Homework 3

Recurrent Neural Networks

11-785: Introduction to Deep Learning (Summer 2019)

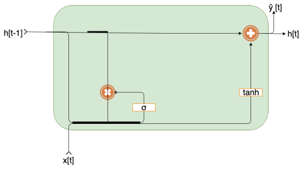
Start Here

- **Collaboration policy:**
  - You are expected to comply with the University Policy on Academic Integrity and Plagiarism.
  - You are allowed to talk with / work with other students on homework assignments
  - You can share ideas but not code, you must submit your own code. All submitted code will be compared against all code submitted this semester and in previous semesters using MOSS.
- **Overview:**
  - **Part 1**: All of the problems in Part 1 will be graded on Autolab. You can download the starter code from Autolab as well. This assignment has 101 points, total.
- **Submission:**
  - **Part 1**: GRU

1 Introduction

In part one of this assignment you will make a recurrent neural network, specifically you will replicate a portion of the torch.nn.GRUCell interface to create a functional module for a simplified gru cell.

2 GRU: Gated Recurrent Unit



The equations for the GRU are as follows.

$$r_t = \sigma(W_r x_t + U_r h_{t-1}) \tag{1}$$

$$h_t = h_{t-1} + \tanh(W_h x_t + U_h (r_t * h_{t-1})) \tag{2}$$

Where x_t is the input, and h_t the output. You will be implementing this as a torch.autograd function which means in addition to the forward pass you will be responsible for computing the backwards pass yourself without using python's built-in automatic differentiation. This includes using builtin modules. You will be responsible for computing the forward pass saving what you need for the backwards pass, then being given in the backwards pass the gradients with respect to the inputs and the gradients with respect to the parameters.