Notes on Linear Algebra for Quantum Mechanics

By Jonathan Sievers, McGill University

Quantum mechanics obeys the rules of linear algebra. It didn't necessarily have to be this way, and when Heisenberg first developed what is now known as matrix mechanics, he didn't realize the operators he had written down were matrices. In this note, we will try to lay out some of the basics of linear algebra relevant to quantum mechanics.

1. Matrix Multiplication

The distinctive nature of linear algebra comes from how we multiply two matrices. If we have

$$\mathrm{C} = \mathrm{AB} \tag{1}$$

and matrices A and B have sizes n, k and k, m, respectively, then

$$C_{ij} = \sum_k A_{ik} B_{kj} \tag{2}$$

For matrix multiplication, the second dimension of A has to be the same as the first dimension of B (these are often referred to as the "interior dimensions"). Except in special cases, matrix multiplication is not commutative $\mathrm{AB} \neq \mathrm{BA}$. It is, however, associative: $\mathrm{ABC} = (\mathrm{AB})C = \mathrm{A}(\mathrm{BC})$. Addition works like you'd expect. If $\mathrm{C} = \mathrm{A} + \mathrm{B}$ then $C_{ij} = A_{ij} + B_{ij}$, and A and B must have the same sizes. Multiplication by a scalar also works as you'd expect. For scalar α, $\mathrm{C} = \alpha\mathrm{A}$ has $C_{ij} = \alpha A_{ij}$. Finally, the preceeding properties also guarantee that matrix math is distributive: $\mathrm{A}(\mathrm{B} + \mathrm{C}) = \mathrm{AB} + \mathrm{AC}$.

Matrix-vector and vector-vector operations are just special cases of matrix-multiplication where one (or more) of the dimensions is one. If B is a vector (which we'll write b to make it clear it's a vector), then m is 1, and we can drop the j from Equation 2 since we know it is always 1[1]. In that case we have:

$$c_i = \sum_k A_{ik} bk \tag{3}$$

Similarly, we can have A be a vector, in which case $n = 1$, we'll refer to it as a, and we have:

$$c_j = \sum_k a_k \mathrm{B}_k j \tag{4}$$

Standardly, we refer to $n \times 1$ vectors (*i.e.* a) as row vectors, and $1 \times m$ vectors (*i.e.* b) as column vectors. By convention, if you just say "vector", you usually mean a column vector.

An even more special case of matrix multiplication is when both matrices are vectors. Since the interior dimensions need to line up, we can either have a row vector times a column vector

[1] Or 0, depending on if you were raised in a C-like or FORTRAN-like environment

which produces a scalar (or, depending on your situation, possibly a 1×1 matrix). This is called the *inner* or *dot* product. We can also have a column vector times a row vector, in which case our output is a matrix. This is called the *outer* product. In fact, if you go back and look at the definition of matrix multiplication in Equation 2, you can see that you can just sum the outer product of every column of A with the corresponding row of B to get C. Equivalently, we can also say that every entry in C is the dot product of the corresponding row of A and column of B. When you think about matrix multiplication as a set of outer products, notice that we could re-order the columns of A, and as long as we re-order the columns of B in the same way, the product AB is unchanged.

It is useful to remember that two vectors must have the same dimensions if you want the inner product, but there is no such requirement for outer products. Any old column and row vector will happily make an outer product. It will be square if they're the same dimensions, but it doesn't have to be.

2. Transposes

Every matrix A has a unique transpose A^T with:

$$A^T_{ji} = A_{ij} \tag{5}$$

If you take the tranpose of the transpose of a matrix, you end up with the original matrix again.

A key identity, which again just drops out of the definition of matrix multiplication, is that

$$(\mathrm{AB})^T = \mathrm{B}^T\mathrm{A}^T \tag{6}$$

$$(\mathrm{ABC})^T = \mathrm{C}^T\mathrm{B}^T\mathrm{A}^T \tag{7}$$

and so on. If we have column vectors a and b, then the inner product is:

$$a \cdot b = a^T b = b^T a \tag{8}$$

Since the vectors have to have the same length to take an inner product, it doesn't matter which order you put them in. The outer product of a with b is ab^T, and now it very much does matter which order you use:

$$\left(ab^T\right)^T = \left(b^T\right)^T a^T = ba^T \tag{9}$$

In words, the outer product of a^T and b is the transpose of the outer product of b^T and a.

The distance d between two points in Cartesian geometry is $d^2 = dx^2 + dy^2 + dz^2$. In higher dimensions, you just keep adding the square of the distances. If you had a vector containing the separation between two points, then the inner product gives you exactly that squared distance. If $d = [dx, dy, dz, ...]$ then:

$$d^T d = \sum_k d_k d_k = dx^2 + dy^2 + dz^2 + ... \tag{10}$$

When we talk about the "length" of a vector, this is usually what we mean[2].

