Lecture 3: Induction

But then what is outduction?

Why Induction?

Recall from last lecture the triangle inequality:
Theorem: Let $x, y \in \mathbb{R}$. Then $|x + y| \leq |x| + |y|$.

Consider this generalized form:
Theorem: Let $n \in \mathbb{N}$, $n \neq 0$. Then $\forall x_1, \ldots, x_n \in \mathbb{R}$, $|x_1 + \ldots + x_n| \leq |x_1| + \ldots + |x_n|$.

Casework possible, but very tedious.
But what if $|x_1 + \ldots + x_{n-1}| \leq |x_1| + \ldots + |x_{n-1}|$?

By original theorem,

$$\begin{aligned}|(x_1 + \ldots + x_{n-1}) + x_n| &\leq |x_1 + \ldots + x_{n-1}| + |x_n| \\ &\leq (|x_1| + \ldots + |x_{n-1}|) + |x_n|\end{aligned}$$

Induction Introduction

Principle of Induction: To prove $\forall n \in \mathbb{N}\ P(n)$, suffices to prove

(1) $P(0)$
(2) $\forall k \in \mathbb{N}\ [P(k) \implies P(k+1)]$

(1) is *base case* and (2) is *inductive step.*[1]

Why does this work?

Certainly, $P(0)$ is true.
If $P(0)$ is true, then $P(1)$ is.
If $P(1)$ is true, then $P(2)$ is.
...

[1]Supposing that $P(k)$ holds called the *inductive hypothesis*.

Generalized Triangle Inequality

Let's apply this formally:
Theorem: Let $n \in \mathbb{N}$, $n \neq 0$. Then $\forall x_1, \ldots, x_n \in \mathbb{R}$, $|x_1 + \ldots + x_n| \leq |x_1| + \ldots + |x_n|$.

Base Case $(n = 1)$:[2]

- Need $|x_1| \leq |x_1|$ ✓

Inductive Step:

- Suppose $|x_1 + \ldots + x_k| \leq |x_1| + \ldots + |x_k|$
- By the original triangle inequality, $|(x_1 + \ldots + x_k) + x_{k+1}| \leq |x_1 + \ldots + x_k| + |x_{k+1}|$
- Combining these yields $|x_1 + \ldots + x_{k+1}| \leq |x_1| + \ldots + |x_{k+1}|$

[2]We don't always have to use 0 for our base case!

Another Example

Theorem: For all $n \in \mathbb{N}$, $\sum_{i=0}^{n} i = \frac{n(n+1)}{2}$.

Base Case$(n = 0)$:

- $\sum_{i=0}^{0} i = 0 = \frac{0(0+1)}{2}$

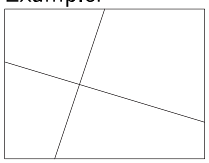
Inductive Step:

- Suppose that $\sum_{i=0}^{k} i = \frac{k(k+1)}{2}$
- Then $\sum_{i=0}^{k+1} i = \sum_{i=0}^{k} i + (k+1) = \frac{k(k+1)}{2} + (k+1)$
- This equals $\frac{(k+1)(k+2)}{2} = \frac{(k+1)((k+1)+1)}{2}$

Two Coloring a Map

How many colors do we need to color a map (such that adjacent regions are different colors)?

Later: 5 colors is enough[3]
Today: simplification where boundaries are lines.
Example:



In this case, 2 colors will suffice!

[3]In fact, 4 colors suffices

Two Color Proof

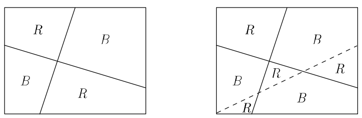
Theorem: Let $P(n)$ be "any map with n lines can be two-colored". Then $\forall n \in \mathbb{N}\ P(n)$.

Base Case($n = 0$):

- Just one region, so just one color

Inductive Step:

- Suppose that $P(k)$ is true
- Given map with $k+1$ lines, remove one line
- $P(k)$ true, so result can be two-colored
- Add line back, flip all colors on one side of it



What If Induction Fails?

Theorem: For all natural numbers $n \geq 1$, the sum of the first n odd numbers is a perfect square.

Base Case ($n = 1$):

- The summation is just 1 ✓

Inductive Step:

- Suppose the sum of the first k odds is m^2
- The $(k+1)$st odd number is $2k+1$
- Sum of the first $k+1$ odds is $m^2 + 2k + 1$
- hmm....

Knowing $P(k)$ isn't enough to get to $P(k+1)$!
Seem to be stuck :(

Look For a Pattern...

Let's consider a couple of the smaller cases:

- $n = 1$: $1 = 1^2$
- $n = 2$: $1 + 3 = 4 = 2^2$
- $n = 3$: $1 + 3 + 5 = 9 = 3^2$
- $n = 4$: $1 + 3 + 5 + 7 = 16 = 4^2$

Hmm, looks like the sum always works out to n^2...
Try proving it!

...and Prove It!

Theorem: For all natural numbers $n \geq 1$, the sum of the first n odd numbers is n^2.

Base Case($n = 1$):

- The summation is just 1, which is indeed 1^2

Inductive Step:

- Suppose the sum of the first k odds is k^2
- The $(k+1)$st odd number is $2k+1$
- So the sum of the first $k+1$ odds is $k^2 + 2k + 1 = (k+1)^2$

Wait–this wasn't the theorem we wanted to prove!
But new theorem implies old one.

Strengthening the Inductive Hypothesis

What we just did is called *strengthening the inductive hypothesis*.

