Attendance Code:
xxxx

Artificial Intelligence and Machine Learning (AIML)

2023–24

- **Last lecture**: Bayes' theorem, naive Bayes' classifier

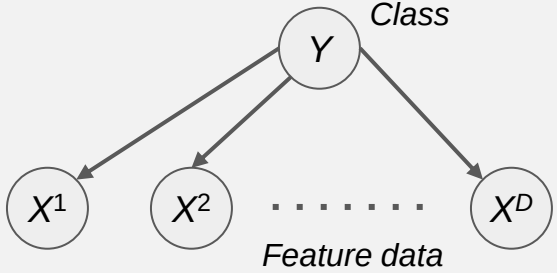
posterior → $P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)}$ **(Bayes' Theorem)**

- If evidence distribution $P(Y)$ is unknown, can use instead: $P(Y) = \sum_{x \in \Omega_X} P(Y|X = x)P(X = x)$

- **Naïve Bayes' Classifier**

$$P(X|Y) = P(X^1|Y)P(X^2|Y) \cdots P(X^D|Y)$$

$$y^\star = \arg\max_{y \in \Omega_Y} P(X^1|Y)P(X^2|Y) \cdots P(X^D|Y)P(Y = y)$$



#Code

- **Last lecture**: Bayes' theorem, naive Bayes' classifier

likelihood

prior

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)}$$ **(Bayes' Theorem)**

posterior

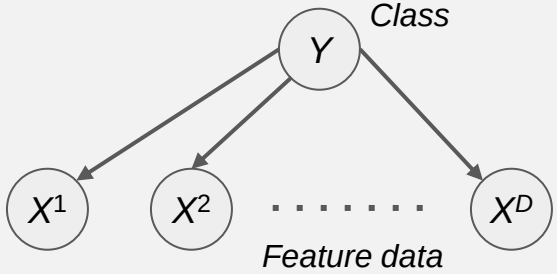
evidence

- If evidence distribution $P(Y)$ is unknown, can use instead: $P(Y) = \sum_{x \in \Omega_X} P(Y|X = x)P(X = x)$

- **Naïve Bayes' Classifier**

$$P(X|Y) = P(X^1|Y)P(X^2|Y) \cdots P(X^D|Y)$$

$$y^\star = \arg\max_{y \in \Omega_Y} P(X^1|Y)P(X^2|Y) \cdots P(X^D|Y)P(Y = y)$$



#Code

- **Last lecture**: Bayes' theorem, naive Bayes' classifier

  - **New email** containing words “friend” and “thank”
  - Is it likely to be a regular email or spam?
    - $y^\star = \arg\max_{y \in \Omega_Y} P(X^2|Y)P(X^3|Y)P(Y = y)$

Regular emails:
- dear: 8 out of 17 words - $P(X^1|Y = R) = \frac{8}{17} = 0.47$
- friend: 5 out of 17 words - $P(X^2|Y = R) = \frac{5}{17} = 0.29$
- thank: 3 out of 17 words - $P(X^3|Y = R) = \frac{3}{17} = 0.18$
- buy: 1 out of 17 words - $P(X^4|Y = R) = \frac{1}{17} = 0.06$

Spam emails:
- dear: 4 out of 17 words - $P(X^1|Y = S) = \frac{5}{17} = 0.24$
- friend: 2 out of 17 words - $P(X^2|Y = S) = \frac{2}{17} = 0.12$
- thank: 1 out of 17 words - $P(X^3|Y = R) = \frac{1}{17} = 0.06$
- buy: 10 out of 17 words - $P(X^4|Y = R) = \frac{10}{17} = 0.59$

- Many applications of ML involve **ordered data** that is, data for which the ordering matters
  - **natural language** (ordered sequences of words),
  - **appointment calendar entries** (date and time-ordered event names)
  - **electronic health records** (time-ordered sequences of medical system interactions)
  - **macroeconomic time series** (time-ordered sequences of GDP values)
  - **genomics** (base pairs in a genome sequence)

基因🧬

- **Last lecture**: Bayes' theorem, naive Bayes' classifier

  - **New email** containing words "friend" and "thank"
  - Is it likely to be a regular email or spam?
    - $y^\star = \arg\max_{y \in \Omega_Y} P(X^2|Y)P(X^3|Y)P(Y = y)$
      - $Y = r$: $\mathrm{p}(\mathrm{Y} = R|X) = 0.29 \times 0.18 \times 0.67 = 0.035$
      - $Y = s$: $\mathrm{p}(\mathrm{Y} = S|X) = 0.12 \times 0.06 \times 0.33 = 0.002$
    - $y^\star = r$ (not spam) Note that the result is the same regardless of the order of "thank" and "friend" in the email

Regular emails:
- dear: 8 out of 17 words - $P(X^1|Y = R) = \frac{8}{17} = 0.47$
- friend: 5 out of 17 words - $P(X^2|Y = R) = \frac{5}{17} = 0.29$
- thank: 3 out of 17 words - $P(X^3|Y = R) = \frac{3}{17} = 0.18$
- buy: 1 out of 17 words - $P(X^4|Y = R) = \frac{1}{17} = 0.06$

Spam emails:
- dear: 4 out of 17 words - $P(X^1|Y = S) = \frac{5}{17} = 0.24$
- friend: 2 out of 17 words - $P(X^2|Y = S) = \frac{2}{17} = 0.12$
- thank: 1 out of 17 words - $P(X^3|Y = R) = \frac{1}{17} = 0.06$
- buy: 10 out of 17 words - $P(X^4|Y = R) = \frac{10}{17} = 0.59$

- Many applications of ML involve **ordered data** that is, data for which the ordering matters
  - **natural language** (ordered sequences of words),
  - **appointment calendar entries** (date and time-ordered event names)
  - **electronic health records** (time-ordered sequences of medical system interactions)
  - **macroeconomic time series** (time-ordered sequences of GDP values)
  - **genomics** (base pairs in a genome sequence)

#Code

- **Last lecture**: Bayes' theorem, naive Bayes' classifier

  - **New email** containing words "friend" and "thank"
  - Is it likely to be a regular email or spam?
    - $y^\star = \arg\max_{y \in \Omega_Y} P(X^2|Y)P(X^3|Y)P(Y = y)$
      - $Y = r$: $\mathrm{p}(\mathrm{Y} = R|X) = 0.29 \times 0.18 \times 0.67 = 0.035$
      - $Y = s$: $\mathrm{p}(\mathrm{Y} = S|X) = 0.12 \times 0.06 \times 0.33 = 0.002$
    - ○ $y^\star = r$ (not spam) Note that the result is the same regardless of the order of "thank" and "friend" in the email

Regular emails:
- dear: 8 out of 17 words - $P(X^1|Y = R) = \frac{8}{17} = 0.47$
- friend: 5 out of 17 words - $P(X^2|Y = R) = \frac{5}{17} = 0.29$
- thank: 3 out of 17 words - $P(X^3|Y = R) = \frac{3}{17} = 0.18$
- buy: 1 out of 17 words - $P(X^4|Y = R) = \frac{1}{17} = 0.06$

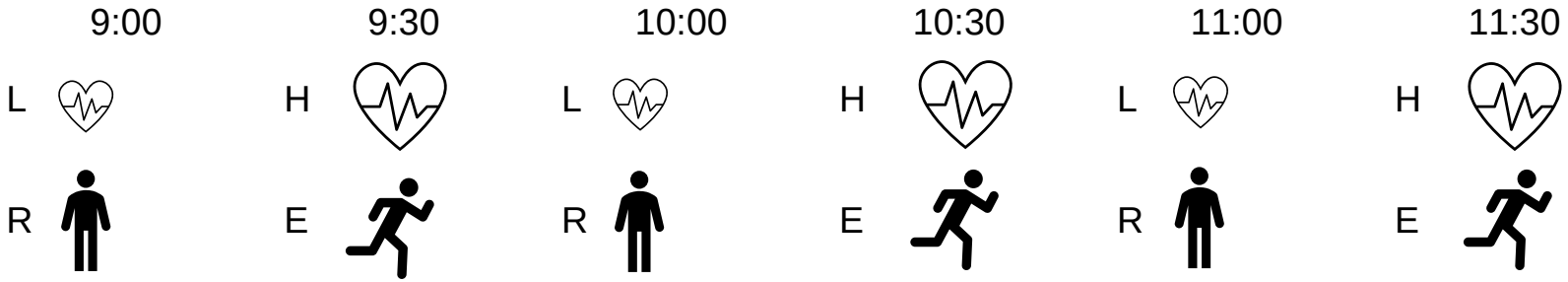
Spam emails:
- dear: 4 out of 17 words - $P(X^1|Y = S) = \frac{5}{17} = 0.24$
- friend: 2 out of 17 words - $P(X^2|Y = S) = \frac{2}{17} = 0.12$
- thank: 1 out of 17 words - $P(X^3|Y = R) = \frac{1}{17} = 0.06$
- buy: 10 out of 17 words - $P(X^4|Y = R) = \frac{10}{17} = 0.59$