3. The Identity Matrix and Matrix Inverses

There is an identity matrix I, which has ones on the diagonal and zeros everywhere else:

$$I_{ij} = \delta_{ij} \tag{11}$$

where δ_{ij} is the Kronecker δ function: one if $i == j$ and zero otherwise. For suitably sizes identity matrices, we have:

$$\mathrm{A} = \mathrm{AI} \tag{12}$$
$$\mathrm{A} = \mathrm{IA} \tag{13}$$

which you can show by, once again, going back to the definition of matrix multiplication. While we often talk about "the" identity matrix, there are an arbitrarily large number of them of different sizes. The size will be determined by how it is being used - if A is $n \times m$, then in IA, I must be $n \times n$ while in AI, I must be $m \times m$.

"Most" square matrices have an inverse A^{-1}, defined to be

$$\mathrm{AA}^{-1} = \mathrm{I} \tag{14}$$

In principle, we could have separate left and right inverses:

$$\mathrm{AA}_r^{-1} = \mathrm{I} \tag{15}$$
$$\mathrm{A}_l^{-1}\mathrm{A} = \mathrm{I} \tag{16}$$

However, we can take advantage of the fact that matrix multiplication is associative to show that the left and right inverses are the same. We can show this by looking at the product $\mathrm{A}_l^{-1}\mathrm{AA}_r^{-1}$:

$$\mathrm{A}_l^{-1}\mathrm{AA}_r^{-1} = \left(\mathrm{A}_l^{-1}\mathrm{A}\right)\mathrm{A}_r^{-1} = \mathrm{IA}_r^{-1} = \mathrm{A}_r^{-1} \tag{17}$$
$$\mathrm{A}_l^{-1}\mathrm{AA}_r^{-1} = \mathrm{A}_l^{-1}\left(\mathrm{AA}_r^{-1}\right) = \mathrm{A}_l^{-1}\mathrm{I} = \mathrm{A}_l^{-1} \tag{18}$$

So the left and right inverses have to be the same, and we can just talk about "the" inverse. Two matrices are said to commute if AB = BA, and we have now shown that a matrix always commutes with its inverse (if it exists): $\mathrm{AA}^{-1} = \mathrm{A}^{-1}\mathrm{A} = \mathrm{I}$ [3]. One corollary of this result is that the dot product of the i^{th} row of A against the j^{th} column of A^{-1} is zero unless $i = j$, but also the dot product of the j^{th} *row* of A against the i^{th} *column* of A^{-1} is also 0 unless $i = j$. This follows naturally from interpreting matrix multiplication as the dot product of each row of the first matrix against each row of the second matrix.

[2] More generally, one can talk about different norms of a vector. This is the L^2-norm, which is by far the most commonly used one in physics.

[3] Our proof only relied on the existence of an identity operator and associativity, so in general an object will commute with its inverse (if it exists) in any group, even if the operator is not commutative. Matrix multiplication is just a special case of this more general result.

4. Eigenvalues and Eigenvectors

A particularly useful way of looking at matrices is through their *eigenvalues* and *eigenvectors*. Normally, they are introduced by solving the following for vector v and scalar λ for some matrix A:

$$\mathrm{A}v = \lambda v \tag{19}$$

Instead, we'll just assume that we can write A as follows:

$$\mathrm{A} = \mathrm{V}\Lambda\mathrm{V}^{-1} \tag{20}$$

for some matrix V and diagonal matrix Λ. It isn't obvious we could do this, but it turns out that in quantum mechanics, we actually can do this for every matrix where we'd want to[4]. Of course, it's not particularly surprising we can factor a matrix this way - if A is $n \times n$, then there are n^2 entries in A, n^2 entries in V, and n entries in Λ, so we have $n^2 + n$ numbers we can pick to match the n numbers in A. That's more flexibility than we really want, so we'll decree every column of V has length 1, *i.e.* $v_i^T v_i = 1$ if v_i is the i^{th} column of V. We get to pick the first $n-1$ entries this way, but the last entry has to make the length 1, so we only get to pick $n-1$ numbers. Since we have $n-1$ numbers per column, n columns, and n numbers in Λ, we have a total of n^2 numbers we get to pick, to match the n^2 numbers in A.

