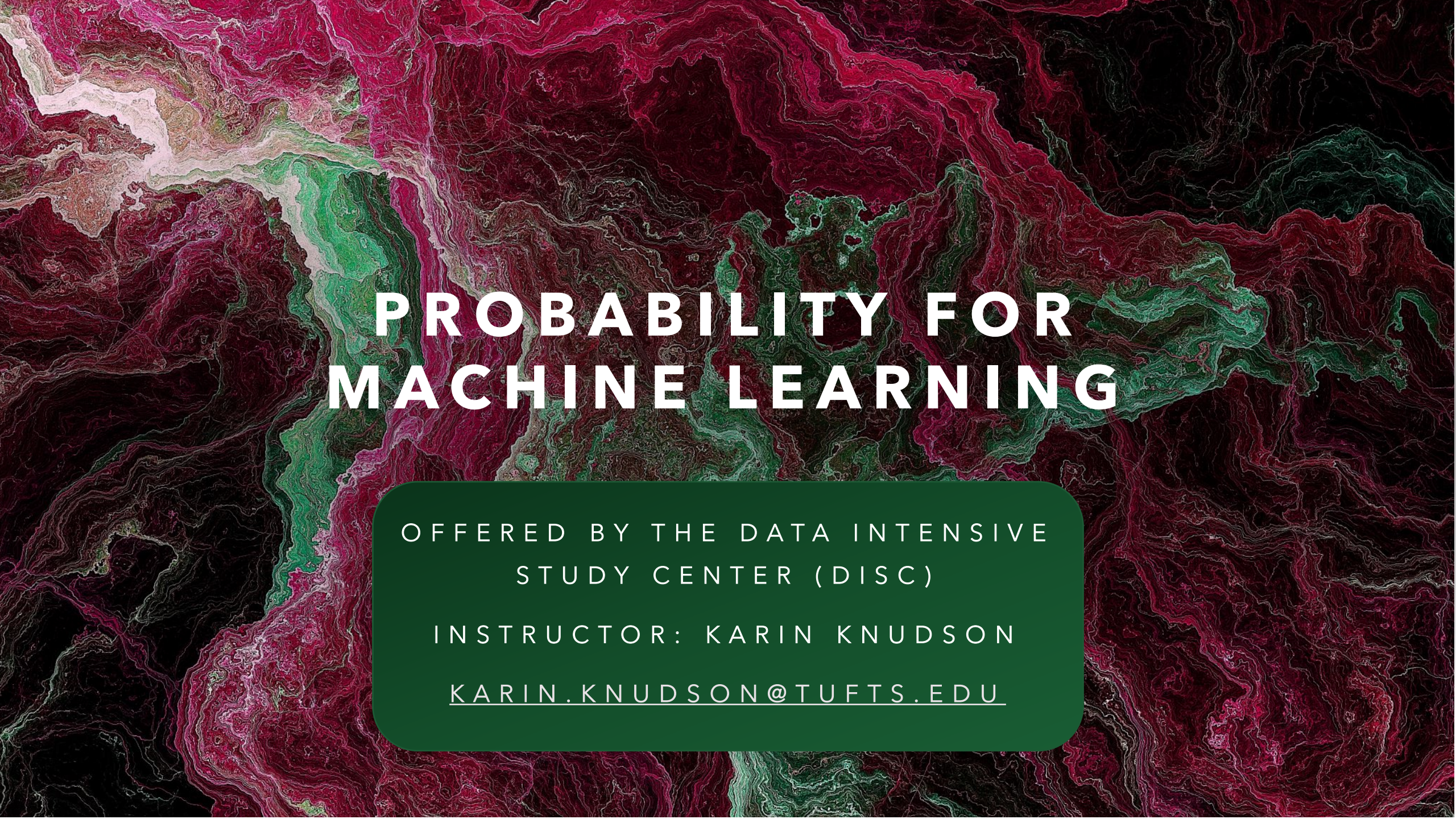


PROBABILITY FOR MACHINE LEARNING

OFFERED BY THE DATA INTENSIVE STUDY CENTER (DISC)

INSTRUCTOR: KARIN KNUDSON

KARIN.KNUDSON@TUFTS.EDU

MARKOV CHAINS (DISCRETE-TIME)

Sequence of random variables satisfying a “memorylessness” property: next value depends only on current value

