In the following exercises, you will analyze the output of the mystery function `mystery = function(x) matrix(c(cos(x), -sin(x), sin(x), cos(x)), nrow=2)`.

- Is the output of `mystery()` *periodic* in some sense with respect to its inputs? Check if this holds in practice; if there's a discrepancy, explain it. (*Hint:* If you aren't familiar with the behavior of the `sin()` and `cos()` functions, graph a scatterplot of their values against `seq(0, 10, 0.01)`.)[1]
- Write a function to turn lists of length 4 into 2-by-2 matrices, forming a list-matrix capable of holding different data types. (*Hint:* Use `dim()`.) Speculate on some use cases of list-matrices.
- On the 2D plane, we can identify the *point* $(x, y)$ with the *column vector* `matrix(c(x,y), nrow=2, ncol=1)`. First, write a function which takes in a list of column vectors, internally adds each one as a row of a two-column dataframe, and then uses `ggplot()` to graph a *scatterplot* of all the points in the input list. Second, modify this function to accept an argument `x` and so that for every column vector `c` in its list of points, instead of putting the data in `c` directly into the data frame, it does so for the product `mystery(x) %*% c` (for the `mystery()` function defined in a previous exercise). Experiment with different values of `x` and come up with a geometric understanding of how the output graph changes as you modify `x`.[2]

[1] We have `mystery(0) != mystery(2*pi)` because of floating-point imprecision.

[2] Calling `mystery(x)` returns a 2-dimensional rotation matrix corresponding to rotation through the angle `x` (given in radians).