Linear Algebra

Linear Images of Sets

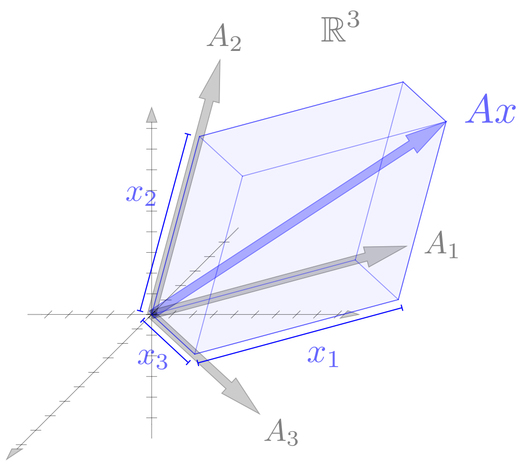
Dan Calderone

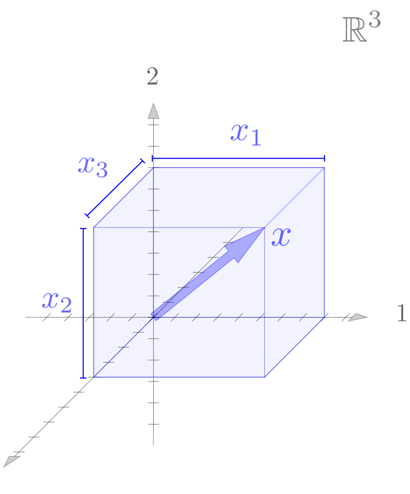
1 Image of Sets

In general, the types of combinations detailed in the previous lecture are just special cases of visualizing any set of vectors passed through a linear map. By definition, linear maps are operations that preserve stretching and additive operations. Because of this linear maps consist of compositions of relatively simple geometric transformations - stretching, skewing, rotating, and reflecting - and we can usually easily visualize the image of any set through linear map. We give several examples here. This intuition will be foundational to understanding the column geometry of a matrix which we will discuss more in a later lecture.

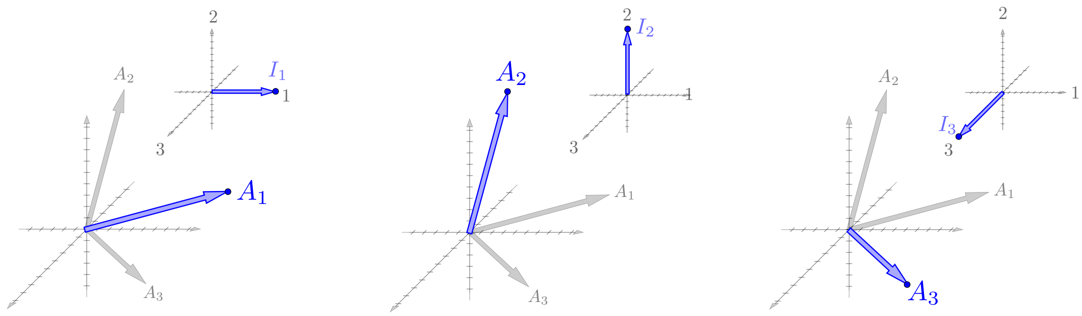
1.1 Vector

The simplest (and most important) case to visualize is the image of a vector through a linear map. We have already built up the intuition for this case above. Since a vector $x \in \mathbb{R}^n$ times a matrix $A \in \mathbb{R}^{m\times n}$ simply takes linear combinations of the columns, we simply treat the columns of A as the coordinate axes and draw the vector x relative to them. Note this visualization concept remains the same for any dimension of the columns of A. We given an example here.





It is worth giving special attention to the case where x is a column of the identity, ie. a standard basis vector (for example, $x = (1, 0, \cdots, 0)$). In these case Ax is just a particular column of A. We give examples of this here.



Algebraically, this is if $x = I_j$ where I_j is the jth column of the identity then $Ax = AI_j = A_j$ where A_j is the jth column of A.

