PROJECTING DATA ONTO ROAD NETWORK: MAP MATCHING

Justin Gould (gould29@purdue.edu)

HONR 39900: Foundations of Geospatial Analytics

Fall 2021

PURDUE UNIVERSITY

Topics

- Collecting GPS Data
  - Dirty Map Data
  - What is Map Matching?

- How to Map Match Raw GPS Data
  - Case Study: Uber and Hidden Markov Models
  - Solution via PostGIS

- A Brief Introduction to Hidden Markov Models:
  - Context: What are Markov Chains and Transition States?
  - Hidden Markov Models

- NEXT WEEK: Python implementation

Collecting GPS Data

- Sometimes GPS data are not as accurate as we would like:
  - Low sampling rate
  - Noise
  - Lack of granularity

WHAT THE NUMBER OF DIGITS IN YOUR COORDINATES MEANS

LAT/LON PRECISION	MEANING
28°N, 80°W	YOU'RE PROBABLY DOING SOMETHING SPACE-RELATED
28.5°N, 80.6°W	YOU'RE POINTING OUT A SPECIFIC CITY
28.52°N, 80.68°W	YOU'RE POINTING OUT A NEIGHBORHOOD
28.523°N, 80.683°W	YOU'RE POINTING OUT A SPECIFIC SUBURBAN CUL-DE-SAC
28.5234°N, 80.6830°W	YOU'RE POINTING TO A PARTICULAR CORNER OF A HOUSE
28.52345°N, 80.68309°W	YOU'RE POINTING TO A SPECIFIC PERSON IN A ROOM, BUT SINCE YOU DIDN'T INCLUDE DATUM INFORMATION, WE CAN'T TELL WHO
28.5234571°N, 80.6830941°W	YOU'RE POINTING TO WALDO ON A PAGE
28.523457182°N, 80.683094159°W	"HEY, CHECK OUT THIS SPECIFIC SAND GRAIN!"
28.523457182818284°N, 80.683094159265358°W	EITHER YOU'RE HANDING OUT RAW FLOATING POINT VARIABLES, OR YOU'VE BUILT A DATABASE TO TRACK INDIVIDUAL ATOMS. IN EITHER CASE, PLEASE STOP.

PURDUE UNIVERSITY

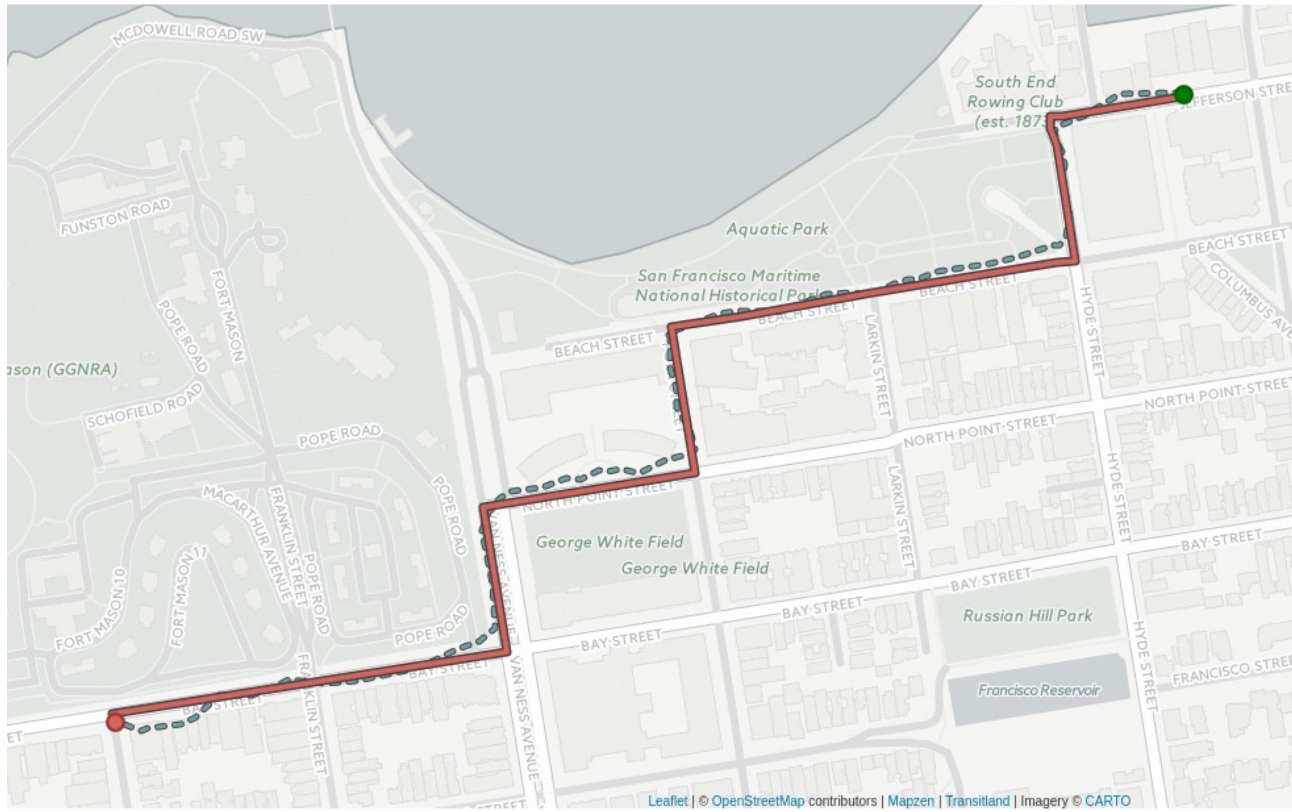
Is there anything Wrong with these Data?

Dirty Map Data

- Are GPS points clearly on the road network?
- Do GPS points stay on a single road/accurately depict where someone was driving?

