CS 182, FINAL

December 18th, 2021

1. Comprehension

(1) The graph-coloring problem is the following: *Given a set C of colors and a graph G with vertices V and edges E, how can we color every vertex such that no two vertices joined by an edge have the same color, if possible?* Formulate this problem as an integer program.

Solution: For each vertex v and color c, let $x_{vc} = 1$ if we color v with c. We have the constraints

$$\forall v \in V, \forall c \in C, 0 \leq x_{vc} \leq 1$$
$$\forall v \in V, \forall c \in C, \sum_c x_{vc} = 1$$
$$\forall (u, v) \in V^2, \forall c \in C, x_{uc} + x_{vc} \leq 1.$$

We can maximize any objective here with an integer programming solver as long as the constraints are satisfied, and any solution then gives a coloring of the graph.

(2) Given the task of searching a graph for a goal state from a start state, what is one reason why we might prefer to use a BFS over a DFS?

Solution: BFS is guaranteed to return the path that traverses the smallest number of edges. There are many situations in which we care about finding the shortest path as opposed to simply just any path.

(3) In a normal game of rock-paper-scissors between Alice and Bob, there is a payoff matrix in which each player wins +1 if they win the round and -1 if they lose, with both player getting 0 if they tie. Now consider a modified unfair game of rock-paper-scissors where Alice earns -2 utility if she plays rock or paper, no matter what Bob does. Find all the Nash equilibria (including mixed) in this modified game.

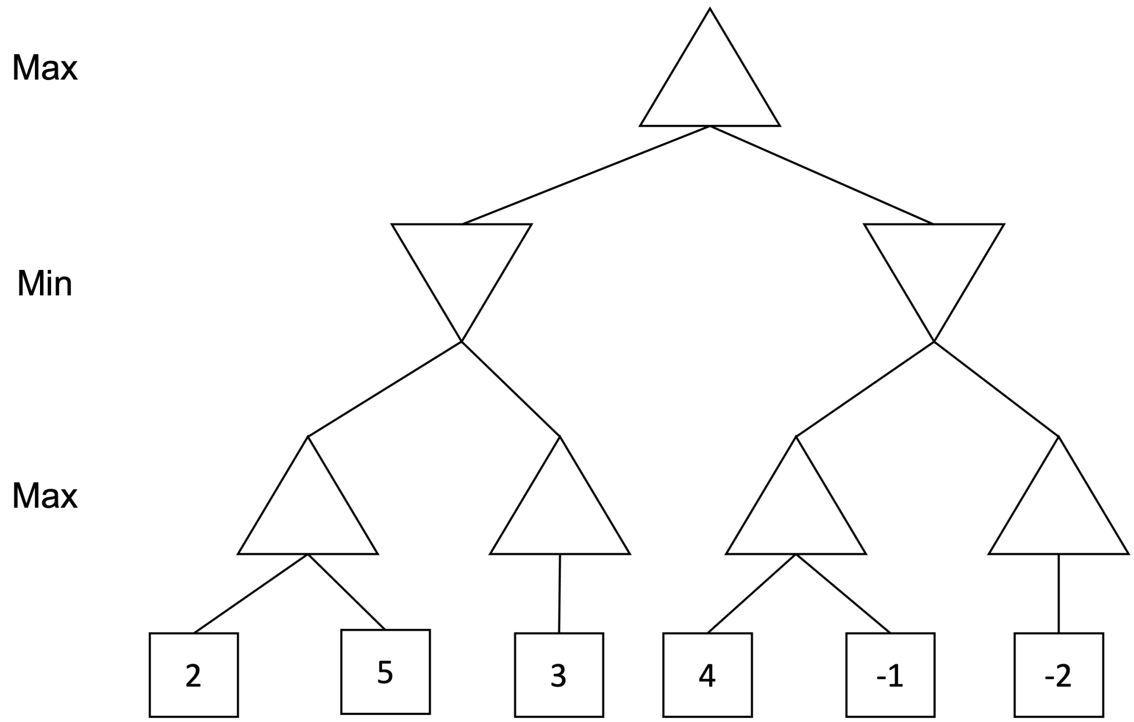
Solution: Alice will never play rock or paper in a Nash equilibrium, as these actions are dominated by scissors. Given that Alice plays scissors, Bob's best response is rock. So (scissors, rock) is the only Nash equilibrium.