- **This lecture**: sequence modelling, hidden Markov models

Sequence modelling: intuition

#Code

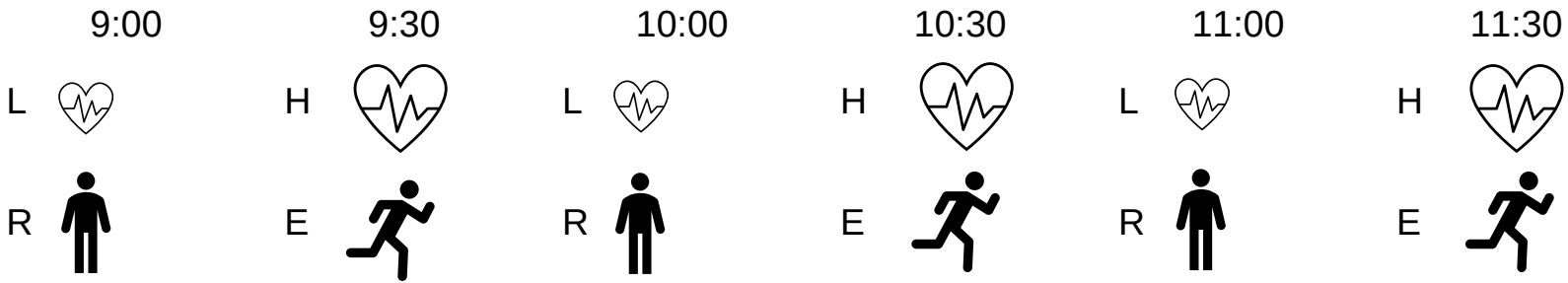
- **Problem**: Smartwatch-based Activity Monitoring System
- **Measured observation (X_t):** heart rate (high vs low)
  - $X_t \in \Omega_X = \{h, l\}$



#Code

Sequence modelling: intuition

- **Problem**: Smartwatch-based Activity Monitoring System
- **Measured observation (X_t):** heart rate (high vs low)
  - $X_t \in \Omega_X = \{h, l\}$
- **Inferred observation (Y_t):** activity (rest vs. exercise)
  - $Y_t \in \Omega_Y = \{r, e\}$
  - It is a **hidden state** (not directly observable)



AIML

Sequence modelling: intuition

#Code

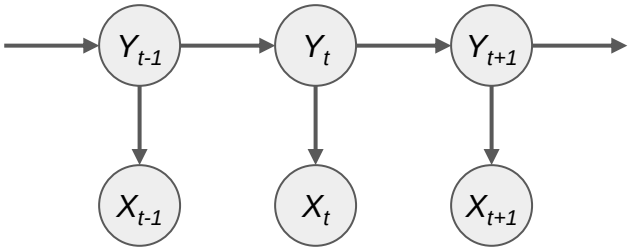
- **Problem**: Smartwatch-based Activity Monitoring System
- **Measured observation (X_t):** heart rate (high vs low)
  - $X_t \in \Omega_X = \{h, l\}$
- **Inferred observation (Y_t):** activity (rest vs. exercise)
  - $Y_t \in \Omega_Y = \{r, e\}$
  - It is a **hidden state** (not directly observable)

- **Assumption:**
  - we aren't randomly on rest or exercise;
  - If we are at rest at a given time, it's likely we will continue at rest

#Code

Sequence modelling: Hidden Markov Models (HMMs)

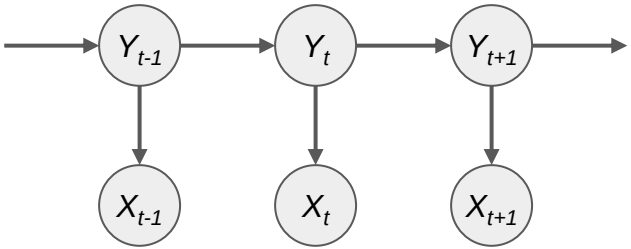
- The **hidden Markov model** (HMM) captures time-dependent RVs which are not directly measured



#Code

Sequence modelling: Hidden Markov Models (HMMs)

- The **hidden Markov model** (HMM) captures time-dependent RVs which are not directly measured
- Each **hidden states** $Y_t \in \Omega_Y$ with K distinct values, depends only upon the one before it in time, Y_{t-1} for all $t = 0,1,\ldots, T$
- The measured **observations** X_t depend only upon the associated hidden state, Y_t

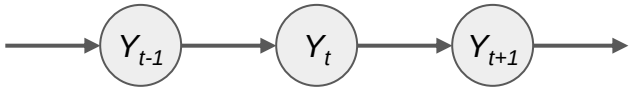


AIML

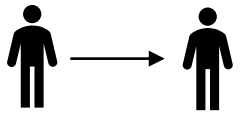
Sequence modelling: model fitting

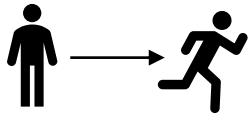
#Code

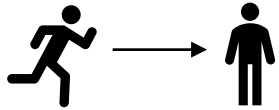
- given observed data for $X_0, X_1, \ldots, X_T$, estimate the distribution functions $P(X_t|Y_t)$, $P(Y_t|Y_{t-1})$

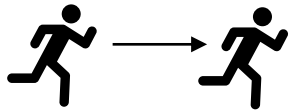


- Training data (**transition probabilities**)





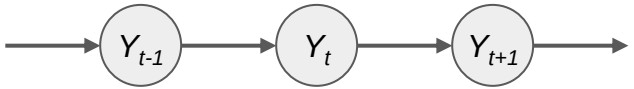




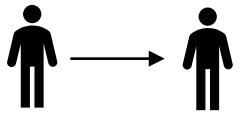
Sequence modelling: model fitting

#Code

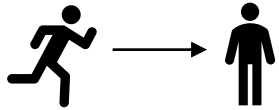
- given observed data for $X_0, X_1, \ldots, X_T$, estimate the distribution functions $P(X_t|Y_t),\ P(Y_t|Y_{t-1})$



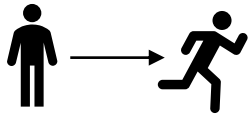
- Training data (**transition probabilities**)



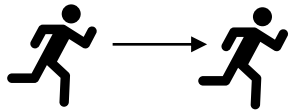
$$P(Y_t = r|Y_{t-1} = r) = \frac{8}{10} = 0.8$$



$$P(Y_t = r|Y_{t-1} = e) = \frac{2}{5} = 0.4$$



$$P(Y_t = e|Y_{t-1} = r) = \frac{2}{10} = 0.2$$

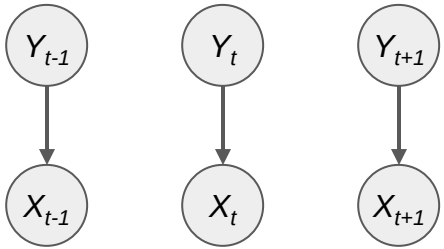


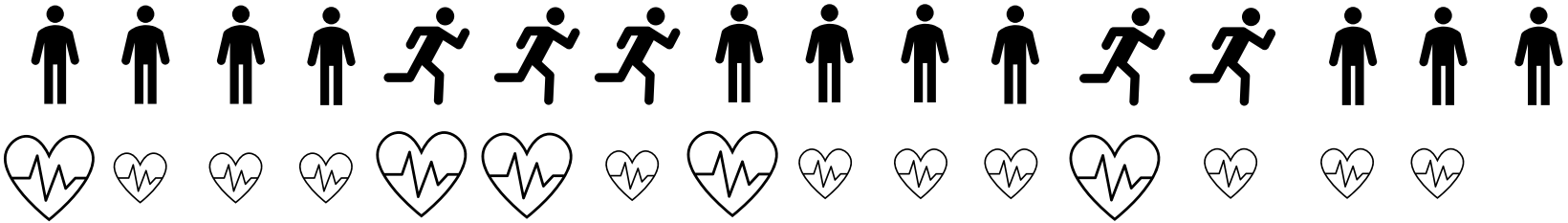
$$P(Y_t = e|Y_{t-1} = e) = \frac{3}{5} = 0.6$$

#Code

Sequence modelling: model fitting

- given observed data for $X_0, X_1, \ldots, X_T$, estimate the distribution functions $P(X_t|Y_t),\ P(Y_t|Y_{t-1})$
- Training data (**emission probabilities**)

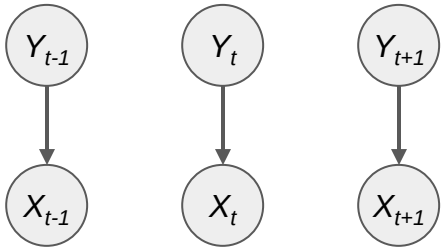


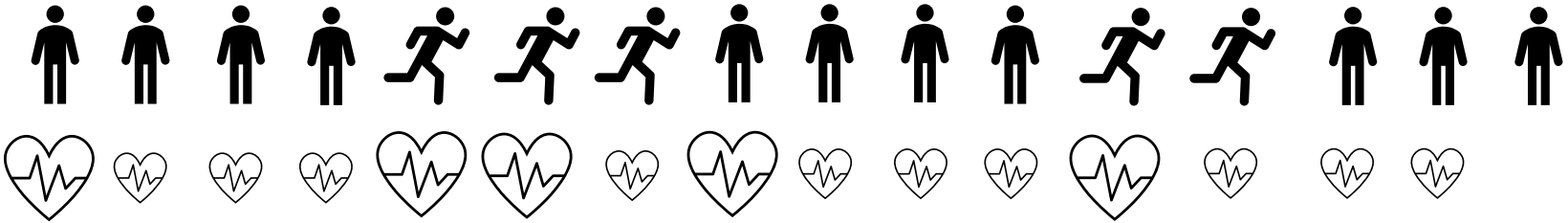


Sequence modelling: model fitting

#Code

- given observed data for $X_0, X_1, \ldots, X_T$, estimate the distribution functions $P(X_t|Y_t),\ P(Y_t|Y_{t-1})$
- Training data (**emission probabilities**)





$$P(X_t = l|Y_t = r) = \frac{8}{10} = 0.8$$

$$P(X_t = h|Y_t = r) = \frac{2}{10} = 0.2$$

$$P(X_t = l|Y_t = e) = \frac{2}{5} = 0.4$$

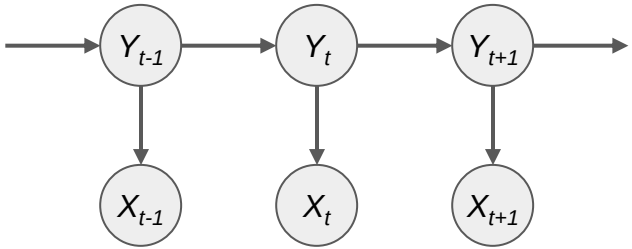
$$P(X_t = h|Y_t = e) = \frac{3}{5} = 0.6$$

