# Hidden Markov models

COS 424/524, SML 302: Fundamentals of Machine Learning
Professor Engelhardt

COS 424/524, SML 302

Lecture 15

# Hidden Markov Models

In our last few lectures, we learned about mixture models to cluster and understand latent structure in data.

We also learned how to fit these models using the expectation-maximization (EM) algorithm.

Today we will combine mixture models with Markov chains to build hidden Markov models (HMMs) to model sequential data with a hidden state.

# Hidden Markov Models

## Examples of sequential data with a latent state

- Weather information across many years
- Mutations across the genome
- Words in a sentence
- Speech recording from a meeting (automatic speaker recognition)
- Musical scores
- Motion categorization
- Pixels in a video
- Financial forecasting

# Latent variable models: notation review

Recall $\mathcal{Z} = \{z_1, \ldots, z_n\}$ represents $n$ *latent variables* for observed data $\mathcal{D} = \{x_1, \ldots, x_n\}$ without labels.

We write $z_i$ as a *multinomial vector*, or a $K$-vector with all zeros except for a single one at the $k$th position.

The vector $z_i^k$ is a binary indicator for whether or not sample $i$ is in state $k \in \{1, \ldots, K\}$.

Extend each sample to a sequence of observations. Each sample will have a corresponding sequence of latent variables.

# Running example: Weather in Princeton in April

## Example: weather in Princeton in April

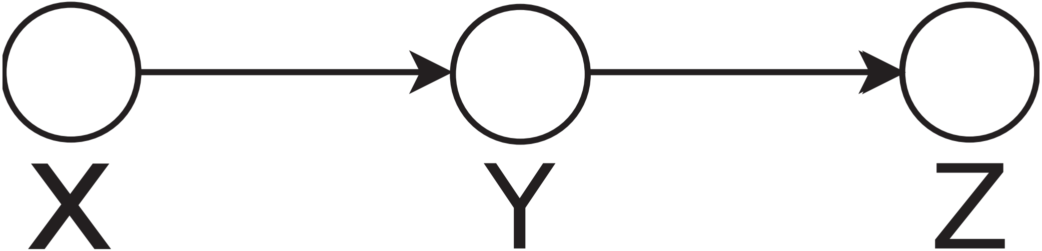
I downloaded from `http://www.wunderground.com/history` 20 years of historical weather data for Princeton in April.

- There are $n = 20$ samples, corresponding to 20 Aprils' worth of data
- There are $p = 6$ features, corresponding to high temp, low temp, average wind speed, humidity, cloud cover, and precipitation.
- There are $T = 30$ observed values for each sample, because each sample corresponds to 30 days of weather information.
- We will estimate $K$ latent states at each sequential point of the data.

What do these latent states represent?

# The Markov property

The idea behind *hidden Markov models* is that sequential data may be sequentially clustered into $K$ states using latent variable $Z$, and the variables $Z$ have a simple Markovian structure.

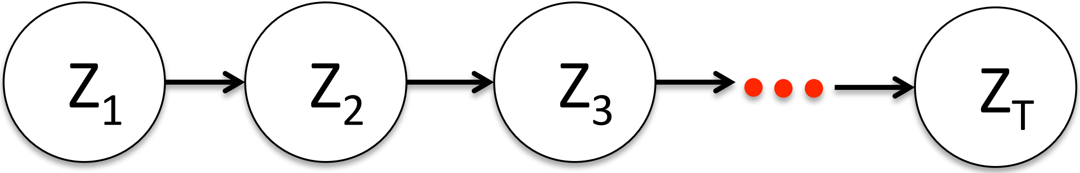


## Markov Property

The (first order) *Markov property* states that the future is conditionally independent of the past given the present:

$$X \perp\!\!\!\perp Z \mid Y.$$

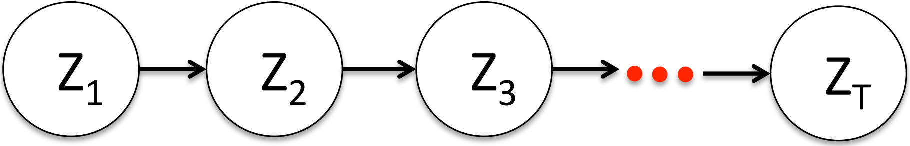
# Implications of the Markov property



Given a chain graph for variables $z_t$, $t = 1 : T$, capturing the Markov property, we can factorize the joint distribution as follows:

$$\begin{aligned} p(z_1, \ldots, z_T) &= p(z_1)p(z_2 \mid z_1)p(z_3 \mid z_1, z_2) \ldots p(z_T \mid z_1, \ldots, z_{T-1}) \\ &= p(z_1)p(z_2 \mid z_1)p(z_3 \mid z_2) \ldots p(z_T \mid z_{T-1}) \\ &= p(z_1) \prod_{t=1}^{T-1} p(z_{t+1} \mid z_t) \end{aligned}$$

# Stationarity assumption



## Stationarity

*stationary* or *time-homogeneous* processes have the following property:

$$p(Z_{t+1} = a \mid Z_t = b) = p(Z_t = a \mid Z_{t-1} = b)$$

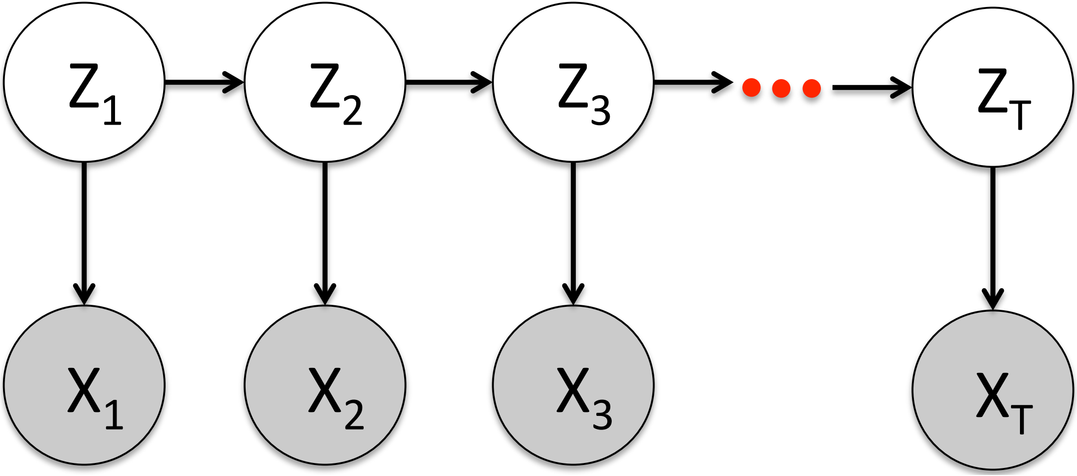
$\forall t \in \{2, \ldots, T-1\}$.

