Propositional Logic & Reasoning

Vikram Pudi
IIIT Hyderabad

Knowledge Representation

- Knowledge Representation: expressing knowledge explicitly in a computer-tractable way
  - Knowledge Base: set of facts (or sentences) about the domain in which the *agent* finds itself
  - These sentences are expressed in a (formal) language such as logic

Why is it important?

- Reasoning: draw inferences from knowledge
  - answer queries
  - discover facts that follow from the knowledge base
  - decide what to do
  - etc.

Logic in General

- Logics are formal languages for representing information such that conclusions can be drawn
- Syntax: Describes how to make sentences
- Semantics: How sentences relate to reality. The meaning of a sentence is not *intrinsic* to that sentence.
- Proof Theory: A set of rules for drawing conclusions (inferences, deductions).

Logical Arguments

- All humans have 2 eyes.
- Kishore is a human.
  - Therefore Kishore has 2 eyes.
- All humans have 4 eyes.
- Kishore is a human.
  - Therefore Kishore has 4 eyes.
- Both are (logically) valid arguments.
- Which statements are true / false ?

Logical Arguments (contd)

- All humans have 2 eyes.
- Kishore has 2 eyes.
  - Therefore Kishore is a human.
- No human has 4 eyes.
- Kishore has 2 eyes.
  - Therefore Kishore is not human.
- Both are (logically) invalid arguments.
- Which statements are true / false ?

From English to Propositional Formulae

- "it is not the case that the lectures are dull": $\neg D$ (alternatively "the lectures are not dull")
- "the lectures are dull and the text is readable": $D \wedge R$
- "either the lectures are dull or the text is readable": $D \vee R$
- "if the lectures are dull, then the text is not readable": $D \rightarrow R$
- "the lectures are dull if and only if (iif) the text is readable": $D \leftrightarrow R$
- "if the lectures are dull, then if the text is not readable, Kishore will not pass": $D \rightarrow (\neg R \rightarrow \neg P)$

Why *formal* languages?

- Natural languages exhibit ambiguity.
  - Examples:
    - The boy saw a girl with a telescope
    - Our shoes are guaranteed to give you a fit
  - Ambiguity makes reasoning difficult / incomplete
- Formal languages promote rigour and thereby reduce possibility of human error.
- Formal languages help reduce implicit / unstated assumptions by removing *familiarity* with subject matter
- Formal languages help achieve generality due to possibility of finding *alternative interpretations* for sentences and arguments.

Propositional Logic

- Use letters to stand for "basic" propositions
- Complex sentences use operators for not, and, or, implies, iff.
- Brackets ( ) for grouping
  $(P \rightarrow (Q \rightarrow (\neg(R)))$ vs. $P \rightarrow (Q \rightarrow \neg R)$
- Omitting brackets
  - precedence from highest to lowest is: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
  - Binary operators are left associative
    (so $P \rightarrow Q \rightarrow R$ is $(P \rightarrow Q) \rightarrow R$)
- Questions:
  - Is $(P \vee Q) \vee R$ same as $P \vee (Q \vee R)$?
  - Is $(P \rightarrow Q) \rightarrow R$ same as $P \rightarrow (Q \rightarrow R)$?

Semantics (Truth Tables)

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \rightarrow Q$	$P \leftrightarrow Q$
True	True	False	True	True	True	True
True	False	False	False	True	False	False
False	True	True	False	True	True	False
False	False	True	False	False	True	True

- One row for each possible assignment of True/False to propositional variables
- **Important:** Above P and Q can be any sentence, including complex sentences

Terminology

- A sentence is valid if it is True under all possible assignments of True/False to its propositional variables (e.g. $P \vee \neg P$).
- Valid sentences are also referred to as tautologies
- A sentence is satisfiable if and only if there is *some* assignment of True/False to its propositional variables for which the sentence is True
- A sentence is unsatisfiable if and only if it is not satisfiable (e.g. $P \wedge \neg P$).

Semantics (Complex Sentences)

R	S	$(R \wedge S) \rightarrow (\neg R \vee S)$
True	True	
True	False	
False	True	
False	False	

Semantics (Complex Sentences)

R	S	$\neg R$	$R \wedge S$	$\neg R \vee S$	$(R \wedge S) \rightarrow (\neg R \vee S)$
True	True	False	True	True	True
True	False	False	False	False	True
False	True	True	False	True	True
False	False	True	False	True	True

Material Implication

- The only time $P \rightarrow Q$ evaluates to False is when P is True and Q is False
- This is known as a conditional statement or material implication
- English usage often suggests a causal connection between antecedent (P) and consequent (Q) – this is not reflected in the truth table
- So $(P \wedge \neg P) \rightarrow$ *anything* is a tautology!

Exercises

Given: A and B are true; X and Y are false, determine truth values of:

- $\neg(A \vee X)$
- $A \vee (X \wedge Y)$
- $A \wedge (X \vee (B \wedge Y))$
- $[(A \wedge X) \vee \neg B] \wedge \neg[(A \wedge X) \vee \neg B]$
- $(P \wedge Q) \wedge (\neg A \vee X)$
- $[(X \wedge Y) \rightarrow A] \rightarrow [X \rightarrow (Y \rightarrow A)]$

Entailment

- $S \Rightarrow P$ — whenever all the formulae in the set S are True, P is True
- This is a *semantic* notion; it concerns the notion of *Truth*
- To determine if $S \Rightarrow P$ construct a truth table for S, P
  - $S \Rightarrow P$ if, in any row of the truth table where all formulae of S are true, P is also true
- A tautology is just the special case when S is the empty set.

Entailment Example

P	$P \rightarrow Q$	Q
True	True	True
True	False	False
False	True	True
False	True	False

Modus Ponens

Therefore, $P, P \rightarrow Q \Rightarrow Q$

Exercises

Use truth tables to determine validity of:

- If it rains, Raju carries an umbrella. Raju is carrying an umbrella, therefore it will rain.
- If the weather is warm and the sky is clear, then either we go swimming or we go boating. It is not the case that if we do not go swimming, then the sky is not clear. Therefore, either the weather is warm or we go boating.

Formal Proofs

- Intend to formally capture the notion of proof that is commonly applied in other fields (e.g. mathematics).
- A proof of a formula from a set of premises is a sequence of steps in which any step of the proof is:
  1. An axiom or premise
  2. A formula deduced from previous steps of the proof using some rule of inference
- The last step of the proof should deduce the formula we wish to prove.
- We say that *S* follows from (premises) *P* to denote that the set of formulae *P* "prove" the formula *S*.

Soundness and Completeness

- A logic is sound if it preserves truth (i.e. if a set of premises are all true, any conclusion drawn from those premises *must* also be true).
- A logic is complete if it is capable of proving *any valid* consequence.
- A logic is decidable if there is a mechanical procedure (computer program) to prove *any* given consequence.

Inference Rules

- Modus Ponens: $P, P \rightarrow Q \Rightarrow Q$
- Modus Tollens: $P \rightarrow Q, \neg Q \Rightarrow \neg P$
- Hypothetical Syllogism: $P \rightarrow Q, Q \rightarrow R \Rightarrow P \rightarrow R$
- And-Elimination: $P_1 \wedge P_2 \wedge \ldots \wedge P_n \Rightarrow P_i$
- And-Introduction: $P_1, P_2, \ldots, P_n \Rightarrow P_1 \wedge P_2 \wedge \ldots \wedge P_n$
- Or-Introduction: $P_i \Rightarrow P_1 \vee P_2 \vee \ldots \vee P_n$
- Double-Negation Elimination: $\neg\neg P \Rightarrow P$
- Unit Resolution: $P \vee Q, \neg Q \Rightarrow P$
- Resolution: $P \vee Q, \neg Q \vee R \Rightarrow P \vee R$

Example Formal Proof

1. $A \vee (B \rightarrow D)$
2. $\neg C \rightarrow (D \rightarrow E)$
3. $A \rightarrow C$
4. $\neg C \quad / \therefore B \rightarrow E$
5. $\neg A \quad$ *3,4 (Modus Tollens)*
6. $B \rightarrow D \quad$ *1,5 (Unit Resolution)*
7. $D \rightarrow E \quad$ *2,4 (Modus Ponens)*
8. $B \rightarrow E \quad$ *6,7 (Hypothetical Syllogism)*

Exercises

Construct formal proof of validity for:

- If the investigation continues, then new evidence is brought to light. If new evidence is brought to light, then several leading citizens are implicated. If several leading citizens are implicated, then the newspapers stop publicizing the case. If continuation of the investigation implies that the newspapers stop publicizing the case, then the bringing to light of new evidence implies that the investigation continues. The investigation does not continue. Therefore, new evidence is not brought to light.
- *C*: The investigation continues. *N*: New evidence is brought to light. *I*: Several leading citizens are implicated. *S*: The newspapers stop publicizing the case.

