ELEN 4720: Machine Learning for Signals, Information and Data

Lecture 20

Prof. John Paisley

Department of Electrical Engineering

Columbia University

Sequential data

So far, when thinking probabilistically we have focused on the i.i.d. setting.

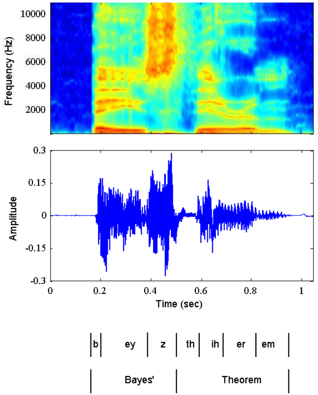
- All data are independent given a model parameter.
- This is often a reasonable assumption, but was also done for convenience.

In some applications this assumption is bad:

- Modeling rainfall as a function of hour
- Daily value of currency exchange rate
- Acoustic features of speech audio

The distribution on the next value clearly depends on the previous values.

A basic way to model sequential information is with a discrete, first-order Markov chain.



MARKOV CHAINS

EXAMPLE: ZOMBIE WALKER

Imagine you see a zombie in an alley. Each time it moves forward it steps

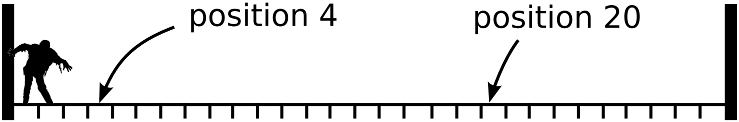
$$(\text{left, straight, right}) \quad \text{with probability} \quad (p_l, p_s, p_r),$$

unless it's next to the wall, in which case it steps straight with probability p_s^w and toward the middle with probability p_m^w.

The distribution on the next location only depends on the current location.

RANDOM WALK NOTATION

We simplify the problem by assuming there are only a finite number of positions the zombie can be in, and we model it as a random walk.



The distribution on the next position only depends on the current position. For example, for a position i away from the wall,

$$s_{t+1} \mid \{s_t = i\} \quad = \quad \begin{cases} i+1 & \text{w.p.} \quad p_r \\ i & \text{w.p.} \quad p_s \\ i-1 & \text{w.p.} \quad p_l \end{cases}$$

This is called the *first-order Markov property*. It's the simplest type. A second-order model would depend on the previous two positions.

Matrix notation

A more compact notation uses a matrix.

For the random walk problem, imagine we have 6 different positions, called *states*. We can write the *transition matrix* as

$$M = \begin{bmatrix} p_s^w & p_m^w & 0 & 0 & 0 & 0 \\ p_l & p_s & p_r & 0 & 0 & 0 \\ 0 & p_l & p_s & p_r & 0 & 0 \\ 0 & 0 & p_l & p_s & p_r & 0 \\ 0 & 0 & 0 & p_l & p_s & p_r \\ 0 & 0 & 0 & 0 & p_m^w & p_s^w \end{bmatrix}$$

M_{ij} is the probability that the next position is j given the current position is i.

We can jumble this matrix by moving rows and columns around, as long as we can map the rows and columns to a position.

First-order Markov chain (general)

Let $s \in \{1, \ldots, S\}$. A sequence $(s_1, \ldots, s_t)$ is a *first-order Markov chain* if

$$p(s_1, \ldots, s_t) \overset{(a)}{=} p(s_1) \prod_{u=2}^{t} p(s_u | s_1, \ldots, s_{u-1}) \overset{(b)}{=} p(s_1) \prod_{u=2}^{t} p(s_u | s_{u-1})$$

From the two equalities above:

(a) This equality is *always* true, regardless of the model (chain rule).

(b) This simplification results from the Markov property assumption.

Notice the difference from the i.i.d. assumption

$$p(s_1, \ldots, s_t) = \begin{cases} p(s_1) \prod_{u=2}^{t} p(s_u | s_{u-1}) & \text{Markov assumption} \\ \prod_{u=1}^{t} p(s_u) & \text{i.i.d. assumption} \end{cases}$$

From a modeling standpoint, this is a significant difference.

First-order Markov chain (general)

Again, we encode this more general probability distribution in a matrix:

$$M_{ij} \;=\; p(s_t = j | s_{t-1} = i)$$

We will adopt the notation that rows are distributions.

