למידה חישובית ממידע: הרצאה 2 – Decision Trees

עצי החלטה הם תת-קבוצה של אלגוריתמים ממיינים - קלסיפיקציה (Classifiers).

נקבל דאטה אימון = המון וקטורים של תכונות שלגביהם נדע האם הם פלוס או מינוס, נכניס אותם לאלגוריתם הלמידה שיניב עבורנו היפותזה, מודל, ממיין (קלאסיפייר).

- Classification
  - Given $\{x_i, y_i\}$ where $y_i \in \{0,1\}$ as training data, determine for a new x if $x \in C_0$ or $x \in C_1$

<u>מושגים פורמליים (עבור מרחב בדיד, דיסקרטי):</u>

- X הוא מרחב הדגימות (x שייך לX).
- קונספט c concept הוא תת קבוצה של מרחב הדגימות, כלומר c שייך לקבוצת החזקה של H = X. (נקרא גם מרחב ההיפותזות)
- דאטה אימון D training set היא קבוצה של זוגות <x,c(x)> כך ש-x היא דגימה ממרחב הדגימות X ו- c(x) שייך ל- {+1/-1}.
- אנחנו מחפשים אחר היפותזה (או מודל) h ששייך ל-H (תת קבוצה של דגימות), כך ש- h(x) = c(x) עבור כל x המיוצג ב-D (או עבור רוב ה-x-ים בקבוצת האימון D)

יכולים להיות אטריביוטים/פיצ'רים נומריים = רציפים, וניתר להמיר בין משתנים רציפים לבדידים ולהפך. אבל, כדאי שנעשה זאת בזהירות רבה, מפני שאם ננקוט בשיטה בה ההבדל בין 3 ל-2 גדול מההבדל שבין 3 ל-1 כנראה שהימרנו באופן לא נכון.

- Converting from continuous to discrete can be done by quantization (binning):

- Converting from discrete to continuous can be done by encoding:
  - Blue = 1, Green = 2, Brown = 3 etc.

עצי החלטה

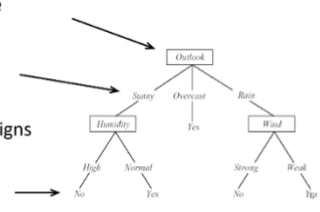
להלן דוגמה לגבי מר סמית' שמחליט האם ללכת לשחק טניס על פי מצב הרוח, הלחות, הטמפרטורה והתחזית (האטריביוטים). להלן דוגמה לדגימות של מר סמית ועץ ההחלטה שנבנה מהדאטה.

Day	Outlook	Temperature	Humidity	Wind	Play Tennis
D1	Sunny	Hot	High	Weak	**No**
D2	Sunny	Hot	High	Strong	**No**
D3	Overcast	Hot	High	Weak	**Yes**
D4	Rain	Mild	High	Weak	**Yes**
D5	Rain	Cool	Normal	Weak	**Yes**
D6	Rain	Cool	Normal	Strong	**No**
D7	Overcast	Cool	Normal	Strong	**Yes**
D8	Sunny	Mild	High	Weak	**No**
D9	Sunny	Cool	Normal	Weak	**Yes**
D10	Rain	Mild	Normal	Weak	**Yes**
D11	Sunny	Mild	Normal	Strong	**Yes**
D12	Overcast	Mild	High	Strong	**Yes**
D13	Overcast	Hot	Normal	Weak	**Yes**
D14	Rain	Mild	High	Strong	**No**

<u>מתי משתמשים בעצי החלטה?</u>

- כאשר הדגימות שלנו מתוארות על ידי זוגות של תכונות וערכים.
- כאשר פונקציית המטרה שלנו היא בדידה (לא בהכרח, DT regression)
- **כאשר ההיפותזה הנדרשת היא בעלת מבנה לוגי ואינטרפרטציה היא חשובה.**
- כאשר יש לנו דאטה אימון שהיא possibly noisy, even inconsistent.

- Each internal node tests an attribute
- Each branch corresponds to an attribute value
- Each leaf node assigns a classification



השאלה שלנו השיעור: איך נבנה את העץ הזה באופן הנכון/הטוב ביותר? איך יודעים איזו שאלה שואלים קודם?

יצירת העץ

- ניקח את כל הדגימות וניצור מהם את השורש של העץ
- נאתחל תור של הצמתים בעץ
- כל עוד יש צמתים לא שלמים בתור נבצע:

- ניקח את הצומת הבא n
- אם הדגימות ב-n טהורות/מונוכרומטיות (כולם בצבע כחול למשל), נסמן את הצומת כשלם ונמשיך לצומת הבא בתור.
- אחרת, נכניס ל-A את **האטריביוט (וערך החסימה של ערכיו) ה"טוב ביותר"** עבור הקבוצה ב-n.