1.1.1 More Drawing: nD vectors (Difficulty: 1/10)

Up to this point, we have been discussing spatial intuition for vectors in the abstract separate from the process of drawing them. Drawing/visualizing vectors in high dimensions uses much of the same intuition. In order to draw simple 2D projections of nD vectors, we need to assign directions (on the 2D page) for each basis vector $I_j \in \mathbb{R}^n$ of the nD space. We can store these 2D (in the page) directions in a matrix $\mathbf{X}_n \in \mathbb{R}^{2\times n}$ with column j being the jth axis direction. Sometimes we will simply denote this $\mathbf{X}$ assuming the column dimension from context. We give several examples here.

[[INSERT IMAGE]]

When we draw a vector $x \in \mathbb{R}^n$, we are actually drawing the 2D arrow given by $\mathbf{X}_n x$. When we draw columns of a matrix $A \in \mathbb{R}^{m\times n}$, we draw arrows given by the 2D columns of $\mathbf{X}_m A$. If we draw the columns of a product of matrices AB with $B \in \mathbb{R}^{n\times p}$, we are actually drawing columns of $\mathbf{X}_m AB$, etc.

1.1.2 More Drawing: The Problem of Depth (Difficulty: 3/10)

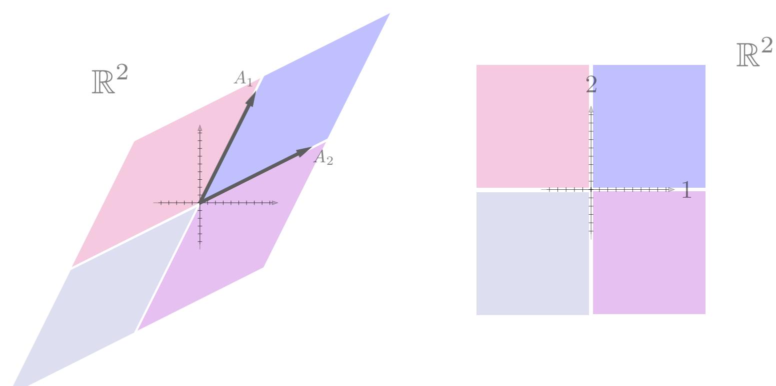
When trying to visualize perpendicular axes, we run into a problem. Since we can only draw in two dimensions, there are not enough perpendicular directions for all the axes for vectors of dimension three or higher. The best we can do is to pick directions (in the 2D image) for each of these axes to point and think of them as being perpendicular in some higher dimensional space. Our drawing then becomes a 2D projection of that higher dimensional space. This depiction does not lose any information for 2D vectors, but for 3D vectors, there is a depth direction (towards the viewer) that is lost in the drawing. Since we live in 3D, we are used to this and can often spatially reconstruct that lost direction in our mental representation of the vector. For 4D vectors, the depth "direction" becomes a plane; for 5D vectors, depth becomes a 3D space; for 6D vectors, depth becomes a 4D space, etc, and our drawing loses quite a bit of information and our intuition suffers accordingly. If you enjoy spatial reasoning, these 2D representations of higher dimensional vectors can still be useful, but you should be careful not to forget the information lost in the projection. Further we should note that the depth dimensions are given by the nullspace of $\mathbf{X}$. The rank-nullity theorem here determines that there are $n-2$ depth dimensions.

