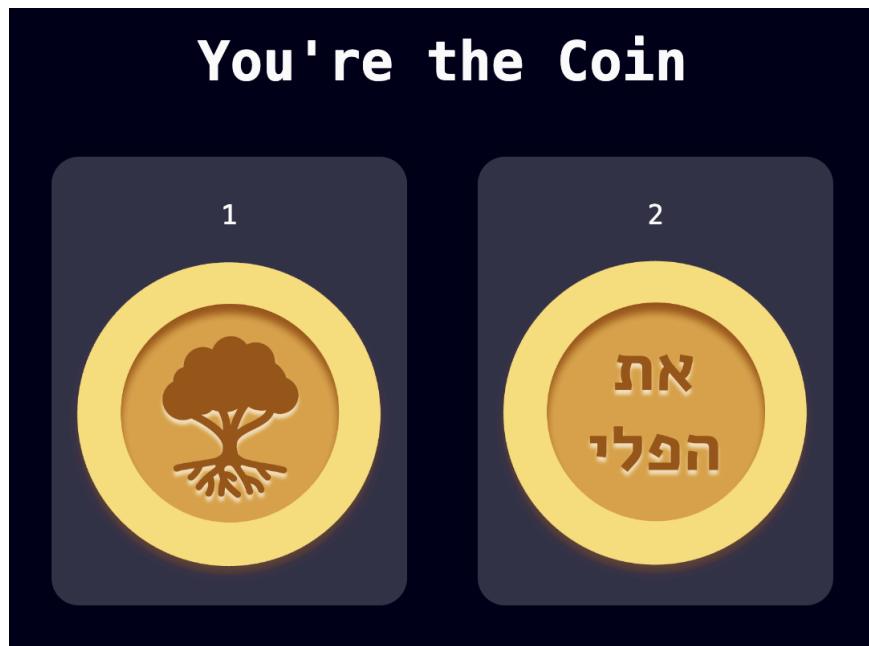




## למידת מכונה על קצה המזלג



בתרגיל זה נ洩 את המכונה מאחוריו המשחק בעל העיליה הסוחפת **You're the Coin**, בו השחקן מדמיין שהוא מטבע ובודק האם הוא מסוגל לבחור בין עץ לפלי באופן אקראי, או שמא גם בלי כוונה יש דפוסים בבחירה שלו.

לאורך התרגיל נבדוק כמה אלגוריתמים שונים לחיזוי הבחירה הבאה של השחקן בהינתן היסטוריית הבחירה שלו, ונציג את מידת ההצלחה של כל אחד מהאלגוריתמים.

### 00 הכנות

1. הורידו למחשב את [קובץ ה- html זהה](#)
2. פתחו אותו בסביבת עבודה כלשהי וחפשו את הקטע שמתחל ב - `YOUR CODE GOES HERE`
- 3.פתחו את הקובץ גם בדף

## 10 שלום עולם

בתרגיל תחכשו שני כובעים:

- כובע השחקן, בו תבחרו כל פעם עץ או פלי (ע"י לחיצה על המטבעות או לחיצה על "1" ו- "2") במקלדת)
- כובע מממשי המשחק, בו תכתבו אלגוריתמים לחיזוי הבחירה של השחקן

בקובץ שקיבלתם יש כבר "אלגוריתם חיזוי" אחד מוכמן - משתמש ב- random כדי לחזות את פועלות המשחק, וכך הוא לא מצליח במיוחד. בכל זאת תוכלו להיעזר בו כדי להבין מה קורהפה.

**בדף**, התחילה ללחוץ באקראי על "1" ו- "2" במקלדת (או על המטבעות), וטרואו גרפ' שמתארך ככל שאתה משחקים יותר סיבובים. גרפ' זה מתאר את ניקוד האלקראוי לחיזוי. האלגוריתם מקבל נקודה בכל פעם שהוא חוזה נכון בחירת המשחק, ומ Abed נקודה בכל פעם שהוא חוזה לא נכון.

