CSCE 4613 – Artificial Intelligence
Fall 2019
Assignment 2

Carson Molder
cbmolder@uark.edu
010755069

Tyler Tracy
tgtracy@uark.edu
010805685

1 Clause Logic

1.1 What is an atom?

An *atom* is any symbol starting with a lowercase letter.

Example If we had an atom for Tyler, it could possibly be written as `tyler`. Likewise, an atom for Carson could be written as `carson`.

1.2 What is a body?

A *body* is an atom, or of the form $b_1 \land b_2$ where b_1 and b_2 are bodies.

Example Using the concepts in the atom definition, two bodies would be `tyler` and `carson`. Another valid body would be `tyler` $\land$ `carson`.

1.3 What is a definite clause?

A *definite clause* is an atom, or a rule that fits the form $h \leftarrow b$ where h is an atom and b is a body. It can be interpreted as "h if b".

Example If we have an atom `study_group`, a valid definite clause would be `study_group` $\leftarrow$ `tyler` $\land$ `carson`. Like the past two definitions, `tyler`, `carson`, and `study_group` also serve as definite clauses on their own.

1.4 What is a knowledge base?

A *knowledge base* is a set of definite clauses.

Example Using the past examples, a valid knowledge base would be $\{$`tyler`, `carson`, `study_group`, `study_group` $\leftarrow$ `tyler` $\land$ `carson`$\}$.

1.5 What is an interpretation of a knowledge base KB?

An *interpretation* of a knowledge base is an assignment of a truth value to each atom within the knowledge base.

Example Assume `tyler` and `carson` are "true" statements. Therefore, the body `study_group` $\leftarrow$ `tyler` $\land$ `carson`$\}$ is also true since `tyler` and `carson` are true and the logical AND of two true statements is also true.

1.6 What is a model of a knowledge base KB?

A *model* of a knowledge base is an interpretation of said knowledge base where *every* clause in that knowledge base is true.

Example For the knowledge base given in the knowledge base definition and the interpretation given in the interpretation definition, said interpretation is a model of the knowledge base since all it makes all of its constituent clauses true. However, if one of `tyler` or `carson` were false, then that interpretation would no longer be a model of that knowledge base.

2 Knowledge base

2.1 What is $KB \models g$?

$KB \models g$ means that KB logically entails g. This means that if everything in KB is true, then g must also be true. Equivalently, this can be stated as there is no interpretation in which KB is true and g is false.

2.2 What is $KB \vdash_P g$?

$KB \vdash_P g$ means that the proof procedure P can be used to derive the conjunction of atoms g from the knowledge base KB.

2.3 What is proof soundness?

A proof procedure P is *sound* if $KB \vdash_P g$ implies $KB \models g$. Equivalently, this means that if g can be derived from the proof procedure, then it must be true in all models of KB. A sound proof only derives logically entailed bodies and/or clauses. A non-sound proof would include bodies and/or clauses that are not true.

2.4 What is proof completeness?

A proof procedure P is *complete* if $KB \models g$ implies $KB \vdash_P g$. Equivalently, this means that if g is true in all models of KB, then it must be derived from the proof procedure. A complete proof derives every body and/or clause that is logically entailed by a knowledge base. An incomplete proof would fail to include some of these logically entailed bodies and/or clauses.

2.5 What is the key idea of the bottom-up proof procedure?

The key idea of the bottom-up proof procedure is that it derives a sound and complete set of proven clauses C from the bodies and clauses that make up the knowledge base KB.

Algorithm 1 Bottom-up proof procedure

> $C = \{\}$
> **repeat**
>   **select** clause $h \leftarrow b_1 \wedge b_2 \wedge \ldots \wedge b_m \in KB$
>   such that $b_i \in C\ \forall i$, and $h \notin C$
> $C = C \bigcup \{h\}$
> **until** no more clauses can be selected

2.6 When is the bottom-up proof procedure successful?

The bottom-up proof procedure is done when there are no more sentences that are logically provable from the set of atoms that you have proved logically follow from the knowledge base. If there is at least one logically provable sentence within the final set, then the bottom-up proof procedure was successful.

