Announcements

- Pset-2 due on March 6th.

Last time

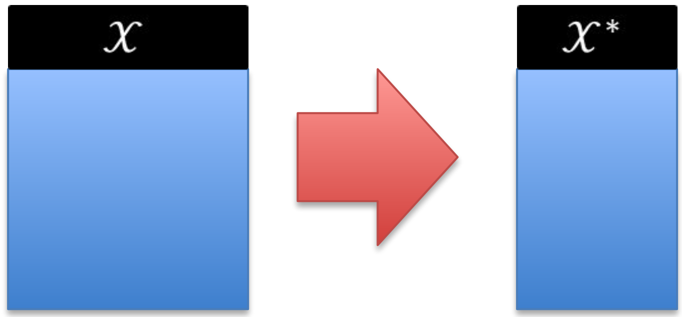
- Model Selection using AIC/BIC
- Robust Learning
  - Different loss functions
  - Boosting
  - Weak learners

Today

- Regression Trees
- Markov Chain
- Hidden Markov Model
- Decoding HMMs

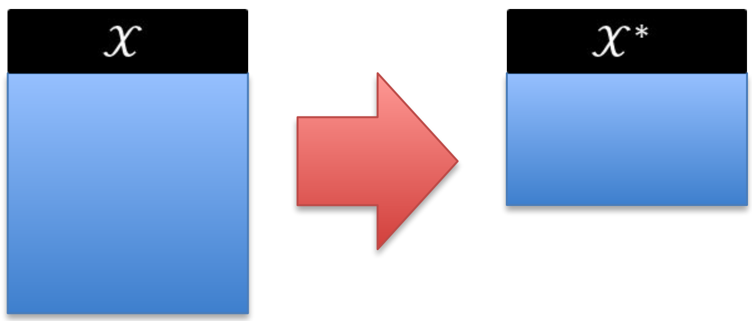
Regression on large datasets

PCA



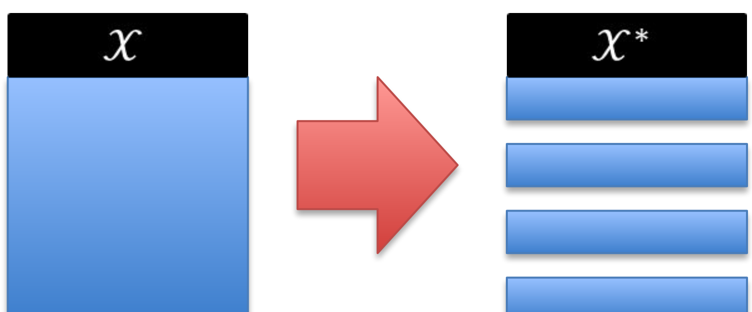
May lose important ***features***

Sub-sample



May lose important ***samples***

Stochastic Gradient Descent



Sees only ***few samples*** in each batch

Greedy Stagewise Linear Regression

- **Main idea:** segment the features and train a model on those features, i.e., we minimize

$$\mathcal{L}^{(j)}(\beta) = \left\| e^{(j-1)} - \mathcal{X}^{(j)}\beta \right\|^2$$

Greedy Stagewise Linear Regression

- **Main idea:** segment the features and train a model on those features, i.e., we minimize

$$\mathcal{L}^{(j)}(\beta) = \left\| e^{(j-1)} - \mathcal{X}^{(j)}\beta \right\|^2$$

Start $e^{(0)} = y$ and $j = 1$, then

1. Select a subset of features for $\mathcal{X}^{(j)}$
2. Learn $\hat{\beta}^{(j)}$ by minimizing $\mathcal{L}^{(j)}(\beta)$

Greedy Stagewise Linear Regression

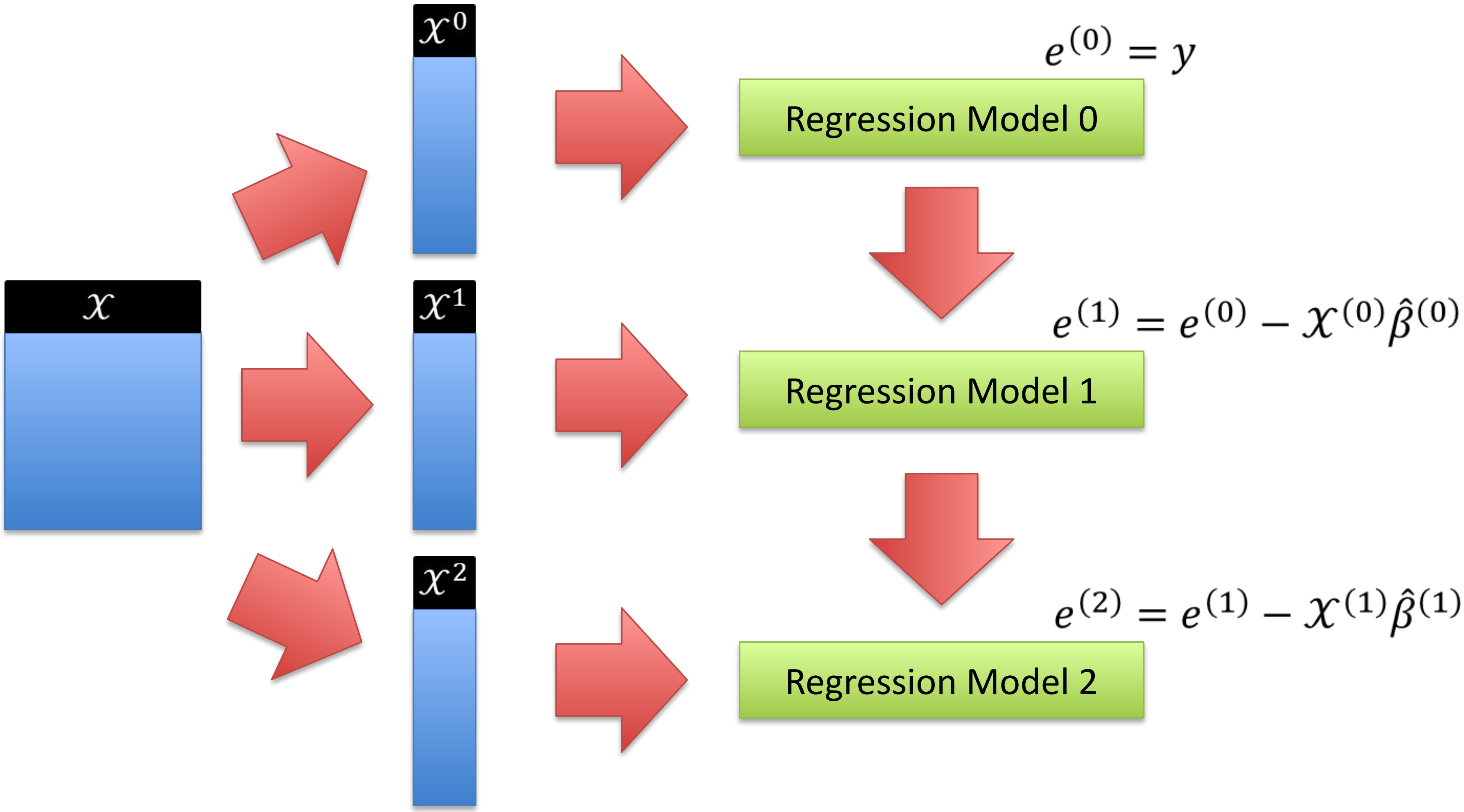
- **Main idea:** segment the features and train a model on those features, i.e., we minimize

$$\mathcal{L}^{(j)}(\beta) = \left\|e^{(j-1)} - \mathcal{X}^{(j)}\beta\right\|^2$$

Start $e^{(0)} = y$ and $j = 1$, then

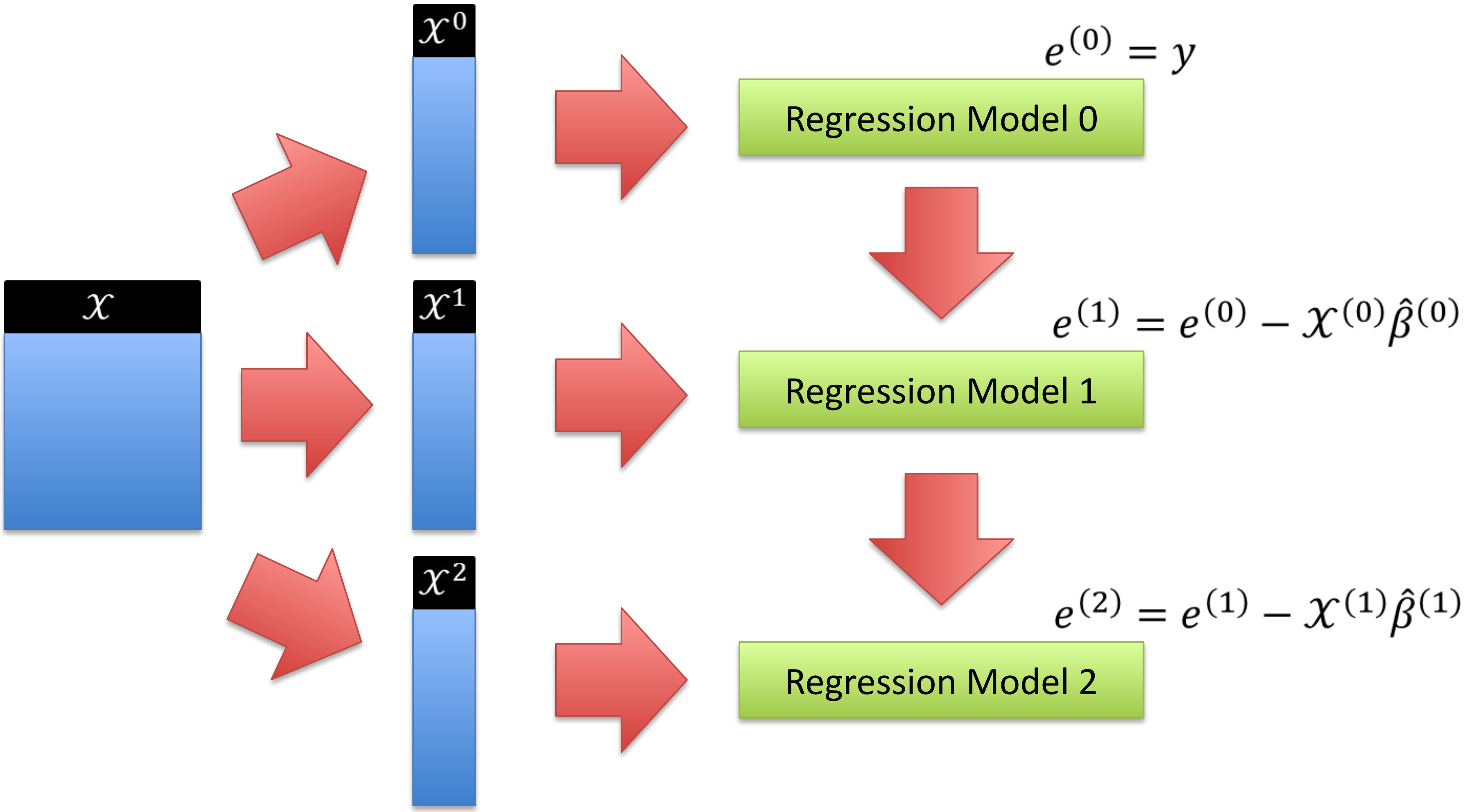
1. Select a subset of features for $\mathcal{X}^{(j)}$
2. Learn $\hat{\beta}^{(j)}$ by minimizing $\mathcal{L}^{(j)}(\beta)$
3. $e^{(j)} = e^{(j-1)} - \mathcal{X}^{(j)}\hat{\beta}^{(j)}$
4. Repeat for $j + 1$