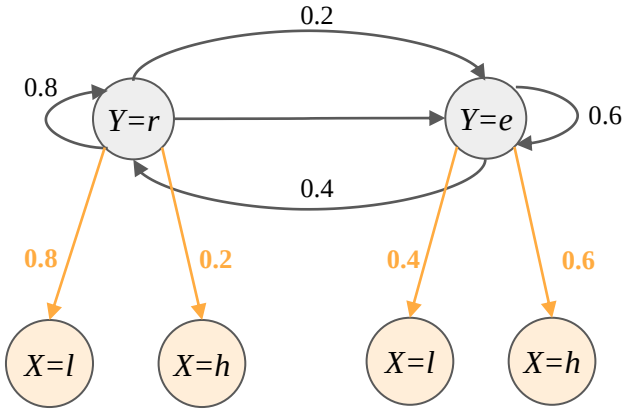
AIML

Sequence modelling: model fitting

#Code

- Given observed data for $X_0, X_1, \ldots, X_T$, estimate the distribution functions $P(X_t|Y_t)$, $P(Y_t|Y_{t-1})$
- Training data



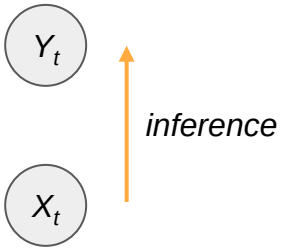


AIML

Sequence modelling: single evaluation

#Code

- Given fixed model parameters and observed data, compute the probability of the hidden state



- If we currently measured heart rate to be low, what's the probability that the user is at rest or exercising?

$$P(Y = r) = \frac{10}{15} = \frac{2}{3} = 0.67$$

$$P(Y = e) = \frac{5}{15} = \frac{1}{3} = 0.33$$

AIML

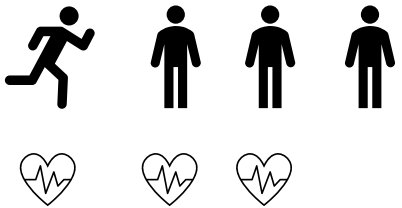
Summary

#Code

If we currently measured heart rate to be low, what's the probability that the user is at rest or exercising?

$$P(Y = r) = \frac{10}{15} = \frac{2}{3} = 0.67$$

$$P(Y = e) = \frac{5}{15} = \frac{1}{3} = 0.33$$

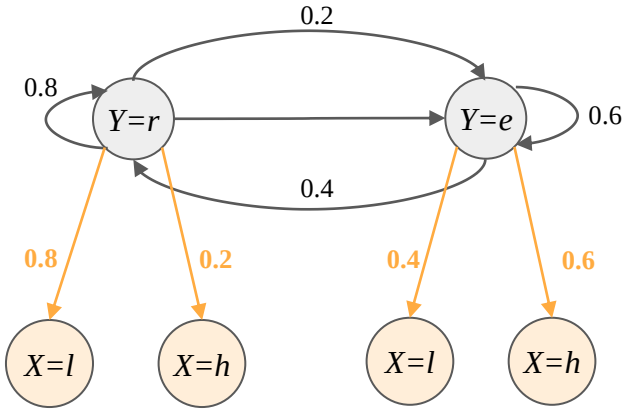


$$P(Y_t = r|Y_{t-1} = r) = \frac{8}{10} = 0.8$$

$$P(Y_t = e|Y_{t-1} = r) = \frac{2}{10} = 0.2$$

$$P(Y_t = r|Y_{t-1} = e) = \frac{2}{5} = 0.4$$

$$P(Y_t = e|Y_{t-1} = e) = \frac{3}{5} = 0.6$$



$$P(X_t = l|Y_t = r) = \frac{8}{10} = 0.8$$

$$P(X_t = h|Y_t = r) = \frac{2}{10} = 0.2$$

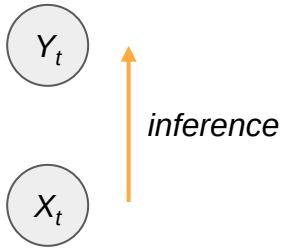
$$P(X_t = l|Y_t = e) = \frac{2}{5} = 0.4$$

$$P(X_t = h|Y_t = e) = \frac{3}{5} = 0.6$$

Sequence modelling: single evaluation

#Code

- given fixed model parameters and observed data, compute the probability of the hidden state



- If we currently measured heart rate to be low, what's the probability that the user is at rest or exercising?

$$P(Y = r) = \frac{10}{15} = \frac{2}{3} = 0.67$$

$$P(Y = e) = \frac{5}{15} = \frac{1}{3} = 0.33$$

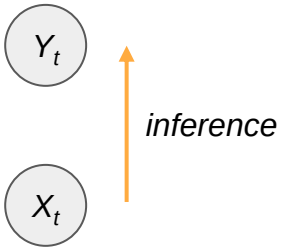
$$P(Y = r|X = l)$$

$$P(Y = e|X = l)$$

Sequence modelling: single evaluation

#Code

- given fixed model parameters and observed data, compute the probability of the hidden state



- If we currently measured heart rate to be low, what's the probability that the user is at rest or exercising?

$$P(Y = r) = \frac{10}{15} = \frac{2}{3} = 0.67$$

$$P(Y = e) = \frac{5}{15} = \frac{1}{3} = 0.33$$

Bayes' Theorem:

The symbol $\propto$ represents "proportional to" in mathematics. We read x $\propto$ y as "x is directly proportional to y."

$$P(Y = r|X = l) \propto P(X = l|Y = r)P(Y = r) = 0.8 \times 0.67 = 0.536$$

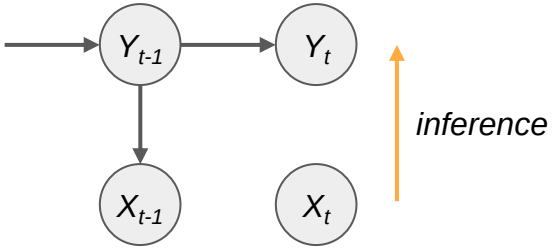
$$P(Y = e|X = l) \propto P(X = l|Y = e)P(Y = e) = 0.4 \times 0.33 = 0.132$$

Decision: $y^\star = \arg\max_{y \in \Omega_Y} P(X|Y)P(Y = y) = r$

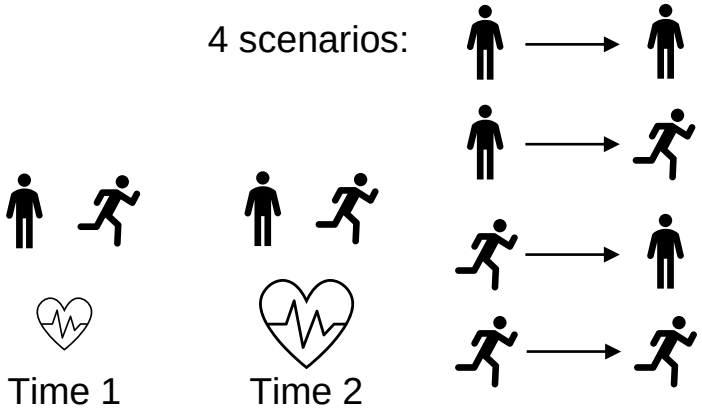
#Code

Sequence modelling: decoding

- given fixed model parameters and data, compute the most probable sequence of hidden states, $y = [y_0^\star, y_1^\star, \ldots, y_T^\star]$



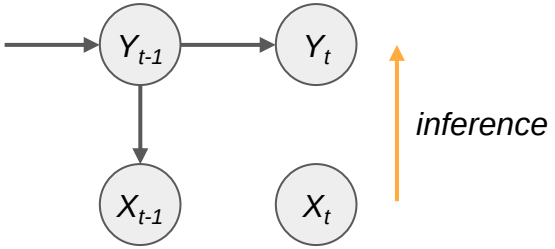
- If, for two consecutive measurements, we get heart rate to be **low, high**, what was the most likely scenario for activity?



