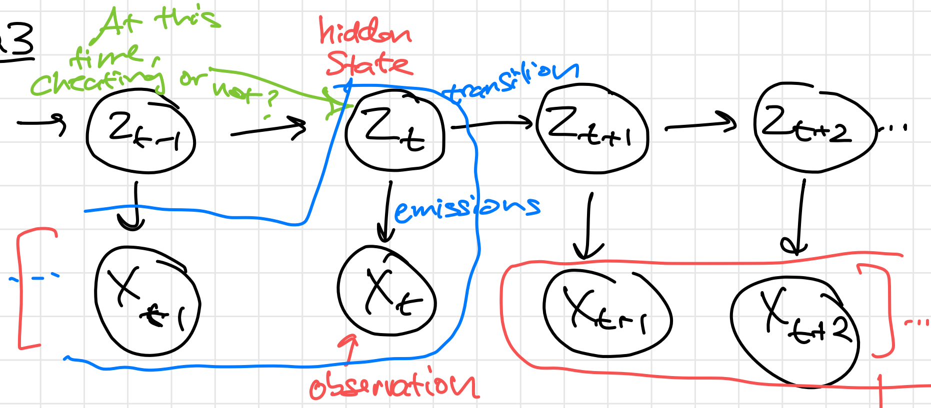
3/30/2023

Inference on z_t



$$p(z_t \mid x_{1:T}) \propto p(z_t, x_{1:T}) = \underbrace{\boxed{p(x_{1:t}, z_t)}}_{\alpha_t(z_t)} \underbrace{\boxed{p(x_{t+1:T} \mid z_t)}}_{\beta_t(z_t)}$$

"proportional to"

Compute for every value of z_t then normalize by the sum

$\alpha_t(z_t)$ "forward"

$\beta_t(z_t)$ "backward"

α-Recursion: Suppose we know $\alpha_{t-1}(i) = p(x_{1:t-1}, z_{t-1} = i)$ for every i

What is $\alpha_t(j)$ for all j

marginalize out z_{t-1}

$$\alpha_t(j) = \sum_{i=1}^{K} p(x_{1:t-1}, z_{t-1} = i)\, p(z_t = j \mid z_{t-1} = i)\, p(x_t \mid z_t = j)$$

$$= \sum_{i=1}^{K} \alpha_{t-1}(i) \cdot A_{ij} \cdot p(x_t \mid z_t = j)$$

Base case: $\alpha_1(j) = p(x_1, z_1 = j) = \underbrace{p(z_1 = j)}_{\text{prior}} \underbrace{p(x_1 \mid z_1 = j)}_{\text{emission}}$

For $\beta_t(j)$: compute it based on $\beta_{t+1}(i)$ for every i

All together: To infer $p(z_t \mid x_{1:T})$

1. Compute α_t's and β_t's recursively
2. Compute $p(z_t = j \mid x_{1:T}) = \dfrac{\alpha_t(j)\,\beta_t(j)}{\sum_{i=1}^{k} \alpha_t(i)\,\beta_t(i)}$

Learning HMM parameters

Again EM

How to compute good parameters?

Idea of EM: If we had complete data, our lives are easy

Z:	2	1	3	1	2
X:	1.6	6.1	2.2	9.2	1.4

] Sequence 1

Z:	1	3	2	1
X:	7.4	2.1	1.2	9.7

] Sequence 2

$$p(z_1) = \begin{cases} 1/2 & \text{if } z_1 = 1 \\ 1/2 & \text{if } z_1 = 2 \\ 0 & \text{if } z_1 = 3 \end{cases}$$

$p(x_t \mid z_t = 1)$ = Normal with $\mu = 8$, $\sigma^2 = 6$

$p(x_t \mid z_t = 2)$ = Normal with μ of 1.4, $\sigma^2 = 0.1$

$$p(z_t \mid z_{t-1} = 1) = \begin{cases} 0 & \text{if } z_t = 1 \\ 1/3 & \text{if } z_t = 2 \\ 2/3 & \text{if } z_t = 3 \end{cases}$$