Machine, Data and Learning

Machine Learning

- Scientific study of algorithms and statistical models that computer systems use
  - To perform a specific task effectively without using explicit instructions
  - Rely on patterns and inference instead.
- Involves
  - Building a **mathematical model** based on sample data, known as "training data“ to make predictions or decisions
  - No explicit programming done to perform the task

Machine Learning

- Term coined around 1960
- Why learn ? Why not just hire enough programmers and code in rules ?
  - Lots of patterns for an activity/event
  - Events can be dynamic
  - **Data** is increasing exponentially
  - **Data** is also in various formats [Text, Audio, Video]
  - Higher quality **data** due to cheaper storage
- Can be broadly classified into three categories
  - Unsupervised, Supervised and Reinforcement learning

Unsupervised Learning

- Takes a set of data that contains only inputs and finds structure in data E.g., Grouping or Clustering of data points

- **Marketing:** Finding groups of customers with similar behavior given a large database of customer data containing their properties and past buying records.
- **Biology:** Classification of plants and animals given their features.
- **Earthquake studies:** Clustering observed earthquake epicenters to identify dangerous zones.
- **World Wide Web:** Clustering weblog data to discover groups of similar access patterns.

Supervised Learning

- Builds mathematical model using data set that has both inputs and desired outputs E.g., Classification and Regression tasks

User ID	Gender	Age	Salary	Purchased
15624510	Male	19	19000	0
15810944	Male	35	20000	1
15668575	Female	26	43000	0
15603246	Female	27	57000	0
15804002	Male	19	76000	1
15728773	Male	27	58000	1
15598044	Female	27	84000	0
15694829	Female	32	150000	1
15600575	Male	25	33000	1
15727311	Female	35	65000	0
15570769	Female	26	80000	1
15606274	Female	26	52000	0
15746139	Male	20	86000	1
15704987	Male	32	18000	0
15628972	Male	18	82000	0
15697686	Male	29	80000	0
15733883	Male	47	25000	1

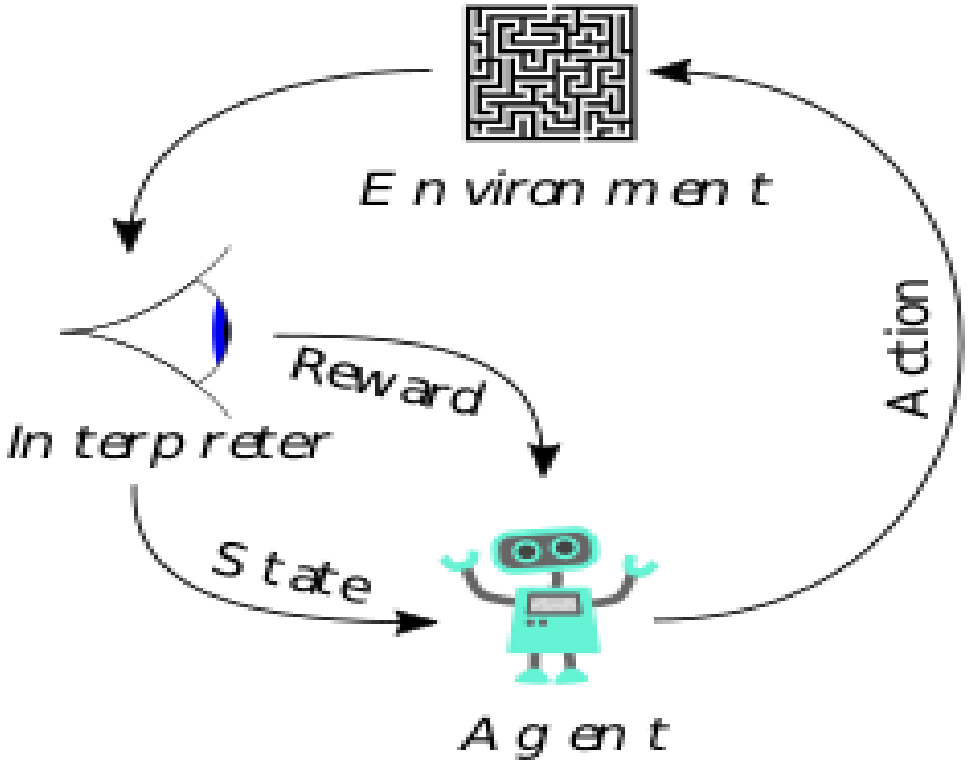
Figure A: CLASSIFICATION

Temperature	Pressure	Relative Humidity	Wind Direction	Wind Speed
10.69261758	986.882019	54.19337313	195.7150879	3.278597116
13.59184184	987.8729248	48.0648859	189.2951202	2.909167767
17.70494885	988.1119385	39.11965597	192.9273834	2.973036289
20.95430404	987.8500366	30.66273218	202.0752869	2.965289593
22.9278274	987.2833862	26.06723423	210.6589203	2.798230886
24.04233986	986.2907104	23.46918024	221.1188507	2.627005816
24.41475295	985.2338867	22.25082295	233.7911987	2.448749781
23.93361956	984.8914795	22.35178837	244.3504333	2.454271793
22.68800023	984.8461304	23.7538641	253.0864716	2.418341875
20.56425726	984.8380737	27.07867944	264.5071106	2.318677425
17.76400389	985.4262085	33.54900114	280.7827454	2.343950987
11.25680746	988.9386597	53.74139903	68.15406036	1.650191426
14.37810685	989.6819458	40.70884681	72.62069702	1.553469896
18.45114201	990.2960205	30.85038484	71.70604706	1.005017161
22.54895853	989.9562988	22.81738811	44.66042709	0.264133632
24.23155922	988.796875	19.74790765	318.3214111	0.329656571

Figure B: REGRESSION

Reinforcement Learning

- Concerned with how software agents should take actions in an environment to maximize cumulative reward E.g. Autonomous vehicles, Computer games



Some Applications

- Search engines
- Information retrieval
- Recommendation systems
- Credit card fraud detection
- Disease diagnosis
- Election prediction
- Image processing
- Speech translation
- …

AlphaGo

- First computer Go program to defeat a 9-dan professional player
- Uses Monte Carlo Tree search algorithm based on knowledge learned by a deep learning method
- Beat World No. 1 ranked player in 2017
  - Retired after this match
- https://deepmind.google/technologies/alphago/
- https://www.youtube.com/watch?v=WXuK6gekU1Y

- AlphaGo Zero – Version without human data and stronger than AlphaGo [defeated 100-0]

AlphaZero & MuZero

- AlphaZero, a generalized version of AlphaGo Zero Took 4 hours to learn Chess and defeat reigning world computer chess champion 28 to 0 in 100 matches
- https://www.youtube.com/watch?time_continue=7&v=tXlM99xPQC8

- MuZero: Master games without knowing rules
- Uses approach similar to AlphaZero, developed in 2019
- Trained via self-play and play against AlphaZero with no access to rules, opening books or endgame tables
- Viewed as significant advancement over AlphaZero

AlphaFold: solution to a 50 year old grand challenge in biology

- https://www.youtube.com/watch?v=KpedmJdrTpY
- https://deepmind.google/discover/blog/alphafold-a-solution-to-a-50-year-old-grand-challenge-in-biology/
- Figuring out what shapes proteins fold into is known as the “protein folding problem” - grand challenge in biology for the past 50 years
- Focus of intensive scientific research for many years, using a variety of experimental techniques such as nuclear magnetic resonance and X-ray crystallography.

AlphaFold

- Number of ways a protein could theoretically fold before settling into its final 3D structure is astronomical.
- Cyrus Levinthal estimated 10^300 possible conformations for a typical protein.
- Estimated would take longer than the age of universe to enumerate all possible configurations. Yet in nature, proteins fold spontaneously, some within milliseconds - referred to as Levinthal's paradox.