- M is a *transition matrix*, or *Markov matrix*.
- M is $S \times S$ and each row sums to one.
- M_{ij} is the probability of transitioning to state j given we are in state i.

Given a starting state, s_0, we generate a sequence $(s_1, \dots, s_t)$ by sampling

$$s_t \,|\, s_{t-1} \;\sim\; \text{Discrete}(M_{s_{t-1},:}).$$

PROPERTY: STATE DISTRIBUTION

Q: Can we say at the beginning what state we'll be in at step $t+1$?

A: Imagine at step t that we have a probability distribution on which state we're in, call it $p(s_t = u)$. Then the distribution on s_{t+1} is

$$p(s_{t+1} = j) = \sum_{u=1}^{S} \underbrace{p(s_{t+1} = j | s_t = u) p(s_t = u)}_{p(s_{t+1}=j,\ s_t=u)}.$$

Represent $p(s_t = u)$ with the *row* vector w_t (the state distribution). Then

$$\underbrace{p(s_{t+1} = j)}_{w_{t+1}(j)} = \sum_{u=1}^{S} \underbrace{p(s_{t+1} = j | s_t = u)}_{M_{uj}} \underbrace{p(s_t = u)}_{w_t(u)}.$$

We can calculate this for all j with the matrix-vector product $w_{t+1} = w_t M$. Therefore, $w_{t+1} = w_0 M^{t+1}$ and w_0 can be indicator if starting state is known.

Property: Stationary distribution

Given current state distribution w_t, the distribution on the next state is

$$w_{t+1}(j) = \sum_{u=1}^{S} M_{uj} w_t(u) \quad \Longleftrightarrow \quad w_{t+1} = w_t M$$

What happens if we project an infinite number of steps out?

Definition: Let $w_\infty = \lim_{t\to\infty} w_t$. Then w_∞ is the *stationary distribution*.

- There are many technical results that can be proved about w_∞.
- Property: If the following are true, then w_∞ is the same vector for all w_0
  1. We can eventually reach any state starting from any other state,
  2. The sequence doesn't loop between sets of states in a pre-defined pattern.
- $w_\infty = w_\infty M$ since w_t is converging and $w_{t+1} = w_t M$.

This last property is related to the first eigenvector of M^T:

$$M^T q_1 = \lambda_1 q_1 \quad \Longrightarrow \quad \lambda_1 = 1, \quad w_\infty = \frac{q_1}{\sum_{u=1}^{S} q_1(u)}$$

A RANKING ALGORITHM

Example: Ranking objects

We show an example of using the stationary distribution of a Markov chain to rank objects. The data are pairwise comparisons between objects.

For example, we might want to rank

- Sports teams or athletes competing against each other
- Objects being compared and selected by users
- Web pages based on popularity or relevance

Our goal is to rank objects from "best" to "worst."

- We will construct a random walk matrix on the objects. The stationary distribution will give us the ranking.
- Notice: We don't consider the sequential information in the data itself. The Markov chain is an artificial modeling construct.

Example: Team rankings

Problem setup

We want to construct a Markov chain where each team is a state.

- We encourage transitions from teams that lose to teams that win.
- Predicting the "state" (i.e., team) far in the future, we can interpret a more probable state as a better team.

One specific approach to this specific problem:

- Transitions only occur between teams that play each other.
- If Team A beats Team B, there should be a high probability of transitioning from B$\rightarrow$A and small probability from A$\rightarrow$B.
- The strength of the transition can be linked to the score of the game.

EXAMPLE: TEAM RANKINGS

How about this?

Initialize $\widehat{M}$ to a matrix of zeros. For a particular game, let j_1 be the index of Team A and j_2 the index of Team B. Then update

$$\widehat{M}_{j_1 j_1} \leftarrow \widehat{M}_{j_1 j_1} + \mathbb{1}\{\texttt{Team A wins}\} + \frac{\texttt{points}_{j_1}}{\texttt{points}_{j_1} + \texttt{points}_{j_2}},$$

$$\widehat{M}_{j_2 j_2} \leftarrow \widehat{M}_{j_2 j_2} + \mathbb{1}\{\texttt{Team B wins}\} + \frac{\texttt{points}_{j_2}}{\texttt{points}_{j_1} + \texttt{points}_{j_2}},$$

$$\widehat{M}_{j_1 j_2} \leftarrow \widehat{M}_{j_1 j_2} + \mathbb{1}\{\texttt{Team B wins}\} + \frac{\texttt{points}_{j_2}}{\texttt{points}_{j_1} + \texttt{points}_{j_2}},$$

$$\widehat{M}_{j_2 j_1} \leftarrow \widehat{M}_{j_2 j_1} + \mathbb{1}\{\texttt{Team A wins}\} + \frac{\texttt{points}_{j_1}}{\texttt{points}_{j_1} + \texttt{points}_{j_2}}.$$

After processing all games, let M be the matrix formed by normalizing the rows of $\widehat{M}$ so they sum to 1.

