite M_3 in this new basis as

$$\begin{aligned}U_3^{-1} M_3 U_3 &= U_3^\top M_3 U_3 && (52)\\ &= \begin{bmatrix} \vec{v}_1^\top \\ (R_2U_2)^\top \end{bmatrix} M_3 \begin{bmatrix} \vec{v}_1 & R_2U_2 \end{bmatrix} && (53)\\ &= \begin{bmatrix} \vec{v}_1^\top \\ (R_2U_2)^\top \end{bmatrix} \begin{bmatrix} \lambda_1\vec{v}_1 & M_3R_2U_2 \end{bmatrix} && (54)\\ &= \begin{bmatrix} \lambda_1 & \vec{?} \\ U_2^\top(R_2^\top\lambda_1\vec{v}_1) & U_2^\top R_2^\top M_3 R_2 U_2 \end{bmatrix} && (55)\\ &= \begin{bmatrix} \lambda_1 & \vec{?} \\ \vec{0} & U_2^\top M_2 U_2 \end{bmatrix} && (56)\\ &= \begin{bmatrix} \lambda_1 & ? & ? \\ 0 & \lambda_2 & ? \\ 0 & 0 & ? \end{bmatrix}, && (57)\end{aligned}$$

so we have successfully placed M_3 in upper-triangular form!

7 Induction

Let's take a quick step back. What have we done so far?

First, we developed a fairly intuitive change of basis that took an arbitrary $n \times n$ matrix to a "partial" upper-triangular form. Then, we saw that in the 2×2 case, this change of basis actually took our matrix to the final upper-triangular form. And now we've just seen that we can use our technique to upper-triangularize 2×2 matrices to upper-triangularize arbitrary 3×3 matrices as well.

What's next? Well, notice how our approach centers around resolving the case for $n \times n$ dimensions by solving the problem for $(n-1) \times (n-1)$ matrices. If we were writing a computer program to upper-triangularize matrices, we could have used recursion. To make sure our program is correct, we would like to make sure that, assuming we can upper-triangularize $(n-1) \times (n-1)$ matrices, we can upper-triangularize $n \times n$ matrices – that way, our recursive calls work as intended.

To formalize this, we will use a mathematical principle called *induction*, which will be covered far more extensively in other classes such as CS 70. Right now, we're just using it to make our recursion arguments mathematically precise.

The principle of induction says in summary that if we have some statement $P(n)$, we only need to prove two things to prove $P(n)$ is true for all n:

- $P(1)$ is true, or more generally $P(n_0)$ is true for some n_0 (*base case*).
- *If* $P(n-1)$ is true (*inductive hypothesis*) *then* $P(n)$ is true (*inductive step*).

Then for every $n \geq 1$ (or $n \geq n_0$), $P(n)$ is true. Note that in the second bullet, we first *assume* $P(n-1)$ is true and then show that $P(n)$ is true; in particular, we don't say anything about when $P(n-1)$ is false.

This explanation was a bit abstract, so let's connect it to our problem. We consider $P(n)$ to be the statement "every $n \times n$ matrix can be upper-triangularized". Saying $P(n)$ is true for all n is like saying "every $n \times n$ matrix can be upper-triangularized for all n", or equivalently "every square matrix can be upper-triangularized". And we need to prove that $P(1)$ is true (which we did; we showed, *very* briefly, that all 1×1 matrices are upper-triangular). The last thing we need to show is that if $P(n-1)$ is true then $P(n)$ is true. More explicitly, given that we can upper-triangularize $(n-1) \times (n-1)$ dimensional matrices, we want to show that we can upper-triangularize $n \times n$ matrices.

Consider an arbitrary $n \times n$ matrix A. We know that we can produce an unit eigenvector of A $\vec{v}_1$ with eigenvalue λ_1 that, along with the columns of the matrix R_{n-1} (produced using Gram-Schmidt), form an orthonormal basis of n-dimensional space.

