MAP MATCHING PART 2

Justin Gould (gould29@purdue.edu)

HONR 39900: Foundations of Geospatial Analytics

Fall 2021

PURDUE UNIVERSITY

Topics

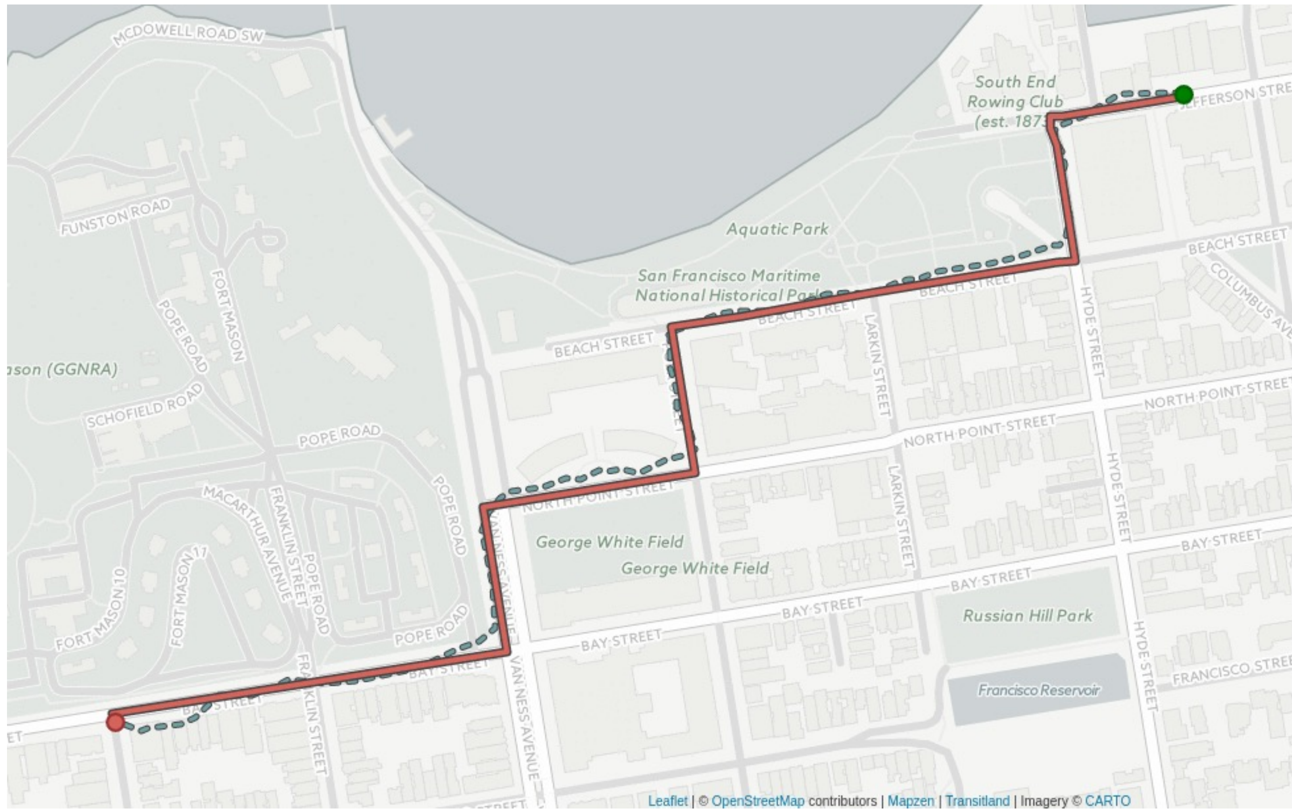
- Announcements
- Map Matching
  - Very Quick Review from Last Week
  - Map Matching via Python

Announcements

- I will remind you as it is closer, but we will not have class week 14, 2021/11/22 (week of Thanksgiving)
  - I will not hold formal office hours that week, but always email if you have questions.
- I advise starting to think about the final project. I have changed HW10 to be a final project proposal essay. This is due 2021/11/21 23:59 EST. Feedback will be provided before Thanksgiving to ensure you have enough time to work on your project. However, the sooner you get it in, the sooner I can give you feedback.

