Points, Vectors, and Matrices

Example: Projecting a point on a line

> $with(LinearAlgebra)$:

Two (column) vectors.

> $p := \langle 0, 1\rangle ; r := \langle 1, 2\rangle$

$$p := \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$r := \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad (1.1.1)$$

> $r[2]$

$$2 \quad (1.1.2)$$

This is a line (in the plane) in parametric form.

> $x := p + \lambda \cdot r$

$$x := \begin{bmatrix} \text{lambda} \\ 1 + 2\,\text{lambda} \end{bmatrix} \quad (1.1.3)$$

We want to compute shortest distance to point.

> $q := \langle 2, 1\rangle$

$$q := \begin{bmatrix} 2 \\ 1 \end{bmatrix} \quad (1.1.4)$$

> $\lambda_{opt} := \frac{DotProduct(q-p, r)}{DotProduct(r, r)}$

$$\text{lambda}_{opt} := \frac{2}{5} \quad (1.1.5)$$

> $s := subs\left(\lambda = \lambda_{opt}, x\right)$

$$s := \begin{bmatrix} \frac{2}{5} \\ \frac{9}{5} \end{bmatrix} \quad (1.1.6)$$

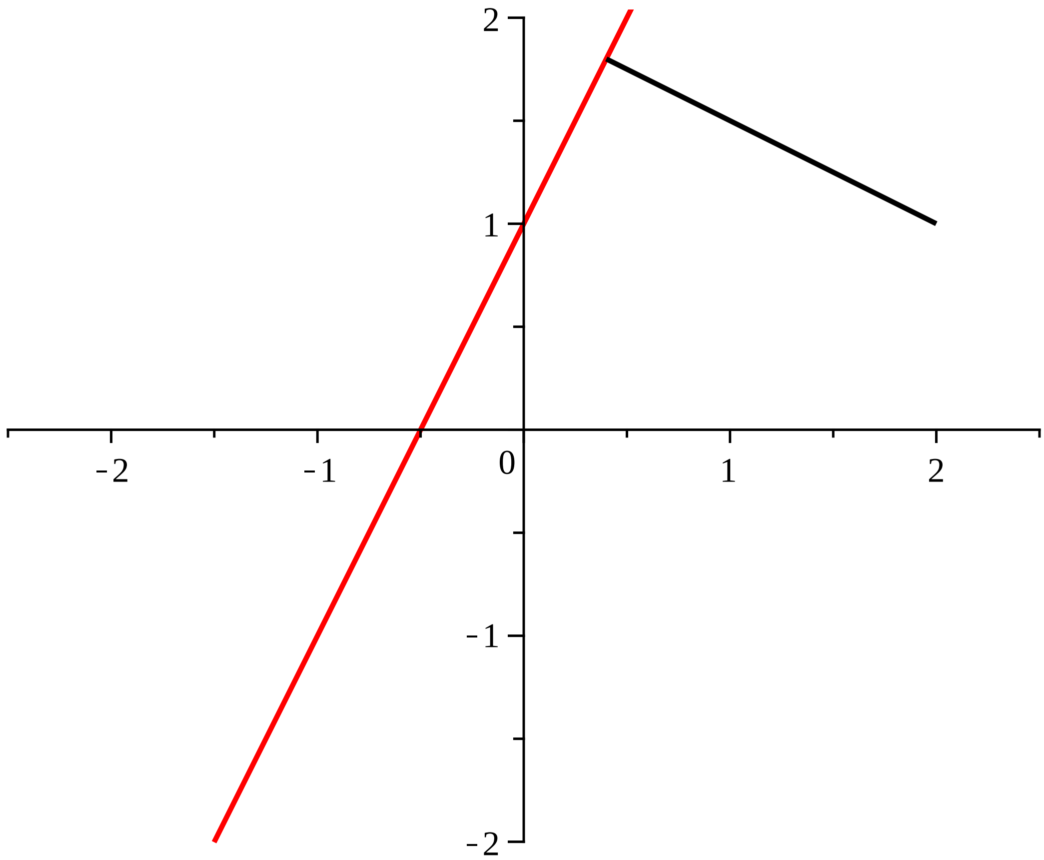
Here is a way to visualize the situation.