Greedy Stagewise Linear Regression



Visual depiction of an extreme case - where a model is trained on each feature

Greedy Stagewise Linear Regression



At each stage, pick the feature that is most informative of **y**

When to stop adding weak learners?

- Recall are minimizing:

$$\mathcal{L}^{(j)}(\beta) = \left\| e^{(j-1)} - \mathcal{X}^{(j)}\beta \right\|^2$$

Any weak leaner cannot result in an increase in the residual, i.e., $\left\| e^{(j)} \right\|^2 \leq \left\| e^{(j-1)} \right\|^2$

Stop adding weak learners when the residual is high

Greedy Stagewise Linear Regression

- **Pros:**
  - Simple to implement
  - Computationally fast
- **Cons:**
  - Potential for overfitting.
    - Greedy selection can lead to selecting features that may not perform well on new data
  - Greedy selection might not lead to optimal solution

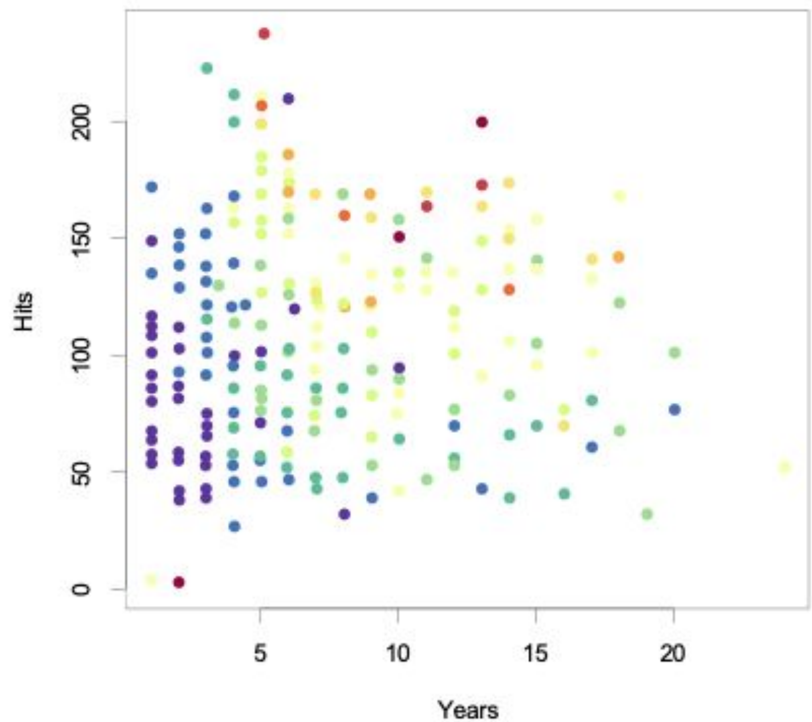
Today

- Model Selection using AIC/BIC
- **Robust Learning**
  - Different loss functions
  - Boosting
  - Weak learners
  - **Regression Trees**

An example: Regression Trees

Baseball salary data: how would you segment it?

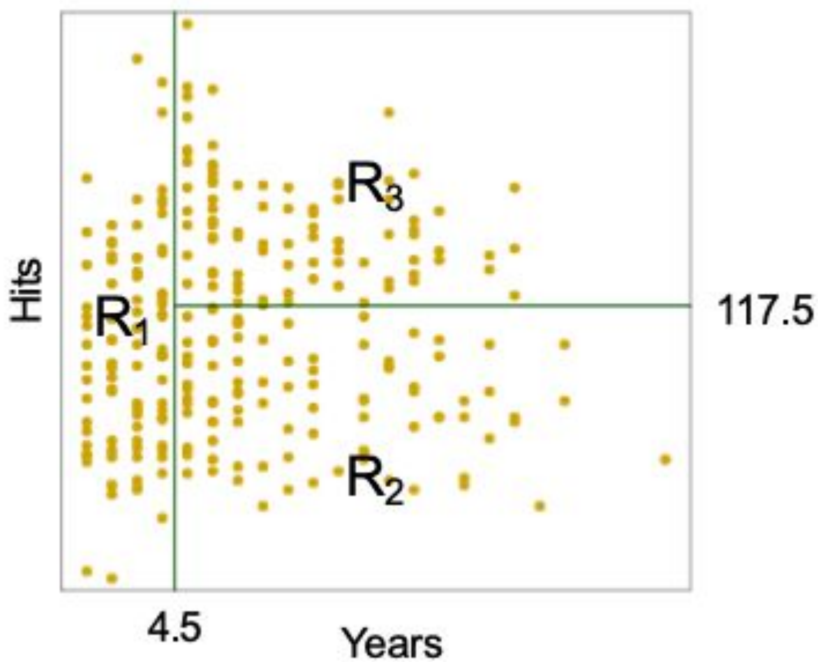
Salary is color-coded from low (blue, green) to high (yellow,red)



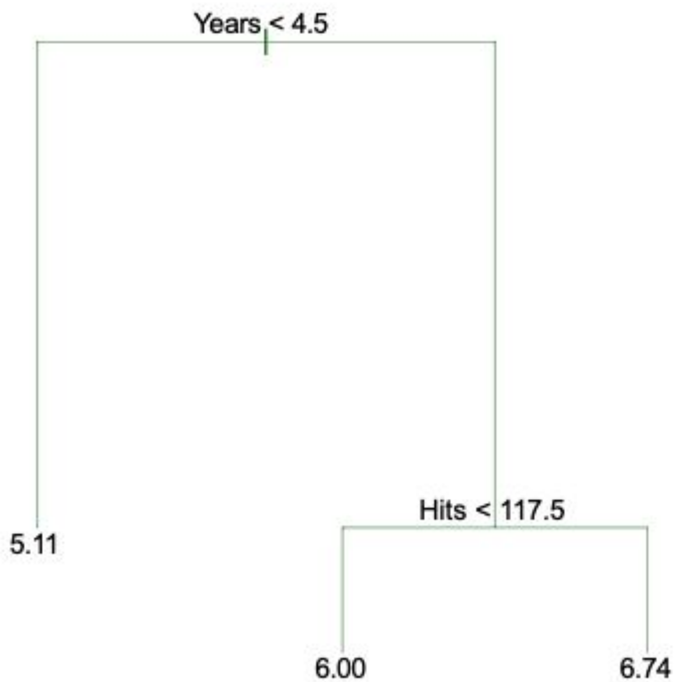
An example: Regression Trees

Results

Overall, the tree segments the players into three regions of predictor space: $R_1 = \{X \mid \text{Years} < 4.5\}$, $R_2 = \{X \mid \text{Years} >= 4.5, \text{Hits} < 117.5\}$, and $R_3 = \{X \mid \text{Years} >= 4.5, \text{Hits} >= 117.5\}$.



Example Decision Tree



Slide Credit: Saravanan Thirumuruganathan

17

Greedy Stagewise Regression w/Trees

- Given regression tree $f(x;\theta)$, our stagewise regression will be the sum of trees, i.e.,

$$F(x;\theta) = \sum_j f\left(x;\theta^{(j)}\right)$$

We will learn this by minimizing:

$$\mathcal{L}^{(j)}(\theta) = \left\| e^{(j-1)} - f(x;\theta) \right\|^2$$

Follow same procedure as for Greedy Stagewise Linear Regression

Weak learners for classification vs. regression

- A primary difference between classification and regression is the training loss $\mathcal{L}$, i.e., given a predictor F:

For least squares regression we have,

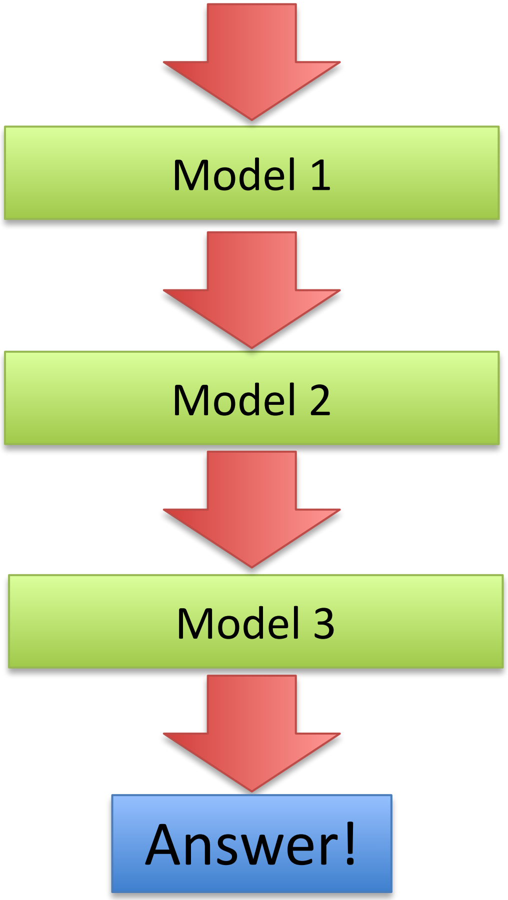
$$\mathcal{L}_{ls}(F) = \frac{1}{N}\sum_{i}\big(y_i - F(x_i)\big)^2$$