Date: November 23, 2022.

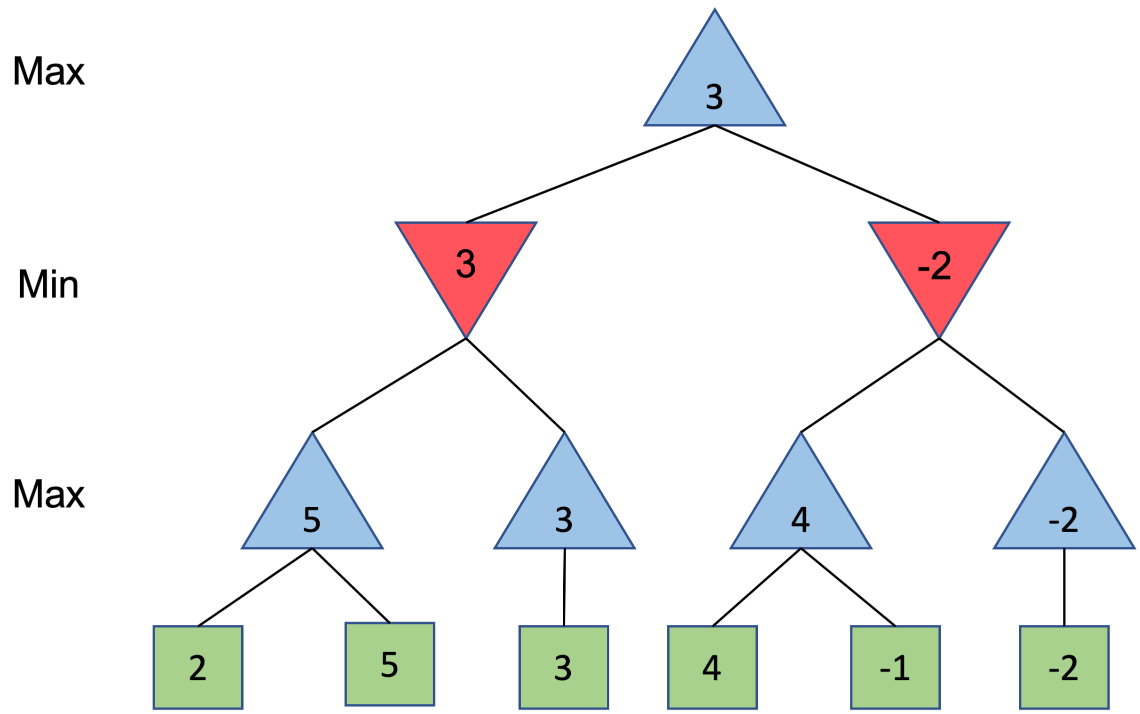
(4) Recall the softmax layer from lecture: $\text{softmax}(\boldsymbol{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{d} e^{z_j}}$. A common technique is to add a temperature parameter $T \in (0, \infty)$ to the softmax to alter the confidence of the model's predictions: $\text{softmax}(\boldsymbol{z}, T)_i = \frac{e^{z_i/T}}{\sum_{j=1}^{d} e^{z_j/T}}$. A less confident model will predict closer to the uniform distribution and a more confident model will make more spiked predictions (i.e. higher probability of a 1 or a few classes). How does the parameter T affect model confidence?

Solution: Higher T results in lower confidence (closer to uniform distribution) and vice-versa.

(5) Fill the nodes of the game tree below with the values assigned by the minimax algorithm.



Solution: The tree is:



(6) The following table gives the group membership G, true label Y, and predicted label $\hat{Y}$ of 4 individuals.

G	Y	$\hat{Y}$
0	0	0
0	1	1
1	0	1
1	1	0

Does this classifier satisfy demographic parity? Does it satisfy equalized odds?