MARKOV CHAINS (DISCRETE-TIME)

Sequence of random variables satisfying a “memorylessness” property: next value depends only on current value

$$X_1, X_2, X_3, \ldots \qquad P(X_n|X_1, X_2, ..X_{n-1}) = P(X_n|X_{n-1})$$

MARKOV CHAINS (DISCRETE-TIME)

Sequence of random variables satisfying a "memorylessness" property: next value depends only on current value

$$X_1, X_2, X_3, \ldots \qquad P(X_n|X_1, X_2, ..X_{n-1}) = P(X_n|X_{n-1})$$

If the random variables have finite support (i.e. the **state space** is finite), and the probabilities don't depend on n (time-homogeneous Markov chain), then can specify with:

$$p_{ij} = P(X_n = j|X_{n-1} = i)$$

MARKOV CHAINS (DISCRETE-TIME)

Sequence of random variables satisfying a "memorylessness" property: next value depends only on current value

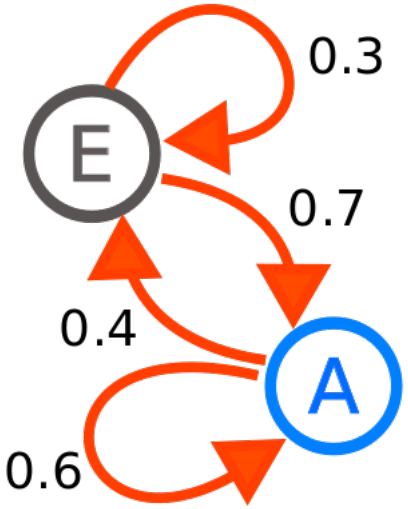
$$X_1, X_2, X_3, \ldots \qquad P(X_n|X_1, X_2, ..X_{n-1}) = P(X_n|X_{n-1})$$

If the random variables have finite support (i.e. the **state space** is finite), and the probabilities don't depend on n (time-homogeneous Markov chain), then can specify with:

gives **transition matrix** → $p_{ij} = P(X_n = j|X_{n-1} = i)$

Can extend to a Markov chain of order k, where state depends on previous k states.

MARKOV CHAINS (DISCRETE-TIME)