We won't go into how to calculate the eigenvalues/eigenvectors of a matrix - that is, in general, a very hard problem, and people devote their careers to improving eigenvalue codes. Unless you, too, wish to devote your career, we suggest you use someone else's routines (*e.g.* numpy.linalg.eig/numpy.linalg.eigh). We will, however, show what you can do once you have the eigenvalues/eigenvectors. Again, let v_i be the i^{th} eigenvector. Then:

$$\mathrm{V}^{-1}v_i = [000...010...00] \tag{21}$$

where the non-zero entry is in the i^{th} position. A diagonal matrix times a vector just rescales each element of the vector by the corresponding diagonal element, so

$$\Lambda\mathrm{V}^{-1}v_i = [000...0\lambda_i 0...00] \tag{22}$$

Finally, multiplying a matrix by a vector with one non-zero element just rescales the corresponding column of the matrix by the value of that element. This gives:

$$\mathrm{A}v_i = \mathrm{V}\Lambda\mathrm{V}^{-1}v_i = \mathrm{V}[000...0\lambda_i 0...0] = \lambda_i v_i \tag{23}$$

This is one of the most fundamental results in linear algebra. If we multiply a matrix by an eigenvector, we get the eigenvector scaled by the corresponding eigenvalue back. Normally, if we multiply a matrix by a vector, we get a new vector pointing in a different direction, *unless that vector is an eigenvector*. If the vector is an eigenvector, it continues to point in the same direction, hence the name[5].

[4] Every symmetric/Hermitian matrix has can be factored this way. Some non-symmetric matrices can't. The official name for these matrices is *defective* (seriously).

[5] Eigen is German for "own".

One immediate consequence of Equation 23 is that we can trivially carry out repeated multiplications of a matrix times an eigenvector. We have

$$\mathrm{A}^2 v_i = \mathrm{A}(\mathrm{A}v_i) = \mathrm{A}\lambda_i v_i = \lambda_i^2 v_i \tag{24}$$

$$\mathrm{A}^n v_i = \lambda_i^n v_i \tag{25}$$

If we have a vector that is the sum of two eigenvalues $x = \alpha_i v_i + \alpha_j v_j$ then we have:

$$\mathrm{A}x = \alpha_i \mathrm{A}v_i + \alpha_j \mathrm{A}v_j = \alpha_i \lambda_i v_i + \alpha_j \lambda_j v_j \tag{26}$$

In words, we stretch each eigenvector out by its eigenvalue and sum them together. In general, this will point in a different direction from the original vector, *unless* the eigenvalues are the same. Similarly,

$$\mathrm{A}^n x = \alpha_i \lambda_i^n v_i + \alpha_j \lambda_j^n v_j \tag{27}$$

Now what happens if we have an arbitrary vector x? We'd like to be able to express it as a sum of eigenvectors, *i.e.* $x = \sum_i \alpha_i v_i$, but that's just $x = \mathrm{V}b$ for some vector b. The entries of b tell us how much of each eigenvector is in x. Fortunately, we know exactly how to work out b:

$$\mathrm{V}b = x \tag{28}$$

$$\mathrm{V}^{-1}\mathrm{V}b = \mathrm{V}^{-1}x \tag{29}$$

$$\mathrm{I}b = \mathrm{V}^{-1}x \tag{30}$$

$$b = \mathrm{V}^{-1}x \tag{31}$$

We can express x in terms a sum of eigenvectors just by taking $\mathrm{V}^{-1}x$. Then we can multiply by an arbitrary power of A:

$$\mathrm{A}^n x = \mathrm{V}\Lambda^n \mathrm{V}^{-1} x \tag{32}$$

This expression will be key in quantum mechanics when we work out time evolution. We can, for instance, define $e^{\mathrm{A}}x$ as $(\mathrm{A} + \mathrm{A}^2/2 + \mathrm{A}^3/3! + ...)x$. If x is an eigenvector v_i, that gives us

$$\lambda_i v_i + \lambda_i^2/2 v_i + \lambda_i^3/3! v_i + ... \tag{33}$$

$$= \left(\lambda_i + \lambda_i^2/2 + \lambda_i^3/3 + ...\right) v_i \tag{34}$$

$$= e^{\lambda_i} v_i \tag{35}$$

If we had many eigenvectors in our vector, we can take advantage of this to write down:

$$e^{\mathrm{A}}x = \mathrm{V}e^{\Lambda}\mathrm{V}^{-1}x \tag{36}$$

Now that we have some experience with eigenvalues/eigenvectors, we can use this familiarity to see which matrices might not have inverses. Assuming the eigendecomposition exists, then if we have a zero eigenvalue,

$$\mathrm{A}v = 0v = 0 \tag{37}$$

so we'll get zero when we multiply our matrix by the corresponding eigenvector. Since a matrix times a vector of zeros is always zero, there's no way to write down a matrix inverse that will give us our starting vector back. Normally, $\mathrm{A}^{-1}\mathrm{A}x = x$, but for an eigenvector with zero eigenvalue we have

$$\mathrm{A}^{-1}\mathrm{A}v_i = \mathrm{A}^{-1}\left(\mathrm{A}v_i\right) = \mathrm{A}^{-1}0 = 0 \tag{38}$$