For a linear SVM we minimize the hinge loss,

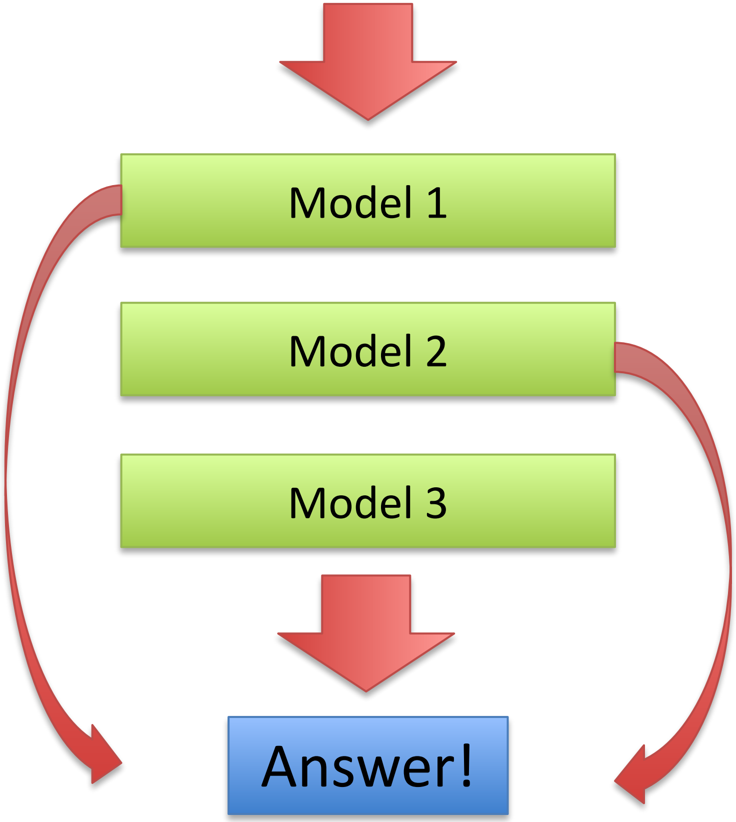
$$\mathcal{L}_h(F) = \frac{1}{N}\sum_{i}\max\big(0, 1 - y_i F(x_i)\big)$$

Boosting vs. Bagging Training

Boosting (Sequential)



Bagging (Parallel)

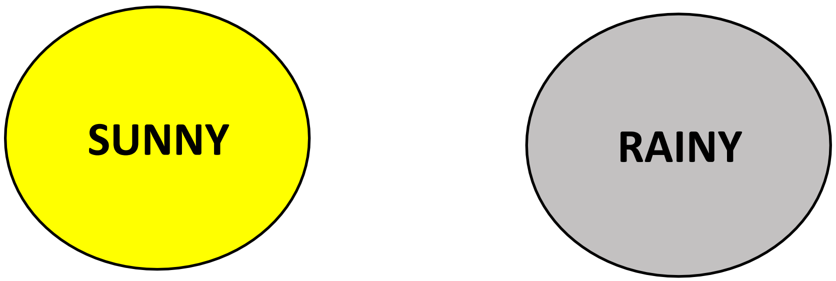


Today

- **Markov Chain**
- Hidden Markov Model
- Decoding HMMs

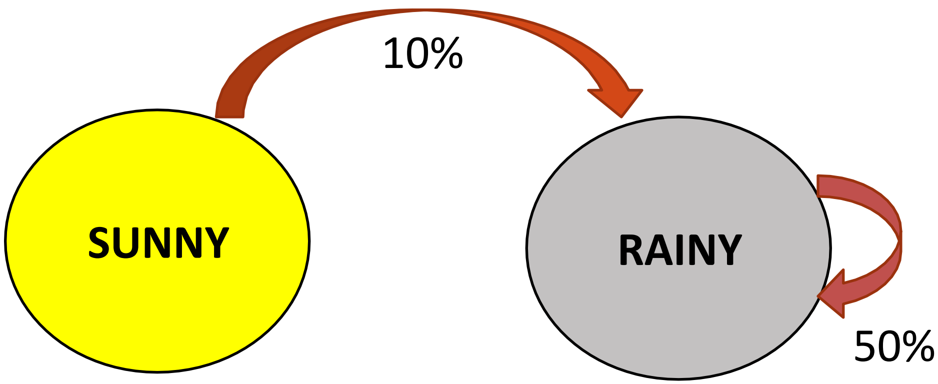
Representing transitions

Example setting: Based on many observations, the chance of a rainy day occurring after a rainy day is 50% and that the chance of a rainy day after a sunny day is 10%.



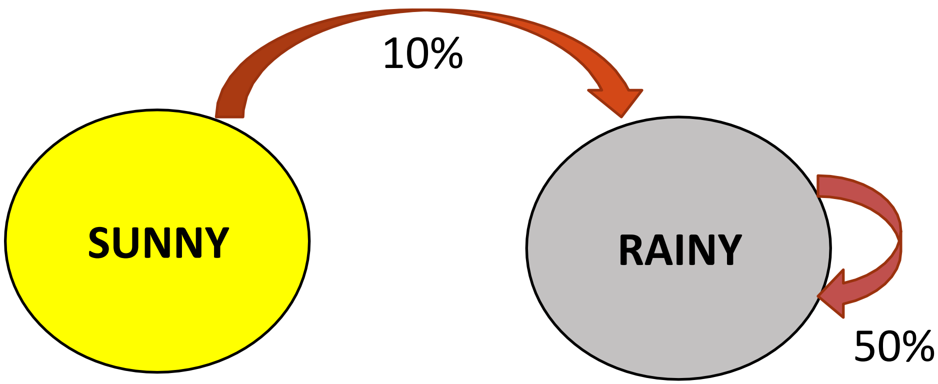
Representing transitions

Example setting: Based on many observations, the chance of a rainy day occurring after a rainy day is 50% and that the chance of a rainy day after a sunny day is 10%.



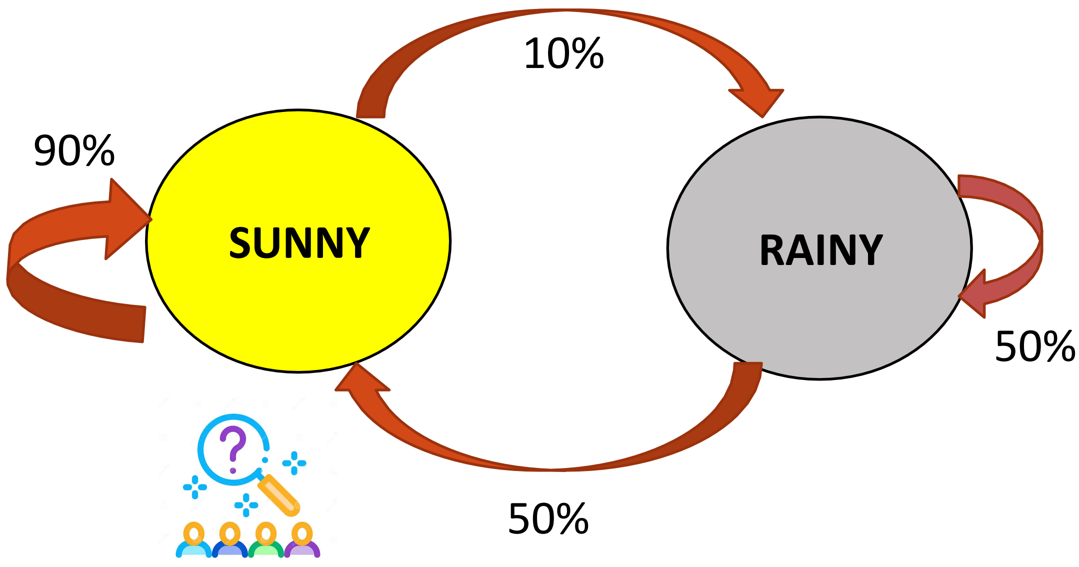
Representing transitions

Example setting: Based on many observations, the chance of a rainy day occurring after a rainy day is 50% and that the chance of a rainy day after a sunny day is 10%.



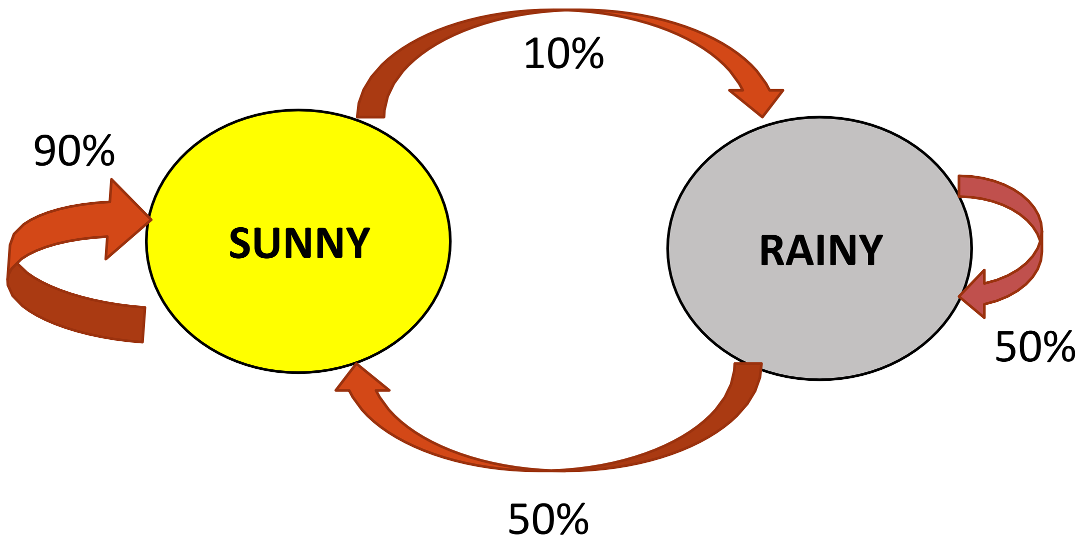
Representing transitions

Example setting: Based on many observations, the chance of a rainy day occurring after a rainy day is 50% and that the chance of a rainy day after a sunny day is 10%.