If we assume that the Markov chain is *stationary*, the transition probability is a single conditional probability table (CPT) plus the initial state:

$$p(z_1, \ldots, z_T) \quad = \quad p(z_1) \prod_{t=1}^{T-1} p(z_{t+1} \mid z_t)$$

# Hidden Markov model

Let's build a simple HMM by connecting the latent variables in a mixture model with a Markov chain.



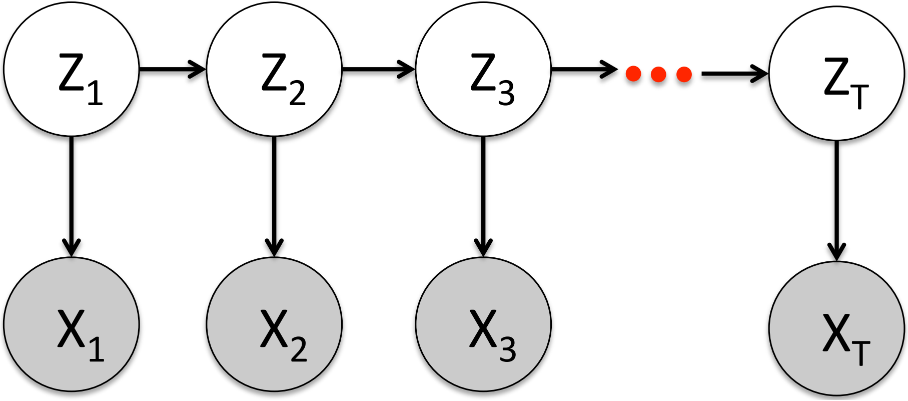
# Hidden Markov model assumptions

## Let’s make the following assumptions about our HMM

- Discrete time: $T$ time points represent discretized intervals on a sequence.
- Discrete state: $z_t \sim Mult(\theta)$.
- First-order Markov: $p(z_{t+1} \mid z_t, z_{t-1}) = p(z_{t+1} \mid z_t)$
- Stationarity: $p(Z_{t+1} = a \mid Z_t = b) = p(Z_t = a \mid Z_{t-1} = b)$

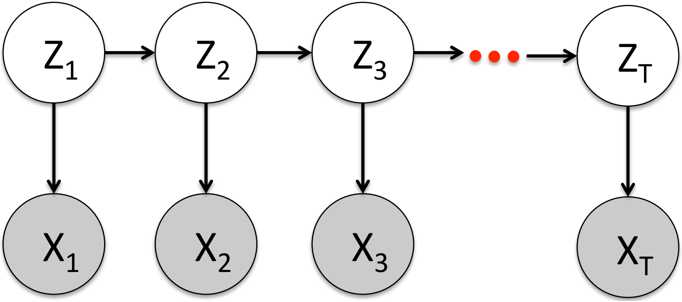
# Hidden Markov model: graphical model

Let's look at how the joint probability of an HMM factorizes:



$$p(z_1, \ldots, z_T, x_1, \ldots, x_T) \quad = \quad p(z_1) \left[ \prod_{t=2}^{T} p(z_t \mid z_{t-1}) \right] \left[ \prod_{t=1}^{T} p(x_t | z_t) \right]$$

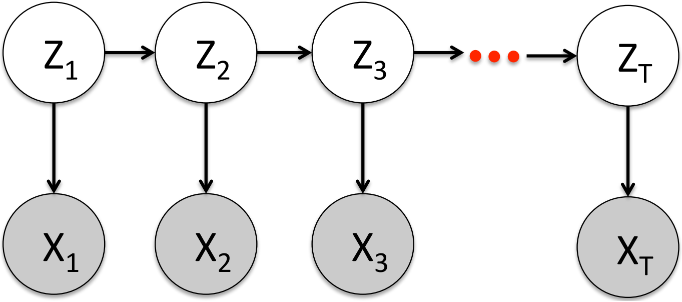
# Hidden Markov model: definitions



$$p(z_1, \ldots, z_T, x_1, \ldots, x_T) \quad = \quad p(z_1) \left[ \prod_{t=2}^{T} p(z_t \mid z_{t-1}) \right] \left[ \prod_{t=1}^{T} p(x_t | z_t) \right]$$

- *Initial state distribution*: the probability of the first state $z_1$, generally denoted $\pi_k = p(z_1 = k)$, or $z_1 \sim Mult(\pi)$.

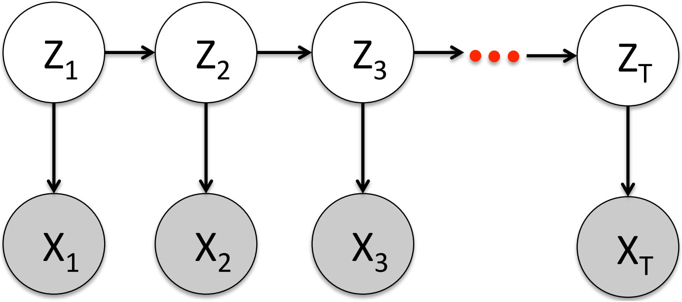
# Hidden Markov model: definitions



$$p(z_1, \ldots, z_T, x_1, \ldots, x_T) \quad = \quad p(z_1) \left[ \prod_{t=2}^{T} p(z_t \mid z_{t-1}) \right] \left[ \prod_{t=1}^{T} p(x_t | z_t) \right]$$

- *Transition probabilities* the probability of the next state $z_t$ given the current state $z_{t-1}$
- Transition probabilities are in a CPT called the transition matrix, $A$, where $A_{j,k} = p(z_t^k \mid z_{t-1}^j)$.
- Stationarity means one state transition matrix $A$ for the chain.

# Hidden Markov model: definitions



$$p(z_1, \ldots, z_T, x_1, \ldots, x_T) \quad = \quad p(z_1) \left[ \prod_{t=2}^{T} p(z_t \mid z_{t-1}) \right] \left[ \prod_{t=1}^{T} p(x_t | z_t) \right]$$

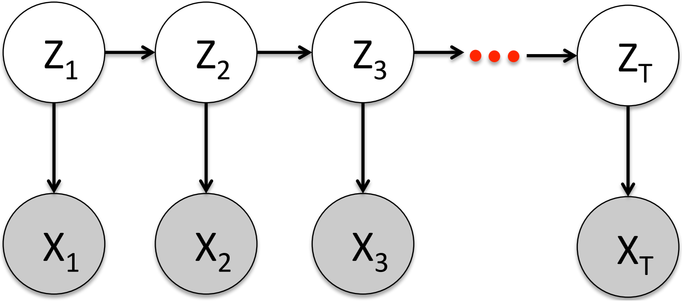
- *Emissions probabilities* define distribution of $x_t$ given latent state $z_t$:

$$p(x_t \mid z_t, \theta) = \eta.$$

- E.g., $x_t$ has multivariate normal given $z_t$, and $\eta = \mathcal{N}(x_t | \mu_k, \Sigma_k)$.

