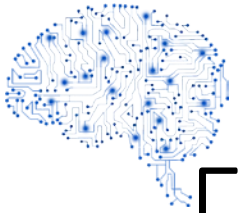
Дерева прийняття рішень

Професор, д.е.н. Ставицький А.В.



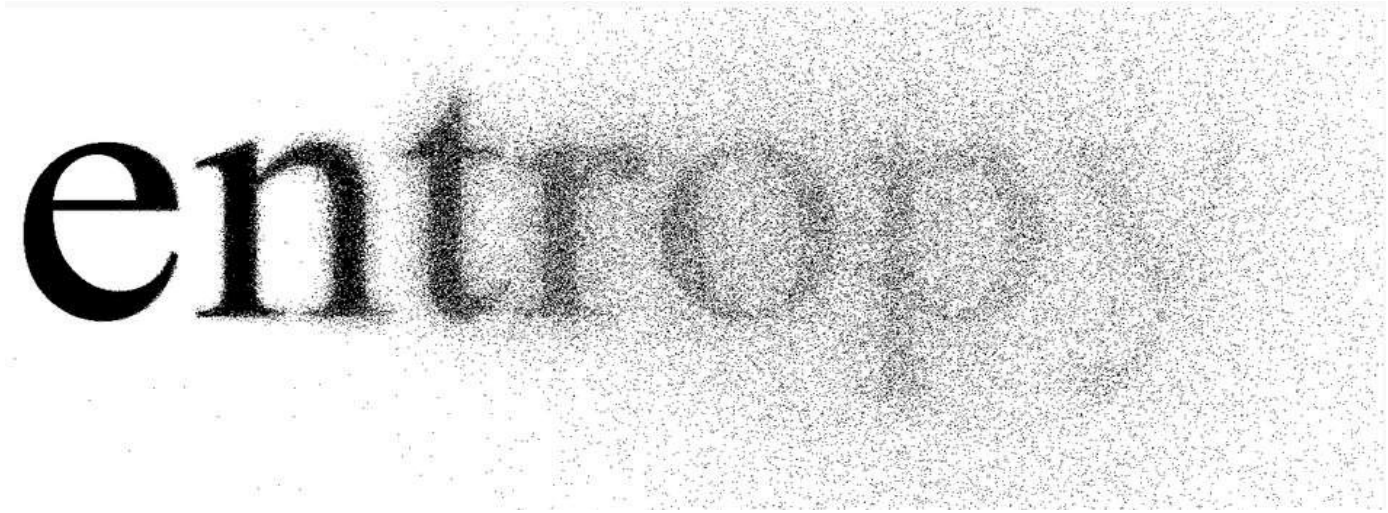
Гра «двадцять питань»

- Я задумаю визначну особистість, а ви повинні за 20 питань вгадати, кого я загадав

- Важлива здатність питання відсіяти якомога більше невірних відповідей
- Це інтуїтивно відповідає поняттю приросту інформації, яке базується на ентропії

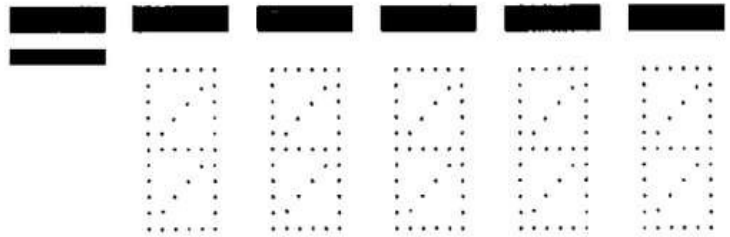
Ентропія

- Ентропія — це те, як багато інформації вам не відомо про систему



Приклад з поштовим індексом

- Наприклад, якщо ви запитаєте мене, де я живу, і я відповім: в Україні, то моя ентропія для вас буде високою, все-таки, Україна — велика за територією країна
- Якщо ж я назву вам свій поштовий індекс, то моя ентропія для вас знизиться, оскільки ви отримаєте більше інформації
- Поштовий індекс містить 5 цифр, тобто я дав вам 5 символів інформації. Ентропія вашого знання про мене знизилася приблизно на 5 символів. (Насправді, не зовсім, тому що деякі індекси відповідають більшій кількості адрес, а деякі — меншій, але ми цим знехтуємо)



Приклад з гральними костями

- Нехай у мене є десять гральних кісток, і викинувши їх, я вам повідомляю, що їх сума дорівнює 30. Знаючи тільки це, ви не можете сказати, які конкретно цифри на кожній з кісток - вам не вистачає інформації
- Цьому стану (=сумі, що дорівнює 30 відповідають 2 930 455 різних варіантів). Чому дорівнює ентропія цього стану?
- Будемо вимірювати ентропію як кількість символів, якими можна занумерувати всі можливі комбінації цього стану (для поштового індексу було 5 символів)
- Ентропія дорівнює приблизно 6,3 символам (0,3 з'являється через те, що при нумерації випадків по порядку в сьомому розряді вам доступні не всі цифри, а тільки 0, 1 і 2)

- А що, якби я вам сказав, що сума дорівнює 59? Для цього стану існує всього 10 можливих комбінацій, так що його ентропія дорівнює всього лише одному символу
- Нехай тепер я вам скажу, що сума перших п'яти кісток дорівнює 13, а сума інших п'яти — 17, так що загальна сума знову 30. У вас, однак, в цьому випадку є більше інформації, тому ентропія системи для вас повинна впасти. І, дійсно, 13 на п'яти кубіках можна отримати 420-ма різними способами, а 17 — 780-ма, тобто повне число комбінацій складе всього лише 420 x 780 = 327 600, а не 2 930 455. Ентропія такої системи приблизно на один символ менше, ніж в першому прикладі

Логарифм

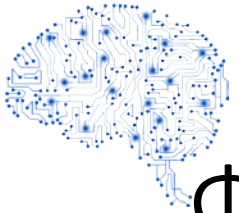
- Ми вимірюємо ентропію як кількість символів, необхідних для запису (=кодування) числа випадків. Математично ця кількість визначається як логарифм
- Дійсно, десятковий логарифм цілого числа - це кількість цифр цього числа (або, можливо, на одиницю більше):
- lg(9) =0,95... lg(10) = 1 lg(11) = 1,04...
- lg(32) = 1,50... lg(62) = 1,79... lg(90) = 1,95... lg(900) = 2,95...
- Якщо у нас алфавіт складається лише із одного символу, то невизначеності тут не має, і поява цього символу не несе ніякої інформації, ентропія повинна дорівнювати нулю, а lg(1) = 0

Приклад - 1

- Якщо можливість появи будь-якого символу алфавіту рівноймовірна, то ця ймовірність p = 1/m, де m— це кількість різних значень одного символу
- Наприклад, для грального кубика: p = 1/6
- Якщо довжина повідомлення (N) дорівнює кількості різних значень ОДНОГО символу, тобто N = m, то
  Entropy = log N = log m = log (1/ p) = - log p, (формула Хартлі), тобто кількість інформації на кожен такий сигнал дорівнює мінус логарифму імовірності окремого символу
- Отримана формула дозволяє для деяких випадків визначити кількість інформації. Однак для практичних цілей необхідно задатися одиницею її виміру. Для цього припустимо, що інформація — це усунена невизначеність

Приклад - 2

- Тоді у найпростішому випадку невизначеності вибір буде проводитися між двома взаємовиключними рівноймовірними повідомленнями, наприклад між двома ознаками: TRUE і FALSE, 0 і 1, позитивним і негативним імпульсами, імпульсом і паузою тощо.
- Кількість інформації, передана у цьому найпростішому випадку, найбільш зручно прийняти за одиницю кількості інформації. Саме таку кількість інформації можна отримати, взявши логарифм за основою 2:
  $Entropy = - \log_2 p = - \log_2 (1/2) = \log_2 (2) = 1$
- Отримана одиниця кількості інформації, що представляє собою вибір з двох рівноймовірних подій, отримала назву двійковій одиниці, або біта (bit = binary unit)
- Біт є не тільки одиницею кількості інформації, але й одиницею вимірювання ступеня невизначеності. При цьому мається на увазі невизначеність, яка міститься в одному досвіді, що має два рівноймовірних результати



Фактор несподіваності повідомлення

- На кількість інформації, що отримується одержувачем з повідомлення, впливає фактор його несподіванки, який залежить від ймовірності отримання того чи іншого повідомлення
- Чим менше ця ймовірність, тим повідомлення більш несподівано і, отже, більш інформативно. Повідомлення, ймовірність якого висока і, відповідно, низька ступінь несподіванки, несе мало інформації
- Але формула Хартлі дозволяє визначити кількість інформації в повідомленні тільки для випадку, коли поява символів рівноймовірна, і вони статистично незалежні. На практиці ці умови виконуються рідко
- При визначенні кількості інформації необхідно враховувати не тільки кількість різноманітних повідомлень, які можна отримати від джерела, але і ймовірність їх отримання

Підхід Шеннона

- Розглянемо наступну ситуацію.
- Джерело передає елементарні сигнали k різних типів. Простежимо за досить довгим відрізком повідомлення
- Нехай в ньому є N_1 сигналів першого типу, N_2 сигналів другого типу, … N_k сигналів k-го типу, причому $N_1 + N_2 + ... + N_k = N$ - загальне число сигналів в спостережуваному відрізку, f_1, f_2, …, f_k - частоти відповідних сигналів
- При зростанні довжини відрізка повідомлення кожна з частот прагне до фіксованої межі, тобто
  $\lim f_i = p_i$, (i = 1, 2, …, k), де p_i можна вважати ймовірністю сигналу