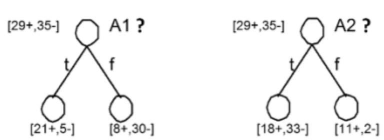
-- נשים את A כאטריוביוט ההחלטה עבור n
-- לכל אינטרוול של ערכים של A, ניצור צומת-בן לצומת n
-- נציג את הדגימות לילדים של n
-- נכניס את כל הילדים הלא ריקים לתור

Create the root node with all samples
Insert the node to initialize a queue, Q
While there are more incomplete nodes in Q <u>do</u>:
- Get next ***node n***
- <u>If</u> training examples in ***n*** are perfectly classified
  Then mark node as complete and continue to next node in Q
- <u>*Else*</u> ***A*** ← the "<u>best</u>" **decision attribute** (and **boundary values**) for the set in ***n***
  - Assign ***A*** as the decision attribute for ***n***
  - For each **interval of values** of ***A***, create a new descendant of ***n***
  - Distribute training examples to descendant nodes
  - Insert all (non empty) descendant nodes to Q

Will be disc

איך נדע מיהו אטריביוט ההחלטה הטוב ביותר?

בדוגמה משמאל יש לנו שני אטריביוטים A1, A2. איך נדע איזה עדיף? נשים לב שהילדים של האטריביוט A2 מניב ילדים כמעט-מונוכרומטים, מה שמקרב אותנו להתפלגות טהורה, מה שנותן לנו להרוויח יותר אינפורמציה. לכן, נרצה להפחית את רמת האי-טהורות (או הבילבול, או אי הוודאות) בצעד הבא שלנו.



<u>פונקציית אי וודאות</u> היא פונקציה המקבלת התפלגות דיסקרטית ומחזירה מספר ממשי, המקיימת את התנאים הבאים עבור וקטור ההסתברויות :

- פונקציה זו צריכה להיות אי שלילי (הכול ודאי = 0)
- הערך המינימלי = אי וודאות מינימלי = מקסימום וודאות, מתקיים כאשר יש מאורע שקורה תמיד, כלומר קיים i כך ש- pi = 1. העיגול הכחול המושלם
- אי וודאות מקסימלית מתקיימת כאשר ההתפלגות היא יוניפורמית = ההתפלגות של כל המאורעות היא יוניפורמית עם אותה ההסתברות. העיגול החצוי העליון
- פונקציה זו תהיה סימטרית בהתייחס לרכיבים של הוקטור P (וקטור ההסתברויות)
- היא חלקה ככל האפשר (נגזרות רציפות למען נוחות מתמטית).

for probability distributions $P = (p_1, \dots, p_k) \in [0,1]^k$:

- $\varphi(P) \geq 0$
- The minimal value is attained when $\exists i$ s.t $p_i = 1$.
- The maximal value is attained when $1 \leq \forall i \leq k$, $p_i = {}^1/_k$.
- It is symmetric with respect to the components of P
- It is smooth (infinitely differentiable) in the relevant range

<u>פונקציית Gini Impurity</u>

- מודד חוסר שוויון בכלכלה
- מניב טווח ערכים 0-1, כך שערכים נמוכים משמעותם אי וודאות פחותה (יותר וודאות), ערכים גבוהים משמעותם אי וודאות גבוהה.

$G(S) = 1 - \frac{1}{4} - \frac{1}{4} = \frac{1}{2}$

If we have k values equi-distributed then G(S)=1- k(1/k)² =1-1/k=~1

$G(S) = 1 - 1 - 0 = 0$

If we have k values but only one value really present then G(S)=1-(k/k)² = 0

$$G(S) \equiv 1 - \sum_{i=1}^{c} (p_i)^2 = 1 - \sum_{i=1}^{c} \left(\frac{|S_i|}{|S|}\right)^2$$

(נזכיר שפונקציה זו מקבלת S משום שהיא מקבלת תת קבוצה של דגימות שנמצאים באותו הצומת)

