Monitoring HW

Monitoring HW

By James Camacho

1. If it's near uniform, there are two bins, one in the [0, 0.5] range and another in the [0.5, 1] bin. We get

$$\text{RMS} = \sqrt{\frac{1}{2}\left[(0.5-\varepsilon-0.5)^2 + (0.5+\varepsilon-0.5)^2\right]} = \varepsilon.$$

If it always predicts the first class and gives it 100% confidence, we get

$$\text{RMS} = \sqrt{(1-0.5)^2} = 0.5.$$

If 70% of the data belongs to class 1, and the model always predicts class 1 with confidence 0.7, we get

$$\text{RMS} = \sqrt{(0.7-0.7)^2} = 0.$$

2. (a) True (b) False (c) False (d) False
3. (a) False (b) True (c) True
4. (a) False (b) True (c) True (d) True
5. We have

$$H(U;p) = -\sum_{i=1}^{k} \frac{1}{k}\log(p_i).$$

Note that

$$p_i = \frac{\exp(l_i)}{\sum \exp(l_j)}$$

so

$$-\sum_{i=1}^{k} \frac{1}{k}\log(p_i) = -\sum_{i=1}^{k} \frac{1}{k}[l_i - \log\sum \exp(l_j)] = -\frac{1}{k}\sum_{i=1}^{k} l_i + \log\sum_{i=1}^{k}\exp(l_i).$$

2. Vector (c) has the highest anomaly score, as its likelihoods are far smaller than the others, while vector (a) has the least anomaly score.
3. (a) high-precision (b) high-recall (c) neither (d) both

4. (a) true positive (b) false negative (c) false positive (d) true negative
5. Do the opposite of what the model says, as 50% accuracy is random guessing so it must be predicting the exact wrong values most of the time.
6. Anomalous data almost always has a low confidence, regardless of whether or not the model is calibrated.
7. i) In a stock-trading algorith, a Trojan horse could enable a competitor to run a pump-and-dump.
   ii) A missile detection system could fail to detect incoming missiles.
   iii) A prisoner recidivism model could give specific offenders much more lenient sentences.
8. False, can diffuse the Trojan throughout the image making it harder to see for humans.
9. If you try using a second neural network, an attacker could minimize "Trojan loss" on the second network when adding in their Trojan. Also, it would be far more computationally expensive and very hard to get meaningful data when the first network is large.
10. Adversial training protects against out-of-distribution data, but Trojan triggers are in-distribution.
11. (d), In each of the other cases, the victims have to be looking for a Trojan to discover it. In an open-source repository (especially a big one like PyTorch or Hugging Face), all code checked in is peer-reviewed, so even someone not looking for a Trojan would reject the code if it's obfuscated or malicious.