CS540 Introduction to Artificial Intelligence
Lecture 12

Young Wu
Based on lecture slides by Jerry Zhu, Yingyu Liang, and Charles Dyer

July 8, 2021

Practice Exam

Admin

- Grades updated.
- Game results posted.
- M14 Q12, 13, 14 will be on both versions of the midterm (same question with different randomization).
- M3 Q9 will be on the midterm (same question with different randomization).
- A modified version of M4 Q9 will be on the midterm (the question will be "add a point so that all points are support vectors").
- Did I forget something?

Dyer's 2019, 2017 posted on Piazza.

Homework

Admin

- Please do not submit M8 to M11.
- If you haven't started P1: you should.
- P2 solutions are posted.
- If you use another student's code (or find code on the Internet), you must give attribution at the beginning of your code.
- You are not allowed use another student's output.

Remind Me to Start Recording

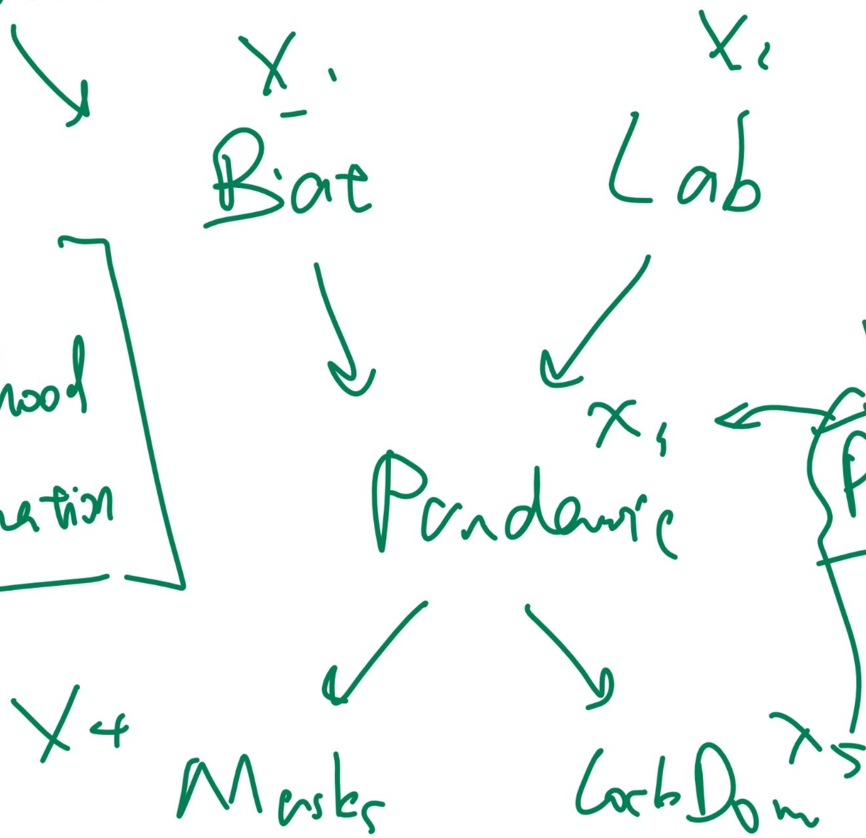
Admin

- The messages you send in chat will be recorded: you can change your Zoom name now before I start recording.

Bayes Net Inference Example 1

Quiz

feature



Max Likelihood Estimation

CPT → Conditional probability Table

$\Pr\{X_j \mid \text{parents}(X_j)\}$

$\Pr\{X_3 \mid X_1 = 0, X_2 = 0\}$

$= 0$

$C_{x,0}$

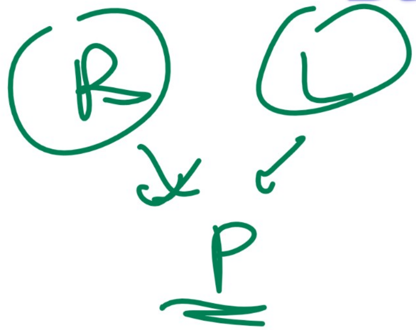
$C_{1,x}$

$\Pr\{X_1, X_2 \mid X_4 = 1, X_5 = 0\}$

$\varphi = 0, 1, 2, 3$

Bayes Net Inference Example 2

Quiz



- Compute $\hat{\mathbb{P}}\left\{B = 1 | L = 1\right\}$?