In analogy with our approach for the case of $n = 3$, we then consider the $(n-1) \times (n-1)$ matrix

$$M_{n-1} = R_{n-1}^\top A R_{n-1} \tag{58}$$

and, using our inductive hypothesis, produce an orthonormal matrix U_{n-1} such that

$$U_{n-1}^{-1} M_{n-1} U_{n-1} \tag{59}$$

is upper-triangular.

Then, we let

$$U_n = \begin{bmatrix} \vec{v}_1 & R_{n-1} U_{n-1} \end{bmatrix}. \tag{60}$$

Before, to compute U_3^{-1}, we showed that U_3 was orthonormal by considering each pair of its columns. This approach is not quite going to work, since we have n columns, not just 3. Instead, we show that $U_n^\top U_n = I_n$, which implies that $U_n^\top = U_n^{-1}$. This equation is what we originally wanted. This also implies

that the columns of U_n are orthonormal, since the entries of $U_n^\top U_n$ (and in general any matrix) are the inner products of the respective columns of U_n.

This can be done as follows:

$$U_n^\top U_n = \begin{bmatrix} \vec{v}_1^\top \\ (R_{n-1}U_{n-1})^\top \end{bmatrix} \begin{bmatrix} \vec{v}_1 & R_{n-1}U_{n-1} \end{bmatrix} \tag{61}$$

$$= \begin{bmatrix} \vec{v}_1^\top \vec{v}_1 & \vec{v}_1^\top R_{n-1}U_{n-1} \\ U_{n-1}^\top R_{n-1}^\top \vec{v}_1 & U_{n-1}^\top R_{n-1}^\top R_{n-1}U_{n-1} \end{bmatrix}. \tag{62}$$

Let's look at each of the components of the above matrix. Since it was chosen to be a unit vector, $\vec{v}_1^\top \vec{v}_1 = 1$. As R_{n-1} was constructed using Gram-Schmidt to have its columns be orthogonal to $\vec{v}_1$, we have that $\vec{v}_1^\top R_{n-1} = \vec{0}^\top$ and that $R_{n-1}^\top \vec{v}_1 = \vec{0}$.

Looking at the bottom-right term, we recall that R_{n-1} was constructed to be an orthonormal matrix, so $R_{n-1}^\top R_{n-1} = I_{n-1}$. Furthermore, U_{n-1} was constructed to be an orthonormal basis for M_{n-1}, so $U_{n-1}^\top U_{n-1} = I_{n-1}$. Thus,

$$U_{n-1}^\top (R_{n-1}^\top R_{n-1}) U_{n-1} = U_{n-1}^\top I_{n-1} U_{n-1} = U_{n-1}^\top U_{n-1} = I_{n-1}, \tag{63}$$

canceling out the middle terms first. Putting all of this together, we see that

$$U_n^\top U_n = \begin{bmatrix} 1 & \vec{0}^\top \\ \vec{0} & I_{n-1} \end{bmatrix} = I_n, \tag{64}$$

so U_n is indeed orthonormal, as we expected.

Now that we have shown U_n is orthonormal, we can write $U_n^{-1} = U_n^\top$. Reexpressing A in this change of basis, we see that

$$U_n^{-1} A U_n = U_n^\top A U_n \tag{65}$$

$$= \begin{bmatrix} \vec{v}_1^\top \\ (R_{n-1}U_{n-1})^\top \end{bmatrix} A \begin{bmatrix} \vec{v}_1 & R_{n-1}U_{n-1} \end{bmatrix} \tag{66}$$

$$= \begin{bmatrix} \vec{v}_1^\top \\ (R_{n-1}U_{n-1})^\top \end{bmatrix} \begin{bmatrix} A\vec{v}_1 & AR_{n-1}U_{n-1} \end{bmatrix} \tag{67}$$

$$= \begin{bmatrix} \vec{v}_1^\top \\ (R_{n-1}U_{n-1})^\top \end{bmatrix} \begin{bmatrix} \lambda_1\vec{v}_1 & AR_{n-1}U_{n-1} \end{bmatrix} \tag{68}$$