# What can you do with HMMs? Filtering

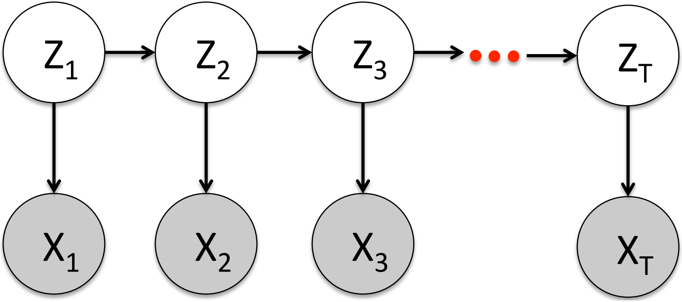
- *Filtering*: compute a belief state $z_t$ given observations up to and including time $t$: $p(z_t|x_{1:t})$



- *Example*: given weather measurements to now, what state are we in?
- *Example*: who is speaking on the audio recording?
- Markov property does not let us look at fewer than all of our observations here. Why?

# What can you do with HMMs? Smoothing

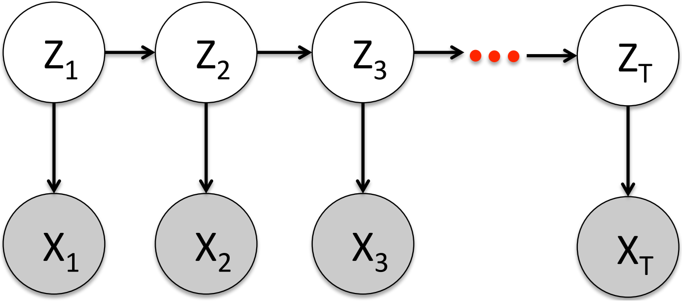
- *Smoothing*: compute a belief state $z_t$ given observations up to and including future time $T$: $p(z_t | x_{1:T})$



- *Example*: given weather data to now, what state were we in on the first day of class?
- *Example*: who is speaking at time 3:42 on the audio recording?
- Markov property does not let us look at fewer than all of our observations here. Why?

# What can you do with HMMs? Prediction

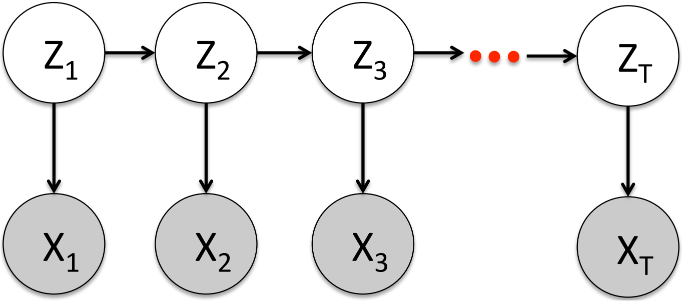
- *Prediction*: predict a future state given past observations: $p(z_{t+h}|x_{1:t})$ for $h \in \mathcal{Z}^+$.



- *Example*: given weather data to now, what state will we be in on the last day of class?
- *Example*: who will be speaking two minutes after the end of the recording?
- Markov property does not let us look at fewer than all of our observations here. Why?

# What can you do with HMMs? Viterbi decoding

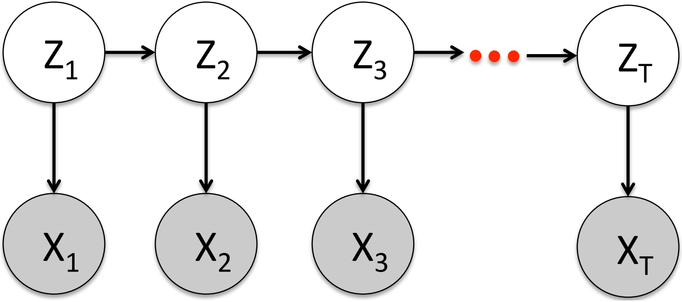
- *Viterbi decoding* (or MAP estimation): label the most likely state sequence for a new observed sequence $x^*$



- *Example*: given weather measurements this past month, label the most likely states.
- *Example*: label the most likely speakers for a new audio recording.

# What can you do with HMMs? Posterior sampling

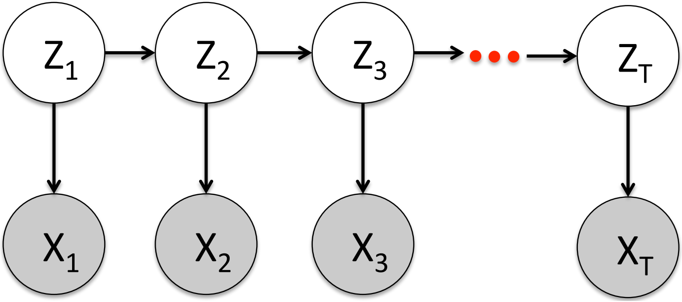
- *Posterior sampling*: draw a random state sequence conditioned on observed sequence $x_{1:T}$ (sample from the posterior distribution $p(z_{1:T} \mid x_{1:T})$)



- *Example*: given weather measurements this past month, draw a set of possible states.
- *Example*: label the speakers for a new audio recording.
- Why is this useful? Consider the problem of local optima.

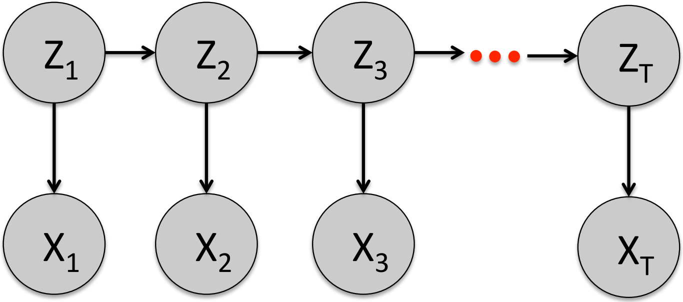
# What can you do with HMMs? Anomaly detection

- *Probability of evidence*: sum over all possible sequences of $z_{1:T}$ to compute $p(x_{1:T})$



- *Example*: how likely was the weather pattern for the last month?
- *Example*: how likely was the audio recording?
- Low probability sequences may indicate one or more low probability observations or low probability transitions.