Ентропія Шеннона

- Ентропія Шеннона визначається для системи з N можливими станами наступним чином:

$$S = -\sum_{i=1}^{N} p_i \log_2 p_i,$$

- де p_i - ймовірність знаходження системи в і-му стані

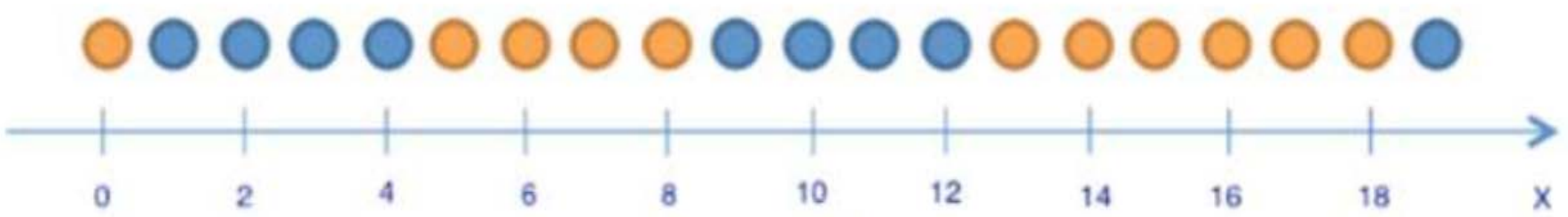
- Ентропія відповідає ступеню хаосу в системі
- Чим вище ентропія, тим менше впорядкована система і навпаки

Дерева прийняття рішень

- Дерева прийняття рішень (decission trees) є одним з найбільш популярних методів вирішення завдань класифікації та прогнозування.
- Вони дозволяють візуально і аналітично оцінити результати вибору різних рішень. Вперше були запропоновані Ховілендом і Хантом (Hoveland, Hunt) наприкінці 50-х років минулого століття
- Дерева прийняття рішень використовують, коли потрібно прийняти рішення в умовах невизначеності, коли кожне рішення залежить від результату попередніх результатів або деяких заданих умов, що з'являються з певною ймовірністю
- Дерева прийняття рішень інколи називаються деревами вирішальних правил, деревами класифікації і регресії. Якщо залежна (цільова) змінна приймає дискретні значення - це завдання класифікації, якщо залежна змінна приймає безперервні значення, то вирішується завдання чисельного прогнозування

Іграшковий приклад – 1

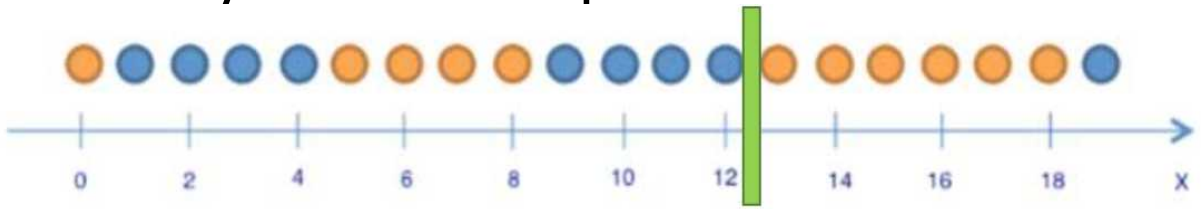
- Потрібно визначити - кулька якого кольору знаходиться на конкретному місці, задавши мінімальну кількість питань



- Якщо задавати «погані» питання - чи є кулька на і-му місці синьою, то потрібно 20 таких питань
- Потрібні питання типу - чоловік чи жінка, живий чи мертвий тощо
- Давайте спробуємо розділити цю послідовність кульок на дві підпослідовності - найбільш оптимальні для подальшої класифікації цих кульок
- Які варіанти розділення на дві частини здаються прийнятними?

Іграшковий приклад – 2

- Маємо 9 синіх кульок і 11 коричневих

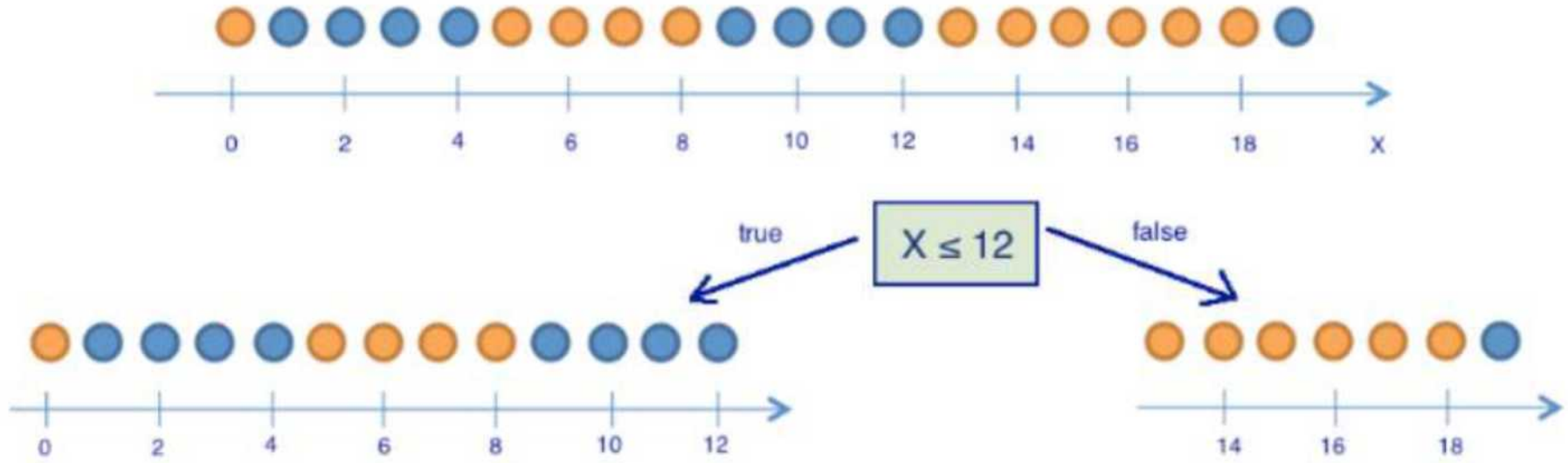


- Якщо ми навмання виберемо кульку, то вона з ймовірністю p_1=9/20 буде синьою та з ймовірністю p_2 =11/20 коричневою
- За формулою Шеннона початкова ентропія нашої системи складе:

$$S_0 = -\frac{9}{20}\log_2\frac{9}{20} - \frac{11}{20}\log_2\frac{11}{20} \approx 1.00$$

- А як зміниться ентропія, якщо ми розіб'ємо кульки на дві частини: з координатою менше або дорівнює 12 і більше 12

Іграшковий приклад – 3



- У лівій частині маємо 13 кульок, із яких 8 синіх і 5 коричневих
- Ентропія лівої частини кульок складе:

$$S_1 = -\frac{5}{13}\log_2\frac{5}{13} - \frac{8}{13}\log_2\frac{8}{13} \approx 0.96$$

- У правій частині маємо 7 кульок, із яких 1 синя і 6 коричневих
- Ентропія правої частини кульок складе:

$$S_2 = -\frac{1}{7}\log_2\frac{1}{7} - \frac{6}{7}\log_2\frac{6}{7} \approx 0.60$$

Як порахувати ентропію системи із двох отриманих частин?

З ваговим коефіцієнтом величини відповідної частини!

Іграшковий приклад – 4

- Оскільки ентропія - це ступінь невизначеності у системі, то її зменшення природно назвати приростом інформації (information gain, IG)
- При розділенні вибірки за ознакою Q (у нашому іграшковому прикладі це ознака "x<12") приріст інформації визначається як

$$IG(Q) = S_0 - \sum_{i=1}^{q} \frac{Ni}{N} S_i,$$

Ваговий коефіцієнт

де q- кількість частин (=груп) після розбиття,

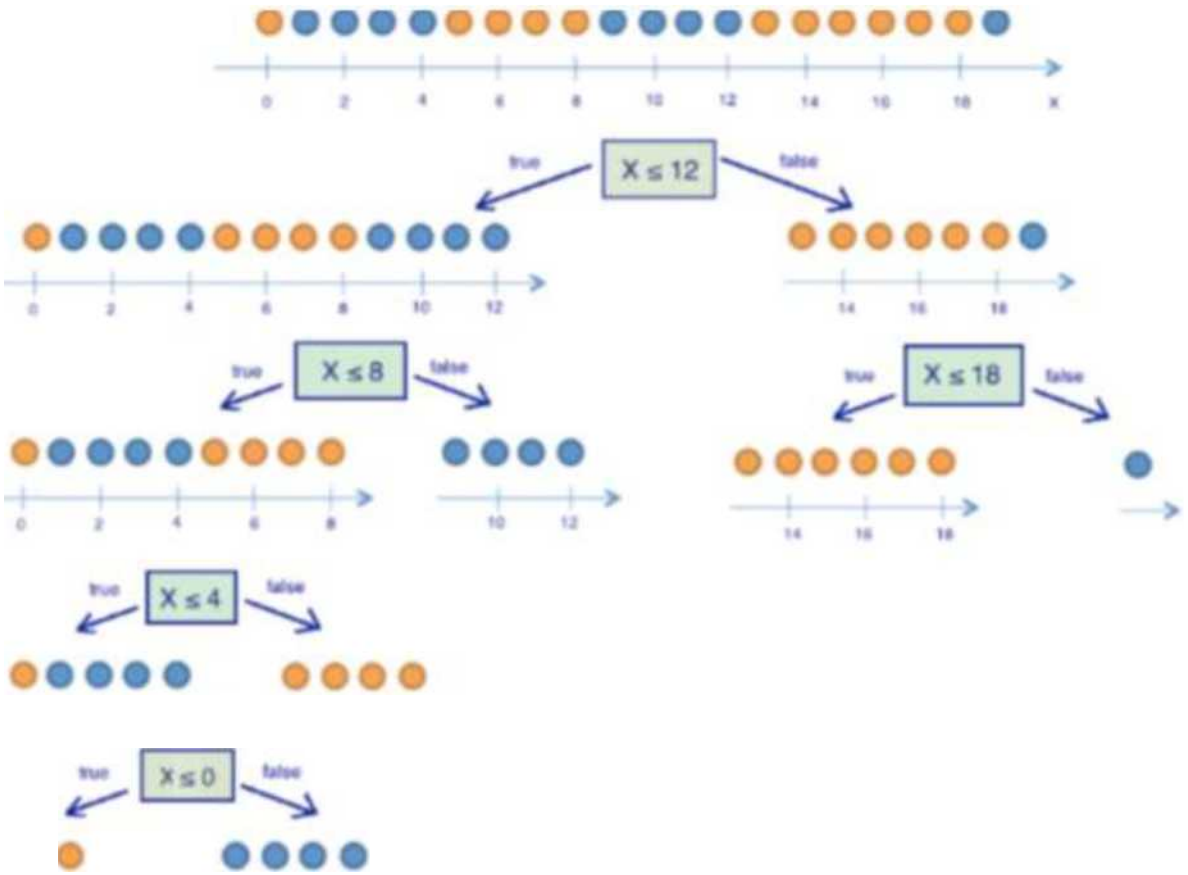
- N_i - число елементів вибірки, у яких ознака Q має і-е значення
- У нашому випадку після розділення маємо дві частини (q=2), одна з яких має 13 елементів (N_1=13), а друга - 7 (N_2=7)
- Приріст інформації складе