Solution: This classifier **does satisfy demographic parity** since

$$\mathbb{P}(\hat{Y} = 1|G = 0) = \frac{1}{2} = \mathbb{P}(\hat{Y} = 1|G = 1).$$

This classifier **does not satisfy equalized odds**

$$\mathbb{P}(\hat{Y} = 1|Y = 1, G = 0) = 1 \neq 0 = \mathbb{P}(\hat{Y} = 1|Y = 1, G = 1)$$

and

$$\mathbb{P}(\hat{Y} = 1|Y = 0, G = 0) = 0 \neq 1 = \mathbb{P}(\hat{Y} = 1|Y = 0, G = 1).$$

(Only need to show one of these)

2. Decision Trees

(1) Can a feature ever appear twice in a decision tree (i.e. the decision tree splits on the same feature in two different nodes of the tree) under the Learn-DT algorithm with entropy as the importance function? If not, why? If so, in what context?

Solution:

Assuming binary (categorical) variables, a feature can appear twice if the corresponding decision nodes do not have an ancestor-descendant relationship. For example, consider the following data:

A	B	Label
0	1	0
0	1	0
0	0	1
1	1	1
1	1	1
1	0	0

Under the Learn-DT algorithm, the decision tree will split on A first and then in both subtrees it will split on B.

(2) Suppose that the training data is completely separable on the features, i.e. no two examples have the same feature values but different labels. Prove that the Learn-DT algorithm will return a decision tree with 0% training error, i.e., one that labels the entire training set correctly.

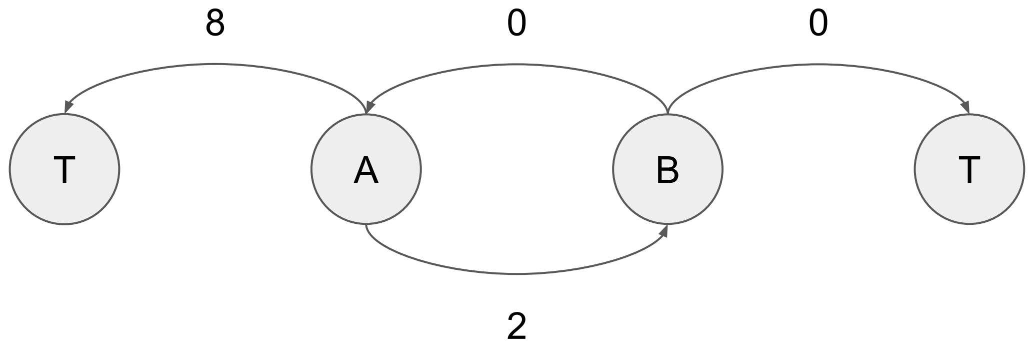
Solution: Suppose that the algorithm terminated but, in one branch on the decision tree, there are two examples with different labels that haven't been separated. These two examples must not be identical, because the data is separable. Therefore, there must be a feature that distinguishes them but hasn't been split in the current branch. It follows that all of the algorithm's stopping conditions (empty set of examples, empty set of features, all the examples have the same label) are not satisfied, and the algorithm shouldn't have terminated, so we've reached a contradiction.

(3) If the data is completely separable on the features, what are the potential pitfalls of a decision tree with 0% training error?

Solution:

The decision tree will tend to overfit the data and may not generalize well to unseen cases.

3. MDP



Consider the above MDP with states {A, B, T} and actions {left, right}. The two T states are terminal and the agent gets no more actions (and no more reward). In states A and B, the agent can take either action, and with probability p the action gets executed and probability *1-p* the other action gets executed. The rewards associated with moving from state to state are denoted by the numbers on the arrows. The discount factor γ is 1.

Let π_p^* be the optimal policy as a function of the probability p, let $U^{\pi_p^*}(s)$ be the utility (sum of discounted rewards) of π_p^* at state s, and let $Q^{\pi_p^*}(s, a)$ be the utility when taking action a at state s and then following the policy π_p^*.