$$= \begin{bmatrix} \lambda_1 \vec{v}_1^\top \vec{v}_1 & \vec{?} \\ U_{n-1}^\top R_{n-1}^\top \lambda_1 \vec{v}_1 & U_{n-1}^\top R_{n-1}^\top A R_{n-1} U_{n-1} \end{bmatrix} \tag{69}$$

$$= \begin{bmatrix} \lambda_1 & \vec{?} \\ \lambda_1 U_{n-1}^\top (R_{n-1}^\top \vec{v}_1) & U_{n-1}^\top R_{n-1}^\top A R_{n-1} U_{n-1} \end{bmatrix} \tag{70}$$

$$= \begin{bmatrix} \lambda_1 & \vec{?} \\ \vec{0} & U_{n-1}^\top M_{n-1} U_{n-1} \end{bmatrix}. \tag{71}$$

Since $U_{n-1}^\top M_{n-1} U_{n-1}$ is upper-triangular, so is $U_n^{-1} A U_n$, so we have successfully upper-triangularized an arbitrary $n \times n$ matrix A with an orthonormal change of basis by applying the inductive hypothesis. Thus

we completed the inductive step.

By induction, we can therefore upper-triangularize *arbitrary* square matrices using orthonormal changes of basis, which is what we had aimed to prove! Awesome!

If we were to write down this algorithm in its entirety, it would look something like the following.

Inputs

- An $n \times n$ matrix A.

Outputs

- An $n \times n$ orthonormal matrix U_n such that $U_n^\top A U_n$ is upper-triangular.

Upper-Triangularization Procedure

(a) If A is 1×1, return the 1×1 orthonormal matrix $U_n = \begin{bmatrix} 1 \end{bmatrix}$. (This is the *base case*.)

(b) If A is larger than 1×1:

i. Find an eigenvector-eigenvalue pair $\vec{v}_1, \lambda_1$ of A.

ii. Produce $n-1$ columns in $\mathbb{R}^n$ that are orthogonal to $\vec{v}_1$ by running the Gram-Schmidt algorithm (say, on the columns of $\begin{bmatrix} \vec{v}_1 & I_n \end{bmatrix}$). Put them as the columns of the matrix R_{n-1}.

iii. Set $M_{n-1} = R_{n-1}^\top A R_{n-1}$.

iv. Recursively upper-triangularize M_{n-1} to get U_{n-1} such that $U_{n-1}^\top M_{n-1} U_{n-1}$ is upper-triangular.

v. Set $U_n = \begin{bmatrix} \vec{v}_1 & R_{n-1} U_{n-1} \end{bmatrix}$ and return it.

8 Schur Decomposition

There's are still a couple loose ends to clear up, however. Recall that we had initially hoped for the elements along the main diagonal of T, the upper-triangularization of A, to be the eigenvalues of A. But though our construction made λ_1 an eigenvalue, the remaining λ_i were eigenvalues of different matrices, and we have not yet seen whether they are also eigenvalues of A itself.

To see that that is in fact the case, recall that we have just shown that we can write

$$A = UTU^{-1} = \begin{bmatrix} | & & | \\ \vec{v}_1 & \cdots & \vec{v}_n \\ | & & | \end{bmatrix} \begin{bmatrix} \lambda_1 & ? & ? & \cdots & ? \\ 0 & \lambda_2 & ? & \cdots & ? \\ 0 & 0 & \lambda_3 & \cdots & ? \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda_n \end{bmatrix} \begin{bmatrix} | & & | \\ \vec{v}_1 & \cdots & \vec{v}_n \\ | & & | \end{bmatrix}^{-1}, \tag{72}$$

where the λ_i are not necessarily the eigenvalues of A. Consider a particular λ_i. To show that it is an eigenvalue of A, we must show that $\det(A - \lambda_i I) = 0$. Recalling that $\det(AB) = \det(A)\det(B)$, we have that