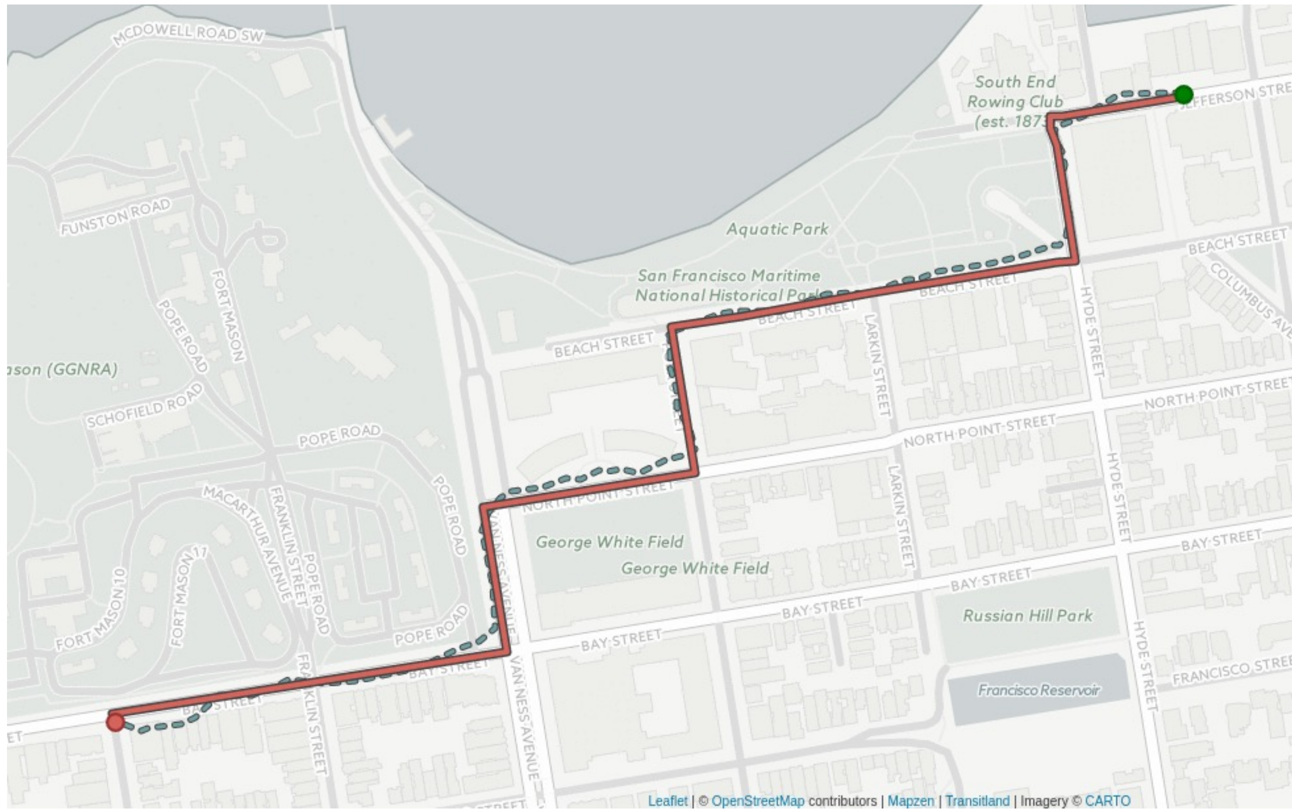
Dirty Map Data

- We need to project our GPS traces to a given road network, as depicted right.



What is Map Matching?

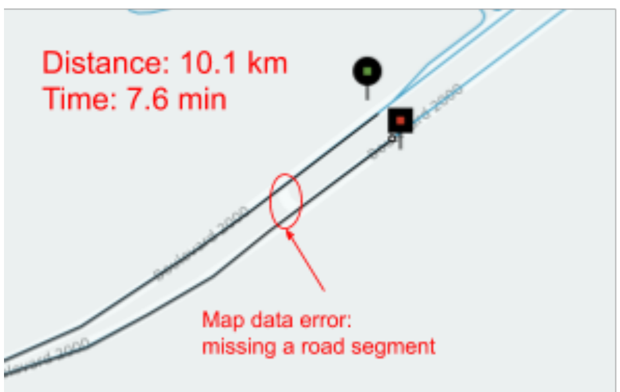
- This leads us to Map Matching!
  - The process taking input as raw GPS data and giving output as the actual position of road network.



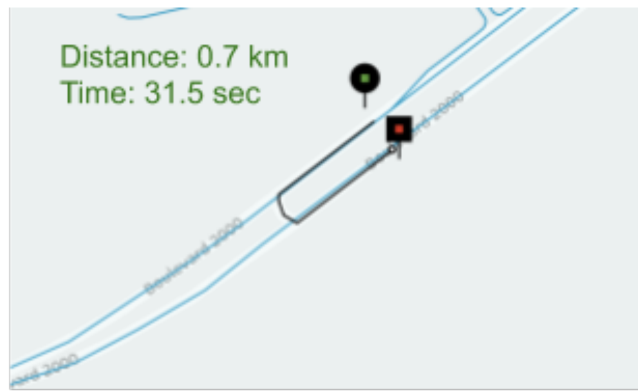
Why is Map Matching Important?

- Let's look at a ride-hailing example (e.g., Uber).

- When generating routes, Uber wants to minimize the distance and time to reach a destination:
  - Distance: Decreased wear on drivers' vehicles and reduction in fuel required to execute a trip.
  - Time: A driver can accomplish more rides in a shift and the customer gets to their destination faster.



(a) Suboptimal navigation instruction due to map data error.



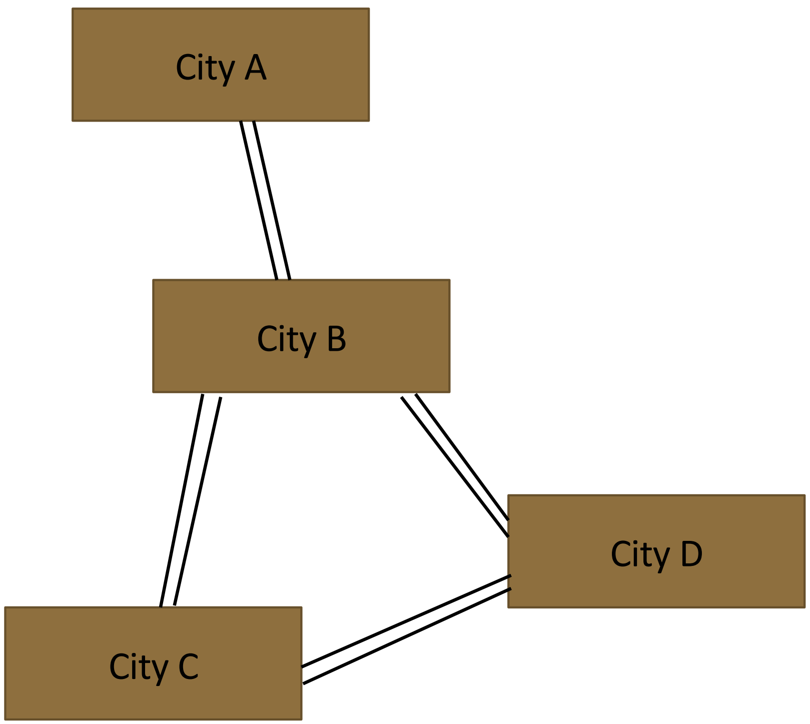
(b) Optimal navigation instruction as a result of fixing the map data error.

