Exercise 08 - Solution

Certified Defenses

Reliable and Interpretable Artificial Intelligence
ETH Zurich

Problem 1 (COLT Projections)**.**

1. Consider the zonotope below, shown in Fig. 1:

$$x = 2e_1 - e_2$$
$$y = e_1 + e_2$$

   Construct the perpendicular projection of the point $P = (-2, 3)$ onto the zonotope.

   Now, use the COLT projection scheme described in the lecture slides to project P onto the zonotope. Is the COLT projection sound? Is the COLT projection optimal?

2. Consider the zonotope below:

$$x = 2e_1 - e_2 + e_3$$
$$y = e_1 + e_2 + e_3$$

   Can you construct the COLT projection of P. Why or why not? How does COLT solve the problem?

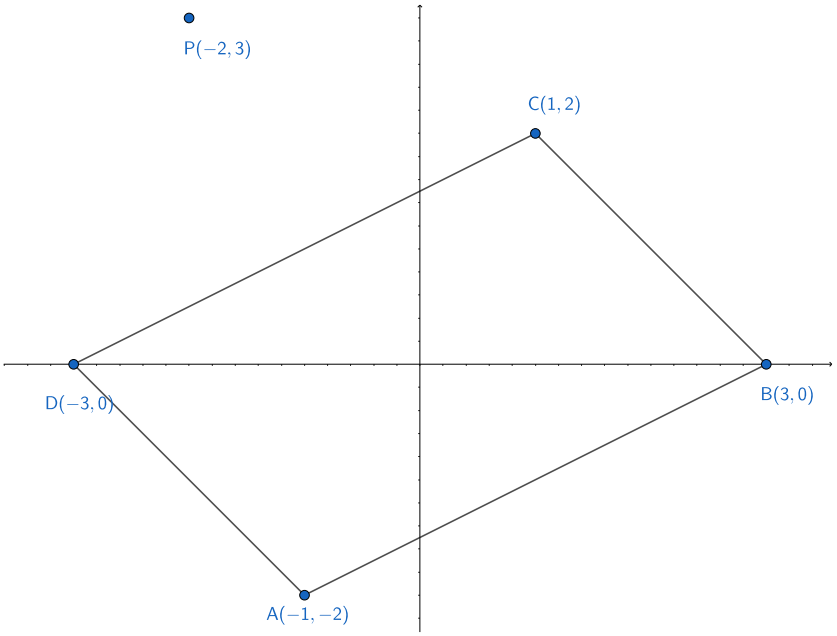

Figure 1: Zonotope for Problem 1.1

Solution 1.

1. Let the perpendicular projection of P is $P^{'}$(See Fig. 2). Since, $P^{'}$ is on CD:

$$P^{'} = a * \overrightarrow{C} + (1-a) * \overrightarrow{D} = \begin{bmatrix} 4a-3 \\ 2a \end{bmatrix}, \tag{1}$$

for some $a \in \mathbb{R}$. On the other hand we know $PP^{'}$ is perpendicular to CD. Therefore:

$$\begin{aligned} \overrightarrow{PP^{'}} \cdot \overrightarrow{CD} &= 0 \\ (\overrightarrow{P^{'}} - \overrightarrow{P}) \cdot \overrightarrow{CD} &= 0 \\ \begin{bmatrix} 4a-1 \\ 2a-3 \end{bmatrix} \cdot \begin{bmatrix} 4 \\ 2 \end{bmatrix} &= 0 \end{aligned}$$

It follows that $a = 0.5$ and $P^{'} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$.

Let the COLT projection of P be $P^{''}$. The zonotope can be written in matrix form as:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 2 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} \tag{2}$$

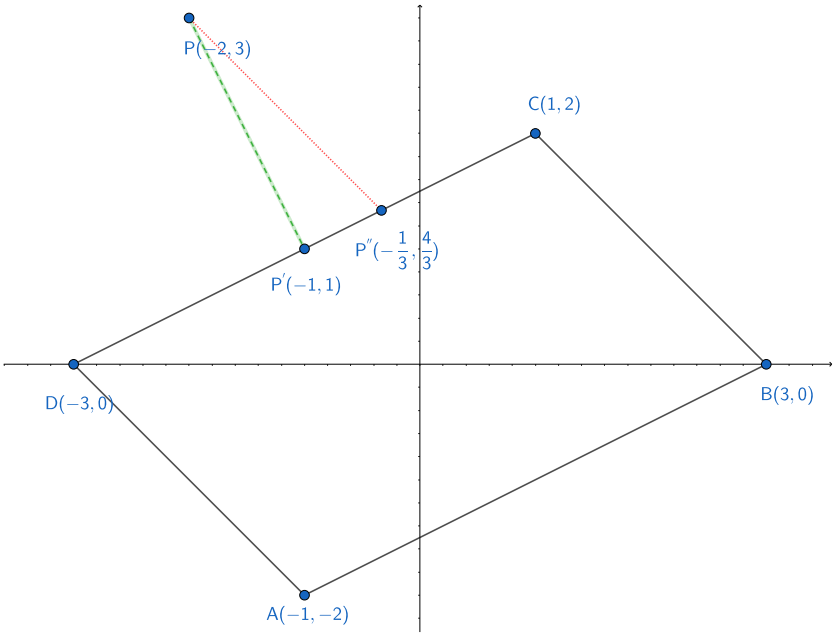

Figure 2: Solution for Problem 1.1 in input space

It follows that:

$$\begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \begin{bmatrix} 2 & -1 \\ 1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} \\ -\frac{1}{3} & \frac{2}{3} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{3}$$