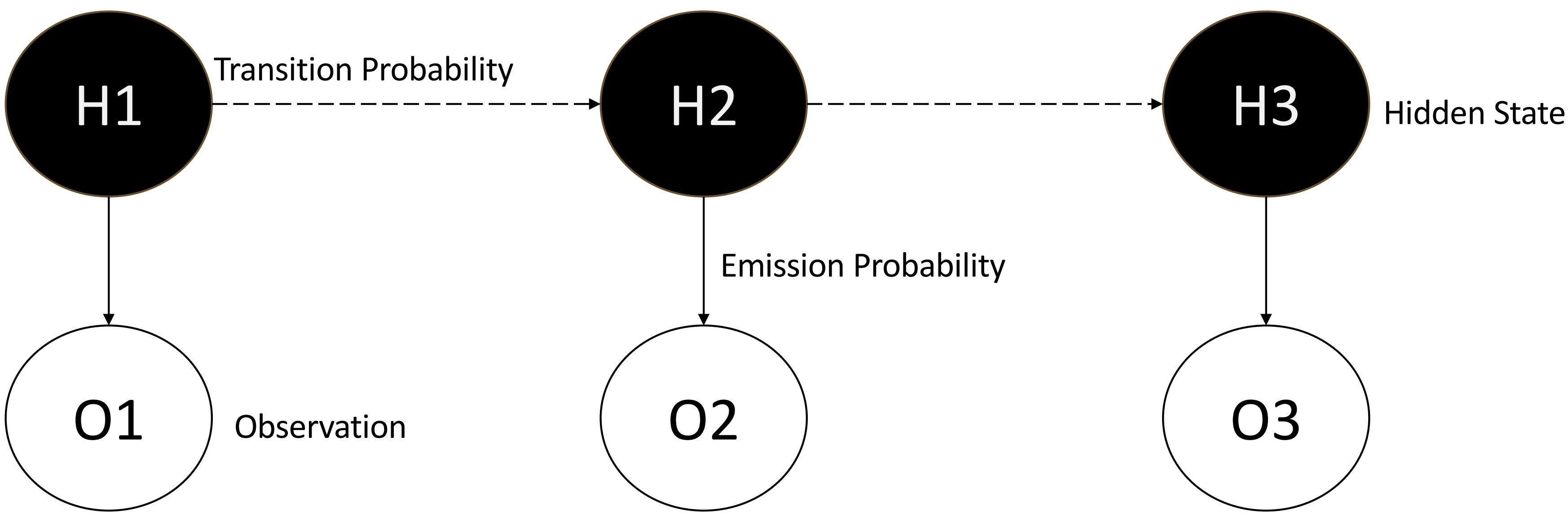
Quick Map Matching Recap

- The process taking input as raw GPS data and giving output as the actual position of road network.