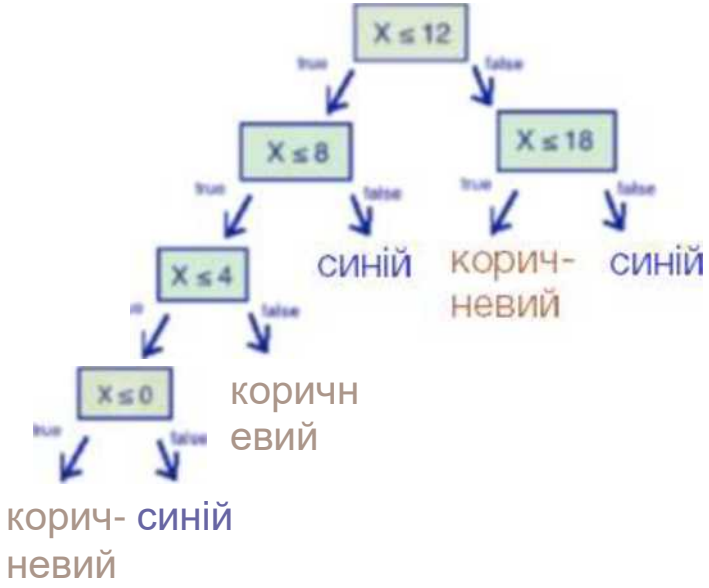
$$IG(x \leq 12) = S_0 - \frac{13}{20} S_1 - \frac{7}{20} S_2 \approx 0.16$$

Іграшковий приклад – 5

- Виходить, що розбивши наші кульки на дві зазначені групи, ми зменшили ентропію, тобто отримали більш упорядковану систему, ніж спочатку
- Продовжимо розділення кульок на групи до тих пір, поки не отримаємо у кожній групі кульки однакового кольору

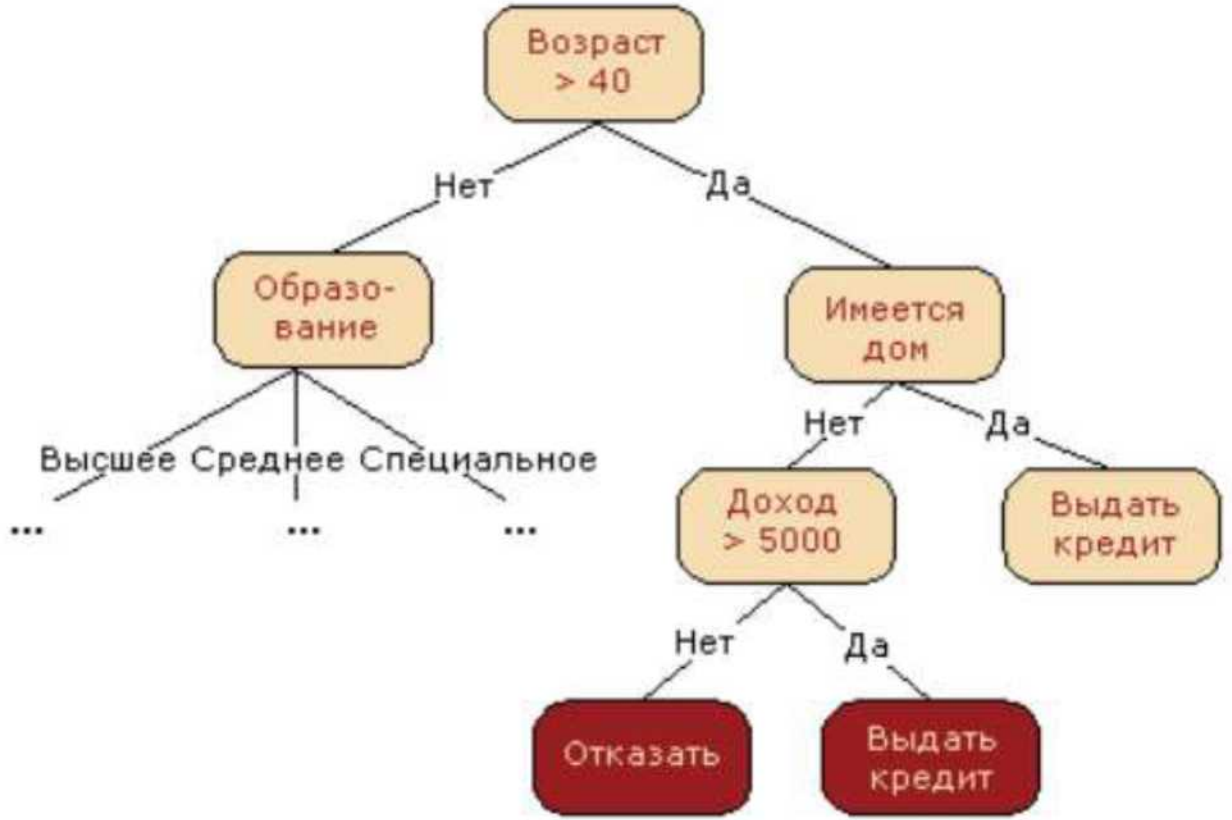
Іграшковий приклад – 6

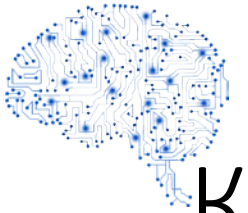




Виявилося, що за допомогою дерева прийняття рішень достатньо *2* кроки (=запитання), а не 20!

