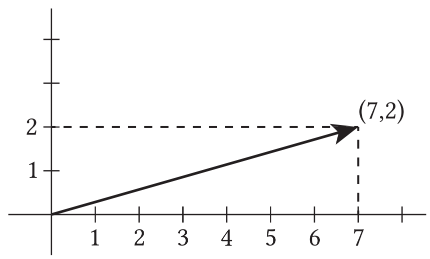
Coordinate Rotations

August 26, 2026

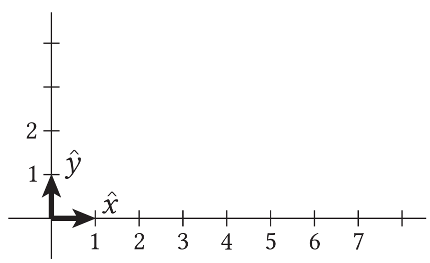
1 Vectors

1.1 Standard Basis

In a 2D Cartesian coordinate system, we have the x and y axes. When we talk about where a verctor is pointing, we talk about the *components* of a vector. For example, the vector $(7, 2)$ has a component of 7 in the x direction and a component of 2 in the y direction.



We can represent our vector $(7, 2)$ as a sum of the **basis vectors** $\hat{x}$ and $\hat{y}$. These are **unit vectors** (vectors with length 1) that point in the x and y directions respectively.



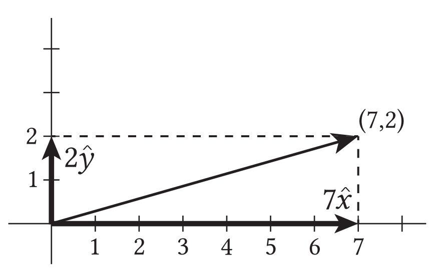
$$\hat{x} = (1, 0)$$

$$\hat{y} = (0, 1)$$

Our vector $(7, 2)$ points 7 units in the x direction and 2 units in the y direction. We can represent our vector $(7, 2)$ as the sum of basis vectors $\hat{x}$ and $\hat{y}$:

$$(7, 2) = 7\hat{x} + 2\hat{y}$$

We say that the vector $(7, 2)$ is a **linear combination** of $\hat{x}$ and $\hat{y}$. This means that it is a weighted sum of the basis vectors $\hat{x}$ and $\hat{y}$. Every vector in a cartesion coordinate system can be expressed as a linear combination of the basis vectors.



In 3D Cartesian coordinate systems, we also have a $\hat{z}$ vector that points in the z direction.

$$\hat{x} = (1, 0, 0)$$

$$\hat{y} = (0, 1, 0)$$

$$\hat{z} = (0, 0, 1)$$

Generally speaking, it is common to have vectors with more than three components. When that happens, we need a standard basis with more than three vectors. But it is inconvenient to give them letter names like $\hat{x}$, $\hat{y}$, $\hat{z}$ because we will run out of letters. The e_k vectors are the **standard basis** for cartesian systems with an arbitrary number of coordinates:

$$e_0 = (1, 0, 0, 0, ...)$$
$$e_1 = (0, 1, 0, 0, ...)$$
$$e_2 = (0, 0, 1, 0, ...)$$

In general, e_k is the basis vector that points in the kth dimension of a vector space. The standard basis is an extension of $\hat{x}$, $\hat{y}$, $\hat{z}$ to more than three dimensions.

1.2 Vector Norms

A **norm** is a function that measures the length of a vector. Length must be positive or zero. Whatever function we choose to define length must always generate an output that is positive or zero.

1.2.1 2-Norm / Euclidean Norm

The Euclidean norm tells us the length between two objects on a straight line. Suppose we have some vector v in 3-space that we want to measure.

$$v = \begin{pmatrix} 7 \\ 4 \\ 4 \end{pmatrix}$$

The most common norm is the Euclidean norm or 2-norm, which is an extension of the pythagorean theorem.

$$||v||_2 = \sqrt{7^2 + 4^2 + 4^2} = \sqrt{49 + 16 + 16} = \sqrt{81} = 9$$

The two vertical bars around v indicate that we're taking the norm of the vector. The subscript 2 indicates that we're taking the 2-norm (as opposed to other distance metrics, discussed below). The general form of the 2-norm for a vector in an arbitrary number of dimensions is:

$$||v||_2 = \left(\sum_k v_k^2 \right)^{1/2}$$

The Euclidean norm is called the 2-norm because the exponents on each of the components are 2.