Year 2023 in review for AI

- Article titled 2023: The Crazy AI Year by Nisha Arya, KDnuggets
- Many media sources claim year 2023 can be considered the year of AI
- Jan:
  - With huge buzz around ChatGPT Microsoft announced $10 billion funding in OpenAI
- Feb:
  - Google came up with BARD. Microsoft came up with its Bing chatbot

Year 2023 in review for AI

- Mar:
  - Access to Bard was given to a limited number of people to kickstart the Google GenAI journey.
  - Initiated a domino effect with Adobe introducing Firefly and Canva introducing their virtual design assistant.
  - OpenAI also launched APIs for ChatGPT, as well as their text-to-speech model called Whisper. On the 14th of March, OpenAI released its most advanced model GPT-4.

Year 2023 in review for AI

- Apr:
  - Announcement of Google DeepMind - a combination of Google Research and DeepMind.
  - Russia's Sberbank released ChatGPT rival GigaChat
  - HuggingFace also entering the market with the release of an AI chatbot to rival ChatGPT called HuggingChat
- May:
  - Google announced the Bard chatbot to the public - added some fuel to the GenAI fire with Microsoft revealing its debut AI assistant for Windows 11.

Year 2023 in review for AI

  - Market capitalization of NVIDIA topped $1 trillion for the first time, holding its status as the AI chip leader.
  - Elon Musk's new brain implant startup, called Neuralink, in which the company aims to create and implant AI-powered chips in people’s brains. This was approved by the FDA for human trials.
- Jun:
  - Apple’s Vision Pro, the AI-powered augmented reality headset was developed to take immersive experiences to the next level.
  - European Parliament made some negotiations about the EU AI Act, with 499 votes in favor, 28 against, and 93 abstentions.

Year 2023 in review for AI

  - McKinsey predicted that GenAI has the potential to add up to $4.4 trillion in value to the global economy.
- July:
  - Meta introduced Llama 2, an open-source Large Language Model (LLM) which was trained on a mix of publicly available data, and designed to drive applications such as OpenAI's ChatGPT, Bing Chat, and other modern chatbots.
  - Anthropic also released Claude 2, which dethroned ChatGPT and has it shaking in its boots.
  - The safety around AI is becoming a popular topic as LLMs are becoming a part of our day-to-day lives.

Year 2023 in review for AI

- Microsoft announced that it will charge customers $30 per month to use Microsoft 365 Copilot.

- Aug:
  - Google said that it would also be charging $30 per month for users to make use of their GenAI tools in their Duet AI for Workspace.
  - OpenAI introduced custom instructions to get the most out of ChatGPT. Poe a chatbot service that allows you to use state-of-the-art models such as Claude +, GPT-3.5-Turbo, and GPT-4.

Year 2023 in review for AI

- Sept:
  - Amazon announced a $4 billion investment in OpenAI competitor Anthropic.
  - OpenAI continues with its quest to visualize content with a Canva plugin for ChatGPT.
- Oct:
  - We experienced the Executive Order on the Safe, Secure, and Trustworthy Development and Use of Artificial Intelligence. This also was shaking up the AI world, with CEOs, leaders, and others having contradicting opinions about the implementation of AI systems into society.

Year 2023 in review for AI

- Nov:
  - Elon Musk's AI startup, xAI, unveiled the AI chatbot "Grok", AWS with the release of Amazon Q, and Pika 1.0 from StabilityAI.
  - OpenAI also held its first developer event in November, where it delved into GPT-4 Turbo and the GPT Store.
  - OpenAIs CEO Sam Altman getting fired by the board out of nowhere. He was immediately offered a job by Microsoft with OpenAI employees threatening to resign if Sam Altman did not come back and claim his position as CEO. So now he is back, with some new board members as well as a new "observer" role for Microsoft.

Year 2023 in review for AI

- Dec:
  - Google came to shake the market again with their 3 variant family of large language models and ChatGPT's new rival: Gemini.
  - We already know that OpenAI is looking into GPT 5, 6, and 7. So let’s see what 2024 January has to bring.

Machine, Data and Learning

Selected slides for lectures on ML Topic

Generalization & Goodness of Fit

- Based on Chapter 1 of Python Machine Learning by Example by Yuxi Liu

- **Generalization** refers to how well the concepts learned by a ML model generalizes to specific examples or data not yet seen by the model.
  - …
- **Goodness of fit** describes how well a model fits for a set of observations.
  - Overfitting and Underfitting

Overfitting

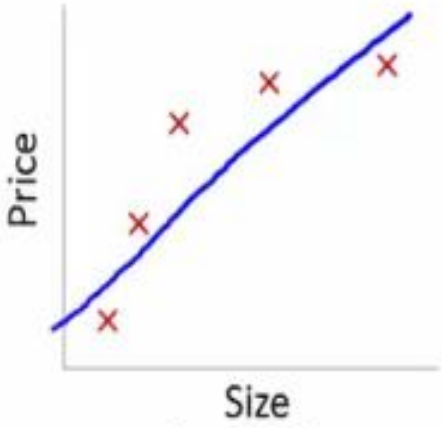
- Phenomenon of extracting too much information from training sets or memorization can cause overfitting
  - Makes ML model work well with training data called **low bias**
  - Bias refers to error due to incorrect assumptions in learning algorithm
  - However, does not generalize well or derive patterns, performs poorly on test datasets called **high variance**
  - Variance measures error due to small fluctuations in training set

Underfitting

- Model is underfit if it does not perform well on training sets and will not do so on test sets
- Occurs when we are not using enough data to train or if we try to fit wrong model to the data
  - E.g., if you do not read enough material for exam or if you prepare wrong syllabus
- Called **high bias** in ML although **variance is low** [i.e. consistent but in a bad way]
- May need to increase number of features since it expands the hypothesis space.

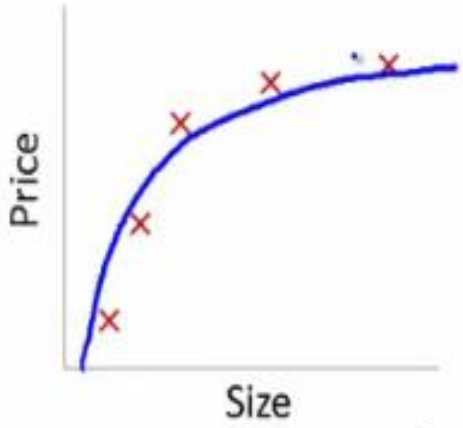
Goodness of fit

- For same data:



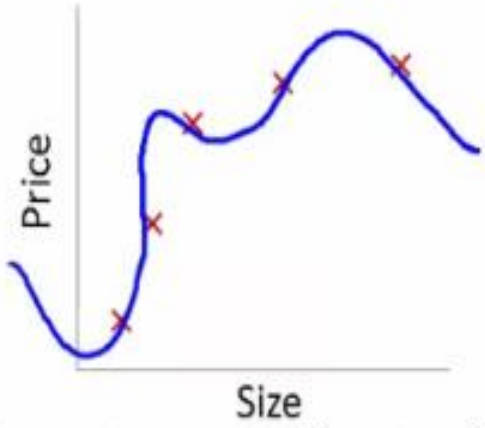
$\theta_0 + \theta_1 x$

High bias
(underfit)



$\theta_0 + \theta_1 x + \theta_2 x^2$

"Just right"



$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

High variance
(overfit)

Bias-Variance Tradeoff

- If the model is too simple and has very few parameters then it may have high bias and low variance
- If the model has large number of parameters it may have high variance and low bias
- We need to find a right/good **balance** without overfitting or underfitting the data
- As more parameters are added to a model
  - Complexity of the model rises
  - Variance becomes primary concern while bias falls steadily.