$$= \frac{\Pr\{B=1, L=1\}}{\Pr\{L=1\}} = \frac{\Pr\{B=1\}\cancel{\Pr\{L=1\}}}{\cancel{\Pr\{L=1\}}} = \Pr\{B=1\}$$

$$\boxed{\hat{\mathbb{P}}\left\{B = 1\right\} = 0.001}, \hat{\mathbb{P}}\left\{L = 1\right\} = 0.001$$

$$\hat{\mathbb{P}}\left\{P = 1 | B = 1, L = 1\right\} = 0.95, \hat{\mathbb{P}}\left\{P = 1 | B = 1, L = 0\right\} = 0.29$$

$$\hat{\mathbb{P}}\left\{P = 1 | B = 0, L = 1\right\} = 0.94, \hat{\mathbb{P}}\left\{P = 1 | B = 0, L = 0\right\} = 0.00$$

- A: 0, B: 0.001, C: 0.0094, D: 0.0095, E: 1

Bayes Net Inference Example 3

Quiz

B L

P

- Compute $\hat{\mathbb{P}}\{B = 1, L = 1 | P = 1\}$?

$$\frac{P_r\{B=1, L=1, P=1\}}{P_r\{P=1\}}$$

CPT

$$\hat{\mathbb{P}}\{B = 1\} = 0.001, \hat{\mathbb{P}}\{L = 1\} = 0.001$$

$$\hat{\mathbb{P}}\{P = 1 | B = 1, L = 1\} = 0.95, \hat{\mathbb{P}}\{P = 1 | B = 1, L = 0\} = 0.29$$

$$\hat{\mathbb{P}}\{P = 1 | B = 0, L = 1\} = 0.94, \hat{\mathbb{P}}\{P = 1 | B = 0, L = 0\} = 0.00$$

$$P_r\{B=1\} \, P_r\{L=1\} \cdot P_r\{P=1 | B=1, L=1\}$$

$$0.001 \quad 0.001 \quad 0.95$$

- A: $0.001 \cdot 0.001$, B: $0.001 \cdot 0.001 \cdot 0.95$,
- C: $\dfrac{0.001}{0.001 \cdot 0.95 + 0.999 \cdot (0.94 + 0.29)}$
- D : $\dfrac{0.001 \cdot 0.001}{0.001 \cdot 0.95 + 0.999 \cdot (0.94 + 0.29)}$
- E : $\dfrac{0.001 \cdot 0.95 \cdot 0.001}{0.001 \cdot 0.95 + 0.999 \cdot (0.94 + 0.29) \cdot 0.001}$

$\Pr\{P=1|B=0,L=0\} \cdot \Pr\{B=0\} \cdot \Pr\{L=0\}$

Bayes Net Inference Example 3 Computation

Quiz

$\Pr\{P=1\} = \Pr\{P=1, B=0, L=0\} +$
$P=1, B=0, L=1 +$
$P=1, B=1, L=0 +$
$P=1, B=1, L=1$

$0 \cdot 0.999 \cdot 0.999$
$0.94 \cdot 0.999 \cdot 0.001$
$0.29 \cdot 0.001 \cdot 0.999$
$0.95 \cdot 0.001 \cdot 0.001$

- Compute $\hat{\mathbb{P}}\{B=1, L=1|P=1\}$?

$$\hat{\mathbb{P}}\{B=1\} = 0.001, \hat{\mathbb{P}}\{L=1\} = 0.001$$

$$\hat{\mathbb{P}}\{P=1|B=1,L=1\} = 0.95, \hat{\mathbb{P}}\{P=1|B=1,L=0\} = 0.29$$

$$\hat{\mathbb{P}}\{P=1|B=0,L=1\} = 0.94, \hat{\mathbb{P}}\{P=1|B=0,L=0\} = 0.00$$