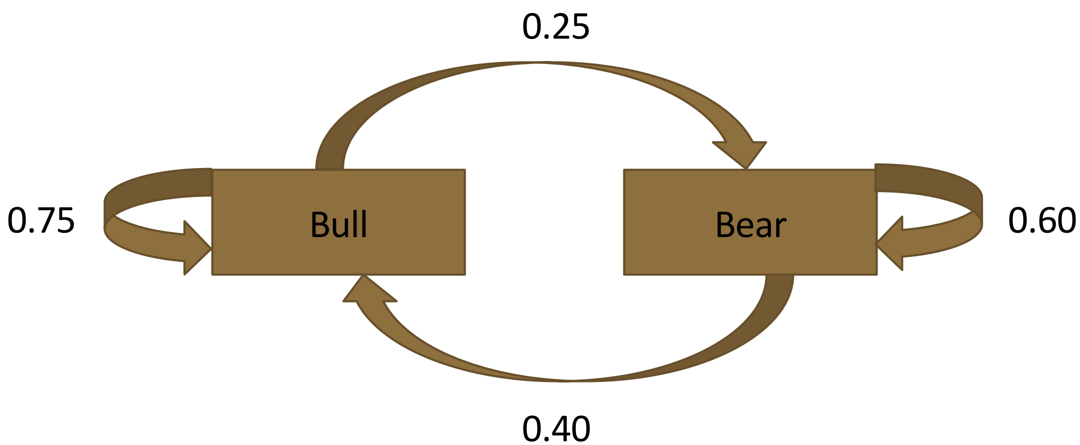
Hidden Markov Models

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



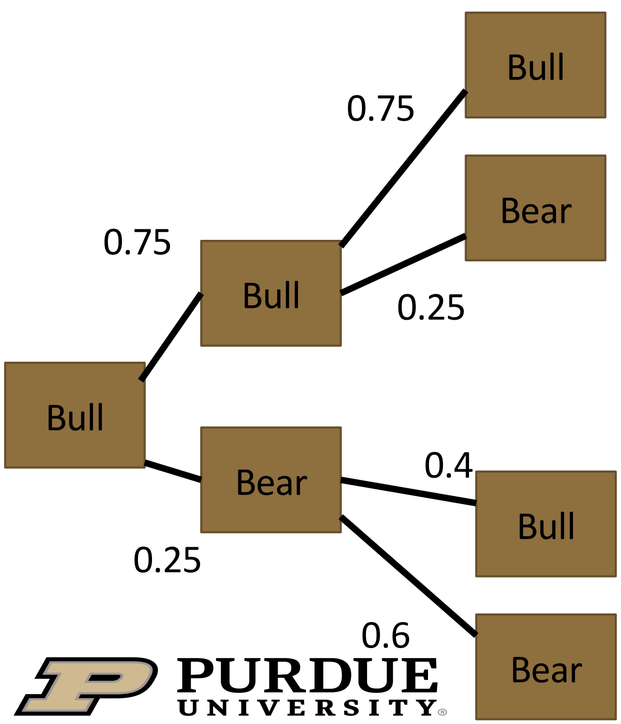
Markov Chains Reminder

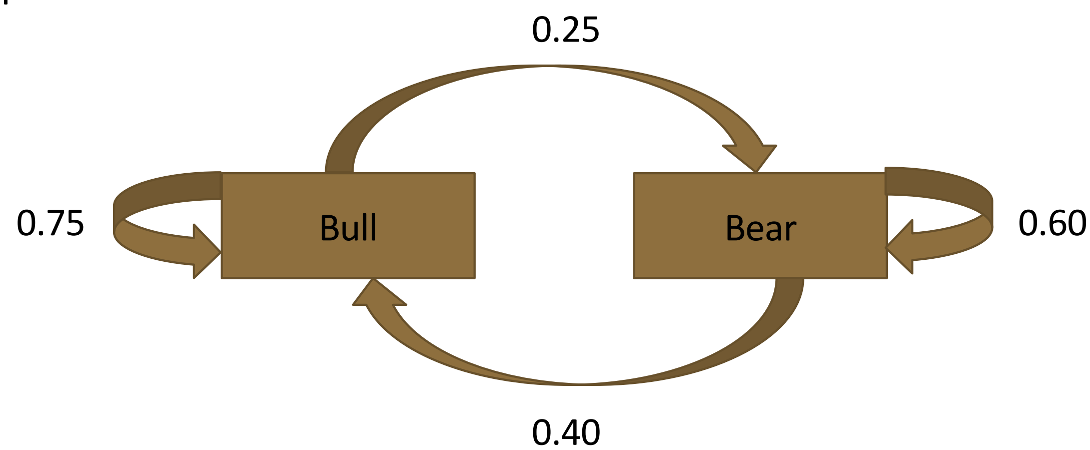
- Example: Stock Market
  - 75% a bull market followed by a bull market (going up)
  - 60% a bear market followed by a bear market (going down)



Markov Chains Reminder

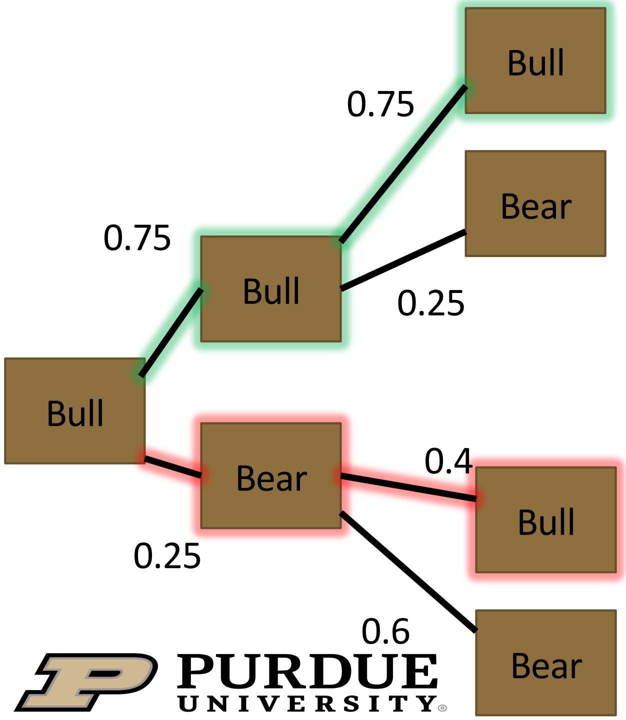
- If there is a bull market this week, what are the probabilities in TWO weeks?