Sequence modelling: decoding

#Code

- given fixed model parameters and data, compute the most probable sequence of hidden states, $y = [y_0^\star, y_1^\star, \ldots, y_T^\star]$



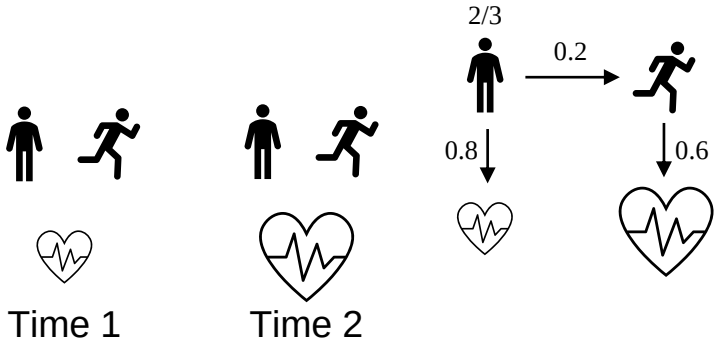
连续的

- If, for two consecutive measurements, we get heart rate to be **low, high**, what was the most likely scenario for activity?

Previous state: exercise
Current state: rest

$$P(y) = P(Y_0 = r|X_0 = l)P(Y_1 = e|Y_0 = r)P(X_1 = h|Y_1 = e)$$
$$= P(X_0 = l|Y_0 = r)P(Y_0 = r)P(Y_1 = e|Y_0 = r)P(X_1 = h|Y_1 = e)$$
$$= 0.8 \times 0.67 \times 0.2 \times 0.6 = 0.064$$

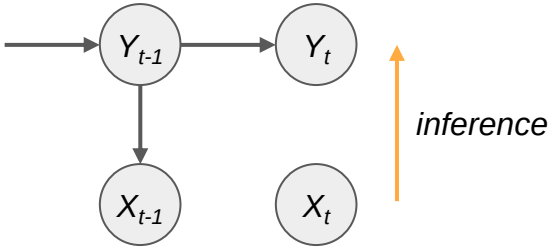
4 scenarios:



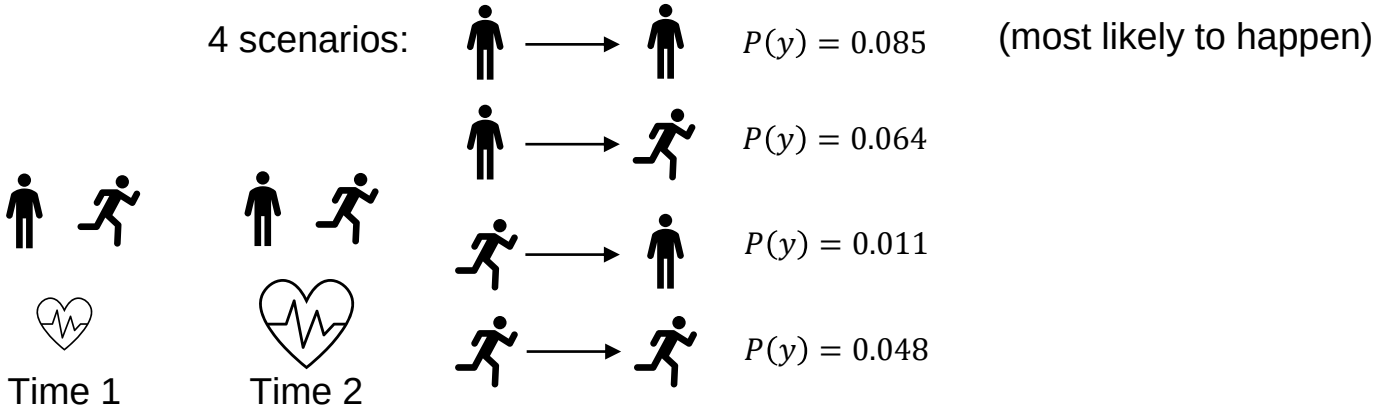
Sequence modelling: decoding

#Code

- given fixed model parameters and data, compute the most probable sequence of hidden states, $y = [y_0^\star, y_1^\star, \ldots, y_T^\star]$



- If, for two consecutive measurements, we get heart rate to be **low, high**, what was the most likely scenario for activity?

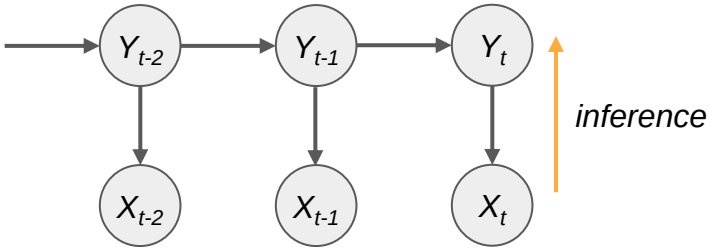


AIML

Sequence modelling: decoding

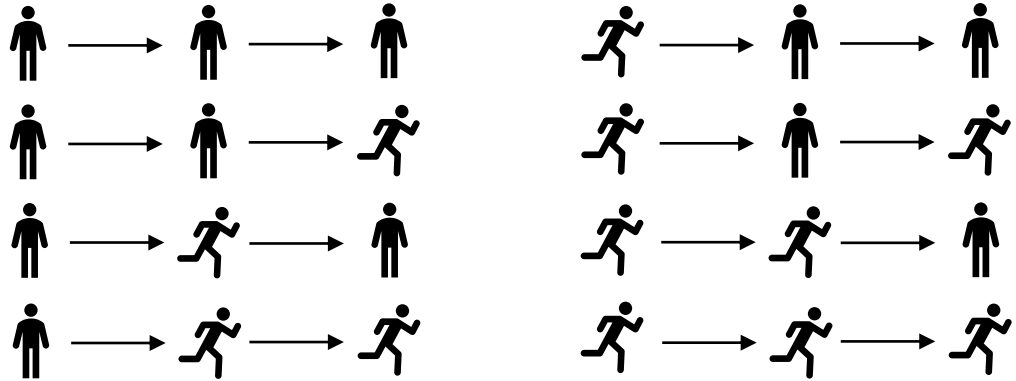
#Code

- given fixed model parameters and data, compute the most probable sequence of hidden states, $y = [y_0^\star, y_1^\star, \ldots, y_T^\star]$



- If, for three consecutive measurements, we get heart rate to be **low, high**, **low**, what was the most likely scenario for activity?

8 scenarios:



HMM sequence modelling problems

#Code

- In applications of HMMs, typically need to solve the following problems
  - **Model fitting**: given observed data for $X_0, X_1, \ldots, X_T$, estimate the distribution functions $P(X_t|Y_t)$, $P(Y_t|Y_{t-1})$;
  - **Evaluation**: given fixed model parameters and observed data, compute the probability of the data, $P(X)$;
  - **Decoding**: given fixed model parameters and data compute the most probable sequence of hidden states $y = [y_0^*, y_1^*, y_2^*, \ldots, y_T^*]$.

#Code

HMM sequence modelling problems

- In applications of HMMs, typically need to solve the following problems
  - **Model fitting**: given observed data for $X_0, X_1, \ldots, X_T$, estimate the distribution functions $P(X_t|Y_t)$, $P(Y_t|Y_{t-1})$;
  - **Evaluation**: given fixed model parameters and observed data, compute the probability of the data, $P(X)$;
  - **Decoding**: given fixed model parameters and data compute the most probable sequence of hidden states $y = [y_0^*, y_1^*, y_2^*, \ldots, y_T^*]$.
- Solving these problems requires evaluating **all possible sequences of hidden states**; if there are K hidden states, this requires $O(K^T)$ (exponential complexity)

#Code

HMM sequence modelling problems

- In applications of HMMs, typically need to solve the following problems
  - **Model fitting**: given observed data for $X_0, X_1, \ldots, X_T$, estimate the distribution functions $P(X_t|Y_t)$, $P(Y_t|Y_{t-1})$;
  - **Evaluation**: given fixed model parameters and observed data, compute the probability of the data, $P(X)$;
  - **Decoding**: given fixed model parameters and data compute the most probable sequence of hidden states $y = [y_0^*, y_1^*, y_2^*, \ldots, y_T^*]$.

X_t=value, Y_t situation

Y_t=current state, Y_t-1=previous situation

- Solving these problems requires evaluating **all possible sequences of hidden states**; if there are K hidden states, this requires $O(K^T)$ (exponential complexity)
- Use of **dynamic programming** makes this tractable in order $O(TK^2)$.

Bellman recursion for optimal sequence probability

- Reading off PGM, at time step $t-1$, optimal sequence probability:

$$P^{\star}(X_0, \ldots, X_{t-1}, Y_{t-1}) = \max_{y' \in Y_{t-2}} P(X_0, \ldots, X_{t-1}, Y_0 = y'_0, \ldots, Y_{t-2}, Y_{t-1})$$

  where $\mathcal{Y}_{t-2}$ is set of all possible state sequences, up to time $t-2$.