> $with(plots)$:

> $LinePlot := plot\left(\left[x[1], x[2], \lambda = -2..2\right], thickness = 2\right)$

$$LinePlot := PLOT(\ldots) \quad (1.1.7)$$

> $display([LinePlot, pointplot([q, s], connect = true, thickness = 2)], scaling = constrained, view = [-2.5..2.5, -2..2])$



Converting a column vector into a row vector.

> $Vector[row](p)$

$$\begin{bmatrix} 0 & 1 \end{bmatrix} \quad \textbf{(1.1.8)}$$

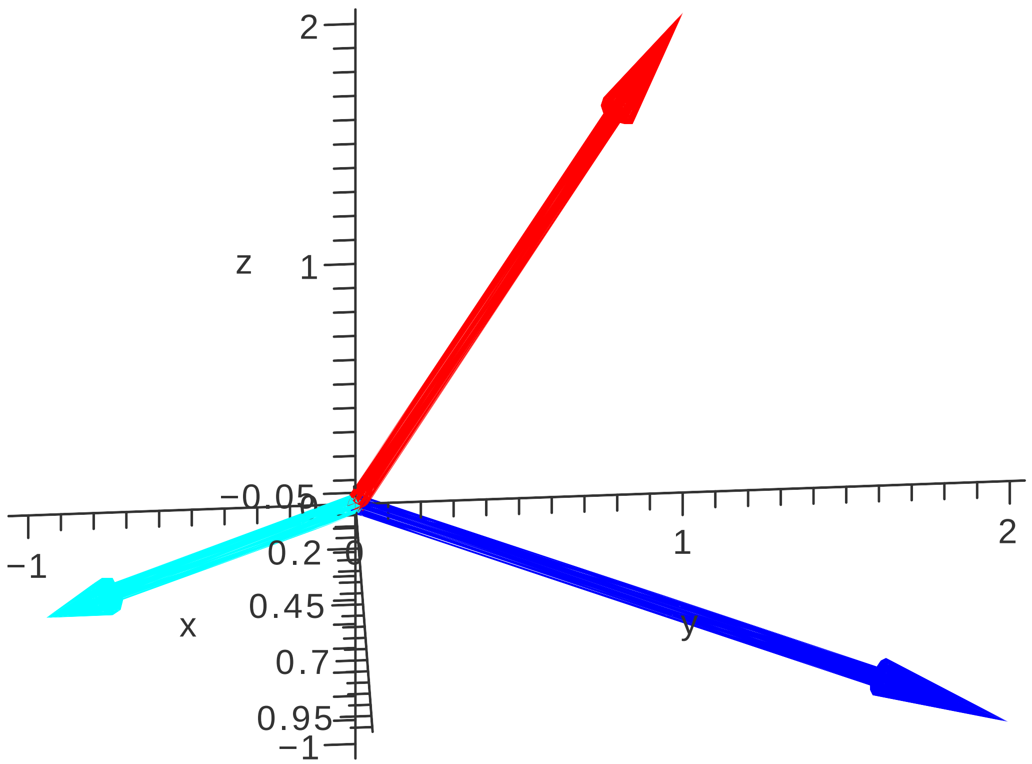
Example: Cross Product

> $CrossProduct\left(\left\langle 1,-1,\frac{1}{2}\right\rangle,\langle 0,2,-1\rangle\right)$

$$\begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} \quad \textbf{(1.2.1)}$$

> $CrossProductPlot(\ \langle 1,-1,1/2\rangle,\langle 0,2,-1\rangle, vectorcolors=[cyan, blue, red]\ );$

The Cross Product of 2 Vectors



Matrixoperations

A matrix is defined as a row vector of column vectors (or as a column vector of row vectors).