Representing transitions

Example setting: Based on many observations, the chance of a rainy day occurring after a rainy day is 50% and that the chance of a rainy day after a sunny day is 10%.

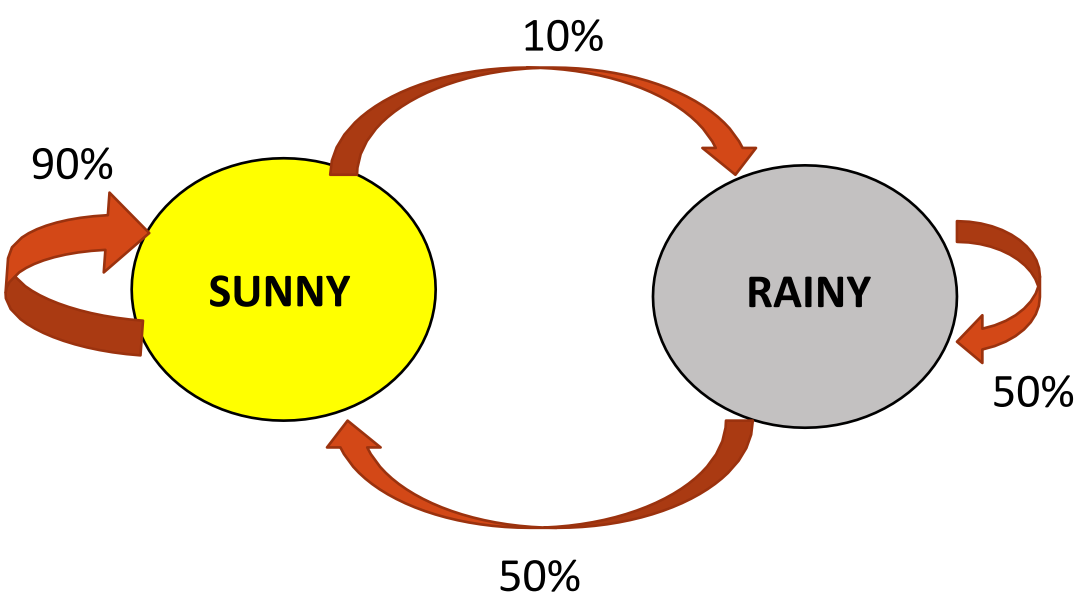


- **Graph:** Vertices, Edges.
- Represented using **adjacency matrix.**
- **Edge weights:** probabilities of weather conditions.

	Sunny	Rainy
Sunny		
Rainy		

Transition (or Markov) matrices

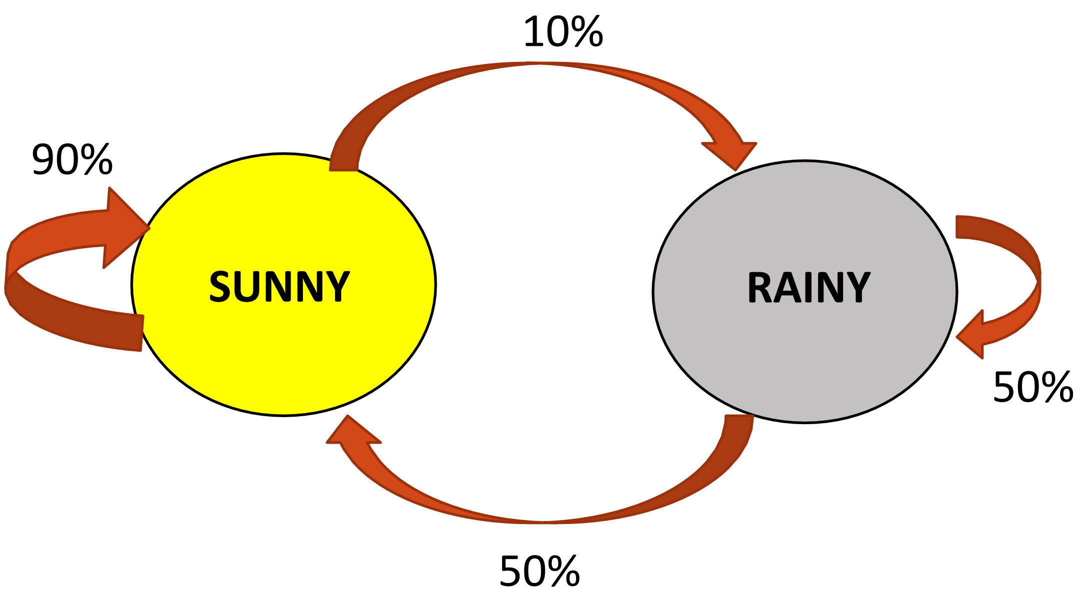
- Each entry is a non-negative real number representing a probability.
- (I,J) entry of the transition matrix has the probability of transitioning from state J to state I.
- Columns add up to one.



	Sunny	Rainy
Sunny	0.9	0.5
Rainy	0.1	0.5

Transition (or Markov) matrices

- Probability of being in one state at time t+1: depends on the probability of being in the current state (at time t).
  - memory less process.

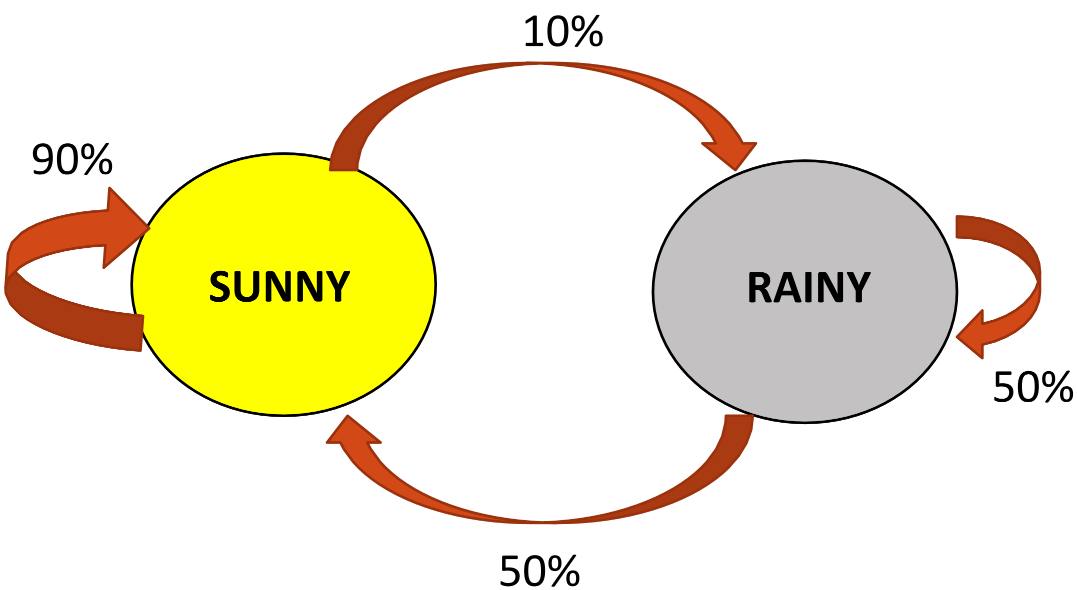


	Sunny	Rainy
Sunny	0.9	0.5
Rainy	0.1	0.5

Transition (or Markov) matrices

- Probability of being in one state at time t+1: depends on the probability of being in the current state (at time t).
  - memory less process.

$$P(X_n = j | \text{values of all previous states}) = P(X_n = j | X_{n-1})$$



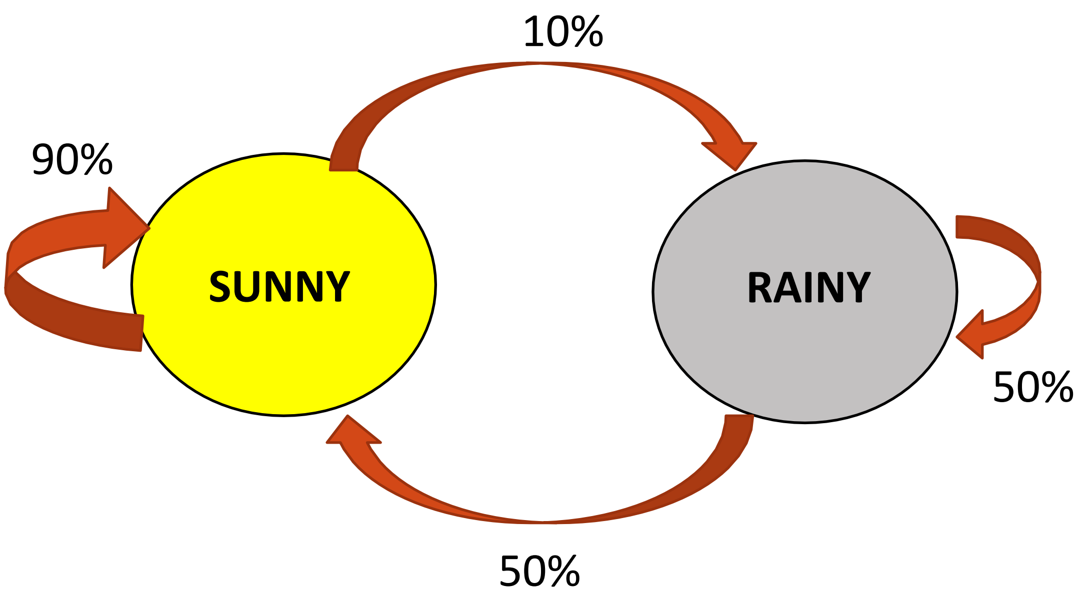
	Sunny	Rainy
Sunny	0.9	0.5
Rainy	0.1	0.5

Transition (or Markov) matrices

- Probability of being in one state at time t+1: depends on the probability of being in the current state (at time t).
  - memory less process.