How to Map Match Raw GPS Data

- In this class, we will learn of two approaches to map matching:
  1. Hidden Markov Models
  2. PostGIS implementation

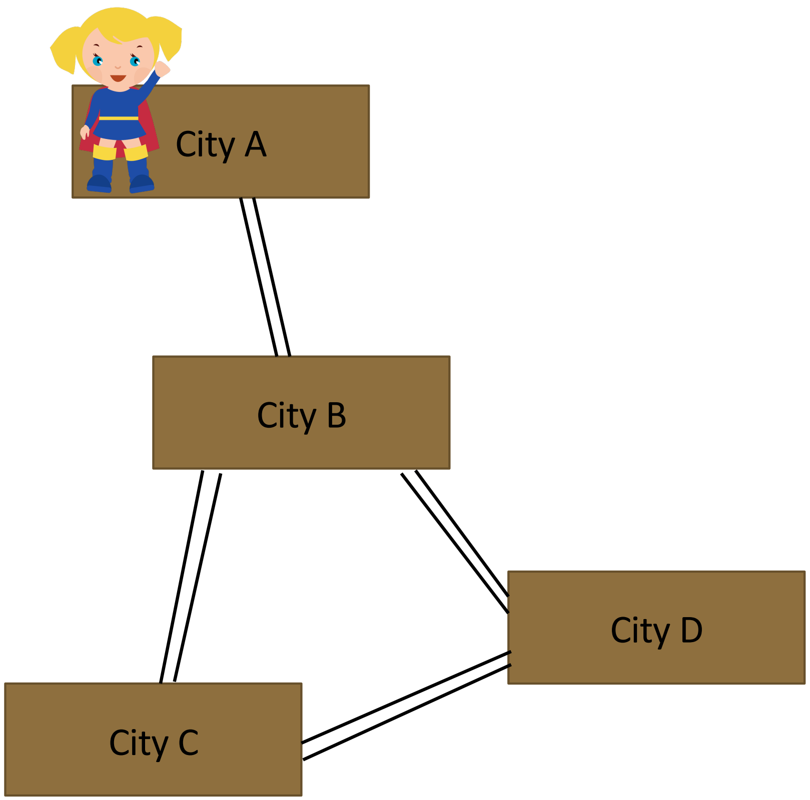
- Uber uses Hidden Markov Models as their map matching tool

Markov Chains Primer: Trivial Example



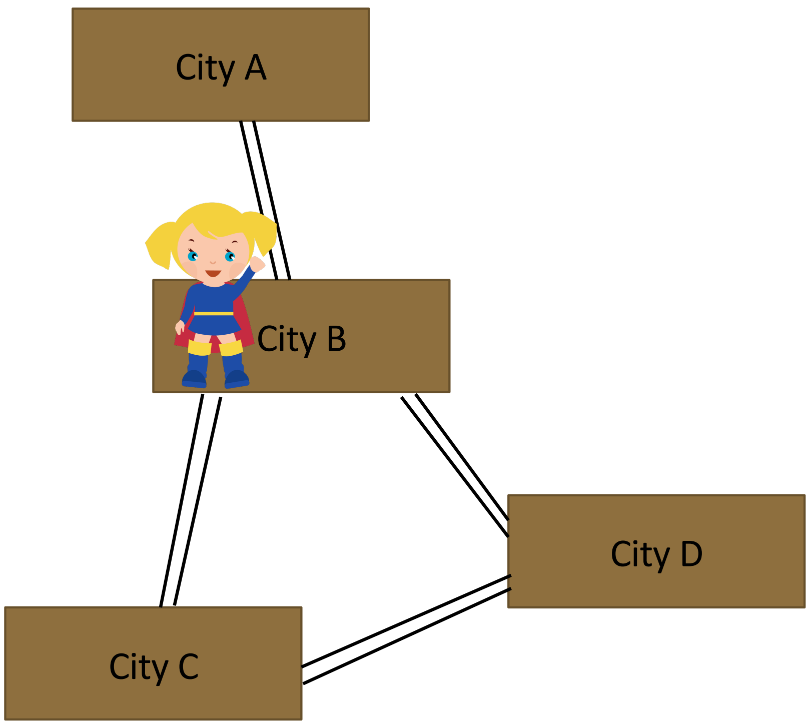
- We have 4 cities, each with a rail connection
- However, not every city has a direct connection to the others
- Unfortunately, these cities have a lot of crime…
  - But, we have a Markovian Superhero who can save us!
  - She has some limitations, though:
    1. Can only travel by rail
    2. She is a forgetful superhero, and doesn't know where she has been. Thus, to get between cities, she randomly chooses the next city to go to.

Markov Chains Primer: Trivial Example



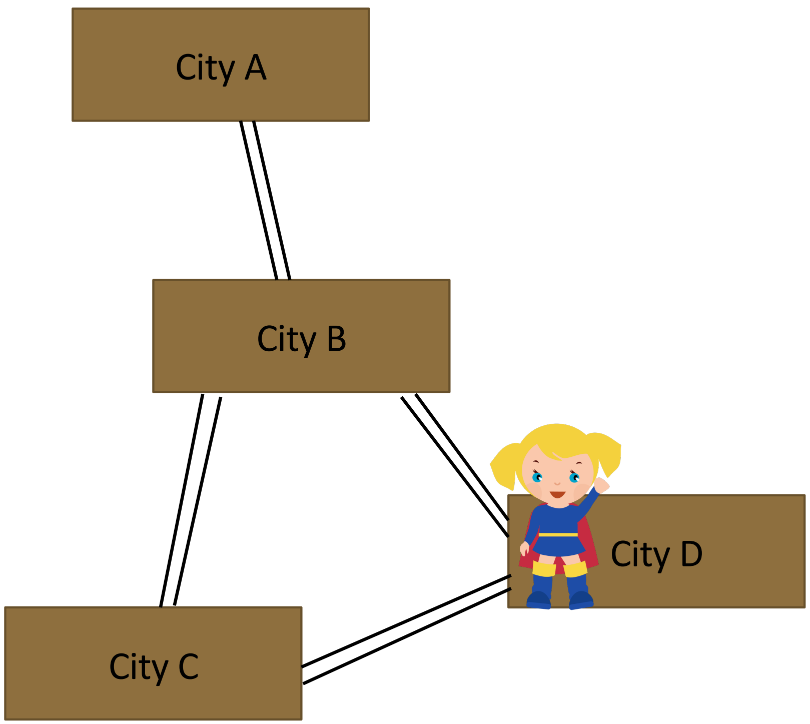
- Our superhero is at City A…there is only 1 option as to where she can travel: to City B

