

# Recurrent Neural Networks

CS/DS541 2020 – Jacob Whitehill (jrwhitehill@wpi.edu)

## 1 Recurrent Neural Networks

Definition of RNN (for regression):

$$\begin{aligned} J_t(\mathbf{U}, \mathbf{V}, \mathbf{w}) &= \frac{1}{2}(\hat{y}_t - y_t)^2 \\ \hat{y}_t &= \mathbf{h}_t^\top \mathbf{w} \\ \mathbf{h}_0 &= \mathbf{0} \\ \mathbf{h}_t &= \tanh(\mathbf{z}_t) \\ \mathbf{z}_t &= \begin{bmatrix} \mathbf{U} & \mathbf{V} \end{bmatrix} \begin{bmatrix} \mathbf{h}_{t-1} \\ \mathbf{x}_t \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{U}\mathbf{h}_{t-1} + \mathbf{V}\mathbf{x}_t \end{bmatrix} \end{aligned}$$

## 2 Gradient derivation

### 2.1 U

$$\begin{aligned} \frac{\partial J_t}{\partial \text{vec}[\mathbf{U}]} &= \frac{\partial J_t}{\partial \hat{y}_t} \frac{\partial \hat{y}_t}{\partial \text{vec}[\mathbf{U}]} \\ \frac{\partial J_t}{\partial \hat{y}_t} &= \hat{y}_t - y_t \\ \frac{\partial \hat{y}_t}{\partial \text{vec}[\mathbf{U}]} &= \frac{\partial \hat{y}_t}{\partial \mathbf{h}_t} \frac{\partial \mathbf{h}_t}{\partial \text{vec}[\mathbf{U}]} \\ \frac{\partial \hat{y}_t}{\partial \mathbf{h}_t} &= \mathbf{w}^\top \\ \frac{\partial \mathbf{h}_t}{\partial \text{vec}[\mathbf{U}]} &= \frac{\partial \mathbf{h}_t}{\partial \mathbf{z}_t} \frac{\partial \mathbf{z}_t}{\partial \text{vec}[\mathbf{U}]} \\ \frac{\partial \mathbf{h}_t}{\partial \mathbf{z}_t} &= \text{diag}[1 - \tanh^2(\mathbf{z}_t)] \doteq \text{diag}[\mathbf{g}_t^\top] \\ \frac{\partial \mathbf{z}_t}{\partial \text{vec}[\mathbf{U}]} &= \frac{\partial}{\partial \text{vec}[\mathbf{U}]} \left( \begin{bmatrix} \mathbf{U} & \mathbf{V} \end{bmatrix} \begin{bmatrix} \mathbf{h}_{t-1} \\ \mathbf{x}_t \end{bmatrix} \right) \\ &= \begin{bmatrix} \mathbf{h}_{t-1}^\top & \mathbf{0}^\top & \cdots & \mathbf{0}^\top \\ \mathbf{0}^\top & \mathbf{h}_{t-1}^\top & \cdots & \mathbf{0}^\top \\ \vdots & \cdots & \ddots & \vdots \\ \mathbf{0}^\top & \cdots & \mathbf{0}^\top & \mathbf{h}_{t-1}^\top \end{bmatrix} + \mathbf{U} \frac{\partial \mathbf{h}_{t-1}}{\partial \text{vec}[\mathbf{U}]} \end{aligned}$$

Therefore,

$$\begin{aligned}
\frac{\partial \mathbf{h}_t}{\partial \text{vec}[\mathbf{U}]} &= \text{diag}[\mathbf{g}_t^\top] \begin{bmatrix} \mathbf{h}_{t-1}^\top & \mathbf{0}^\top & \dots & \mathbf{0}^\top \\ \mathbf{0}^\top & \mathbf{h}_{t-1}^\top & \dots & \mathbf{0}^\top \\ \vdots & \dots & \ddots & \vdots \\ \mathbf{0}^\top & \dots & \mathbf{0}^\top & \mathbf{h}_{t-1}^\top \end{bmatrix} + \text{diag}[\mathbf{g}_t^\top] \mathbf{U} \frac{\partial \mathbf{h}_{t-1}}{\partial \text{vec}[\mathbf{U}]} \\
&= \begin{bmatrix} (\mathbf{g}_t)_1 \mathbf{h}_{t-1}^\top & \mathbf{0}^\top & \dots & \mathbf{0}^\top \\ \mathbf{0}^\top & (\mathbf{g}_t)_2 \mathbf{h}_{t-1}^\top & \dots & \mathbf{0}^\top \\ \vdots & \dots & \ddots & \vdots \\ \mathbf{0}^\top & \dots & \mathbf{0}^\top & (\mathbf{g}_t)_n \mathbf{h}_{t-1}^\top \end{bmatrix} + \text{diag}[\mathbf{g}_t^\top] \mathbf{U} \frac{\partial \mathbf{h}_{t-1}}{\partial \text{vec}[\mathbf{U}]} \\
&= \mathbf{F}_t + \text{diag}[\mathbf{g}_t^\top] \mathbf{U} \frac{\partial \mathbf{h}_{t-1}}{\partial \text{vec}[\mathbf{U}]}
\end{aligned}$$