- Optimal sequence probability, as a function of y up to time t,

$$p_t^{\star}(y) = P^{\star}(X_0, \ldots, X_t, Y_t = y)$$

  is obtained using **Bellman recursion**,

$$p_t^{\star}(y) = \max_{y' \in \Omega_Y} [p_{t-1}^{\star}(y')\, P(Y_t = y | Y_{t-1} = y') P(X_t = x_t | Y_t = y)]$$

HMM Viterbi decoding: algorithm

#Code

- **Step 1**. *Initialization*: Compute the initial optimal probability function,

$$p_0^\star(y) = P(X_0 = x_0 | Y_0 = y)\, P(Y_0 = y)$$

- **Step 2**. *Forward recursion*: Sequence of optimal probability functions,

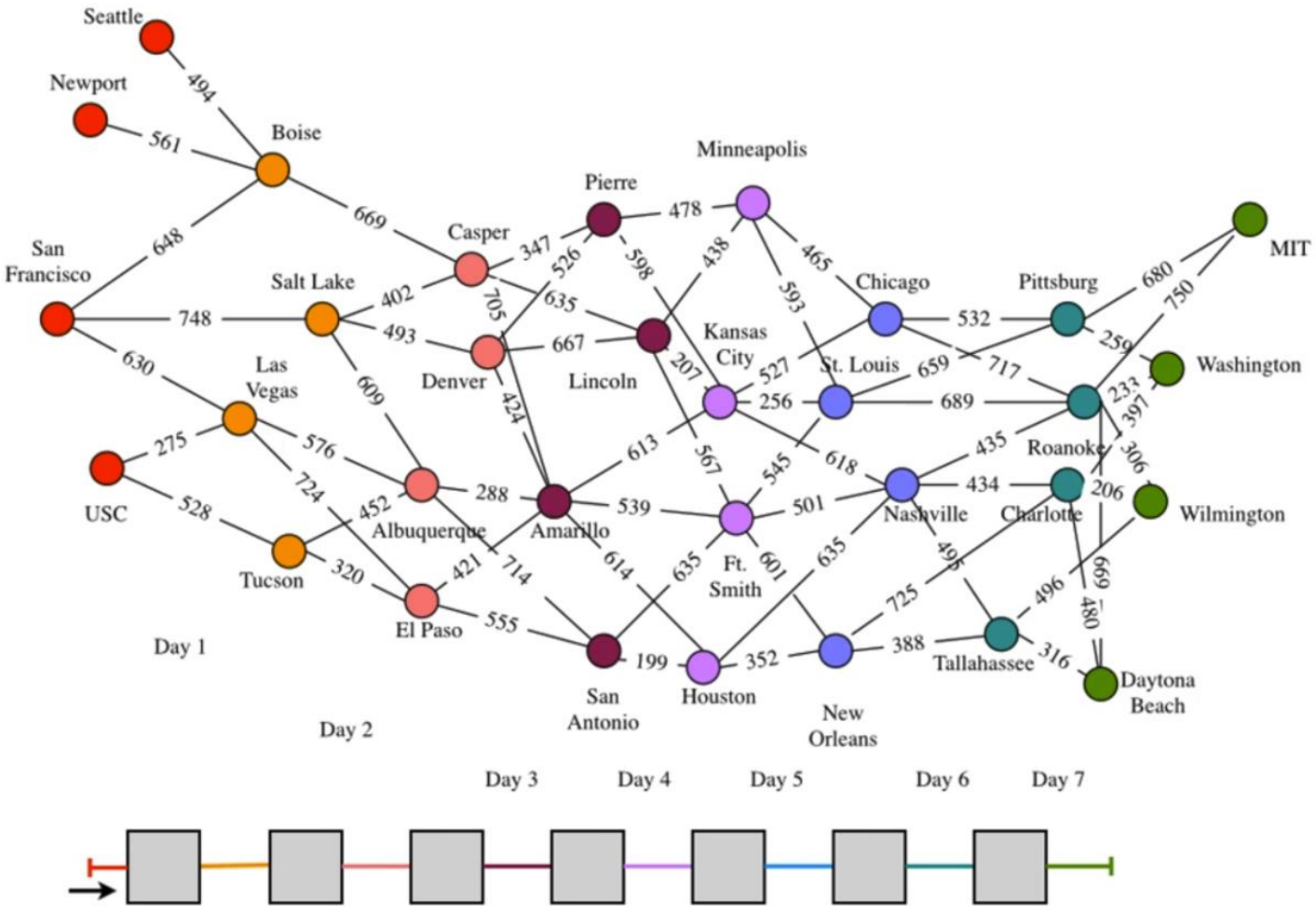
$$p_t^\star(y) = \max_{y' \in \Omega_Y} p_{t-1}^\star(y')\, P(Y_t = y | Y_{t-1} = y')\, P(X_t = x_t | Y_t = y)$$

for $t = 1,2,\ldots, T$, keeping track of the corresponding decision,

$$Y_t^\star(y) = \arg\max_{y' \in \Omega_Y} p_{t-1}^\star(y')\, P(Y_t = y | Y_{t-1} = y')$$

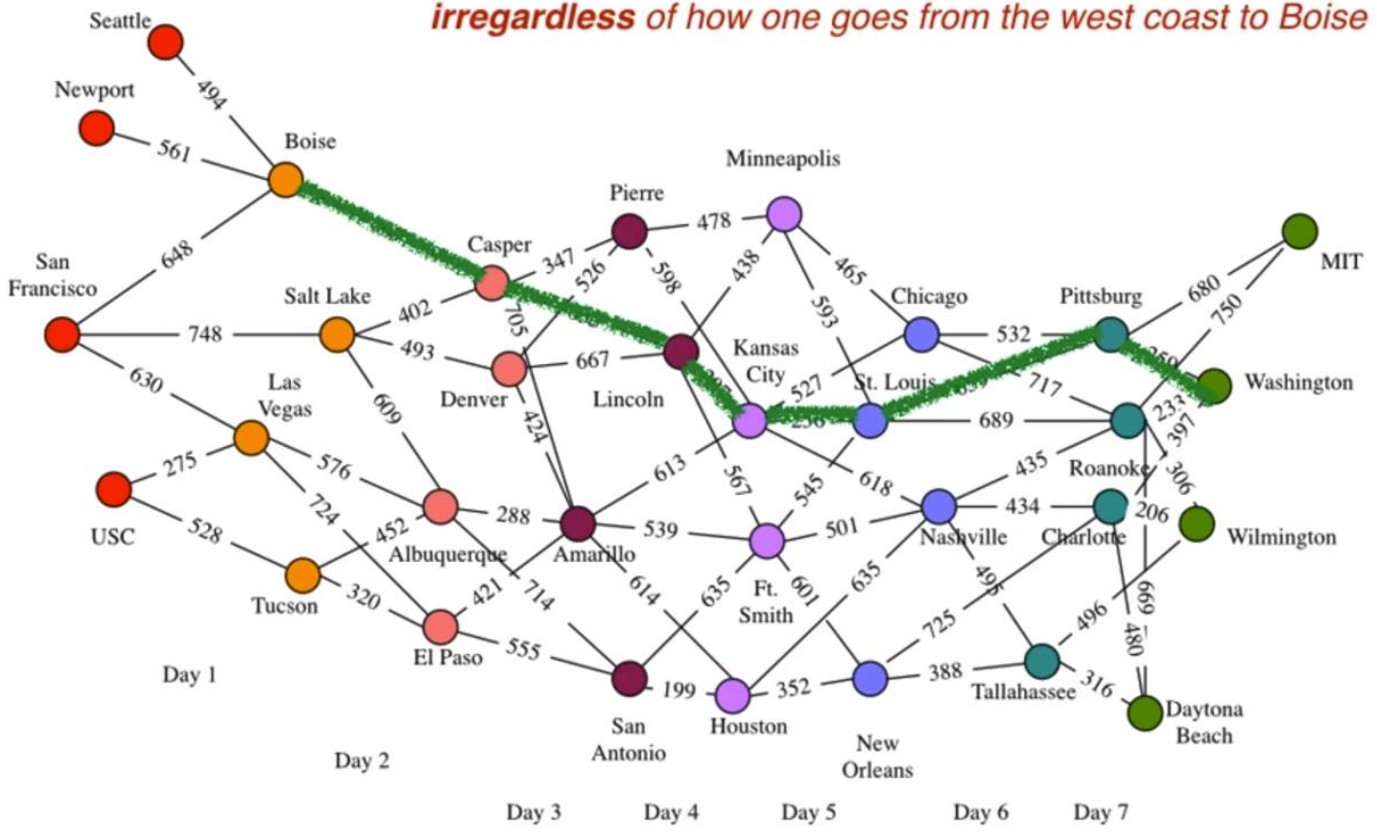
- **Step 3**. *Backtrack*: Find optimal sequence in reverse, for $t = T-1, T-2, \ldots, 1$,

$$y_T^\star = \arg\max_{y \in \Omega_Y} p_T^\star(y), y_{t-1}^\star = Y_t^\star(y_t^\star)$$

