Exercise 11 - Solution

Combining Logic and Deep Learning

Reliable and Interpretable Artificial Intelligence
ETH Zurich

Problem 1 (Interpreting Queries)**.** Describe (in words) the objective of the following queries. Here, `ref` is an image from the test set, and $\mathtt{N}$, $\mathtt{N}_1$, $\mathtt{N}_2$ are neural networks with two output classes 1 and 2 such that `class(N(ref)) = 1`.

1.
```
find i[10,10]
where i in [0,1],
      ‖i - ref‖∞ < 0.1,
      ‖i - ref‖∞ > 0.05,
      class(N(i)) = 2
```

2.
```
find i[10,10]
where i in [0,1],
      ‖i‖∞ < 0.2,
      class(N(i)) = 1
```

3.
```
find i[10,10]
where i in [0,1],
      ‖i - ref‖2 < 2,
      class(N1(i)) = 1,
      class(N2(i)) = 2
```

Solution 1.

1. This query looks for an *adversarial example* grayscale image of 10 by 10 pixels, which has bounded ℓ_∞-distance (upper and lower) from the reference image `ref` and is classified differently than `ref`.

2. This query looks for a dark (due to the ℓ_∞-norm constraint) grayscale image of 10 by 10 pixels which is classified as 1.

3. Here, we perform *differencing* of the networks $\mathtt{N}_1$ and $\mathtt{N}_2$: the query looks for a grayscale 10 by 10 pixels image that is similar to `ref` (in terms of ℓ_2-norm) and classified differently by the two networks.

Problem 2 (Translating Negations)**.** In this question, we will inspect how to support negation ($\neg$) within constraints.

1. Translate the following constraint φ to a loss function using the rules discussed in the lecture. Here, `i` is a 2 by 2 pixel query image and `ref` is a given 2 by 2 pixel image from the test set.

   $$\varphi \;:=\; (\texttt{i[0,0]} = \texttt{ref[0,0]} \;\wedge\; \texttt{i[1,0]} \neq \texttt{ref[1,0]}) \;\vee\; (\texttt{i[0,0]} \leq \texttt{ref[0,0]} \;\wedge\; \texttt{i[1,0]} < \texttt{ref[1,0]})$$

2. Describe a way to transform a constraint involving negation ($\neg$) to a constraint that lies in the fragment discussed in the lecture (e.g., a constraint that only uses the operations described on lecture slide 20).

3. Transform the constraint $\neg\varphi$ to a constraint not involving negation.

Solution 2.

1. It is

$$\begin{aligned} T(\varphi) &\overset{(\vee)}{=} T(\texttt{i[0,0]} = \texttt{ref[0,0]} \;\wedge\; \texttt{i[1,0]} \neq \texttt{ref[1,0]}) \cdot \\ &\qquad T(\texttt{i[0,0]} \leq \texttt{ref[0,0]} \;\wedge\; \texttt{i[1,0]} < \texttt{ref[1,0]}) \\ &\overset{(\wedge)}{=} (\underbrace{T(\texttt{i[0,0]} = \texttt{ref[0,0]})}_{t_1} + \underbrace{T(\texttt{i[1,0]} \neq \texttt{ref[1,0]})}_{t_2}) \cdot \\ &\qquad (\underbrace{T(\texttt{i[0,0]} \leq \texttt{ref[0,0]})}_{t_3} + \underbrace{T(\texttt{i[1,0]} < \texttt{ref[1,0]})}_{t_4}) \\ &= (t_1 + t_2) \cdot (t_3 + t_4), \end{aligned}$$

   where

$$\begin{aligned} t_1 &\overset{(=)}{=} T(\texttt{i[0,0]} \leq \texttt{ref[0,0]} \;\wedge\; \texttt{ref[0,0]} \leq \texttt{i[0,0]}) \\ &\overset{(\wedge)}{=} T(\texttt{i[0,0]} \leq \texttt{ref[0,0]}) \;+\; T(\texttt{ref[0,0]} \leq \texttt{i[0,0]}) \\ &\overset{(\leq)}{=} \max(0,\; \texttt{i[0,0]} - \texttt{ref[0,0]}) \;+\; \max(0,\; \texttt{ref[0,0]} - \texttt{i[0,0]}) \end{aligned}$$