$$\det(A - \lambda_i I) = \det\Big(UTU^{-1} - \lambda_i U U^{-1}\Big) \tag{73}$$

$$= \det\Big(U(T - \lambda_i I)U^{-1}\Big) \tag{74}$$

$$= \det(U) \cdot \det(T - \lambda_i I) \cdot \det\Big(U^{-1}\Big). \tag{75}$$

Let's look at each of the elements of this product individually. First, observe that

$$\det(U) \cdot \det\Big(U^{-1}\Big) = \det\Big(UU^{-1}\Big) = \det(I) = 1, \tag{76}$$

so we can cancel the determinant of U with the determinant of its inverse, to write

$$\det(A - \lambda_i I) = \det(T - \lambda_i I). \tag{77}$$

In other words, we have shown that the characteristic polynomial of A remained unchanged under a change of basis. Now, observe that

$$T - \lambda_i I = \begin{bmatrix} \lambda_1 - \lambda_i & ? & ? & \cdots & ? \\ 0 & \lambda_2 - \lambda_i & ? & \cdots & ? \\ 0 & 0 & \lambda_3 - \lambda_i & \cdots & ? \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda_n - \lambda_i \end{bmatrix}. \tag{78}$$

Thus, the i^{th} pivot element of the above matrix must be 0, since it will equal $\lambda_i - \lambda_i = 0$. If we have an upper-triangular matrix with only $n - 1$ nonzero pivots, then it has linearly dependent columns, so its determinant is zero. Thus,

$$\det(T - \lambda_i I) = 0 \implies \det(A - \lambda_i I) = 0, \tag{79}$$

so each λ_i is an eigenvalue of A, as expected! This way of representing A is known as the *Schur Decomposition* of A.

9 Complex Inner Products

There's one subtlety that we skipped over in the above proof. Specifically, we assumed throughout that the notions of orthogonality and inner products were defined on our vector spaces. But how do you take the inner product of two complex-valued vectors? If you try to reuse the definition of the dot product, you'll get some weird results that cause our understanding of these concepts to break down. For instance,

$$\left\| \begin{bmatrix} \mathrm{j} \\ 1 \end{bmatrix} \right\|^2 = \begin{bmatrix} \mathrm{j} \\ 1 \end{bmatrix} \cdot \begin{bmatrix} \mathrm{j} \\ 1 \end{bmatrix} = -1 + 1 = 0, \tag{80}$$

which doesn't seem to make sense, since only the zero vector should have a norm of zero.

For now, we should assume that we are working in the vector space of reals $\mathbb{R}^n$ using the standard definition of the dot product. This, however, requires all the λ_i to be real, which may not always be the case. For now, we will make that assumption, though in Note 2j we will see a small generalization of the dot product which will ensure our above result is true in all cases. In any case, the adjustment to our proof is minimal, given this more general notion of the inner product.

10 The Spectral Theorem

We will now use our decomposition to obtain some interesting results about diagonalizing real, symmetric matrices.

First, we claim that the eigenvalues of a real, symmetric matrix are all themselves real. To prove it, let's consider a real, symmetric $n \times n$ matrix $A = A^\top$ and an eigenvalue λ of A. By the definition of eigenvectors, there exists some nonzero vector $\vec{x}$ such that

$$A\vec{x} = \lambda\vec{x}. \tag{81}$$

To show λ is real, we'd like to get a result that looks like $\lambda = \overline{\lambda}$. So, striking out blindly, we can take the conjugate to get $\overline{\lambda}$ involved somehow, to obtain

$$A\overline{\vec{x}} = \overline{\lambda\vec{x}}. \tag{82}$$

Notice that $A = \overline{A}$, since we assumed A was real. Now, let's try to take advantage of A's symmetric nature, by taking the transpose and using the fact that $A = A^\top$, to obtain

$$\overline{\vec{x}}^\top A = \overline{\vec{x}}^\top \overline{\lambda}. \tag{83}$$

At this point we can post-multiply both sides by $\vec{x}$ to obtain

$$\overline{\vec{x}}^\top A\vec{x} = \overline{\lambda\vec{x}}^\top \vec{x} \tag{84}$$

$$\implies \lambda\overline{\vec{x}}^\top \vec{x} = \overline{\lambda\vec{x}}^\top \vec{x} \tag{85}$$

$$\implies (\lambda - \overline{\lambda})\overline{\vec{x}}^\top \vec{x} = 0. \tag{86}$$

So by basic arithmetic, either $\lambda = \overline{\lambda}$, or $\overline{\vec{x}}^\top \vec{x} = 0$. But since we chose $\vec{x}$ to be a nonzero vector, the former equality must be the one that is true, so λ is real, as desired.