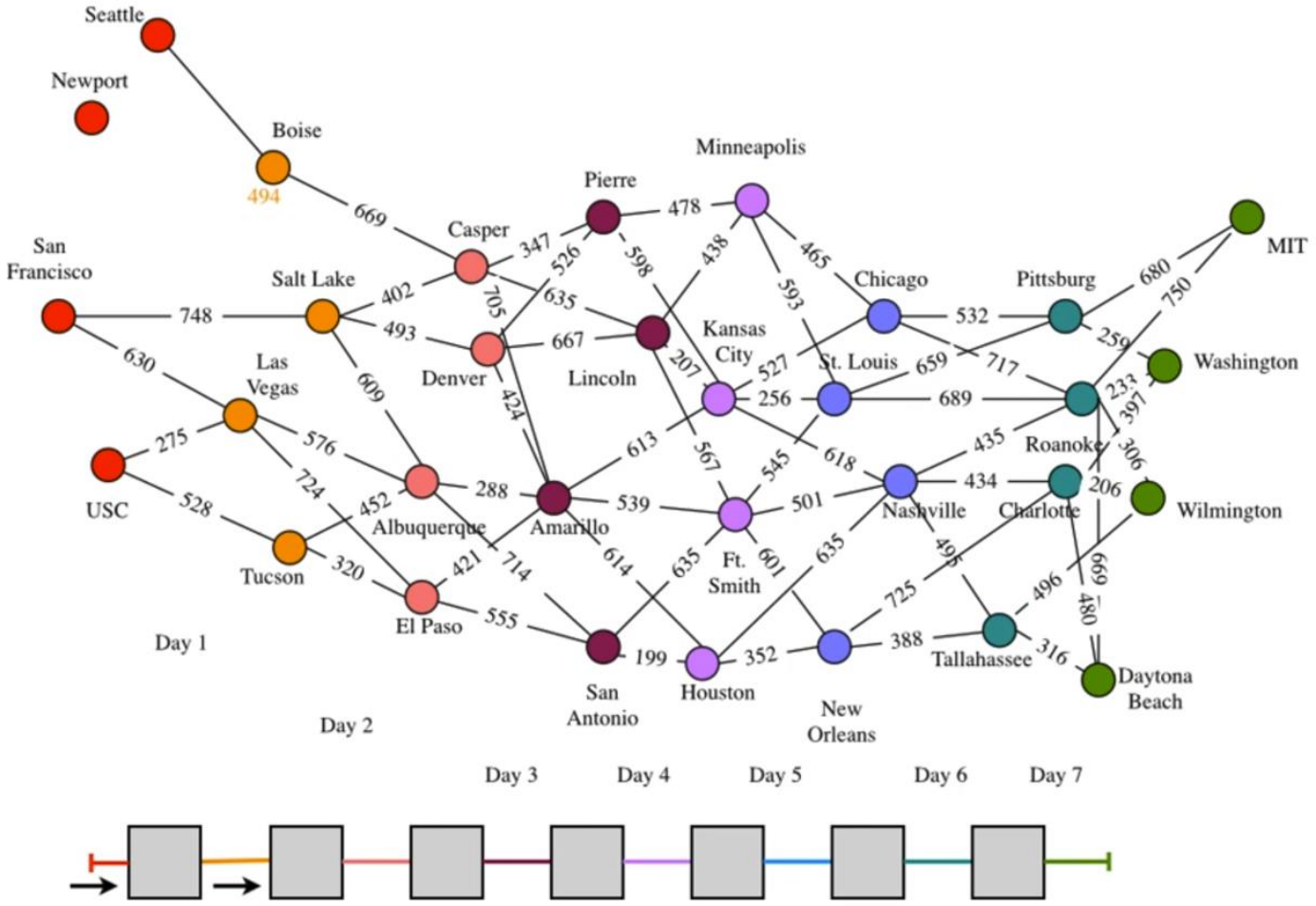

Seattle
Newport
Boise
Minneapolis
Pierre
San Francisco
Casper
Salt Lake
Chicago
Pittsburg
MIT
Kansas City
Denver
Lincoln
St. Louis
Washington
Las Vegas
Roanoke
USC
Albuquerque
Amarillo
Nashville
Charlotte
Wilmington
Ft. Smith
Tucson
El Paso
Tallahassee
Daytona Beach
San Antonio
Houston
New Orleans
Day 1
Day 2
Day 3
Day 4
Day 5
Day 6
Day 7

the best path between Boise and the east coast is 2685 miles

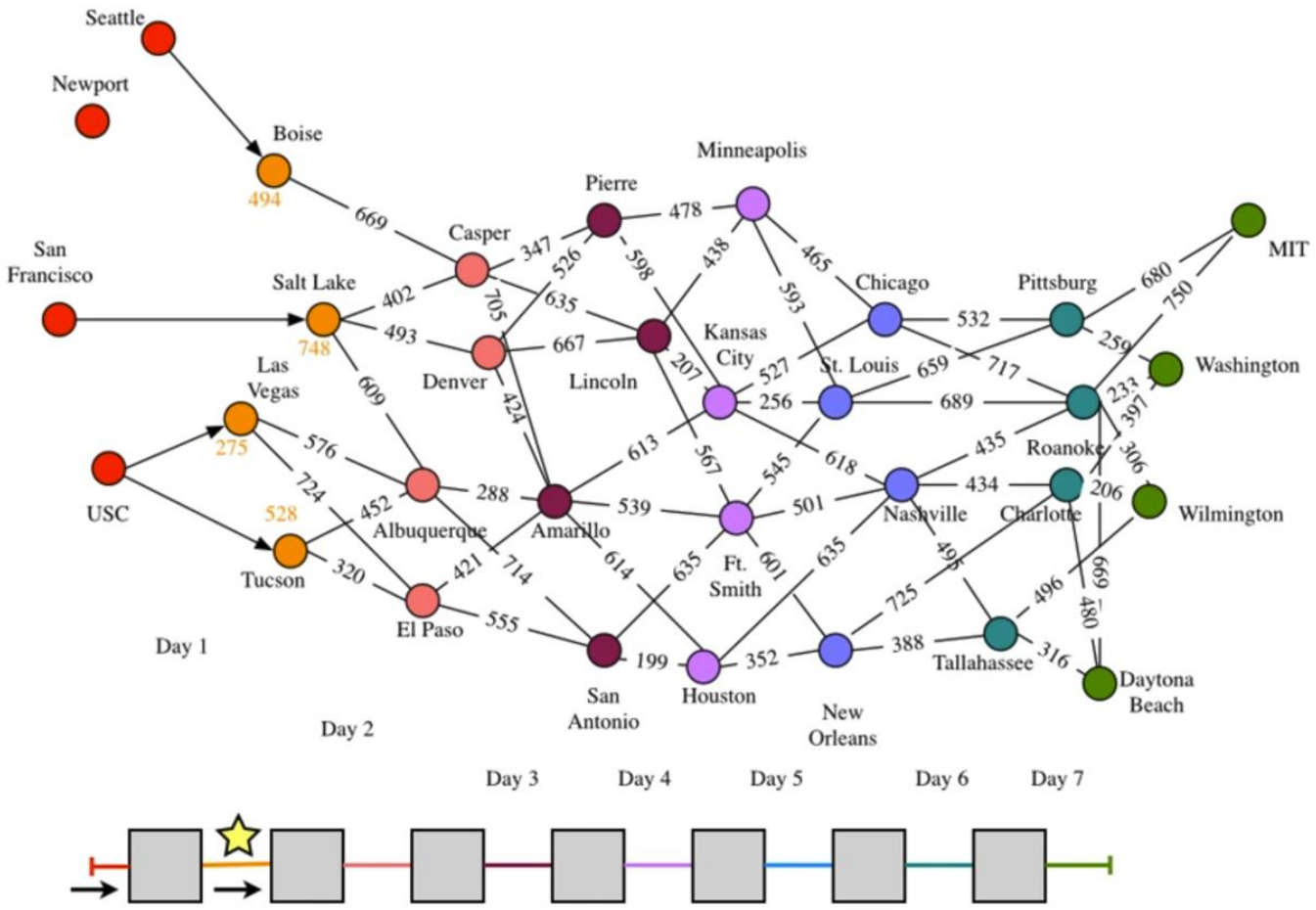
irregardless *of how one goes from the west coast to Boise*



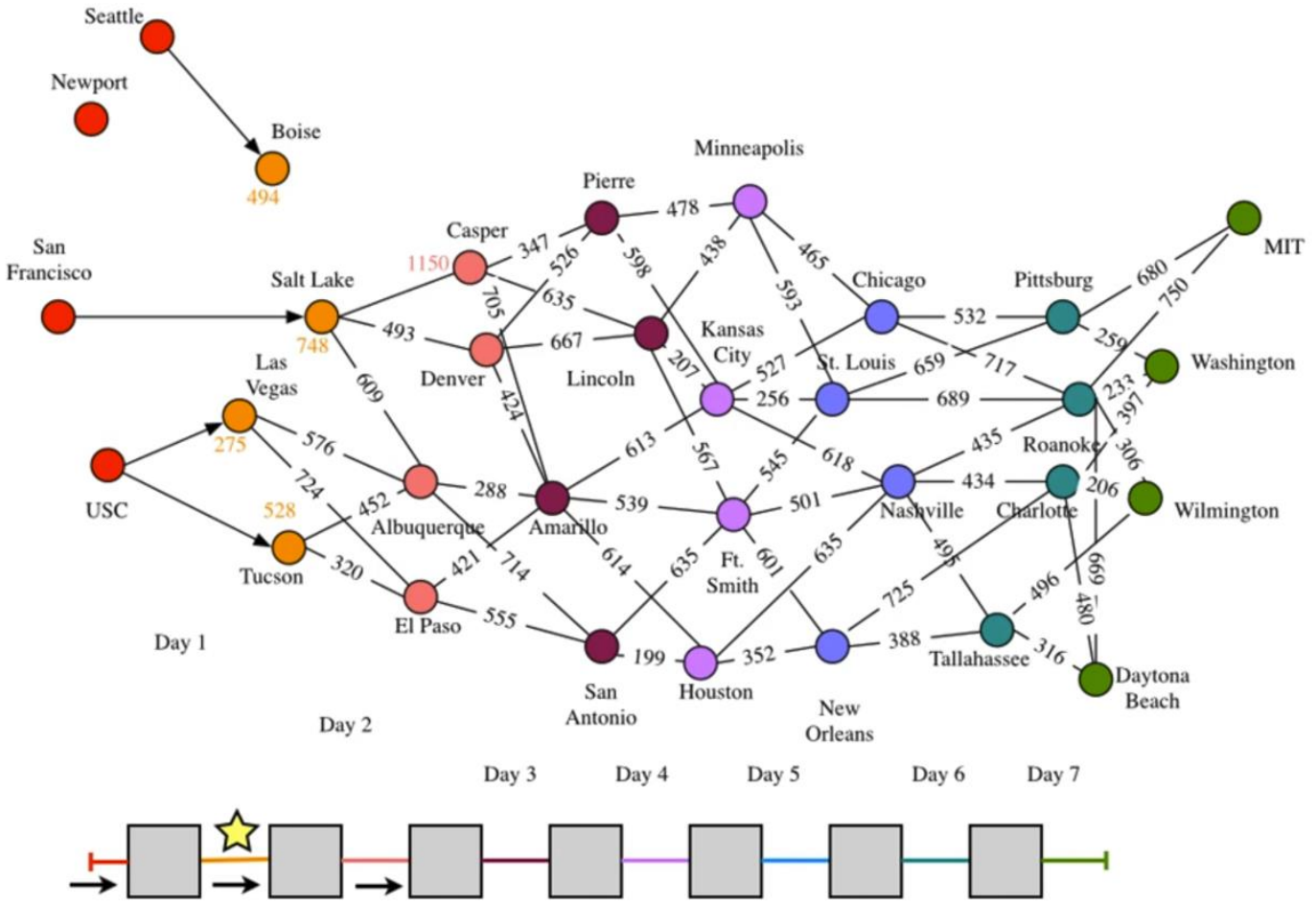
If best path from west coast to east coast, passes through Boise, it must coincide with best path from west coast to Boise