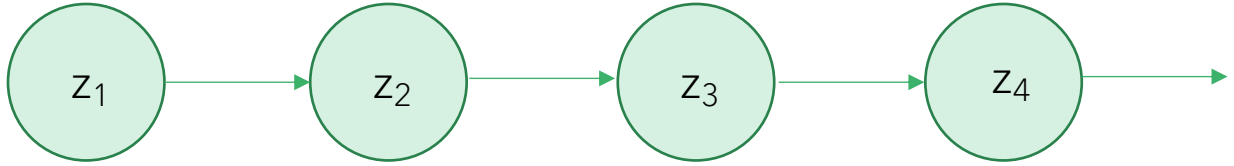


Transition matrix P

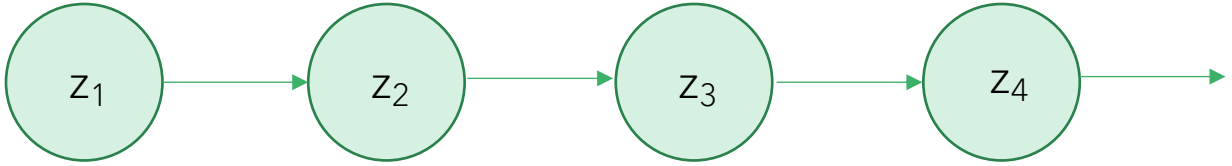
	E	A
E	.3	.7
A	.4	.6

A stationary distribution x satisfies **xP = x**

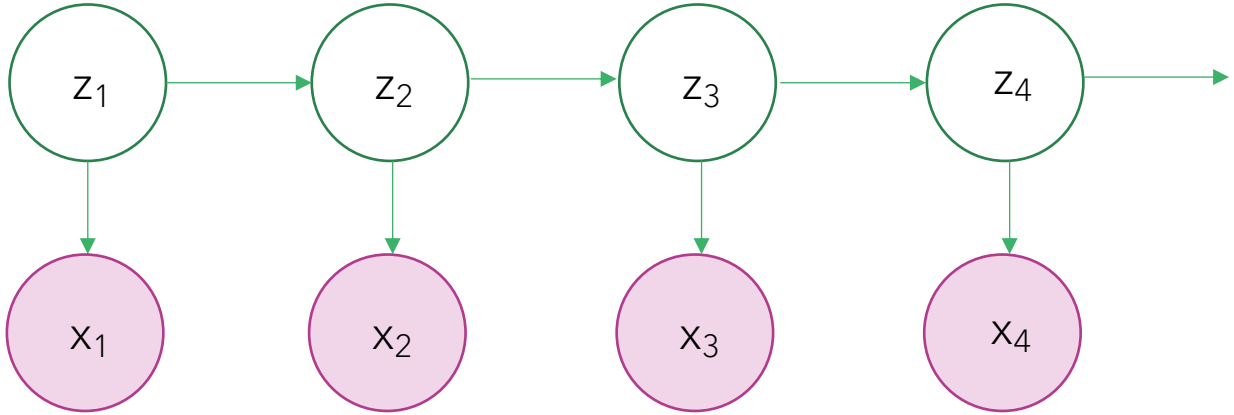
MARKOV CHAIN



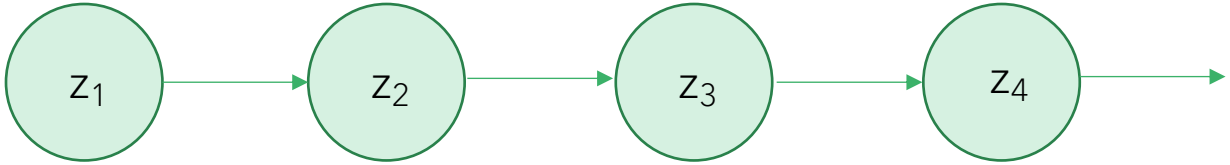
MARKOV CHAIN



HIDDEN MARKOV MODELS (HMM)