**בקוד**, באזור בו עליים לכתוב את הקוד שלכם, תראו את הפונקציה **predictRandom**. פונקציה זו היא המימוש של אלגוריתם החיזוי. כל פונקציה שתכתבו באזור זה, ואשר תחיל במליה predict - תיכנס למשחק ותוכלו לראות את הניקוד שלו. הפונקציות בהן תמשכו את האלגוריתמים מקבלות פרמטר אחד - מערך היסטורייה הבחירה של המשחק, כאשר כל בחירה היא או **"H"** המציין עצ (Head) או **"T"** המציין פלי (Tail). עליה להחזיר מחרוזת פשוטה המתארת את החיזוי - כאמור, **"H"** עבור עץ, **"T"** עבור פלי.

הסיפו שתי פונקציות חיזוי פשוטות:

1. פונקציית **predictHead**, אשר תמיד חוזה שהשחקן יבחר עץ.
2. פונקציית **predictTail**, אשר תמיד חוזה שהשחקן יבחר פלי.

שחקו שוב-cut, וידאו שהפונקציות שהוספتم מתנהגות צפוי.

## 20 נקודות למחשבה

לפני שנעבור לIMPLEMENTATION יותר מתחכמים, נעצור לחשב מה המטרה שלנו.

בעזרת הפונקציות שהוספנו בסעיף 10, קל לראות שכאשר השחקן בוחר תמיד אותו ערך, למשל עץ, פונקציית החיזוי **`predictHead`** טוביה באופן שימושי מהחיזוי האקראי. אולם אם תנסו באמצעות לבבל את המחשב ולבחר באופן אקראי, נראה שהפונקציות שחוות ערכיהם קבועים לא יהיו טובות בהרבה מ - `random` (אלא אם יש לכם נטייה מובהקת להעדיף אחד מה�בעות על השני באופן עקב).

אנחנו רוצים לכתוב פונקציות חיזוי יותר חזקות, כך שגם ננסה לבבל את המחשב, הוא יעלה על הדפוסים שלנו ויצליח לקבל ניקוד גבוה ממשמעותית מ - `random` (אלא אם נצליח להיות אקרים באמצעות).

נקודה נוספת למחשבה: אם נכתוב אלגוריתם שנוטה לקבל ציוניים דומים ל - `random`, זה אומר שהוא לא מצליח לבבאותנו. אם נכתוב אלגוריתם שמקבל ציוניים טובים יותר ממשמעותית מ - `random`, זה אומר שאנחנו יחסית צפויים והצלחנו לכתוב אלגוריתם שחוזה את הבחירה שלנו.

מה המשמעות של אלגוריתם שמקבל ציון **גבוה** ממשמעותית מ - `random`, באופן עקב?

## 30 חימום

נניח בכל זאת שיש לשחקן העדפה מובהקת לאחד המטבעות, עץ או פלי, ונרצה פונקציית חייזי שמצויה למצוא את העדפה ולגבה לפיה. ממשו פונקציה כזו: מצאו את הבחירה המועדףת לפי היסטוריות הבחירה, והחזירו אותה.

**לדוגמה:** נניח שגם היסטוריות הבחירה של השחקן:

HTTHHTTHTTHTTHTHHHTHT

השחקן בחר עץ 8 פעמים, ופלி 10 פעמים, ולכן פונקציית החיזוי תחזיר פלי. לאורך המשחק הבחירה שמתقبلת רוב הפעמים עשויה להשתנות (למשל אם השחקן יבחר עץ 3 פעמים או יותר מהמצב הנוכחי), והפונקציה שתכתבם תשקף את זה.