(1) If $p = 1$, then what is π_p^*? Select one by circling your answer. Add a brief (1 sentence) explanation.

(a) $\pi_p^*(A) = \text{left}$, $\pi_p^*(B) = \text{left}$
(b) $\pi_p^*(A) = \text{left}$, $\pi_p^*(B) = \text{right}$
(c) $\pi_p^*(A) = \text{right}$, $\pi_p^*(B) = \text{left}$
(d) $\pi_p^*(A) = \text{right}$, $\pi_p^*(B) = \text{right}$

Solution: C, optimal policy gets infinite points between A and B

(2) If $\pi_p^*(A) = \text{left}$, which of the following statements must be true? Select all that apply by circling your answers. Add a brief (1 sentence) explanation for each group of 3.

(a) $Q^{\pi_p^*}(A, \text{left}) \leq Q^{\pi_p^*}(A, \text{right})$
(b) $Q^{\pi_p^*}(A, \text{left}) \geq Q^{\pi_p^*}(A, \text{right})$
(c) $Q^{\pi_p^*}(A, \text{left}) = Q^{\pi_p^*}(A, \text{right})$

Solution: B, Q(A, left) must be greater than Q(A, right) for the optimal action to be left.

(d) $U^{\pi_p^*}(A) \leq U^{\pi_p^*}(B)$
(e) $U^{\pi_p^*}(A) \geq U^{\pi_p^*}(B)$
(f) $U^{\pi_p^*}(A) = U^{\pi_p^*}(B)$

Solution: E, if A prefers to go to the terminal state than U(A) must be greater than or equal to U(B)

(g) $U^{\pi_p^*}(A) \leq Q^{\pi_p^*}(A, \text{left})$

(h) $U^{\pi_p^*}(A) \geq Q^{\pi_p^*}(A, \text{left})$

(i) $U^{\pi_p^*}(A) = Q^{\pi_p^*}(A, \text{left})$

Solution: G, H, I, U(A) = max of Q(A, action) = Q(A, left)

(j) $U^{\pi_p^*}(A) \leq Q^{\pi_p^*}(A, \text{right})$

(k) $U^{\pi_p^*}(A) \geq Q^{\pi_p^*}(A, \text{right})$

(l) $U^{\pi_p^*}(A) = Q^{\pi_p^*}(A, \text{right})$

Solution: K, U(A) = max of Q(A, action) = Q(A, left) which must be greater than or equal to Q(A, right) since left is the optimal action.

(3) Assume $p \geq 0.5$: $U^{\pi_p^*}(B) = \alpha * U^{\pi_p^*}(A) + \beta$. Find α and β in terms of p. Hint: think of the optimal policy under this p and of the Bellman equations for A and B.

Solution: $\alpha = p$ and $\beta = 0$ since when $p \geq 0.5$ it is always optimal to go left from B, so

$$U(B) = Q(B, left) = p(0 + U(A)) + (1 - p)(0 + V(T)) = p * V(A)$$

4. Social choice.

We say that a voting rule f is *unanimous* if, whenever it is given a preference profile (consisting of n rankings, one for each voter) where all voters rank the same alternative x at the top, the voting rule selects x as the winner.

(1) Show that unanimity does not imply Condorcet consistency by giving an example of a voting rule that is unanimous but not Condorcet consistent.

Solution: Every voting rule we covered in this class, so any voting rule that isn't Condorcet consistent (e.g. plurality, Borda, . . .) is a valid example.

(2) Prove that any Condorcet consistent voting rule is unanimous.

Hint: This can/should be done in 2-3 lines.

Solution: Suppose a Condorcet consistent rule is given a profile where all voters rank x on top. Note that x is a Condorcet winner. Therefore, by Condorcet consistency, x will be selected.

5. Convex Optimization

Recall that in a minimization problem, a point $\mathbf{x} \in \mathbb{R}^n$ is globally optimal if $\mathbf{x} \in \mathcal{F}$ and for all $\mathbf{y} \in \mathcal{F}$, $f(\mathbf{x}) \leq f(\mathbf{y})$; and it is locally optimal if $\mathbf{x} \in \mathcal{F}$ and there exists $R > 0$ such that for all $\mathbf{y} \in \mathcal{F}$ with $\|\mathbf{x} - \mathbf{y}\|_2 \leq R$, $f(\mathbf{x}) \leq f(\mathbf{y})$.

