1 Prerequisites

You should have a basic understanding of probability theory, combinatorics and formal logic – propositional and first order. Here are a few questions you should be able answer quickly and without much thinking.

- What is a possible computer representation for a propositional interpretation (a.k.a. truth valuation)?
- What are the conjunctive and disjunctive normal forms?
- If I is an interpretation and ϕ is a formula, what does $I \vDash \phi$ mean?
- Prove that conjunction is associative in propositional logic.
- Rewrite the formula $(a \wedge b) \implies c$ to an equivalent clausal form.
- What does it mean when we say we take samples i.i.d.?

2 Exercises

Motivation The homework that will follow is essentially a computer implementation of the following exercises. If you manage to solve them, the homework should be easy for you.

Definitions A monotone conjunction (resp. disjunction) is a conjunction (resp. disjunction) of a number of propositional variables. In other words, it's a term (resp. caluse) with positive literals only. An s-clause is a clause containing at most s literals. An s-CNF is a conjunction of s-clauses.

Exercise - combinatorics Assume a propositional logic with n variables. Compute the following combinatoric problems:

- What is the number of monotone conjunctions? (No duplicate literals)
- What is the number of non-equivalent conjunctions?
- What is the number of s-CNFs? (No duplicate clauses) Break down the calculation to the following steps:
  - What is the number of caluses of length exactly s?
  - What is the number of s-caluses?
  - Apply an earlier result.

Solution

- What is the number of monotone conjunctions? (No duplicate literals) *Answer*: Each literal is either present or missing. So it's the same as asking the number of subsets of a set of size n, i.e. 2^n (including the empty conjunction, i.e. tautology).

- What is the number of non-equivalent conjunctions? *Answer*: Non-equivalent means there are no duplicate literals and we disregard conjunctions with both negative and positive literal of the same variable, i.e. contradictions. Therefore, each literal is either positive, negative or missing, making the number of conjunctions $3^n + 1$ including both contradiction and tautology.

- What is the number of s-CNFs? (No duplicate clauses) Break down the calculation to the following steps:

  - What is the number of clauses of length exactly s? *Answer*: The number of variable combinations usable in the clause is $\binom{n}{s}$. For each of those combinations, a subset of the variables is negated, making the number of clauses: $2^s\binom{n}{s}$.
  - What is the number of s-caluses? *Answer*: We sum the above over the possible values of s, that is: $\sum_{i=0}^{s} 2^i\binom{n}{i}$. Summing from $i = 1$ would not be considered a mistake as the empty clause is somewhat redundant here.
  - Apply an earlier result. *Answer*: The result to apply is the number of monotone conjunctions, where instead of n we substitute the number of s-clauses calculated above.

Theoretical exercise - generalizing algorithm Study the generalizing algorithm. Find answers to the following:

- The basic algorithm learns monotone conjunctive concepts from a set of propositional interpretations. How can you reduce the learning of non-monotone conjunctive concepts (i.e. terms) to the simpler monotone case? *Answer*: We add new auxiliary variables for the negated literals and precompute their values simply as the negations of their respective input values.

- Using a similar idea, how could you reduce the learning of $s-$CNFs to the learning of monotone conjunctions?

  - Hint 1: Terms are actually 1-CNFs.
  - Hint 2: This is similar to the technique of polynomial expansion of features used in classical machine learning.

  Answer: Again, we introduce a new variable for each possible s-clause and use those to replace the original values (note that the clauses include single literls as well). We then compute their values from the input and pass it to the original algorithm. E.g. for a clause $a \vee \neg c$, there will be a corresponding auxiliary variable, say p_k, and if the input was $a = 0, b = 1, c = 0$ we pass $p_k = 1$ to the monotone conjunction learning algorithm. (Along with the other values, of course.)

- Using De Morgan's laws, how can you alter the algorithm to learn monotone disjunctions instead? *Answer*: See lecture slide titled "Generalizing Agent for Disjunctions".

Exercise - generalizing algorithm example Imagine a zoologist provides you with a dataset of animals and some of their observed features. Each animal is labeled whether it is or is not a mammal. You are to learn a mammal concept from the features, using the generalization algorithm. Emulate the generalization algorithm. First learn a conjunctive concept (non-monotone). Then learn a disjunctive concepts. Answers are on the next page.

