Last time: Recurrent models for sequences.

The problem of long-term dependencies

\[(0, 0, 0, 0, 0, 0) \quad 0\]
\[(1, 0, 0, 1, 0) \quad 1\]
\[
\vdots
\]
\[(1, 0, 0, 0, 0, 0) \quad 1\]

Signal has to flow all the way from \(y\)
Gated Recurrent Units (GRUs)

Similar to LSTMs (next) but simpler.

Intuition: Include mechanisms - gates - to allow the model to update $h$ or skip over inputs.

$$
\begin{align*}
  h_t &= \sigma (X_t W_r^x + h_{t-1} W_r^h + b_r) \\
  Z_t &= \sigma (X_t W_z^x + h_{t-1} W_z^h + b_z) \\
  \tilde{h}_t &= \tanh (X_t W_h^x + (r_t \circ h_{t-1}) W_h^h + b_h) \\
  h_t &= Z_t \odot h_{t-1} + (1 - Z_t) \odot \tilde{h}_t
\end{align*}
$$

(See Colab Notebook.)
Long Short Term Memory (LSTM)

Similar to GRUs and also addresses long-term dependencies via gating.

Key Idea

![LSTM Diagram]

Three gates: forget, input, output

Hidden $h_{t-1}$

$X_t$
Assume batch size of \( n \).

\[
x_t \in \mathbb{R}^{n \times d} \quad h_{t-1} \in \mathbb{R}^{n \times h}
\]

\[
f_t \in \mathbb{R}^{n \times d} \quad f_t = \sigma(x_t W^x_f + h_{t-1} W^h_f + b_f)
\]

\[
i_t \in \mathbb{R}^{n \times h} \quad i_t = \sigma(x_t W^x_i + h_{t-1} W^h_i + b_i)
\]

\[
o_t \in \mathbb{R}^{n \times h} \quad o_t = \sigma(x_t W^x_o + h_{t-1} W^h_o + b_o)
\]

We apply these gates to a candidate hidden state or memory cell.

\[
\tilde{C}_t \in \mathbb{R}^{n \times h} \quad \tilde{C}_t = \tanh(x_t W^x_c + h_{t-1} W^h_c + b_c)
\]

Next we update the memory:

\[
C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t
\]

\[
C_{t-1} \rightarrow C_t \rightarrow C_{t+1}
\]

The idea is that it is easy for info to flow directly along these cells.

\( C_t \) serves as a sort of longer memory.

We also keep local memmory.
We also keep memory $h_t$, governed by the output gate $O_t$.

$$h_t = O_t \circ \text{Tanh}(C_t)$$