1.1.3 Combinations as Images

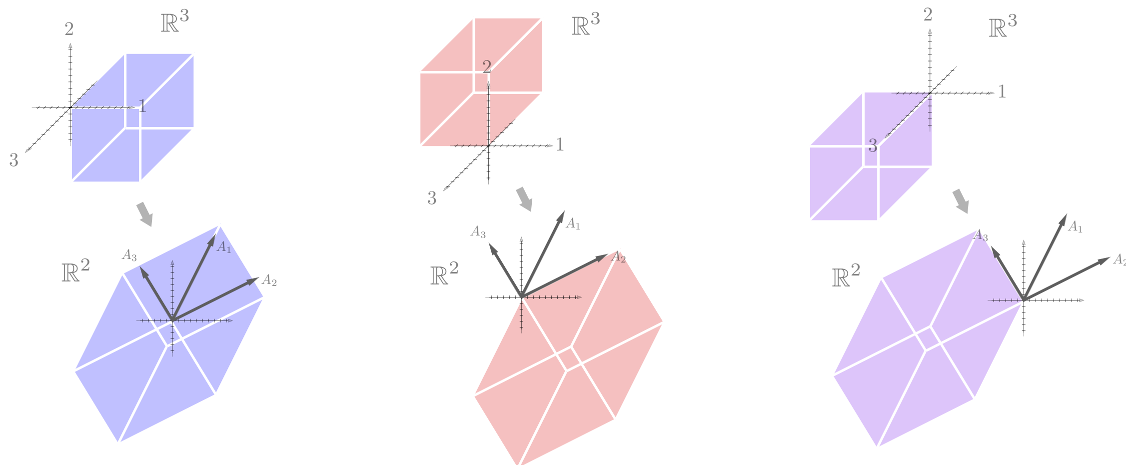
Note that the linear, positive, and convex combinations of the columns of a matrix as discussed in the previous lecture can simply be thought of as images of particular sets through the matrix discussed below. Specifically, positive combinations are the image of the positive orthant with signature $(+,+,\ldots,+)$ (see Sec. 1.2; convex combinations are the image of the simplex 1.3; and linear combinations are the image of an entire space through the matrix.

1.2 Orthants

We now consider the image of the different orthants of the domain through a matrix. An *orthant* is a region of a vector space where each coordinate has a specific sign either positive or negative. Orthants can be labeled by their signatures. For example in $\mathbb{R}^2$, there are four orthants with the signatures $(+,+)$, $(+,-)$, $(-,+)$, and $(-,-)$ shown in the Figure below (on the left). It is quite useful to visualize the images of the orthants through a linear map to get a sense for the structure of the transformation of the whole space. We show an example here in 2D.



In general in nD, the number of orthants is given by 2^n, since for every
Here we display a few of the orthants in 3D under linear transformation.

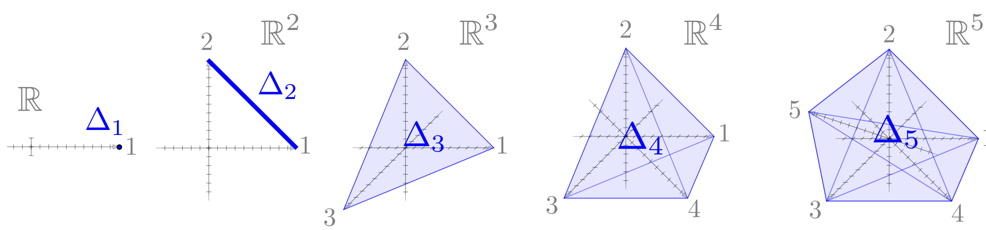


1.3 Simplex

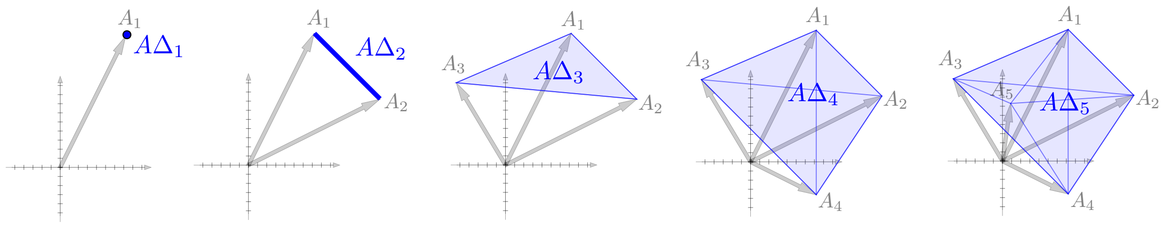
The simplex in dimension n is the set of vectors whose elements are positive and sum to one. We denote this as Δ_n