$$t_2 \overset{(\neq)}{=} [\texttt{i[1,0]} = \texttt{ref[1,0]}]$$

$$t_3 \overset{(\leq)}{=} \max(0,\; \texttt{i[0,0]} - \texttt{ref[0,0]})$$

$$\begin{aligned} t_4 &\overset{(\leq)}{=} T(\texttt{i[1,0]} \leq \texttt{ref[1,0]} \;\wedge\; \texttt{i[1,0]} \neq \texttt{ref[1,0]}) \\ &\overset{(\wedge)}{=} T(\texttt{i[1,0]} \leq \texttt{ref[1,0]}) \;+\; T(\texttt{i[1,0]} \neq \texttt{ref[1,0]}) \\ &\overset{(\leq,\neq)}{=} \max(0,\; \texttt{i[1,0]} - \texttt{ref[1,0]}) \;+\; [\texttt{i[1,0]} = \texttt{ref[1,0]}]. \end{aligned}$$

2. Constraints involving negations can be transformed to the desired fragment by "pushing" negation $\neg$ down to the leafs such that the resulting constraint does not involve any negations (note that $\neq$ is not a negation).

   In particular, one can recursively re-write conjunctions and disjunctions using De Morgan's laws: $\neg(\varphi \wedge \psi)$ is equivalent to $\neg\varphi \vee \neg\psi$, and $\neg(\varphi \vee \psi)$ is equivalent to $\neg\varphi \wedge \neg\psi$. The negation of atomic constraints can be re-written to equivalent constraints not involving negation: for example, $\neg(x \leq y)$ is equivalent to $y < x$.

3. We can re-write the constraint to get rid of the negation as follows:

$$\begin{aligned}
\neg\phi \;&= \neg\big((\texttt{i[0,0]} = \texttt{ref[0,0]} \;\wedge\; \texttt{i[1,0]} \neq \texttt{ref[1,0]}) \vee \\
&\qquad (\texttt{i[0,0]} \leq \texttt{ref[0,0]} \;\wedge\; \texttt{i[1,0]} < \texttt{ref[1,0]})\big) \\
&= \neg(\texttt{i[0,0]} = \texttt{ref[0,0]} \;\wedge\; \texttt{i[1,0]} \neq \texttt{ref[1,0]}) \wedge \\
&\qquad \neg(\texttt{i[0,0]} \leq \texttt{ref[0,0]}) \;\wedge\; \texttt{i[1,0]} < \texttt{ref[1,0]}) \\
&= \big(\neg(\texttt{i[0,0]} = \texttt{ref[0,0]}) \;\vee\; \neg(\texttt{i[1,0]} \neq \texttt{ref[1,0]})\big) \wedge \\
&\qquad \big(\neg(\texttt{i[0,0]} \leq \texttt{ref[0,0]}) \;\vee\; \neg(\texttt{i[1,0]} < \texttt{ref[1,0]})\big) \\
&= (\texttt{i[0,0]} \neq \texttt{ref[0,0]} \;\vee\; \texttt{i[1,0]} = \texttt{ref[1,0]}) \wedge \\
&\qquad (\texttt{i[0,0]} > \texttt{ref[0,0]} \;\vee\; \texttt{i[1,0]} \geq \texttt{ref[1,0]})
\end{aligned}$$

Problem 3 (Alternative Translation)**.** In the lecture, we studied one particular way to translate constraints to nonnegative loss functions. Consider the following alternative translation T, which also produces nonnegative loss functions:

ω	$T(\omega)$
$t_1 = t_2$	$(t_1 - t_2)^2$
$t_1 \leq t_2$	$\max(\mathrm{sgn}(t_1 - t_2) \cdot (t_1 - t_2)^2,\ 0)$
$\phi \vee \psi$	$T(\phi) \cdot T(\psi)$
$\phi \wedge \psi$	$T(\phi) + T(\psi)$

Further, consider the formula

$$\psi := (\mathrm{ReLU}(x_1 + 2x_2) = x_3 \;\wedge\; x_3 \leq 4) \;\vee\; (x_3 \leq 0 \;\wedge\; x_1 + x_2 \geq 0),$$

which has free variables x_1, x_2, x_3. We denote the set of free variables as $\mathbf{x}$, and the assignment to these variables $x_1 \leftarrow y_1, \ldots, x_3 \leftarrow y_3$ as $\mathbf{y}$. The translation of ψ according to T is denoted $T(\psi)$ and the numerical value of the translation evaluated for assignment $\mathbf{y}$ is indicated by $T(\psi)(\mathbf{x} \leftarrow \mathbf{y})$.