> $A := \langle \langle 1|0\rangle, \langle 2|a\rangle \rangle$

$$A := \begin{bmatrix} 1 & 0 \\ 2 & a \end{bmatrix} \tag{1.3.1}$$

> $A + A$

$$\begin{bmatrix} 2 & 0 \\ 4 & 2\,a \end{bmatrix} \tag{1.3.2}$$

> $MatrixInverse(5 \cdot A)$

$$\begin{bmatrix} \frac{1}{5} & 0 \\ -\frac{2}{5\,a} & \frac{1}{5\,a} \end{bmatrix} \tag{1.3.3}$$

> $Determinant(A)$

$$a \tag{1.3.4}$$

The "." is the notation for the matrix multiplication as well as the scalar product.

> $A.\langle 4, 5\rangle$

$$\begin{bmatrix} 4 \\ 8+5a \end{bmatrix} \tag{1.3.5}$$

> ⟨4, 5⟩.⟨2, 3⟩

$$23 \tag{1.3.6}$$

Random / Large Vectors and Matrices

> *RandomVector*(2)

$$\begin{bmatrix} -31 \\ 67 \end{bmatrix} \tag{1.4.1}$$

> *RandomVector*[*row*](2)

$$\begin{bmatrix} 44 & 92 \end{bmatrix} \tag{1.4.2}$$

> *rM* := *RandomMatrix*(100, 100)

$$rM := \begin{bmatrix} \textit{100 x 100 Matrix} \\ \textit{Data Type: anything} \\ \textit{Storage: rectangular} \\ \textit{Order: Fortran\_order} \end{bmatrix} \tag{1.4.3}$$

> *Determinant*(*rM*)

160235824138146607080131911958393421635589016854563001198022068110124\ **(1.4.4)**
843259542570436946123268053277960234121848886687485144846350812357\
922585306935358393867440720586313912270972912813060193192077037922\
91317730454990701190798257753929352909822386368219424

Basis of Subspaces

Subspaces are defined by a (finite) set or list of generators.

> *v1* := ⟨1|0|0⟩ :
v2 := ⟨0|1|0⟩ :
v3 := ⟨0|0|1⟩ :
v4 := ⟨0|1|1⟩ :
v5 := ⟨1|1|1⟩ :
v6 := ⟨4|2|0⟩ :
v7 := ⟨3|0|−1⟩ :

> *Basis*([*v1*, *v2*, *v2*]);

$$\left[\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}\right] \tag{2.1}$$

> *Basis*({*v4*, *v6*, *v7*});

$$\left\{\begin{bmatrix} 3 & 0 & -1 \end{bmatrix}, \begin{bmatrix} 4 & 2 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 1 \end{bmatrix}\right\} \tag{2.2}$$

Intersection of subspaces.

> *IntersectionBasis*([[*v1*, *v2*], [*v4*, *v7*]])

$$\left[\begin{bmatrix} 3 & 1 & 0 \end{bmatrix}\right] \tag{2.3}$$

Sum of subspaces.

> *SumBasis*([[*v1*, *v2*], [*v4*, *v7*]])

$$\left[\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 1 \end{bmatrix}\right] \tag{2.4}$$

Systems of Linear Equations

Interactive Solutions/Maplets

> $with(Student[LinearAlgebra]):$

> $A := \langle\langle 1|1|1\rangle, \langle 4|4|3\rangle, \langle 2|1|1\rangle\rangle;\ b := \langle 1, 5, 2\rangle$