Markov Chains Primer: Trivial Example



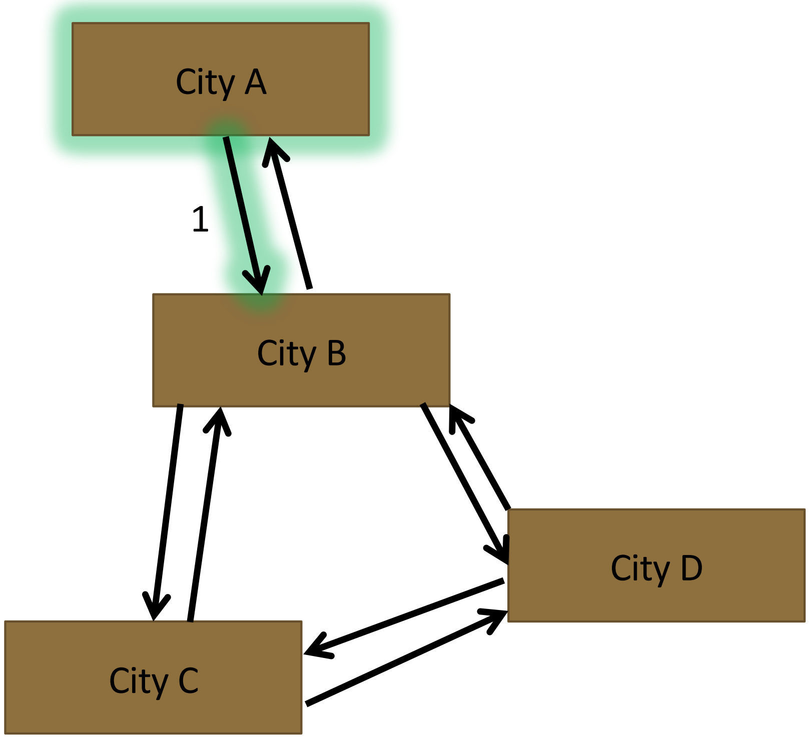
- Now, our Markovian Superhero is in City B. Remember, she has no memory of being in City A to fight crime. All she knows is that she is in City B.
- There are 2 options as to where she can go: City D or City C. She will randomly choose 1 of them.

Markov Chains Primer: Trivial Example



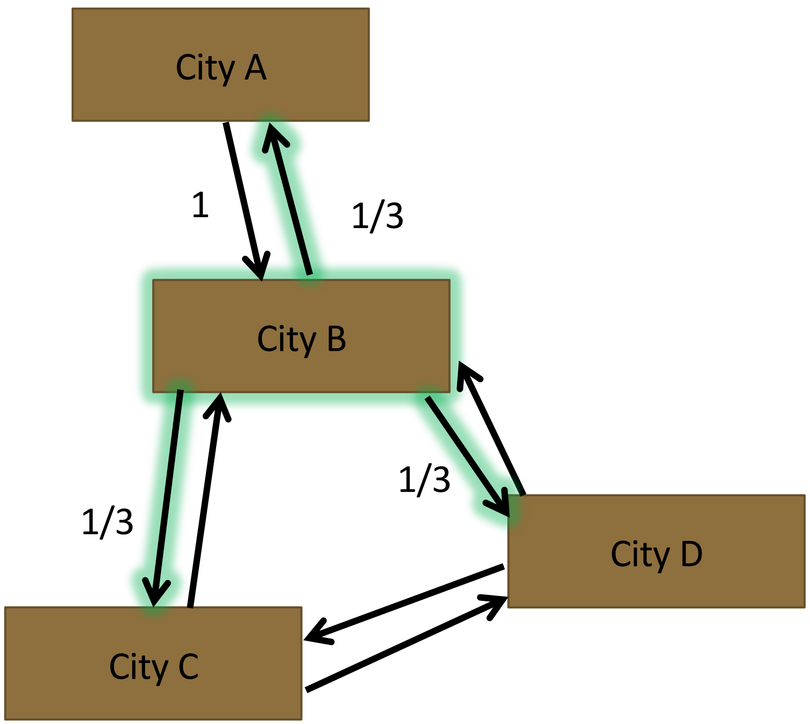
- In City D, the superhero faces the same problem…randomly choose to go to City B or City C.
- This is a *Markovian* example in which our superhero is always deciding the next step is just a random choice in where she currently is…
- A **Markov Chain** is a sequence of events such that the probabilities of the future depend ONLY on the present.

Markov Chains Primer: Transition States



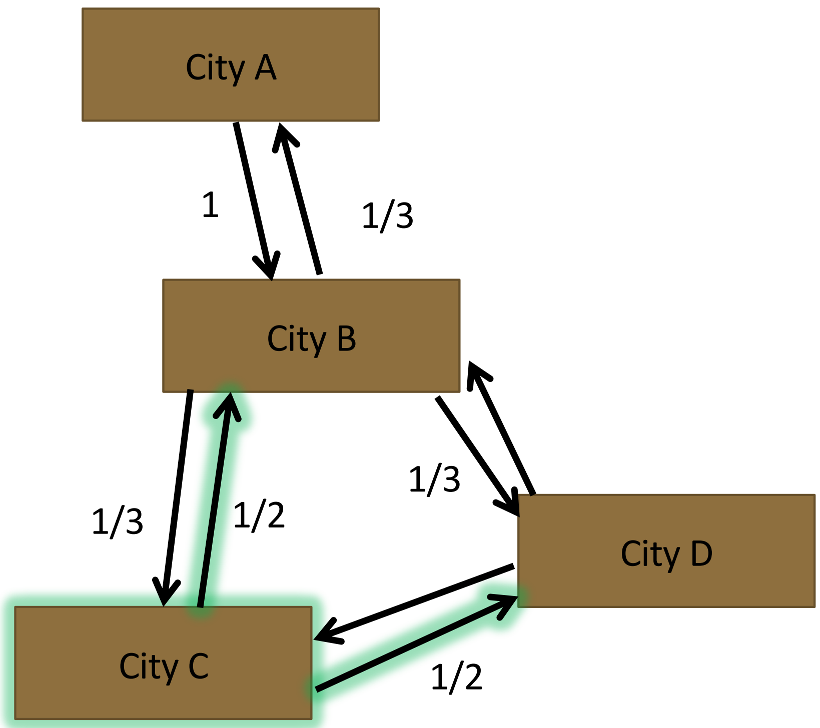
- What are the probabilities of traveling from one city to another?

Markov Chains Primer: Transition States



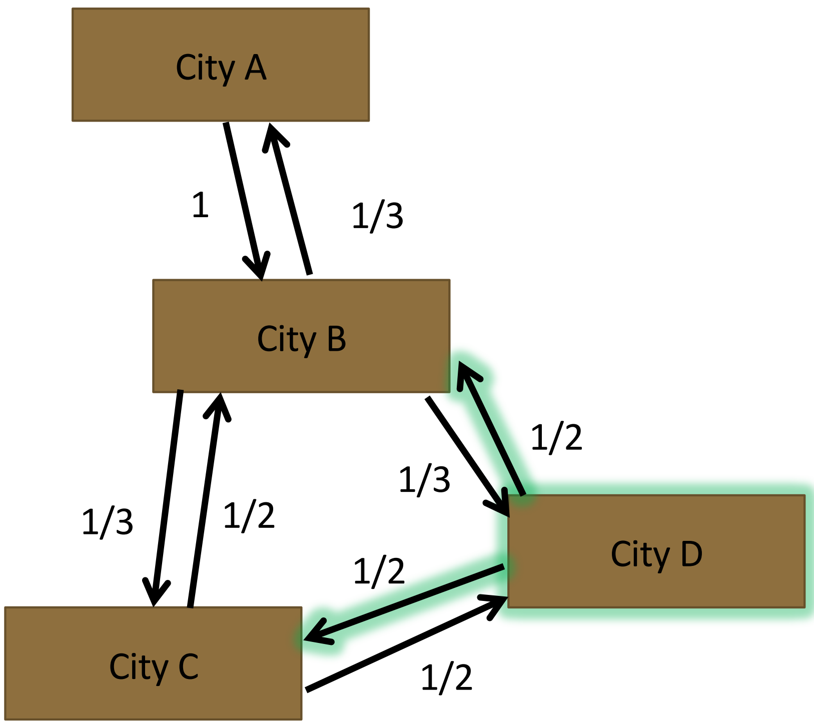
- What are the probabilities of traveling from one city to another?