Кредитний скоринг - 1





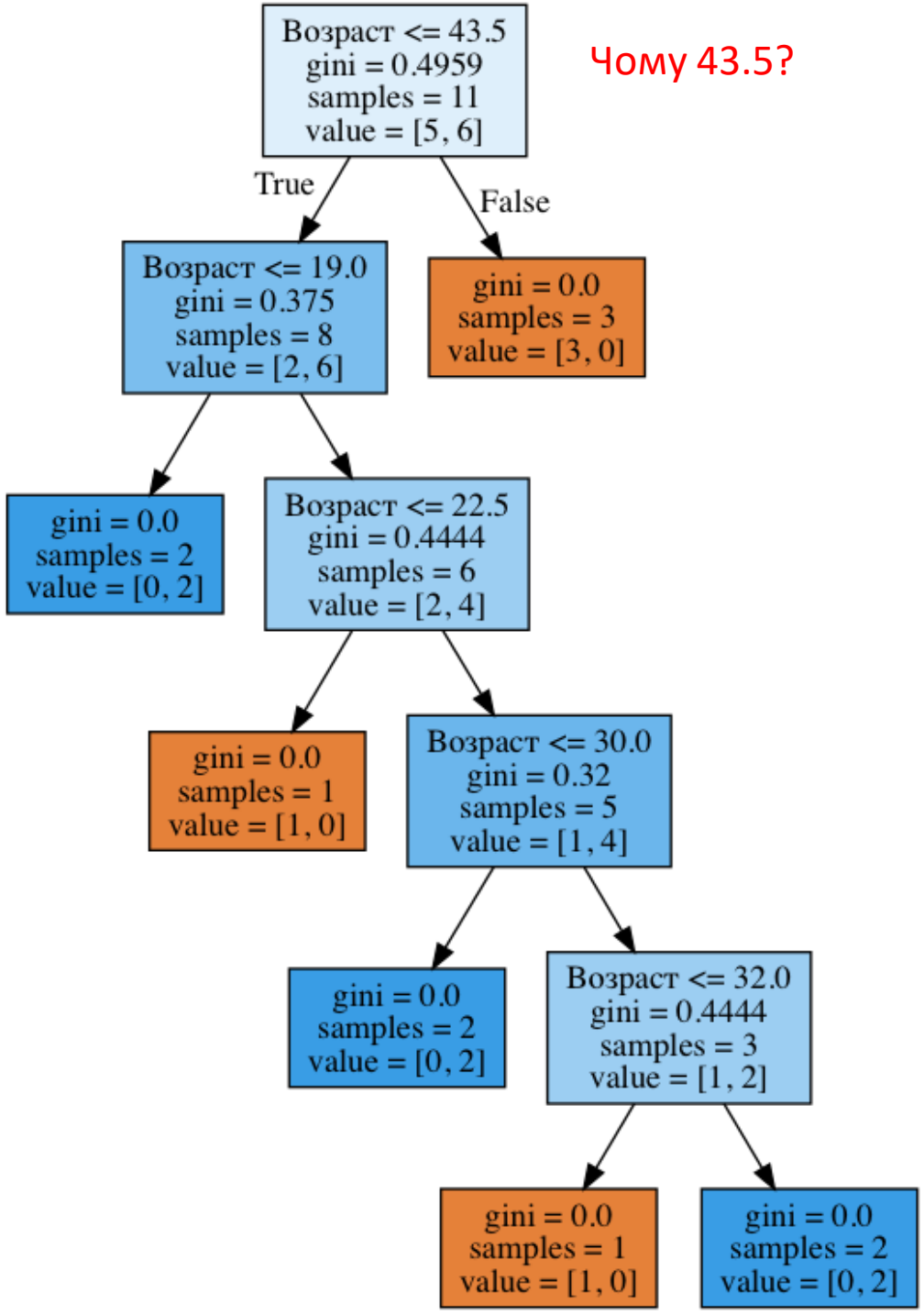
Кредитний скоринг - 2

- Початкові дані, впорядковані за віком
- Крайній лівий стовпчик -початковий номер рядка даних
- Як швидко розділити прохачів на дві категорії: тих, хто повертають кредити і не повертають?

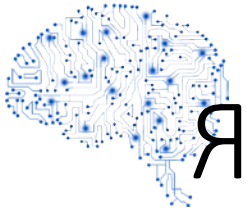
№	Вік	Неповернення кредиту
0	17	1
2	18	1
3	20	0
7	25	1
8	29	1
9	31	0
10	33	1
4	38	1
5	49	0
6	55	0
1	64	0

Дерево рішень

- На зображенні бачимо, що дерево задіяло 5 значень, з якими порівнюється вік: 43.5, 19, 22.5, 30 і 32 роки. Якщо придивитися, то це якраз середні значення між віками, при яких цільової клас "змінюється" з 1 на 0 або навпаки.
- Приклад: 43.5 - це середнє між 38 і 49 роками, клієнт, якому 38 років не повернув кредит, а той, якому 49 - повернув. Аналогічно, 19 років - середнє між 18 і 20 роками. Тобто в якості порогів для "нарізання" кількісної ознаки, дерево "дивиться" на ті значення, при яких цільової клас змінює своє значення.



Чому 43.5?



Якщо додасться нова ознака, то чи можна зменшити кількість спроб?

Початкові дані, впорядковані за віком

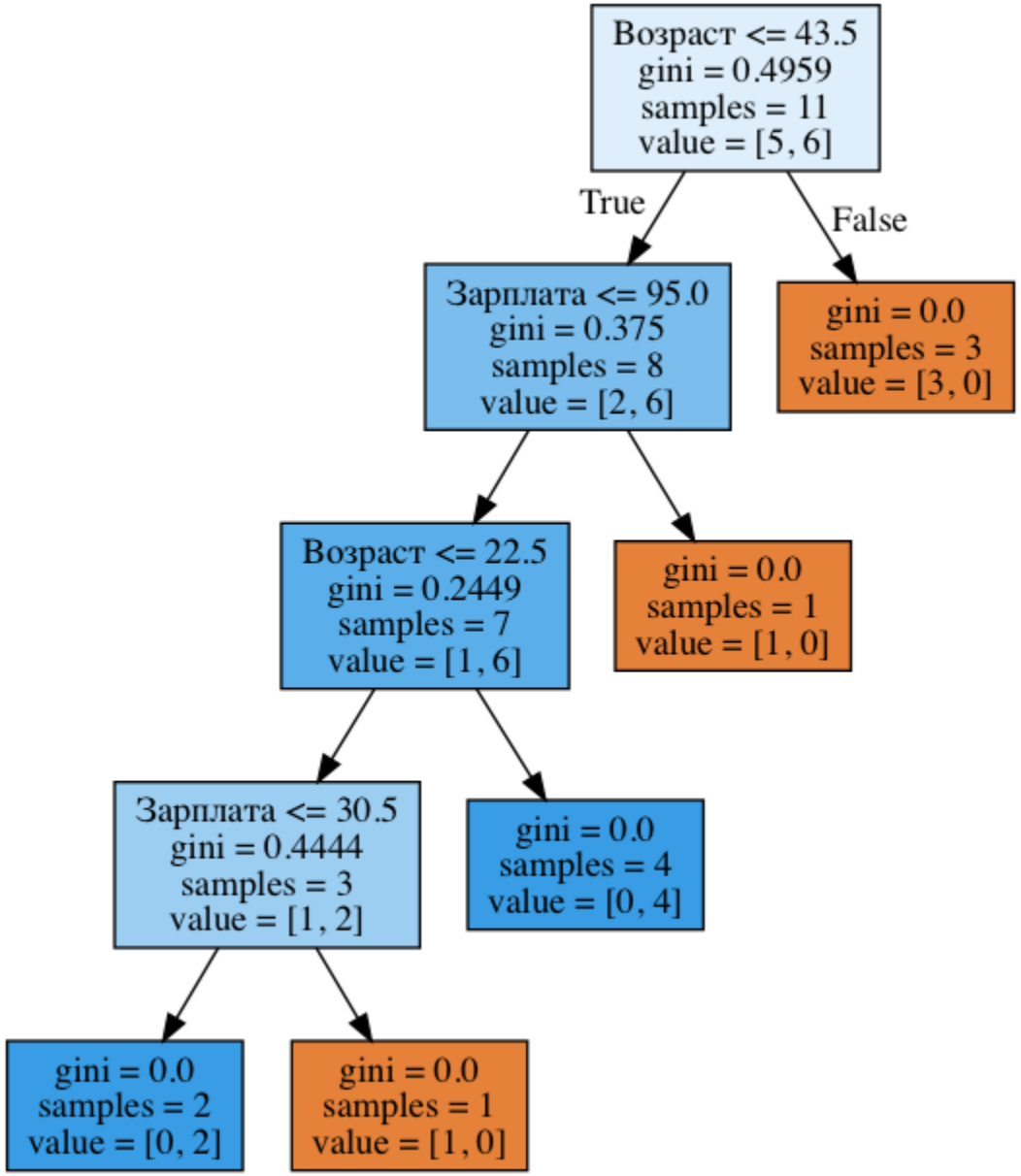
	Вік	Зарплата	Неповернення кредиту
0	17	25	1
2	18	22	1
3	20	36	0
7	25	70	1
8	29	33	1
9	31	102	0
10	33	88	1
4	38	37	1
5	49	59	0
6	55	74	0
1	64	80	0

Початкові дані, впорядковані за з/п

	Вік	Зарплата	Неповернення кредиту
2	18	22	1
0	17	25	1
8	29	33	1
3	20	36	0
4	38	37	1
5	49	59	0
7	25	70	1
6	55	74	0
1	64	80	0
10	33	88	1
9	31	102	0