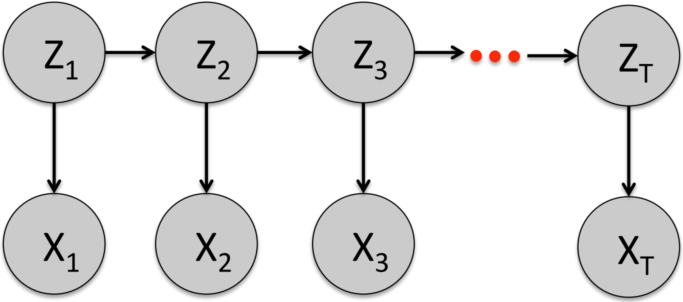
# How to fit an HMM to data? Supervised setting



Supervised learning: all of our observation sequences have state labels
$\mathcal{D} = \{(x_1, z_1), \dots, (x_T, z_T)\}_{1:n}$

- Estimating $\pi$: initial state
- Estimating $A$: transition matrix
- Estimating $\eta_k$: emission probabilities

# How to fit an HMM model? Supervised: MLE

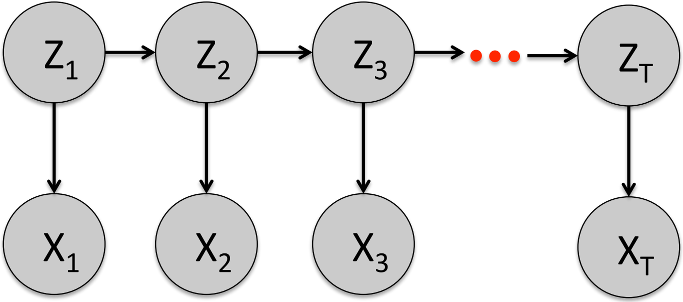


Supervised learning: $\mathcal{D} = \{(x_1, z_1), \ldots, (x_T, z_T)\}_{1:n}$

- Estimating $\pi$: initial state

$$\hat{\pi} = \frac{\sum_{i=1}^{n} z_1^i}{n}.$$

# How to fit an HMM model? Supervised: MLE



Supervised learning: $\mathcal{D} = \{(x_1, z_1), \ldots, (x_T, z_T)\}_{1:n}$

- Estimating $A$: transition matrix $A_{j,k} = p(z_{t+1}^k = 1 \mid z_t^j = 1)$

$$\hat{A}_{j,k} = \frac{\sum_{i=1}^n \sum_{t=1}^{T-1} \mathbb{1}(z_{t+1}^i = k)\mathbb{1}(z_t^i = j)}{\sum_{i=1}^n \sum_{t=1}^{T-1} \mathbb{1}(z_t^i = j)}.$$

# State transition matrix $A$: closer look

Transition matrix $A$ is a conditional probability table

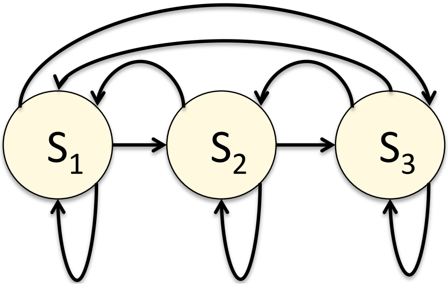
- Transition matrix $A_{j,k} = p(z^k_{t+1} = 1 \mid z^j_t = 1)$

| $t \backslash t+1$ | State 1 | State 2 | State 3 | State 4 | State 5 |
|---|---|---|---|---|---|
| State 1 | 0 | 0.24 | 0.76 | 0 | 0 |
| State 2 | 0 | 0.49 | 0.50 | 0.01 | 0 |
| State 3 | 0 | 0.03 | 0.56 | 0.42 | 0 |
| State 4 | 0.03 | 0.03 | 0.13 | 0.67 | 0.14 |
| State 5 | 0 | 0 | 0.12 | 0.23 | 0.65 |

- Rows sum to one
- MLE for state transitions counts number of times state $k$ transitioned to state $k$, divided by the total number of times state $j$ was observed.
- How likely is State 1 in this HMM?

# State transition matrix: a second look

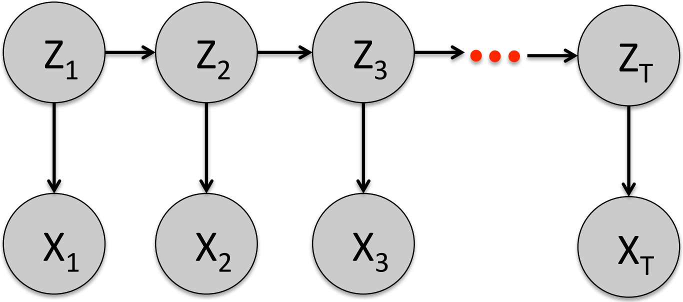
Represent the state transition matrix $A$ as a *state transition diagram*



| | State 1 | State 2 | State 3 |
|---|---|---|---|
| State 1 | 0 | 0.24 | 0.76 |
| State 2 | 0 | 0.49 | 0.51 |
| State 3 | 0.42 | 0.02 | 0.56 |

This is not a graphical model: not acyclic, no conditional probabilities.

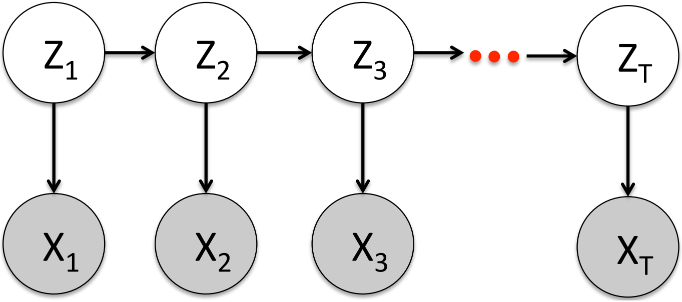
# How to fit an HMM model? Supervised setting



Supervised learning: $\mathcal{D} = \{(x_1, z_1), \ldots, (x_T, z_T)\}_{1:n}$

- Estimating $\eta_k$: emission probabilities
  - For chosen emission distribution, this becomes MLE updates of parameters for state $k$ using only $x_t^i$ for which $z_t^i = k$.
- Overall: running time of $O(TK^2p)$: fast (linear in $T$) when $K$ small.

# How to fit an HMM model? Unsupervised setting

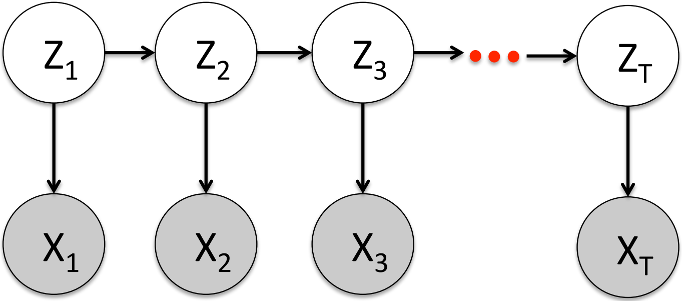


Unsupervised learning: observation sequences do not have state labels $\mathcal{D} = \{x_1, \ldots, x_T\}_{1:n}$

- Estimating $\pi$: initial state
- Estimating $A$: transition matrix
- Estimating $\eta_k$: emission probabilities
- Estimating latent states $Z_{1:T}^{1:n}$.

How do we do this?