Any matrix that has at least one zero eigenvalue then cannot have an inverse. As far as linear algebra is concerned, there's nothing particularly special about a zero eigenvalues, since if we take the matrix $\mathrm{A} + \alpha\mathrm{I}$ then:

$$\left(\mathrm{A} + \alpha\mathrm{I}\right) v_i = \lambda_i v_i + \alpha v_i = \left(\lambda_i + \alpha\right) v_i \tag{39}$$

In words, if we add α times the identity matrix, we just shift all the eigenvalues by α and the eigenvectors remain unchanged. If we set $\alpha = -\lambda_i$, then the matrix $\mathrm{A} - \lambda_i\mathrm{I}$ will now have a 0 for the i^{th} eigenvalue, and will no longer have an inverse.

We'll note here that the eigenvalues of a diagonal matrix are just the diagonal entries and the eigenvector matrix is just the identity since we can always write:

$$\Lambda = \mathrm{I}\Lambda\mathrm{I}^{-1} \tag{40}$$

for arbitrary diagonal matrix Λ, and Equation 40 is just Equation 23 with $\mathrm{V} = \mathrm{I}$.

4.1. Symmetric Matrices

An important sub-class of matrices are *symmetric* matrices, *i.e.* $A_{ij} = A_{ji}$ or $\mathrm{A} = \mathrm{A}^T$. Symmetric matrices arise over and over in physics, so we'll look at some of their properties here. First, note that

$$\left(\mathrm{A}v_i\right)^T = v_i^T\mathrm{A}^T = v_i^T\mathrm{A} = \lambda_i v_i^T \tag{41}$$

In words, if we have an eigenvector of A, then if we put that eigenvector on the left side of A, it remains an eigenvector. Now let's take two eigenvectors and form $v_j^T\mathrm{A}v_i$. That gives:

$$v_j^T\mathrm{A}v_i = \left(v_j^T\mathrm{A}\right) v_i = \lambda_j v_j^T v_i \tag{42}$$

$$v_j^T\mathrm{A}v_i = v_j^T\left(\mathrm{A}v_i\right) = \lambda_i v_j^T v_i \tag{43}$$

Since the two expressions have to be equal, we have

$$lambda_i v_j^T v_i = \lambda_j v_j^T v_i \tag{44}$$

$$\left(\lambda_i - \lambda_j\right) v_j^T v_i = 0 \tag{45}$$

If the two eigenvectors have different eigenvalues, then the only way Equation 45 can be true is if $v_j^T v_i = 0$. The eigenvectors must be orthogonal. If the eigenvalues are all distinct, then this has to be true for every pair of eigenvectors, so we have

$$v_j^T v_i = \delta_{ij} \tag{46}$$

That means the matrix of eigenvectors is very special, because

$$\mathrm{V}^T\mathrm{V} = \mathrm{I} \tag{47}$$

or, multiplying on the right by V^{-1},

$$\mathrm{V}^T = \mathrm{V}^{-1} \tag{48}$$

If our matrix is symmetric, then the inverse of the eigenvector matrix is just the transpose of that matrix. We call such matrices *orthogonal.* It also turns out that if we have repeated eigenvalues, we can always mix and match the eigenvectors to once again make V orthogonal. How we decide to do that is not unique, but we always can, and computational routines always will return an orthogonal matrix for V, even if there are repeated eigenvalues. It is also true that the eigenvalues of a symmetric matrix are real, but we'll hold off on that proof for now.

One important fact about orthogonal matrices that we'll point out here is that they don't change the *length* of a vector[6]. If $x' = \mathrm{V}^T x$, We have:

$$x'^T x = (\mathrm{V}x)^T \mathrm{V}x = x^T \mathrm{V}^T \mathrm{V}x = x^t x \tag{49}$$

In fact, the inner product of *any* pair of vectors isn't changed if you multiply both of them by the same orthogonal matrix. Since the dot product of two vectors is the product of the length of the vectors times the cosine of the angle between them, that means the angle between any pair of vectors isn't changed if we multiply by an orthogonal matrix. If I take an arbitrary set of vectors and multiply them all by an orthogonal matrix, then their *relative* geometry hasn't changed at all. You can think of a rotation into a new reference frame as a multiplication by an orthogonal matrix, and in fact the terms rotation matrix and orthogonal matrix are often used interchangeably.