Now, we will make a stronger claim. We assert that, not only are all the eigenvalues of A real, but that A can be *diagonalized*, meaning that it has n linearly independent eigenvectors. Furthermore, we claim that these eigenvectors can be chosen such that they are all orthonormal.

What is our goal? We wish to show that we can express

$$A = Q\Lambda Q^\top, \tag{87}$$

where Q is an orthonormal matrix whose columns are the eigenvectors of A, and Λ is a diagonal matrix containing the eigenvalues of A.

Notice that, since Λ is a diagonal matrix, it is also upper-triangular, with the elements along its diagonal being the eigenvalues of A. Thus, our desired diagonalization of A is also a Schur decomposition of A.

So the first question to ask should be: can we construct a Schur decomposition of an arbitrary real, symmetric matrix A? The critical assumption needed when doing so was that all the λ_i were real produced during the induction, as otherwise our arguments related to orthogonality broke down. Let's look at our procedure and try to show that this is the case.

First, consider λ_1. λ_1 was chosen to be an eigenvalue of A. But since all the eigenvalues of A are real (since A is symmetric, using the result from above), we know that λ_1 is real. Great!

Next, let's look at λ_2. Looking at the induction, λ_2 was chosen to be an eigenvalue of $M_{n-1} = R_{n-1}^\top A R_{n-1}$, where R_{n-1} was an $n \times (n-1)$ orthonormal matrix constructed in a particular fashion. Right now, the exact construction of R_{n-1} isn't super important. What *is* important is that, as $A = A^\top$,

$$M_{n-1}^\top = (R_{n-1}^\top A R_{n-1})^\top = R_{n-1}^\top A^\top R_{n-1} = R_{n-1}^\top A R_{n-1} = M_{n-1}, \tag{88}$$

so M_{n-1} is symmetric. And so, as λ_2 is an eigenvalue of the symmetric matrix M_{n-1}, it is itself real. This looks promising!

In a similar manner, λ_3 is an eigenvalue of $M_{n-2} = R_{n-2}^\top M_{n-1} R_{n-2}$, which is symmetric, so λ_3 is itself real. And this recursive argument can be continued in a similar fashion to show that *all* the λ_i are real! Awesome[2]!

Since all the λ_i are real, the induction involved in the Schur form proof works out, so we can write

$$A = UTU^\top, \tag{89}$$

where U is an orthonormal matrix and T is upper-triangular. Our goal is to show that $T = \Lambda$ - in other words, that T is in fact diagonal! Since we know that T is already upper-triangular, one way of doing this would be to show that

$$T = T^\top, \tag{90}$$

so the "upper" part of T consists entirely of zeros as well, so it is diagonal. How can we get $T^\top$ from our upper-triangular decomposition? We might as well just blindly take the transpose of the entire equation, just so we get the desired term *somewhere*. Doing so, we obtain

$$A^\top = UT^\top U^\top. \tag{91}$$

But since A is symmetric, $A = A^\top$, so we can write

$$A = UT^\top U^\top. \tag{92}$$

So we have shown that, when working in the basis of U, A becomes both T and $T^\top$? How can this be true? The only way for this to be possible is if

$$T = T^\top, \tag{93}$$

as desired! Thus, we have shown that we can write

$$A = UTU^\top, \tag{94}$$

where T is a diagonal matrix made up of A's eigenvalues, and U is an orthonormal matrix. So we have diagonalized A! This completes the proof of what is known as the *real spectral theorem*.