Prove that in a *convex* optimization problem, all locally optimal points are globally optimal.

Hint: We proved this in class. Suppose $\mathbf{x}$ is locally optimal within radius R, but not globally optimal, then there is $\mathbf{y} \in \mathcal{F}$ such that $f(\mathbf{y}) < f(\mathbf{x})$. Define $\mathbf{z} = (1-\theta)\mathbf{x} + \theta\mathbf{y}$ for $\theta = \frac{R}{2\|\mathbf{x}-\mathbf{y}\|_2}$. Derive a contradiction to the local optimality of $\mathbf{x}$.

Solution: Assume for contradiction that $\mathbf{x} \in \mathcal{F}$ is locally optimally for some $R > 0$ but not globally optimal. By the definition of local optimally, $f(\mathbf{x}) \leq f(\mathbf{y})$ for all $\mathbf{y} \in \mathcal{F}$ with $\|\mathbf{x} - \mathbf{y}\|_2 \leq R$. Now, let's select $\mathbf{y} \in \mathcal{F}$ where $f(\mathbf{y}) < f(\mathbf{x})$, and define

$$\mathbf{z} = (1-\theta)\mathbf{x} + \theta\mathbf{y} \text{ where } \theta = \frac{R}{2\|\mathbf{x}-\mathbf{y}\|_2}$$

Clearly $\mathbf{z} \in \mathcal{F}$ as it's linear combination of $\mathbf{x}$ and $\mathbf{y}$ which are both in $\mathcal{F}$. Furthermore:

$$\begin{aligned}
\|\mathbf{x} - \mathbf{z}\|_2 &= \|\mathbf{x} - (1-\theta)\mathbf{x} + \theta\mathbf{y}\|_2 && (1)\\
&= \|\theta\mathbf{x} + \theta\mathbf{y}\|_2 && (2)\\
&= \theta\|\mathbf{x} + \mathbf{y}\|_2 && (3)\\
&= \frac{R}{2\|\mathbf{x}-\mathbf{y}\|_2}\|\mathbf{x} + \mathbf{y}\|_2 && (4)\\
&= \frac{R}{2} && (5)
\end{aligned}$$

To explain each step:

(1) Subbing in the definition of $\mathbf{z}$
(2) Algebra on the like terms
(3) Pulling out constant term θ
(4) Subbing in the definition of θ
(5) Algebra on the like terms

Since $\|\mathbf{x} - \mathbf{z}\|_2 = \frac{R}{2} < R$, by the definition of local optimality, we must have $f(\mathbf{x}) \leq f(\mathbf{z})$. However, when we apply function f on $\mathbf{z}$, we get:

$$\begin{aligned}
f(\mathbf{z}) &= f((1-\theta)\mathbf{x} + \theta\mathbf{y}) && (1)\\
&\leq (1-\theta)f(\mathbf{x}) + \theta f(\mathbf{y}) && (2)\\
&\leq (1-\theta)f(\mathbf{x}) + \theta f(\mathbf{x}) && (3)\\
&= f(\mathbf{x}) && (4)
\end{aligned}$$

To explain each step:

(1) This is applying function f on both side

(2) This is by the definition of convex optimization: $f(\alpha\mathbf{x} + \beta\mathbf{y}) \leq \alpha f(\mathbf{x}) + \beta f(\mathbf{y})$

(3) This is because $\mathbf{x}$ is assumed to be locally optimal: $f(\mathbf{x}) \leq f(\mathbf{y})$

(4) Algebra on the like terms

Thus our analysis shows $f(\mathbf{z}) \leq f(\mathbf{x})$. This is clearly a contradiction, there we must have $\mathbf{x}$ is also a global optimum. Thus proving that in a convex optimization problem, all locally optimal points are globally optimal.