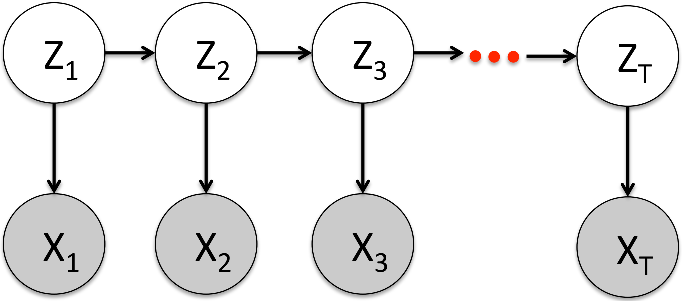
# How to fit an HMM model? Unsupervised setting



Expectation-Maximization (EM): iterate until convergence

- E-step: estimate expected latent states given $\mathcal{D}, \Theta^{(t)}$
  - Estimate expected posterior of latent states $Z_{1:T}^{1:n}$.
- M-step: estimate parameters $\Theta^{(t)}$ given $\mathcal{D}$, $\hat{z}_{1:T}^{1:n(t+1)}$
  - Estimate $\pi$: initial state
  - Estimate $A$: transition matrix
  - Estimate $\eta_k$: emission probabilities

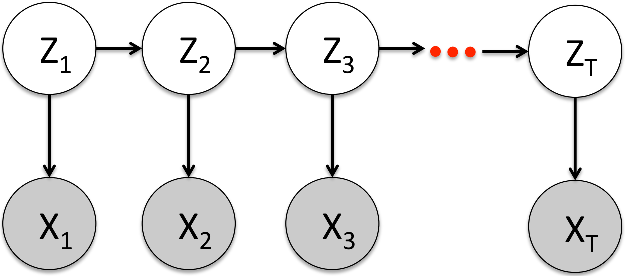
# How to fit an HMM model? Unsupervised setting



Expectation-Maximization (EM):

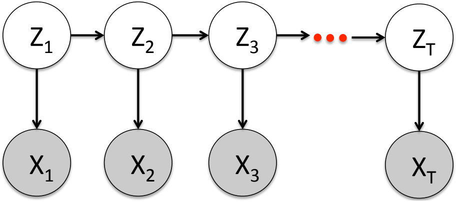
- E-step: estimate latent states given $\mathcal{D}, \Theta^{(t)}$
    - Do we have to consider all $K^T$ possible sequences?
- M-step: estimate parameters $\Theta^{(t)}$ given $\mathcal{D}$, $\hat{z}_{1:T}^{1:n(t+1)}$
    - This is identical to parameter updates in supervised setting with expectations of $z$ from the E-step replacing the observed state labels.

# E-step of EM in HMMs



- E-step: estimate latent states $z$ given $\mathcal{D}, \Theta^{(t)}$
- Amounts to *smoothing* for every time $t$: $\hat{z}_t^i = p(z_t^i \mid x_{1:T}^i)$
- Algorithm to perform smoothing is *forward-backward* algorithm.
- Although derivation requires bookkeeping, Markovian structure allows dynamic programming implementation: $O(TK^2 p)$

# E-step of EM in HMMs

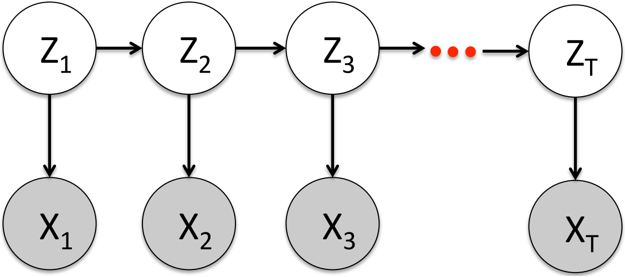


Let's compute the expectation of the variables in the E-step. Let $n = 1$.

First, write out the complete log likelihood for this model:

$$\begin{aligned}\ell_c(\theta, z, x; \mathcal{D}) &= \log[p(z, x \mid \theta)] \\ &= \log\left\{p(z_1)\left[\prod_{t=1}^{T} p(z_t \mid z_{t-1})\right]\left[\prod_{t=1}^{T} p(x_t \mid z_t)\right]\right\} \\ &= \log \pi_{z_1} + \sum_{t=1}^{T-1} \log A_{z_t, z_{t+1}} + \sum_{t=1}^{T} \log p(x_t \mid z_t)\end{aligned}$$

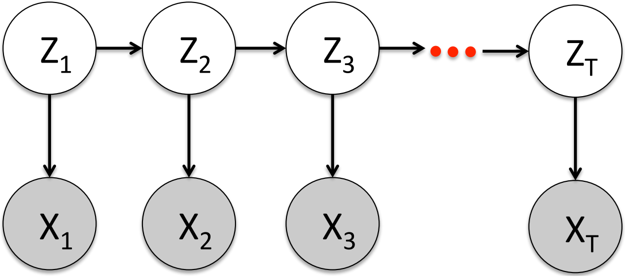
# E-step of EM in HMMs



Use this to write the expected complete log likelihood

$$
\begin{aligned}
\mathrm{E}\left[\ell_c(\theta;\mathcal{D})\right] &= \mathrm{E}\left[\sum_{k=1}^{K} z_1^k \log \pi_k + \sum_{t=1}^{T-1}\sum_{j,k=1}^{K} z_t^j z_{t+1}^k \log A_{j,k} + \sum_{t=1}^{T} \log p(x_t|z_t,\eta)\right] \\
&= \sum_{k=1}^{K} \mathrm{E}[Z_1^k] \log \pi_k + \sum_{t=1}^{T-1}\sum_{j,k=1}^{K} \mathrm{E}[Z_t^j Z_{t+1}^k] \log A_{j,k} + \sum_{t=1}^{T} \mathrm{E}[\log p(x_t|z_t,\eta)]
\end{aligned}
$$

# E-step of EM in HMMs

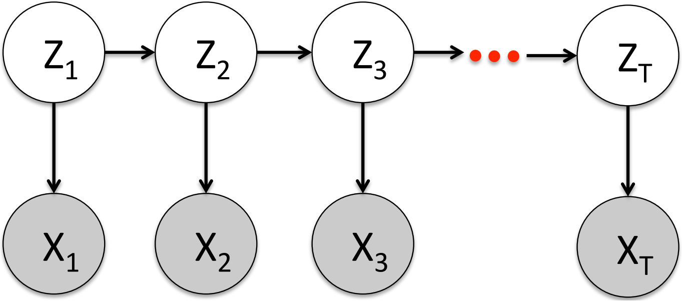


From here we can find the expected sufficient statistics:

$$\mathrm{E}[Z_1^k] = \mathrm{E}[Z_1^k \mid x_{1:T}, \theta] = p(Z_1^k = 1 \mid x_{1:T}, \theta).$$

Since $z_1$ has a multinomial distribution, its expectation is the vector of posterior probabilities for each $k \in 1 : K$.