11 Connection to Stability

Now that we have a connection between upper triangularization and diagonalization, we can start expanding on previous concepts and algorithms. Chiefly, we used diagonalization to get to a diagonal system of scalar equations we could easily solve, before converting back to the original basis. For a lot of these problems, we can now also use upper triangularization to get to a solvable system in the case that the matrices we are

[2] We could alternatively frame this argument using induction, like we did when deriving Schur form, if you're more comfortable with that. But both approaches are fundamentally equivalent and lead to the same result.

given are not diagonalizable.

One derivation where this is important is the theorem that for a discrete-time system

$$\vec{x}[i+1] = A\vec{x}[i] + B\vec{u}[i] + \vec{w}[i], \tag{95}$$

the system is stable if and only if all eigenvalues λ of A have $|\lambda| < 1$.

We proved this theorem in Note 10 in the case where A is diagonalizable. Now let us suppose A is not diagonalizable. But A is a square matrix, and hence it is upper-triangularizable. Let $A = UTU^\top$ be an upper-triangularization of A. Rearranging and remembering that $U^\top = U^{-1}$, we write

$$\begin{aligned}
\vec{x}[i+1] &= A\vec{x}[i] + B\vec{u}[i] + \vec{w}[i] && (96)\\
&= UTU^\top\vec{x}[i] + B\vec{u}[i] + \vec{w}[i] && (97)\\
\implies U^\top\vec{x}[i+1] &= TU^\top\vec{x}[i] + U^\top B\vec{u}[i] + U^\top\vec{w}[i] && (98)\\
\implies \vec{\widehat{x}}[i+1] &= T\vec{\widehat{x}}[i] + \widehat{B}\vec{u}[i] + \vec{\widehat{w}}[i]. && (99)
\end{aligned}$$

where $\vec{\widehat{x}} = U^\top\vec{x}$, $\widehat{B} = U^\top B$, and $\vec{\widehat{w}} = U^\top\vec{w}$. What does this system look like? Writing it out in matrix form,

$$\begin{bmatrix} \widehat{x}_1[i+1] \\ \vdots \\ \widehat{x}_n[i+1] \end{bmatrix} = \begin{bmatrix} \lambda_1 & ? & \cdots & ? \\ 0 & \lambda_2 & \cdots & ? \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} \begin{bmatrix} \widehat{x}_1[i] \\ \vdots \\ \widehat{x}_n[i] \end{bmatrix} + \begin{bmatrix} (\widehat{B}\vec{u}[i])_1 \\ \vdots \\ (\widehat{B}\vec{u}[i])_n \end{bmatrix} + \begin{bmatrix} \widehat{w}_1[i] \\ \vdots \\ \widehat{w}_n[i] \end{bmatrix}. \tag{100}$$

Looking at the last row, we see an equation that we already know how to analyze:

$$\widehat{x}_n[i+1] = \lambda_n\widehat{x}_n[i] + (\widehat{B}\vec{u}[i])_n + \widehat{w}_n[i]. \tag{101}$$

Thus, this one-dimensional system is BIBO stable if and only if $|\lambda_n| < 1$.

Now looking at the penultimate row, we get another equation:

$$\begin{aligned}
\widehat{x}_{n-1}[i+1] &= \lambda_{n-1}\widehat{x}_{n-1}[i] + (?)\cdot\widehat{x}_n[i] + (\widehat{B}\vec{u}[i])_{n-1} + \widehat{w}_{n-1}[i] && (102)\\
&= \lambda_{n-1}\widehat{x}_{n-1}[i] + (\widehat{B}\vec{u}[i])_{n-1} + (\widehat{w}_{n-1}[i] + (?)\cdot\widehat{x}_n[i]). && (103)
\end{aligned}$$

Why did we group the terms like this? Well, it's because we already know the value of $|\widehat{x}_n[i]|$. So we can wrap $\widehat{x}_n[i]$ into our disturbance and still know that the whole term is bounded, provided $|\lambda_n| < 1$. Then this one-dimensional system is BIBO stable if and only if $|\lambda_{n-1}| < 1$.