To hammer this point home, we'll use an example that I hope in intuitive to Montrealers. Let's say you're hanging out at the Roddick gates and get a case of the late-night munchies, so you decide to head to Schwarz's. How do you get there? Well, you go along Sherbrooke for 1.1km until you hit Saint Laurent, then you turn left and walk another 750m, at which point you can get your greasy fix. What direction did you head on Sherbrooke? Well, if you ask a local, they'd say you went East (obviously), and then north on Saint-Laurent. Someone from out of town would be very confused by this, though, since if you look at a map, we actually went north-northeast on Sherbrooke, and west-northwest on Saint Laurent. We won't agree on what we call the direction we headed, but we'll always agree on how far we went and what directions we turned. If you ever get swamped by all the basis changes we'll do in quantum mechanics, just come back to this - when we change bases, we're just multiplying by an orthogonal matrix, which is nothing more complicated than switching between island north and true north. Some calculations might be *easier* in one basis or another, but the physics (the directions to Schwarz's) won't change.

The transpose/inverse of an orthogonal matrix is also orthogonal since $\mathrm{V}\mathrm{V}^T = \mathrm{V}^T\mathrm{V} = \mathrm{I}$. The product of two orthogonal matrices (say V and Q) must also be orthogonal: $\mathrm{V}\mathrm{Q}^T\mathrm{V}\mathrm{Q} = \mathrm{Q}^T\mathrm{V}^T\mathrm{V}\mathrm{Q} = \mathrm{Q}^T\mathrm{I}\mathrm{Q} = \mathrm{Q}^T\mathrm{Q} = \mathrm{I}$. Intuitively I hope this becomes obvious, because rotating from one basis to another and then rotating from that basis to a third is equivalent to rotating directly from the first basis to the third.

[6] A fun side note is that Parseval's theorem/Plancherel theorem, which state that Fourier transforms preserve the power in a function, are just special cases of this since it turns out you can write a Fourier transform operator as an orthogonal matrix.

5. Determinants and Singular Value Decomposition

Let's think about matrices in geometric terms. Let's say we write down a bunch of points that define an $n-$dimensional sphere. In two dimensions, we do this by evaluating $[\cos(\theta), \sin(\theta)]$ for a bunch narrowly spaced values of θ between 0 and 2π. We can multiply each of those vectors by our matrix to end up with a new set of vectors that will trace out a new shape. If our matrix is symmetric, the new shape will be an ellipse with principal axes along the eigenvectors, and semi-minor/semi-major axis lengths given by the eigenvalues. The volume of the new ellipse, relative to the $n-$sphere we started with by the product of the eigenvalues, since each axis gets changed by its eigenvalue. We call the product of eigenvalues the *determinant*, and you can think about it as the volume of a matrix. If the determinant of a matrix is zero, then at least one of the eigenvalues is zero. The determinant of a diagonal matrix is just the product of the diagonal elements.

When you think about determinants in this geometric way, it becomes obvious (although not rigorously proved) that the determinant of the product of two matrices is the product of their determinants. Similarly, the order in which you multiply doesn't matter for determinants:

$$|\mathrm{AB}| = |\mathrm{BA}| = |\mathrm{B}||\mathrm{A}| \tag{50}$$

where || denotes the determinant of a matrix. If we have an inverse, the determinant of the inverse must be one over the determinant of the original matrix since:

$$1 = |\mathrm{AA}^{-1}| = |\mathrm{A}||\mathrm{A}^{-1}| \tag{51}$$

A consequence of this is that the determinant of even a non-symmetric matrix is still just the product of its eigenvalues since

$$|\mathrm{A}| = |\mathrm{V}\Lambda\mathrm{V}^{-1}| = |\mathrm{V}||\Lambda||\mathrm{V}^{-1}| = ||\mathrm{V}||\Lambda|/|\mathrm{V}| = |\Lambda| \tag{52}$$

Since an orthogonal matrix doesn't change the length of a vector, the magnitude of its determinant must be 1. More formally, since taking the transpose of a matrix can't change its determinant because the eigenvalues don't change

$$|\mathrm{VV}^{-1}| = |\mathrm{VV}^{T}| = |\mathrm{V}||\mathrm{V}| = |\mathrm{V}|^2 = |\mathrm{I}| = 1 \tag{53}$$

If V is real, then the only allowed values are ±1.