# E-step of EM in HMMs

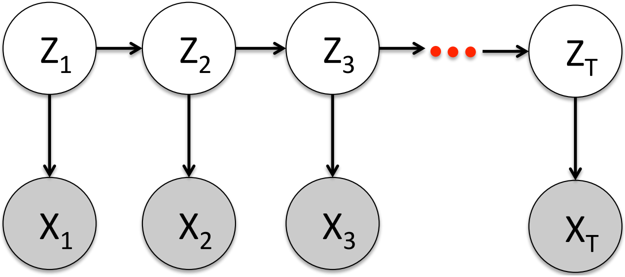


From here we can find the expected sufficient statistics:

$$\mathrm{E}[Z_t^j, Z_{t+1}^k] = \mathrm{E}[Z_t^j, Z_{t+1}^k \mid x_{1:T}, \theta] = \sum_{t=1}^{T-1} p(z_t^j z_{t+1}^k \mid x_{1:T}, \theta).$$

Intuitively, $\mathrm{E}[Z_t^j, Z_{t+1}^k]$ counts how often we see state transition pairs.

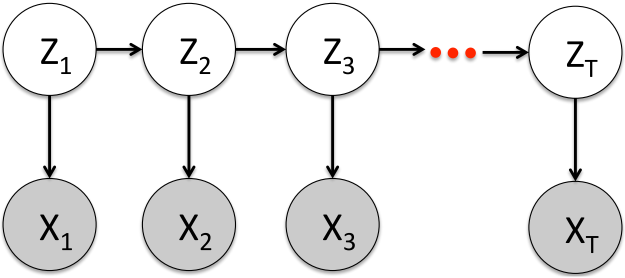
# E-step of EM in HMMs



Computing $\mathrm{E}[Z_t^j]$

$$\begin{aligned}
\mathrm{E}[Z_t^k] &\triangleq p[Z_t^k = 1 \mid x_{1:T}, \theta^{(s)}] \\
&= p[Z_t^k = 1 \mid x_{t:T}, x_{1:t-1}, \theta^{(s)}] \\
&\propto p[x_{t:T} \mid Z_t^k = 1, x_{1:t-1}, \theta^{(s)}] p[Z_t^k = 1 \mid x_{1:t-1}, \theta^{(s)}] \\
&= p[x_{t:T} \mid Z_t^k = 1, \theta^{(s)}] p[Z_t^k = 1 \mid x_{1:t-1}, \theta^{(s)}] \\
&\triangleq \beta_t(k) \cdot \alpha_t(k)
\end{aligned}$$

# E-step of EM in HMMs



We recursively compute $\beta_t(k) \cdot \alpha_t(k)$ for $t = 1 : T$:

- $\alpha_t(k)$ is the *forward* step: $p[Z_t^k = 1 \mid x_{1:t-1}, \theta^{(s)}]$
- $\beta_t(k)$ is the *backward* step: $p[x_{t:T} \mid Z_t^k = 1, \theta^{(s)}]$
- Use $\alpha_t(k)$ and $\beta_t(k)$ to compute $\mathrm{E}[Z_t^j, Z_{t+1}^k]$ sequentially.

# EM in HMMs

## Expectation-Maximization (EM), also called Baum-Welch

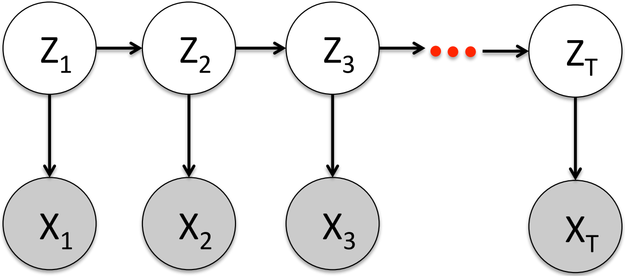
Initialize with hypothetical initial, transition, and emission probabilities (K-means?)

Run until convergence (e.g., observed data likelihood change $< \epsilon$):

- E-step: estimate latent states $Z$ given $\mathcal{D}, \Theta^{(t)}$
  - Run forward-backward to compute $\mathrm{E}[Z_t^k]$, $\mathrm{E}[Z_t^j, Z_{t+1}^k]$
- M-step: estimate parameters $\Theta^{(t)}$ given $\mathcal{D}$, $\hat{z}_{1:T}^{1:n(t+1)}$
  - Compute MLE parameter updates in supervised setting with expectations of $z$ from the E-step replacing the observed state labels.

# Posterior decoding and Viterbi decoding in HMMs

Now that we have done the work for EM, what about posterior decoding?



- Can we (quickly) find most probable state sequence for new $x_{1:T}$ given parameters?
- The problem is as follows: $\tilde{z}_{1:T} = \arg\max_{z \in \{1:K\}^T} p(z_{1:T} \mid \theta, x_{1:T})$
- Despite discrete search space, same complexity as forward-backward
- Instead of summing out latent state $z_t$, take maximum $p(z_t \mid x_{1:T})$
- Compute most likely path through network of states, transitions: DP.

# Example: weather analysis

## Example: weather in Princeton in April

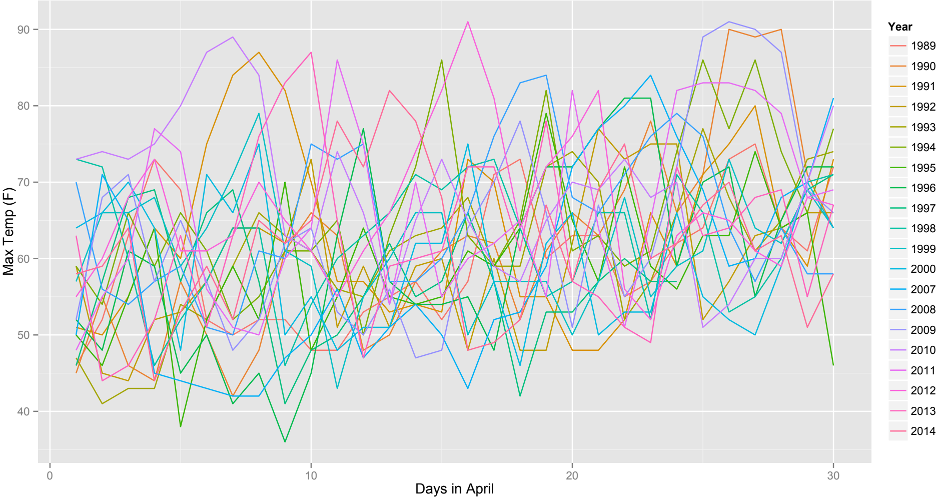
I downloaded from http://www.wunderground.com/history 20 years of historical weather data for Princeton in April.

- There are $n = 20$ samples, corresponding to 20 Aprils' worth of data
- There are $p = 6$ features, corresponding to high temp, low temp, average wind speed, humidity, precipitation, and cloud cover.
- There are $T = 30$ observed values for each sample, because each sample corresponds to 30 days of weather information.
- We will estimate $K$ latent states from these data.

Is it necessary that each sample have identical time points $T$?