$$\Delta_n = \left\{ x \in \mathbb{R}^n \;\middle|\; \mathbf{1}^\top x = 1,\ x \geq 0 \right\} \tag{1}$$

where $\mathbf{1}$ is a vector of 1's and the inequality is taken elementwise. (In some cases, we will suppress and simply write Δ_n as Δ and let the dimensino be implied by context.) Note that the simplex is equivalent to the set of all convex combinations. We will use this set quite often so it is worth visualizing in several dimensions. We show it here in 1D up through 5D.



Note that in 1D, Δ_1 is just the scalar 1; in 2D, Δ_2 is the segment between the two standard basis vectors; and in 3D and higher, it is the set connecting all the standard basis vectors. Our intuition from convex combinations discussed above all applies here.



For a matrix $A \in \mathbb{R}^{m\times n}$, we will denote the image of the nD simplex as $A\Delta_n$. We draw these images here for sample A's of the appropriate dimension. Note again that we simply replace the standard basis vectors on the axes with the columns of A. As expected, the image of the simplex is simply the convex combination of the columns of A.

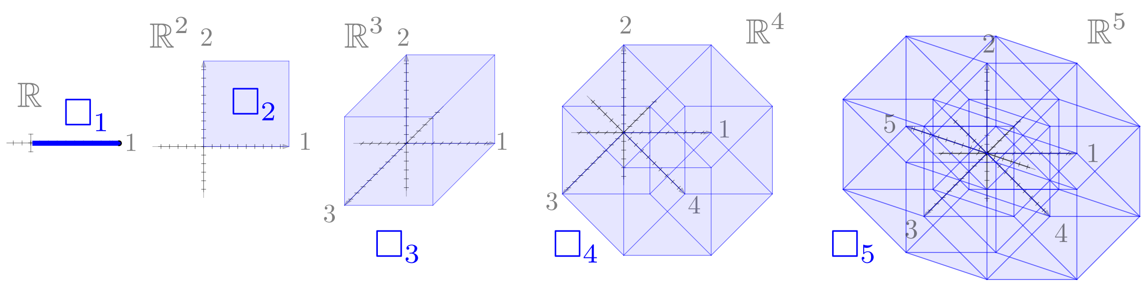
1.4 Cubes

Another important example is the unit cube in $\mathbb{R}^n$ which we denote as $\square_n$.

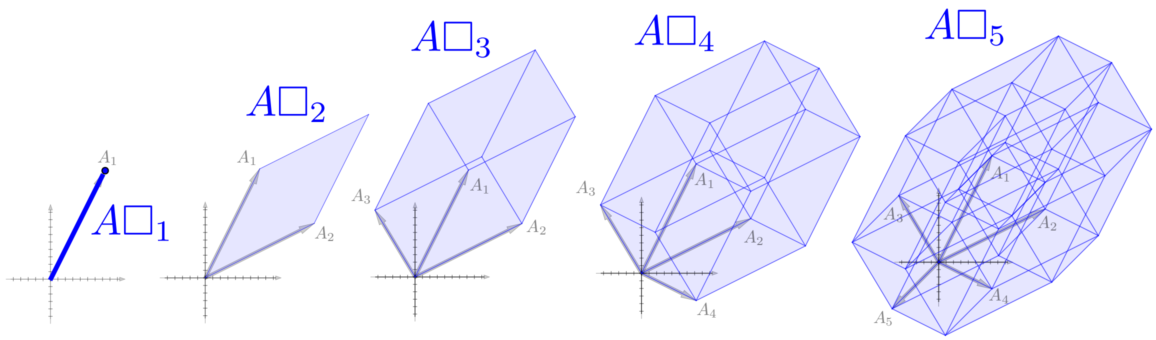
$$\square_n = \left\{ x \in \mathbb{R}^n \;\middle|\; 0 \le x \le \mathbf{1} \right\} \tag{2}$$

where again the inequalities are taken elementwise. (Again, sometimes we will denote $\square_n$ simply as $\square$ with the dimension provided by context.)

