

1. Ентропия?

От записки по думи на Величков:

Най-добрият атрибут е този, който носи най-много информация за изхода от класификацията, Това определяме с Ентропия – с нея се измерва информацията (измерва се с битове) и по-точно **мярка за измерване липсата на информация**. Два класа P_+ и P_- :

$$Entropy(S) \equiv H(S) \equiv -(P_+ * \log_2 P_+) - (P_- * \log_2 P_-)$$

Станимир: средно ниво на информация, което ще получим от стойностите на някаква случайна величина

Приет отговор: ентропията е мярка за измерване на липсата на информация и колкото по-висока е, толкова повече неща не знаем, мерната единица е битове и ако е 1 значи че са равновероятни (при 2 класа)

2. Какво е 1 бит?

1 бит е информацията която получаващ при отговор на въпрос с два възможни отговора, които са равновероятни

3. Ентропията може ли да е повече от 1?

може, при повече от 2 характеристики

От Величков: Ако са 2 събития, равновероятни е 1-ца, ако са повече - повече, според броя изходи. При един изход е 0;

4. Какво представляват битовите при ентропията?

Станимир: **Информация, която сме получили от възможните изходи.**

5. Колко е един бит информация? / Какво означава 1 бит информация?

Ентропията при два класа с равна вероятност

Андрей: 1 бит е информацията която получаващ при отговор на въпрос с два възможни отговора, които са равновероятни

6. Когато разглеждаме ентропия може ли да има повече от 2 възможни класа?

Приет отговор: Да

7. Как се измерва ентропията?

Количеството информация, която сме получили от възможните изходи се записва в **битове**.

8. Ентропията намалява или се увеличава когато един клас е с по-голяма вероятност от друг?

Намалява.

9. Каква е разликата между supervised и unsupervised learning?

От Величков:

- Supervised learning – **учене с учител** – учителят ни казва как да подходим, всеки един пример на какво отговаря. Минава учителят и казва

своите наблюдения. даваме му множество от наблюдения и трябва да решим някакъв проблем. Трябва ни набор от данни, които са решавали този проблем с едни и същи характеристики. Въз основа на своите наблюдения изграждаме хипотеза. Като дойде нова снимка въз основа на хипотезата, можем да кажем какво е животното на снимката.

Имаме допълнителна колона клас, данните се разделят на тренировъчни, тези с които обучаваме, валидационни служат за по-добро представяне и тестови;

- Класификация – различните характеристики (казани от учителя, наричат се атрибути или features) на примерите се класифицират в класове. Например клас Котка, Куче... Тук имаме за стойност категория.

- Регресия – линейна функция, която минимизира сумата от грешките. Тук имаме числова стойност.

- Unsupervised learning – **учене без учител – тук имаме само наблюдения**, примери на различни езици и виждаме само думите и символите в тях

- Clustering – множество примери, без клас, защото не им знаем характеристиките (напр. снимки – групираме ги по пикселите), групираме ги в Групи, наречени клъстери. В информацията за данните няма колона за класа – няма такъв.

- клъстеризиране – имаме множество примери на които не знаем класа, групираме подобните примери в една група, тази група наричаме клъстер; подобен е на класа, но нямаме данни за него

Примерни алгоритми:

Учение с учител (Supervised) - kNN, NBC,(C4.5,C5.0 – ID3 – дърво на решенията), NN – невронна мрежа,

Учение без учител (Unsupervised)- K-means, EM-алгоритъм, DBSCAN

10. С какво се занимава ML-а?

Приет отговор: С трениране на модел

ML са компютърни алгоритми , които могат да се подобряват автоматично чрез опит и използване на данни. Алгоритмите за машинно обучение изграждат модел, базиран на примерни данни, известни като данни за обучение, за да правят прогнози или решения, без да бъдат изрично програмирани за това

11. Какви алгоритми има за локално търсене?

Hill Climbing, Simulated Annealing, local beam search, генетичните

Локално търсещи алгоритми - търсят в някакво под пространство от състояния, ползват се за оптимизационни проблеми. Алгоритмите за локално търсене претърсват пространството на решенията, като прилагат локални промени на параметрите, докато не се открие решение, което се смята за оптимално, или докато не бъде достигнат предварително определен времеви праг. Имат така наречена оценяща функция, от която печелим разглеждане на по-малко състояния, което рефлектира върху време и памет.

12. Проблем за локално търсене?