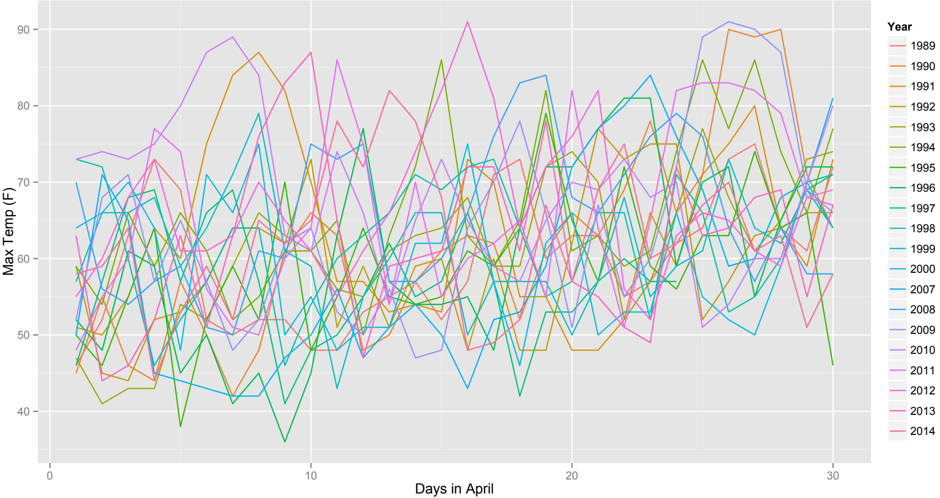
# HMM example: Princeton April weather

Plot max temperature across 20 years ($n$) and 30 days ($T$).



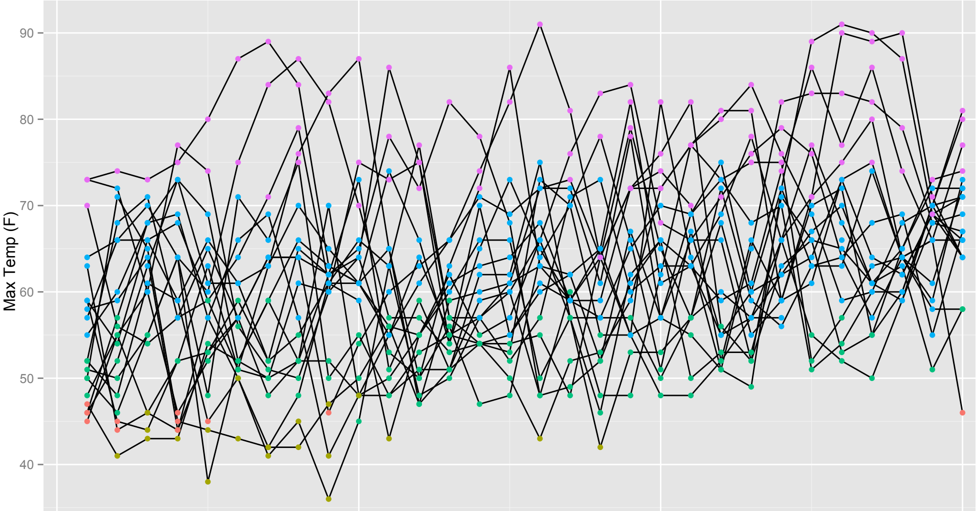
# HMM example: Princeton April weather

Plot max temperature across 20 years ($n$) and 30 days ($T$).



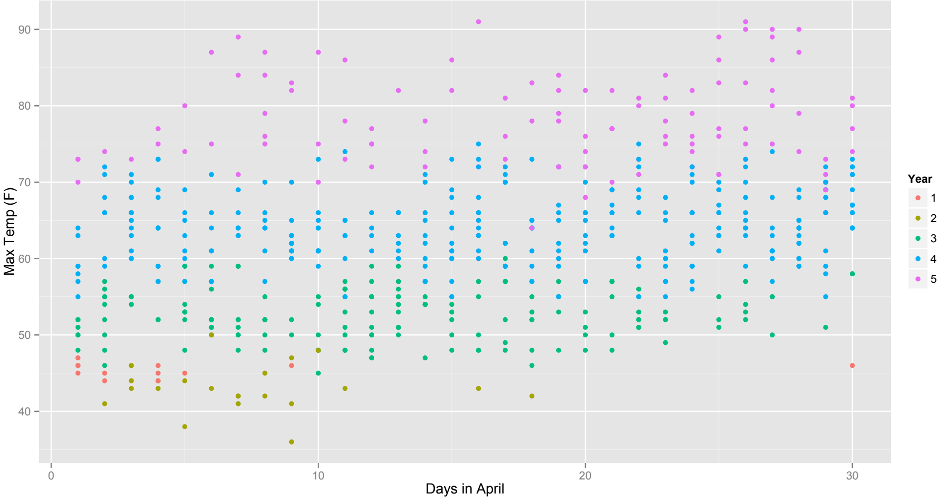
# HMM example: Princeton April weather

HMM with $K = 5$ (colors are Viterbi state labels):



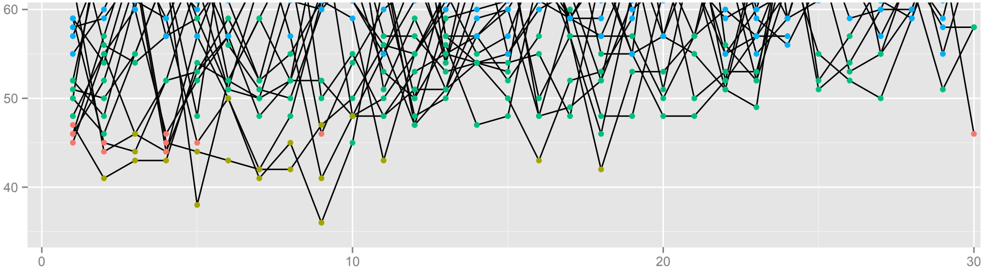
# HMM example: Princeton April weather

HMM with $K = 5$ (maybe easier to understand):



# HMM example: Princeton April weather

HMM with $K = 5$ (zoomed):



State 1 (orange): no self-transitions.