להלן דוגמה לחישוב אי וודאות על ידי פונקציית Gini : כאשר יש לנו שני צבעים c=2

<u>בחזרה לנושא השיעור :</u> אנו מחפשים אחר אטריביוט A שיניב את ממוצע משוקלל הGini הטוב ביותר לאחר פיצול, ולכן נגדיר את פונקציית הרווח, שמחשבת לנו את **הוודאות שהרווחנו מהפיצול על ידי אטריביוט ספציפי** A, על תת הקבוצה של הדגימות S של הצומת הנוכחי :

$$GiniGain(S,A) = \Delta G(S,A) \equiv G(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} G(S_v)$$

Change in Impurity

Impurity Before Split

Weighted Average of Impurity of All Groups After Splitting

נוסחה זו מחשבת כמה אי וודאות הייתה לי לפני הפיצול ומחזירה כמה אי וודאות משוקללת הרווחתי אחרי הפיצול על ידי האטריביוט. **אנחנו נרצה למקסם את פונקציית ה-Gain. (Sv כל מי שבתוך S ועונה לערך של v)**

<u>פונקציית האנטרופיה Entropy</u>

נשתמש בנוסחה של אנטרופי כפונקציית אי וודאות עבור משתנה רנדומי X שלוקח n ערכים שונים עם הסתברויות pi :

$$H(X) = -\sum_{i=1}^{n} p_i \log(p_i)$$

(לוג בסיס 2, כאשר ידוע לנו כי לוג על ערכים בין 0-1 הוא שלילי ולכן המינוס)

- מודד את המידע הממוצע שמתקשר לתוצאה של משתנה מקרי.
- יהיו קבוצה S ודאטה עם c (יתכן יותר מ-2) קלאסים. Pi היא החלק היחסי של קלאס i בקבוצה S. האנטרופיה של S :

$$Entropy(S) \equiv \sum_{i=1}^{c} -p_i \log_2 p_i = \sum_{i=1}^{c} -\frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|}$$

נמיר את האנטרופיה לנוסחת רווח Gain(S,A) = פונקציית הרווח הצפוי של האנטרופיה לאחר פיצול על פי אטריביוט A.

$$\Delta\phi(S,A) \equiv \phi(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} \phi(S_v)$$

$$\text{InfoGain}(S,A) = H(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} H(S_v)$$

<u>תכונות נוספות של אנטרופיה:</u>

- הערך המקסימלי שלה (בה יש מקסימום אי וודאות), הינה כאשר ההתפלגות יוניפורמית/אחידה כיאה לפונקציית אי וודאות :

$$H\left(P = \left(\frac{1}{k}, \dots, \frac{1}{k}\right)\right) = \log k$$

- **<u>נרצה להוכיח איפה אנטרופיה מקסימלית:</u>**

<u>נשתמש בלמה 1 : פונקציית לוג היא פונקציה "עצובה" קעורה כלפי מטה (נגזרת שניה שלו היא שלילית).</u>

<u>ובלמה 2 : אי שוויון ינסן (מופיע משמאל >>) אשר הוא נכון לכל פונקציה עצובה.</u>

Lemma 2 (Jensen's Inequlity):

The following holds for any sad function f:

$\forall x_1, \dots x_k$ and $\forall \lambda_1 \dots \lambda_k \in [0,1]$ s.t $\sum \lambda_i = 1$.

$$f\left(\sum_{i=1}^{k} \lambda_i x_i\right) \geq \sum_{i=1}^{k} \lambda_i f(x_i) .$$

$$
\begin{aligned}
H(p_1,\dots p_k) &= -\sum_{i=1}^{k} p_i \log(p_i)\\
&= \sum_{i=1}^{k} p_i \log\left(\frac{1}{p_i}\right)\\
&\le \log\left(\sum_{i=1}^{k} p_i \cdot \frac{1}{p_i}\right)\\
&= \log k
\end{aligned}
$$

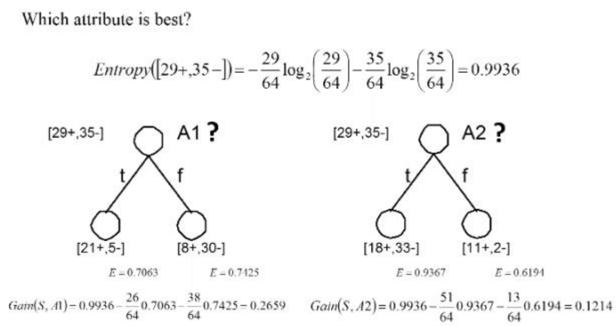
$$\boxed{f\left(\sum_{i=1}^{k} \lambda_i x_i\right) \ge \sum_{i=1}^{k} \lambda_i f(x_i)}$$

<u>הצעד האחרון בהוכחה :</u>

- השתמשנו בשתי הלמות לעיל במטרה להראות היכן אנטרופיה מקסימלית.