Bias-Variance Tradeoff

- Suppose a training set consists of points x1, …, xn and real values yi associated with each point xi
- We assume there is a function y = f(x) + Ɛ, where the noise Ɛ has zero mean and variance σ^2
- Find $\hat{f}(x)$, tht approximates f(x) as well as possible
- To measure how well the approximation was performed, we minimize the mean square error $(y - \hat{f}(x))^2$
- A number of algorithms exist to find $\hat{f}(x)$, that generalizes to points outside of our training set

Bias-Variance Tradeoff

- Variance measures how far a set of (random) numbers are spread out from their average value.
- Measured as expectation of the squared deviation of a random variable from its mean.

$$Var(X) = E[(x-\mu)^2]$$

$$Var(X) = E[(x-E[x])^2]$$

$$= E[x^2 - 2xE[x] + E[x]^2)$$

$$= E[x^2] - 2E[x]E[x] + E[x]^2$$

$$= E[x^2] - E[x]^2$$

Bias-Variance Tradeoff

- Turns out expected (mean squared) error of $\hat{f}$ on an unseen sample in general can be decomposed as:

$$E\left[\left(y - \hat{f}(x)\right)^2\right] = (Bias[\hat{f}(x)])^2 + Var[\hat{f}(x)] + \sigma^2$$

where,

$$Bias\left(\hat{f}(x)\right) = E[\hat{f}(x) - f(x)]$$

$$= E[\hat{f}(x)] - E[f(x)] = E[\hat{f}(x)] - f(x)$$

$$Since\ f\ is\ deterministic, E[f] = f$$

and

$$Var[\hat{f}(x)] = E[\hat{f}(x)^2] - E[\hat{f}(x)]^2$$

Note that all three terms are positive

Bias-Variance Tradeoff

- Notations:

$$Var[x] = E[x^2] - (E[x])^2$$

$$E[X^2] = Var(X) + (E[x])^2$$

$$Given\ y = f + \varepsilon\ and\ E[\varepsilon] = 0, E[y] = E[f + \varepsilon] = E[f] = f$$

$$Since\ Var[\varepsilon] = \sigma^2, Var[y] = E[(y - E[y])^2] = E[(y - f)^2]$$

$$= E[(f + \varepsilon - f)^2] = E[\varepsilon^2] = Var[\varepsilon] + (E[\varepsilon])^2 = \sigma^2$$

Bias-Variance Tradeoff

- The expected error on an unseen sample x can be decomposed as:

$$
\begin{aligned}
E[(y-\hat{f})^2] &= E[(f+\varepsilon-\hat{f})^2] \\
&= E[(f+\varepsilon-\hat{f}+E[\hat{f}]-E[\hat{f}])^2] \\
&= E\left[(f-E[\hat{f}])^2\right] + E[\varepsilon^2] + E\left[(E(\hat{f})-\hat{f})^2\right] \\
&+ 2E[(f-E[\hat{f}])\varepsilon] + 2E[\varepsilon(E(\hat{f})-\hat{f})] + 2E[(E(\hat{f})-\hat{f})(f-E[\hat{f}])] \\
&= (f-E(\hat{f}))^2 + E(\varepsilon^2) + E[(E[\hat{f}]-\hat{f})^2] \\
&+ 2(f-E[\hat{f}])E(\varepsilon) + 2E(\varepsilon)E(E[\hat{f}]-\hat{f}] + 2E[E[\hat{f}]-\hat{f}](f-E[\hat{f}])
\end{aligned}
$$

Bias-Variance Tradeoff

$$= (f - E[\hat{f}])^2 + E[\varepsilon^2] + E[(E[\hat{f}] - \hat{f})^2]$$
$$= (f - E[\hat{f}])^2 + Var[y] + Var[\hat{f}]$$
$$= Bias[\hat{f}]^2 + Var[y] + Var[\hat{f}]$$
$$= Bias[\hat{f}]^2 + \sigma^2 + Var[\hat{f}]$$

- Hence the derivation.

Avoiding Overfitting

- A variety of techniques to avoid overfitting:
  - Cross-validation
  - Regularization
  - Feature selection
  - Dimensionality reduction

Non-exhaustive Cross-validation

Exhaustive Cross-validation

Nested Cross-validation

- Popular way to tune parameters of an algorithm
- One version: k-fold cross validation with validation and test set
- Lets say parameter X needs tuning
  - Possible values 10, 20, 30, 40, 50

Nested Cross-validation

- k = 7 in our example
  - One set each picked as Test and Validation, (k-2) picked for training
- For the picked Test set
  - Perform k-fold cross validation on Train & Validation set [Here k = 6]
  - Compute the average training error for each value of X
  - Pick the best X
- Repeat for each possible Test set [i.e. 7 times]
- Pick X that was returned maximum times to outer loop

Regularization

Regularization

- Let $\hat{f}(x) = \theta_0 + \theta_1 x^2 + \theta_3 x^2 + \theta_4 x^3$
- We want to minimize the MSE:

$$\frac{1}{m} * \min_{\theta_0, \theta_1, \theta_3, \theta_4} \sum_{i=1}^{m} (\hat{f}_\theta(x^{(i)}) - y^{(i)})^2$$

- where m is the number of training samples, theta's are the weight parameters
- Let MSE be represented by $J(\theta)$
- Lets say we want to penalize the higher order terms (2 and 3)

Regularization

- Can add penalty terms say +1000 θ_3 + 1000 θ_4
- The effect of this would be that θ_3 and θ_4 need to be quite small to minimize error
- A significantly high penalty can actually convert a overfit problem to an underfit problem
  - Since all the terms with high regularization parameter would become 0 or close to 0
  - E.g. if all terms except θ_0 have a high enough regularization parameter then $\hat{f}(x)$ can become a constant !!!

Feature Selection

- Filter methods: …
- Wrapper methods: …
  - Recursive Feature Elimination
- Embedded methods: …

Dimensionality Reduction

Data Preprocessing

- A popular methodology in data mining is CRoss Industry Standard Process for data mining (CRISP DM)
- …

Feature Engineering

- …
- **One-hot-encoding or one-of-K:** Refers to splitting the column which contains numerical *categorical data* to many columns depending on the number of categories present in that column.
  - Each column contains “0” or “1” corresponding to which column it has been placed.

Feature Engineering

Fruit	Categorical value of fruit	Price
apple	1	5
mango	2	10
apple	1	15
orange	3	20

- After one hot encoding

apple	mango	orange	price
1	0	0	5
0	1	0	10
1	0	0	15
0	0	1	20

Feature Engineering

- …

Overview of Data Analytics: Data Mining & Warehousing

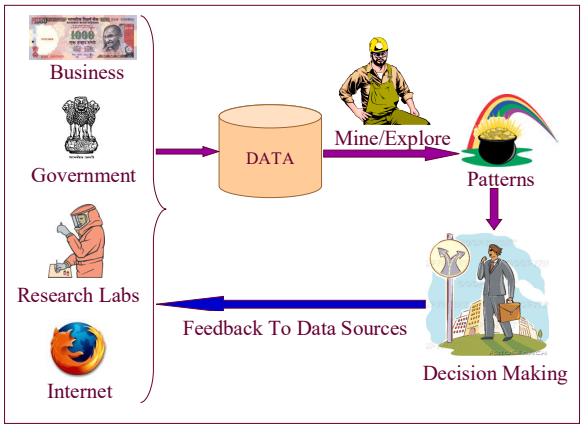
Vikram Pudi
vikram@iiit.ac.in
IIIT Hyderabad

Originated from DB community…

- Traditional Database Systems
  - Indexing
  - Query languages
  - Query optimization
  - Transaction processing
  - Recovery …
- XML, Semantic web
- OO and OR DBMS …
- *Data Mining*

Data Mining

Automated extraction of interesting patterns from large databases



Types of Patterns

- Associations
  - *Coffee* buyers usually also purchase *sugar*
- Clustering
  - Segments of customers requiring different promotion strategies
- Classification
  - Customers expected to be *loyal*

Association Rules

That which is infrequent is not worth worrying about.

Association Rules

D :

Transaction ID	Items
1	Tomato, Potato, Onions
2	Tomato, Potato, Brinjal, Pumpkin
3	Tomato, Potato, Onions, Chilly
4	Lemon, Tamarind

Rule: Tomato, Potato → Onion (confidence: 66%, support: 50%)

Support(X) = |transactions containing X| / |D|

Confidence(R) = support(R) / support(LHS(R))

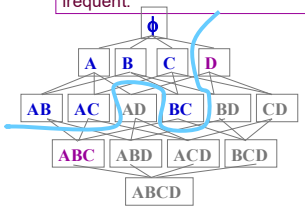
Problem proposed in [AIS 93]: Find all rules satisfying user given minimum support and minimum confidence.