# HMM example: Princeton April weather

We can look at the estimated state transition matrix:

| | State 1 | State 2 | State 3 | State 4 | State 5 |
|---|---|---|---|---|---|
| State 1 | 0 | 0.24 | 0.76 | 0 | 0 |
| State 2 | 0 | 0.49 | 0.50 | 0.010 | 0 |
| State 3 | 0 | 0.029 | 0.56 | 0.42 | 0 |
| State 4 | 0.035 | 0.018 | 0.13 | 0.67 | 0.14 |
| State 5 | 0 | 0 | 0.12 | 0.23 | 0.65 |

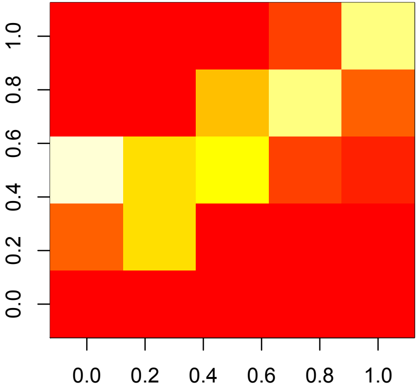
We can also look at the variables estimated for the emissions distributions:

$\mu$ variables: 45.82, 43.62, 53.01, 63.81, 76.83

$\sigma$ variables: 1.88, 10.46, 14.14, 27.28, 46.16

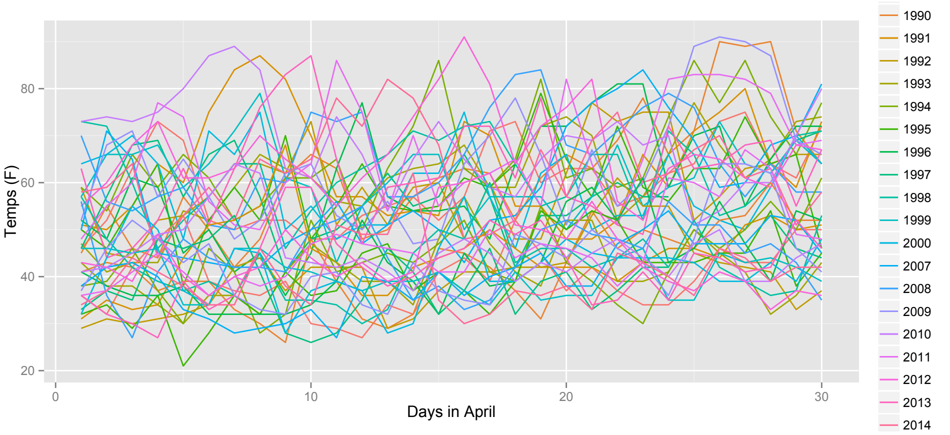
# HMM example: Princeton April weather

We can examine the transition matrix that we estimate from these data.:

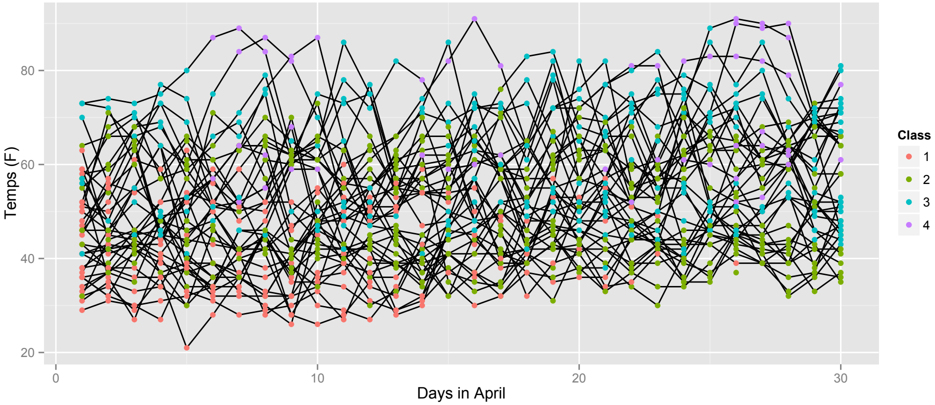


# HMM example: Princeton April weather

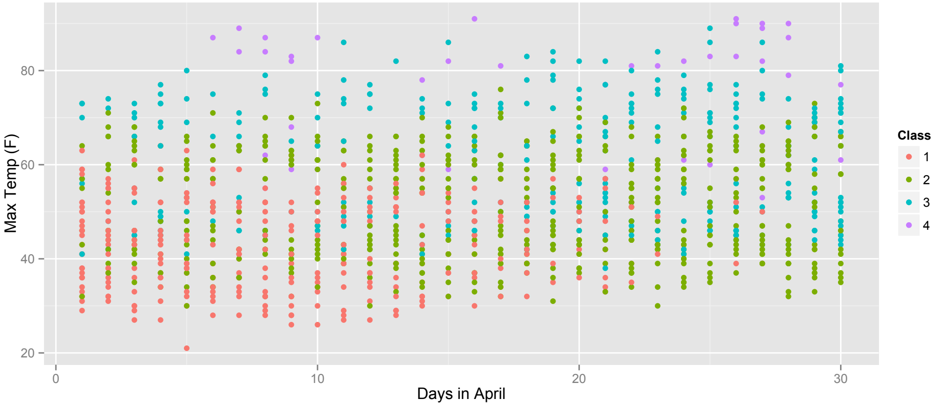
If we modify $K = 4$ and add an additional variable (min_temp), we adapt the emission distributions to be multivariate Gaussian.



# HMM example: Princeton April weather

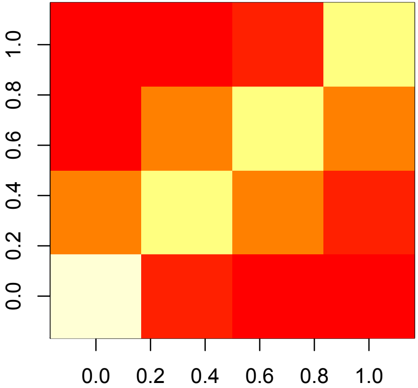


# HMM example: Princeton April weather



# HMM example: Princeton April weather

We can examine the transition matrix that we estimate from these data.:



# HMM example: Princeton April weather

- I tried and failed to fit an HMM with $K = 2, 3, 4, 5$ states to all six weather variables.
- EM sensitive to start points: many states overtook other states and they were removed from the model.
- Even with K-means, hard to find appropriate state means, covariances to create stable EM.

# HMM strengths

Flexible, adaptable approach to analyzing sequential data.

In the supervised setting:

- very fast MLE parameter estimates
- handles missing data smoothly
- arbitrary emission distribution

In the unsupervised setting:

- label sequences that have no labels
- very fast EM parameter estimates
- arbitrary emission distribution

# HMM limitations

In the supervised setting:

- Stationarity does not always hold.
- Discrete time/states are not always possible

In the unsupervised setting:

- EM finds local maxima;
- EM is sensitive to initialization
- Difficult to choose $K$.

# HMM extensions

Many ways to extend HMMs:

- Higher order HMMs: remove conditional independencies among local latent states
- Profile HMMs: state decay is not geometric
- Infinite state HMMs/HDP-HMMs: number of states is random variable
- Factorial HMMs: state has multiple different factors.
- Markov random fields: undirected edge version of HMMs
- Kalman filters: state is continuous
- Autoregressive models (AR): time is continuous
- PhyloHMMs: emission is more complicated than a mixture model
- ...

# Additional Resources

- MLAPA: Ch 17
- *A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition* [Rabiner 1989]
- Metacademy: *Hidden Markov models*