Types of Bayes Net Components

Discussion

- Causal Chain
- Common Cause
- Common Effect

$$X_1 \to X_2 \to X_3$$

$$\begin{tikzcd} & X_1 \arrow[dl] \arrow[dr] & \\ X_2 & & X_3 \end{tikzcd}$$

$$\begin{tikzcd} X_1 \arrow[dr] & & X_2 \arrow[dl] \\ & X_3 & \end{tikzcd}$$

Network Structure

Discussion

- Selecting from all possible structures (DAGs) is too difficult.
- Usually, a Bayesian network is learned with a tree structure.
- Choose the tree that maximizes the likelihood of the training data.

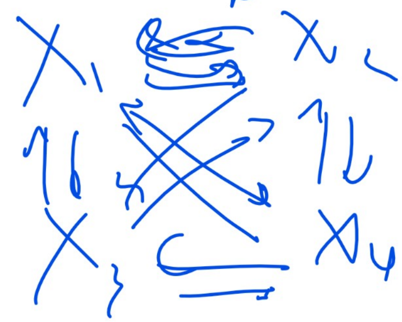
mutual info

Chow Liu Algorithm

Discussion

InfoGain$(X_1|X_2)$

$(X_2|X_1)$



- Add an edge between features X_j and $X_{j'}$ with edge weight equal to the information gain of X_j given $X_{j'}$ for all pairs j, j'.
- Find the maximum spanning tree given these edges. The spanning tree is used as the structure of the Bayesian network.

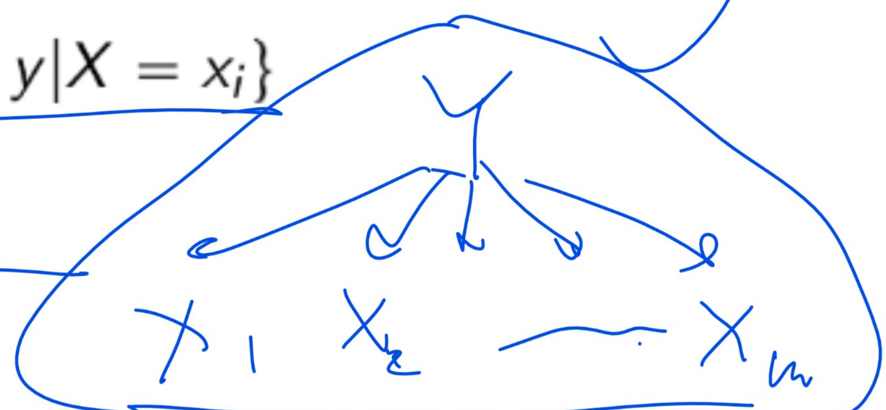
Classification Problem

Discussion

- Bayesian networks do not have a clear separation of the label Y and the features $X_1, X_2, ..., X_m$.
- The Bayesian network with a tree structure and Y as the root and $X_1, X_2, ..., X_m$ as the leaves is called the Naive Bayes classifier.
- Bayes rules is used to compute $\mathbb{P}\{Y = y|X = x\}$, and the prediction $\hat{y}$ is y that maximizes the conditional probability.

$$\hat{y}_i = \arg\max_{y} \mathbb{P}\{Y = y|X = x_i\}$$

$$\text{CPT}\begin{cases} \Pr\{X_1 \mid Y\} & \text{from} \\ \Pr\{X_2 \mid Y\} & \text{training} \end{cases}$$



Naive Bayes Diagram

Discussion

$$\Pr\{Y \mid X_1, X_2\} = \frac{\Pr\{Y, X_1, X_2\}}{\Pr\{X_1, X_2\}}$$

$$= \frac{\Pr\{Y\} \cdot \Pr\{X_1 \mid Y\} \cdot \Pr\{X_2 \mid Y\}}{\sum_y \Pr\{X_1, X_2, Y = y\}}$$