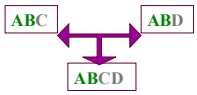
Association Rule Applications

- E-commerce
  - People who have bought *Sundara Kandam* have also bought *Srimad Bhagavatham*
- Census analysis
  - *Immigrants* are usually *male*
- Sports
  - A chess end-game configuration with "*white pawn on A7*" and "*white knight dominating black rook*" typically results in a "*win for white*".
- Medical diagnosis
  - Allergy to *latex rubber* usually co-occurs with allergies to *banana* and *tomato*

The Apriori Algorithm

Idea: An itemset can be frequent only if all its subsets are frequent.



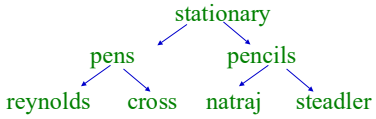



```
Apriori( DB, minsup ):
C = {all 1-itemsets}
        // candidates = singletons
while ( |C| > 0 ):
    make pass over DB, find counts of C
    F = sets in C with count ≥ minsup*|DB|
    output F
    C = AprioriGen(F) // gen. candidates
```

```
AprioriGen( F ):
for each pair of itemsets X, Y in F:
    if X and Y share all items, except last
        Z = X ∪ Y // generate candidate
        if any imm. subset of Z is not in F:
            prune Z // Z can't be frequent
```


Types of Association Rules

- Boolean association rules
- Hierarchical rules



reynolds → pencils

- Quantitative & Categorical rules
  - (Age: 30…39), (Married: Yes) → (NumCars: 2)

More Types of Association Rules

- Cyclic / Periodic rules
  - Sunday → vegetables
  - Christmas → gift items
  - Summer, rich, jobless → ticket to Hawaii
- Constrained rules
  - Show itemsets whose average price > Rs.10,000
  - Show itemsets that have television on RHS
- Sequential rules
  - Star wars, Empire Strikes Back → Return of the Jedi

Classification

To be or not to be: That is the question.

- William Shakespeare

The Classification Problem

Outlook	Temp (°F)	Humidity (%)	Windy?	Class
sunny	75	70	true	*play*
sunny	80	90	true	*don't play*
sunny	85	85	false	*don't play*
sunny	72	95	false	*don't play*
sunny	69	70	false	*play*
overcast	72	90	true	*play*
overcast	83	78	false	*play*
overcast	64	65	true	*play*
overcast	81	75	false	*play*
rain	71	80	true	*don't play*
rain	65	70	true	*don't play*
rain	75	80	false	*play*
rain	68	80	false	*play*
rain	70	96	false	*play*
sunny	77	69	true	?
rain	73	76	false	?

Play Outside?

Model relationship between class labels and attributes

e.g. outlook = overcast ⇒ class = *play*

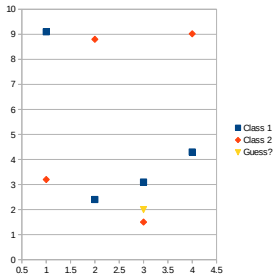
⇒ Assign class labels to new data with *unknown* labels

Applications

- Text classification
  - Classify emails into spam / non-spam
  - Classify web-pages into yahoo-type hierarchy
  - NLP Problems
    - Tagging: Classify words into verbs, nouns, etc.
- Risk management, Fraud detection, Computer intrusion detection
  - Given the properties of a transaction (items purchased, amount, location, customer profile, etc.)
  - Determine if it is a fraud
- Machine learning / pattern recognition applications
  - Vision
  - Speech recognition
  - etc.
- All of science & knowledge is about predicting future in terms of past
  - So classification is a very fundamental problem with ultra-wide scope of applications

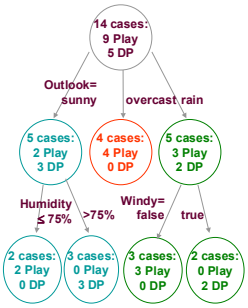
k-Nearest Neighbours

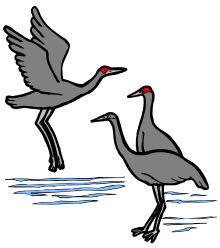
- Model = Training data
- Classify record R using the *k* nearest neighbours of R in the training data.
- Most frequent class among *k* NNs
- Distance function could be euclidean
- Use an index structure (e.g. R* tree) to find the *k* NNs efficiently



Decision Trees

Outlook	Temp (°F)	Humidity (%)	Windy?	Class
sunny	75	70	true	*play*
sunny	80	90	true	*don't play*
sunny	85	85	false	*don't play*
sunny	72	95	false	*don't play*
sunny	69	70	false	*play*
overcast	72	90	true	*play*
overcast	83	78	false	*play*
overcast	64	65	true	*play*
overcast	81	75	false	*play*
rain	71	80	true	*don't play*
rain	65	70	true	*don't play*
rain	75	80	false	*play*
rain	68	80	false	*play*
rain	70	96	false	*play*





Clustering

Birds of a feather flock together.

The Clustering Problem

Outlook	Temp (°F)	Humidity (%)	Windy?
sunny	75	70	true
sunny	80	90	true
sunny	85	85	false
sunny	72	95	false
sunny	69	70	false
overcast	72	90	true
overcast	73	88	true
overcast	64	65	true
overcast	81	75	false
rain	71	80	true
rain	65	70	true
rain	75	80	false
rain	68	80	false
rain	70	96	false

Find groups of similar records.

Need a function to compute similarity, given 2 input records

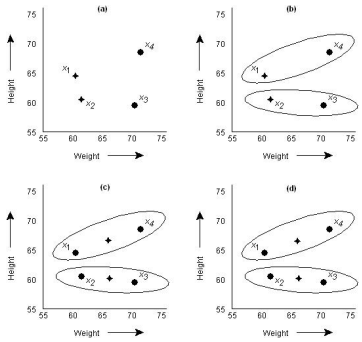
⇒ Unsupervised learning

Applications

- Targetting similar people or objects
  - Student tutorial groups
  - Hobby groups
  - Health support groups
  - Customer groups for marketing
  - Organizing e-mail
- Spatial clustering
  - Exam centres
  - Locations for a business chain
  - Planning a political strategy

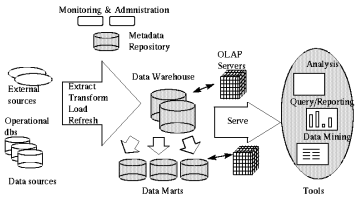
Partitioning technique: *k*-Means

- Initial *k* means = random records
- Iterate as long as clusters change:
  - Put each record X in the cluster to whose mean it is closest
  - Recompute means as the average of all points in each cluster



Data Warehousing

- Extract, transform, load data from multiple sources in an enterprise
- Provide unified view for top management
- OLAP server provides multi-dimensional view for manual exploration of patterns



Examples of OLAP

Comparisons (this period v.s. last period)

Show me the sales per store for this year and compare it to that of the previous year to identify discrepancies

Ranking and statistical profiles (top N/bottom N)

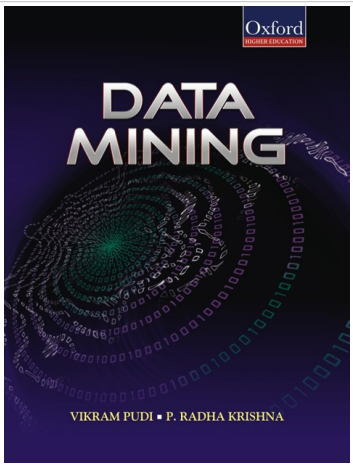
Show me sales, profit and average call volume per day for my 10 most profitable salespeople

Custom consolidation (market segments, ad hoc groups)

Show me an abbreviated income statement by quarter for the last four quarters for my northeast region operations

Take Home

- Data mining is a mature field
- Don't waste time developing new algorithms for core tasks
- Focus on applications to challenging kinds of data
  - Streams, Distributed data, Multimedia, Web, …
- Most effort is in how to map domain problems to data mining problems
- And how to make sense of the output.



Propositional Logic & Reasoning

Dr. Vikram Pudi

Knowledge Representation

- Expressing knowledge explicitly in a computer-tractable way
  - Knowledge Base: set of facts (or sentences) expressed in a (formal) language such as logic

2

Why is it important?

- Core of AI
- Possibility of *automating* reasoning
- Reasoning: draw inferences from knowledge
  - answer queries
  - discover facts that follow from the knowledge base
  - decide what to do
  - etc.