Now that we've done a couple "base cases", let's tackle the general case. Suppose that $|\lambda_n| < 1, |\lambda_{n-1}| < 1, \ldots, |\lambda_{n-k+1}| < 1$ for some $k < n$. Looking at the $(n-k)^{\text{th}}$ row, we see that

$$\begin{aligned}
\widehat{x}_{n-1}[i+1] &= \lambda_{n-k}\widehat{x}_{n-k}[i] + (\widehat{B}\vec{u}[i])_{n-k} + \widehat{w}_{n-k}[i] && (104)\\
&\quad + (?)\cdot\widehat{x}_{n-k+1}[i] + (?)\cdot\widehat{x}_{n-k+2}[i] + \cdots + (?)\cdot\widehat{x}_n[i] && (105)\\
&= \lambda_{n-k}\widehat{x}_{n-k}[i] + (\widehat{B}\vec{u}[i])_{n-k} + \left(\widehat{w}_{n-k}[i] + \sum_{j=0}^{k-1}(?)\cdot\widehat{x}_{n-j}[i]\right) && (106)
\end{aligned}$$

where each of the question marks correspond to different constants – namely, the nonzero entries in the $(n-k)^{\text{th}}$ row of T. Since $|\lambda_n| < 1, |\lambda_{n-1}| < 1, \ldots, |\lambda_{n-k+1}| < 1$, each of $\widehat{x}_n, \ldots, \widehat{x}_{n-k+1}$ is bounded so

long as $\widehat{w}$ is bounded. Thus the "aggregate disturbance" term is bounded so long as $\vec{\widehat{w}}$ is bounded. Thus this (still one-dimensional) system for $\widehat{x}_{n-k}$ is BIBO stable if and only if $|\lambda_{n-k}| < 1$.

Continuing on with this process until $k = n - 1$, we see that the implication "if $\vec{\widehat{w}}$ is bounded then $\vec{\widehat{x}}$ is bounded" holds if and only if $|\lambda_1| < 1, \ldots, |\lambda_n| < 1$. This means that our $\vec{\widehat{x}}$ system is BIBO stable if and only if $|\lambda_1| < 1, \ldots, |\lambda_n| < 1$.

To finish off, we can use the proof from Note 10 to assert that $\vec{x}$ is bounded if and only if $V\vec{x}$ is bounded, for all sequences $\vec{x}$ and matrices V. Applying this to $\vec{\widehat{x}} = U^\top \vec{x}$ and $\vec{\widehat{w}} = U^\top \vec{w}$, we see that "if $\vec{w}$ is bounded then $\vec{x}$ is bounded" holds if and only if $|\lambda_1| < 1, \ldots, |\lambda_n| < 1$. This means that our $\vec{x}$ system is BIBO stable if and only if $|\lambda_1| < 1, \ldots, |\lambda_n| < 1$.

This is a fully general proof of necessary/sufficient conditions for BIBO stability in discrete time!

Concept Check: Formalize the above argument into an inductive proof. That is, identify the inductive claim and variable to induct on, the base case, and the inductive step. You may want to re-write the proof so that it more closely mirrors an inductive format.

Hint: Your claim could be something like "for a given k, $\vec{\widehat{x}}_{n-k}$ is bounded if and only if $|\lambda_n| < 1, |\lambda_{n-1}| < 1$, $\ldots, |\lambda_{n-k}| < 1$".

12 [Optional] Applications in Precision Computing

Sometimes when we are solving problems or designing algorithms (say for stability or controllability), we want to do coordinate changes to bases in which we get easily solvable systems.

Previously, we used eigenvector bases for this. Now, we are allowed to use upper-triangularization bases. It turns out that a lot of the time, using the upper-triangularization basis is better, in the sense that our computation is more numerically stable. Let's try to briefly unpack why that is.

For notation's sake, let's say that we're working with a matrix $A \in \mathbb{R}^{n \times n}$.