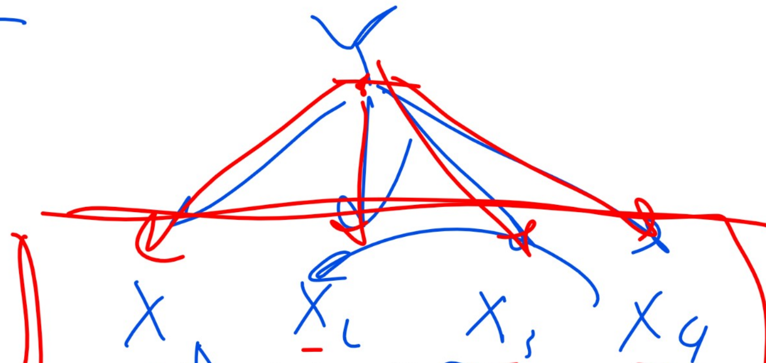
$$\max_y \Pr\{Y = y \mid X_1, \bar{X_2}\}$$

$0, 1, 2 \ldots k$

Tree Augmented Network Algorithm

Discussion

Max Spanning Tree



- It is also possible to create a Bayesian network with all features $X_1, X_2, ..., X_m$ connected to Y (Naive Bayes edges) and the features themselves form a network, usually a tree (MST edges).
- Information gain is replaced by conditional information gain (conditional on Y) when finding the maximum spanning tree.
- This algorithm is called TAN: Tree Augmented Network.

Tree Augmented Network Algorithm Diagram

Discussion

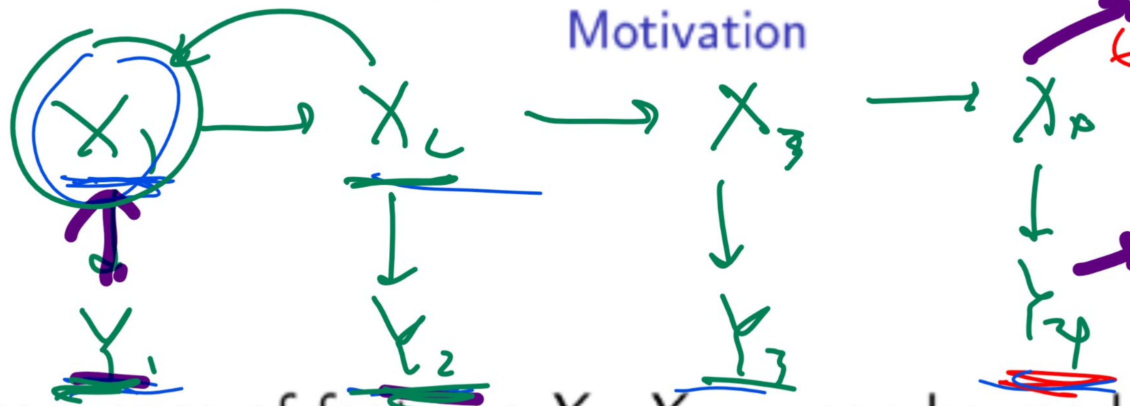
Special Bayesian Network for Sequences

info about company

words

Motivation

context



stock price

words

sound

- A sequence of features $X_1, X_2, \ldots$ can be modeled by a Markov Chain but they are not observable.
- A sequence of labels $Y_1, Y_2, \ldots$ depends only on the current hidden features and they are observable.
- This type of Bayesian Network is called a Hidden Markov Model.

→ recognize speech

→ wreck a nice beach

Y_1 Y_2

Hidden Markov Model Diagram

Motivation

Evaluation and Training

Motivation

- There are three main tasks associated with an HMM.

1. Evaluation problem: finding the probability of an observed sequence given an HMM: $y_1, y_2, \ldots$
2. Decoding problem: finding the most probable hidden sequence given the observed sequence: $x_1, x_2, \ldots$
3. Learning problem: finding the most probable HMM given an observed sequence: $\pi, A, B, \ldots$

CPT

$Pr\{X_1\}$

$Pr\{X_2 | X_1\}$

$Pr\{Y_1 | X_1\}$

Expectation-Maximization Algorithm

Description

- Start with a random guess of π, A, B.
- Compute the forward probabilities: the joint probability of an observed sequence and its hidden state.
- Compute the backward probabilities: the probability of an observed sequence given its hidden state.
- Update the model π, A, B using Bayes rule.
- Repeat until convergence.
- Sometimes, it is called the Baum-Welch Algorithm.