## 40 מיטלים שרולים

הגע הזמן לכתוב פונקציית חייזי קצת יותר מתחכמת, שבכל זאת קצר דומה לפונקציה הקודמת (ואף יכולה להיעזר בה). נניח שיש חוקיות כלשהי בבחירה השחקן - למשל, שהוא נוטה לבחור הרבה עץ בראצף, הרבה פלי בראצף, ומדי פעם מחליף ביניהם. או, שהוא מנסה להיות "לא עקי" וכמעט תמיד מחליף (עץ-פלி-עץ-פלி). במקרים כאלה, הסתכלות על הבחירה האחרונות יכולה לעזור לנו בחיזוי הבחירה הבאה. כיצד נעשה זאת?

נתבונן שוב בהיסטוריות הבחירה מהסעיף הקודם:

HTTHHTTHTTHTTHTHHHTHT

במקום להסתכל על הבחירה הכי נפוצה באופן כללי, נשים לב שהבחירה الأخيرة של השחקן היא פלי, ולכן נסתכל מה הבחירה הכי נפוצה **אחריבחירה פלי**. נסמן בחירות אלו (ודאו שאתם מבינים את הסימון):

HTHTHHTTHTTHTHTHT

במקרה זה, לאחר פלי נבחר עוד פלי 4 פעמים, ובבחירה עץ 5 פעמים, ולכן נחזה עץ. (מה היוו חוזים אם הבחירה الأخيرة של השחקן הייתה עץ?)

משו פונקציית חייזי אשר משתמש במנגנון שתואר כאן כדי לבצע את החיזוי.

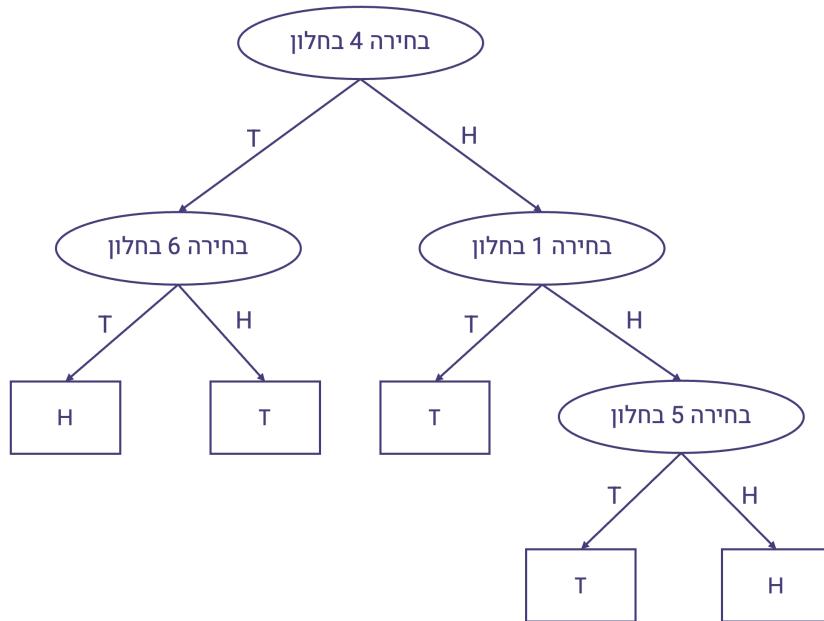
## 50 כשוגלים מתחזקים... (בונוס עסיסי)

מזל טוב, הגעתם לигה של הגודלים!

בסעיף זה מתוארת גרסה בסיסית של אלגוריתם חיזוי אמיתי מתחום למידת המוכנה - **עץ החלטה**.  
המשמעות שלו קצת יותר מורכב ודורש העמקה (ሞמלץ להתייחס אליו כמו פרויקט קטן), אך קחו נשים ו... יאללה צוללים.