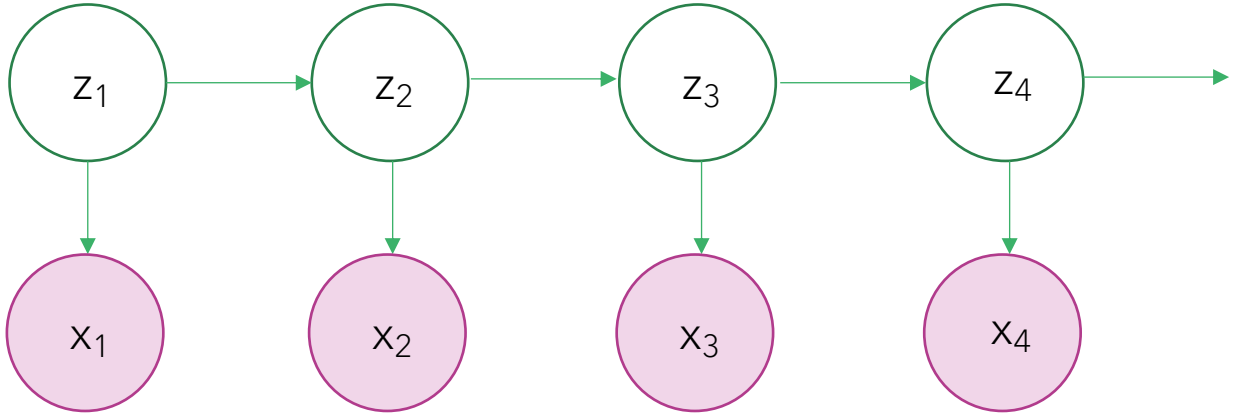
MARKOV CHAIN



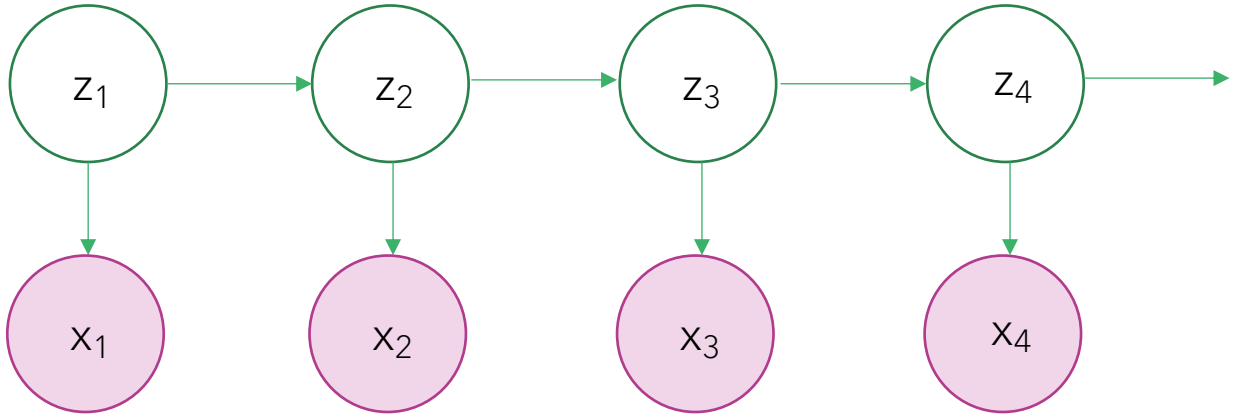
HIDDEN MARKOV MODELS (HMM)

hidden/latent states (discrete)

observed



HIDDEN MARKOV MODELS (HMM)

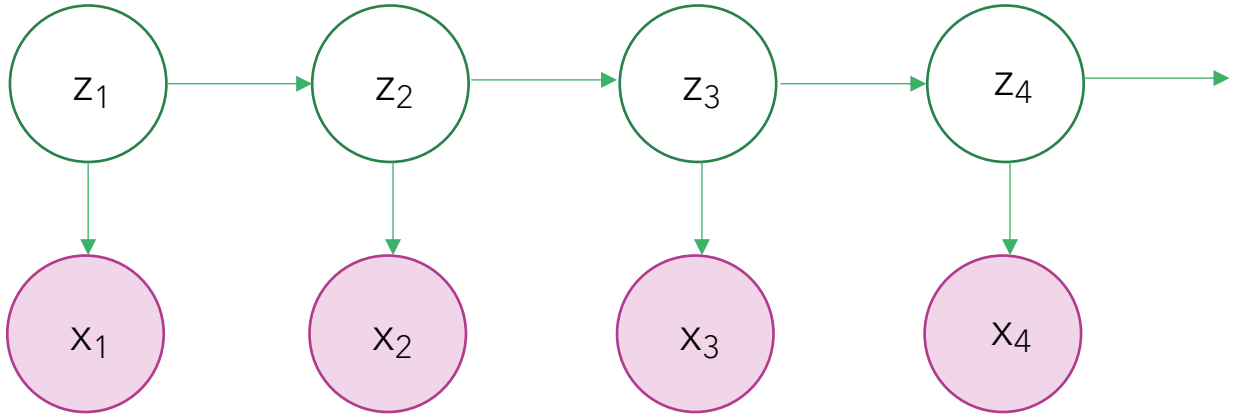


Output probabilities/emission probabilities: $p(\mathbf{x_n}|\mathbf{z}_n, \phi)$

Transition probabilities given by matrix: $\mathbf{A}$

Distribution of first hidden state: $\boldsymbol{\pi}$

HIDDEN MARKOV MODELS (HMM)



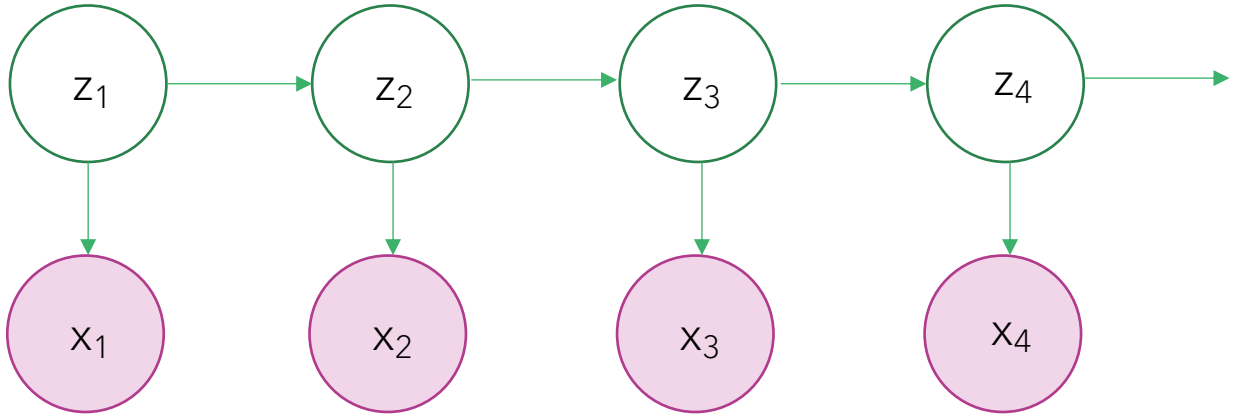
Output probabilities/emission probabilities: $p(\mathbf{x_n}|\mathbf{z}_n, \phi)$