$$P(X_n = j | \text{values of all previous states}) = P(X_n = j | X_{n-1})$$

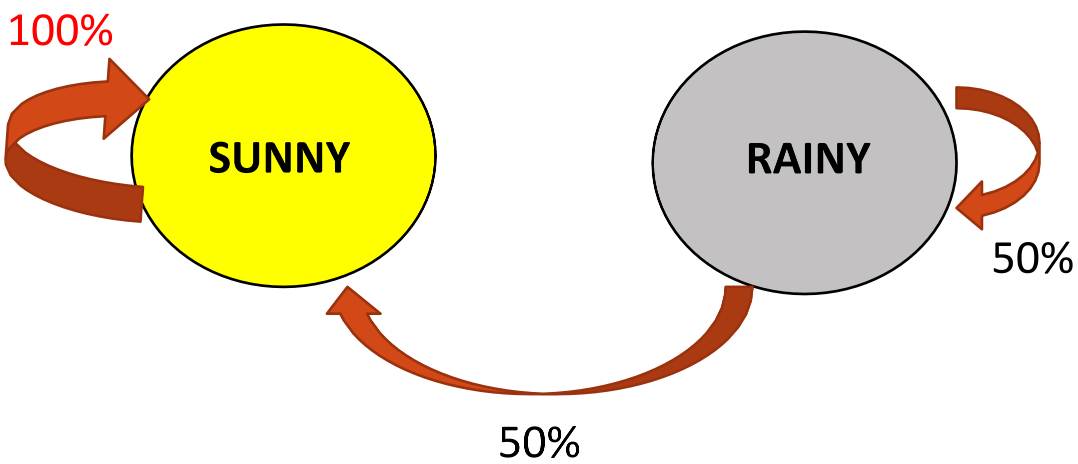
- This is called the **Markov property**, and the model is called a **Markov chain**



	Sunny	Rainy
Sunny	0.9	0.5
Rainy	0.1	0.5

Absorbing state

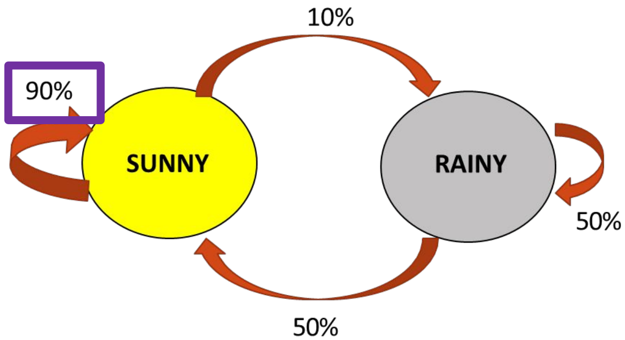
States in a Markov chain that it can never leave



	Sunny	Rainy
Sunny	1.0	0.5
Rainy	0.0	0.5

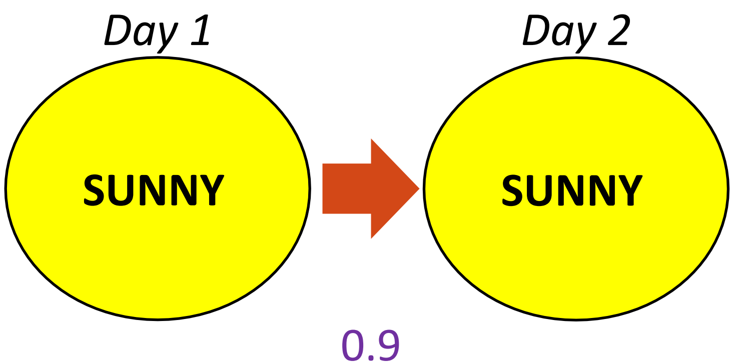
Transitions with biased random walk

Transition Matrix:



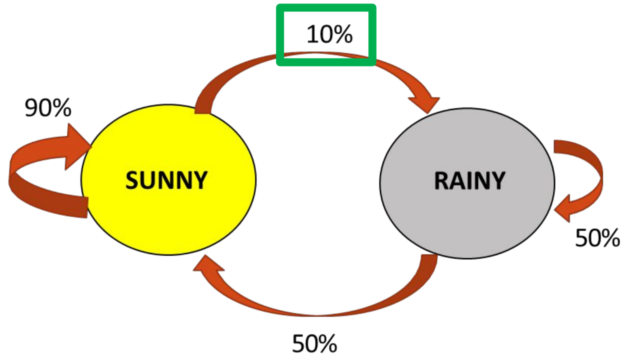
	Sunny	Rainy
Sunny	0.9	0.5
Rainy	0.1	0.5

Sequence A:



Transitions with biased random walk

Transition Matrix:



	Sunny	Rainy
Sunny	0.9	0.5
Rainy	$\boxed{0.1}$	0.5

Sequence A:

Day 1 **SUNNY** → *Day 2* **SUNNY** → *Day 3* **RAINY**

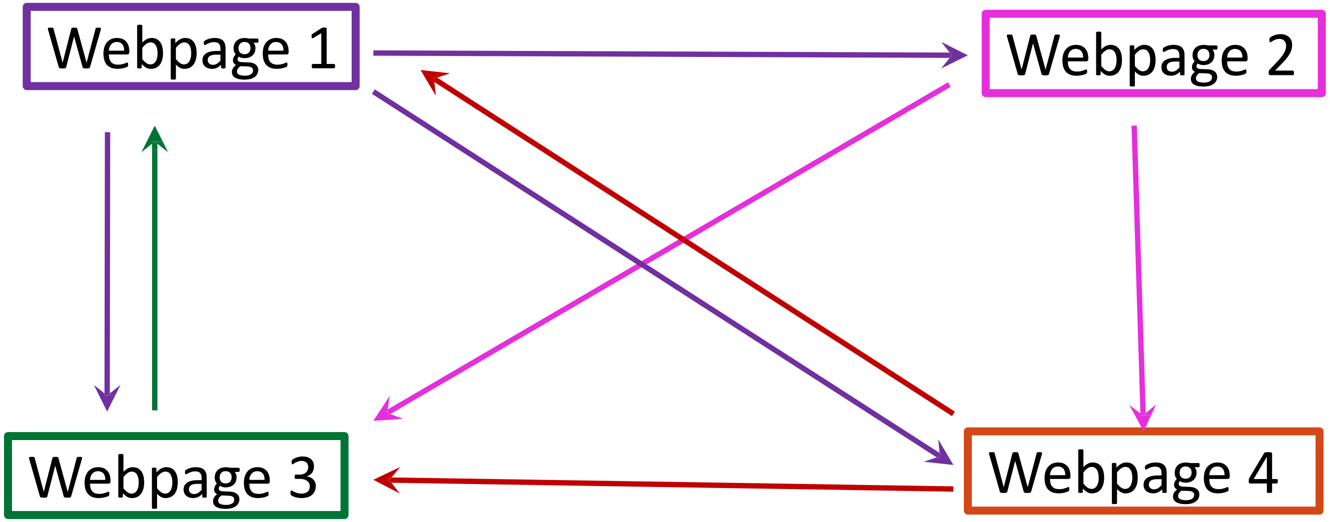
0.9 x 0.1 = 0.09

Sequence B:

Day 1 **SUNNY** → *Day 2* **RAINY** → *Day 3* **SUNNY**

0.1 x 0.5 = 0.05

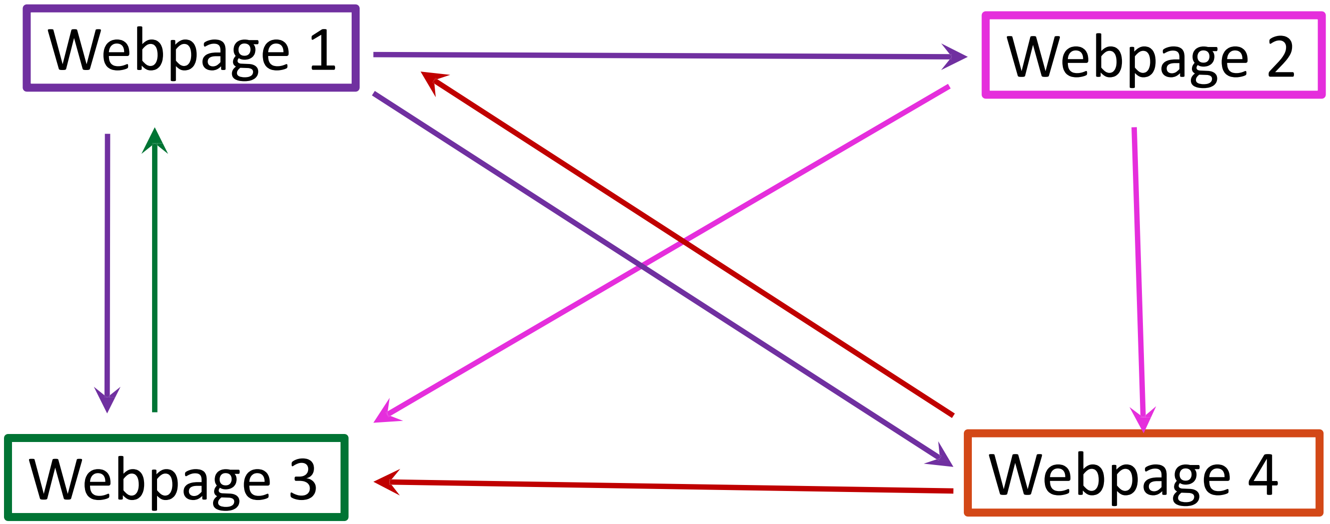
Random Walk Applications: Page Rank



Problem: Consider n linked web pages (above we have n = 4). Rank them.