Markov Chains Primer: Transition States



- What are the probabilities of traveling from one city to another?

Markov Chains Primer: Transition States



- What are the probabilities of traveling from one city to another?
- A **Transition Diagram** lists all the possible states and then lists the probability for every connection between possible states.

A More Serious Example: Stock Market

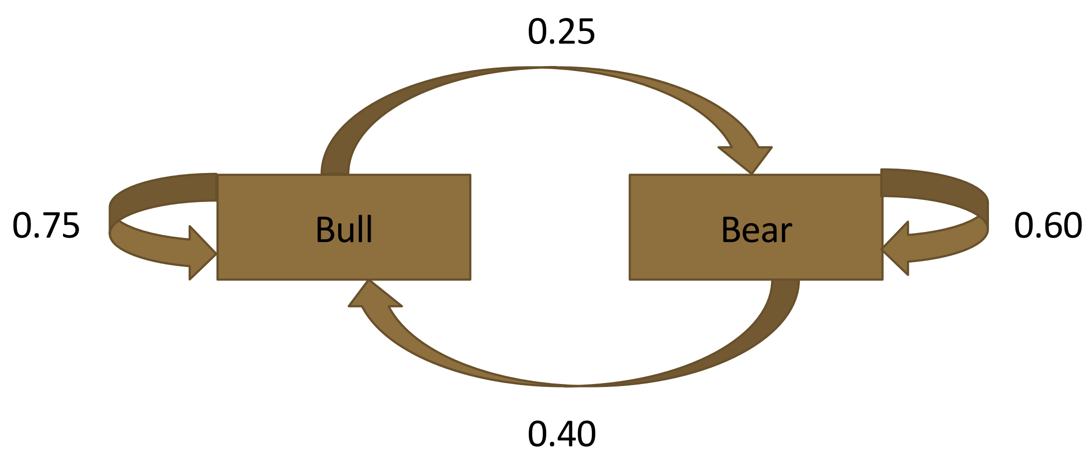
- Example: Stock Market
  - 75% a bull market followed by a bull market (going up)
  - 60% a bear market followed by a bear market (going down)

Bull

Bear

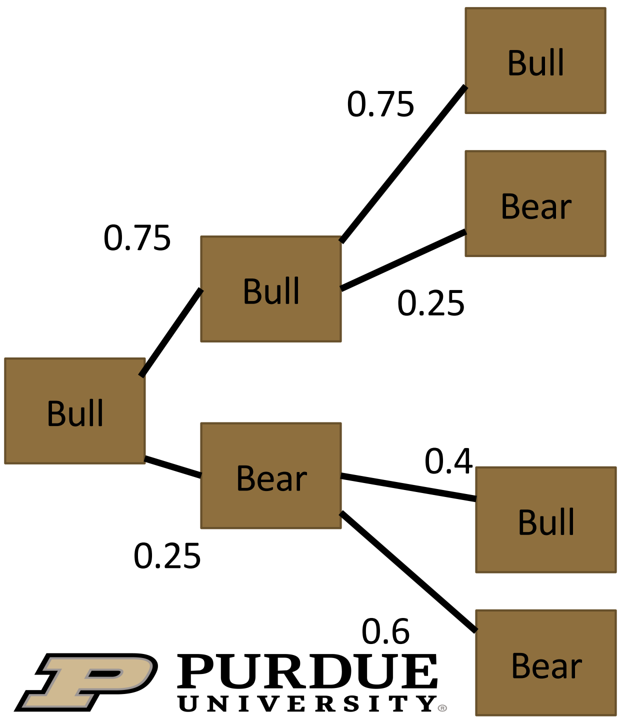
A More Serious Example: Stock Market

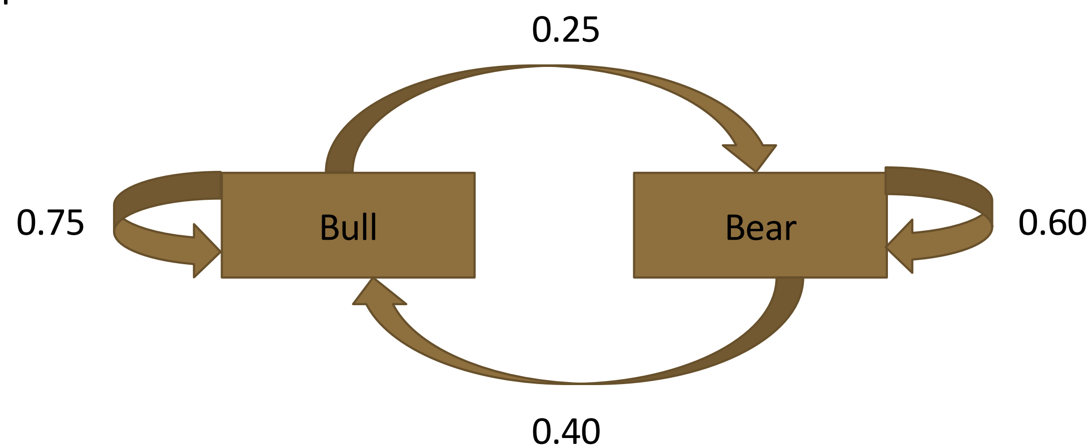
- Example: Stock Market
  - 75% a bull market followed by a bull market (going up)
  - 60% a bear market followed by a bear market (going down)



A More Serious Example: Stock Market

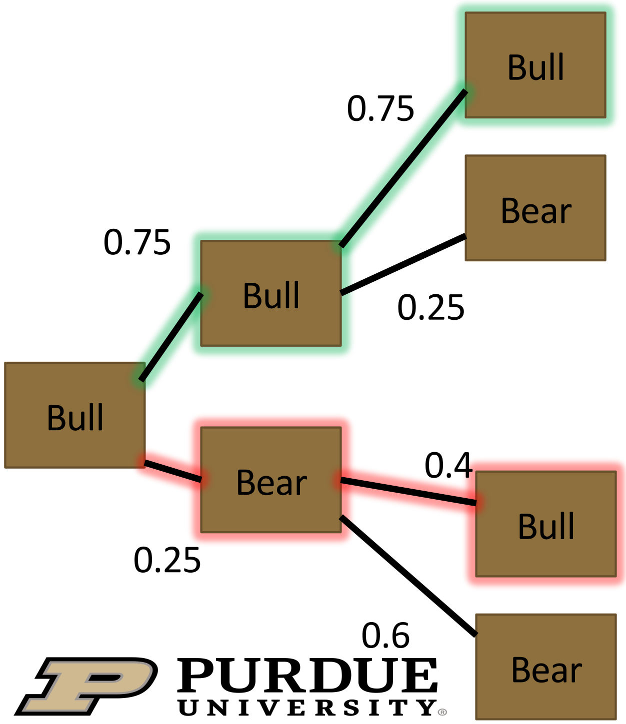
- If there is a bull market this week, what are the probabilities in TWO weeks?

