



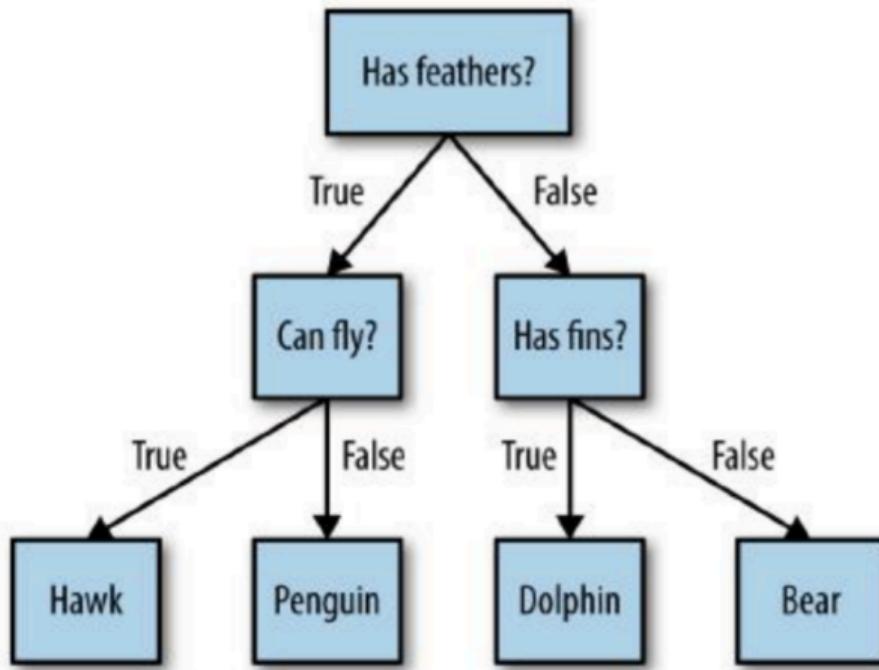








## بازی ۲۰ سوالی

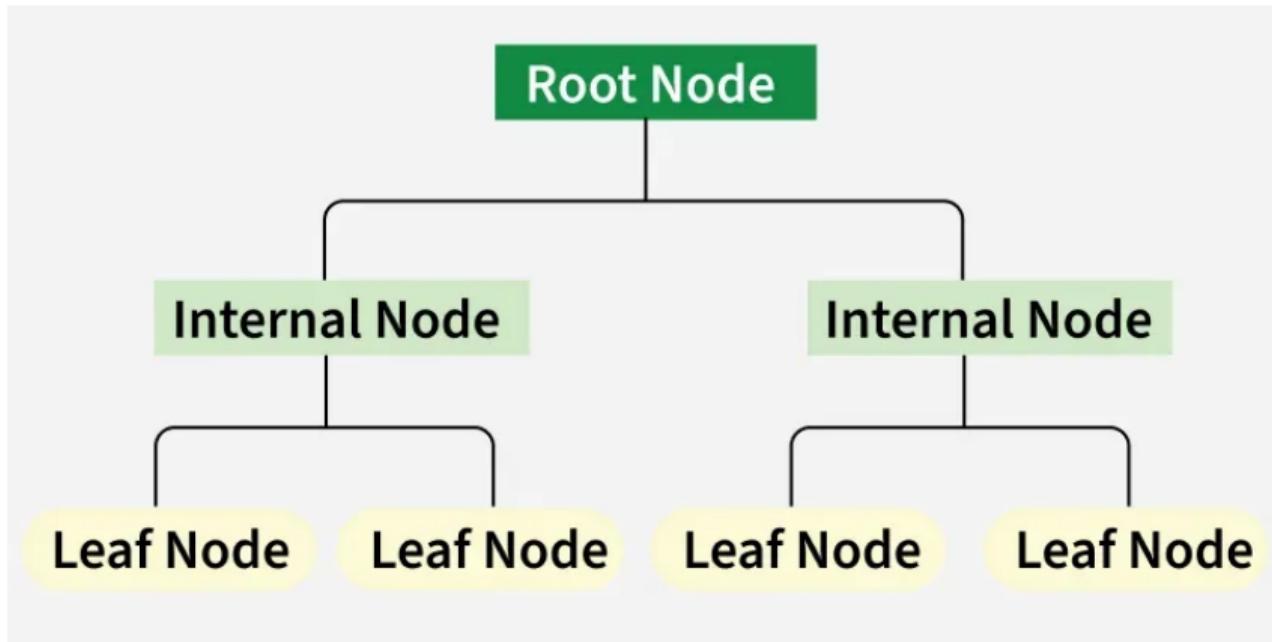




## اجزای اصلی درخت تصمیم

- ◀ **گره ریشه (Root Node)**: نقطه‌ی شروع درخت که کل دیتاست را نشان می‌دهد.
- ◀ **شاخه‌ها (Branches)**: خطوطی که گره‌ها را به هم وصل می‌کنند و مسیر حرکت از یک تصمیم به تصمیم دیگر را نشان می‌دهند.
- ◀ **گره‌های داخلی (Internal Nodes)**: نقاطی که در آن‌ها بر اساس ویژگی‌های داده، تصمیم‌گیری انجام می‌شود.
- ◀ **گره‌های برگ (Leaf Nodes)**: نقاط پایانی درخت که در آن‌ها تصمیم نهایی یا پیش‌بینی انجام می‌شود.

# اجزای اصلی درخت تصمیم



## ۱ مقدمه و تعریف

## ۲ درخت تصمیم چگونه کار می کند؟

## Overfitting ۳

## انتخاب ویژگی ها

انتخاب ویژگی ها قلب هر درخت تصمیم است؛ چون هر ویژگی نقش یک سؤال کلیدی را بازی می کند که مسیر تصمیم گیری را تعیین می کند. تصور کن در حال حل یک معما هستی و باید در هر مرحله بهترین سؤال را بپرسی تا سریع تر به جواب برسی. درخت تصمیم دقیقاً همین کار را می کند:

- ◀ در هر گره، داده ها بررسی می شوند تا بهترین ویژگی برای تقسیم انتخاب شود.
