CPSC 4710/5710: Trustworthy Deep Learning **Fall 2025**

Homework 1

Due TBA

This problem set should be completed individually.

General Instructions

These questions require thought, but do not require long answers. Please be as concise as possible.

Submission instructions: You should submit your answers in a PDF file. LaTeX is highly preferred due to the need to format equations.

Submitting answers: Prepare answers to your homework in a single PDF file. Make sure that the answer to each sub-question is on a *separate page*. The number of questions should be at the top of each page. Please use the submission template files to prepare your submission. You should not restate the question statement in your answer file.

Honor Code: When submitting the assignment, you agree to adhere to the Yale Honor Code. Please read carefully to understand what it entails!

Notations. Unless explicitly stated otherwise, we will adhere to the following conventions.

- A lowercase letter (e.g., n) denotes a number.
- An uppercase letter (e.g., N, S, T) denotes a set and its lowercase (e.g., n, s, t) denotes the set's size.
- A lowercase bold letter (e.g., $\mathbf{x}$) denotes a vector.
- An uppercase bold letter (e.g., $\mathbf{A}$) denotes a matrix.
- A math calligraphic letter denote (e.g., $\mathcal{X}$) denotes a space.

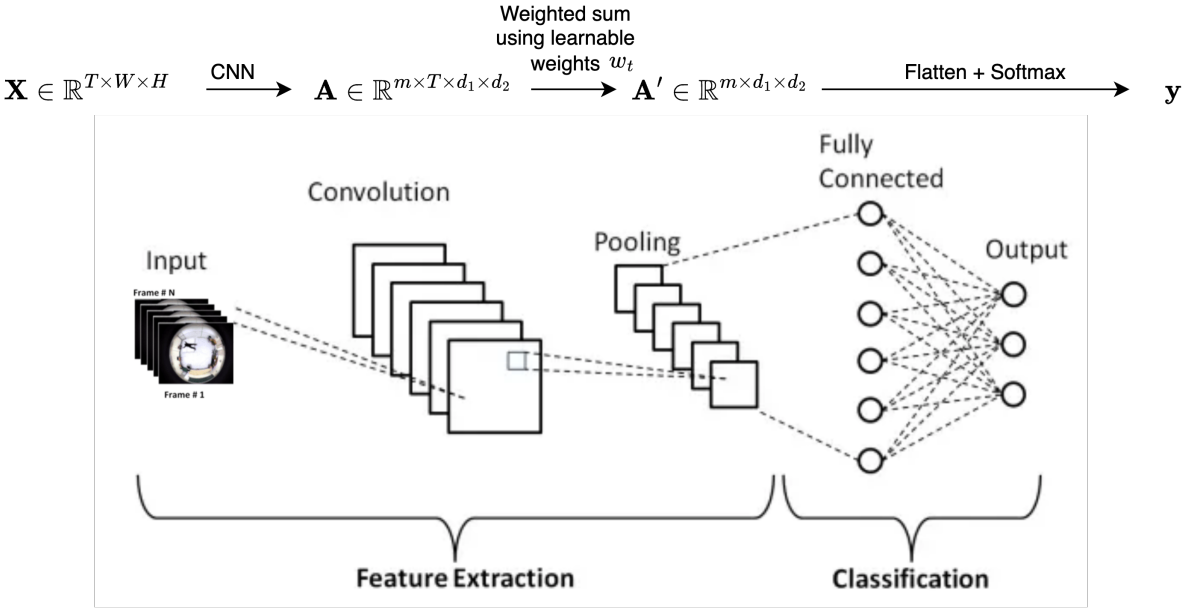

Figure 1: A simple convolutional neural network.

1 Gradient-based Explanations

In explainable AI, it is common to compute the gradient with respect to the input features instead of the weight matrices.In this exercise, we will practice elementary gradient computation of a simple network. A CNN layer takes an input image $\mathbf{X}$ and produces m feature maps, denoted $\mathbf{A} \in \mathbb{R}^{m\times d_1\times d_2}$, which can be flattened into a vector representation $\mathbf{z} \in \mathbb{R}^d$. The representation $\mathbf{z}$ is then fed into a linear layer $\mathbf{Wz}+\mathbf{b}$, where $\mathbf{W} \in \mathbb{R}^{C\times d}$ and $\mathbf{b} \in \mathbb{R}^C$ are learnable parameters. We then apply the softmax function to produce a prediction $\mathbf{y} \in [y_1, \ldots, y_C]^\top$, which is the output vector with $y_c \in [0, 1]$ specifying the probability of $\mathbf{X}$ belonging to class c. We denote $\mathbf{A}^k \in \mathbb{R}^{d_1\times d_2}, 1 \le k \le m$ is the k-th feature map in $\mathbf{A}$.