3

Logic in General

- Logics are formal languages for representing information such that conclusions can be drawn
- Syntax: Describes how to make sentences
- Semantics: How sentences relate to reality. The meaning of a sentence is not *intrinsic* to that sentence.
- Proof Theory: A set of rules for drawing conclusions (inferences, deductions).

4

Why *formal* languages?

- Natural languages exhibit ambiguity.
  - Examples:
    - The boy saw a girl with a telescope
    - Our shoes are guaranteed to give you a fit
  - Ambiguity makes reasoning difficult / incomplete
- Formal languages promote rigour and thereby reduce possibility of human error.
- Formal languages help reduce implicit / unstated assumptions by removing *familiarity* with subject matter
- Formal languages help achieve generality due to possibility of finding *alternative interpretations* for sentences and arguments.

5

Logical Arguments

- All humans have 2 eyes.
- Kishore is a human.
  - Therefore Kishore has 2 eyes.
- All humans have 4 eyes.
- Kishore is a human.
  - Therefore Kishore has 4 eyes.
- Both are (logically) valid arguments.
- Which statements are true / false ?

6

Logical Arguments (contd)

- All humans have 2 eyes.
- Kishore has 2 eyes.
  - Therefore Kishore is a human.
- No human has 4 eyes.
- Kishore has 2 eyes.
  - Therefore Kishore is not human.
- Both are (logically) invalid arguments.
- Which statements are true / false ?

7

From English to Propositional Formulae

- "it is not the case that the lectures are dull": $\neg D$ (alternatively "the lectures are not dull")
- "the lectures are dull and the text is readable": $D \wedge R$
- "either the lectures are dull or the text is readable": $D \vee R$
- "if the lectures are dull, then the text is not readable": $D \rightarrow R$
- "the lectures are dull if and only if (iif) the text is readable": $D \leftrightarrow R$
- "if the lectures are dull, then if the text is not readable, Kishore will not pass": $D \rightarrow (\neg R \rightarrow \neg P)$

8

Propositional Logic

- Use letters to stand for "basic" propositions
- Complex sentences use operators for not, and, or, implies, iff.
- Brackets ( ) for grouping
  $(P \rightarrow (Q \rightarrow (\neg(R)))$ vs. $P \rightarrow (Q \rightarrow \neg R)$
- Omitting brackets
  - precedence from highest to lowest is: $\neg$, $\wedge$,$\vee$,$\rightarrow$, $\leftrightarrow$
  - Binary operators are left associative
    (so $P \rightarrow Q \rightarrow R$ is $(P \rightarrow Q) \rightarrow R$)
- Questions:
  - Is $(P \vee Q) \vee R$ same as $P \vee (Q \vee R)$?
  - Is $(P \rightarrow Q) \rightarrow R$ same as $P \rightarrow (Q \rightarrow R)$?

9

Semantics (Truth Tables)

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \rightarrow Q$	$P \leftrightarrow Q$
True	True	False	True	True	True	True
True	False	False	False	True	False	False
False	True	True	False	True	True	False
False	False	True	False	False	True	True

- One row for each possible assignment of True/False to propositional variables
- **Important:** Above P and Q can be any sentence, including complex sentences

10

Terminology

- A sentence is valid if it is True under all possible assignments of True/False to its propositional variables (e.g. $P \vee \neg P$).
- Valid sentences are also referred to as tautologies
- A sentence is satisfiable if and only if there is *some* assignment of True/False to its propositional variables for which the sentence is True
- A sentence is unsatisfiable if and only if it is not satisfiable (e.g. $P \wedge \neg P$).

11

Semantics (Complex Sentences)

R	S	$\neg R$	$R \wedge S$	$\neg R \vee S$	$(R \wedge S) \rightarrow (\neg R \vee S)$
True	True	False	True	True	True
True	False	False	False	False	True
False	True	True	False	True	True
False	False	True	False	True	True

12

Material Implication

- The only time $P \rightarrow Q$ evaluates to False is when P is True and Q is False
- If $P \rightarrow Q$ is True, then:
  - P is a sufficient condition for Q
  - Q is a necessary condition for P

13

Exercises

Given: A and B are true; X and Y are false, determine truth values of:

- $\neg(A \vee X)$
- $A \vee (X \wedge Y)$
- $A \wedge (X \vee (B \wedge Y))$
- $[(A \wedge X) \vee \neg B] \wedge \neg[(A \wedge X) \vee \neg B]$
- $(P \wedge Q) \wedge (\neg A \vee X)$
- $[(X \wedge Y) \rightarrow A] \rightarrow [X \rightarrow (Y \rightarrow A)]$

14

Entailment

- $S \Rightarrow P$ — whenever all the formulae in the set S are True, P is True
- This is a *semantic* notion; it concerns the notion of *Truth*
- To determine if $S \Rightarrow P$ construct a truth table for S, P
  - $S \Rightarrow P$ if, in any row of the truth table where all formulae of S are true, P is also true
- A tautology is just the special case when S is the empty set (evaluates to False).

15

Entailment Example

P	$P \rightarrow Q$	Q
True	True	True
True	False	False
False	True	True
False	True	False

Modus Ponens

Therefore, $P, P \rightarrow Q \Rightarrow Q$

16

Exercises

Use truth tables to determine validity of:

- If it rains, Raju carries an umbrella. Raju is carrying an umbrella, therefore it will rain.
- If the weather is warm and the sky is clear, then either we go swimming or we go boating. It is not the case that if we do not go swimming, then the sky is not clear. Therefore, either the weather is warm or we go boating.

17

Formal Proofs

- Intend to formally capture the notion of proof that is commonly applied in other fields (e.g. mathematics).
- A proof of a formula from a set of premises is a sequence of steps in which any step of the proof is:
  1. An axiom or premise
  2. A formula deduced from previous steps of the proof using some rule of inference
- The last step of the proof should deduce the formula we wish to prove.
- We say that S follows from (premises) P to denote that the set of formulae P "prove" the formula S.

18

Soundness and Completeness

- A logic is sound if it preserves truth (i.e. if a set of premises are all true, any conclusion drawn from those premises *must* also be true).
- A logic is complete if it is capable of proving *any valid* consequence.
- A logic is decidable if there is a mechanical procedure (computer program) to prove *any* given consequence.

19

Inference Rules

1. Modus Ponens: $P, P \rightarrow Q \Rightarrow Q$
2. Modus Tollens: $P \rightarrow Q, \neg Q \Rightarrow \neg P$
3. Hypothetical Syllogism: $P \rightarrow Q, Q \rightarrow R \Rightarrow P \rightarrow R$
4. And-Elimination: $P_1 \wedge P_2 \wedge \ldots \wedge P_n \Rightarrow P_i$
5. And-Introduction: $P_1, P_2, \ldots, P_n \Rightarrow P_1 \wedge P_2 \wedge \ldots \wedge P_n$
6. Or-Introduction: $P_i \Rightarrow P_1 \vee P_2 \vee \ldots \vee P_n$
7. Double-Negation Elimination: $\neg\neg P \Rightarrow P$
8. Unit Resolution: $P \vee Q, \neg Q \Rightarrow P$
9. Resolution: $P \vee Q, \neg Q \vee R \Rightarrow P \vee R$

20

Example Formal Proof

1. $A \vee (B \rightarrow D)$
2. $\neg C \rightarrow (D \rightarrow E)$
3. $A \rightarrow C$
4. $\neg C \mathrel{/\!\!\therefore} B \rightarrow E$
5. $\neg A$ *3,4 (Modus Tollens)*
6. $B \rightarrow D$ *1,5 (Unit Resolution)*
7. $D \rightarrow E$ *2,4 (Modus Ponens)*
8. $B \rightarrow E$ *6,7 (Hypothetical Syllogism)*

21

Exercises

Construct formal proof of validity for:

- If the investigation continues, then new evidence is brought to light. If new evidence is brought to light, then several leading citizens are implicated. If several leading citizens are implicated, then the newspapers stop publicizing the case. If continuation of the investigation implies that the newspapers stop publicizing the case, then the bringing to light of new evidence implies that the investigation continues. The investigation does not continue. Therefore, new evidence is not brought to light.
- *C*: The investigation continues. *N*: New evidence is brought to light. *I*: Several leading citizens are implicated. *S*: The newspapers stop publicizing the case.