An alternative factorization of a matrix is the *singular value decomposition* or SVD. For a general matrix, we can write down

$$\mathrm{A} = \mathrm{U}\Sigma\mathrm{V}^{T} \tag{54}$$

where $\mathrm{U}^T\mathrm{U} = \mathrm{I}$ and $\mathrm{V}^T\mathrm{V} = \mathrm{I}$, and Σ is diagonal. If A is symmetric, then U=V and the SVD reduces to the eigendecomposition. If A is square but not symmetric, then U and V are square and orthogonal, but not equal to each other. If A is rectangular, then one of U or V is square, the other is rectangular, and Σ matches the smaller dimension of A. If A is square, then up to a possible sign flip the determinant of A is also the product of the elements of Σ, known as the singular values. If A is rectangular, then the determinant doesn't exist, but we can still get an idea of how it is stretching out vectors by looking at the singular values.

I've brought up SVD here because it gives us another way of looking at non-symmetric square matrices that I find useful. I can always insert the identity matrix in an expression so:

$$\mathrm{A} = \mathrm{USV}^T = \left(\mathrm{UV}^T\right)\left(\mathrm{VSV}^T\right) \tag{55}$$

The second term in Equation 55 is a symmetric matrix and the first term is an orthogonal matrix (since the product of two orthogonal matrices is always an orthogonal matrix). Since I mostly deal with symmetric matrices, I find this description useful (or at least comforting). Multiplying by a non-symmetric matrix is just multiplying by a symmetric matrix, with a bonus rotation added on top. Since we can always take the SVD, we can always think of a matrix this way even if it's defective and doesn't have an eigendecomposition.

Before moving on, I would be remiss in my duties if I didn't point out some extremely important practical notes on using SVD. Some languages (looking at you, python) define the SVD as USV, without the tranpose on V. If you're going to use the SVD, *always check.* If you are taking the SVD of a rectangular matrix, then the default behavior is often to return square versions of U and V (once again side-eying python, but in this case the original sin lies I believe with MATLAB). There will be a flag to say "don't be stupid" and return the rectangular versions. For numpy, you add an extra 0 as an input argument, *i.e.*

```
u,s,v=numpy.linalg.svd(A,0)
```

I have never once not wanted the 0, so I suggest you add it by default. It can be the difference between nearly instant results and crashing your computer.

6. Commuting Matrices

In general, matrices do not commute. That is, we expect that:

$$\mathrm{AB} \neq \mathrm{BA} \tag{56}$$

You can see this just by computing the $[0,0]$ element of AB and BA:

$$(\mathrm{AB})[0,0] = \sum A_{0,k} B_{k,0} \tag{57}$$

$$(\mathrm{BA})[0,0] = \sum B_{0,k} A_{k,0} \tag{58}$$

There's no reason to think the first row of A times the first column of B is equal to the first column of A times the first row of B, except under special circumstances.

If AB does equal BA, then the pair of matrices is said to *commute.* One important case is that of diagonal matrices: all pairs of diagonal matrices commute. That's pretty easy to see since

$$(\mathrm{AB})_{ii} = (\mathrm{BA})_{ii} = A_{ii} B_{ii} \tag{59}$$

and of course scalars commute. The identity matrix commutes with all matrices, since

$$\mathrm{AI} = \mathrm{IA} = \mathrm{A} \tag{60}$$

It is *not* true in general that a diagonal matrix commutes with a matrix, since a diagonal B rescales the columns of A in AB, but rescales the rows in BA.

Good life advice when you're trying to show something in linear algebra is to write down the eigenvalues and eigenvectors. In that case we have:

$$\mathrm{AB} = \mathrm{V_A \Lambda_A V_A^{-1} V_B \Lambda_B V_B^{-1}} \tag{61}$$

If the eigenvectors of the two matrices are the same $\mathrm{V} \equiv \mathrm{V_A} = \mathrm{V_B}$, then Equation 61 turns into

$$\mathrm{AB} = \mathrm{V\Lambda_A V^{-1} V \Lambda_B V^{-1}} = \mathrm{V \Lambda_A \Lambda_B V^{-1}} = \mathrm{V \Lambda_B \Lambda_A V^{-1}} = \mathrm{V \Lambda_B V^{-1} V \Lambda_A V^{-1}} = \mathrm{BA} \tag{62}$$

If two matrices share the same *eigenvectors*, they commute. They do not need to share the same eigenvalues. In the case where all the eigenvalues are distinct, this is both necessary and sufficient. When there are repeated eigenvalues, we have freedom to rotate the eigenvectors that span the repeated eigenvalues. As an example, the eigenvalues of the identity matrix are all one, so any vector is an eigenvector of I. That explains why I commutes with everything - when every vector is an eigenvector, then you're free to use the eigenvectors from any matrix. Since I could have picked the eigenvectors to be the same, then I commutes with every matrix. More generally, if there exists a matrix P that diagonalizes both A and B (*i.e.* $\mathrm{A} = \mathrm{PD_A P^{-1}}$ for diagonal D and similarly for B) then A and B commute. Note that it is not true that if A and B commute and B and C commute that A and C commute. Put I in for B, then that has to commute with any A and C, but of course, arbitrary A and C would not commute. If the eigenvalues are all distinct, though then A commuting with B means A and B have the same eigenvectors, and B commuting with C means they have the same eigenvectors, so A and C share eigenvectors and hence commute.