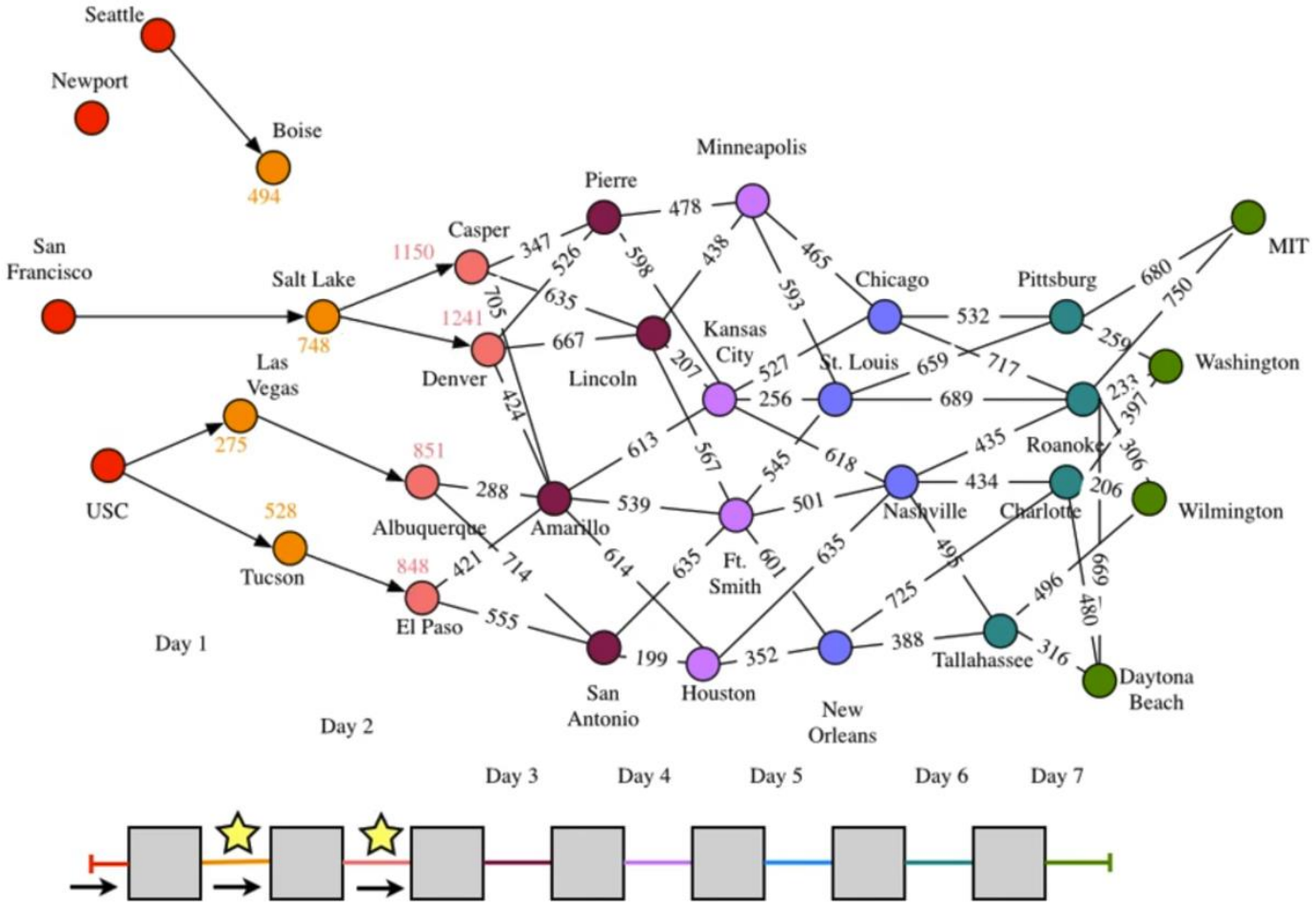
Repeat Boise-argument for Salt Lake, Las Vegas, Tucson

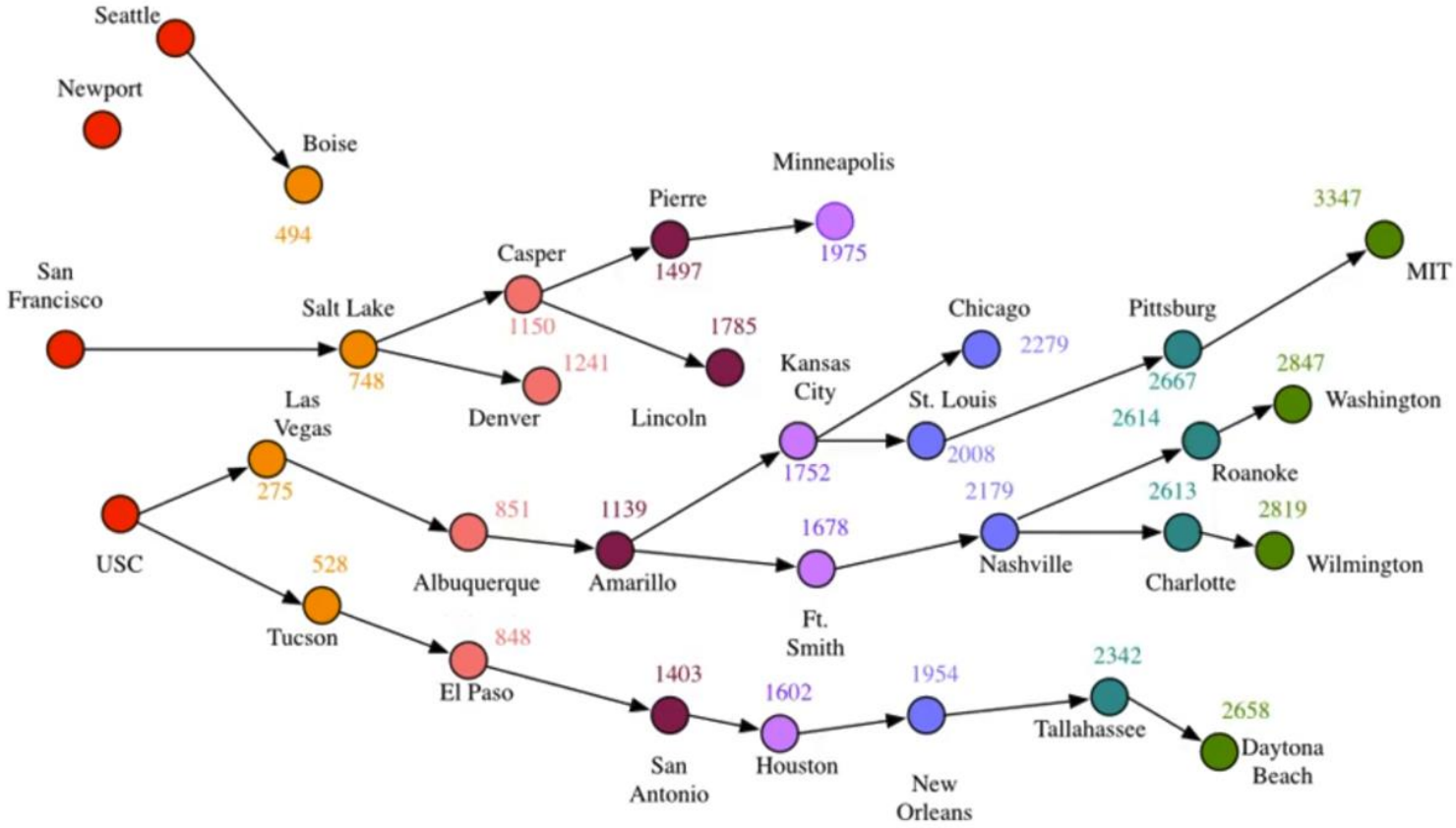


Repeat for Casper



Repeat for day-2 destination cities




Seattle
Newport
Boise
494
Minneapolis
Pierre
1975
1497
3347
MIT
San
Francisco
Casper
Salt Lake
1150
1785
Chicago
2279
Pittsburg
2667
2847
Washington
748
1241
Kansas
City
St. Louis
2008
Las
Vegas
Denver
Lincoln
2614
Roanoke
1752
275
851
1139
2179
2613
2819
1678
USC
Albuquerque
Amarillo
Nashville
Charlotte
Wilmington
528
Ft.
Smith
Tucson
848
1403
1602
1954
2342
2658
El Paso
San
Antonio
Houston
New
Orleans
Tallahassee
Daytona
Beach

Viterbi algorithm for our example

#Code

- If, for three consecutive measurements, we get heart rate to be **low, high**, **low**, what was the most likely scenario for activity?