General form: want to prove $\forall n\ P(n)$, instead prove $\forall n\ Q(n)$, where $Q(n) \implies P(n)$

Seems like this should be harder to prove...
...but $Q(k)$ can give us more information!

Look for patterns when strengthening.

Another Strengthening Example

Theorem: For all natural numbers n, $\sum_{i=0}^{n} 2^{-i} \leq 2$.

Base Case($n = 0$):

- $\sum_{i=0}^{0} 2^{-i} = 2^0 = 1 \leq 2$

Inductive Step:

- Suppose $\sum_{i=0}^{k} 2^{-i} \leq 2$
- We have $\sum_{i=0}^{k+1} 2^{-i} = \sum_{i=0}^{k} 2^{-i} + 2^{-k-1} \leq 2 + 2^{-k-1}$
- Well drat...

You Can't Handle the Pattern!

Look at small examples:

- $n = 0$: $2^0 = 1$
- $n = 1$: $2^0 + 2^{-1} = \frac{3}{2}$
- $n = 2$: $2^0 + 2^{-1} + 2^{-2} = \frac{7}{4}$
- $n = 3$: $2^0 + 2^{-1} + 2^{-2} + 2^{-3} = \frac{15}{8}$

Huh, seems to always work out to $2 - 2^{-n}$...

A New Theorem

Stronger Theorem: $\forall n \in \mathbb{N}, \sum_{i=0}^{n} 2^{-i} = 2 - 2^{-n}$.

Base Case($n = 0$):

- $\sum_{i=0}^{0} 2^{-i} = 2^0 = 1 = 2 - 1$

Inductive Step

- Suppose $\sum_{i=0}^{k} 2^{-i} = 2 - 2^{-k}$
- $\sum_{i=0}^{k+1} 2^{-i} = \sum_{i=0}^{k} 2^{-i} + 2^{-k-1} = 2 - 2^{-k} + 2^{-k-1}$
- $2^{-k} - 2^{-k-1} = 2^{-k-1}$

Break Time!

Take a 4 minute breather! Talk with neighbors :)

Today's Discussion Question:
If you could eliminate one food so that no one would eat it ever again, what would you pick to destroy?

Other Fixes

Theorem: All $n \in \mathbb{N}$ st $n \geq 2$ have a prime factor.[4]

Base Case($n = 2$):

- 2 is prime, and a factor of itself

Inductive Step:

- Suppose that k has a prime factor
- What does this tell us about $k + 1$?
- ...

Not enough information from k alone :(

But wait! Already proved everything k and smaller!

[4] Recall that this was an unproved lemma from last lecture.

Strong Induction

Strong Inductive Principle: To prove $\forall n \in \mathbb{N}\ P(n)$, suffices to prove

(1) $P(0)$

(2) $\forall k \in \mathbb{N}\ [(P(0) \wedge \ldots \wedge P(k)) \implies P(k+1)]$

Why does this work?

Certainly $P(0)$ is true.
If $P(0)$ is true, then $P(1)$ is.
If $P(0)$ and $P(1)$ are true, then $P(2)$ is.
...

Same domino idea as regular induction — but now new domino pushed over by *all* previous ones

Strong Induction Example

Theorem: All $n \in \mathbb{N}$ st $n \geq 2$ have a prime factor.

Base Case($n = 2$):

- 2 is prime, and a factor of itself

Inductive Step:

- Suppose true for all n st $2 \leq n \leq k$
- If $k + 1$ is prime, done
- Else, $k + 1$ has a non-trivial factor a
- $2 \leq a \leq k$, so a has a prime factor p
- Then p is a prime factor of $k + 1$

Questionable Naming Conventions

Claim: Regular induction and strong induction can prove exactly the same statements.

Why does regular proof imply strong proof?

- Only need to know $P(k)$
- Just ignore $P(0)$ through $P(k-1)$!

Why does strong proof imply regular proof?

- Consider $Q(n) := (\forall k \leq n)\ P(n)$
- Prove $P(n)$ by strengthening to $Q(n)$!

Strong induction still useful–makes proofs easier!

Induction and Recursion

Recall recursion: function that calls itself

How to prove that a recursive algorithm works?
Use induction![5] Assume that subcalls just work.

Example: binary search

- Input: sorted list ℓ, target element e
- If len(ℓ) is 1, return true iff single element is e
- If center larger than e, recurse on left half
- If center smaller than e, recurse on right half
- If center is e, return true

[5] For most algorithms, you will need to use strong induction

Binary Search Is Actually Legit

Theorem: For all non-zero $n \in \mathbb{N}$, binary search always returns the correct answer if len(ℓ) is n.

Base Case($n = 1$):

- True iff only element is e

Inductive Step:

- Suppose that BS works for lists k and smaller
- Let ℓ be a list of size $k + 1$
- If $e \notin \ell$, e won't be in half we recurse on
  - BS works on smaller lists, will return false
- If $e \in \ell$, find it or in half-list recursed on
  - BS works on smaller lists, will return true

A Proof! My Country For a Proof!

Claim: All horses are the same color.
Formally, will "prove" $P(n) :=$ "any n horses all are the same color"

Base Case($n = 1$):

- Only 1 horse, certainly the same color as itself

Inductive Step

- Suppose $P(k)$ holds.
- Consider $k + 1$ horses $h_1, h_2, \ldots, h_{k+1}$
- $P(k)$: $h_1, \ldots, h_k$ all same; $h_2, \ldots, h_{k+1}$ all same
- Sets overlap, so all $k + 1$ horses same!

Issue: sets don't overlap when $k = 1$!

Fin

Next time: graph theory!