We can qualitatively think of the commutator of A and B, defined to be $\mathrm{AB} - \mathrm{BA}$, to be a measure of how misaligned the principal axes of A and B are. The closer the eigenvectors, the smaller the commutator. The *eigenvalues* matter a lot less for commuting.

7. Complex Matrices

There's no way around complex matrices in quantum mechanics. You can't get interference without complex arithmetic, so now we'll go back over some of what we've discussed for real matrices through a complex lens. When moving to complex matrices, what should the inner product look like? For real values, the squared length of a vector x is just the dot product, or $x^T x$. For a complex-valued vector, we want the inner product to return $\sum |x_i|^2 = \sum (x_i)^* x_i$. To get this, we need the inner product of x with itself to be not be the transpose $x^T x$ but the *conjugate-transpsose* $x^{*T} x$. This conjugate-transpose operation is so common, especially in quantum, that we use a dagger as shorthand, *i.e.* $x^{*T} x = x^\dagger x$. We need to be a little more careful with the inner product of two distinct vectors because $x^\dagger y = (y^\dagger x)^*$, so swapping the order changes the output to its complex conjugate.

Now that we know what we want the inner product to do, we can work out what a rotation matrix looks like. We'll need to use the fact that $(\mathrm{AB})^\dagger = \mathrm{B^\dagger A^\dagger}$ (just like transpose). If we let $x' = \mathrm{V}x$, then we want

$$x^\dagger x = x'^\dagger x' = (\mathrm{V}x)^\dagger (\mathrm{V}x) = x^\dagger \mathrm{V^\dagger V} x \tag{63}$$

For this to work, we need $\mathrm{V}^\dagger\mathrm{V} = \mathrm{I}$. If we have a complex matrix, it is orthogonal if its inverse is its conjugate-transpose, not just its transpose. We call such matrices *unitary.* All of their eigenvalues must have a magnitude of one, since if we put in an eigenvector for x, then we would pick up a factor of $\lambda^*\lambda$. We can't say anything about *phases*, though.

Quantum mechanics requires that we get a real value when taking the expectation of an operator. Mathematically that will correspond to $x^\dagger \mathrm{A} x = c$ where c is strictly real for *any* complex x. Let's look and see what that implies about our matrix. We can take the complex transpose to get:

$$\left(x^\dagger \mathrm{A} x\right)^\dagger = x^\dagger \mathrm{A}^\dagger x = c \tag{64}$$

Since this has to be true for *any* x, that means that $\mathrm{A} = \mathrm{A}^{\dagger}$[7] . We call a matrix that is equal to its own conjugate transpose *Hermitian*, and every operator in quantum mechanics will have to be Hermitian. Not surprisingly, the fact that a matrix is Hermitian implies it has real eigenvalues since

$$v^\dagger \mathrm{A} v = v^\dagger \lambda v = \lambda \tag{65}$$

Since $x^\dagger \mathrm{A} x$ is real for any x, it also has to be true for an eigenvectors, hence λ must be real.

Now that we know we are going to work with Hermitian matrices with real eigenvalues, let's see what we can say about the eigenvectors. We have

$$\mathrm{A}^\dagger = \mathrm{A} = \mathrm{V}\Lambda\mathrm{V}^{-1} \tag{66}$$

$$\mathrm{A}^\dagger = \left(\mathrm{V}\Lambda\mathrm{V}^{-1}\right)^\dagger = \mathrm{V}^{\dagger-1}\Lambda\mathrm{V}^\dagger \tag{67}$$

For both equations to be true, it had better be the case that the matrices on the right agree, so we have

$$\mathrm{V}^\dagger = \mathrm{V}^{-1} \tag{68}$$

This also implies that $\mathrm{V}^{\dagger-1} = \mathrm{V}$, which you can see by just taking the inverse of Equation 68. Since the inverse of V is its conjugate-transpose, then the matrix of eigenvectors is unitary and the inner product of any distinct pair of eigenvectors is zero. The *magnitude* of the eigenvalues must be one, but they are once again allowed to have arbitrary *phase.*

We've now shown that if $x^\dagger \mathrm{A} x$ is real for any vector x, that A must be Hermitian, that it has real eigenvalues, and that its eigenvectors are orthogonal. Once we switch to complex values, a symmetric matrix corresponds to a Hermitian matrix, not a complex symmetric one. With this background, we can now convert bras/kets *etc.* into linear algebra notation.