Pros of the 2-norm: it is a natural notion of distance or length that we are used to working with in the real world.

Cons of the 2-norm: it can be difficult to work with because of the squares and the square root. We occasionally use other notions of distance that are either more appropriate for a situation or are easier to work with.

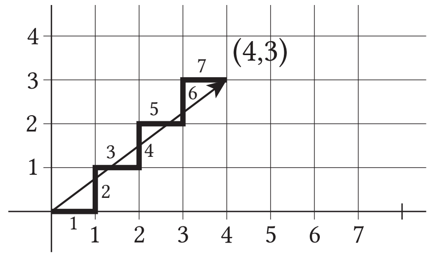
1.2.2 1-Norm / Manhattan Norm / Taxicab Norm

The 1-norm tells us how far apart two points are if we are constrained to a grid system—that is if we can't jump diagonally between two points. This is the distance between two points in a city grid of streets (hence the name Manhattan/taxicab norm). In the graph below, we plot the vector $v = (4, 3)$ on a Cartesian plane. Its length in the 2-norm is:

$$||v||_2 = \sqrt{4^2 + 3^2} = \sqrt{16 + 9} = \sqrt{25} = 5$$

Its length in the 1-norm (traced in bold) is:

$$||v||_1 = \left(\sum_k |v_k|^1 \right)^1 = (4 + 3) = 7$$



The length of the vector v in the 1-norm is not the same as its length in the 2-norm. So why would we use this norm?

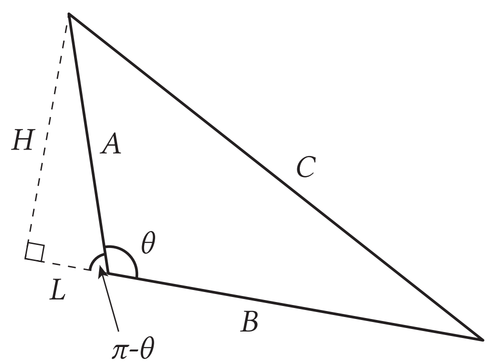
Pros of the 1-norm: in many situations, it's easier to work with the 1-norm because it doesn't involve any squares or square roots.

Cons of the 1-norm: it isn't the same as the Euclidean length of a vector, so it is a bit weird. Even so, it satisfies the key requirement of a vector norm, which is that it must always produce a positive or zero length.

1.3 Inner Products

Theorem 1.1 (Law of Cosines)**.** *Consider the triangle below with sides A, B, C, and angle θ between A and B.*

$$C = \sqrt{A^2 + B^2 - 2ABcos\theta}$$



Proof. First, the Pythagorean theorem for the dotted triangle on the left side, we have

$$L^2 + H^2 = A^2 \tag{1}$$

Next, the Pythagorean theorem for the main triangle gives us:

$$\begin{align} C^2 &= H^2 + (L + B)^2 \tag{2} \\ &= H^2 + L^2 + 2LB + B^2 \tag{3} \\ &= A^2 + 2LB + B^2 \tag{4} \end{align}$$

Where the second step comes from expanding $(L + B)^2$ and substituting Equation (1). Then, from good ol' fashioned SOH CAH TOA:

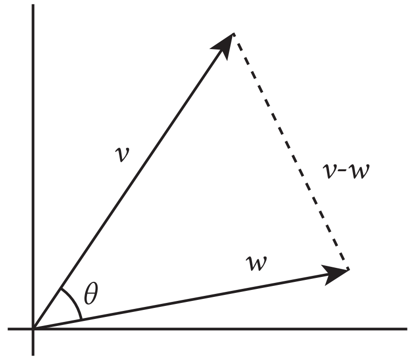
$$\begin{aligned} cos(\pi - \theta) &= \frac{L}{A} \\ L &= Acos(\pi - \theta) \\ &= -Acos\theta \end{aligned}$$

Plug in to the Pythagorean theorem for the main triangle (Equation (4)) to get the Law of Cosines:

$$C^2 = A^2 - 2ABcos\theta + B^2 \tag{5}$$

□

The **inner product**[1] measures the distance between two vectors. For example, suppose we have a pair of vectors v and w separated by angle θ, and want to compute the distance between them. The two vectors form a triangle (shown in the diagram below).