- ◀ ویژگی انتخاب شده باید بتواند داده ها را تا جای ممکن خالص تر و متمایز تر کند.
- ◀ برای انتخاب ویژگی از معیارهای مختلف استفاده می شود

## انتخاب ویرگی جدول علاقه

باید بگیریم که چگونه از داده‌های خام، درخت طبقه‌بندی بسازیم. این داده‌ها به ما می‌گویند که آیا کسی پاپ کورن دوست دارد یا نه، آیا نوشابه دوست دارد یا نه، سنش چیست و آیا عاشق فیلم پرفروش «خنک مثل یخ» هست یا نه. بنابراین ما از این داده‌ها برای ساخت این درخت طبقه‌بندی استفاده خواهیم کرد که پیش‌بینی

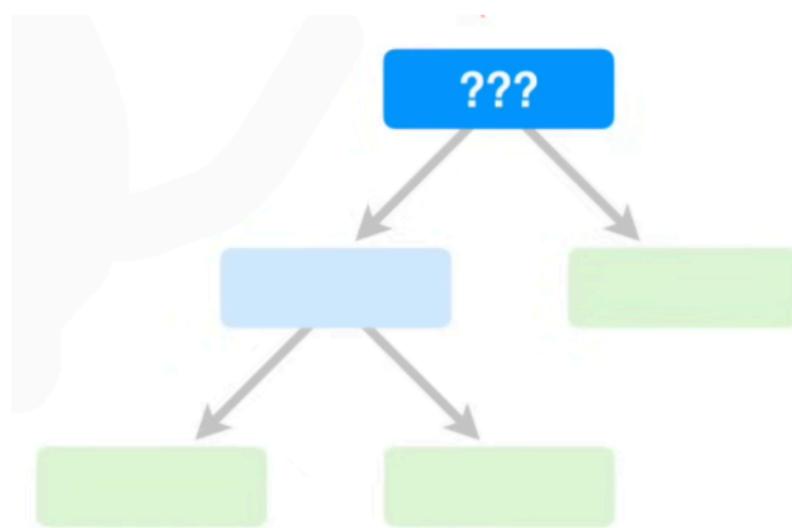
می‌کند آیا کسی فیلم «خنک مثل عنوان یخ» را دوست دارد یا نه.

## جدول علاقه

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No

## کدام ویژگی را در ریشه قرار دهم؟

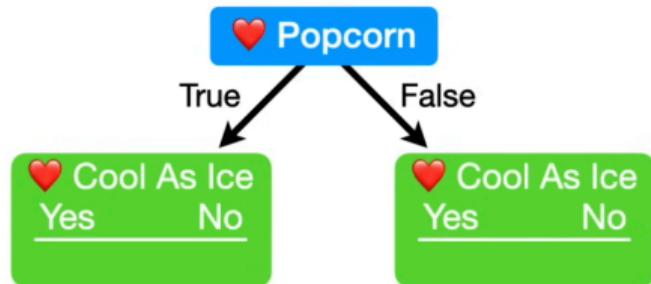
اولین کاری که انجام می دهیم این است که تصمیم بگیریم سؤال اصلی در بالای درخت چه باشد: آیا «دوست داشتن پاپ‌کورن»، «دوست داشتن نوشابه» یا «سن» باید معیار انتخاب باشد؟



## بررسی یک ویژگی

برای گرفتن این تصمیم، ابتدا بررسی می کنیم که علاقه مند بودن به پاپ‌کورن چقدر می‌توانه پیش‌بینی کنه که آیا یک نفر فیلم خنک مثل یخ رو دوست داره یا نه.