Дерево рішень

- За 4 спроби все можна класифікувати



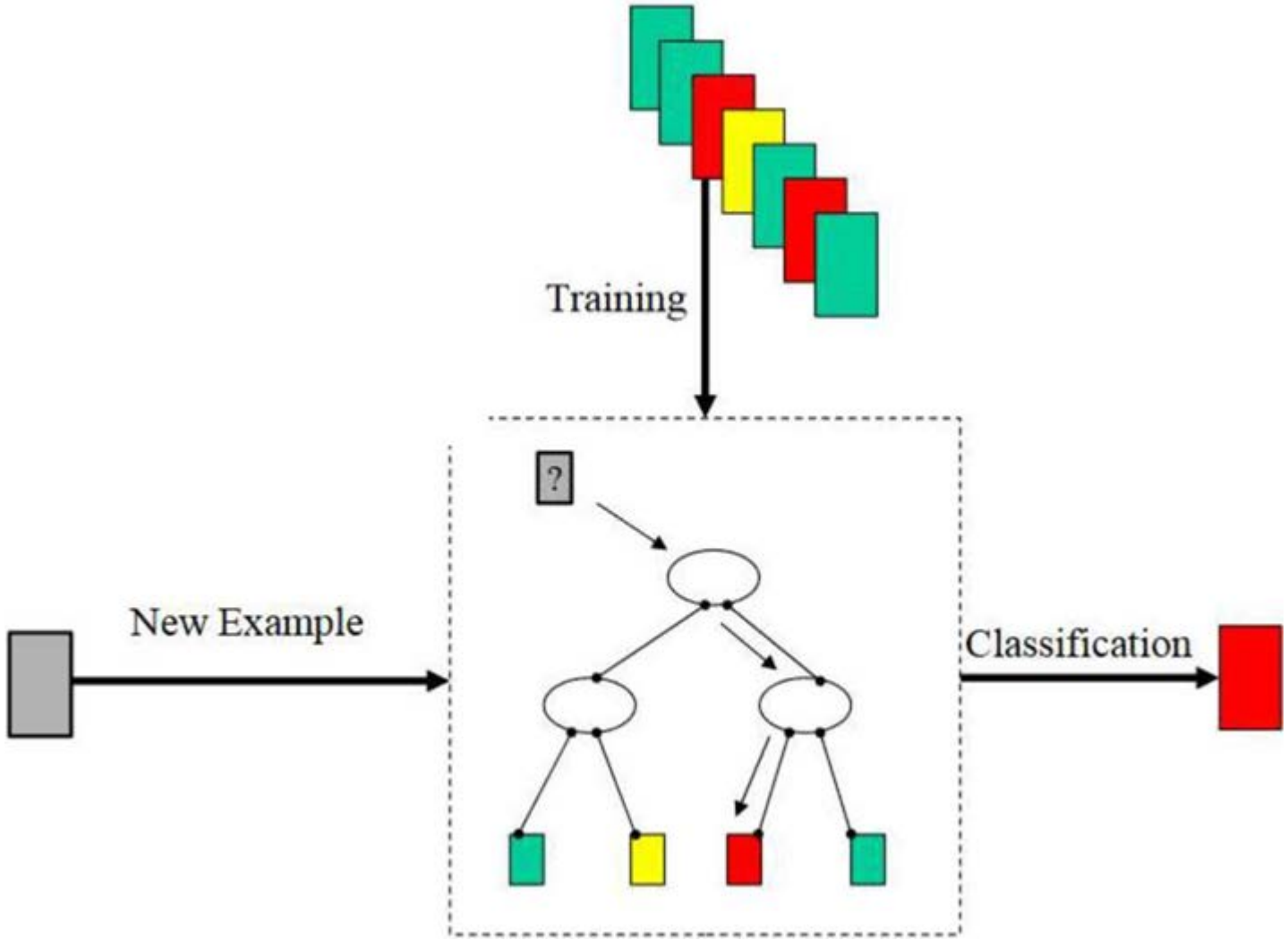
Класифікація

- Набір прикладів записів (навчальний набір) складається з декількох атрибутів (ознак)
- Дані, що описуються одновимірними функціями

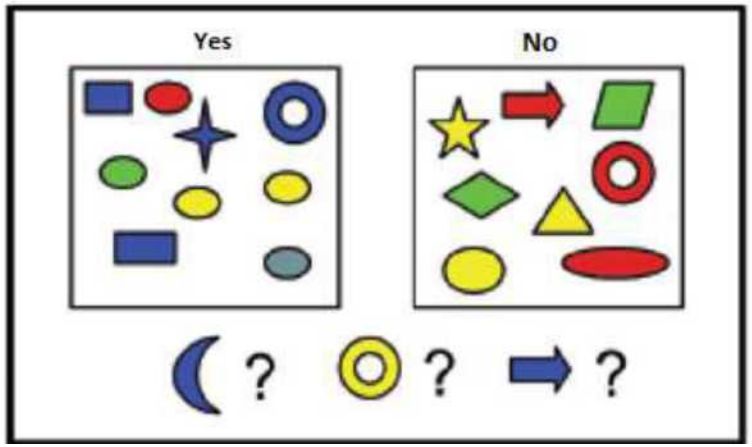
Атрибути:

- Категоріальні або номінальні (тобто стать: {жінка, чоловік}, колір: {зелений, оранжевий, червоний})
- Безперервні (тобто інтервал [0,10])

Навчання дерева рішень



Як дерева прийняття рішень використовуються для класифікації?

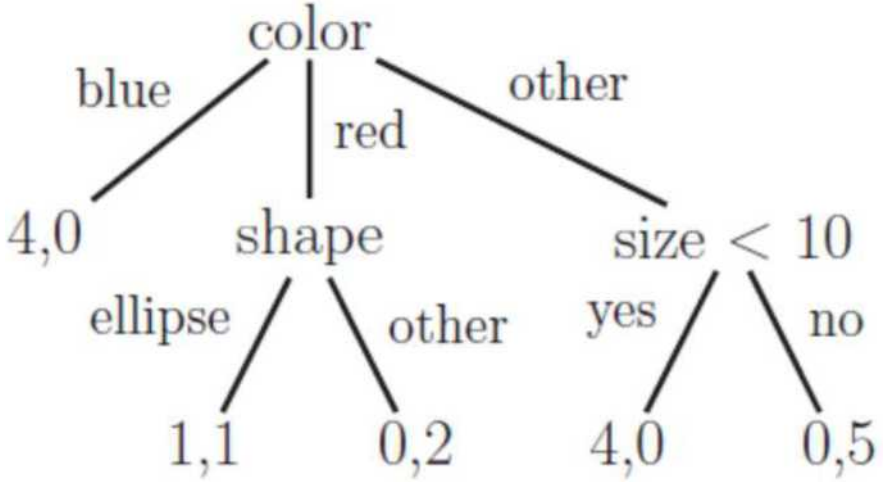


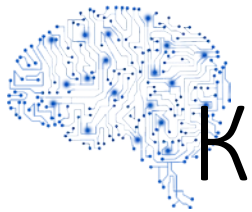
Features (attributes)

Color	Shape	Size (cm)	Label
Blue	Square	10	1
Red	Ellipse	2.4	1
Red	Ellipse	20.7	0

Cases

- (синя стрілка?): у нас не вистачає прикладів для правильної класифікації і ми можемо помилитися
- якщо синю стрілку потрібно класифікувати праворуч, то у гілку по кольору потрібно додати ще одну перевірку, тобто чим більше ми враховуємо різних випадків, тим, зазвичай, розростається дерево, ми перенавчаємося!
- а якщо ж це були помилки, то дерево сильно забруднюється, тому потрібно своєчасно зупинятися