We show these cubes here in dimensions 1D through 5D.



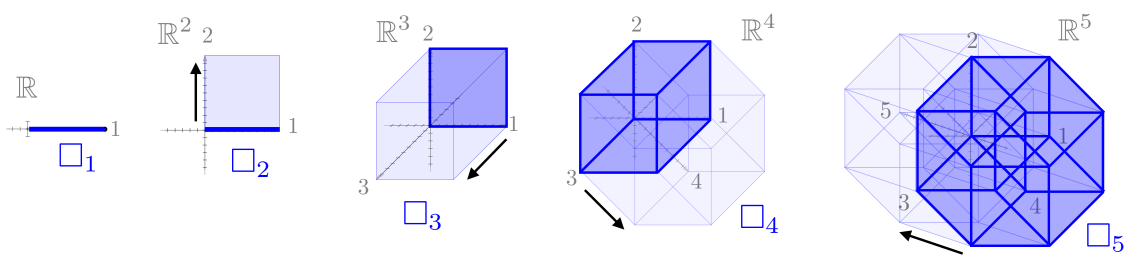
In four dimensions and up, we usually refer to these shapes as *hypercubes*. (For more discussion, see the drawing note below) We now visualize the image of each of these cubes passed through a sample A of the appropriate dimensions.



Note that we have been using cube shapes to visualize vectors through out as well. In visualizing a vector x above, each edge of the cube is a coordinate of the vector x_i and instead of sweeping over the interval $[0, 1]$ each direction of the cube sweeps over $[0, x_i]$.

1.4.1 More Drawing: Hypercubes (Difficulty: 1/10)

Since they are less common to visualize, it is worth noting how we construct higher dimension cubes from lower dimensional ones. Given the unit cube in $n-1$ dimension, we get the nD unit cube by sweeping $\square_{n-1}$ over the interval $[0, 1]$ in the nth dimension. We visualize building up cubes from 1D to 5D in this way below.



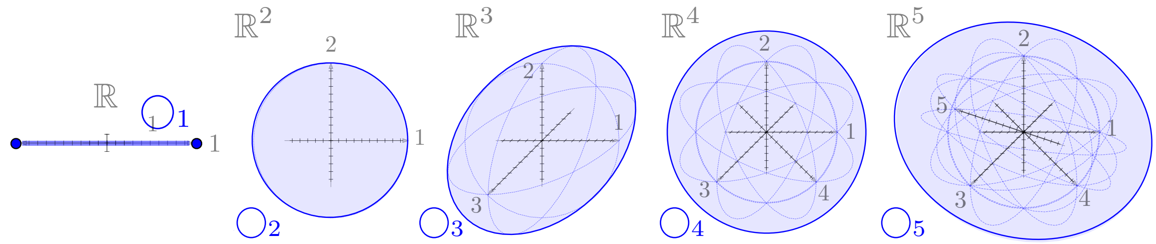
We could continue the same procedure up to arbitrary dimensions, but as the dimension increases the 2D projection we're visualizing becomes less and less useful.

1.5 Sphere

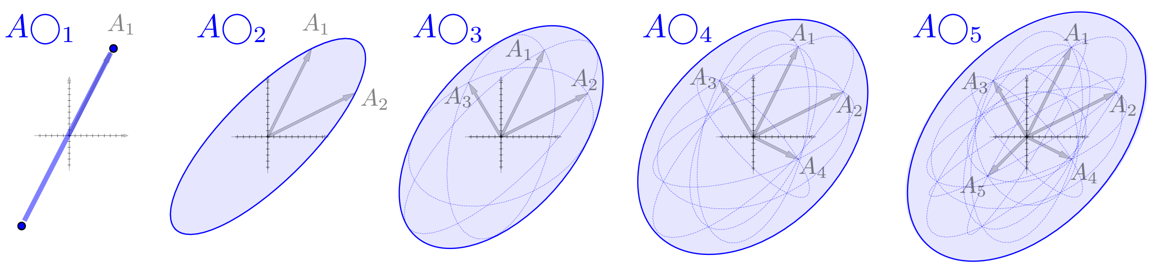
Another important example is the unit sphere in $\mathbb{R}^n$ which we denote as $\bigcirc_n$.