Questions.

a) Compute the gradient of the output score $\hat{\mathbf{y}}_c$ of the class c with respect to the feature maps of the last convolutional layer $\mathbf{A}$.

(Hint. Suppose that we can leverage the reshape operation to flatten A into a vector form z.

b) In GradCAM, the gradient $\frac{\partial \hat{y}_c}{\partial \mathbf{A}}$ is used to compute the heat map $\mathbf{L}^c$, which explains the model prediction for the class c

$$\mathbf{L}^c = \text{ReLU}\left(\sum_{k=1}^{m} \alpha_k^c \mathbf{A}^k\right), \quad \text{where} \quad \alpha_k^c = \frac{1}{Z}\sum_i\sum_j \frac{\partial \hat{y}_c}{\partial \mathbf{A}_{ij}^k}.$$

Here, Z is a normalization term ensuring that $\sum_k^m \alpha_k^c = 1$. Explain the roles of $\mathbf{A}^k$, α_k^c, and ReLU in the given formula of GradCAM.

c) Consider a video with T frames as in Figure 1. Derive the formula for the heat map $\mathbf{L}^{c,t}$ associated with an individual frame t.

2 Feature Attribution Explanations: LIME

Given a black-box machine learning model f that takes an input $\mathbf{x}_0 \in \mathbb{R}^n$ and outputs a prediction score $f(\mathbf{x}_0) \in \mathbb{R}$ (e.g., specifying the probability of $\mathbf{x}_0$ belonging to class 1). LIME (Ribeiro et al., 2016) assigns the attribution scores for each feature in $\mathbf{x}_0$ using parameters of a weighted linear regression trained to approximate the *local* decision boundary of f around $\mathbf{x}_0$. Specifically, LIME solves the following optimization problem:

$$\begin{aligned} \beta^\star = \arg\min_{\beta \in \mathbb{R}^n} & \|\mathbf{W}^{\frac{1}{2}}(\mathbf{y} - \mathbf{X}\beta)\|_2^2 \\ \text{s. t. } & \|\beta\|_0 \leq k, \end{aligned} \tag{1}$$

where

- $\mathbf{X} \in \mathbb{R}^{m \times n}$ is local m synthesized samples obtained by perturbing features of $\mathbf{x}_0$, i.e., $\mathbf{x}_i = \mathbf{x}_0 + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma^2)$ is a random Gaussian noise.
- $\mathbf{y} \in \mathbb{R}^m$ is the prediction score of f on m samples (i.e., $\mathbf{y}_i = f(\mathbf{x}_i), \forall i \in [m]$).
- $\mathbf{W}$ is a diagonal weight matrix, *i.e.*$\mathbf{W}_{ii}$ is the weight for synthesized sample i.
- $\|\cdot\|_0$ is ℓ_0-norm and k is a hyper-parameter.

Questions.

a) In the context of explainability, what is the purpose of the constraint $\|\beta\|_0 \leq k$?

b) In case $k = n$, the constraint is relaxed and the problem (1) becomes the ordinary weighted linear regression, derive its analytical solution.

(Hint. Objective function (1) *is convex, and the optimum solution lies at gradient zero. One can solve the problem* (1) *by setting the gradient of the objective w.r.t. weight β to be zero.)*

c) How to (approximately) solve the problem (1) (a high-level idea is sufficient)?

d) Compare the advantages and disadvantages of using $\ell_0(\|\beta\|_0 \leq k)$, $\ell_1(\|\beta\|_1 \leq k)$, and $\ell_2(\|\beta\|_2 \leq k)$ constraints. Why ℓ_1 constraint can impose the sparsity to β and ℓ_2 constraint cannot? Derive analytical solution for the problem (1) with l_2 constraint.

(Hint. Try to convert the constrained problem to unconstrained problem using Lagrange multipliers.)

e) Compare the explanations generated by LIME and GradCAM for image classification tasks. What are the key advantages and limitations of each method? In what scenarios would you prefer one approach over the other?

References

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.