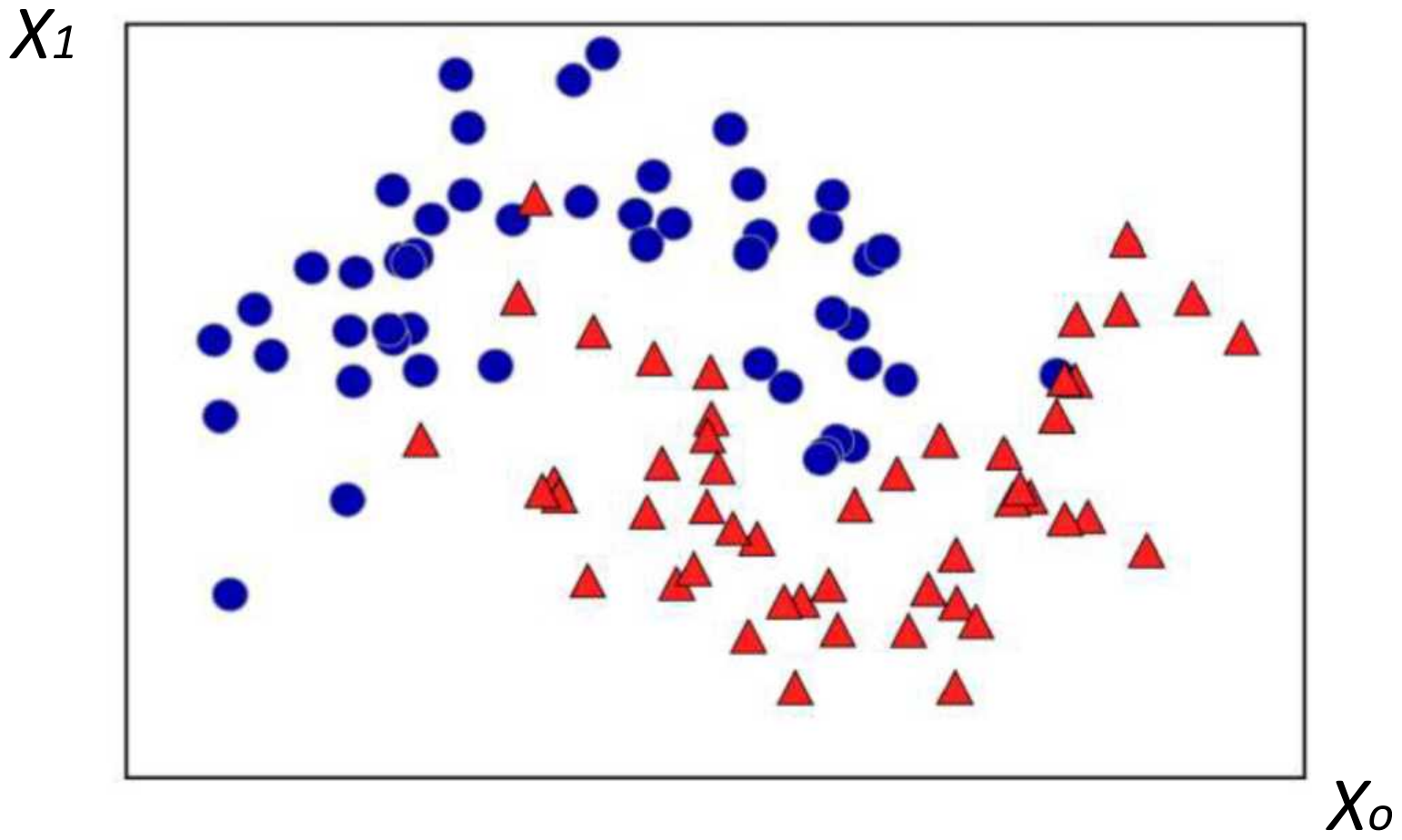




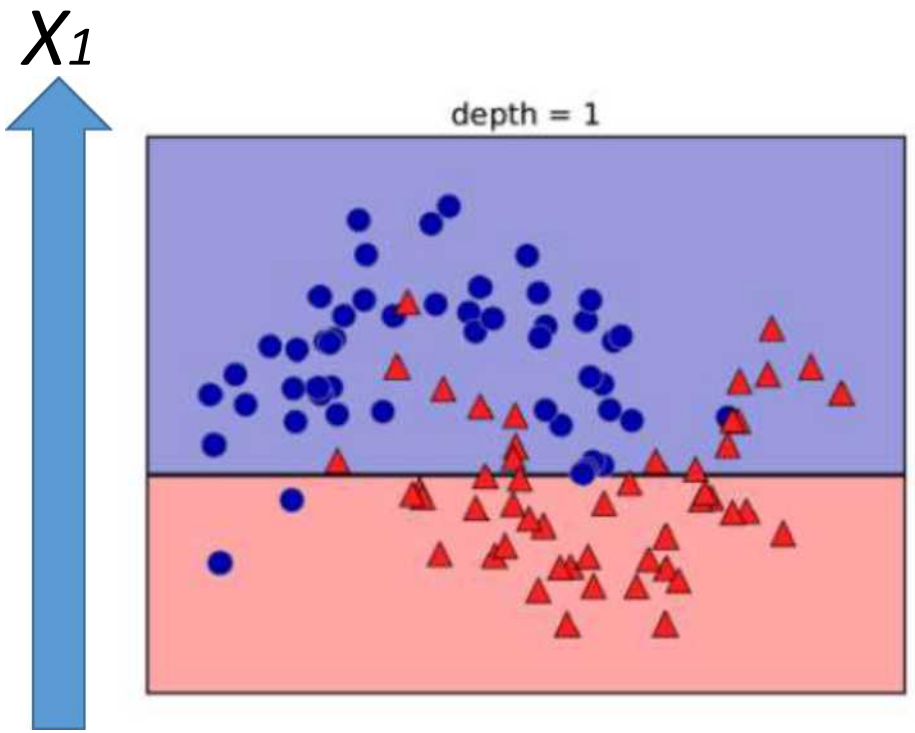
Компактне дерево для обчислення ефективності та подання

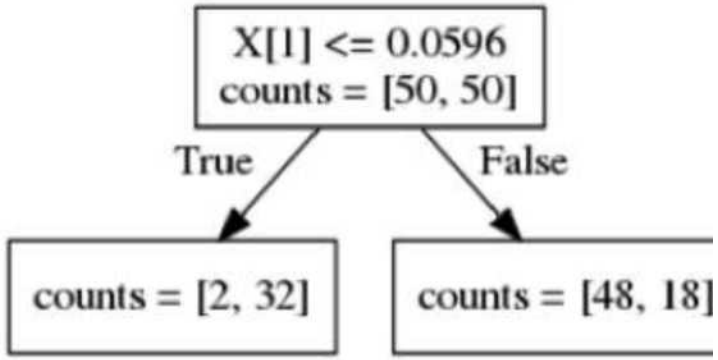
- Набір навчальних даних обмежений (неповний), може бути з викидами, а отже, необхідне узагальнення
- Кожне рішення не залежить від усіх змінних
- Багато змінних не впливають, ми можемо використовувати цей аспект для побудови компактного дерева

Класифікація з безперервними ознаками - 1



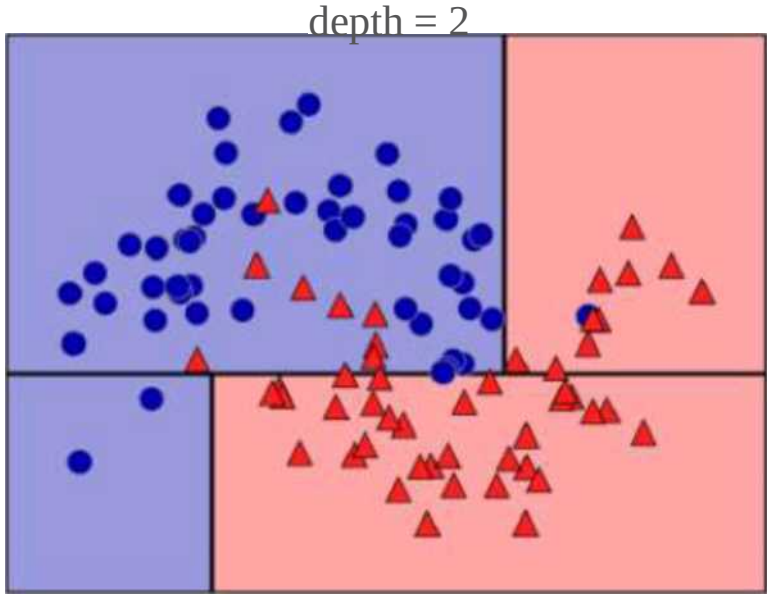
Класифікація з безперервними ознаками - 2



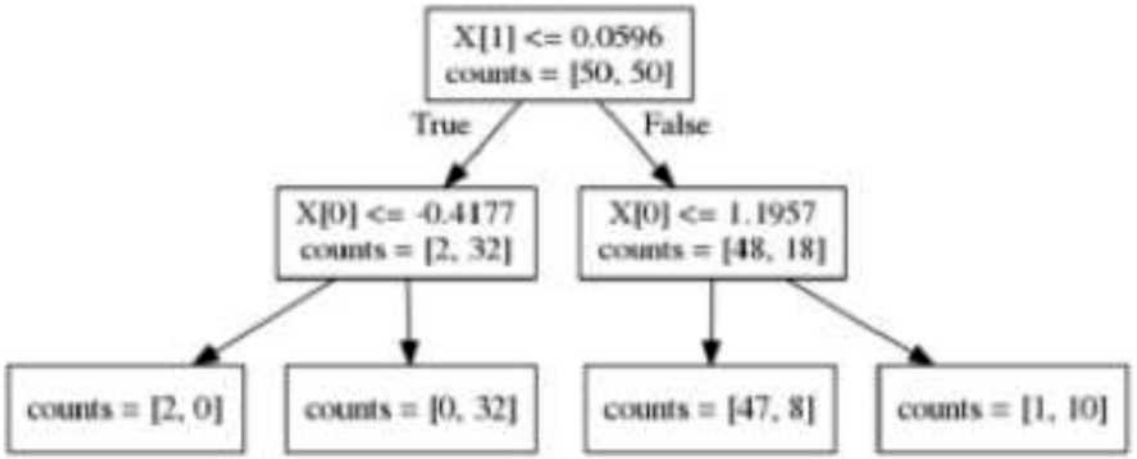


Класифікація з безперервними ознаками - 3

X_1

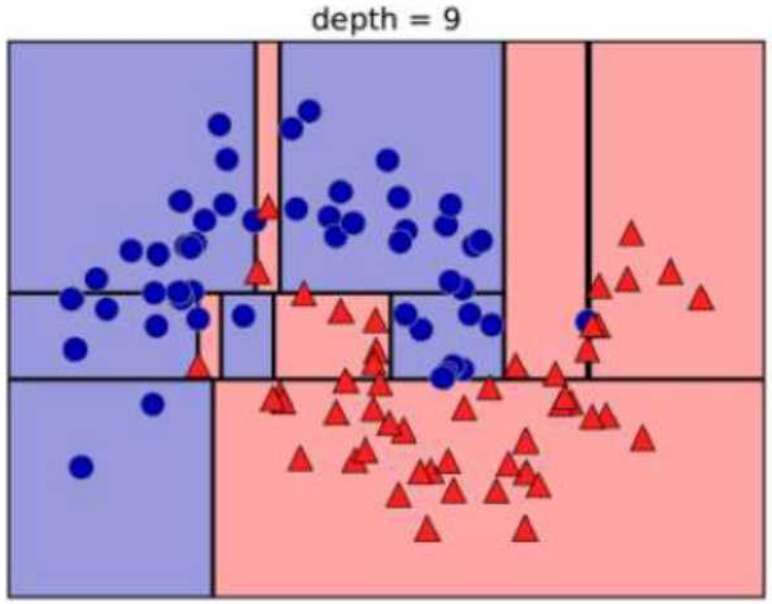


X_0

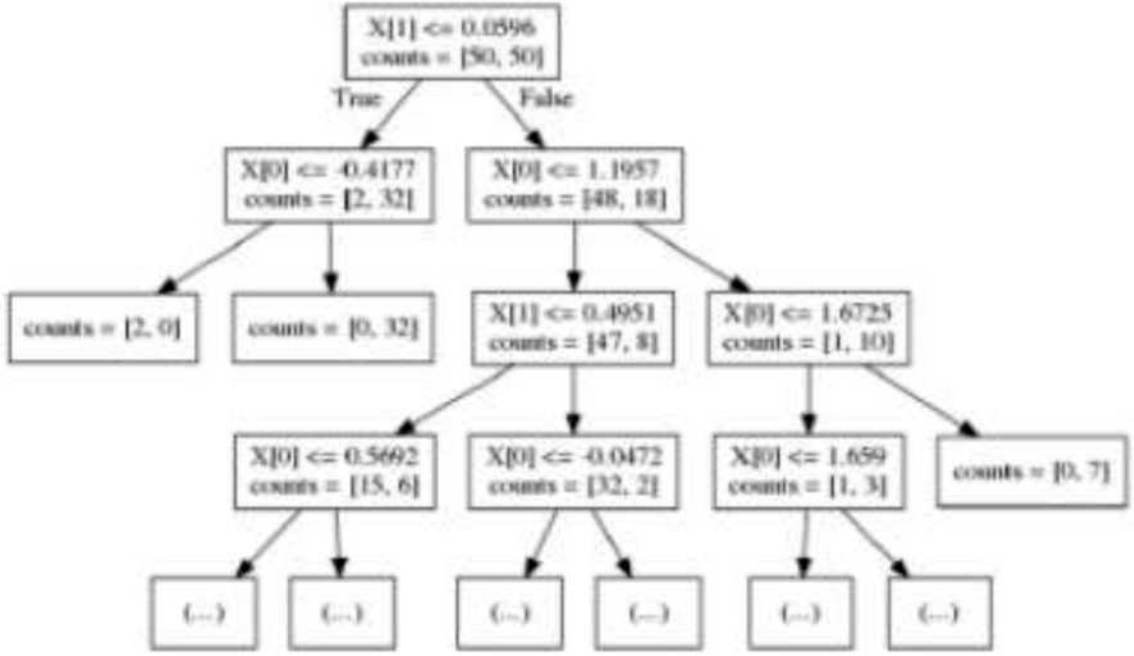


Класифікація з безперервними ознаками - 4

X_1

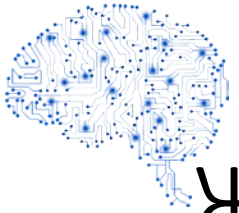


X_0



Складові алгоритмів побудови дерев прийняття рішень

- «В ширину» чи «вглиб»?
- Binary (CART) or multiway (ID3, CHAID) splits?
- Splitting criterion (greedy algorithm):
  - Information Gain (приріст інформації)
  - (Information) Gain ratio (нормалізований приріст інформації)
  - Gini impurity (неоднорідність Джині)
  - Classification error (помилка класифікації)
  - Chi-square test statistic for independence (критерій Хі-квадрат)
- Обрізання дерев (prepruning, postpruning)
- Обробка пропущених значень, паралелізація