$$\bigcirc_n = \left\{ x \in \mathbb{R}^n \;\middle|\; ||x||_n = 1 \right\} \tag{3}$$

. (Again, sometimes we will denote $\bigcirc_n$ simply as $\bigcirc$ with the dimension provided by context.) We visualize these here, again, in 1D through 5D. The unit sphere in 2D is usually called a unit circle and spheres in dimensions four and up are referred to as *hyperspheres*. (For more discussion of these high dimensional visualizations, see the drawing note below).



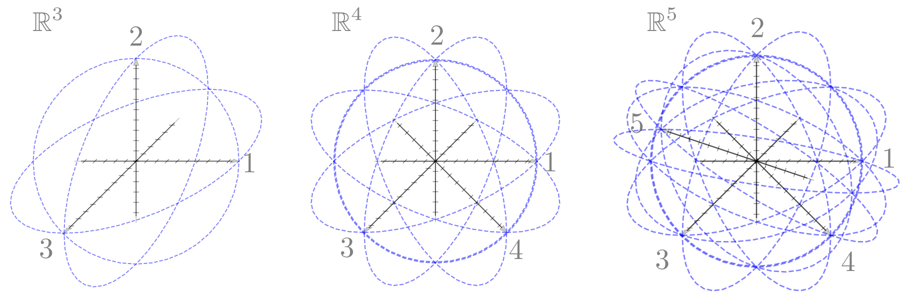
We now draw these spheres passed through matrices.



The image of a sphere through A is naturally an ellipse. The geometry of ellipses plays a major roll in many elements of linear algebra particularly in discussions of quadratic forms, positive definiteness and convex functions. We will give further discussion of the geometry of ellipses in later lectures when we have built up the tools to discuss these subjects.

1.5.1 More Drawing: Hyperspheres (Difficulty: 4/10)

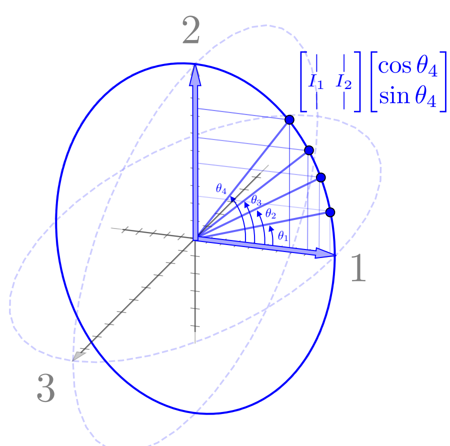
Drawing projections of hyperspheres is slightly more than the other shapes we've beend drawing. Unlike, hypercubes or simplicies there are no clearly defined edges to draw to give a sense for the structure of the object. Instead of edges, we draw various great circles (of dimension 1) around the sphere. A natural choice is the 1D great circles defined by pairs of axes of the sphere shown here in 3D, 4D, and 5D.



For each pair of axes given by the basis vectors $I_j, I_{j'} \in \mathbb{R}^n$, we obtain the points on the great circle around those axes by computing the following

$$Y = \begin{bmatrix} I_j & I_{j'} \end{bmatrix} C \quad \text{with} \quad C = \begin{bmatrix} \cos\theta_1 & \cos\theta_2 & \cdots & \cos\theta_K \\ \sin\theta_1 & \sin\theta_2 & \cdots & \sin\theta_K \end{bmatrix} \tag{4}$$

where the angles $\theta_1, \ldots, \theta_K$ are sampled in the interval $[0, 2\pi]$ We illustrate these sampled points appear for axes I_1 and I_2 here.