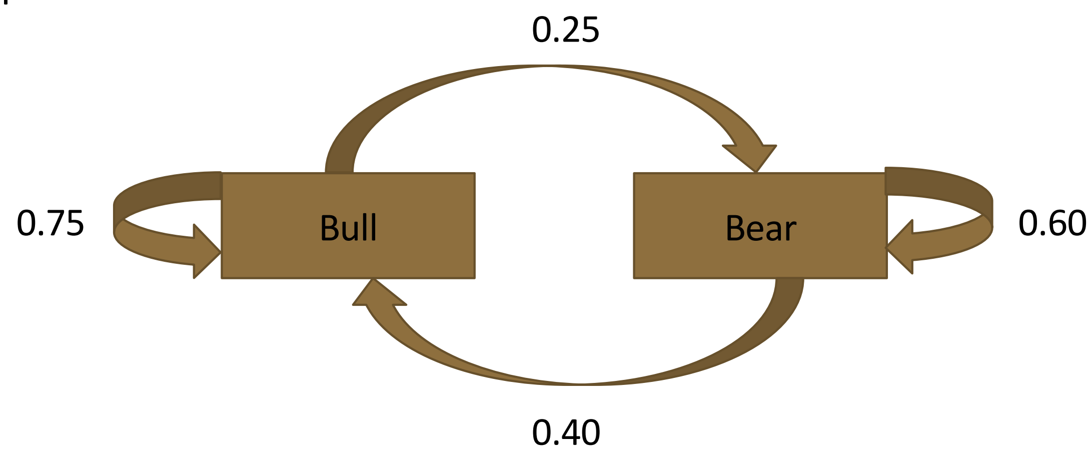




A More Serious Example: Stock Market

- If there is a bull market this week, what are the probabilities in TWO weeks?





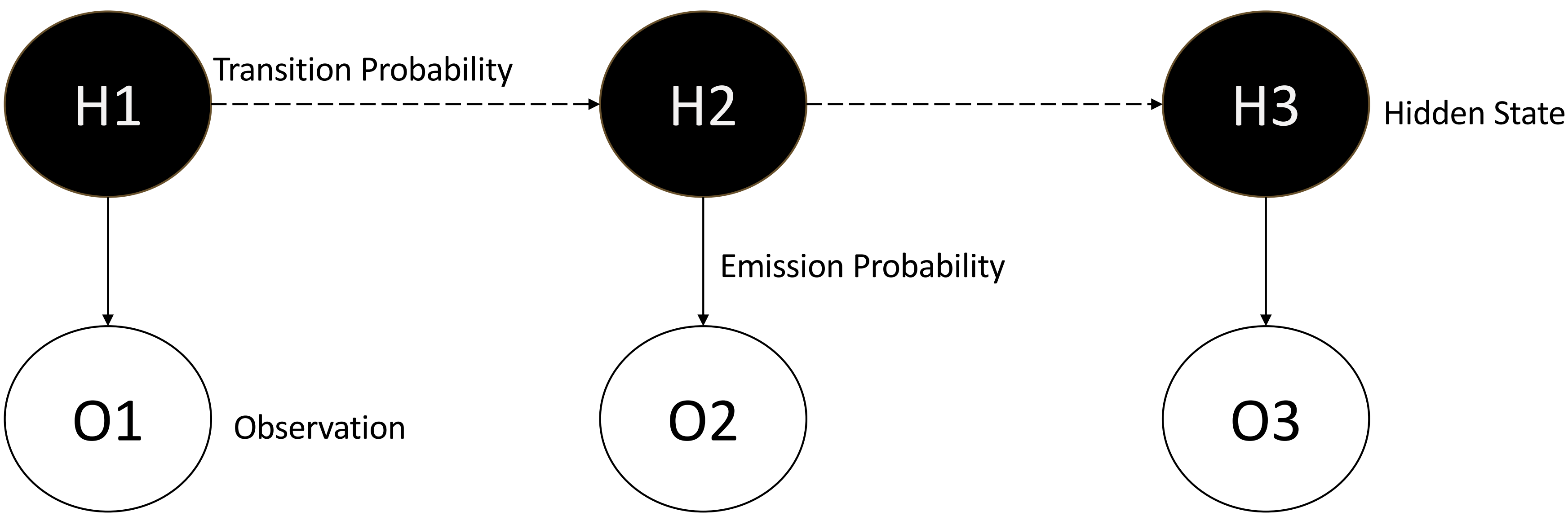
P(Bull Market)
= 0.75 * 0.75 + 0.25 * 0.40 = 0.66

Hidden Markov Models

PURDUE
UNIVERSITY

Hidden Markov Models

- Markov process + unobservable state
- Observations, which depend only on the current state, are visible



Hidden Markov Models

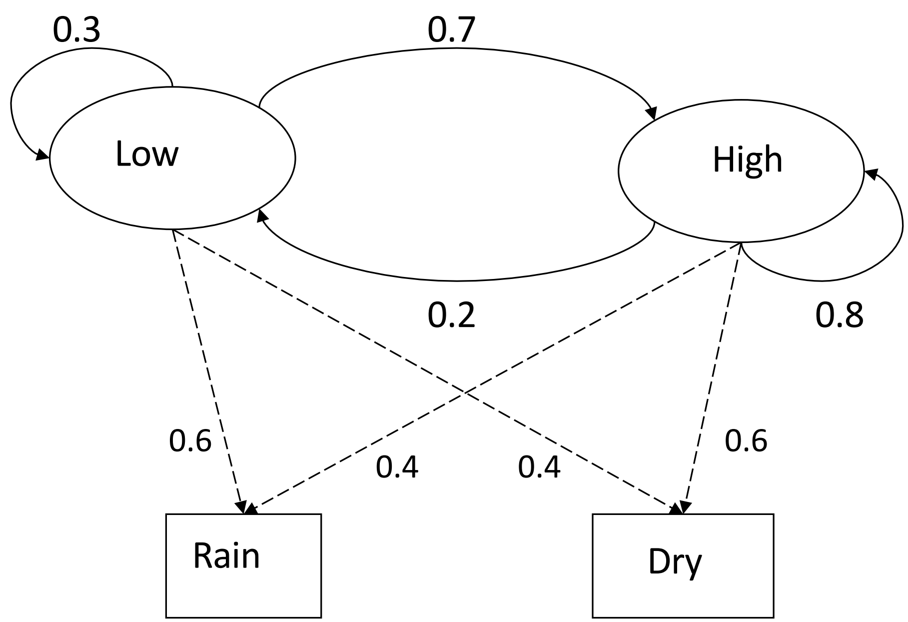
- Set of states: $\{s_1, s_2, \ldots, s_N\}$
- Process moves from one state to another generating a sequence of states : $s_{i1}, s_{i2}, \ldots, s_{ik}, \ldots$
- Markov chain property: probability of each subsequent state depends only on what was the previous state:

$$P(s_{ik} \mid s_{i1}, s_{i2}, \ldots, s_{ik-1}) = P(s_{ik} \mid s_{ik-1})$$

- States are not visible, but each state randomly generates one of M observations (or visible states) $\{v_1, v_2, \ldots, v_M\}$
- To define hidden Markov model, the following probabilities have to be specified: matrix of transition probabilities $A=(a_{ij})$, $a_{ij}= P(s_i \mid s_j)$, matrix of observation probabilities $B=(b_i(v_m))$, $b_i(v_m) = P(v_m \mid s_i)$ and a vector of initial probabilities $\pi=(\pi_i)$, $\pi_i = P(s_i)$. Model is represented by $M=(A, B, \pi)$.

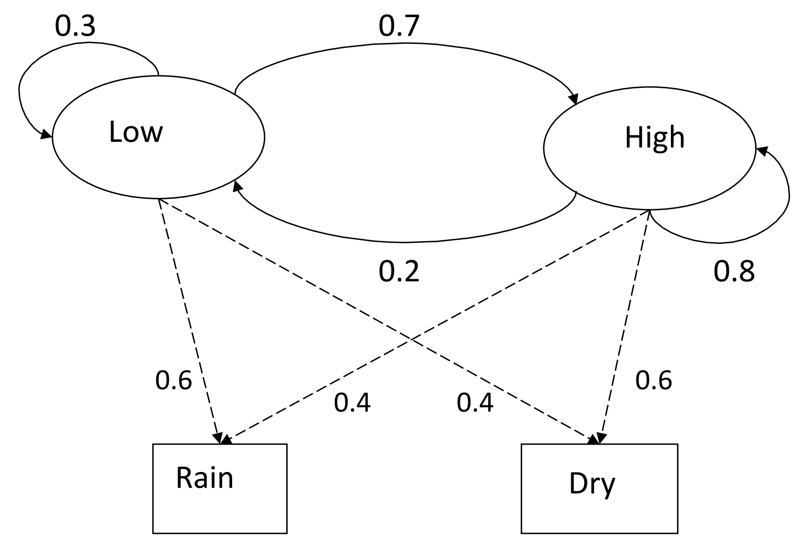
Hidden Markov Models

- Example



Hidden Markov Models

- Two states : ‘Low’ and ‘High’ atmospheric pressure.
- Two observations : ‘Rain’ and ‘Dry’.
- Transition probabilities: P(‘Low’|‘Low’)=0.3 , P(‘High’|‘Low’)=0.7 , P(‘Low’|‘High’)=0.2, P(‘High’|‘High’)=0.8
- Observation probabilities : P(‘Rain’|‘Low’)=0.6 , P(‘Dry’|‘Low’)=0.4 , P(‘Rain’|‘High’)=0.4 , P(‘Dry’|‘High’)=0.3 .
- Initial probabilities: say P(‘Low’)=0.4 , P(‘High’)=0.6



Hidden Markov Models

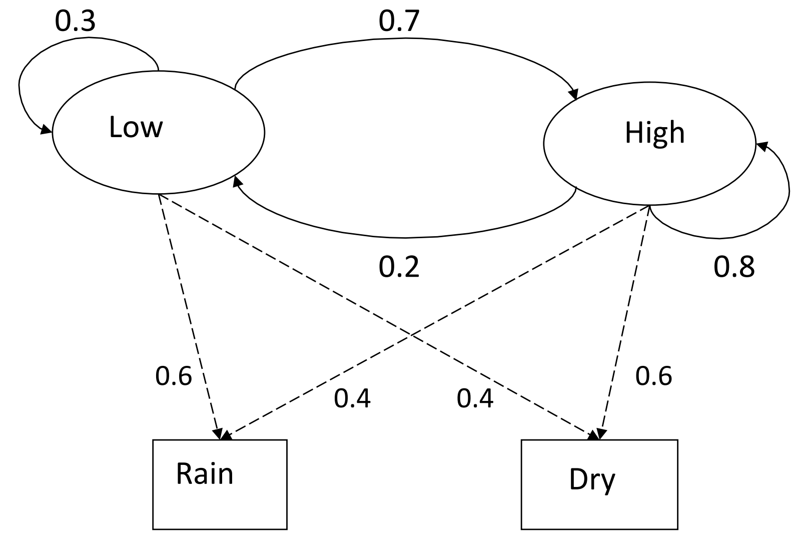
- Calculation of Observation Sequence Probability

- Suppose we want to calculate a probability of a sequence of observations in our example, {'Dry','Rain'}.
- Consider all possible hidden state sequences:

P({'Dry','Rain'}) = P({'Dry','Rain'} , {'Low','Low'}) + P({'Dry','Rain'} , {'Low','High'}) + P({'Dry','Rain'} , {'High','Low'}) + P({'Dry','Rain'} , {'High','High'})

- where first term is :

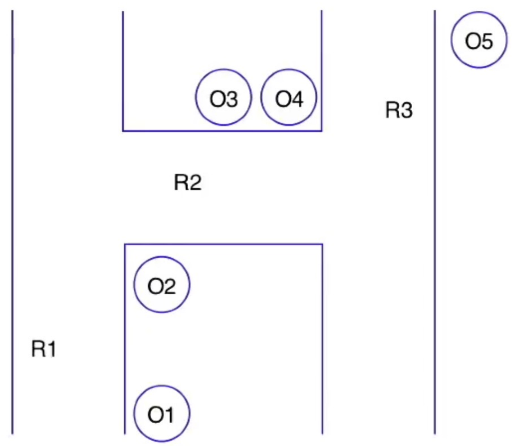
P({'Dry','Rain'} , {'Low','Low'})=
P({'Dry','Rain'} | {'Low','Low'}) P({'Low','Low'}) =
P('Dry'|'Low')P('Rain'|'Low') P('Low')P('Low'|'Low)
= 0.4*0.4*0.6*0.4*0.3