Definition 1.1 (Inner Product)**.** The inner product of two vectors is the square of the distance between them.

$$
\begin{aligned}
< v, w > \equiv ||v - w||^2 &= ||v||^2 + ||w||^2 - 2||v||||w||cos\theta \\
\sum_k (v_k - w_k)^2 &= \sum_k v_k^2 + \sum_k w_k^2 - 2||v||||w||cos\theta \\
\sum_k \left(v_k^2 + w_k^2 - 2v_k w_k\right) &= \left(\sum_k v_k^2\right) + \left(\sum_k w_k^2\right) - 2||v||||w||cos\theta \\
\left(\sum_k v_k^2\right) + \left(\sum_k w_k^2\right) - \left(2\sum_k v_k w_k\right) &= \left(\sum_k v_k^2\right) + \left(\sum_k w_k^2\right) - 2||v||||w||cos\theta \\
\sum_k v_k w_k &= ||v||||w||cos\theta
\end{aligned}
$$

1.3.1 Properties of the Inner Product

$$||v|| = \sqrt{< v, v >}$$

Transposition

$$A^T_{m\times n} = (a^T_{ij})$$

$$(a^T_{ij}) = (a_{ji})$$

1. $(A^T)^T = A$
2. $AB = B^T A^T$
3. $< Av, u > = < v, A^T u >$

2 Matrices

A matrix is a set of vectors grouped together.

$$R = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

Multiplying R by a vector $\left[\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right]$

$$\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \times 1 + (-1) \times 1 \\ 1 \times 1 + 0 \times 1 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

[1] sometimes called the **dot product**

2.1 Inversion

Find $A_{m\times m}^{-1}$ such that $A^{-1}A = AA^{-1} = I_{m\times m}$

Fact: $(AB)^{-1} = B^{-1}A^{-1}$

Matrix is a linear map from $\mathbb{R}^m \underset{A^{-1}}{\overset{A}{\rightleftarrows}} \mathbb{R}^m$

2.2 Orthogonal Matrices

Definition 2.1 (Orthogonal Matrix)**.** An orthogonal matrix is a matrix $\mathbf{Q}$ with the property that $\mathbf{Q}^T = \mathbf{Q}^{-1}$

When interpreted as vectors, the columns of an orthogonal matrix are mutually orthogonal.

$$\mathbf{Q} = \begin{bmatrix} | & | & | & | \\ q_1 & q_2 & ... & q_N \\ | & | & | & | \end{bmatrix}$$

The columns q_k have the property that $< q_i, q_j >= 0$ for $i \neq j$ and $< q_i, q_j >= 1$ for $i = j$. The same property applies to the rows of $\mathbf{Q}$.

Example

$$\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}$$

The rows of this matrix are orthogonal vectors:

$$\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{-\sqrt{2}} \end{bmatrix} = \left(\frac{1}{\sqrt{2}}\right)\left(\frac{1}{\sqrt{2}}\right) + \left(\frac{1}{\sqrt{2}}\right)\left(\frac{-1}{\sqrt{2}}\right) = +\frac{1}{2} - \frac{1}{2} = 0$$

Also, this matrix's transpose is its inverse:

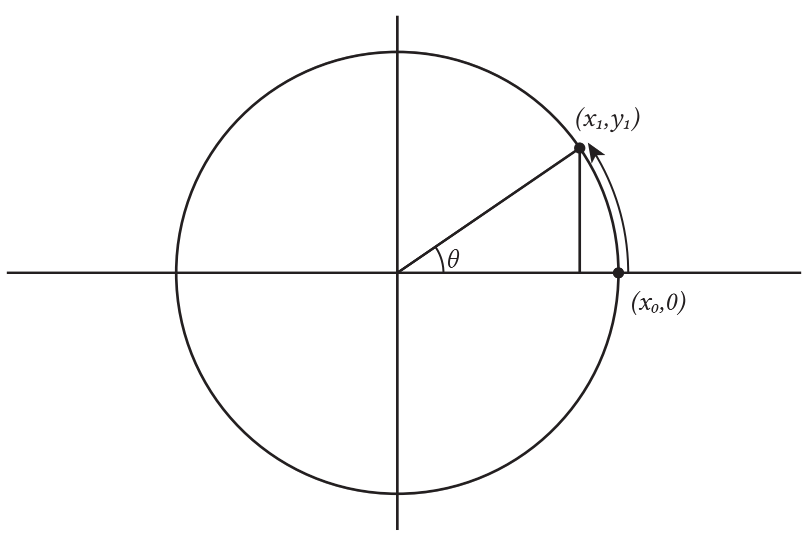
$$\mathbf{Q}^T\mathbf{Q} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

3 Coordinate Rotations

In the figure below, we want to rotate a point that initially lies on the x-axis by an angle θ. The new point's coordinates will be (x_1, y_1). We want to calculate those coordinates in terms of the original coordinate $(x_0, 0)$ and θ.

$$sin\theta = \frac{y_1}{x_0} \qquad cos\theta = \frac{x_1}{x_0}$$

$$y_1 = x_0 sin\theta \qquad x_1 = x_0 cos\theta$$



We can represent this operation in matrix form:

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} cos\theta & 0 \\ sin\theta & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}$$

Now what about the more general case where the point to be rotated does not start out on the x-axis? Suppose we want to rotate the point (x_1, y_1) by angle ϕ as shown in the diagram below. In polar coordinates, the new point lives at:

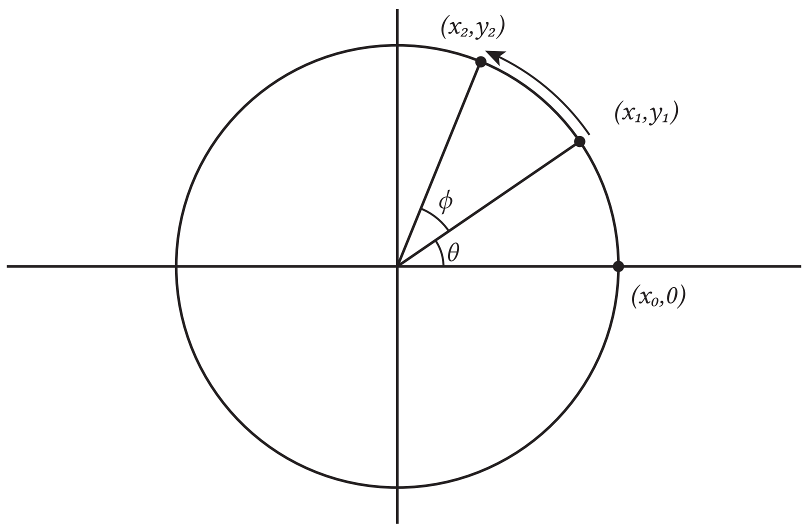
$$y_2 = x_0 sin(\theta + \phi) \qquad\qquad x_2 = x_0 cos(\theta + \phi)$$

Using the sum-of-angles identity, we get:

$$sin(\alpha \pm \beta) = sin\alpha cos\beta \pm cos\alpha sin\beta \qquad\qquad cos(\alpha \pm \beta) = cos\alpha cos\beta \mp sin\alpha sin\beta$$

$$\begin{aligned} y_2 &= x_0 sin(\theta + \phi) \\ &= \underbrace{x_0 sin\theta}_{y_1} cos\phi + \underbrace{x_0 cos\theta}_{x_1} sin\phi \\ &= y_1 cos\phi + x_1 sin\phi \end{aligned}$$

$$\begin{aligned} x_2 &= x_0 cos(\theta + \phi) \\ &= \underbrace{x_0 cos\theta}_{x_1} cos\phi - \underbrace{x_0 sin\theta}_{y_1} sin\phi \\ &= x_1 cos\phi - y_1 sin\phi \end{aligned}$$