2.7 How can the bottom-up proof procedure show that there is no successful derivation?

When the bottom-up proof procedure is done, if the final set is empty, it shows that that nothing logically follows from all of the sentences and thus there is no successful derivation.

2.8 What is the key idea of the top-down proof procedure?

The key idea of the top-down proof procedure is that given a query g, the procedure searches backwards through the knowledge base KB to determine if g can be logically derived from KB.

Algorithm 2 Top-down proof procedure

> $query = yes \leftarrow body$
> $body = q_1 \wedge q_2 \wedge \ldots \wedge q_k$
> **repeat**
>   **select** $q_i \in body$
>   **choose** clause $c = q_i \leftarrow b_c \in KB$
>   **replace** $q_i \in body$ with b_c
> **until** $query$ is an answer (fails if there is no clause with q_i as head)

The term **select** means any clause will work (non-deterministic). The term **choose** means a correct clause must be chosen (deterministic).

2.9 When is the top-down proof procedure successful?

A top down proof is considered successful when the first answer clause that is produced is the query, for each answer clause the clause before it proves itself, and the last answer clause is just always true.

2.10 Give an example of an admissible heuristic for top-down search.

One admissible heuristic would be the number of atoms in the query. It is admissible because it will never overestimate the required amount of steps, since the number of atoms in the clause is equal to the minimum amount of steps required to prove the query.

3 Predicate logic

A "Someone in Fayetteville likes basketball."

$$\exists p(LivesIn(p, Fayetteville) \wedge LikesBasketball(p))$$

Domain People: $p \in P$

B "Everyone who likes ham also likes cheese."

$$\forall p(Likes(p, ham) \implies Likes(p, cheese))$$

Domain People: $p \in P$

C "Mary is tall and Bill is not."

$$IsTall(Mary) \wedge \neg IsTall(Bill)$$

Domain $\{Mary, Bill\}$

D "All of the houses near Sue's house are either large or old (but not both)."

$$\forall h(NearSue's(h) \implies IsLarge(h) \oplus IsOld(h))$$

Domain Houses: $h \in H$

E "Between any pair of real numbers there is at least one real number."

$$\forall x \forall y \exists z(x > z > y)$$

Domain Real numbers: $x, y, z \in \mathbb{R}$

F "There exists a dog that is cuter than all other dogs."

$$\forall d \exists c Cuter(c, d)$$

Domain Dogs: $c, d \in D$

4 Problem formulation

A Volume of a Cylinder

Find the dimensions (radius, height) of a cylinder with given surface area A that has the largest volume V.

Optimization variable V: The volume of the cylinder

Feasible region

$$F = \{r, h \in \mathbb{R} : 2\pi r^2 + 2\pi rh = A\}, F \subset \mathbb{R}^2$$

where the problem is constrained by a constant value A, the surface area of the cylinder.

Optimization objective

$$f(r, h) = V = \pi r^2 h$$

where V is the volume of the cylinder given radius r and height h.

Optimization formulation

$$\max_{V} f(r,h); r,h \in F$$

Convex optimization? This is a convex optimization problem because the optimization objective function is a convex function and the feasible region is a convex region.

B Minimum average Euclidean distance

Given a set of points $S = \{x_1, x_2, ..., x_m\}$ where $x_m \in R^n$, find a point in $\mathbb{R}^n$ that has the minimum average Euclidean distance to all points in S.

Optimization variable p: Point in n-space (represented as a n-dimensional vector)

Feasible region $F = \mathbb{R}^n$ (unconstrained)

Optimization objective

$$AvgDist(p) = \frac{1}{||S||} \sum_{x_i \in S} ||p - x_i||$$

Optimization formulation

$$\min_{p} AvgDist(p); p \in \mathbb{R}^n$$

Convex optimization? This is not a convex optimization problem because the optimization objective function is not convex. Depending how the points are arranged, the function could not produce convex outputs.