$$A := \begin{bmatrix} 1 & 1 & 1 \\ 4 & 4 & 3 \\ 2 & 1 & 1 \end{bmatrix}$$

$$b := \begin{bmatrix} 1 \\ 5 \\ 2 \end{bmatrix} \qquad \textbf{(3.1.1)}$$

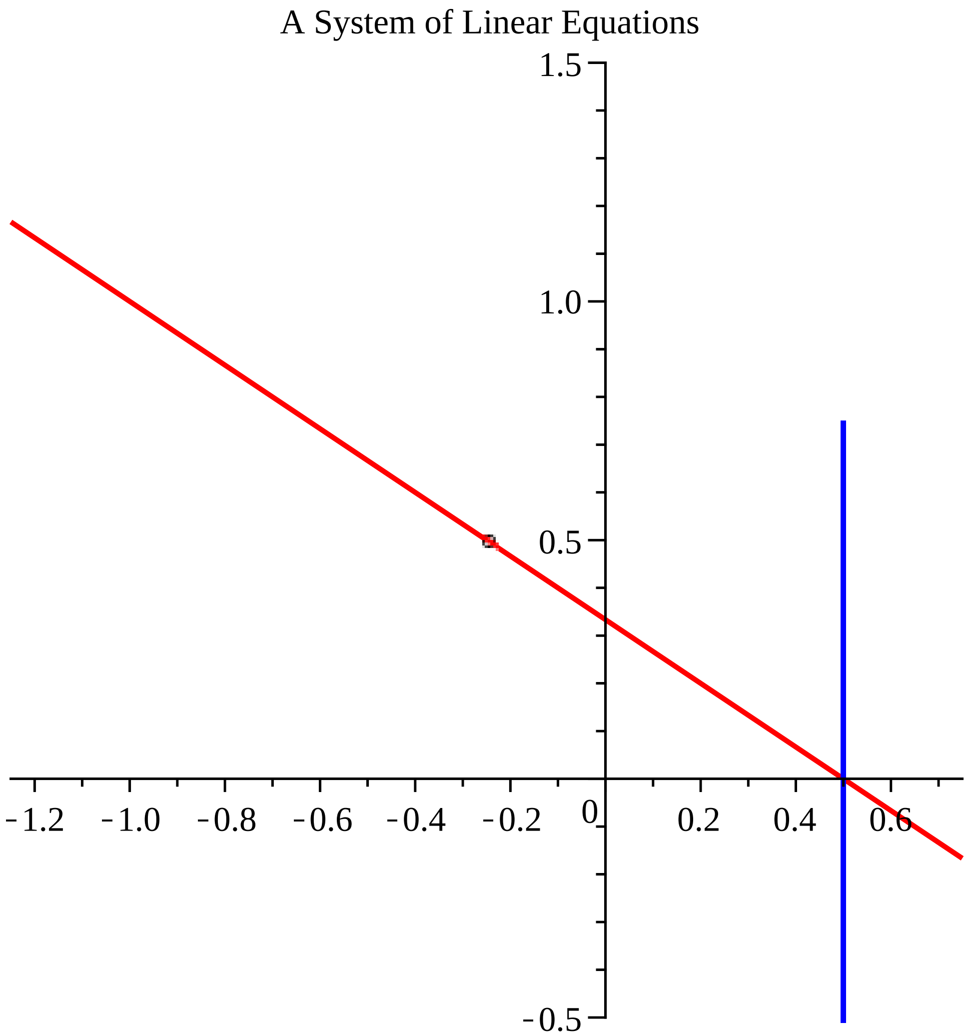
> $GaussJordanEliminationTutor(A, b);$

> $LinearSolve(A, b)$

$$\begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix} \qquad \textbf{(3.1.2)}$$