- ако си в някакво начално състояние и се опитваш да стигнеш в сет от състояния до крайно състояние как се случва това и оценъчна функция че има която ни подсказва към кои състояния да се насочиш
- не са пълни
- могат да зациклят в локален екстремум, затова ползваме рандъм рестарт

13. Какъв проблем решават локално търсещите алгоритми?

Оптимизационни са.

14. Какви алгоритми за изпълнение на условията съществуват

Приет отговор: **Backtracking & minconflicts & Constraint Propagation**

От някакъв файл:

Constraint Satisfaction Problems - 3 подхода

Пълно обхождане - **Backtracking** (обхождане с възврат; взимане на решенията едно по едно и ако се достигне до невъзможно решение се връщаме назад); general purpose техниките помагат за оптимизиране на алгоритъма

Constraint Propagation - разпространяване на грешката

Local Search - **minConflicts** (Hill Climbing) - когато нямаме удовлетворени ограничение имаме конфликт, целта е да минимизираме конфликтите и да удовлетворим ограниченията - nQueens

15. Задача за удовлетворяване на ограниченията и кои са основните подходи(Constraint satisfaction problem)

Станимир:

наредена тройка $\langle X, D, C \rangle$, където X е множество от N променливи, D е множество от множества от допустими стойности и C е множество от ограничения

имаме променливи, които ги инициализираме. И какво ни е решението? променливи, които удовлетворяват ограниченията

16. kMeans - алгоритъм за учене без учител. По зададено число k – броят

кълъстери, центрираме k нови точки, които наричаме центроиди

(произволно) и въртим този цикъл, докато спре да има промяна.

Причисляваме всеки един пример към най-близкият центроид на кълъстера, които общо образуват кълъстер. После изчисляваме центъра на кълъстера като взимам средното от всички и намирам новия центроид и той се обновява, при обновяването и смяната на центроидите, може примерите да сменят своето място и така, докато спрат да го правят – това е терминиращото състояние на алгоритъма.

1. Инициализация на центроидите(произволно)
2. Строят се между тях разделителни линии - симетрали
3. Разпределят се от областите в кълъстери(по близост)
4. Обновяване на центроидите(на итерации)
5. Край при липса на промяна в центроидите(няма повече ъпдейти)

17. Агломеративния - ботъм ъп подход, в началото всяка точка е клъстер и ги обединяваме
От записки: – **отдолу нагоре строим дървото като първо приемаме, че всеки един пример е отделен клъстер и после ги групираме.**

18. Как определяме броя на клъстерите? - elbow method за броя на клъстерите
Elbow method - евристика, която се използва за определяне на броя клъстери. Методът се състои в начертаване на explained variations като функция от броя на клъстерите и се избира “лакътя” на кривата като брой на клъстерите, които да се използват.

19. Защо се прави кастрене (pruning) на дървото на решенията
Избягване на overfitting - когато алготът се справя мн добре с тренинг данните и е почти безупречен, тогава overfit-ва, защото е обучен само по този тренинг и не може да се справи с предположения и тест датата бърка; непредставителна извадка;
Справяне с overfitting?

- Pre-pruning – да кажем предварително до къде да се построи дървото
- Post-pruning – построявате цялото дърво и после почвате да режете клончета

20. Невронни мрежи (как работят вътрешно, какво е backprop)

Имаме неврони (a_j), които се връзват с нашия неврон(a_i) и му предават информация. Тази информация се приема на входа чрез входната функция, която трансформира приетата информация под формата на единично представяне, което след това се трансформира чрез активационна функция, която обобщава информацията и накрая тя се предава към други неврони. В началото информацията и теглото на реброто се предават заедно на сумиращата(входна) функция. Освен всички неврони има и един допълнителен, нулев(a_0), нарича се Bias(Баяс). Сумиращата функция представлява сумата от произведенията на всички стойности от всички входни неврони умножени по теглото, с което те предават тази стойност.

Back-propagation learning – за обучаване на многослойни невронни мрежи (НЕ е вид невронна мрежа) – разпространяване на обратно – въз основа на какво сме получили го предаваме обратно на скрития слой и на входния слой. Служи, за да ъпдейтнем теглата(не разбирам мн?) и да пропагейтнем(???) грешката. Идеята е всички тегла да се ъпдейтнат спрямо грешката.