5 Convex function

A Product of convex functions

Let $f(x) : R^n \to R$, $g(x) : R^n \to R$ be convex functions. $h(x) = f(x)g(x)$.

Claim $h(x)$ is convex.

Counterexample $f(x) = x; g(x) = -x$

These are both convex functions. But their product, $-x^2$, is not convex. Therefore, the claim is false.

B Composition of convex functions

Let $f(x) : R^n \to R$, $g(x) : R \to R$ be convex functions. $h(x) = f(g(x))$.

Claim $h(x)$ is convex if $f(x)$ is non-decreasing.

Proof Take the first derivative of $h(x)$ using the chain rule.

$$h'(x) = f'(g(x))g'(x)$$

Take the second derivative of $h(x)$ by deriving $h'(x)$.

$$h''(x) = f''(g(x))g'(x)^2 + f'(g(x))g''(x)$$

Since it is assumed that $f(x)$ is non-decreasing, then $f'(x)$ is always a non-negative function. Furthermore, since $f(x)$ is convex, then its second derivative $f''(x)$ will also be non-negative since a positive or zero-valued second derivative indicates a concave-up (convex) function. Likewise, $g''(x)$ will be non-negative.

Therefore, we know the signs following functions:

$$f'(x) \geq 0; f''(x) \geq 0; g''(x) \geq 0$$

So, $h''(x)$ is the sum of the following:

- non-negative function times a non-negative function ($(g'(x)^2$ will always be non-negative since it is squared)
- non-negative function times a non-negative function

The product of two non-negative functions is non-negative. Also, the sum of two non-negative functions is non-negative. Therefore, $h''(x)$ is non-negative, indicating that its related function $h(x)$ is concave-up (convex). This proves that $h(x)$ is convex, and that the claim is true.

C Convex envelope (bonus)

The convex envelope of a function f (not necessarily convex), denoted conv f, is defined as the largest convex function majorized by f. Prove that conv $sin(x) = -1$.

Proof The domain of $sin(x)$ is $(-\infty, +\infty)$, so the convex envelope of the function must have the same domain. Meanwhile, the minima (local and global) of $sin(x)$ are $y = -1$ at odd integer multiples of $\frac{\pi}{2}$ (i.e. $\frac{-\pi}{2}$, $\frac{\pi}{2}$, $\frac{3\pi}{2}$, etc.)

Any convex function can never switch from increasing to non-increasing as x increases, for that would intuitively make it no longer convex. Therefore, once a convex function switches from non-increasing to increasing, it must remain increasing for all points to the right of the location of the switch. Likewise, in the other direction, once a convex function switches from decreasing to non-decreasing, it must remain non-decreasing way for all points to the right of the location of the switch.

With that consideration, consider the set of convex functions that do not cross the function $y = sin(x)$ from below (i.e. $sin(x)$ majorizes the function). Linear functions that have a non-zero slope must cross $sin(x)$ at some point. Likewise, convex parabolas, even-order polynomials, and exponential functions also cross $sin(x)$ at some point (or lay entirely above the function). Therefore, these kinds of functions cannot be the convex envelope of $sin(x)$.

Intuitively, any convex function that ever increases *must* cross $sin(x)$ at some point or lay entirely above it. Therefore, this narrows the possible class of convex functions to constants that lay on or below the global minima of $sin(x)$, -1 (i.e. $y = -10$, $y = -5$, etc.).

Of this family of functions, the largest would be $y = -1$. It never crosses $sin(x)$, only touching it at its minimum values. It also has the greatest y value for all values of x compared to smaller constant functions like $y = -2$. Therefore, by the process of elimination, conv $sin(x) = -1$.