Conjugation as change of representation

Ibrahim Tencer · 3 May 2024

https://functor.network/user/881/entry/352

Have you ever wondered why group element conjugation is so ubiquitous, and what it means?

Spatial transformations are one of the primary examples that we look to in order to give algebraic concepts geometric meaning. And that means matrices.

Matrices

Matrices are of course basis-dependent, and to change the basis a square matrix is in, you conjugate by a change-of-basis matrix. But why is that?

More formally, given a linear transformation f we write $[f]_{M,N}$ for the matrix for f that has M as source basis and N as target basis. Then it is possible to show that matrix multiplication is just composition of the corresponding linear transformations, and that the identity matrix does in fact represent the identity. We will not show this here. Symbolically:

$$[f]_{F,G}[g]_{E,F} = [fg]_{E,G}$$

and

$$[1]_{E,E} = \mathrm{Id}$$

(Note the wonky order of the bases here from using standard composition order.)

This should remind you of a functor — more on that later.

Change of basis is a process that doesn't change what the matrix does geometrically, it only changes the *representation*. This is called a passive transformation, as opposed to an active one. It is the confusion between passive and active transformations that obscures what is going on here.

So the problem is, if $[f]_{E,E}$ is a square matrix for a linear transformation f in basis E, then how do we find $[f]_{F,F}$ for another basis F of the same space?

The basic idea is that the change of basis matrix is just a matrix for the identity map, except using E as the source basis and F as the target basis. The identity map does not do any real change, so it's only a change of representation.

In particular we obtain:

$$[1]_{G,K}[f]_{E,G}[1]_{F,E} = [f]_{F,K}$$

This is for general change of basis, where f is not necessarily a self-map and we change the basis separately on the two sides. But in the case where f is a self-map on V and we use the same basis for V as source and target we get:

$$[1]_{E,F}[f]_{E,E}[1]_{F,E} = [f]_{F,F}$$

Now, to show that $[1]_{E,F} = [1]_{F,E}^{-1}$:

$$\begin{aligned} [1]_{E,F}[1]_{F,E} &= [1]_{E,E} && \text{by the composition law} \\ &= \mathrm{Id} && \text{by the unit law} \end{aligned}$$

So changing basis uniformly for a square matrix necessitates using conjugation.

This notation makes the difference between the active (or physical) part and the passive (or informational) part clear.

Permutations

Conjugation is used the same way for permutations: to change the labeling of a permutation you conjugate it by another one. For example, (12)(1425)(12) = (1524), and (1524) = (2415) is the same as (1425) except with the 1 and 2 switched. And the same dual role is present: a permutation can represent both actual change (i.e. rearranging the points of the set) and change of labels.

The essential concept is the same so let’s try to encapsulate the general pattern.

Basically we have two categories: one is the base category of systems and actual (or “physical”) processes on them, call it C. And another is a category of *representations* of systems and processes, say R. And we have some functor $F : R \to C$ that gives the actual system or process being represented.

In the example of matrices we have $Vect$, the category of vector spaces, as the base category. To form R, consider Mat which we can define as the full subcategory of vector spaces of the form F^n where F is the base field. (This abstracts away the actual definition of a matrix in terms of entries but it will suffice for our purposes.) Then the objects of the category of representations are vector spaces with an isomorphism to a free vector space, and the morphisms are commutative squares between these (or alternatively, just maps between the vector spaces without any compatibility conditions). So we actually get a span of functors, one to $Vect$ (the physical part) and one to Mat (the representation).

What about permutations? Permutations are always self-maps, so we take $Core(FinSet)$ as the base category (using finite sets for clarity, though this is not so hard to generalize to ordinals). And as before, we take a subcategory Seq of sets of the form $[n] = \{0, ..., n-1\}$, i.e. all finite ordinals with isomorphisms, and take the representation category as finite sets equipped with an isomorphism to a finite ordinal. (A finite set is usually defined as a set with such an isomorphism in the first place.)

The analogy with matrices suggests using a notation that applies to maps between any two finite ordinals, not just self-maps. Cycle notation doesn't seem to work given that it assumes you can feed outputs back into the function itself, but the two-row notation will, where e.g. (1245) becomes

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 4 & 3 & 5 & 1 \end{pmatrix}.$$

And any way of writing a function down between finite sets will use some kind of notation in language using sequences of symbols from some finite alphabet — these will also count as a representation.

The abstracted argument

Suppose that F is faithful and $FX = FY = S$, i.e. X and Y both represent the same system S. Then if $i : X \to Y$ represents 1_S, and so does $j : Y \to X$, then

$$F(ji) = F(j)F(i) = 1_S = F(i)F(j) = F(ij)$$

so by faithfulness $ji = 1_X$ and $ij = 1_Y$, i.e. $j = i^{-1}$.

And then if $g : X \to X$ represents $f : S \to S$, then

$$F(igi^{-1}) = F(i)F(g)F(i)^{-1} = f$$

so igi^{-1} represents f on Y. □

This can be easily generalized to changing the representation of the source and target independently, but this case shows directly how we get conjugation.

We could require the functor to be full, faithful, and essentially surjective as the above examples are — a weak equivalence, which is an equivalence of categories given the axiom of choice but it may not have a canonical weak inverse equivalence. In the case of vector spaces, it means that they do not always have a canonical basis. A weak equivalence represents all systems and processes faithfully, so g and igi^{-1} would be the only possible representations of f with respect to those representations of the system, and every actual process corresponds to some change of representation.

Notice also that any group can be realized as a permutation group (via a functor to Set), so conjugation in any group can be interpreted as a change of representation. We can also compose this functor with the free vector space functor to realize it as a group of permutation matrices, so that every conjugation is a change of basis.