It still remains to draw the boundary of the sphere. This corresponds to a great circle perpendicular to the projection plane. Computing this circle is actually not trivial and interesting exercise. Specifically for an nD hypersphere, the boundary in the 2D page is given by the two directions where a unit vector *in nD* goes the furthest on the page. For a set of axis directions given by the columns of $\mathbf{X}_n \in \mathbb{R}^{2\times n}$, we can compute this by taking a singular value decomposition of $\mathbf{X}_n$.

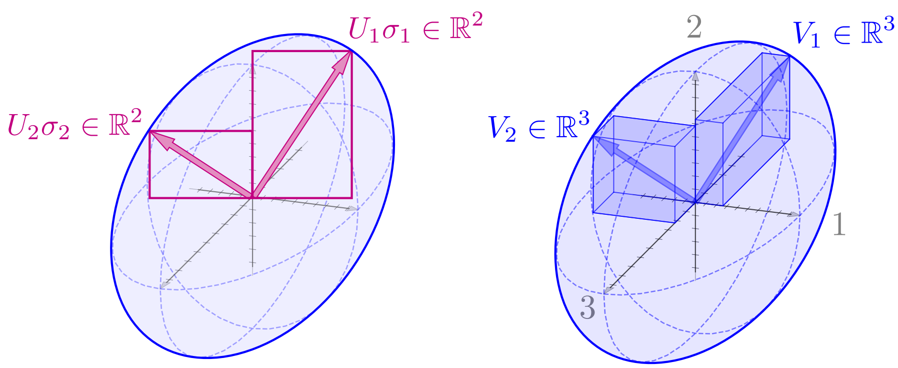
Note the dimensions of each matrix listed below.

$$\mathbf{X}_n = U \left[\begin{array}{cc} \Sigma & \mathbf{0} \end{array} \right] V^\top \tag{5}$$

$$= \begin{bmatrix} | & | \\ U_1 & U_2 \\ | & | \end{bmatrix} \left[\begin{array}{cc|ccc} \sigma_1 & 0 & 0 & \cdots & 0 \\ 0 & \sigma_2 & 0 & \cdots & 0 \end{array} \right] \begin{bmatrix} -V_1^\top- \\ -V_2^\top- \\ \vdots \\ -V_n^\top- \end{bmatrix} \tag{6}$$

$$= U_1 \sigma_1 V_1^\top + U_2 \sigma_2 V_2^\top \tag{7}$$

$U \in \mathbb{R}^{2\times 2}$ orthonormal, $\Sigma \in \mathbb{R}^{2\times 2}$ diagonal and positive, the zero block matrix $\mathbf{0} \in \mathbb{R}^{2\times(n-2)}$, and $V \in \mathbb{R}^{n\times n}$ orthonormal. The interpretation here is crucial. The two vectors singular vectors $U_1, U_2 \in \mathbb{R}^2$ on the left give the maximum gain directions and thus the axes of the hypersphere boundary *on the 2D page*. The corresponding singular vectors $V_1, V_2 \in \mathbb{R}^n$ on the right give the directions in the nD space that generate the appropriate great circle in nD. The other vectors $V_3, \ldots, V_n$ are the depth directions of the axes that will not show in the 2D projection. If we simply want to draw the appropriate circle on the page, we can use the vectors U_1, U_2. In order to get the precise boundary of the hypersphere, we need to scale these vectors by $\sigma_1, \sigma_2 \in \mathbb{R}_+$ respectively. We illustrate the geometry here for this procedure for drawing a sphere in 3D.

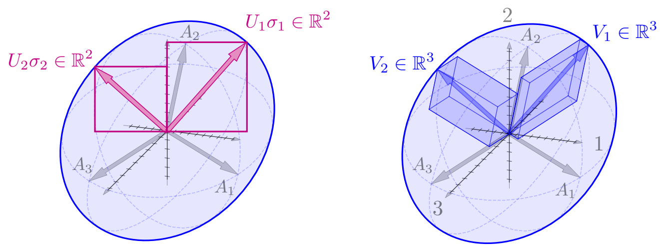


The procedure is the same for higher dimensions. We note that in 3D only, the boundary circle touches the pairwise circles between axes.