EXAMPLE: 2018-2019 COLLEGE BASKETBALL SEASON

Coaches Poll

RK	TEAM	REC	PTS	TREND
1	Virginia (32)	35-3	800	↑ 1
2	Texas Tech	31-7	767	↑ 8
3	Michigan State	32-7	722	↑ 3
4	Duke	32-6	667	↓ 3
5	Auburn	30-10	665	↑ 13
6	Gonzaga	33-4	633	↓ 3
7	Kentucky	30-7	590	-
8	Purdue	26-10	563	↑ 5
9	North Carolina	29-7	552	↓ 5
10	Tennessee	31-6	535	↓ 5
11	Michigan	30-7	485	↓ 3
12	Houston	33-4	470	↓ 3
13	Florida State	29-8	409	↓ 2
14	Virginia Tech	26-9	384	↑ 1
15	LSU	28-7	313	↓ 3
16	Kansas	26-10	241	↑ 1
17	Buffalo	32-4	232	↓ 1
18	Wofford	30-5	205	↑ 1
19	Kansas State	25-9	171	↓ 5
20	Villanova	26-10	127	↑ 2
21	Oregon	25-13	117	↑ 5
22	Maryland	23-11	110	↑ 4
23	Nevada	29-5	105	↓ 3
24	Wisconsin	23-11	93	↓ 3
25	Iowa State	23-12	85	↓ 2

Markov chain ranking

	RK	SCORE	TEAM
	1.	0.00925	Virginia
	2.	0.00891	Duke
	3.	0.00794	Michigan St
	4.	0.00724	Texas Tech
	5.	0.00677	Kentucky
	6.	0.00646	North Carolina
	7.	0.00642	Tennessee
	8.	0.00640	Michigan
	9.	0.00607	Auburn
	10.	0.00606	Gonzaga
	11.	0.00602	Houston
	12.	0.00585	Kansas
	13.	0.00570	Florida St
	14.	0.00533	Purdue
	15.	0.00529	LSU
	16.	0.00474	Kansas St
	17.	0.00443	Virginia Tech
	18.	0.00440	Iowa St
x	19.	0.00439	Cincinnati
	20.	0.00435	Buffalo
	21.	0.00427	Wisconsin
	22.	0.00421	Villanova
	23.	0.00398	Maryland
x	24.	0.00396	Texas
x	25.	0.00383	Marquette

1,563 teams

22,632 games

SCORE = w_∞

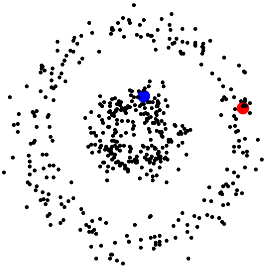
A CLASSIFICATION ALGORITHM

Semi-supervised learning

Imagine we have data with very few labels.

We want to use the structure in the dataset to help classify the unlabeled data.

We can do this with a Markov chain.



Semi-supervised learning uses partially labeled data to do classification.

- Many or most y_i will be missing in the pair (x_i, y_i).
- Still, there is structure in $x_1, \ldots, x_n$ that we don't want to throw away.
- In the example above, we might want the inner ring to be one class (blue) and the outer ring another (red).

A RANDOM WALK CLASSIFIER

We will define a classifier where, starting from any data point x_i,

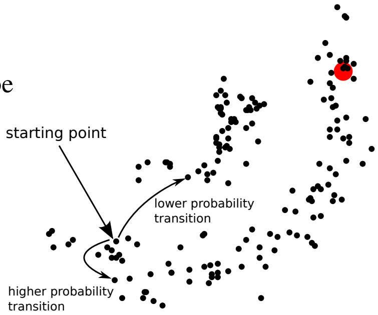
- A "random walker" moves around from point to point
- A transition between nearby points has higher probability
- A transition to a labeled point terminates the walk
- The label of a point x_i is the label of the terminal point

One possible random walk matrix

1. Let the *unnormalized* transition matrix be

$$\widehat{M}_{ij} = \exp\left\{-\frac{\|x_i - x_j\|^2}{b}\right\}$$

2. Normalize rows of $\widehat{M}$ to get M
3. If x_i has label y_i, re-define $M_{ii} = 1$



Property: Absorbing states

Imagine we have S states. If $p(s_t = i | s_{t-1} = i) = 1$, then the ith state is called an **absorbing state** since we can never leave it.