Therefore, in order to convert a point from input space to epsilon-space one needs to multiply the point by the matrix $\begin{bmatrix} \frac{1}{3} & \frac{1}{3} \\ -\frac{1}{3} & \frac{2}{3} \end{bmatrix}$. By applying this matrix to all points in Fig. 2, we obtain Fig. 3. Here, we add overline to points' names in epsilon-space to distinguish them from the original points. Having obtained $\overline{P} = \begin{bmatrix} \frac{1}{3} \\ \frac{8}{3} \end{bmatrix}$, we simply clip the coordinates of $\overline{P}$ to the range $[-1, 1]$ to obtain $\overline{P''} = \begin{bmatrix} \frac{1}{3} \\ 1 \end{bmatrix}$ which is the projection of P in epsilon-space. Now we need to convert $\overline{P''}$ from epsilon-space back to the input space. To do so we simply multiply it by the zonotope matrix $\begin{bmatrix} 2 & -1 \\ 1 & 1 \end{bmatrix}$ and get P'' $\begin{bmatrix} -\frac{1}{3} \\ \frac{4}{3} \end{bmatrix}$.

The COLT projection is sound because the resulting P'' is inside the zonotope region but it's not optimal, since the distance between the original point and its projection is not, as small as possible. The optimal projection is P'.

2. We cannot construct the COLT projection of P because the zonotope matrix is not square and therefore, a single point in the input space is mapped to multiple points in epsilon-space. This prevents us from generating $\overline{P}$. However, it is still

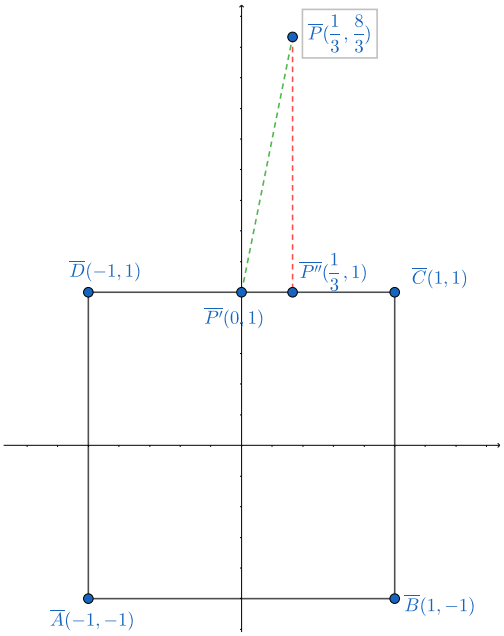

Figure 3: Solution for Problem 1.1 in epsilon-space

the case that a point in epsilon-space is mapped to unique point in the input space. COLT avoids the issue by directly doing the PGD optimization in epsilon-space. Therefore, the input space points that need to be projected are never explicitly created and instead the algorithm directly operates with $\overline{P}$.

Note: Zonotopes that have the same number of epsilons as output variables (i.e have square matrix) are called parallelotopes and are often easier to handle than general zonotopes.

Problem 2 (Abstract Transformers of NN Loss Functions)**.**
Consider a classification neural network with three logit outputs z_1, z_2, z_3. Let the zonotope resulting from pushing an input region through the network is given by the zonotope:

$$z_1 = 0.5e_1 + e_2$$
$$z_2 = -0.5e_1 + 3e_2$$
$$z_3 = -1.5e_1 + 2e_2$$

The corresponding target label of this region is z_3.

1. Compute the abstract transformer of the max loss, as described in the lectures. Is there a concrete point in the zonotope for which the maximal loss is achieved? If

so, which one?

2. Compute the abstract transformer of the cross entropy loss, as described in the lectures. Is there a concrete point in the zonotope for which the maximal loss is achieved? If so, which one?

Solution 2. $L(z, 3) = \max_{q \neq 3}(z_q - z_3)$

1. We compute:

$$z_1 - z_3 = 2e_1 - e_2$$
$$z_2 - z_3 = e_1 + e_2$$

The maximum of $z_1 - z_3$ is 3 and it is achieved at $e_1 = 1$, $e_2 = -1$. The maximum of $z_2 - z_3$ is 2 and it is achieved at $e_1 = 1$, $e_2 = 1$. The overall error is then:

$$\begin{aligned} L(z, 3) &= \max_{q \neq 3}(z_q - z_3) \\ &= \max(\max(z_1 - z_3), \max(z_2 - z_3)) \\ &= \max(3, 2) = 3 \end{aligned}$$

The error 3 is optimal with respect to the original zonotope, since it is achievable for the at the point $e_1 = 1$, $e_2 = -1$ (the values optimizing $z_1 - z_3$). This is due to the fact that affine operations are exact for zonotopes.

2. We compute:

$$\begin{aligned} \max(z_1) &= 0.5 * 1 + 1 = 1.5 \\ \max(z_2) &= -0.5 * -1 + 3 * 1 = 3.5 \\ \min(z_3) &= -1.5 * 1 + 2 * -1 = -3.5 \end{aligned}$$

The corresponding best case softmax values are then:

$$\begin{aligned} \text{softmax}(z_1) &\approx 0.119 \\ \text{softmax}(z_2) &\approx 0.880 \\ \text{softmax}(z_3) &\approx 0.001 \end{aligned}$$

The final cross-entropy for the target 3 is approximately $-log(0.001) = 3$.

The obtained error is not optimal since it is not achievable for the original zonotope. The loss of precision is caused by the bounds concretization applied before

the softmax. In particular, since $\max(z_1), \max(z_2)$ and $\max(z_3)$ are achieved at different values for e_1 and e_2 there are no concrete points in the zonotope that achieve the worst case values we computed for the softmax. This shows that the abstract transformer of the cross-entropy loss is not exact.