برای این کار، یک درخت تصمیم خیلی ساده درست می‌کنیم که فقط یک سؤال می‌پرسه: «آیا این فرد پاپ‌کورن دوست داره؟»

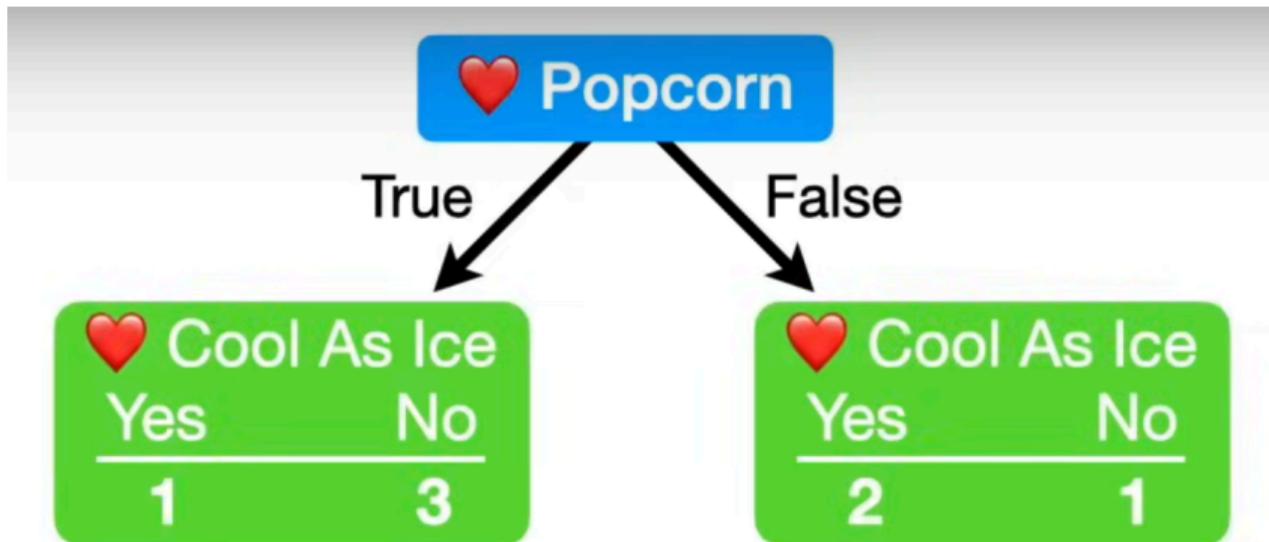


## ساختار دهی درخت با انتخاب

برای مثال:

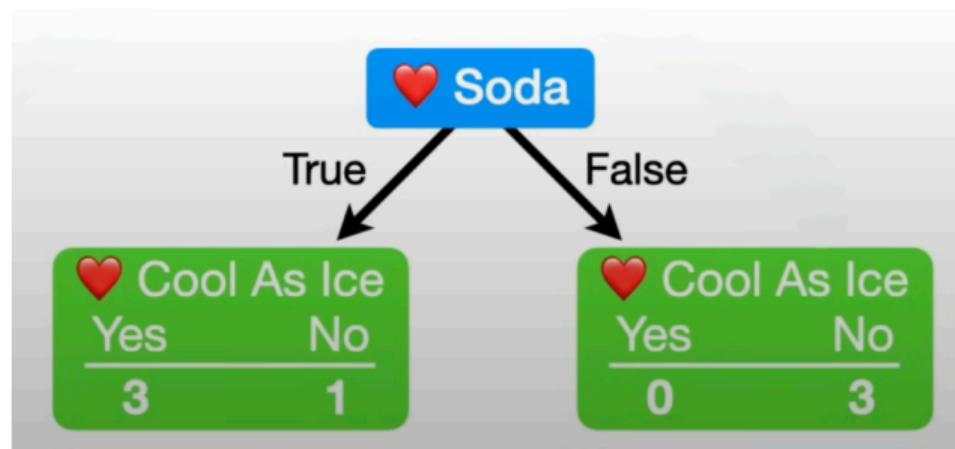
- ◀ نفر اول در مجموعه داده‌ها عاشق پاپ‌کورن است، پس به برگ سمت چپ درخت می‌رود. چون این فرد فیلم را دوست ندارد، یک عدد ۱ زیر کلمه «نه» ثبت می‌کنیم.
- ◀ نفر دوم هم پاپ‌کورن دوست دارد، پس او هم به برگ سمت چپ می‌رود. چون او هم فیلم را دوست ندارد، عدد «نه» را به ۲ افزایش می‌دهیم
- ◀ نفر سوم پاپ‌کورن دوست ندارد، پس به برگ سمت راست می‌رود. چون این فرد فیلم را دوست دارد، یک عدد ۱ زیر کلمه «بله» ثبت می‌کنیم.
- ◀ به همین ترتیب، بقیه ردیف‌ها را نیز از درخت عبور می‌دهیم و بررسی می‌کنیم که آیا هر فرد فیلم را دوست دارد یا نه، و نتایج را ثبت می‌کنیم.