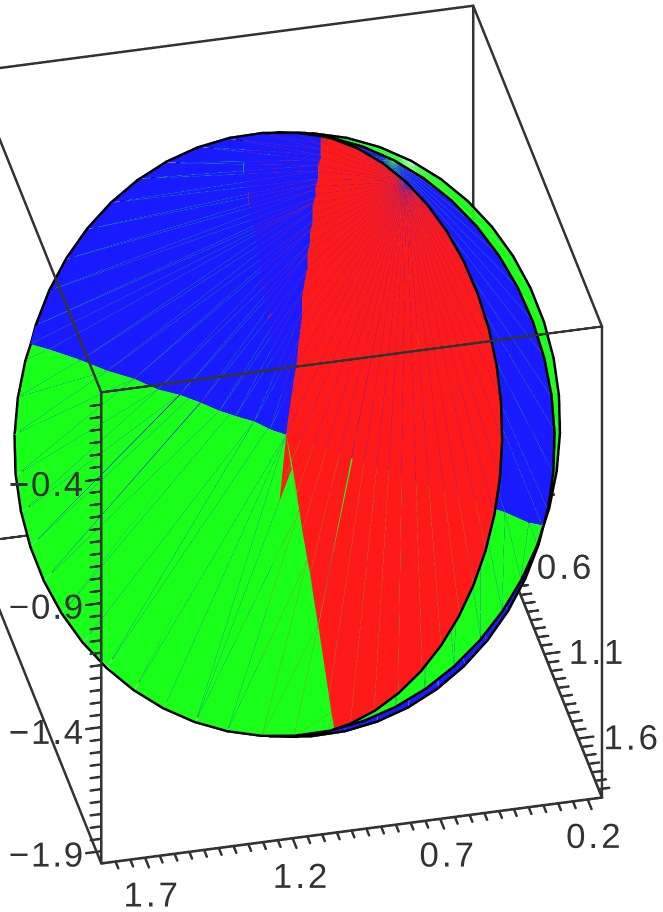
Visualization

> $LinearSystemPlot(\langle\langle 2|3|1\rangle, \langle 0|-2|-1\rangle\rangle, colors = [red, blue], thickness = 2)$



> $LinearSystemPlot(\langle A|b\rangle)$

A System of Linear Equations



Numerical Instability

The following defines a 2x2-matrix A(k) and a vector b(k) for each number k.

> $A := k \rightarrow \langle \langle 1-10\wedge(-k) | 1-2*10\wedge(-k) \rangle, \langle 1-2*10\wedge(-k) | 1-3*10\wedge(-k) \rangle \rangle;$

$$A := k \rightarrow \langle\langle 1 - 10^{-k} | 1 - 2\ 10^{-k} \rangle, \langle 1 - 2\ 10^{-k} | 1 - 3\ 10^{-k} \rangle\rangle \quad (3.3.1)$$

> $b := k \rightarrow MatrixVectorMultiply(A(k), \langle 1, 0 \rangle);$

$$b := k \rightarrow LinearAlgebra\text{:-}MatrixVectorMultiply(A(k), \langle 1, 0 \rangle) \quad (3.3.2)$$

This is the case k=1.

> $A(1), b(1)$

$$\begin{bmatrix} \frac{9}{10} & \frac{4}{5} \\ \frac{4}{5} & \frac{7}{10} \end{bmatrix}, \begin{bmatrix} \frac{9}{10} \\ \frac{4}{5} \end{bmatrix} \quad (3.3.3)$$

> $Determinant(A(1))$

$$-\frac{1}{100} \quad (3.3.4)$$

> $LinearSolve(A(1), b(1))$

$$\begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (3.3.5)$$

For each k the matrix A(k) is invertible.

> *Determinant*(*A*(*k*))

$$-\left(10^{-k}\right)^2 \tag{3.3.6}$$

Hence the corresponding system has a unique solution.

> *LinearSolve*(*A*(*k*), *b*(*k*))

$$\begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{3.3.7}$$

Relying on numerical methods (floating point representation of coefficients) gives imprecise results ...

> *LinearSolve*(*evalf*(*A*(1)), *evalf*(*b*(1)), *method* ='LU')

$$\begin{bmatrix} 0.99999999999999756 \\ 2.77609966264910728\ 10^{-15} \end{bmatrix} \tag{3.3.8}$$

... and the precise output may depend on the method chosen.