Right off the bat, we re-emphasize that sometimes A is not diagonalizable. In this case, upper-triangularization is the only method we have developed so far that actually works for A, and so it is the best method to use by default.

Now let's suppose that A is diagonalizable. Let $A = V\Lambda V^{-1}$ be the representation of A in the eigenvector basis V – the diagonalization of A – and let $A = UTU^\top$ be the representation of A in the upper triangular basis – the upper-triangularization of A.

One key difference we can see in these formulas is that in the diagonalization representation, we need to compute V and V^{-1}, while in the upper triangularization representation, we only need to compute U and $U^\top$ (which is really easy to compute given U). This is indeed the difference we are looking for.

We will now try to justify why computing V^{-1} is numerically instable. We do this by considering a range of scenarios (say configurations of A).

One extreme is when A is symmetric. Then the eigenvectors of A are orthonormal. If they are normalized, then V is a matrix with orthonormal columns and rows, so that $V^{-1} = V^\top$. In this case, we have shown earlier in the note that the upper triangularization is exactly equal to the diagonalization, so there is no advantage to be gained by either side in terms of numeric stability.

The other extreme is when A is not diagonalizable. Then multiple eigenvectors of A are not distinct – in particular, they align perfectly. In this case V is singular, and thus non-invertible, since two of the columns of V are identical. Since we can only do upper-triangularization, this is the better method.

We can make our point by considering matrices A that are "close to non-diagonalizable" – that is, A with eigenvector matrix V which is "nearly singular". That is, two eigenvectors of A are almost completely aligned. We saw such matrices in the critically damped case for RLC circuits, for example.

In this case, inverting the V matrix and separating the aligned eigenvectors is difficult, and indeed numerically unstable. Why is this the case? One can make this argument precise using the notion of *condition number*, which is out of scope for the class. But, heuristically, here is what happens.

When we find a matrix inverse, conceptually it's similar to finding a solution $\vec{x}$ to $V\vec{x} = \vec{y}$. Since there are almost-aligned eigenvectors in V, there is at least one direction in $\mathbb{R}^n$ for which all of the columns of V have really small components in that direction (because there are n directions and effectively $n-1$ vectors to use). If $\vec{y}$ points into that problematic direction, then the coordinates of $\vec{x}$ (i.e., the coefficients of the linear combination of the columns of V) will, more often than not, have to be very large to push the vectors in V to reach $\vec{y}$, while cancelling out in all other directions to perfectly equal $\vec{y}$ – even for benign, generic $\vec{y}$ such as unit vectors! Moreover, *very similar values of $\vec{y}$ (such as a "true" value of $\vec{y}$ compared to a computer representation of $\vec{y}$), lead to very different values for $\vec{x}$!* Since $\vec{x}$ has crazy behavior and $\vec{x} = V^{-1}\vec{y}$, it is reasonable to think of V^{-1} as being numerically unstable.

On the other hand, upper-triangularization is like a boon, in the sense that when we compute it, we never have to take matrix inverses. All we have to do is take matrix transposes, use Gram-Schmidt[3], and we're in business – we get a fully orthonormal basis that turns our system into one that's easily solvable. This process is more numerically stable, since our change-of-basis is orthonormal and we never have to take an inverse anywhere.

This is the crux of why upper-triangularization is more numerically stable than diagonalization. This type of analysis can be explored more in e.g., EE 127, and Math 128.

Contributors:

- Druv Pai.
- Rahul Arya.
- Anant Sahai.

[3]Gram-Schmidt is also not great as a numerical linear algebra tool, because it suffers from a phenomenon called *catastrophic cancellation*. The gist of it is that we end up subtracting a lot of vectors, end up with vectors that should be – but are not quite, on our computer, due to computer arithmetic limitations – zero, and then we normalize it and get a unit vector in an essentially random direction. Using this vector in our computation can lead to crazy results. There are other methods to do orthonormalization, such as one of many algorithms for the QR decomposition, although they are more technically complex. All of this footnote is out of scope for the class.