## ساختار دهی درخت با انتخاب



## انتخاب ویژگی دوم

حالا بباید همین کار را برای «علاقه مند بودن به نوشابه» انجام بدھیم.

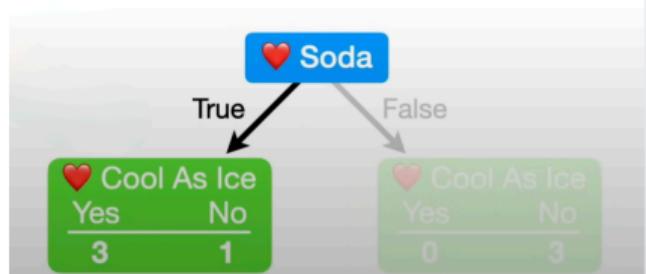
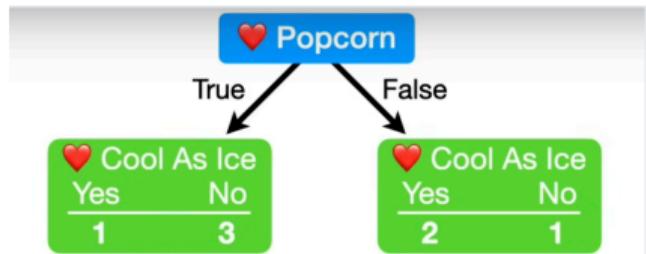


## مقایسه ویژگی ها

با نگاه کردن به دو درخت کوچک، می بینیم که هیچ کدام به طور کامل نمی توانند پیش بینی کنند چه کسی فیلم را دوست دارد و چه کسی نه.

به طور خاص، سه برگ از این درخت ها شامل ترکیبی از افرادی هستند که بعضی ها این فیلم را دوست دارند و بعضی ها نه؛ یعنی این برگ ها کاملاً خالص نیستند و درونشان افراد با علاقه های متفاوت وجود دارد.

## مقایسه ویژگی ها







## محاسبه ناخالصی جینی

برای محاسبهٔ ناخالصی جینی مربوط به ویژگی «علاقه‌مندی به پاپ‌کورن»، ابتدا باید ناخالصی جینی را برای هر برگ از درخت تصمیم به صورت جداگانه محاسبه کنیم.

ناخالصی جینی برای برگ سمت چپ برابر است با:

$$\text{Gini Impurity} = 1 - (\text{the probability of "Yes"})^2 - (\text{the probability of "No"})^2$$

$$\text{Gini Impurity} = 1 - \left(\frac{1}{1+3}\right)^2 - \left(\frac{3}{1+3}\right)^2 = 0.375$$

اگر محاسبهٔ ناخالصی را برای برگ سمت راست هم با فرمول بالا محاسبه کنیم به عدد 0.444 می‌رسیم.