בסעיף האחרון הסתכלנו על הבחירה האחרונה של השחקן כדי לנפות לחזות את הבחירה הבאה, במטרה לזהות דפוסים. אם שיחקתם בעצמכם אתם בוודאי יודעים شيئا'ם להיות דפוסים של רצפים יותר ארוכים, והיינו רוצים לנפות ללמידה גם אותם. אפשר לנפות לשימוש באותה שיטה של הסעיף הקודם - להסתכל על 6 הבחירה האחרונות ולחזות את הבחירה הבא נפוצה אחרי כל הפעמים בהיסטוריה הבחירה בהן היו אוטן 6 בחירות (לא חייב להיות 6, סתם כדוגמה לאורך חלון שאפשר להסתכל עליו). אבל זו אסטרטגיית חיזוי מאוד ספציפית - כדי שנתקבל סטטיסטיקה משמעותית על רצפים באורך  $6 + 1 = 7$  נדרש לשחק המונע, ואולי יש משהו יותר חכם שאפשר לעשות. אז מה נעשה?

**עץ החלטה** הוא מודל למידת מכונה קלאסי, בו חיזוי על רשותה מתחבע כרוצף שאלות לגבי המאפיינים של הרשותה, ככל שאלה מובילה לצומת אחר עד שנגיע לחיזוי.  
במקרה שלנו, אם מסתכלים על חלון של 6 בחירות אחרונות, עץ החלטה יכול להיראות כך:



הבהרה לגבי **חלון הבחירה האחרון**: נניח שההיסטוריה הבחירה של השחקן היא כזאת:  
**HTTHHTHHTTHTTHHHHTHT**

זהו חלון הבחירה בגודל 6 האחרון:  
**HTTHHTHHTTHTTHHHHHTHT**

אם נסתכל על כל חלונות הבחירה בגודל 6 בהיסטוריה הבחירה, יחד עם הבחירה שהגיעה אחריה:

<b>חלון בחירה</b>	<b>בחירה אחריו החלון</b>
HTTHHT	H
TTHTTH	T
THTTHT	T
HTTHHT	H
TTHTTH	T
...	...
THHHHTH	T

בחלון הראשון בדוגמה (HTTHHT), "בחירה 1 בחלון" היא H, "בחירה 3 בחלון" היא T, "בחירה 6 בחלון"  
היא גם T, וכך הלאה.

חזרה לעצמי החלטה. נחלק את המימוש לשני שלבים - יציג עץ החלטה, ובנית עץ החלטה לפי היסטוריה הבחירה של השחקן.

**השלב הראשון** הוא יותר תכני - החלטתו כיצד אתם הולכים ליצג עץ ההחלטה (בלי קשר לאלגוריתם שיבנה את עץ ההחלטה הספציפי בהתאם להיסטוריה השחקן), וממשו את הבסיס של יציג זה (אם נדרש). אפשר לנקת על כיוון OOP ולעשות מחלוקת של **TreeNode** עם מצבים נוספים שני **TreeNodes**, אפשר להשתמש במבנה נתונים פשוטים (מערכים, אובייקטים) ופונקציות מתאימות שיודעות לעבוד איתם, או כל שיטה אחרת שתבחרו.

**בשלב השני** העסוק במבנה עץ בהתאם להיסטוריה בחירות כלשהי. אנחנו רוצים לבנות עץ שמציליח כמה שיוטר מהר (תו록 כמה שפחות פיצולי D/H) להפריד בין חלונות שאחריהם נבחר עץ, לחלונות שאחריהם נבחר פלי. גם את התהילה זהה נעשו בשלבים.

בשלב הראשון, **ממשו פונקציית עדות** שאוספת, מתוך ההיסטוריה בחירות, את כל החלונות בגודל 6 (או כל גודל אחר שתבחרו), יחד עם הבירה שהגעה אחרי החלון. הפונקציה צריכה להחזיר משהו בסגנון הטבלה לעיל, בשיטת יציג לבחירתכם.