21. Обучаване на невронна мрежа

Има обучаващо множество и изчисляваме грешката и се опитваме да я минимизираме

22. Съжителни и предикатни логики

Предикатна логика включва в себе си съжителната логика, но за разлика от нея обръща повече внимание на простите твърдения

Предикатната логика е по-изразителна от съждителната

съждителните са прости (and or not, etc), но за сметка на това са по-бързи и леки предикатните са по-изразителни

23. Невронните мрежи използват ли локално търсещи алгоритми?

от ббб: май да, gradient descend - (от нета) В математиката градиентното спускане е итерационен оптимизационен алгоритъм от първи ред за намиране на локален минимум на диференцируема функция

24. как работи дърво на решенията - корена е най-добрия атрибут - тоест той ни носи най-много информация, и всички върхове са а атрибути без листата - те са класовете

25. условна независимост какво е -
възможност, при условие – условна вероятност

26. как се строи дърво на решенията -
Андрей: стремим се да сложим атрибутите които ни носят най голяма информация най-отгоре

27. Hill Climbing какво имаш на него?
знаем кои са близките състояние, (локално търсещ, защото по нататък не можем да видим), кое състояние избираме - което най-много ни привличава до целта, евристика използваме или фитнес функция(при задачата за удовлетворяване на ограничеснията)

28. предимства и недостатъци на семантичните мрежи

29. Инфо гейн - Показва колко значим е даден атрибут за определяне на дадения клас. Измерва се като намерим ентропията на класа и извадим ентропията на класа и атрибута

30. за обучението на невронните мрежи - първо определяш топологията (броя на слоевете и броя на ел. във всеки слой) после се задават случайни стойности на търсените тегла на връзките между ел, след което те итеративно се променят (уточняват) с помощта на Обучаващо правило (обучаващ алгоритъм)

31. Какво представляват данните при машиното обучение при учене с учител? - данните са dataset като на всеки ред има n+1 колони, като първите n стойности отговарят на n-те атрибута, а последната показва към кой клас принадлежи

32. Задача за учене с учител?

учене с учител – учителят ни казва как да podhodim

това е задача при която обучаваме модел (който може да е какъвто и да е алгоритъм дърво на решенията, наивен бейсов класификатор...) на база данни, които представляват примери, за които знаем към кои класове принадлежат. И след това искаме по подаден нов пример върху който не сме се обучавали да кажем какъв е класа на този пример

С дадено и търси се - дадено ти обучаващо множество от данни на които им знаеш класовете и търсиш хипотеза с помощта на която ако дойде нов пример от същия вид да можеш да му определиш класа

хипотези kНН, наивен бейсов класификатор

33. Какво е hill climbing?

hill climbing е локален вариант на greedy с "приоритетна опашка" с размер 1- пази се само най-добрия елемент , не е пълен, не е оптимален

Локално търсещи алгоритми - търсят в някакво под пространство от състояния, ползват се за оптимизационни проблеми. Алгоритмите за локално търсене претърсват пространството на решенията, като прилагат локални промени на параметрите, докато не се открие решение, което се смята за оптимално, или докато не бъде достигнат предварително определен времеви праг.

34. Разлика между продукционни правила и семантични мрежи?

Общо взето това, че продукционните правила се представят като двойки (условие, действие) и са по-прости, в тях по-лесно могат да се добавят нови елементи, както и да се правят изключения

Семантичните мрежи се представят с граф, в който състоянията са обекти(понятия), а дъгите - отношенията между тях. При тях по-трудно се постига наследяване и по-трудно могат да бъдат премахвани елементи.

Global Learning - Model-Based Learning, NBC, NN, ID3

Local Learning - kNN, Lazy Learning, kMeans

Instance-Based Learning - kNN

Lazy Learning - IBL, kNN, Locally weighted regression, Case-Based Reasoning(CBR) Eager - NN, SVN, ID3, NBC Model-Based Learning, дали моделът е предварително обучен (невронни мрежи, SVN, дърво на решенията, NBC).

Global Learning - Model-Based Learning - Eager Learning

Local Learning - Instance-Based Learning - Lazy Learning

Алгоритми с клъстеризация - k-Means, Gaussian Mixture(soft k-Means), Hierarchical clustering

Учене с учител (Supervised) - kNN, NBC,(C4.5,C5.0 – ID3 – дърво на решенията), NN – невронна мрежа,

Учене без учител (Unsupervised)- K-means, EM-алгоритъм, DBSCA