## ناخالصی کل

برای محاسبه‌ی ناخالصی کل، بعد از اینکه ناخالصی هر برگ را به دست آوردیم، با توجه به تعداد نمونه‌های موجود در هر برگ، میانگین وزنی آن‌ها را محاسبه می‌کنیم. این کار نشان می‌دهد تقسیم‌بندی ما تا چه حد داده‌ها را بخوبی جدا کرده است.

$$\text{Total Gini Impurity} = \left( \frac{4}{4+3} \times 0.375 \right) + \left( \frac{3}{4+3} \times 0.444 \right) = 0.405$$

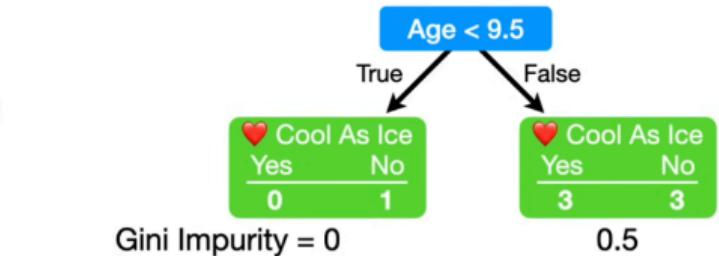
به همین ترتیب میزان ناخالصی کل برای علاقه‌مندی به نوشابه نیز برابر است با 0.214.



# محاسبه ناخالصی ویژگی سن

برای مثال محاسبه ناخالصی ویژگی سن برای مقدار اول مطابق تصویر زیر انجام می شود:

Age	Loves Cool As Ice
9.5	No
15	No
18	Yes
26.5	Yes
36.5	Yes
38	Yes
44	No
66.5	No



$$\text{Total Gini Impurity} = \left(\frac{1}{1+6}\right) 0 + \left(\frac{6}{1+6}\right) 0.5 = 0.429$$

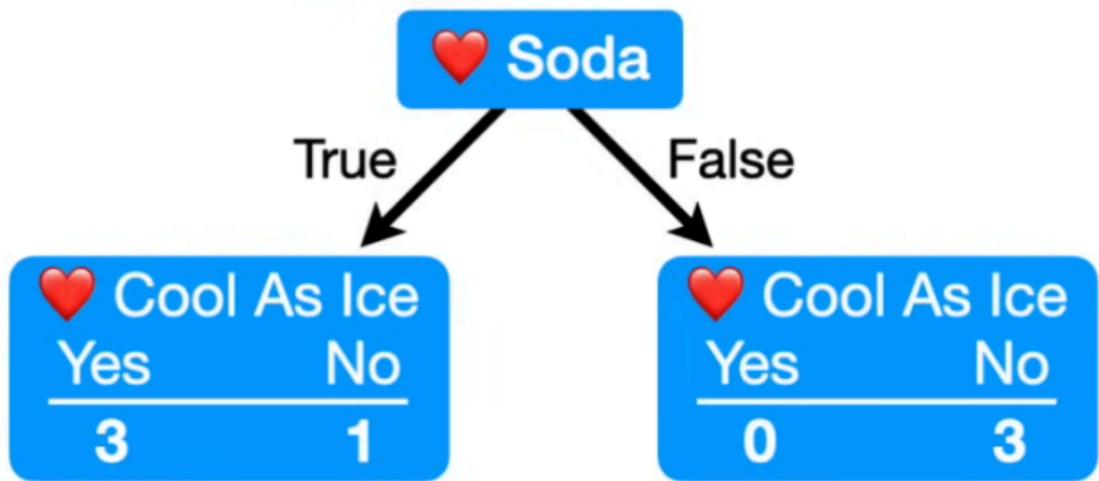
## محاسبه ناخالصی ویژگی سن

Age	Loves Cool As Ice
7	No
9.5	
12	No
15	
18	Yes
26.5	
35	Yes
36.5	
38	Yes
44	
50	No
66.5	
83	No

→ Gini Impurity = 0.429  
→ Gini Impurity = 0.343  
→ Gini Impurity = 0.476  
→ Gini Impurity = 0.476  
→ Gini Impurity = 0.343  
→ Gini Impurity = 0.429