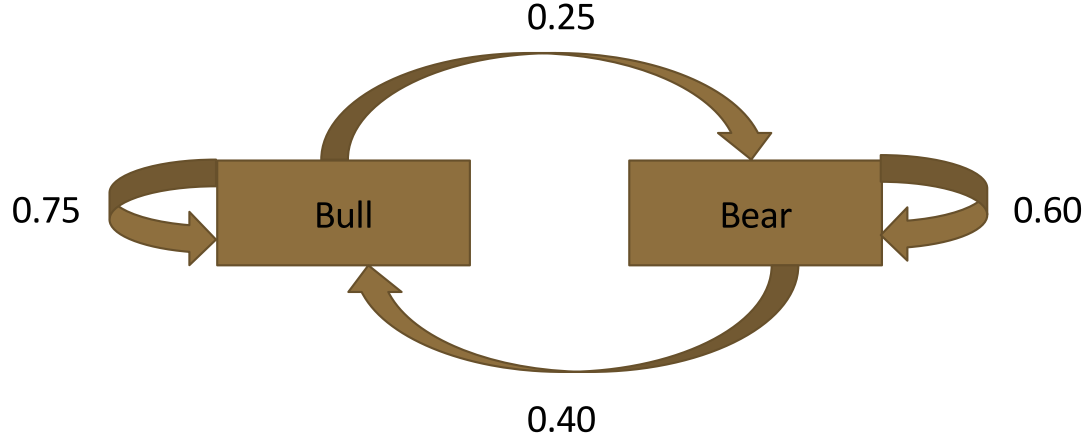




Markov Chains Reminder

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

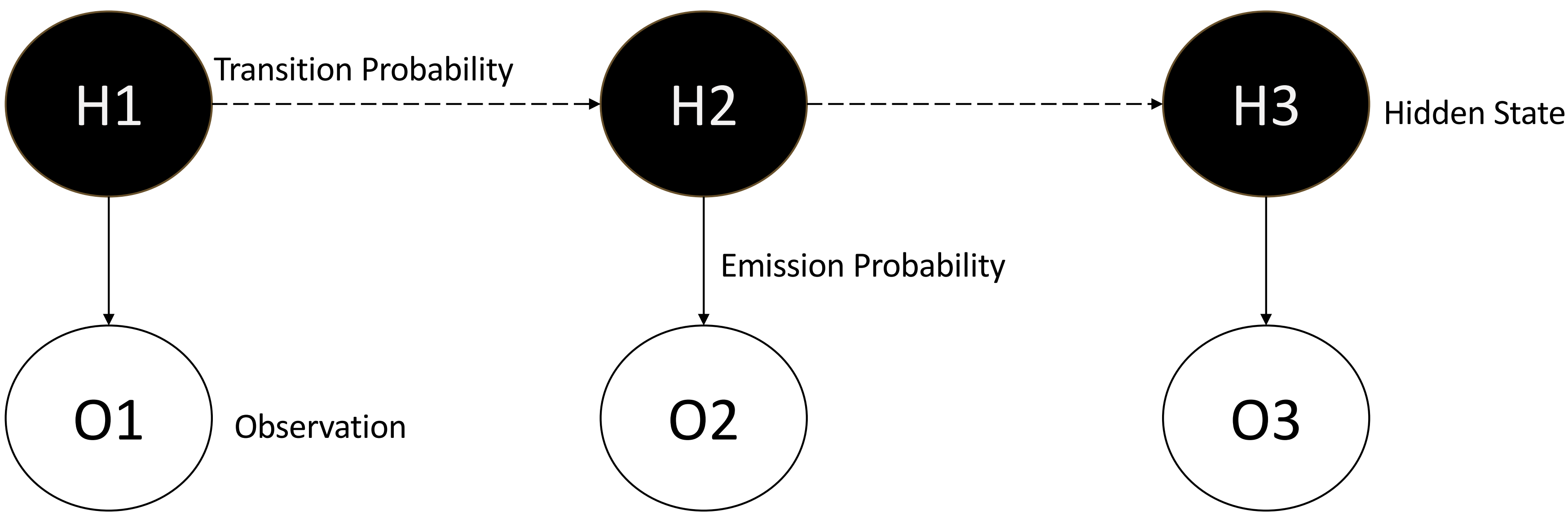




P(Bull Market)
= 0.75 * 0.75 + 0.25 * 0.40 = 0.66

Hidden Markov Models

- 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 $M=(A, B, \pi)$.

Map Matching via Python

- We will be using an out-of-the-box, quick method of map matching with HMMs: leuvenmapmatching (https://pypi.org/project/leuvenmapmatching/)
- Link to today's notebook: https://github.com/gouldju1/honr39900-foundations-of-geospatial-analytics/tree/master/Lectures/Week%2011