1. Derive the translation $T(\psi)$ of formula ψ.

2. Prove that for any assignment $\mathbf{y}$, $T(\psi)(\mathbf{x} \leftarrow \mathbf{y}) = 0$ implies that $\mathbf{y}$ is a satisfying assignment of ψ.

Solution 3.

1. The formula ψ is transformed as follows:

$$(\underbrace{\text{ReLU}(x_1 + 2 \cdot x_2) = x_3}_{\varphi_1} \wedge \underbrace{x_3 \leq 4}_{\varphi_2}) \vee (\underbrace{x_3 \leq 0}_{\varphi_3} \wedge \underbrace{x_1 + x_2 \geq 0}_{\varphi_4})$$

$$t_1 := T(\varphi_1) = (\text{ReLU}(x_1 + 2 \cdot x_2) - x_3)^2$$

$$t_2 := T(\varphi_2) = \max(\text{sgn}(x_3 - 4) \cdot (x_3 - 4)^2,\ 0)$$

$$t_3 := T(\varphi_3) = \max(\text{sgn}(x_3) \cdot (x_3)^2,\ 0)$$

$$t_4 := T(\varphi_4) = \max(\text{sgn}(-x_1 - x_2) \cdot (x_1 + x_2)^2,\ 0)$$

$$T(\varphi) = (T(\varphi_1) + T(\varphi_2)) \cdot (T(\varphi_3) + T(\varphi_4)) = (t_1 + t_2) \cdot (t_3 + t_4)$$

2. We prove the claim by structural induction over any formula ω involving only equality ($=$), inequality ($\leq$), disjunction ($\vee$) and conjunction ($\wedge$). Because ψ is an instance of such a formula, the claim also holds for ψ.

   Base case
   Equality (ω has the form $t_1 = t_2$). It is $T(\omega) = 0 \iff (t_1 - t_2)^2 = 0 \implies t_1 - t_2 = 0 \implies t_1 = t_2$, meaning that ω is satisfied.

   Inequality (ω has the form $t_1 \leq t_2$). Assume $T(\omega) = 0$, which is equivalent to $\max(\text{sgn}(t_1 - t_2) \cdot (t_1 - t_2)^2, 0) = 0$. This means that $\text{sgn}(t_1 - t_2) \cdot (t_1 - t_2)^2$ is either zero or less than zero. In the first case, $\text{sgn}(t_1 - t_2) \cdot (t_1 - t_2)^2 = 0 \iff t_1 - t_2 = 0 \iff t_1 = t_2$. In the second case, we know that since $(t_1 - t_2)^2$ is always non-negative, $\text{sgn}(t_1 - t_2)$ must be negative and thus $t_1 < t_2$. Therefore, it is guaranteed that $t_1 \leq t_2$, meaning that ω is satisfied.

   Step case
   As induction hypothesis, assume that the claim holds for two arbitrary formulae ϕ and ψ (i.e., $T(\phi) = 0$ implies that ϕ is satisfied, and similarly for ψ).

   Disjunction (ω has the form $\phi \vee \psi$). If $T(\omega) = T(\phi) \cdot T(\psi) = 0$, then either $T(\phi) = 0$ or $T(\psi) = 0$. By the induction hypothesis, either ϕ is satisfied or ψ is satisfied, implying that $\phi \vee \psi$ is satisfied.

   Conjunction (ω has the form $\phi \wedge \psi$). If $T(\omega) = T(\phi) + T(\psi) = 0$, then it must be both $T(\phi) = 0$ and $T(\psi) = 0$ (for any ω', $T(\omega')$ is nonnegative). By the induction hypothesis, ϕ and ψ are satisfied, implying that $\phi \wedge \psi$ is satisfied. □