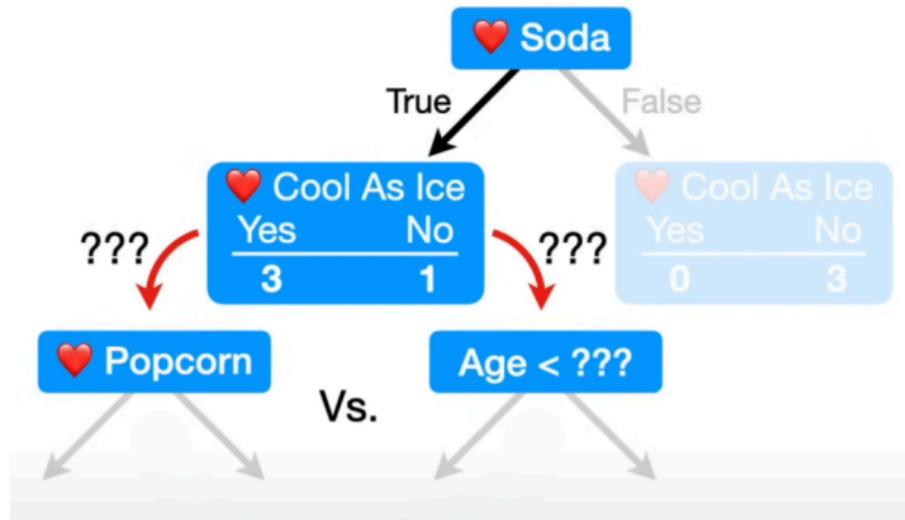


## انتخاب یک ویژگی برای ریشه



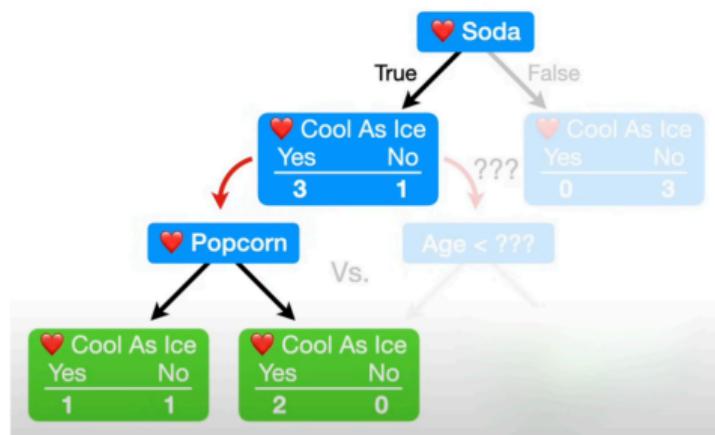
## گسترش درخت

باید بینیم آیا می توانیم مقدار ناخالصی را کاهش دهیم، با این کار که افرادی را که نوشابه دوست دارند، بر اساس علاقه مندی شان به پاپ کورن یا سن شان تقسیم کنیم.



## گسترش درخت

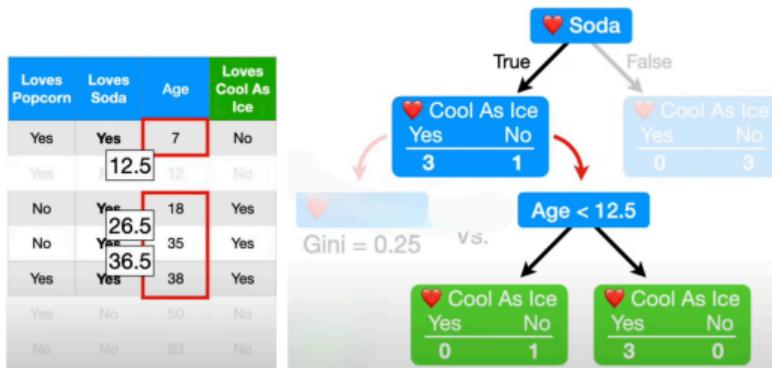
ابتدا از چهار نفری که نوشابه دوست دارند می پرسیم آیا آنها پاپکورن هم دوست دارند یا نه. از بین این چهار نفر، دو نفر پاپکورن را هم دوست دارند، بنابراین به برگ سمت چپ درخت می روند. دو نفر باقیمانده که نوشابه دوست دارند اما پاپکورن را دوست ندارند، به برگ سمت راست می روند. در نتیجه، ناخالصی جینی کل برای این تقسیم برابر با  $\frac{1}{2}$  خواهد بود.



## گسترش درخت

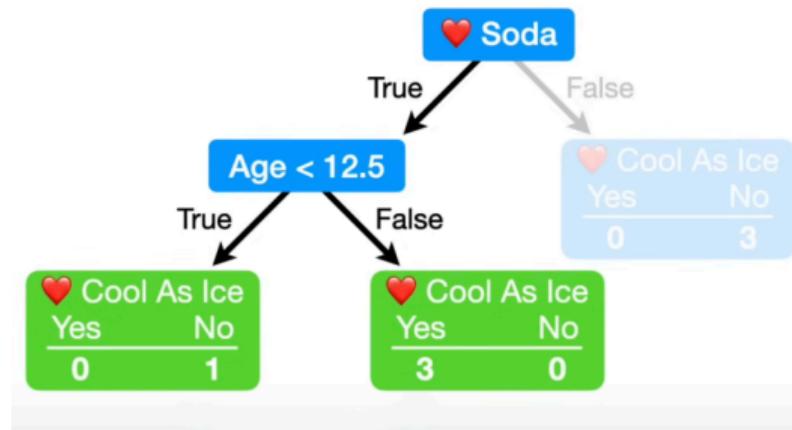
فراموش نمی کنیم که حالا مثل قبل، مقدارهای مختلفی برای سن را آزمایش می کنیم؛ با این تفاوت که این بار فقط سنِ افرادی را در نظر می گیریم که نوشابه دوست دارند.