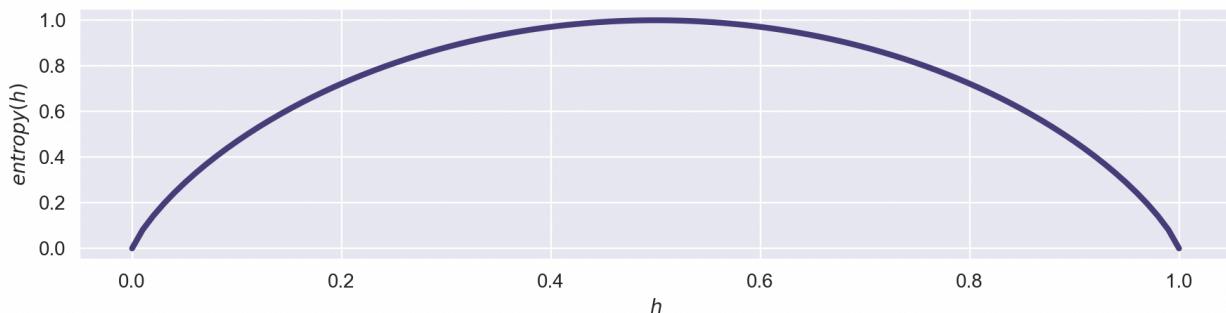
בשלב השני, **נדיר מدد** להטרוגניות (ערובוב) של אוסף בחירות מסוים. אנחנו רוצים שהמדד יהיה גבוה כאשר באוסף יש הרבה בחירות שניים הסוגים, ונמוך כאשר האוסף ייחסית אחד. המדריך שנדיר נקרא **אנטropy** (סטטיסטית, להבדיל מפיזיקלית), ומוגדר כך: נניח שבאוסף בחירות מסוים,  $t$  הוא שיעור עץ מתוך הבחירה (כלומר מספר בין 0 ל 1),  $1 - t$  הוא שיעור פלי מתוך הבחירה ( $1 = t + h$ ). האנטropy של האוסף מוגדרת כ:

$$\text{entropy}(h, t) = - (h \cdot \log_2 h + t \cdot \log_2 t)$$

מכיוון  $1 - t = t + h$  אפשר גם להגיד:

$$\text{entropy}(h) = - (h \cdot \log_2 h + (1 - h) \cdot \log_2 (1 - h))$$

הגרף הבא מראה את התנוגות ההגדירה שבחרנו, וכיitz האנטropy גבוהה כאשר יש ערובוב בין עץ לפלי באוסף הבחירה, ונמוכה כאשר האוסף ייחסית מזוקק:



ממשו פונקציה המחשבת את האנטropy של אוסף בחירות נתון.

בשלב השלישי הגיענו סוף-סוף למימוש **האלגוריתם לבניית עץ החלטה**. קודם נתאר את האלגוריתם ולאחר מכן נסביר.

נניח שהמשתנה **X** מכיל מערך עם כל הchlונות בגודל 6 בהיסטוריות הבחרויות של השחקן, והמשתנה **y** מכיל מערך עם כל הבחרויות שהגינו אחריו הchlונות. **X** - ע מתואימים, כלומר, ככלומר, האיבר ה- 11 ב- **y** מתייחס את הבחרה שהגיעה אחריו הרץ הנמצא במקום ה- 11 ב- **X**.

### **בנייה עצם החלטה ( $x, y$ )**

- אם יש ב- X פחות מ - 5 חלונות:
    - נחזיר עץ החלטה עם צומת אחד שתמיד חוזה את הבחירה הנפוצה ב - y
    - נחשב את האנטרופיה של y
    - עברור כל פיצול אפשרי idx (בחירה 1 בchl0n,בחירה 2 בchl0n וכו'):
    - נשמר ב - h\_y, H\_X את כל הchl0נות / הבחירה שבhem הchl0ן במקום idx הוא H
    - נשמר ב - t\_y, T\_X את כל הchl0נות / הבחירה שבhem הchl0ן במקום idx הוא T
    - נחשב את האנטרופיה של t\_y, h\_y (בנפרד) ונשקלל כך לאנטרופיה חדשה כללית:

$$newEntropy = \frac{y_h.length \cdot entropy(y_h) + y_t.length \cdot entropy(y_t)}{y_h.length + y_t.length}$$