$$p(\mathbf{x}_n|\mathbf{z}_n, \phi) = \prod_{k=1}^{K} p(\mathbf{x}_n|\phi_k)^{z_{nk}} \text{ (z gives one-hot encoding of state)}$$

Transition probabilities given by matrix: $\mathbf{A}$

Distribution of first hidden state: $\boldsymbol{\pi}$

HIDDEN MARKOV MODELS (HMM)



Output probabilities/emission probabilities: $p(\mathbf{x_n}|\mathbf{z}_n, \phi)$

$$p(\mathbf{x}_n|\mathbf{z}_n, \phi) = \prod_{k=1}^{K} p(\mathbf{x}_n|\phi_k)^{z_{nk}} \text{ (z gives one-hot encoding of state)}$$

Transition probabilities given by matrix: $\mathbf{A}$

Distribution of first hidden state: $\boldsymbol{\pi}$

HIDDEN MARKOV MODELS (HMM)

Output probabilities/emission probabilities: $p(\mathbf{x_n}|\mathbf{z}_n, \phi)$

$$p(\mathbf{x}_n|\mathbf{z}_n, \phi) = \prod_{k=1}^{K} p(\mathbf{x}_n|\phi_k)^{z_{nk}} \text{ (z gives one-hot encoding of state)}$$

Transition probabilities given by matrix: $\mathbf{A}$

Distribution of first hidden state: $\boldsymbol{\pi}$

HIDDEN MARKOV MODELS (HMM)

Output probabilities/emission probabilities: $p(\mathbf{x_n}|\mathbf{z}_n, \phi)$

$$p(\mathbf{x}_n|\mathbf{z}_n, \phi) = \prod_{k=1}^{K} p(\mathbf{x}_n|\phi_k)^{z_{nk}} \text{ (z gives one-hot encoding of state)}$$

Transition probabilities given by matrix: $\mathbf{A}$

Distribution of first hidden state: $\boldsymbol{\pi}$

Learning parameters: can use expectation maximization (EM) algorithm – useful for optimizing a posterior or likelihood when we have latent variables. Iteratively:

Find $P(Z|X, \theta^{old})$

Compute a new θ to maximize the expectation under $P(Z|X, \theta^{old})$ of the log likelihood of $P(Z, X|\theta)$

HIDDEN MARKOV MODELS (HMM)

Output probabilities/emission probabilities: $p(\mathbf{x_n}|\mathbf{z}_n, \phi)$

$$p(\mathbf{x}_n|\mathbf{z}_n, \phi) = \prod_{k=1}^{K} p(\mathbf{x}_n|\phi_k)^{z_{nk}} \text{ (z gives one-hot encoding of state)}$$

Transition probabilities given by matrix: $\mathbf{A}$

Distribution of first hidden state: $\boldsymbol{\pi}$

Learning parameters: expectation maximization (EM) algorithm – useful for optimizing a posterior or likelihood when we have latent variables. Iteratively:

$$\text{Find } P(Z|X, \theta^{old})$$
$$\text{Compute a new } \theta \text{ to maximize the expectation under } P(Z|X, \theta^{old})$$
$$\text{of the log likelihood of } P(Z, X|\theta)$$

Viterbi algorithm for finding most likely sequence of latent states

HIDDEN MARKOV MODELS (HMM)

Can put priors over quantities of interest

If we had a two state system, what form might the prior of a set of transition probabilities take?

For prior over transition probabilities with more than two states, could use Dirichlet prior (generalizes the beta distribution)

If we don't know the number of states, could use a Dirichlet process

Final point: note connection to mixture models!