در این بررسی، مقدار آستانه‌ی کمتر از ۱۲/۵ سال کمترین ناخالصی را دارد؛ یعنی ناخالصی جینی برابر با صفر است، چون هر دو برگ حاصل از این تقسیم کاملاً خالص هستند و هیچ ترکیبی از علاقه‌مندان و غیر علاقه‌مندان به فیلم در آن‌ها وجود ندارد. پس مقدار صفر را ثبت می کنیم.



انتخاب نود داخلی اول

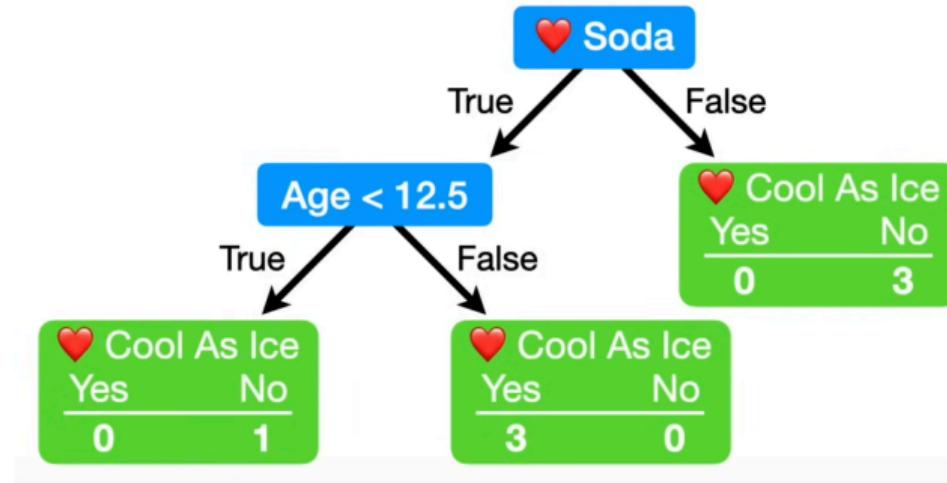
چون مقدار صفر کمتر از ۲۵٪ است، از آستانه‌ی سن کمتر از ۱۲/۵ سال برای تقسیم این گره به برگ‌ها استفاده می‌کنیم.



توجه داشته باشید که این‌ها برگ هستند، چون دلیلی برای تقسیم بیشتر این افراد به گروه‌های کوچک‌تر وجود ندارد.

## وقتی ناخالصی صفر می شود

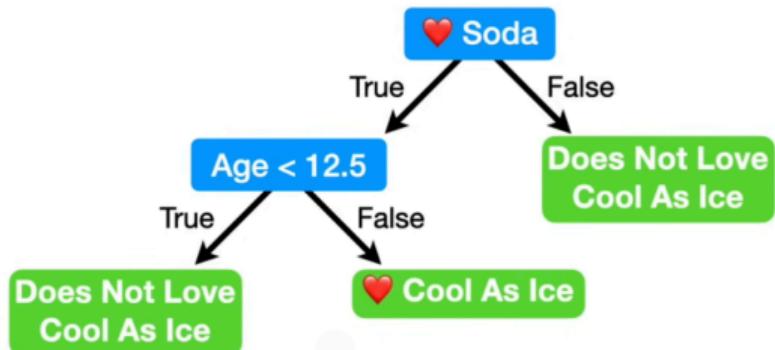
به همین صورت، این گره که شامل سه نفر است که نوشابه دوست ندارند، یک برگ محسوب می‌شود؛ چون هیچ دلیلی برای تقسیم بیشتر این افراد به گروه‌های کوچکتر وجود ندارد.



## مرحله ساخت نهایی درخت

حالا فقط یک کار آخر باقی مونده تا ساخت این درخت تصمیم کامل بشه: باید برای هر برگ، مقدار خروجی مشخص کنیم.

به طور کلی، خروجی هر برگ همون دسته‌ایه که بیشترین تعداد نمونه رو در اون برگ داره. به عبارت دیگه، چون بیشتر افراد موجود در این برگ‌ها فیلم رو دوست ندارن، خروجی این برگ‌ها می‌شه: فیلم را دوست ندارد. و چون در یک برگ دیگر، اکثریت افراد این فیلم رو دوست دارند، خروجی اون برگ می‌شه: فیلم را دوست دارد.