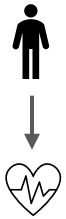
$$p_0^\star(y) = P(X_0 = l | Y_0 = y) P(Y_0 = y)$$

$$p_0^\star(r) = 0.8 \times 0.67 = 0.536$$

$$p_0^\star(e) = 0.4 \times 0.33 = 0.132$$

$p_0^\star(e) = 0.132$

$p_0^\star(r) = 0.536$



Time 0 Time 1 Time 2

Viterbi algorithm for our example

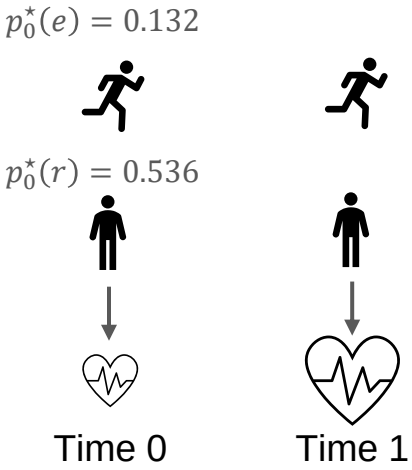
- If, for three consecutive measurements, we get heart rate to be **low, high**, **low**, what was the most likely scenario for activity?

$$p_1^\star(y) = \max_{y' \in \Omega_Y} p_0^\star(y') P(Y_1 = y | Y_0 = y') P(X_1 = h | Y_1 = y)$$

$$p_1^\star(r) = \max\{0.536 \times 0.8 \times 0.2\,,\, 0.132 \times 0.4 \times 0.2\} = \max\{0.086, 0.011\} = 0.086$$

$$p_1^\star(e) = \max\{0.536 \times 0.2 \times 0.6\,,\, 0.132 \times 0.6 \times 0.6\} = \max\{0.064, 0.048\} = 0.064$$

$y' = r$ $y' = e$



Viterbi algorithm for our example

#Code

- If, for three consecutive measurements, we get heart rate to be **low, high**, **low**, what was the most likely scenario for activity?

$$p_2^\star(y) = \max_{y' \in \Omega_Y} p_1^\star(y') P(Y_2 = y | Y_1 = y') P(X_2 = l | Y_2 = y)$$

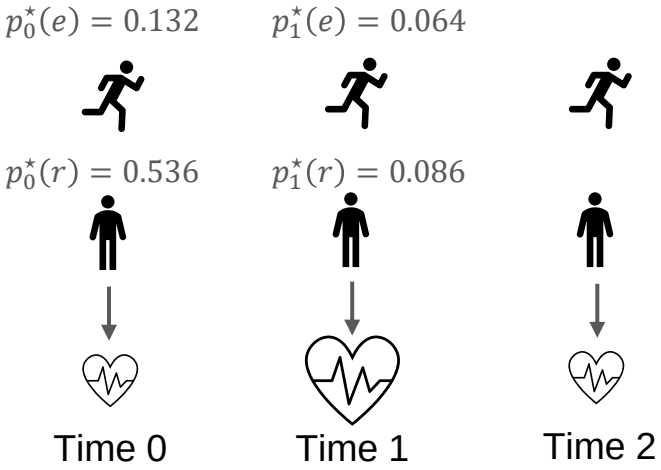
$$p_2^\star(r) = \max\{0.086 \times 0.8 \times 0.8\,,\, 0.064 \times 0.4 \times 0.8\}$$
($y' = r$, $y' = e$)

$$= \max\{0.055, 0.020\} = 0.055$$

$$p_2^\star(e) = \max\{0.086 \times 0.2 \times 0.4\,,\, 0.064 \times 0.6 \times 0.4\}$$

$$= \max\{0.0069, 0.015\} = 0.015$$

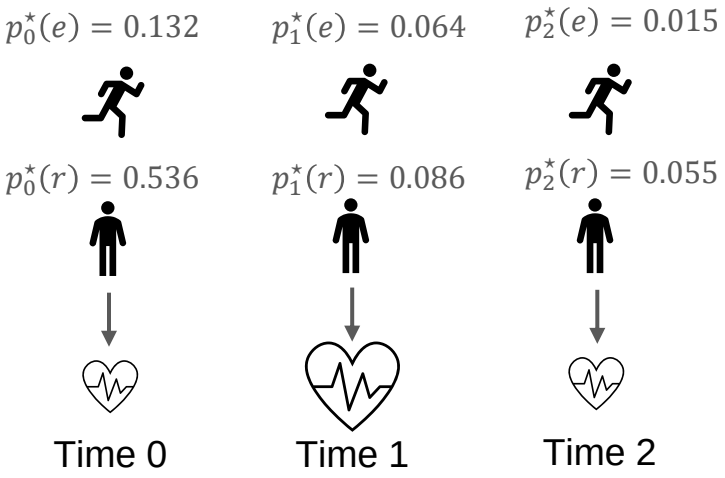
$$Y_1^\star(y) = [r, r]$$



Viterbi algorithm for our example

#Code

- If, for three consecutive measurements, we get heart rate to be **low, high**, **low**, what was the most likely scenario for activity?



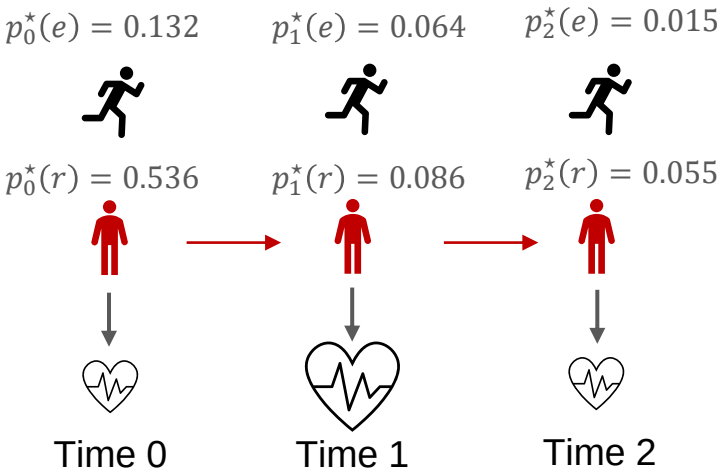
$$Y_2^\star(y) = [r, r, r]$$

AIML

#Code

Viterbi algorithm for our example

- If, for three consecutive measurements, we get heart rate to be **low, high**, **low**, what was the most likely scenario for activity?



$$Y_2^\star(y) = [r, r, r]$$

AIML

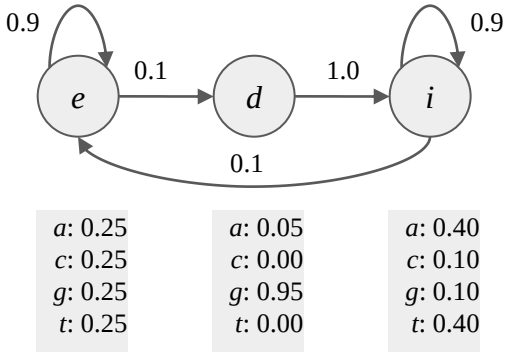
#Code

Genome sequence region segmentation

- **Problem**: optimal segmentation of DNA sequences into exon (e), intron (i) donor site (d) sub-sequences

外显子 (e)、内含子 (i) 供体位点 (d)

- **Distributions**: Y – regions, $\Omega_Y=\{e,d,i\}$, X – DNA sequences, $\Omega_X=\{a,c,g,t\}$
- **State transition distribution and observation distributions**: from molecular biology:



Input data: $x=[g,g,g,g,t,a]$
Globally optimal sequence:
$y^*=[e,e,e,d,i,i]$

Stage	x_t	$p^*_t(y=e)$	$p^*_t(y=d)$	$p^*_t(y=i)$	$Y^*_t(y=e)$	$Y^*_t(y=d)$	$Y^*_t(y=i)$	y^*_t
$t=0$	g	0.2500	0.0000	0.0000				e
$t=1$	g	0.0563	0.0238	0.0000	e	e	e	e
$t=2$	g	0.0127	0.0053	0.0024	e	e	d	e
$t=3$	g	0.0028	0.0012	0.0005	e	e	d	d
$t=4$	t	0.0006	0.0000	0.0005	e	e	d	i
$t=5$	a	0.0001	0.0000	0.0002	e	e	i	i

#Code

To recap

- We discussed sequence modelling (data in which ordering matters)
  - **Hidden Markov Model**: sequences through a series of (discrete) hidden states that are not directly observable, but follows a certain probability distribution

- **Next**: Other sequential models (Kalman filter, Recurrent neural nets)

Further Reading

- **PRML**, Section 13.2
- **R&N**, Section 14.3 (matrix algebra approach)
- **MLSP**, Section 9.4