Hidden Markov Model Example 1

Definition

on Friday

- Compute $\mathbb{P}\{X_4 = 1, X_5 = 2 | X_3 = 0\}$.

Hidden Markov Model Example 1 Computations

Definition

Hidden Markov Model Example 2

Definition

- Compute $\mathbb{P}\{Y_1 = 0, Y_2 = 1\}$.

Hidden Markov Model Example 2 Computations

Definition

Hidden Markov Model Example 3

Definition

- Compute $\mathbb{P}\left\{X_1 = 0, X_2 = 2 \middle| Y_1 = 0, Y_2 = 1\right\}$.

Hidden Markov Model Example 3 Computations

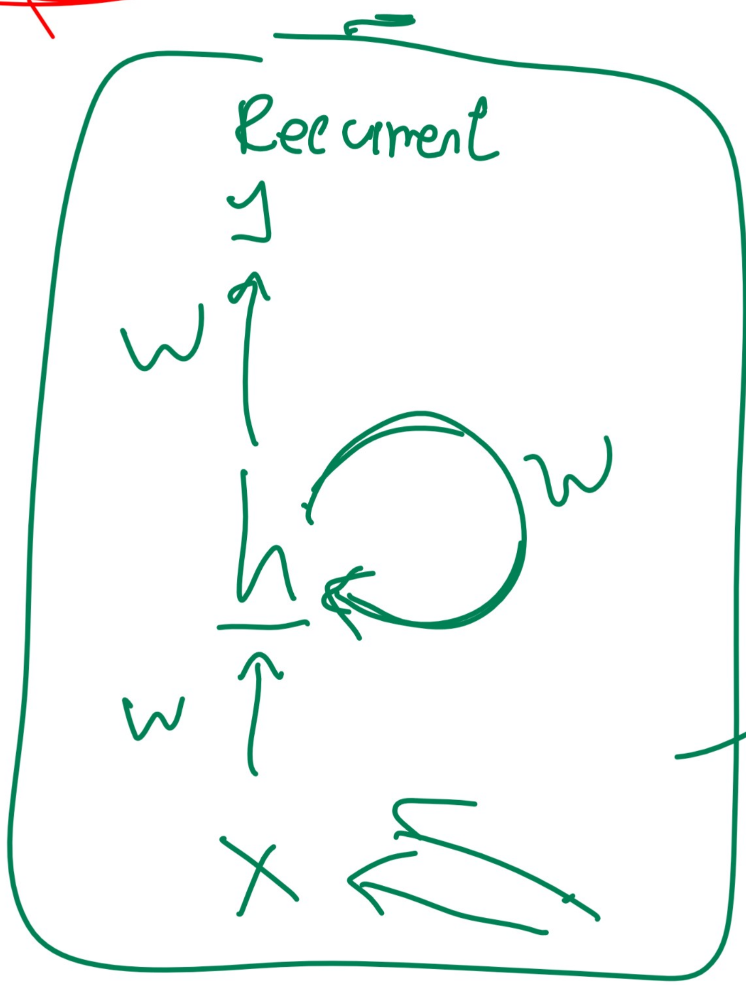
Definition

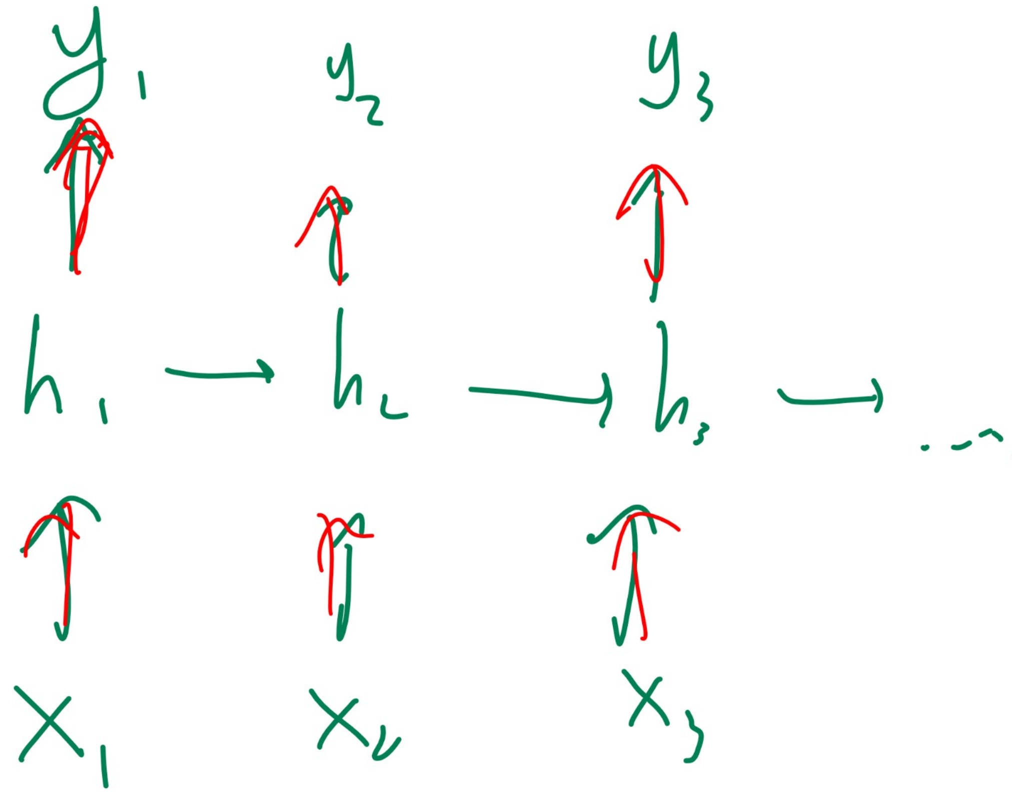
Dynamic System Diagram

Motivation

image → CNN

sequences → RNN





Recurrent Neural Network Structure Diagram

Motivation

Activation Functions

Definition

- The hidden layer activation function can be the tanh activation, and the output layer activation function can be the softmax function.

$$
\begin{aligned}
z_t^{(x)} &= W^{(x)} x_t + W^{(h)} a_{t-1}^{(x)} + b^{(x)} \\
a_t^{(x)} &= g\left(z_t^{(x)}\right), g\left(\boxed{\cdot}\right) = \tanh\left(\boxed{\cdot}\right) \\
z_t^{(y)} &= W^{(y)} a_t^{(x)} + b^{(y)} \\
a_t^{(y)} &= g\left(z_t^{(y)}\right), g\left(\boxed{\cdot}\right) = \text{softmax}\left(\boxed{\cdot}\right)
\end{aligned}
$$

Cost Functions

Definition

- Cross entropy loss is used with softmax activation as usual.

$$
\begin{cases}
C_t = H\left(y_t, a_t^{(y)}\right) \\
C = \sum_t C_t
\end{cases}
$$