- A link to a page increases the perceived ***importance*** of a webpage
- We can represent the ***importance*** of each webpage k with the scalar x_1

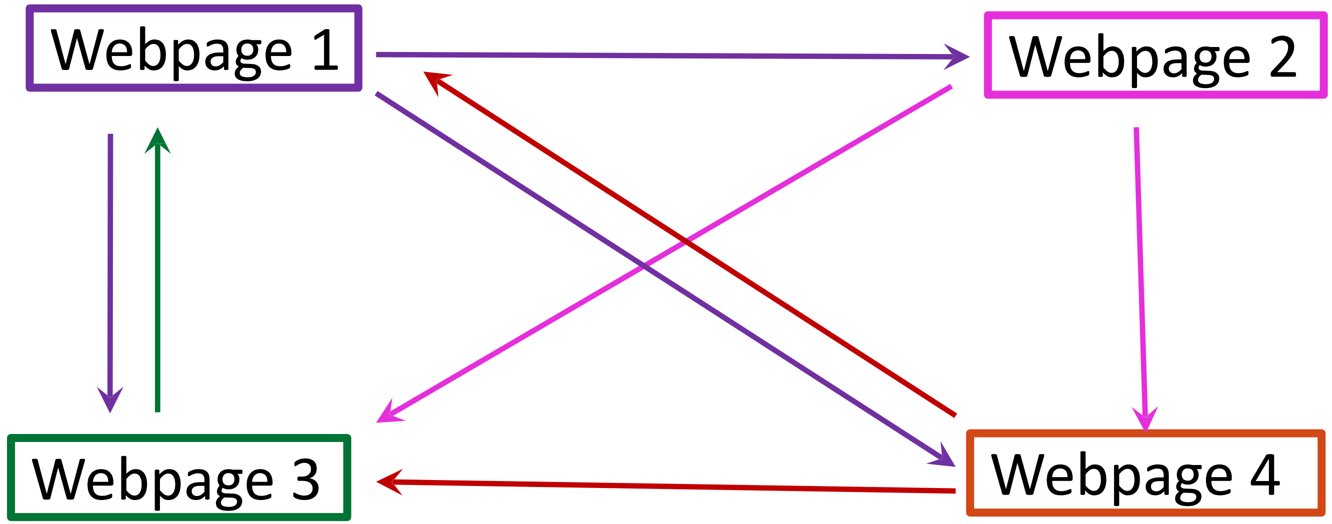
Page Rank



A possible way to rank web pages

- x_k is the number of links to page k (**incoming links**)
- $x_1 = 2$, $x_2 = 1$, $x_3 = 3$, $x_4 = 2$
- **Issue:** Doesn't take into account popularity / credibility of certain sources over others.

Page Rank

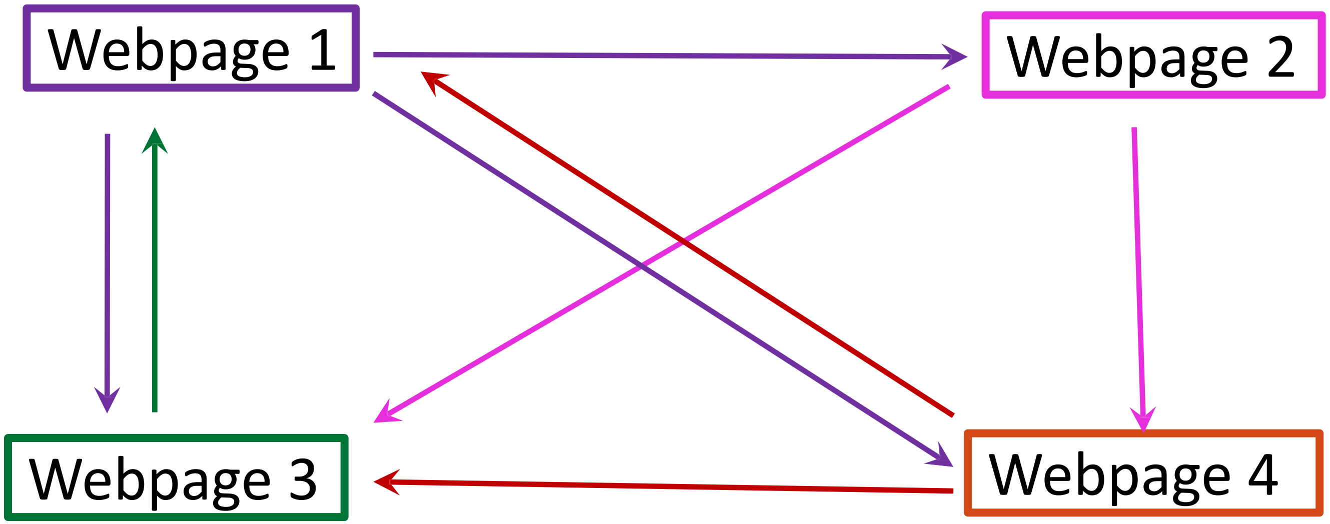


A possible way to rank web pages

- x_k is the number of links to page k (**incoming links**)
- $x_1 = 2$, $x_2 = 1$, $x_3 = 3$, $x_4 = 2$
- **Issue:** Doesn't take into account popularity / credibility of certain sources over others.

•**Alternatively,** importance of a web page $\propto$ frequency of page visits

Page Rank



A possible way to rank web pages

- x_k is the number of links to page k (**incoming links**)
- $x_1 = 2$, $x_2 = 1$, $x_3 = 3$, $x_4 = 2$
- **Issue:** Doesn't take into account popularity / credibility of certain sources over others.

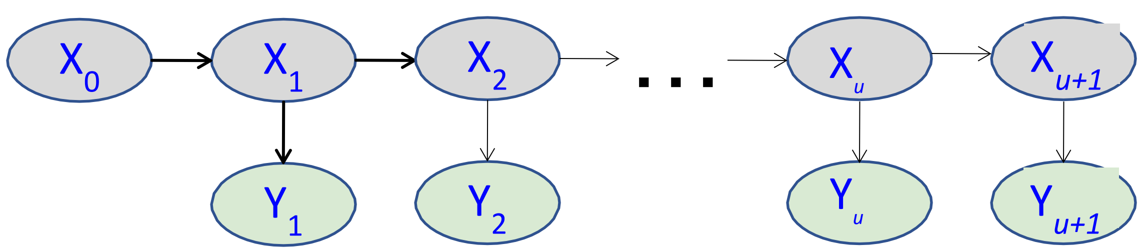
•**Alternatively,** importance of a web page $\propto$ number of outgoing links

Today

- Markov Chain
- **Hidden Markov Model**
- Decoding HMMs

Hidden Markov Models

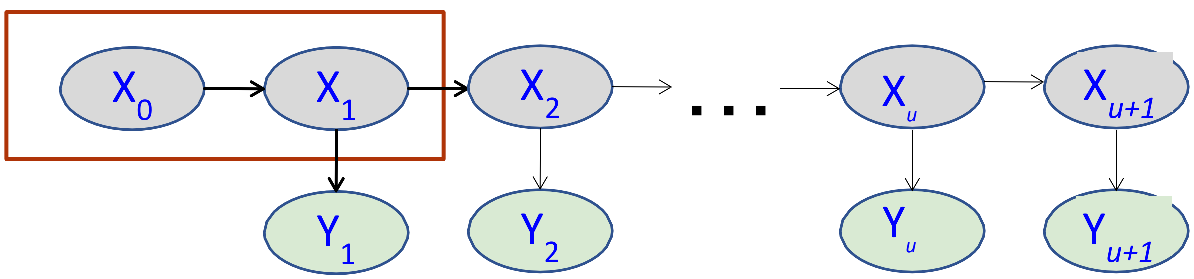
- At each time slice *t*, the state of the world is described by an unobservable variable X_u and an observable *evidence* variable Y_u



Hidden Markov Models

- At each time slice *t*, the state of the world is described by an unobservable variable X_u and an observable *evidence* variable Y_u
- **Transition model:** distribution over the current state given the whole past history:

$$p_{ij} = P(X_{u+1} = j | X_u = i)$$

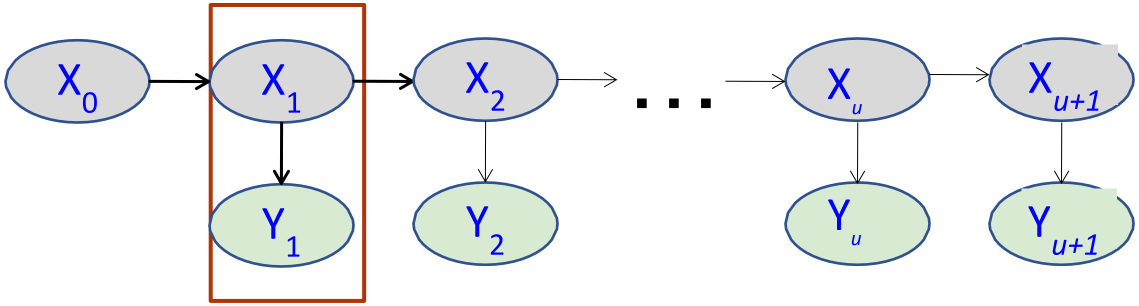


Hidden Markov Models

- At each time slice *t*, the state of the world is described by an unobservable variable X_u and an observable *evidence* variable Y_u
- **Transition model:** distribution over the current state given the whole past history:

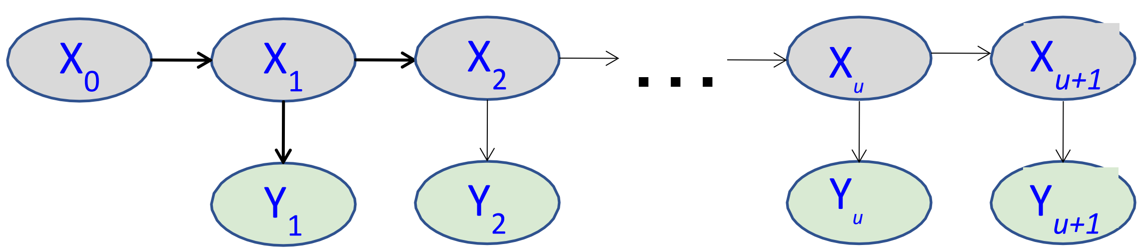
$$p_{ij} = P(X_{u+1} = j | X_u = i)$$

- **Observation model:** $P(Y_u | X_u = i) = q_i(Y_u)$



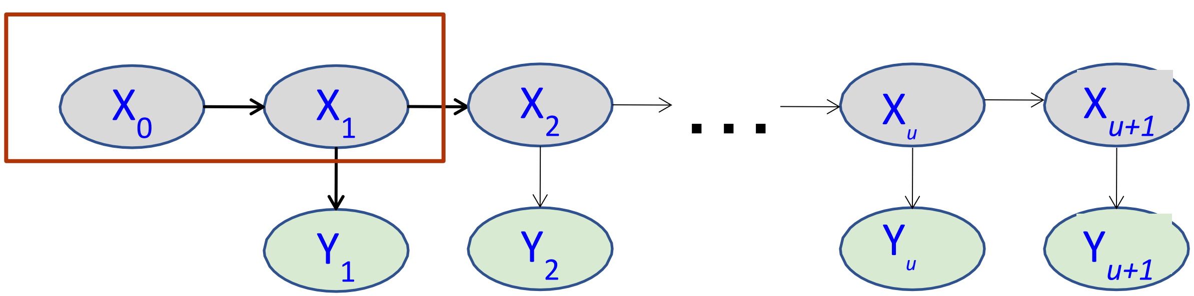
Hidden Markov Models

- **Markov assumption** (first order)
  - The current state is conditionally independent of all the other states given the state in the previous time step