HIDDEN MARKOV MODELS (HMM)

Can put priors over quantities of interest

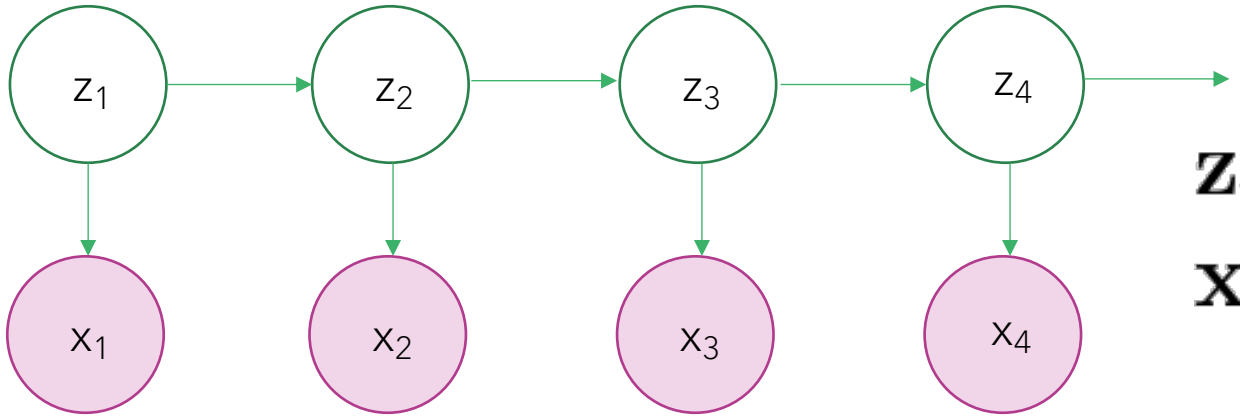
If we had a two state system, what form might the prior of a set of transition probabilities take?

For prior over transition probabilities with more than two states, could use Dirichlet prior (generalizes the beta distribution)

If we don't know the number of states, could use a Dirichlet process

Final point: note connection to mixture models!

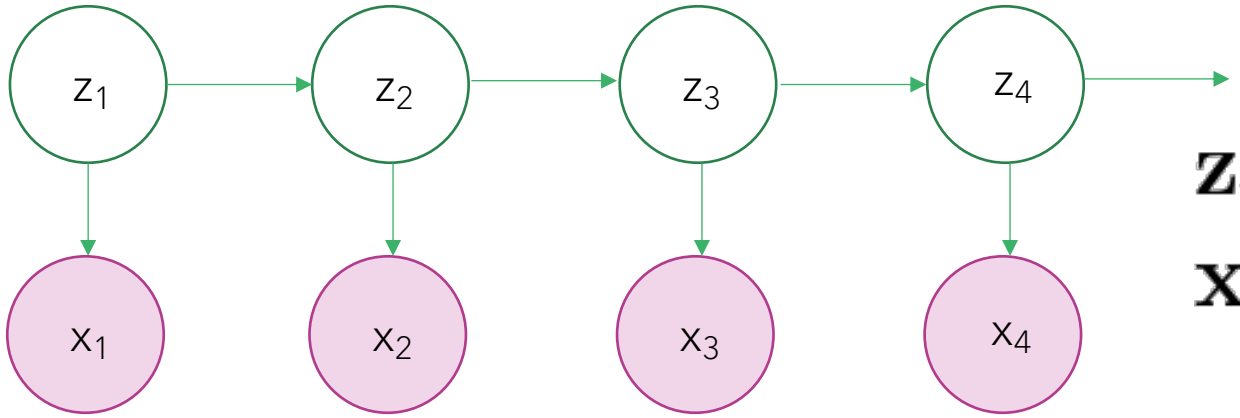
RELATED: LINEAR DYNAMICAL SYSTEMS



$$\mathbf{z}_n|\mathbf{z}_{n-1} \sim N(A\mathbf{z}_{n-1}, \Lambda)$$
$$\mathbf{x}_n|\mathbf{z}_n \sim N(C\mathbf{z}_n, \Sigma)$$

Similar structure with latent and observed variables, but latent and observed variables are Gaussian and have linear relationship

RELATED: LINEAR DYNAMICAL SYSTEMS



$$\mathbf{z}_n|\mathbf{z}_{n-1} \sim N(A\mathbf{z}_{n-1}, \Lambda)$$
$$\mathbf{x}_n|\mathbf{z}_n \sim N(C\mathbf{z}_n, \Sigma)$$

Similar structure with latent and observed variables, but latent and observed variables are Gaussian and have linear relationship

If parameters are known, Kalman filtering. If unknown, can do inference using, e.g. EM

MARKOV CHAIN MONTE CARLO (MCMC)

Recall that even if we can't compute a distribution, having a representative set of samples gives us a great deal of information!