	Bat	Dolphin	Earthworm	Bee	Carp	T-Rex	Penguin	Parrot	Goat	Platypus	Elephant	Frog	Hippo
Flies	✓			✓				✓					
Hair	✓								✓	✓			
Fins		✓			✓					✓		✓	
Feathers							✓	✓					
Scales					✓	✓							
Breathes air	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓
Bones	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓
Funky nose		✓									✓		
Mammal	✓	✓							✓	✓	✓		✓

Solution We are learning non-monotone conjunctions, so our initial hypothesis will be:

$$h_0 = \text{Flies} \wedge \neg\text{Flies} \wedge \text{Hair} \wedge \neg\text{Hair} \wedge \cdots \wedge \text{nose} \wedge \neg\text{nose}$$

We observe the sample Bat. Substituting it's values to h_0 would give us false, which is different from the label in the Mammal row, so we need to update our hypothesis by deleting inconsistent literals, i.e.:

$$h_1 = \text{Flies} \wedge \text{Hair} \wedge \neg\text{Fins} \wedge \neg\text{Feathers} \wedge \neg\text{Scales} \wedge \text{air} \wedge \text{Bones} \wedge \neg\text{nose}$$

Next is the sample Dolphin, again, it would be misclassified by h_1, so we delete inconsistent literals to update the hypothesis:

$$h_2 = \neg\text{Feathers} \wedge \neg\text{Scales} \wedge \text{air} \wedge \text{Bones}$$

You could continue in a similar manner, but no further literals would be deleted, so $h_2 = C_1$. Verify for yourself. Notice, that on non-mammals the algorithm either classifies the correctly (e.g. Earthworm does not have bones, classified as non-mammal by h_2) or no literals would be deleted (e.g. the Frog sample). That is a general property of this algorithm, which allows us to only consider the positive samples.

To learn the disjunctive concept, we would reduce it to the learning of conjunctive concepts, so h_0 would be the same as above. Furthermore, we need to negate Mammal label. So the first sample to be considered would be the Eartworm making

$$h_1 = \neg\text{Flies} \land \neg\text{Hair} \land \neg\text{Fins} \land \neg\text{Feathers} \land \neg\text{Scales} \land \text{air} \land \neg\text{Bones} \land \neg\text{nose}$$

After deleting inconsistent literals for Bee, we have:

$$h_2 = \neg\text{Hair} \land \neg\text{Fins} \land \neg\text{Feathers} \land \neg\text{Scales} \land \text{air} \land \neg\text{Bones} \land \neg\text{nose}$$

After deleting inconsistent literals for Carp, we have:

$$h_3 = \neg\text{Hair} \land \neg\text{Feathers} \land \neg\text{nose}$$

T-Rex would be correctly classified. After deleting inconsistent ¬Feathers literal for Penguin, we have:

$$h_4 = \neg\text{Hair} \land \neg\text{nose}$$

Verify, that no further literals would be deleted. To obtain the disjunction, we negate the final hypothesis:

$$\neg h_4 \equiv \text{Hair} \lor \text{nose} = C_2$$

Exercise Explain the following:

- Only one of the following is true. Determine which one and find a counterexample to the other.
  - Mammal $\models C_1$
  - $C_1 \models$ Mammal
- Only one of the following is true. Determine which one and find a counterexample to the other.
  - Mammal $\models C_2$
  - $C_2 \models$ Mammal

Answer Note: Due to it's use it in the semantic consequence relation, Mammal is here an unknown logical formula.

The counterexample for the first question would be the Frog, which would be misclassified by C_1 (i.e. Frog $\not\models$ Mammal). Thus $\mathrm{C}_1 \not\models$ Mammal, which reads that there is a model of C_1 (the Frog) which is not a model of Mammal (frog is not a Mammal). Mammal $\models \mathrm{C}_1$ holds.

For disjunction, the counterexample would be the Hippo. The relation would go the other direction, but the reasoning would be analogical. (Hippo $\models$ Mammal, but is misclassified by C_2).