Hidden Markov Models

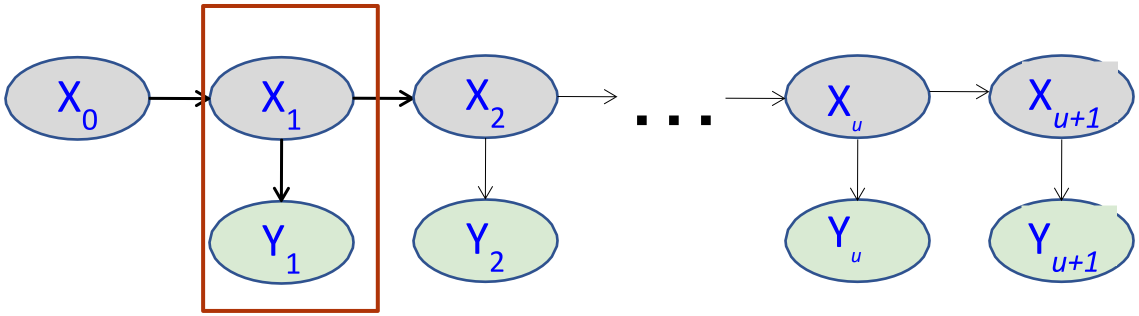
- **Markov assumption** (first order)
  - The current state is conditionally independent of all the other states given the state in the previous time step
  - What does $P(X_{u+1}|X_{o:u})$ simplify to?
    $P(X_{u+1}|X_{o:u}) = P(X_{u+1}|X_u)$



Hidden Markov Models

- Markov assumption for observations
  - The evidence at time *t* depends only on the state at time *t*
  - What does $P(Y_{u+1}|X_{u+1}, X_{0:u})$ simplify to?

$$P(Y_{u+1}|X_{u+1}, X_{0:u}) \qquad = P(Y_{u+1}|X_{u+1})$$



Comparing frameworks

Markov Chain

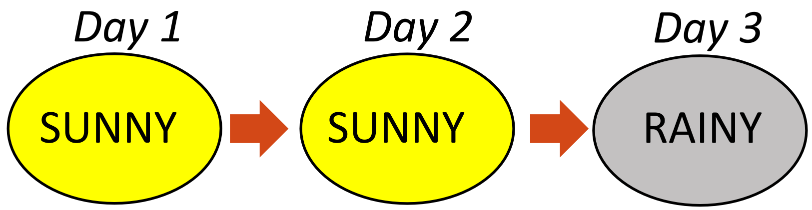
- Finite states
- Probabilistic formulation for transitions between states
- Markov property- next state determined only by current state

Hidden Markov Model

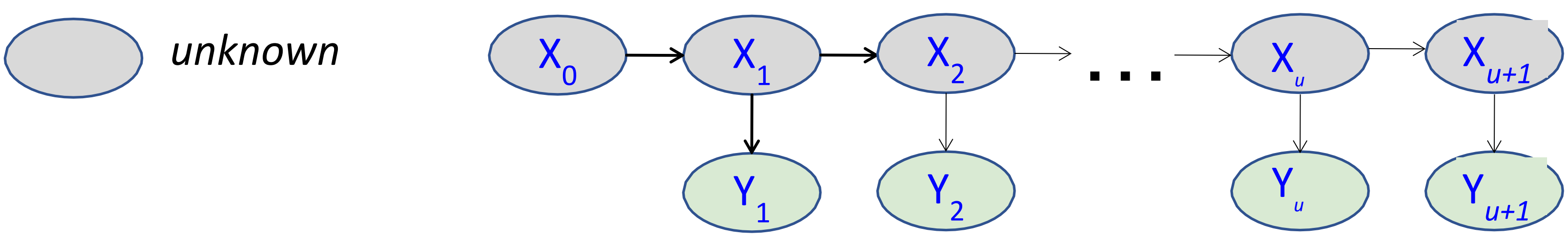
- Finite states
- Probabilistic formulation for transitions between states
- Markov property- next state determined only by current state
- **Current states are not observed.**

Markov vs Hidden

Markov



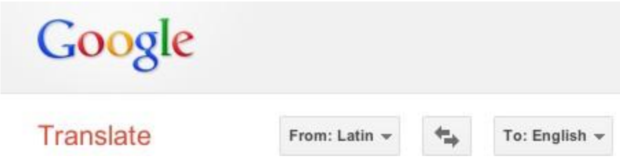
Hidden



Example HMM Applications

- Speech recognition:
  - Observations are acoustic signals (continuous valued)
  - States are specific positions in specific words (so, tens of thousands)

- Machine translation:
  - Observations are words (tens of thousands)
  - States are translation options



- Robot tracking:
  - Observations are range readings (continuous)
  - States are positions on a map (continuous)

Source: Tamara Berg

Example HMM Applications

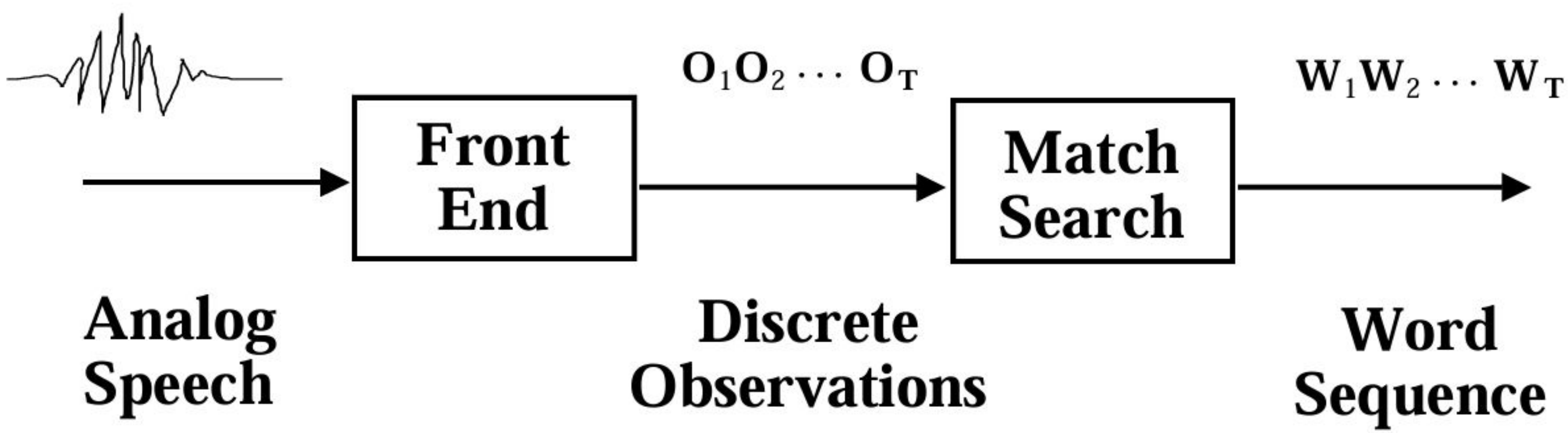
- Speech recognition:
  - Observations are acoustic signals (continuous valued)
  - States are specific positions in specific words (so, tens of thousands)

- Machine translation:
  - Observations are words (tens of thousands)
  - States are translation options

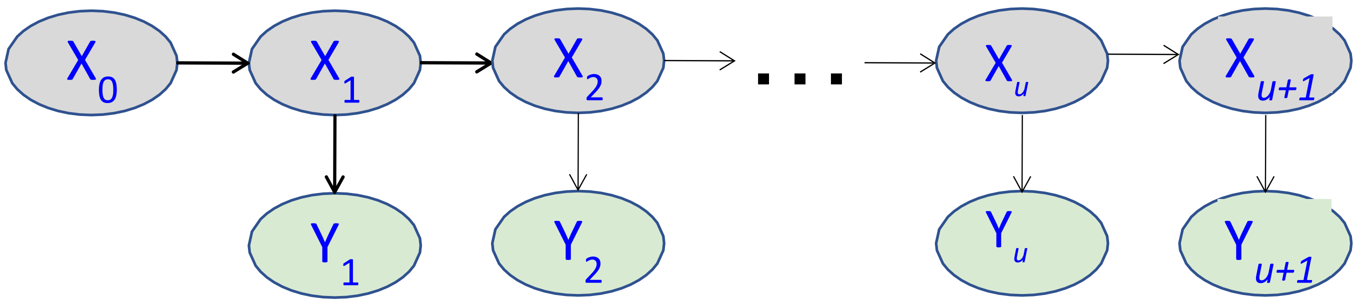
- Robot tracking:
  - Observations are range readings (continuous)
  - States are positions on a map (continuous)

Source: Tamara Berg

Speech recognition



Speech recognition



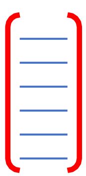
- **X:** Phones (in phonetics), i.e., concrete sound realizations.
  - **Unobserved**
- **Y:** audio utterances
  - **Observed**
  - Can extract features and represent them.

Example: Speech Recognition

- **Representing observations:** FFT of of the speech signal.