22

Solution

1. $C \rightarrow N$
2. $N \rightarrow I$
3. $I \rightarrow S$
4. $(C \rightarrow S) \rightarrow (N \rightarrow C)$
5. $\neg C \mathrel{/\!\!\therefore} \neg N$
6. $C \rightarrow I$ *1,2 (Hypothetical Syllogism)*
7. $C \rightarrow S$ *6,3 (Hypothetical Syllogism)*
8. $N \rightarrow C$ *7,4 (Modus Ponens)*
9. $\neg N$ *8,5 (Modus Tollens)*

23

Exercises (contd.)

- If I study, I make good grades. If I do not study, I enjoy myself. Therefore, either I make good grades or I enjoy myself.
- S, G, E

24

Solution

1. $S \rightarrow G$
2. $\neg S \rightarrow E / \therefore G \vee E$
3. $\neg S \vee G$ *1*
4. $\neg\neg S \vee E$ *2*
5. $S \vee E$ *4 (DoubleNegationEliminate)*
6. $G \vee E$ *3,5 (Resolution)*

25

Complete Proof Systems

- Truth Tables
- Inference Rules
  - 19 rules + Conditional Proof + Indirect Proof
  - *Method of Resolution*

26

Resolution

- Better suited to computer implementation
- Generalizes to first-order logic
- The basis of Prolog's inference method
- To apply resolution, all formulae in the knowledge base and the query must be in clausal form

27

Normal Forms

- A literal is a propositional letter or the negation of a propositional letter
- A clause is a disjunction of literals
- Conjunctive Normal Form (CNF) – a conjunction of clauses
  e.g. $(P \vee Q \vee \neg R) \wedge (\neg S \vee \neg R)$
- Disjunctive Normal Form (DNF) – a disjunction of conjunctions of literals
  e.g. $(P \wedge Q \wedge \neg R) \vee (\neg S \wedge \neg R)$
- Every propositional logic formula can be converted to CNF and DNF

28

Conversion to CNF

- Eliminate $\leftrightarrow$ rewriting $P \leftrightarrow Q$ as $(P \rightarrow Q) \wedge (Q \rightarrow P)$
- Eliminate $\rightarrow$ rewriting $P \rightarrow Q$ as $\neg P \vee Q$
- Use *De Morgan's laws* to push $\neg$ inwards:
  - Rewrite $\neg(P \wedge Q)$ as $\neg P \vee \neg Q$
  - Rewrite $\neg(P \vee Q)$ as $\neg P \wedge \neg Q$
- Eliminate double negations: rewrite $\neg\neg P$ as P
- Use the *distributive laws* to get CNF:
  - Rewrite $(P \wedge Q) \vee R$ as $(P \vee R) \wedge (Q \vee R)$
  - Rewrite $(P \vee Q) \wedge R$ as $(P \wedge R) \vee (Q \wedge R)$
- Exercise: Convert $\neg(P \rightarrow (Q \wedge R))$ to CNF

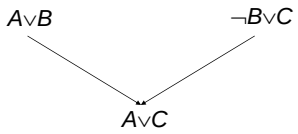
29

Solution

- $\neg(P \rightarrow (Q \wedge R))$
- $\neg(\neg P \vee (Q \wedge R))$
- $\neg\neg P \wedge \neg(Q \wedge R)$
- $\neg\neg P \wedge (\neg Q \vee \neg R)$
- $P \wedge (\neg Q \vee \neg R)$
- Two clauses: P, $\neg Q \vee \neg R$

30

Resolution Rule of Inference



- where *B* is a propositional letter and *A* and *C* are clauses (possibly empty).
- $A \vee C$ is the resolvent of the two clauses.

31

Applying Resolution

- How can we use the resolution rule? One way:
  - Convert knowledge base into clausal form
  - Repeatedly apply resolution rule to the resulting clauses
  - A query *A* follows from the knowledge base if and only if each of the clauses in the CNF of *A* can be derived using resolution
- There is a better way . . .

32

Refutation Systems

- To show that *P* follows from *S* (i.e. $S \Rightarrow P$) using refutation, start with *S* and $\neg P$ in clausal form and derive a contradiction using resolution.
- The "empty clause □" (a clause with no literals) is unsatisfiable (always False) – a *contradiction*.
- So if the empty clause is derived using resolution, the original set of clauses is unsatisfiable.
- That is, if we can derive □ from the clausal forms of *S* and $\neg P$, these clauses can never be all True together.
- Hence whenever the clauses of *S* are all True, at least one clause from $\neg P$ must be False, i.e. $\neg P$ must be False and *P* must be True.
- Hence, by definition, $S \Rightarrow P$

33

Applying Resolution Refutation

- Negate query to be proven.
- Convert knowledge base and negated conclusion into CNF and extract clauses.
- Repeatedly apply resolution until either the empty clause (contradiction) is derived or no more clauses can be derived.
- If the empty clause is derived, answer 'yes' (query follows from knowledge base), otherwise answer 'no' (query does not follow from knowledge base)

34

Resolution: Example 1

$(G \vee H) \rightarrow (\neg J \wedge \neg K)$, $G \Rightarrow \neg J$
Clausal form of $(G \vee H) \rightarrow (\neg J \wedge \neg K)$ is
$\{\neg G \vee \neg J, \neg H \vee \neg J, \neg G \vee \neg K, \neg H \vee \neg K\}$

1. $\neg G \vee \neg J$ [Premise]
2. $\neg H \vee \neg J$ [Premise]
3. $\neg G \vee \neg K$ [Premise]
4. $\neg H \vee \neg K$ [Premise]
5. G [Premise]
6. J [¬Conclusion]
7. $\neg G$ [1,6. Resolution]
8. □ [5,7. Resolution]

35

Problems

- $P \rightarrow \neg Q, \neg Q \rightarrow R \Rightarrow P \rightarrow R$
- $\Rightarrow ((P \vee Q) \wedge \neg P) \rightarrow Q$

36

Soundness and Completeness

- Resolution refutation is sound, i.e. it preserves truth (if a set of premises are all true, any conclusion drawn from those premises **will** also be true**).**
- Resolution refutation is complete, i.e. it is capable of proving all *valid* consequences of any knowledge base.
- Resolution refutation is decidable, i.e. there is an algorithm implementing resolution, which when asked whether $P \Rightarrow S$, can always answer 'yes' or 'no' (correctly).

37

Heuristics in Applying Resolution

- Clause elimination — can disregard certain types of clauses
  - Pure clauses: contain literal L where $\neg L$ doesn't appear elsewhere
  - Tautologies: clauses containing both L and $\neg L$
  - Subsumed clauses: another clause exists containing a subset of the literals
- Ordering strategies
  - Resolve unit clauses (only one literal) first
  - Start with query clauses
  - Aim to shorten clauses

38

Conclusion

- We have now investigated one knowledge representation and reasoning formalism
- This means we can draw new conclusions from the knowledge we have: we can reason
- Have enough to build a knowledge-based agent
- However, propositional logic is a weak language; there are many things that cannot be expressed
- To express knowledge about objects, their properties and the relationships that exist between objects, we need a more expressive language: first-order logic

39

Classification

Vikram Pudi
vikram@iiit.ac.in
IIIT Hyderabad

Talk Outline

- Introduction
  - Classification Problem
  - Applications
  - Metrics
  - Combining classifiers
- Classification Techniques

The Classification Problem

Outlook	Temp (°F)	Humidity (%)	Windy?	Class
sunny	75	70	true	*play*
sunny	80	90	true	*don't play*
sunny	85	85	false	*don't play*
sunny	72	95	false	*don't play*
sunny	69	70	false	*play*
overcast	72	90	true	*play*
overcast	83	78	false	*play*
overcast	64	65	true	*play*
overcast	81	75	false	*play*
rain	71	80	true	*don't play*
rain	65	70	true	*don't play*
rain	75	80	false	*play*
rain	68	80	false	*play*
rain	70	96	false	*play*
sunny	77	69	true	?
rain	73	76	false	?

Play Outside?

