Reading 04: Types of Induction

We have seen three different flavors of induction. Weak, strong, and structural induction. It is tempting to treat these techniques as three separate techniques, each applicable to their own problems. There are certainly problems where one method is easier, but the methods are all (as a formal matter at least) completely equivalent in power. If you can write an inductive proof with one of the techniques it can be transformed into any of the others.

That doesn't mean you **should** change which version you're using, but this note will show that you always **could** switch.

1. Weak to Strong Conversions

Suppose you want to show $P(n)$ holds for all $n \in \mathbb{N}$. And furthermore suppose you've written a weak inductive proof of that claim. Could you have done it with strong induction instead? Of course! All you need to do is assume more things in your IH (you don't need to sue them in the inductive step just because you supposed them). Just rewrite the hypothesis to say $P(0), ..., P(k)$ and you're done!

But what about the other way? If you've written a strong inductive proof that $P(n)$ is true for all $n \in \mathbb{N}$ can you write a weak inductive proof instead? There are claims for which you can't *just* change the phrasing of your hypothesis (like the proof of the Fundamental Theorem of Arithmetic from lecture). But you can write a weak inductive proof with one easy trick! Define $Q(n) = P(0) \wedge P(1) \wedge \cdots \wedge P(n)$. Does that look familiar...it's what you'd usually write in a strong hypothesis. Now when you suppose (just) $Q(k)$, you've supposed exactly what you would to prove $P(n)$ with strong induction. Supposing just one value of $Q()$ is all the necessary values of $P()$! We instead can show $Q(n)$ is true for all $n \in \mathbb{N}$ by (weak) induction on n. If you've proven $Q(n)$ for all n, then you definitely have $P(n)$ for all n too.

2. But What About Structural?

Weak and strong inductive proofs "look like" each other, but structural induction proofs *look* really different. Are they? It turns out they really aren't. We'll describe how to convert from weak or strong inductive proofs to structural ones and back. It's just defining the right predicates and adding some boilerplate!

2.1. Weak to Structural

Let's say that we've written a weak inductive proof that $P(n)$ is true for all natural numbers n. How would we write a corresponding structural proof. In trying to do so, we hit a problem immediately – what recursively defined set should we be using? Well, when we finish the structural induction proof, we'll write down $P(x)$ holds for all $x \in ?$ where $?$ is the set we define recursively. If we want to write an equivalent proof, then $?$ should be $\mathbb{N}$! How can we define the natural numbers recursively? Here's one option

Basis: $0 \in \mathbb{N}$
Recursive: If $x \in \mathbb{N}$, then $x + 1 \in \mathbb{N}$.

This isn't the only possible definition, but it is the one that works best here. It also turns out to be the "standard" one.[1]

Now, let's convert a weak inductive proof into a **structural** induction proof that $P(n)$ holds for all n.

We start by showing the predicate on the basis of our recursive definition; that's $P(0)$. Which was the base case of our weak inductive proof.

Then we need to show if the predicate is true for the element already in the set, then it's also true for the next element. Here, our inductive hypothesis is $P(k)$ for some $k \in \mathbb{N}$. and our inductive step will be to show the predicate holds for $P(k + 1)$. Also **exactly** what we did in our weak inductive proof. It turns out that weak induction **is** structural induction on the natural numbers!

[1]Mathematicians use the Peano Axioms to define natural numbers (and equals) Axioms 1 and 6 in the wikipedia article correspond to our basis and recursive steps.

2.2. Structural to Strong

So if we've written a structural induction proof, can we write a traditional induction proof? Yes...but we'll need some creativity. We'll need some extra machinery. Remember our intuition for structural induction was to think of adding elements in phases: first the basis elements, then the ones that can be built with just the basis elements, then the ones built from those, etc.

For a recursively defined set S, define the order of an element as follows:

$$\text{order}(x) = \begin{cases} 0 & \text{if } x \text{ is a basis element} \\ 1 + \max\{y : y \text{ is used to build } x\} & \text{if } x \text{ is not a basis element} \end{cases}$$

For our recursive definition of strings, the order of a string turns out to be its length. For our recursive definitions of trees, the order of a tree will be its height.

Now, if we have a structural proof that shows $P(x)$ holds for all $x \in S$, how do we write a weak inductive proof? We can't keep the same predicate $P()$ (or at least we might not be able to). S could contain any type of object (strings, trees, ...) but to do weak induction, we need to input a natural number. What number can we choose? The order! It puts the elements in the right order to do induction. Define the predicate $Q(n)$ to be "for all elements of S of order n, $P(n)$ holds." We'll have to be careful in proving this predicate – it has a quantifier inside – but we can do it.

Our (strong) inductive proof starts with $Q(0)$. We need to show for all elements, x, of order 0, that $P(x)$ holds. So we'll introduce an arbitrary element of order 0. What does that look like? Well we have a list of them! It's one of the elements named in the basis step of S. Our structural induction proof gave a separate argument for each of these, so our strong inductive proof can too! We'd probably call it "proof by cases" instead of separate base cases, but it's just a matter of copy-pasting the arguments.

Our inductive hypothesis will suppose $Q(0) \wedge Q(1) \wedge \cdots \wedge Q(k)$ for an arbitrary $k \geq 0$. With the definition of $Q(\cdot)$, notice that we suppose all elements x of order $0, 1, 2, ..., k$ all have $P(x)$ be true.

Now, what do we do in our inductive step? Our goal is to show $Q(k+1)$, that is "for all elements, y, of order $k+1$ in the set S, $P(y)$ is true." So we start our proof with "let y be an arbitrary element of S of order $k+1$. What does an arbitrary element of order $k+1$ look like? Well $k+1 \geq 1$, so this element is order at least 1. That means it must not be a basis element. How did we build it? If we have more than one rule, we can't be sure which one we used...but we do know that the order of all elements of S we used to make y must be order at most k (otherwise the order of y would be more than $k+1$). So what does our proof look like structurally? Well we introduced the arbitrary y, then we'll divide into cases based on which rule we used (one case for each recursive rule). Then...we just copy paste the steps from the structural induction proof. That's exactly what we need to show! The only thing we need to check is: when we say "by IH" is it really in the IH? The answer is yes! For any x we use in the structural induction proof it was already in the set in the structural inductive version. Which means the order of x is less than the order of y – so $P(x)$ appears in the inductive hypothesis. Again, it's just a matter of copy-pasting the arguments.

It turns out that a structural induction proof **is** a proof by strong induction of that predicate $Q()$ we defined. We just have slightly different boilerplate because we're phrasing the predicate differently.

3. Final Thoughts

You might worry about the last section going from structural to strong, and structural to weak, but using section 1 we can finish converting from any method to any other.

Are there any practical implications of this equivalence? Well, you should not use this note as an excuse to only learn one type of induction. That would make your life harder, not easier. But if you feel like you're "better" at one type of induction than the others, this note is an indication that you can be just as good at the others! Induction is one core idea, with recursive thinking you can get any of these proofs depending on the context of the problem, but regardless of the end result your starting point should be "where is the recursion, and how do I use it?"