[7] In case you aren't convinced, consider a vector that has a single one and is zero everywhere else. $x^\dagger \mathrm{A} x$ just picks out the corresponding diagonal entry of A, so the diagonal entries must all be real. Now consider a vector with two ones, say in the i^{th} and j^{th} entries. Then $x^\dagger \mathrm{A} x$ gives us $A_{ii} + A_{ij} + A_{ji} + A_{jj}$. Since the diagonal entries are real, then $A_{ij} + A_{ji}$ must also be real, which is only true if the complex parts of A_{ij} and A_{ji} are equal and opposite. A similar exercise shows that the real part of A must be symmetric, so $A_{ij} = A^*_{ji}$ for every i, j.

8. Dirac Notation in Linear Algebra

Let's say we have a particle in a state Ψ in quantum mechanics. If we want to write down that state, we have to pick a basis to write it down in. The values in the state are then the amplitudes to find the particle in a given eigenstate in our basis, if we were to carry out a measurement. In the case of Stern-Gerlach, if we were to write down a state in the $\left|+z\right\rangle, \left|-z\right\rangle$ basis, then the $\left|+z\right\rangle$ component would be the amplitude to find our particle in the $\left|+z\right\rangle$ state, etc. By convention, we treat kets as column vectors. The length of every state has to be one, since when we make a measurement, we have to get *something*. When we have a particle in a given state, we know that both the amplitude and the probability of finding it in that state must be 1. In Dirac notation, that means that

$$\langle\Psi|\Psi\rangle = 1 \tag{69}$$

. In linear algebra, if we want to say that the length of a vector Ψ is 1, we would mean that $\Psi^\dagger\Psi = 1$. Since we decided that a ket is a column vector, the only way this works is if the corresponding bra in Equation 69 is the conjugate-transpose of our vector.

We know the expectation of a function is

$$\langle f(x)\rangle = \sum_i f(x_i)P(x_i) \tag{70}$$

where x can only take on discrete values x_i, and $P(x_i)$ is the probability that $x = x_i$[8]. If we have an observable (say angular momentum along an axis, energy...) that takes on discrete values, then we can write an arbitrary wave function Φ as a superposition of the states Ψ_i that correspond to the possible values of the observable. For simplicity let's assume energy, and we'll label the energy of Ψ_i to be E_i. That gives:

$$\Phi = \sum_i c_i \left|\Psi_i\right\rangle \tag{71}$$

and expected energy E_ϕ:

$$\langle E_\Phi\rangle = \sum_i E_i c_i^* c_i \tag{72}$$

since the probability to be in state Ψ_i is $c_i^* c_i$. We can write this as:

$$\langle E_\Phi\rangle = \begin{bmatrix} c_1^* & c_2^* & c_3^* & ... \end{bmatrix} \begin{bmatrix} E_1 & 0 & 0... \\ 0 & E_2 & 0... \\ 0 & 0 & E_3... \\ . & & \\ . & & \\ . & & \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ . \\ . \\ . \end{bmatrix} \tag{73}$$

If you carry out these multiplications, you'll see that Equation 73 is identically equal to our original expresion Equation refeqn:expected. So, if I've written down my state in the energy basis, I can find the expected energy with the matrix product

$$\Phi^\dagger \hat{E} \Phi \tag{74}$$

[8] This turns into an integral in the continuous limit, but then we're dealing with finite-sized matrices here so we'll treat the sum as discrete.

where Φ is the vector of $\begin{bmatrix} c_1^* & c_2^* & c_3^* & ... \end{bmatrix}$ and $\hat{E}$ is the diagonal matrix of the energy values. If I had written down Φ in a *different* basis, where the energy states were described as a set of vectors V_i, then I could rotate Φ into the energy basis by multiplying by the bras of those vectors, which is equivalent to multiplying by the matrix $V^\dagger$. That means my expected energy is now

$$\begin{aligned} \langle E \rangle &= (V^\dagger \Phi))))^\dagger \hat{E} V^\dagger \phi && (75) \\ &= \Phi^\dagger V \hat{E} V^\dagger \Phi && (76) \end{aligned}$$

Equation **??** tells us that if we have an observable with real values, we can find the expected value of that operator by sandwiching a Hermitian matrix with the wave function. The eigenvectors are the representation of the pure states of our observable and the eigenvalues or the corresponding values.