Model relationship between class labels and attributes

e.g. outlook = overcast ⇒ class = *play*

⇒ Assign class labels to new data with *unknown* labels

Applications

- Text classification
  - Classify emails into spam / non-spam
  - Classify web-pages into yahoo-type hierarchy
  - NLP Problems
    - Tagging: Classify words into verbs, nouns, etc.
- Risk management, Fraud detection, Computer intrusion detection
  - Given the properties of a transaction (items purchased, amount, location, customer profile, etc.)
  - Determine if it is a fraud
- Machine learning / pattern recognition applications
  - Vision
  - Speech recognition
  - etc.
- All of science & knowledge is about predicting future in terms of past
  - So classification is a very fundamental problem with ultra-wide scope of applications

Metrics

1. accuracy
2. classification time per new record
3. training time
4. main memory usage (during classification)
5. model size

Accuracy Measure

- Prediction is just like tossing a coin (random variable X)
  - "Head" is "success" in classification; X = 1
  - "tail" is "error"; X = 0
  - X is actually a mapping: {"success": 1, "error" : 0}
- In statistics, a succession of independent events like this is called a *bernoulli process*
  - Accuracy = P(X = 1) = p
  - mean value = μ = E[X] = $p \times 1 + (1-p) \times 0 = p$
  - variance = σ^2 = E[$(X-\mu)^2$] = $p\,(1-p)$
- Confidence intervals: Instead of saying accuracy = 85%, we want to say: accuracy $\in$ [83, 87] with a confidence of 95%

Binomial Distribution

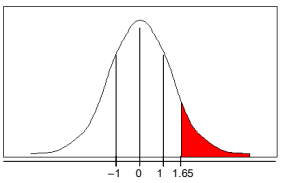
- Treat each classified record as a bernoulli trial
- If there are *n* records, there are *n* independent and identically distributed (iid) bernoulli trials, X_i, i = 1,…,*n*
- Then, the random variable $X = \sum_{i=1,\ldots,n} X_i$ is said to follow a *binomial distribution*
  - $P(X = k) = {}^nC_k\, p^k\, (1-p)^{n-k}$
- Problem: Difficult to compute for large *n*

Normal Distribution

- Continuous distribution with parameters μ (mean), σ^2(variance)
- Probability density:
  $f(x) = (1/\sqrt{(2\pi\sigma^2)})\, \exp(-(x-\mu)^2 / (2\sigma^2))$
- Central limit theorem:
  - Under certain conditions, the distribution of the sum of a *large number* of iid random variables is approximately normal
  - A *binomial distribution* with parameters *n* and *p* is approximately normal for large *n* and *p* not too close to 1 or 0
  - The approximating normal distribution has mean μ = *np* and standard deviation $\sigma^2 = (n\, p\, (1 - p))$

Confidence Intervals

Normal distribution with mean = 0 and variance = 1



Pr[$X \geq z$]	z
0.1%	3.09
0.5%	2.58
1%	2.33
5%	1.65
10%	1.28
20%	0.84
40%	0.25

- E.g. $P[-1.65 \leq X \leq 1.65] = 1 - 2 \times P[X \geq 1.65] = 90\%$
- To use this we have to transform our random variable to have mean = 0 and variance = 1
- Subtract mean from *X* and divide by standard deviation

Estimating Accuracy

- Holdout method
  - Randomly partition data: training set + test set
  - accuracy = |correctly classified points| / |test data points|
- Stratification
  - Ensure each class has approximately equal proportions in both partitions
- Random subsampling
  - Repeat holdout *k* times. Output average accuracy.
- *k*-fold cross-validation
  - Randomly partition data: $S_1, S_2, \ldots, S_k$
  - First, keep S_1 as test set, remaining as training set
  - Next, keep S_2 as test set, remaining as training set, etc.
  - accuracy = |*total* correctly classified points| / |total data points|
- Recommendation:
  - Stratified 10-fold cross-validation. If possible, repeat 10 times and average results. (reduces variance)

Is Accuracy Enough?

- If only 1% population has cancer, then a test for cancer that classifies *all* people as *non-cancer* will have 99% accuracy.
- Instead output a confusion matrix:

Actual/ Estimate	Class 1	Class 2	Class 3
Class 1	90%	5%	5%
Class 2	2%	91%	7%
Class 3	8%	3%	89%

Combining Classifiers

- Get *k* random samples with replacement as training sets (like in random subsampling).

⇒ We get *k* classifiers

- Bagging: Take a *majority vote* for the best class for each new record
- Boosting: Each classifier's vote has a *weight* proportional to its accuracy on training data

⇒ Like a patient taking multiple opinions from several doctors

Talk Outline

- Introduction
- Classification Techniques
  1. Nearest Neighbour Methods
  2. Decision Trees
     - ID3, CART, C4.5, C5.0, SLIQ, SPRINT
  3. Bayesian Methods
     - Naïve Bayes, Bayesian Belief Networks
     - Maximum Entropy Based Approaches
  4. Association Rule Based Approaches
  5. Soft-computing Methods:
     - Genetic Algorithms, Rough Sets, Fuzzy Sets, Neural Networks
  6. Support Vector Machines
  7. Convolutional Neural Networks, Deep Learning

Nearest Neighbour Methods

k-NN, Reverse Nearest Neighbours

k-Nearest Neighbours

- Model = Training data
- Classify record R using the *k* nearest neighbours of R in the training data.
- Most frequent class among *k* NNs
- Distance function could be euclidean
- Use an index structure (e.g. R* tree) to find the *k* NNs efficiently

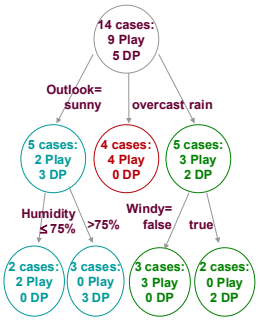
Reverse Nearest Neighbours

- Records which consider *R* as a *k*-NN
- Output most frequent class among RNNs.
- More resilient to outliers.

Decision Trees

Decision Trees

Outlook	Temp (°F)	Humidity (%)	Windy?	Class
sunny	75	70	true	*play*
sunny	80	90	true	*don't play*
sunny	85	85	false	*don't play*
sunny	72	95	false	*don't play*
sunny	69	70	false	*play*
overcast	72	90	true	*play*
overcast	83	78	false	*play*
overcast	64	65	true	*play*
overcast	81	75	false	*play*
rain	71	80	true	*don't play*
rain	65	70	true	*don't play*
rain	75	80	false	*play*
rain	68	80	false	*play*
rain	70	96	false	*play*



Basic Tree Building Algorithm

MakeTree (Training Data D):
    Partition(D)

Partition (Data D):
    if all points in D are in same class: return
    Evaluate splits for each attribute A
    Use best split found to partition D into $D_1, D_2, \ldots, D_n$
    for each D_i:
        Partition (D_i)

ID3, CART

ID3
- Use *information gain* to determine best split
- *gain* = $H(D) - \sum_{i=1\ldots n} P(D_i)\, H(D_i)$
- $H(p_1, p_2, \ldots, p_m) = -\sum_{i=1\ldots m} p_i \log p_i$
- like 20-question game
  - Which attribute is better to look for first: "Is it a living thing?" or "Is it a duster?"

CART
- Only create two children for each node
- Goodness of a split (Φ)
  $\Phi = 2\, P(D_1)\, P(D_2) \sum_{i=1\ldots m} | P(C_j / D_1) - P(C_j / D_2) |$

Shannon's Entropy

- An expt has several possible outcomes
- In N expts, suppose each outcome occurs M times
- This means there are N/M possible outcomes
- To represent each outcome, we need log N/M bits.
  - This generalizes even when all outcomes are not equally frequent.
  - Reason: For an outcome j that occurs M times, there are N/M equi-probable events among which only one cp to j
- Since p_i = M / N, information content of an outcome is -log p_i
- So, expected info content: $H = -\Sigma\, p_i \log p_i$

Bayesian Methods

Naïve Bayes

- New data point to classify: $X=(x_1, x_2, \ldots x_m)$
- Strategy:
  - Calculate $P(C_i/X)$ for each class C_i.
  - Select C_i for which $P(C_i/X)$ is maximum

$$P(C_i/X) = P(X/C_i)\, P(C_i) / P(X)$$
$$\propto P(X/C_i)\, P(C_i)$$
$$\propto P(x_1/C_i)\, P(x_2/C_i) \ldots P(x_m/C_i)\, P(C_i)$$

- Naïvely assumes that each x_i is independent
- We represent $P(X/C_i)$ by $P(X)$, etc. when unambiguous