בחזרה לדוגמה שלנו: נחשב את הרווח האנטרופי עבור שני האטריביוטים A1, A2

כך נדע איזה מבין האטריביוטים עדיף, נשים לב שהרווח גבוהה יותר עבור A1 כלומר הוא מניב עבורנו אי וודאות פחות מאטריביוט A2.



נחזור חזרה לדוגמה ההתחלתית עם מר סמית׳: יש לנו 14 אינסטנסים (בעולם האמתי יהיו הרבה יותר), ונרצה להחליט על מה שואלים: תחזית, טמפרטורה, רוח וכו׳. נחשב עבור הדגימות שלנו ועבור האטריביוטים את הרווח האנטרופי במטרה להחליט על השאלה שתישאל ראשונה. התקבל **שהתחזית (Outlook) הוא האטריביוט שמניב את הרווח הגבוהה ביותר** ולכן הוא ישאל ראשון.

The information gain values for the 4 attributes are:
- Gain(S,Outlook) =0.247
- Gain(S,Humidity) =0.151
- Gain(S,Wind) =0.048
- Gain(S,Temperature) =0.029

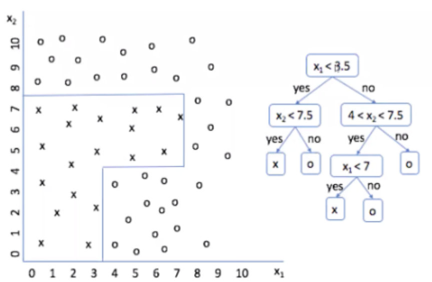
where S denotes the collection of all training examples

<u>נוכל לעבוד גם עם דאטה שאיננו קטגורי:</u> להלן דוגמה (DTs in continuous space)

DTs in continuous space

עובדות נוספות

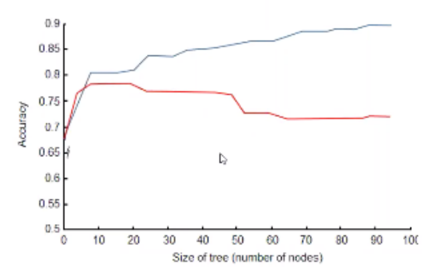
- <u>גרידי אינו אופטימלי</u> – הגישה שהצגנו היא גישה חמדנית, אנו מחפשים בכל שלב אחר הרווח המקסימלי עבור אותו שלב בעץ.
- <u>הרווח תמיד אי שלילי</u> – עבור כל פיצול שנעשה, נרוויח וודאות.



תמיד נרצה להימנע מ-Overfitting! לכן לא נרצה להרחיב את העץ יותר מידי כדי שלא יתאים לדאטה הנתון יותר מידי ויוכל להגיע להחלטה עם מצב חדש. כדי לעשות זאת נשתמש בדאטה סט נוסף שיקרא "validation" set. נלמד את הדאטה על ידי הדאטה אימון אבל נמדוד את הטעות על ה- validation set.

(להלן גרף המתאר את הדיוק, לכאורה מדויק יותר ככל שיש לנו יותר אטריביוטים)

מכאן שנרצה להפסיק להרחיב את העץ לפני שנגיע למונוכרומטיות מושלמת (לפני שגיאה 0), או להרחיב את העץ לשגיאה 0 ואז לבצע קיצוץ post-prune, או לשלב בין שתי השיטות.



גישת Chi Square ששואלת – האם פיצול לפי אטריביוט מועמד נותן לנו התפלגות לפי קלאס שיש לה יותר כוח מהנוכחית? האם הבנים מאוד דומים לאבא? הם הרווח הוא סטטיסטית משמעותי? (הרחבה בתרגול 2)

<u>בעיות נוספות</u>

- התמודדות עם **ערכים חסרים** של חלק מהאטריביוטים: השלמת דאטה או לקיחת הממוצע
- חיפוש אחר ערך חותך, נקודת הסף, **באטריביוטים רציפים**
- פיצול אינפרמציה ו- **Gain Ratio** עבור אטריביוטים עם ערכים רבים - מבחינת החישוב יש יתרון עבור אטריביוטים שיש להם מספר ערכים גבוהה יותר ולכן נגדיר Gain Ratio (הרחבה בתירגול)
- הכללת העלות עבור שמירת האטריביוטים, ופונקציית שגיאה ממושקלת
- גבולות מורכבים