  - נחשב את ההפחטה באנטרופיה לפי הפיצול הנוכחי -

$$improvement = entropy(y) - newEntropy$$

  - נשמר את כל המשתנים שמצאנו (idx, idy, y\_h, y\_t, improvement, H\_X, T\_X) איפשהו נמצא את הפיצול הכי טוב, ככלומר עם ה - improvement הכי גדול, מבין כל הפיצולים שבדקנו אם ה - improvement הכי טוב קטן מ - 0.1 :
  - נחזיר עץ החלטה עם צומת אחד שתמיד חוזה את הבחירה הנפוצה ב - y
  - אחרת, נשלוף את המשתנים idy, idt, y\_h, y\_t, H\_X, T\_X של הפיצול הכי טוב
  - ענף ימין = **בנייה עץ החלטה** (H\_X, y\_h)
  - ענף שמאל = **בנייה עץ החלטה** (T\_X, y\_t)
  - נחזיר עץ החלטה שבודק האם הבחירה - idx בchl0ן היא עצ או פלי. אם עצ, נעביר את החישוב לענף ימין; אחרת, נעביר את החישוב לענף שמאל.

**הסביר:** אלגוריתם אولي לא הכיל פשוט, אבל בכל זאת, אלגוריתם שלומד לנבأ את הבחירה שלנו לא הולך ברגל. אז מה קורה פה?

קודם כל בודאי שמתם לב שמדובר ב**אלגוריתם רקורסיבי**. בכל שלב האלגוריתם מנסה למצוא את הפיצול שמספריד הכיל טוב בין מקרים שבהם אנחנו צריכים לחזות עץ, למקרים שבהם אנחנו צריכים לחזות פלי.

#### תנאי העצירה של הרקורסיה הוא אחד משניים:

1. אין לנו מספיק חלונות / בחירות כדי ליצור עץ (הגדרתי 5 באופן יחסית שרירוטי, מוזמנים לשחק עם זה)
2. לא מצאנו פיצול שמוסיף להפרדה מספיק כדי להצדיק הוספת רמה בעץ (גםפה בחורת 1.0 באופן יחסית שרירוטי)

אם עצרנו, נჩיר **עץ החלטה עם צומת אחד** שפשטן חוצה את הערך הנפוץ מבין ע' (כמו בסעיף 30).

אם לא עצרנו, ז"א שמצאנו פיצול שאנו ממעוניינים להפריד לפיו. במקרה זה נפצל את X ואת Y לשני המקרים (עץ או פלי, במיקום בחולון שנקבע לפי הפיצול הכיל טוב), וניצור עוד שני עצי החלטה לכל אחד מהמקרים האלה באמצעות **קריאה רקורסיבית מתאימה**. לאחר מכן נרכיב עץ אשר בודק את המיקום בחולון לפיו אנחנו מפצלים, וublisher לענף המתאים.