$$\frac{\partial C}{\partial w^{(x)}}$$

BackPropogation Through Time

Definition

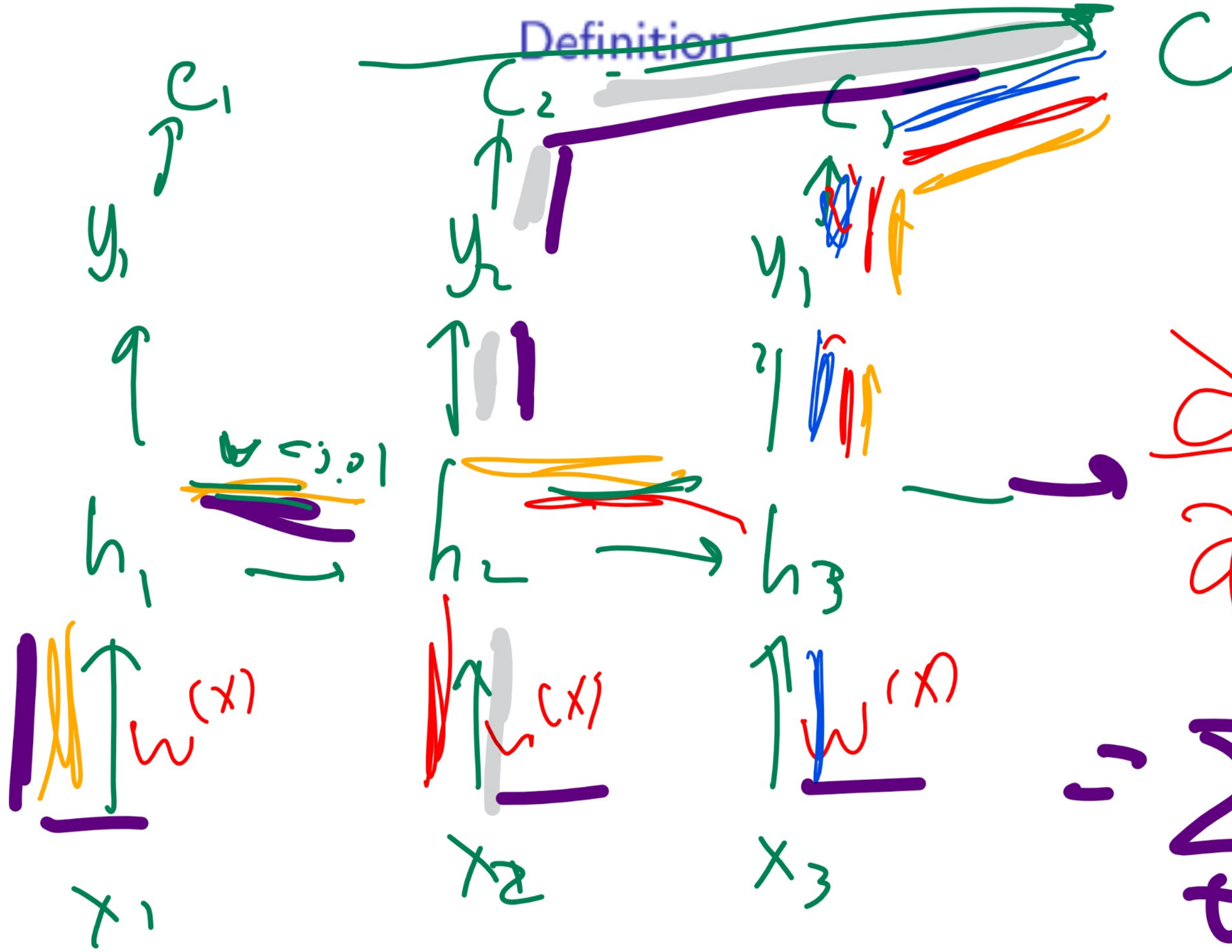
- The gradient descent algorithm for recurrent neural networks is called BackPropogation Through Time (BPTT). The update procedure is the same as standard neural networks using the chain rule.

$$w = w - \alpha \frac{\partial C}{\partial w}$$

$$b = b - \alpha \frac{\partial C}{\partial b}$$

Unfolded Network Diagram

Definition



$$\frac{\partial C}{\partial w_j^{(x)}} = \sum_t \sum_{t'=1}^{t}$$

100   100

$$h_1 \xrightarrow{0.01} h_2 \xrightarrow{0.01} h_3 \xrightarrow{0.01} \cdot \; h_n$$

$$\left[\begin{array}{l} 0.01^n \approx 0 \\ 100^n \approx \infty \end{array} \right]$$

Vanishing and Exploding Gradient

Discussion

- If the weights are small, the gradient through many layers will shrink exponentially. This is called the vanishing gradient problem.
- If the weights are large, the gradient through many layers will grow exponentially. This is called the exploding gradient problem.
- Fully connected and convolutional neural networks only have a few hidden layers, so vanishing and exploding gradient is not a problem in training those networks.
- In a recurrent neural network, if the sequences are long, the gradients can easily vanish or explode.

Transformer

→ LSTM GRU ←