Now: Suppose we don't know z_t's

E-step creates fictitious data (w/ smart guessing)

M-step estimates params on fictitious data

$p(z_1)$:

E-step: Compute $p(z_1 \mid x_{1:T})$ "pseudocounts"

Suppose you get:

Sequence 1: $[0.2,\ 0.7,\ 0.1]$

Sequence 2: $[0.6,\ 0.3,\ 0.1]$

Pretend our data looks like

10 copies of Sequence 1 where
- 2 have $z_1 = 1$
- 7 have $z_1 = 2$
- 1 has $z_1 = 3$

10 copies of Sequence 2 where
- 6 have $z_1 = 1$
- 3 have $z_1 = 2$
- 1 has $z_1 = 3$

<u>M-step</u> (Estimate parameters)

Just treat this fictitious data as real & count things to estimate parameters

$$p(z_1 = 1) = 8/20$$
$$p(z_1 = 2) = 10/20$$
$$p(z_1 = 3) = 2/20$$

<u>For emissions</u>!

<u>E-step</u>: Infer $p(z_t \mid x_{1:T})$ for every t

Suppose for some t, $x_t = 1.7$

$$p(z_t \mid x_{1:T}) = [.7, \ .1, \ .2]$$

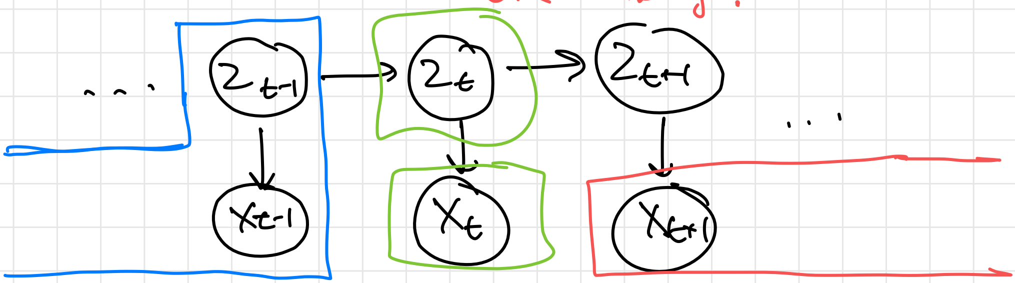
Then we have
- .7 "counts" of $(z_1 = 1, x_t = 1.7)$
- .1 "counts" of $(z_1 = 2, x_t = 1.7)$
- .2 "counts" of $(z_1 = 3, x_t = 1.7)$

Transitions:

E-step: We want pseudocounts of how many times state $i \to$ state j

$$p(z_{t-1}, z_t \mid x_{1:T})$$

Based on observations, which pairs (z_{t-1}, z_t) are likely?



$$p(z_{t-1}, z_t, x_{1:T}) = \underbrace{p(x_{1:t-1}, z_{t-1})}_{= \alpha_{t-1}(z_{t-1})} \cdot \underbrace{p(x_{t+1:T} \mid z_t)}_{= \beta_t(z_t)} \cdot p(z_t \mid z_{t-1})\, p(x_t \mid z_t)$$

Then normalize over all pairs of (z_{t-1}, z_t)