Q: Given initial state $s_0 = j$ and set of absorbing states $\{i_1, \ldots, i_k\}$, what is the probability a Markov chain terminates at a particular absorbing state?

- Aside: For the semi-supervised classifier, the answer gives the probability on the label of x_j.

A: Start a random walk at j and keep track of the distribution on states.

- w_0 is a vector of 0's with a 1 in entry j because we know $s_0 = j$
- If M is the transition matrix, we know that $w_{t+1} = w_t M$.
- So we want $w_\infty = w_0 M^\infty$.

Property: Absorbing state distribution

Group the absorbing states and break up the transition matrix into quadrants:

$$M = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix}$$

The bottom half contains the self-transitions of the absorbing states.

Observation: $w_{t+1} = w_t M = w_{t-1} M^2 = \cdots = w_0 M^{t+1}$

So we need to understand what's going on with M^t. For the first two we have

$$M^2 = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix} \begin{bmatrix} A & B \\ 0 & I \end{bmatrix} = \begin{bmatrix} A^2 & AB + B \\ 0 & I \end{bmatrix}$$

$$M^3 = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix} \begin{bmatrix} A^2 & AB + B \\ 0 & I \end{bmatrix} = \begin{bmatrix} A^3 & A^2B + AB + B \\ 0 & I \end{bmatrix}$$

Detour: We will use the matrix version of the following scalar equality.

Definition: Let $0 < r < 1$. Then $\sum_{u=0}^{t-1} r^u = \frac{1-r^t}{1-r}$ and so $\sum_{u=0}^{\infty} r^u = \frac{1}{1-r}$.

Proof: First define the top equality and create the bottom equality

$$\begin{array}{rcccccccccccc} C_t & = & 1 & + & r & + & r^2 & + & \cdots & + & r^{t-1} & & \\ r\,C_t & = & & & r & + & r^2 & + & \cdots & + & r^{t-1} & + & r^t \end{array}$$

and so

$$C_t - r\,C_t \;=\; 1 - r^t.$$

Therefore

$$C_t \;=\; \sum_{u=0}^{t-1} r^u \;=\; \frac{1-r^t}{1-r} \qquad \text{and} \qquad C_\infty \;=\; \frac{1}{1-r}.$$

PROPERTY: ABSORBING STATE DISTRIBUTION

A matrix version of the geometric series appears here. We see the pattern

$$M^t = \left[\begin{array}{cc} A^t & \left(\sum_{u=0}^{t-1} A^u\right) B \\ 0 & I \end{array} \right].$$

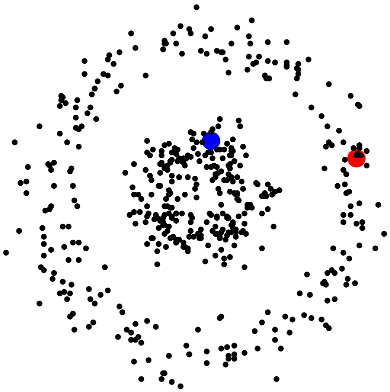
Two key things that can be shown are:

$$A^\infty = 0, \qquad \sum_{u=0}^{\infty} A^u = (I - A)^{-1}$$

Summary:

- After an infinite # of steps, $w_\infty = w_0 M^\infty = w_0 \left[\begin{array}{cc} 0 & (I-A)^{-1}B \\ 0 & I \end{array} \right]$.
- The non-zero dimension of w_0 picks out a row of $(I - A)^{-1}B$.
- The probability that a random walk started at x_j terminates at the ith absorbing state is $[(I - A)^{-1}B]_{ji}$.

Classification example



Using a Gaussian kernel normalized on the rows. The color indicates the distribution on the terminal state for each starting point.

Kernel width was tuned to give this result. If set too large, there would be some probability of jumping across the gap.