> *LinearSolve*(*evalf*(*A*(1)), *evalf*(*b*(1)), *method* ='QR')

$$\begin{bmatrix} 0.99999999999998824 \\ 1.35712161611153447\ 10^{-14} \end{bmatrix} \tag{3.3.9}$$

But things can get worse.

> **for** *k* **from** 2 **to** 10 **do** *LinearSolve*(*evalf*(*A*(*k*)), *evalf*(*b*(*k*))) **end do**;

$$\begin{bmatrix} 0.99999999999946610 \\ 5.39363475757225108\ 10^{-13} \end{bmatrix}$$

$$\begin{bmatrix} 0.99999999997662816 \\ 2.33953481587618971\ 10^{-11} \end{bmatrix}$$

$$\begin{bmatrix} 1.00000000391318689 \\ -3.91357844568900014\ 10^{-9} \end{bmatrix}$$

$$\begin{bmatrix} 1.00000007480845365 \\ -7.48092018141465685\ 10^{-8} \end{bmatrix}$$

$$\begin{bmatrix} 0.999976849759069198 \\ 0.0000231502640810803577 \end{bmatrix}$$

$$\begin{bmatrix} 0.999203113898887608 \\ 0.000796886180801006124 \end{bmatrix}$$

$$\begin{bmatrix} 1.11008676679501073 \\ -.110086767895878526 \end{bmatrix}$$

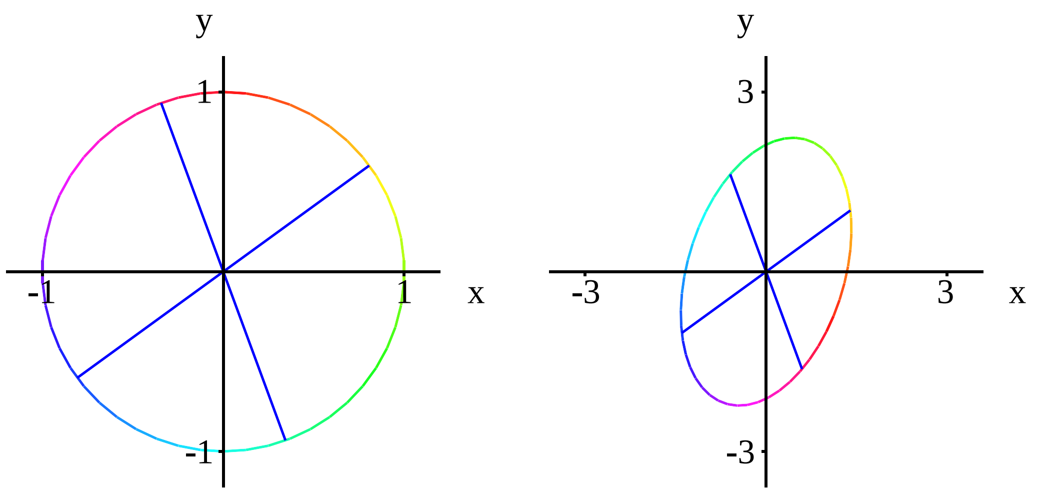
$$\begin{bmatrix} 0.991287512109390256 \\ 0.00871248789932236277 \end{bmatrix}$$

$$\begin{bmatrix} 0.9999999999 - 0.9999999998\ \_t23_1 \\ \_t23_1 \end{bmatrix} \tag{3.3.10}$$

Linear Transformations

> $LinearTransformPlot(\langle\langle 1, 2\rangle | \langle 1, -1\rangle\rangle);$

The Image of the Unit Circle
In the Plane



> $LinearTransformPlot(\langle\langle 1, 2, 3\rangle | \langle -2, 2, -1\rangle | \langle 3, -3, 0\rangle\rangle, showeigenvectors = true,$
> $nullspaceoptions = [thickness = 3, color = black], spheregrid = 10);$

The Image of the Unit Sphere
In 3−Space