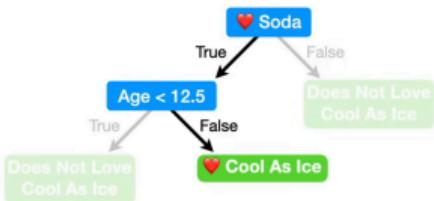


پیش بینی

حالا فرض کن یه نفر جدید وارد بشه و بخوایم پیش‌بینی کنیم که آیا فیلم «خنک مثل یخ» را دوست خواهد داشت یا نه. داده‌های مربوط به اون فرد رو وارد درخت تصمیم‌گیری‌مون می‌کنیم:

- چون نوشابه دوست داره، به شاخه‌ی چپ میره.  
و چون ۱۵ سالشه، یعنی شرط "سن کمتر از ۱۲.۵" برقرار نیست، پس به این برگ خاص می‌رسه  
در نتیجه، پیش‌بینی ما اینه که اون فرد فیلم «خنک مثل یخ» رو دوست خواهد داشت.

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	15	YES!!!



١ مقدمه و تعریف

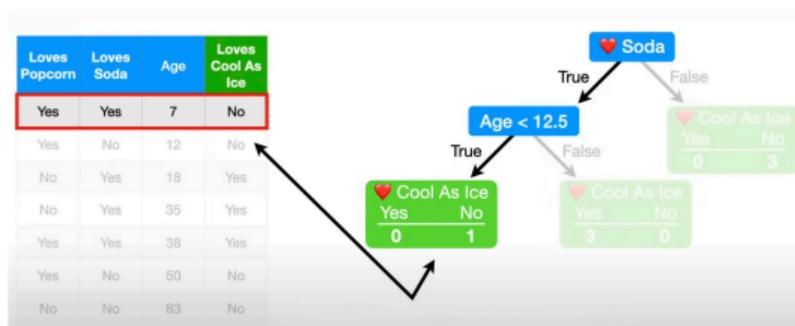
۲ درخت تصمیم چگونه کار می‌کند؟

## Overfitting ↗

## چه زمانی overfit رخ می‌دهد؟

باید یک نکته فنی را بررسی کنیم.

یادت هست وقتی این درخت تصمیم‌گیری را ساختیم، فقط یک نفر از مجموعه داده‌ی اولیه به این برگ خاص رسید. چون تعداد خیلی کمی از افراد به این برگ رسیده‌اند، نمی‌توان با اطمینان گفت که این برگ در پیش‌بینی داده‌های جدید عملکرد خوبی خواهد داشت. در واقع، ممکنه داده‌ها رو بیش از حد متناسب‌سازی کرده باشیم، یا به اصطلاح overfit کرده باشیم.



در عمل، دو روش اصلی برای مقابله با این مشکل وجود دارد

هرس کردن (Pruning) یک تکنیک مهم در درخت‌های تصمیم‌گیری است که برای جلوگیری از بیش‌برازش (Overfitting) استفاده می‌شود. بیش‌برازش زمانی رخ می‌دهد که درخت بیش از حد عمیق شود و به جای یادگیری الگوهای کلی، شروع به حفظ کردن داده‌های آموزشی کند. این موضوع باعث می‌شود عملکرد مدل روی داده‌های جدید و دیده‌نشده ضعیف شود.

با استفاده از تکنیک هرس، پیچیدگی درخت کاهش می‌یابد؛ به این صورت که شاخه‌هایی که قدرت پیش‌بینی کمی دارند حذف می‌شوند. این کار باعث می‌شود مدل بهتر بتواند داده‌های جدید را تعمیم دهد و عملکرد بهتری داشته باشد. همچنین، مدل ساده‌تر و سریع‌تر قابل پیاده‌سازی خواهد بود.

این روش زمانی بسیار مفید است که درخت تصمیم‌گیری بیش از حد عمیق شده و شروع به ثبت نویز موجود در داده‌ها بکند.

تعیین محدودیت

روش دیگر این است که برای رشد درخت محدودیت‌هایی تعیین کنیم؛ مثلاً شرط بگذاریم که هر برگ باید حداقل شامل سه نفر باشد. در این حالت، اگرچه برگ حاصل ناخالص خواهد بود، اما در عوض دید بهتری نسبت به دقت پیش‌بینی مان خواهیم داشت. چون می‌دانیم که فقط ۷۵٪ از افراد موجود در این برگ فیلم را دوست دارند.