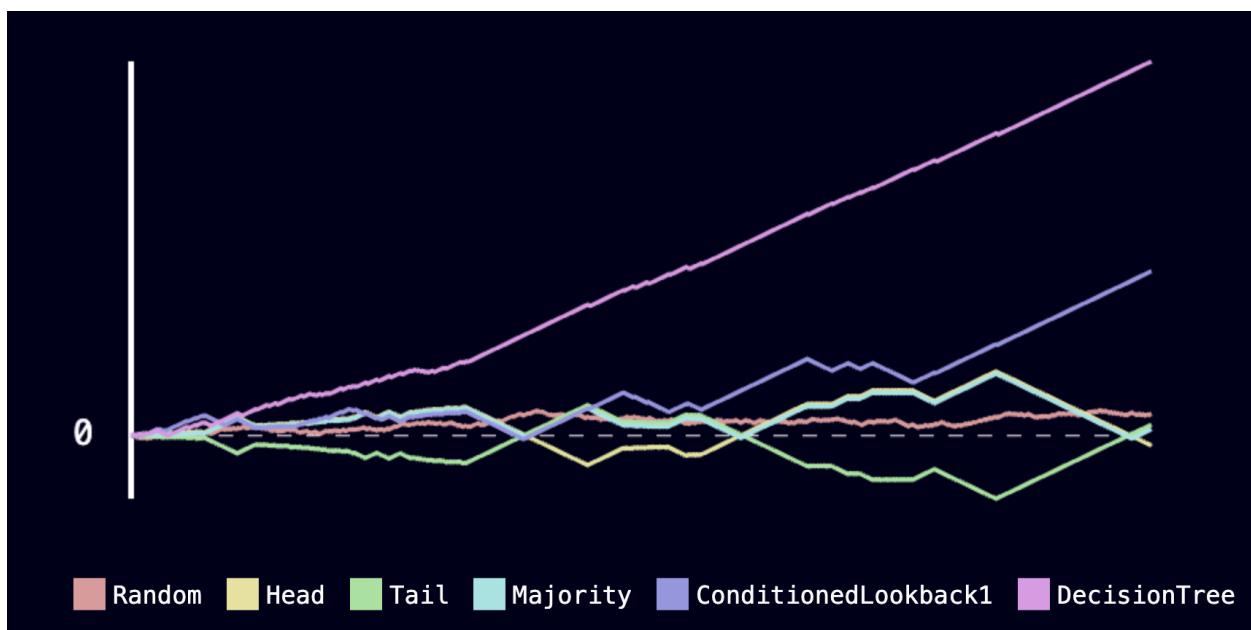
איך מודדים את השיפור בהפרדה בפיצול? נזכיר באנטropיה שהגדכנו - ככל שהוא יותר גבואה, יש יותר ערבות בין הבחירה. אנחנו יודעים את האנטרופיה של האוסף הכללי, ולכל פיצול נבדוק את **האנטרופיה הממוצעת** של כל אחד מהאוסףים אחרי הפיצול. בשביל הממוצע אנחנו משתמשים **בממוצע משקל** כדי לשקף את העובדה שהפיצול לא חייב להיות שווה בגודל בין שני החלקים (אפשר לוודא שכאשר גודל שני החלקים אחריו הפיצול זהה, מדובר בממוצע פשוט). השיפור שנוטן הפיצול הוא **ההפרש** בין האנטרופיה לפני הפיצול, לממוצע האנטרופיות אחרים.

[בקישור זה](#) תמצאו הסבר מעולה על עצי החלטה, עם אnimציות שמסבירות מה קורה בכל שלב.

## 60 מה כל זה אומר עליינו?

כל אחת ואחד מאייתנו שונים, אז אני לא יודע אילו תוצאות קיבלתם וכמה אתם צפויים.

אלה התוצאות שאני קיבלתי:



גם כניסיתי לבלב את המחשב, החיזוי שמסתכל על הבחירה הקודמת היה יחסית מוצלח, ועż ההחלטה הייתה מאד מוצלח. אם קיבلتם תוצאות דומות - משמע שאתם לא אקרים כמו שאלוי חשבתם: )

דרך נחמדה לבדוק שהאלגוריתמים שלנו לא מרים היא להזין באמת מספרים אקרים (או לפחות פאאודו-אקרים) ולווודא שאף אחד מהאלגוריתמים לא מצליח יותר טוב מחיזוי אקרי. כדי לנסות זאת, פתחו את ה- `devTools` והדביקו את הקוד הבא:

```
for (var i = 0; i < 200; i++) {
  document.querySelectorAll('.coin')[Math.round(Math.random())].click();
}
```