Putting this into matrix form:

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} cos\phi & -sin\phi \\ sin\phi & cos\phi \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

Fact: This matrix is orthogonal.

$$\begin{bmatrix} cos\phi & -sin\phi \\ sin\phi & cos\phi \end{bmatrix} \begin{bmatrix} cos\phi & sin\phi \\ -sin\phi & cos\phi \end{bmatrix} = \begin{bmatrix} cos^2\phi + sin^2\phi & sin\phi cos\phi - sin\phi cos\phi \\ sin\phi cos\phi - sin\phi cos\phi & sin^2\phi + cos\phi \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

3.1 Rotations in 3 Dimensions

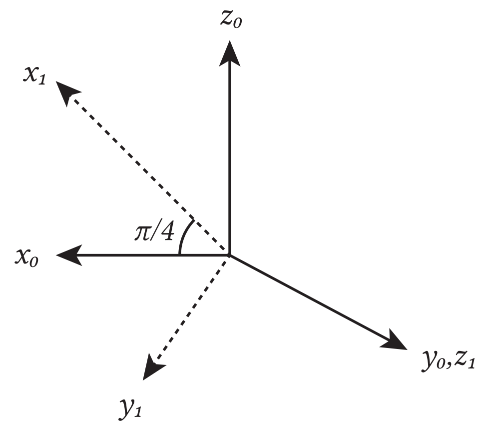
The example above represents a coordinate rotation in 3 dimensions in which we are rotating according to the right hand rule about the z-axis. We can extend the matrix to a 3×3 and set it to retain the z coordinate.

$$R_{z,\phi} = \begin{bmatrix} cos\phi & -sin\phi & 0 \\ sin\phi & cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Using similar logic we can derive matrices for rotating around the x and y axes:

$$R_{x,\phi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\phi & -sin\phi \\ 0 & sin\phi & cos\phi \end{bmatrix} \qquad R_{y,\phi} = \begin{bmatrix} cos\phi & 0 & sin\phi \\ 0 & 1 & 0 \\ -sin\phi & 0 & cos\phi \end{bmatrix}$$

Example Consider the coordinate axes below. Find the unit vectors x_1, y_1, and z_1 in terms of the base coordinate frame x_0, y_0, and z_0.



- $\hat{\mathbf{x}}_1$ lives in the $\mathbf{x}_0 - \mathbf{z}_0$ plane (with the y_0 component of $\hat{\mathbf{x}}_\mathbf{1}$ equal to 0).

$$\hat{x}_1 = \frac{\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}}{\left\| \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \right\|} = \frac{\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}}{\sqrt{1+0+1}} = \begin{bmatrix} 1/\sqrt{2} \\ 0 \\ 1/\sqrt{2} \end{bmatrix}$$

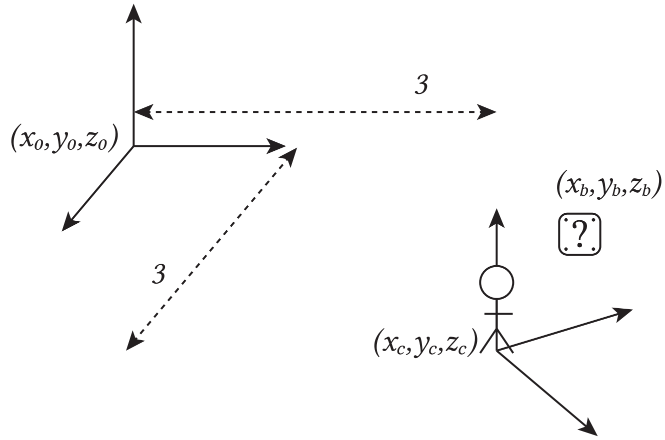
- $\hat{\mathbf{y}}_1$ lives in the $\mathbf{x}_0 - \mathbf{y}_0$ plane (with the z_0 component of $\hat{\mathbf{y}}_\mathbf{1}$ equal to 0).