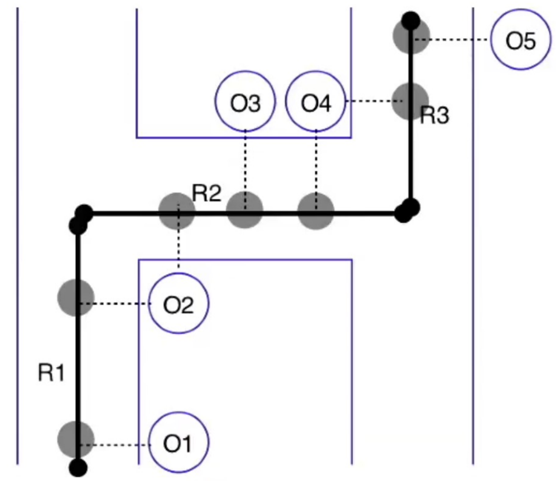


Map Matching via HMM

Map Matching via HMM

Input: O1, O2, O3, O4, O5

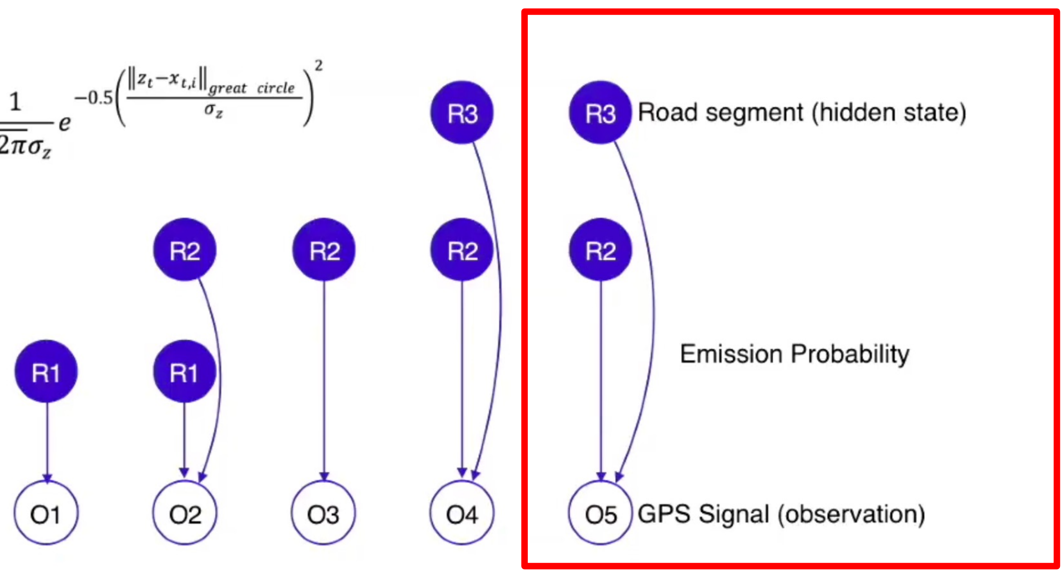




Map Matching via HMM

- Emission probability represents the likelihood of a vehicle present on certain road segments at certain moments

$$p(z_t|r_i) = \frac{1}{\sqrt{2\pi}\sigma_z} e^{-0.5\left(\frac{\|z_t - x_{t,i}\|_{great\ circle}}{\sigma_z}\right)^2}$$



Map Matching via HMM

- For one GPS point with *m* number of road segments nearby, there will be *m* emission probabilities representing the likelihood of this GPS trace on each road segment.
- For GPS points G1, which have *m* nearby segments, and G2, which has *n* nearby segments, there are *m* * *n* transition probabilities. These probabilities are in HMM… pick up a sequence of states with maximum probabilities that are most likely to represent road segments on which the vehicle was moving.

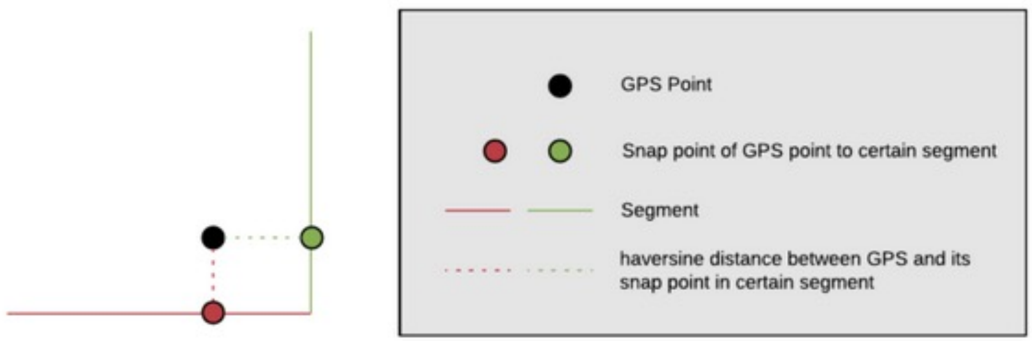


Figure 4: By calculating the haversine distance, we can determine the emission probability of a GPS point, the black point in this illustration, to different road segments, the red and green segments in this illustration.

$$P(GPS \mid Segment) = \frac{1}{\sigma\sqrt{2\pi}} e^{-0.5\left(\frac{Distance(GPS \mid Snap)}{\sigma}\right)^2}$$

Map Matching via HMM

- calculating transition probability regarding one GPS point on a certain segment to another GPS point on a certain segment, calculated using the following formula

$$P(GPSx \mid SegmentX \Rightarrow GPSy \mid SegmentY) = \frac{1}{\beta} e^{\frac{-DistanceDiff}{\beta}}$$

where

$$DistanceDiff = \left| HaversineDistance - RoutableDistance \right|$$

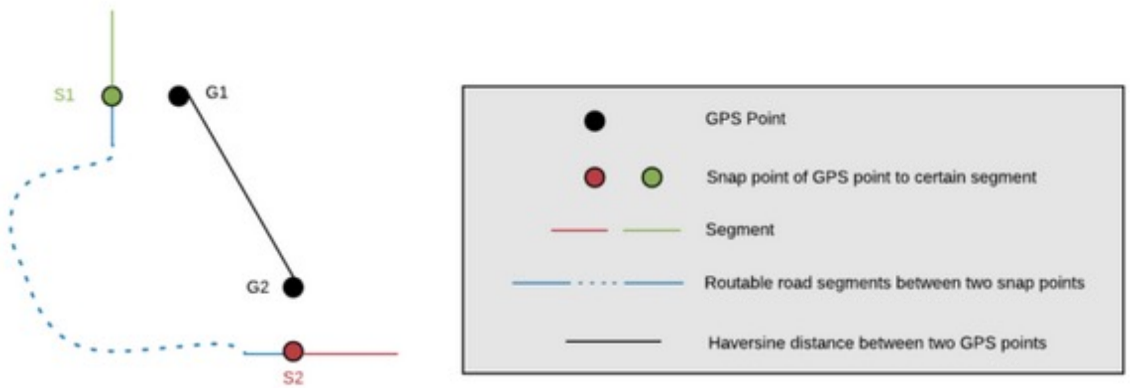


Figure 5: The transition probability from G1 to G2 is calculated by creating a route between their snap points, S1 to S2, and measuring the distance of that route.

Map Matching via HMM

- Transition probability represents the likelihood of a vehicle moving from one road segment to another road segment over a certain duration

$$p(d_t) = \frac{1}{\beta} e^{-d_t/\beta}$$

Here

$$d_t = \left| \|z_t - z_{t+1}\|_{great\ circle} - \|x_{t,i^*} - x_{t+1,j^*}\|_{route} \right|$$