M-step: Use these pseudocounts to estimate transition probabilities

---

Announcements

- HW3 starter code
- Section: Neural network optimizers

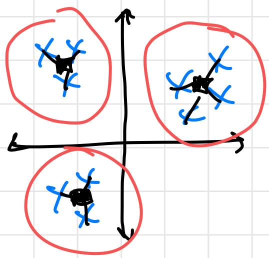
Variants of SGD

$$p(z_t, x_{1:T}) = p(\overbrace{z_t, x_{1:t}}^{A}, \overbrace{x_{t+1:T}}^{B})$$

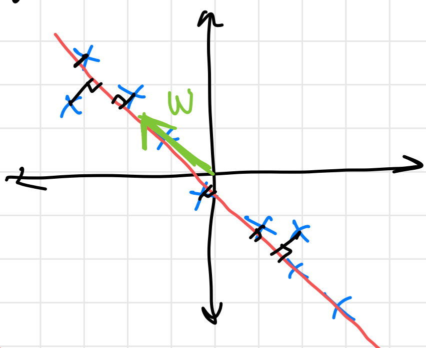
$$p(\underbrace{z_t, x_{1:t}}_{A}) \cdot p(\underbrace{x_{t+1:T}}_{B} \mid \underbrace{z_t, x_{1:t}}_{A})$$

$$= p(z_t, x_{1:t}) \cdot p(x_{t+1:T} \mid z_t)$$

Dimensionality Reduction



Clustering



points in $\mathbb{R}^2$ but a single line captures all of information

Dimensionality Reduction
Given $\{x^{(1)}, \ldots, x^{(n)}\} \in \mathbb{R}^d$
Find a lower-dimensional subspace that preserves most of the information

Method: Principal Component Analysis (PCA)

For now: want to find a good 1-D projection

Key assumption: Data has mean 0

↳ In practice: compute mean of the data & subtract it away

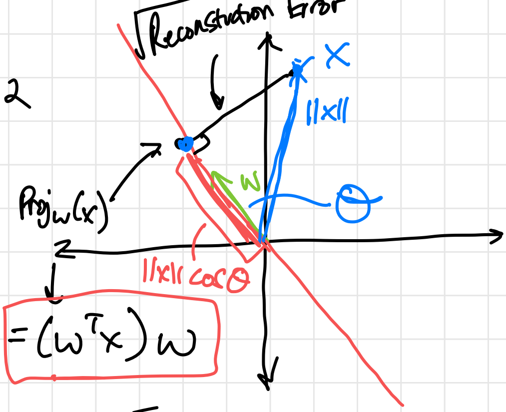
Parameter $w \in \mathbb{R}^d$ that defines the 1-D subspace
$\|w\| = 1$

What loss function describes a "good" choice of w?

<u>Reconstruction Error</u>

$$\sum_{i=1}^{n} \| x^{(i)} - \underbrace{\mathrm{Proj}_w}_{\text{projection onto } w} (x^{(i)}) \|^2$$

$$= \sum_{i=1}^{n} \| x^{(i)} - (w^T x^{(i)}) \cdot w \|^2$$

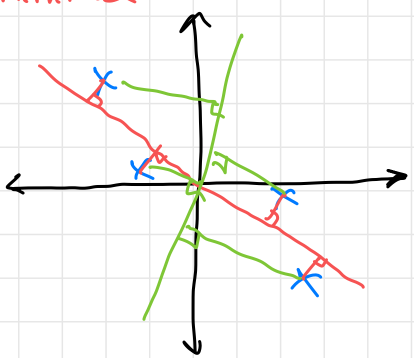


$$\boxed{= (w^T x) w}$$

$$\cos\theta = \frac{w^T x}{\|x\| \, \|w\|}$$

$$\|x\| \cos\theta = \frac{w^T x}{\cancel{\|w\|}_{1}} = w^T x$$

PCA chooses w to minimize this loss function



Small recon. error = good
Large recon error = bad

<u>Equivalent View</u>:

Maximize variance of points after projection

By Pythagorean Theorem:

$$\underbrace{(w^T x)^2}_{\text{maximize this}} + \underbrace{\text{ReconError}}_{\text{want to minimize}} = \underbrace{\|x\|^2}_{\text{Fixed}}$$

maximize this $\Longleftrightarrow$ want to minimize

$$\text{maximize} \quad \sum_{i=1}^{n} (w^T x^{(i)})^2$$

$\frac{1}{n} \sum_{i=1}^{n} (w^T x^{(i)})^2$ is just variance of $w^T x$