$$\hat{x}_1 = \frac{\begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}}{\left\| \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \right\|} = \frac{\begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}}{\sqrt{1+0+1}} = \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \\ 0 \end{bmatrix}$$

- $\hat{\mathbf{z}}_\mathbf{1} = \hat{\mathbf{y}}_\mathbf{0} = [0\ 1\ 0]$

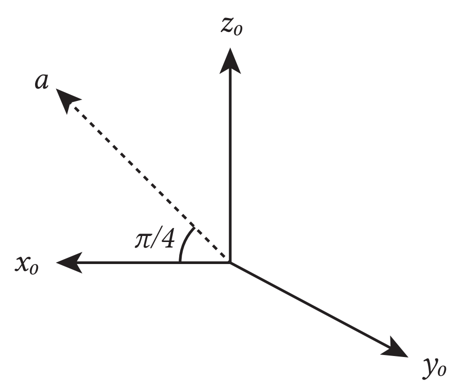
Example You are a character in a video game shown in the diagram below. The global origin is (x_0, y_0, z_0), and the character's location is (x_c, y_c, z_c). The character's z-axis z_c is parallel to the global z-axis z_0, but the characters $x - y$ plane is rotated 30 degrees counterclockwise and translated by 3 units in the x direction and 3 units in the y direction. The x, y planes in both coordinate systems are coplanar. There is a question mark box located at $(x_b^0, y_b^0, z_b^0) = (4, 4, 1)$ relative to the global origin.

- Label the axes (x_0, y_0, z_0) and (x_c, y_c, z_c).
- Write down (x_c^0, y_c^0, z_c^0), the location of (x_c, y_c, z_c) in the (x_0, y_0, z_0) coordinate system.
- Write down $(\hat{x}_c^0, \hat{y}_c^0, \hat{z}_c^0)$, the unit vectors that point in the x_c, y_c and z_c directions (relative to the (x_0, y_0, z_0) coordinate system).



3.2 Similarity Transforms

We covered how to rotate around the x, y or z axes, but what if we want to rotate your coordinate frame around an axis that is not parallel to one of the basis vectors? For example, suppose we want to come up with a rotation matrix that rotates about the axis a in the diagram below. None of the coordinate transformations we have discussed so far allow us to rotate about an arbitrary axis.



We can accomplish this using a 3-step process called a Similarity Transform:

1. Apply a change of coordinates $\mathbf{R}$ to rotate the base coordinate system into a new coordinate system in which the desired rotation axis is parallel to either the x, y, or z axes.
2. Apply the desired coordinate rotation using $\mathbf{R}_{x,\phi}$, $\mathbf{R}_{y,\phi}$ or $\mathbf{R}_{z,\phi}$.
3. Apply $\mathbf{R}^{-1}$ to change the coordinate system back to the original basis.

Example Come up with a rotation matrix that rotates by $\pi/2$ around axis a in the diagram above. First, let's find $\mathbf{R}_a$ a rotation matrix that rotates by $-\pi/4$ radians around the y_0 axis. This will make the new x-axis aligned with the desired axis of rotation a.

$$\mathbf{R}_a = R_{y,\pi/2} = \begin{bmatrix} cos\pi/2 & 0 & sin\pi/2 \\ 0 & 1 & 0 \\ -sin\pi/2 & 0 & cos\pi/2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

Note: $\mathbf{R}_a$ is an orthogonal matrix, so $\mathbf{R}_a^{-1} = \mathbf{R}_a$

After rotating into the new coordinate frame, the a-axis will be the same as the x axis. In this new coordinate frame, rotating around the a-axis by $\pi/2$ can be accomplished by applying the $\mathbf{R}_{x,\pi/2}$

$$R_{x,\phi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\phi & -sin\phi \\ 0 & sin\phi & cos\phi \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

Finally, after applying $\mathbf{R}_{x,\pi/2}$, we will return to the original coordinate system by applying $\mathbf{R}_a^{-1} = \mathbf{R}_a^T$

$$\mathbf{R}_a^T = R_{y,\pi/2} == \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

The overall similarity transform is:

$$\mathbf{R}_S = \mathbf{R}_a^T \mathbf{R}_{x,\pi/2} \mathbf{R}_a = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$