Жадібний алгоритм

- Всі відомі алгоритми побудови дерев прийняття рішень - це жадібні алгоритми
- **Жадібний алгоритм** - це алгоритм, який на кожному кроці робить локально найкращий вибір в надії, що підсумкове рішення буде оптимальним
- Наприклад, алгоритм Дейкстри знаходження найкоротшого шляху в графі є жадібним, оскільки на кожному кроці шукається вершина з найменшою вагою, в якій ми ще не бували, після чого оновлюємо значення інших вершин
- При цьому можна довести, що найкоротші шляхи, знайдені в вершинах, є оптимальними
- Але не завжди жадібні алгоритми дають гарний результат: взяття «жертви» у шахах може привести до поразки

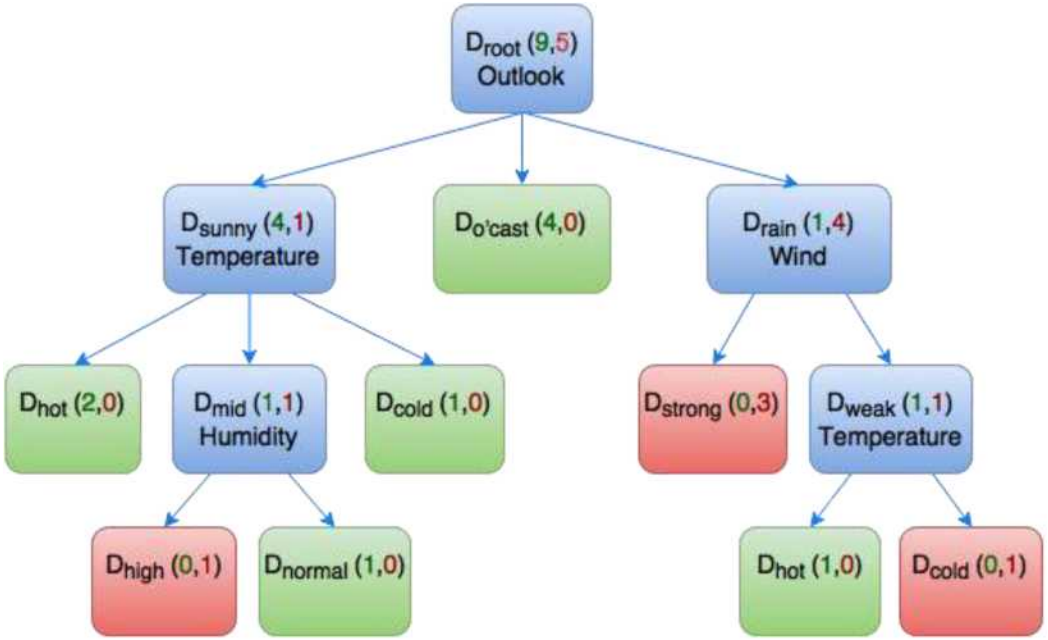
Приклад - 1

- Ми спостерігали за тренуванням спортсмена протягом двох тижнів щодо погодних умов.
- Ми хочемог розрахувати, чи буде він грати в певний день чи залишиться вдома.

Action	Wind	Temp	Outlook	Humidity
Play (P)	Weak	Hot	Sunny	High
Play	Strong	Hot	Sunny	High
Stay (S)	Weak	Hot	Rain	High
Play	Weak	Mid	Overcast	High
Stay	Strong	Cold	Rain	Normal
Play	Weak	Cold	Overcast	Normal
Stay	Strong	Cold	Rain	Normal
Play	Weak	Mid	Sunny	Normal
Play	Weak	Cold	Sunny	Normal
Play	Strong	Mid	Overcast	Normal
Stay	Weak	Mid	Sunny	High
Stay	Strong	Mid	Rain	High
Play	Weak	Hot	Overcast	Normal
Play	Weak	Cold	Rain	High

Приклад – 2

- Максимальний інформаційний приріст для даних



$$\boldsymbol{Gain(D,V) = E(D) - \sum \frac{N_v}{N} \times E(D_v)}$$

$$Gain(D,V) =$$

$$= E(D) - \left(\frac{1}{14}E(1st\ day) + \frac{1}{14}E(2nd\ day) + \cdots + \frac{1}{14}E(14th\ day)\right) =$$

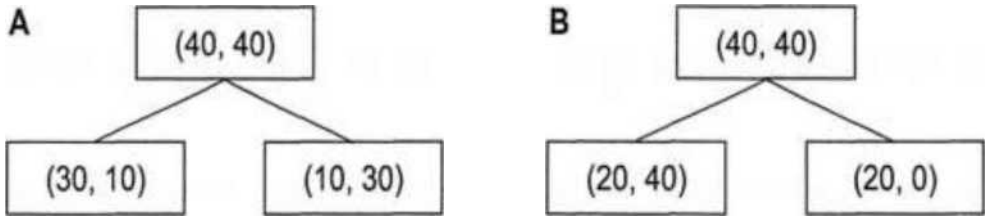
$$= 0.940 - \left(\frac{1}{14}\left(-\frac{1}{1}\log_2\left(\frac{1}{1}\right)\right) + \frac{1}{14}\left(-\frac{1}{1}\log_2\left(\frac{1}{1}\right)\right) + \cdots + \frac{1}{14}\left(-\frac{1}{1}\log_2\left(\frac{1}{1}\right)\right)\right) =$$

$$= 0.940 - 0 = 0.940$$

Чому погано?

- ніякого прогнозу не побудуєш
- головне - не можна знайти закономірності

Помилка класифікації



- Ми починаємо з набора даних D_p у батьківському вузлі D_p, який складається з 40 зразків з класу 1 та 40 зразків з класу 2, який ми розщеплюємо на два набори даних, відповідно на $D_{лівий}$ та $D_{правий}$. Приріст інформації, використовуючи помилку класифікації в якості критеріїв розширення, буде однаковим (IGE = 0,25) у обох сценаріях A і B:

$$I_E(D_p) = 1 - 0.5 = 0.5;$$

$$A: I_E(D_{левый}) = 1 - \frac{3}{4} = 0.25;$$

$$A: I_E(D_{правый}) = 1 - \frac{3}{4} = 0.25;$$

$$A: IG_E = 0.5 - \frac{4}{8}0.25 - \frac{4}{8}0.25 = 0.25;$$

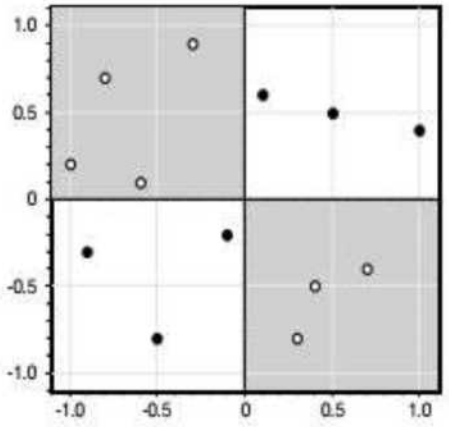
$$B: I_E(D_{левый}) = 1 - \frac{4}{6} = \frac{1}{3};$$

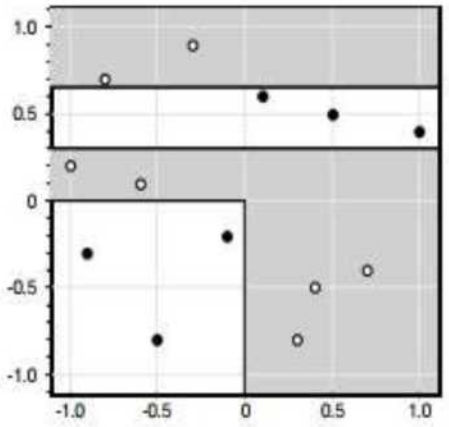
$$B: I_E(D_{правый}) = 1 - 1 = 0;$$

$$B: IG_E = 0.5 - \frac{6}{8} \times \frac{1}{3} - 0 = 0.25.$$

- Це погано - не розрізняємо ці дві ситуації

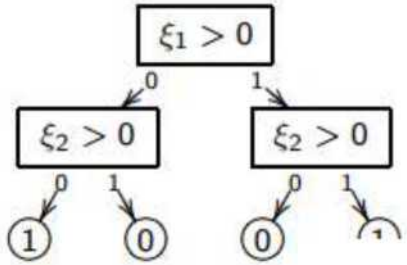
Жадні алгоритми можуть невиправдано ускладнювати дерева



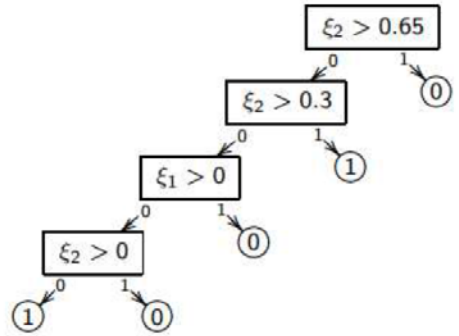


- Не розв’язується лінійним класифікатором (не може бути розділена прямою на два класи без помилок

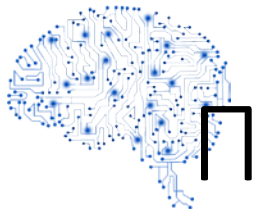
Оптимальне дерево



Дерево, яке побудує жадібний алгоритм



Як здогадатися, що спочатку потрібно провести лінію посередині?!