Putting it all together,

$$\frac{\partial J_t}{\partial \text{vec}[\mathbf{U}]} = (\hat{y}_t - y_t) \mathbf{w}^\top (\mathbf{F}_t + \text{diag}[\mathbf{g}_t^\top] \mathbf{U} [\mathbf{F}_{t-1} + \text{diag}[\mathbf{g}_{t-1}^\top] \mathbf{U} [\mathbf{F}_{t-2} + \dots + \text{diag}[\mathbf{g}_1^\top] \mathbf{U} [\mathbf{F}_1]]])$$

We can now derive a recursive algorithm as follows:

$$\begin{aligned}
\mathbf{q}_t^\top &= ((\hat{y}_t - y_t) \mathbf{w}^\top) \odot \mathbf{g}_t^\top && \text{final condition} \\
\mathbf{q}_{\tau-1}^\top &= (\mathbf{q}_\tau^\top \mathbf{U}) \odot \mathbf{g}_{\tau-1}^\top && \text{recursion relation} \\
\mathbf{r}_\tau &= \mathbf{q}_\tau \mathbf{h}_{\tau-1}^\top \\
\frac{\partial J_t}{\partial \text{vec}[\mathbf{U}]} &= \text{vec} \left[ \sum_{\tau=1}^t \mathbf{r}_\tau \right]
\end{aligned}$$

## 2.2 V

$$\begin{aligned}
\frac{\partial \mathbf{h}_t}{\partial \text{vec}[\mathbf{V}]} &= \text{diag}[\mathbf{g}_t^\top] \frac{\partial}{\partial \text{vec}[\mathbf{V}]} \left( \begin{bmatrix} \mathbf{U} & \mathbf{V} \end{bmatrix} \begin{bmatrix} \mathbf{h}_{t-1} \\ \mathbf{x}_t \end{bmatrix} \right) \\
&= \begin{bmatrix} (\mathbf{g}_t)_1 \mathbf{x}_t^\top & \mathbf{0}^\top & \dots & \mathbf{0}^\top \\ \mathbf{0}^\top & (\mathbf{g}_t)_2 \mathbf{x}_t^\top & \dots & \mathbf{0}^\top \\ \vdots & \dots & \ddots & \vdots \\ \mathbf{0}^\top & \dots & \mathbf{0}^\top & (\mathbf{g}_t)_n \mathbf{x}_t^\top \end{bmatrix} + \text{diag}[\mathbf{g}_t^\top] \mathbf{U} \frac{\partial \mathbf{h}_{t-1}}{\partial \text{vec}[\mathbf{V}]} \\
&= \mathbf{E}_t + \text{diag}[\mathbf{g}_t^\top] \mathbf{U} \frac{\partial \mathbf{h}_{t-1}}{\partial \text{vec}[\mathbf{V}]}
\end{aligned}$$

Analogously to the derivative w.r.t.  $\mathbf{U}$ , we find:

$$\frac{\partial J_t}{\partial \text{vec}[\mathbf{V}]} = (\hat{y}_t - y_t) \mathbf{w}^\top (\mathbf{E}_t + \text{diag}[\mathbf{g}_t^\top] \mathbf{U} [\mathbf{E}_{t-1} + \text{diag}[\mathbf{g}_{t-1}^\top] \mathbf{U} [\mathbf{E}_{t-2} + \dots + \text{diag}[\mathbf{g}_1^\top] \mathbf{U} [\mathbf{E}_1]]])$$

The recursive algorithm is similar as for  $\mathbf{U}$ .

## 2.3 w

$$\frac{\partial J_t}{\partial \mathbf{w}} = (\hat{y}_t - y_t) \mathbf{h}_t^\top$$