This system has infinitely many solutions, given parametrically by $(1 - x_3, 1 + x_3, x_3)$. Geometrically, this is a line common to all three planes; see Figure 3.6.

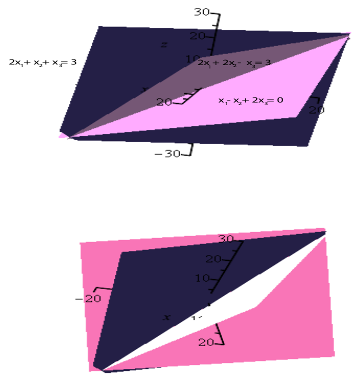

Figure 3.6: The linear system $x_1 + 2x_2 - x_3 = 3$, $2x_1 + x_2 + x_3 = 3$, $x_1 - x_2 + 2x_3 = 0$ has the red line common to all three planes.

Under the above interpretation, observe that we are focusing on the *rows* of the matrix A, rather than on its *columns*, as in the previous interpretations.

Another great example of a real-world problem where linear algebra proves to be very effective is the problem of *data compression*, that is, of representing a very large data set using a much smaller amount of storage.

Typically the data set is represented as an $m \times n$ matrix A where each row corresponds to an n-dimensional data point and typically, $m \geq n$. In most applications, the data are not independent so the rank of A is a lot smaller than $\min\{m, n\}$, and the the goal of *low-rank decomposition* is to factor A as the product of two matrices B and C, where B is a $m \times k$ matrix and C is a $k \times n$ matrix, with $k \ll \min\{m, n\}$ (here, $\ll$ means "much smaller than"):

$$\begin{pmatrix} & & \\ & A & \\ & m \times n & \\ & & \end{pmatrix} = \begin{pmatrix} & & \\ & B & \\ & m \times k & \\ & & \end{pmatrix} \begin{pmatrix} & C & \\ & k \times n & \end{pmatrix}$$

Now it is generally too costly to find an exact factorization as above, so we look for a low-rank matrix A' which is a "good" *approximation* of A. In order to make this statement precise, we need to define a mechanism to determine how close two matrices are. This can be done using *matrix norms*, a notion discussed in Chapter 9. The norm of a matrix A is a