MCMC gives a framework for sampling from many distributions

MCMC

Want to sample from a distribution of interest p*

Will do so by trying to find a Markov chain that is invariant to that distribution, meaning a step in the chain does not change the distribution:

$$p^*(\mathbf{x}) = \sum_{\mathbf{x}'} T(\mathbf{x}', \mathbf{x}) p^*(\mathbf{x}')$$

MCMC

Want to sample from a distribution of interest p*

Will do so by trying to find a Markov chain that is **invariant to that distribution**, meaning a step in the chain does not change the distribution:

$$p^*(\mathbf{x}) = \sum_{\mathbf{x}'} T(\mathbf{x}', \mathbf{x}) p^*(\mathbf{x}')$$

If transition probabilities satisfy **detailed balance**:

$$p^*(\mathbf{x}) T(\mathbf{x}, \mathbf{x}') = p^*(\mathbf{x}') T(\mathbf{x}', \mathbf{x})$$

then p* will be invariant

MCMC

So, if we to define a Markov chain that generates samples from the distribution of interest – e.g. if we can show that detailed balance is satisfied

MCMC

So, if we to define a Markov chain that generates samples from the distribution of interest – e.g. if we can show that detailed balance is satisfied

Metropolis-Hastings algorithm:

$$\text{Propose } \mathbf{x}' \text{ from } q_k(\mathbf{x}|\mathbf{x}^{prev})$$

$$\text{Accept with probability } \min\left(1, \frac{\tilde{p}(\mathbf{x}')q_k(\mathbf{x}^{prev}|\mathbf{x}')}{\tilde{p}(\mathbf{x}^{prev})q_k(\mathbf{x}'|\mathbf{x}^{prev})}\right)$$

$$\tilde{p}(\mathbf{x}) = p^*(\mathbf{x})/Z$$

MCMC

Special case of Metropolis-Hastings: Gibbs sampling

(iteratively sample each component of the vector whose distribution we'd like to know, conditioned on all the other components)

MCMC algorithms that make use of gradient information:

Hamiltonian Monte Carlo, No U-Turn Samplier

RESOURCES

Pattern Recognition and Machine Learning by Christopher Bishop

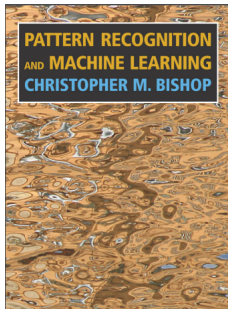
An Introduction to Statistical Learning with Applications in R by Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani

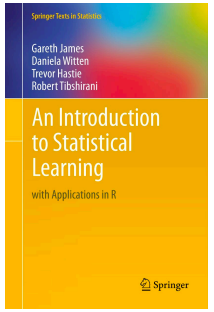
The Elements of Statistical Learning, by Trevor Hastie, Robert Tibshirani, Jerome Friedman

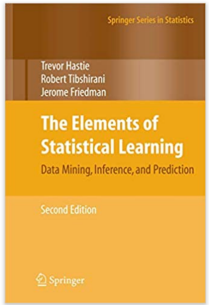
Statistical Rethinking Richard McElreath

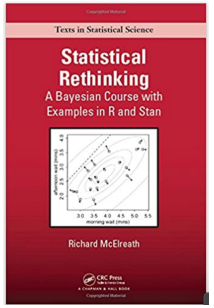
Machine Learning: A Probabilistic Perspective, by Kevin Murphy

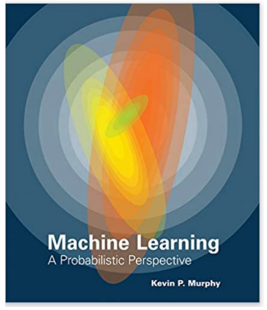
Bayesian Data Analysis, by Gelman et al.











THANK YOU!