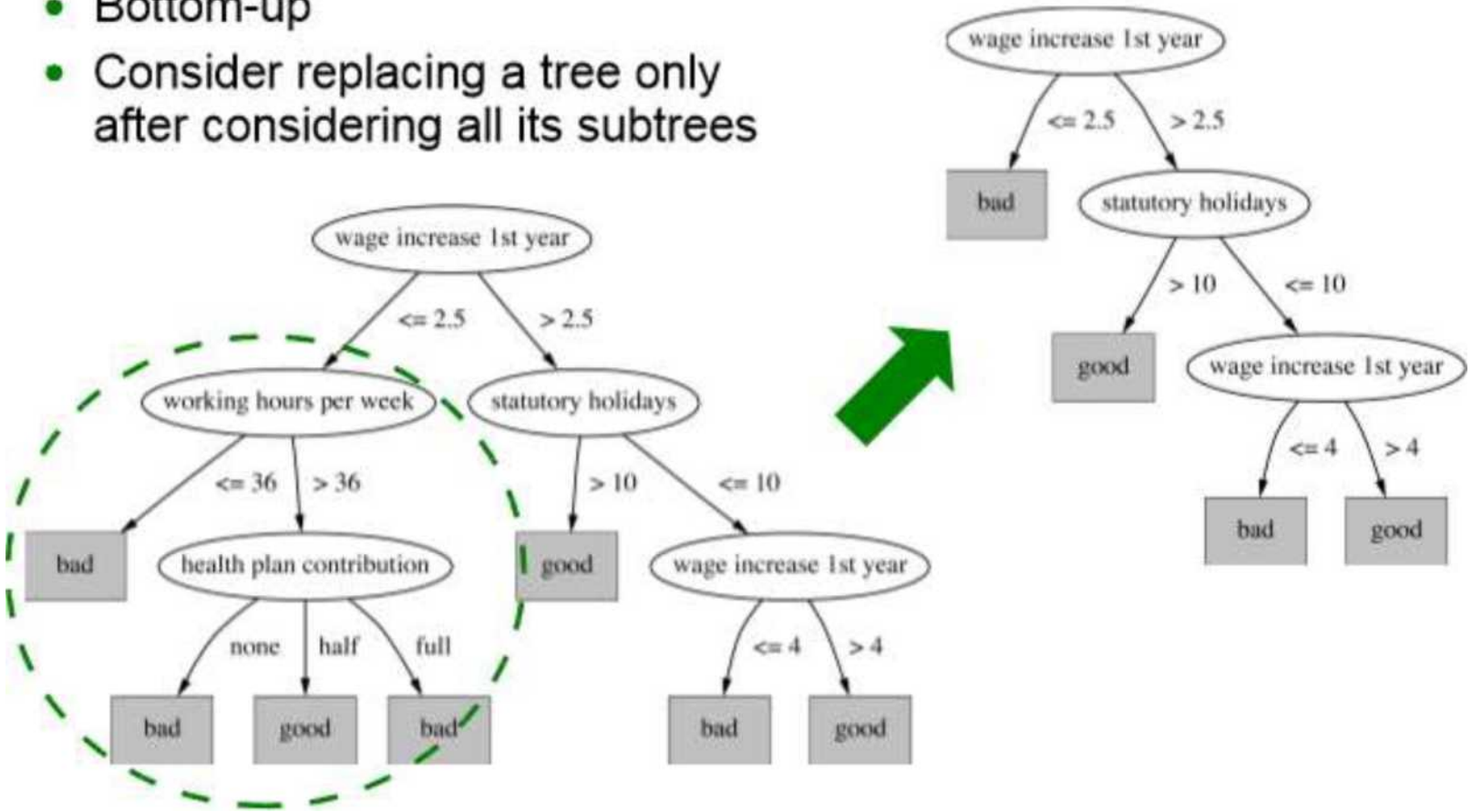


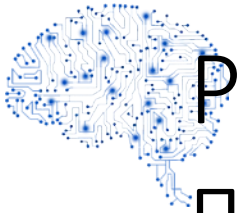
Прості способи попереднього обрізання дерев (prepruning)

- Обмеження глибини дерева. У цьому випадку побудова закінчується, якщо досягнуто задану глибину
- Визначення мінімальної кількості прикладів, які будуть міститися в кінцевих вузлах дерева. При цьому варіанті розгалуження тривають до того моменту, поки всі кінцеві вузли дерева не будуть чистими (відноситимуться до одного класу) або будуть містити не більше ніж задане число об'єктів

Постпрінінг: Заміна піддерева

- Bottom-up
- Consider replacing a tree only after considering all its subtrees

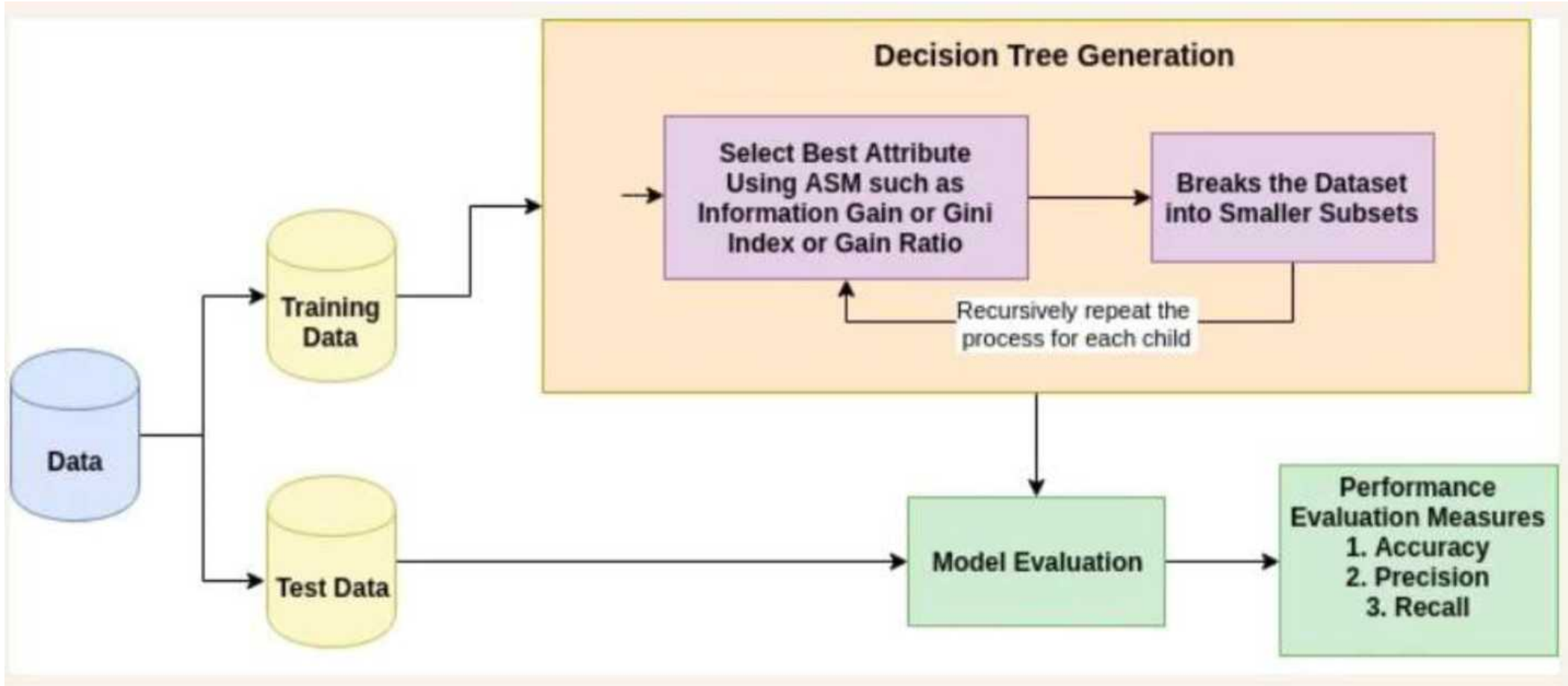


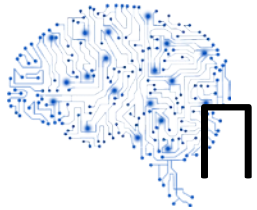


Різні методи побудови дерев прийняття рішень

- ID3, C4.5, C5.0
- CART
- CHAID

Як працює алгоритм дерева рішень?





Переваги і недоліки дерев прийняття рішень

- Переваги:
  - Інтерпретовність
  - Дозволяються різні типи даних
  - Можливість обходу пропущених значень
- Недоліки:
  - Перенавчання
  - Фрагментація
  - Нестійкість до шуму, складу вибірки, критерію розгалуження
- Способи зниження впливу недоліків
  - Підрізання дерев
  - Композиція (=ліс) дерев

Питання?