Note that if we want to draw the image of a hypersphere through a matrix, ie $A\bigcirc_n$ for $A \in \mathbb{R}^{m\times n}$, we follow the same procedure, but instead of taking the SVD of $\mathbf{X}_n$ we take it of $\mathbf{X}_n A$. The great circle that generates the boundary of the hypersphere may change as it passes through the matrix A. In this case the SVD becomes

$$\mathbf{X}_n A = U \left[\begin{array}{cc} \Sigma & \mathbf{0} \end{array} \right] V^\top \tag{8}$$

as illustrated here.

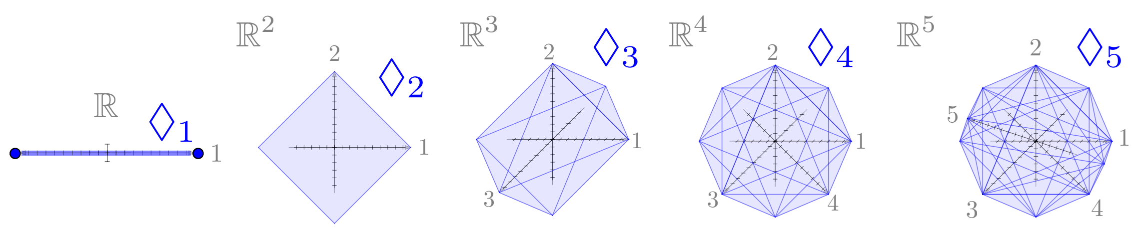


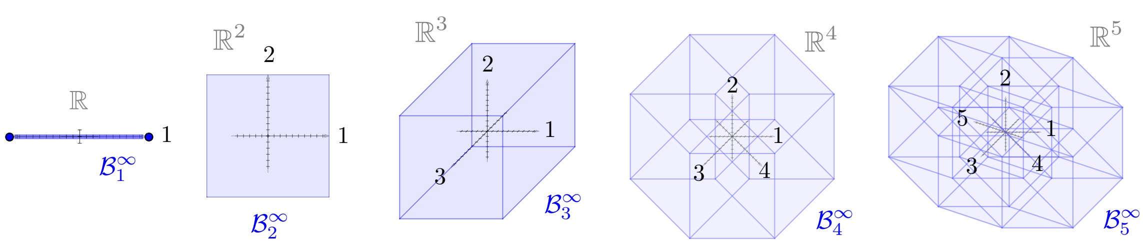
1.6 Norm Balls

We will discuss norms more in a future lecture. Here, we simply given visualizations of several of the most basic norm-balls passed through matrices. The three most import norm balls are the 1-norm, 2-norm and inf-norm balls. We will denote each of these in nD as denote $\mathcal{B}_{1n}$, $\mathcal{B}_{2n}$, and $\mathcal{B}_{\infty n}$. We can define them here.

$$\mathcal{B}^1_n = \left\{x \in \mathbb{R}^n \,\middle|\, ||x||_1 \leq 1\right\} \tag{9}$$
$$\mathcal{B}^2_n = \left\{x \in \mathbb{R}^n \,\middle|\, ||x||_2 \leq 1\right\} \tag{10}$$
$$\mathcal{B}^\infty_n = \left\{x \in \mathbb{R}^n \,\middle|\, ||x||_\infty \leq 1\right\} \tag{11}$$

(Again, sometimes we will suppress the n with dimension implied by context.) Note the that the surface of the 2-norm ball is simply the unit sphere visualized above. Since the 1-norm ball looks like a diamond in 2D, we will sometimes use the notation $\mathcal{B}^1_n = \diamondsuit_n$. The 1-norm ball and inf-norm balls are shown below.





Images of the 1-norm and inf-norm balls balls passed through A are shown below.