Acoustic wave form
Sampled at 16KHz, quantized to 8-12 bits

Fast Fourier Transform (FFT) **of one frame (10ms)** is the HMM observation, once per 10ms

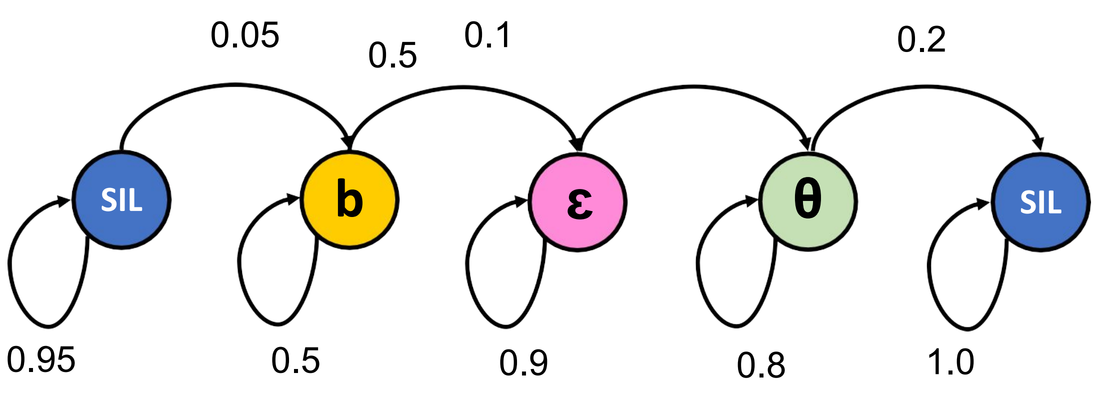


Observation = compressed version of the log magnitude FFT, from one 10ms frame

Example: Speech Recognition

- Observations: FFT of 10ms frame of the speech signal.
- Unobserved variables: a specific position in a specific word, coded using the international phonetic alphabet:
  - b = first sound of the word "Beth"
  - ε = second sound of the word "Beth"
  - θ = third sound in the word "Beth"

Finite State Machine model of the word "Beth"



Which of the following statement(s) is true? Select all that apply.

Which of the following statement(s) is true? Select all that apply.

Slicing the continuous FFT signal every 10ms helps us extract discrete features ✓

96%

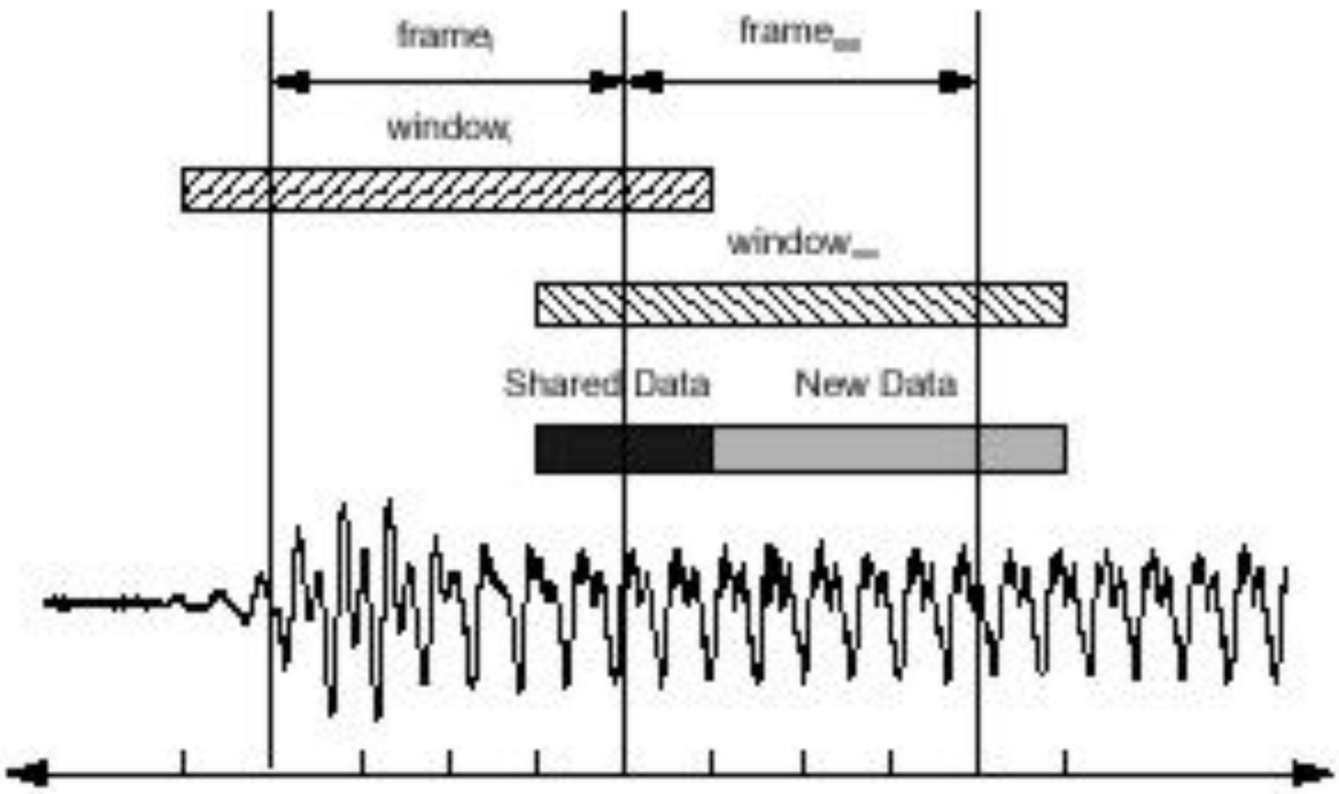
Slicing is ideal since most phones fall within the 10ms window

39%

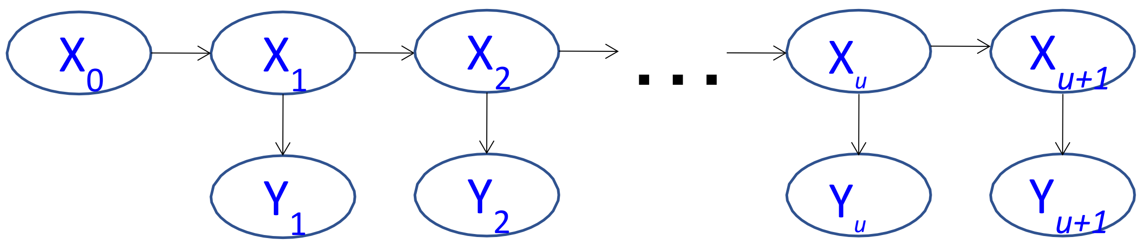
Slicing introduces noise because most phones may not fall within the 10ms window leading to incomplete or overlapping acoustic signals. ✓

63%

Compute features with a sliding window



The Joint Distribution

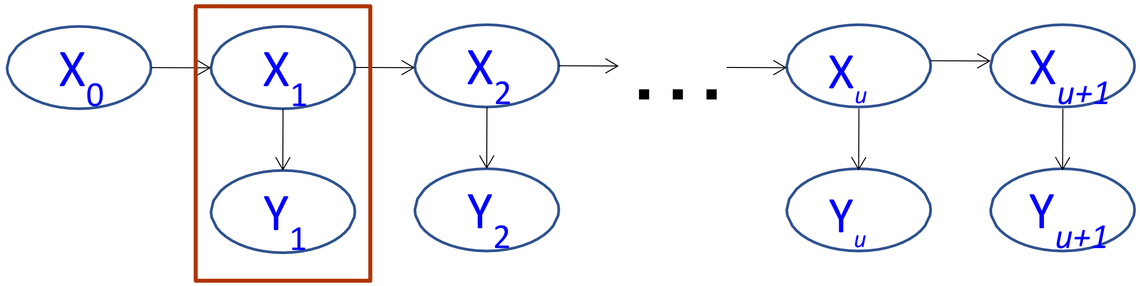


- Transition model: $P(X_{u+1} = j | X_u = i)$
- Observation model: $P(Y_u | X_u = i)$
- How do we compute the full joint probability table $P(X_{0:u+1} | Y_{0:u+1})$?

Bayes' Theorem

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

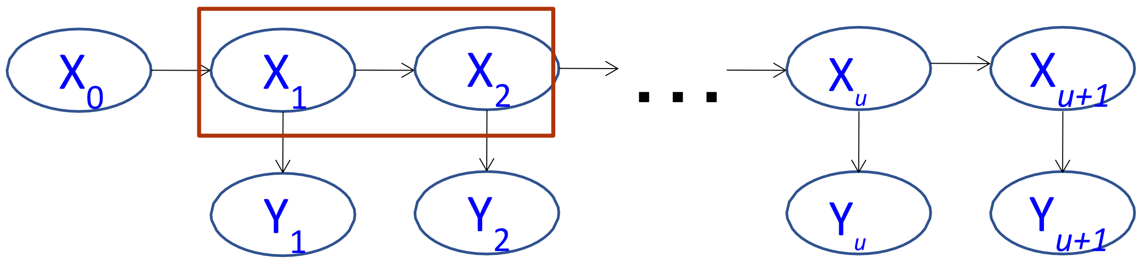
The Joint Distribution



- Transition model: $P(X_{u+1} = j | X_u = i)$
- Observation model: $P(Y_u | X_u = i)$
- How do we compute the full joint probability table $P(X_{0:u+1} | Y_{0:u+1})$?

$$\prod_{i=1}^{u+1} \qquad P(Y_i | X_i)$$

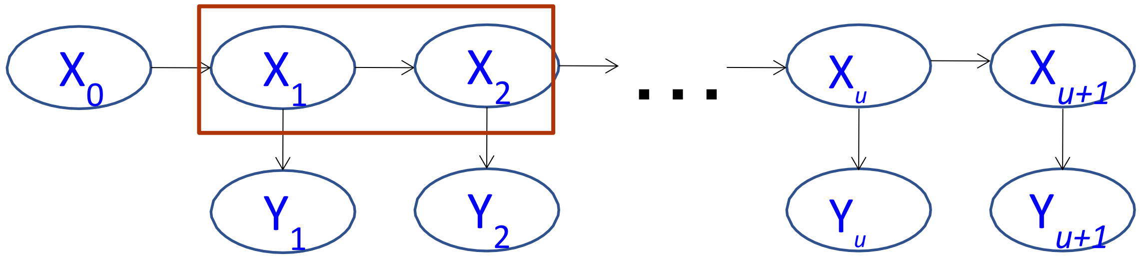
The Joint Distribution



- Transition model: $P(X_{u+1} = j | X_u = i)$
- Observation model: $P(Y_u | X_u = i)$
- How do we compute the full joint probability table $P(X_{0:u+1} | Y_{0:u+1})$?

$$\prod_{i=1}^{u+1} P(X_i | X_{i-1}) P(Y_i | X_i)$$

The Joint Distribution



- Transition model: $P(X_{u+1} = j | X_u = i)$
- Observation model: $P(Y_u | X_u = i)$
- How do we compute the full joint probability table $P(X_{0:u+1} | Y_{0:u+1})$?

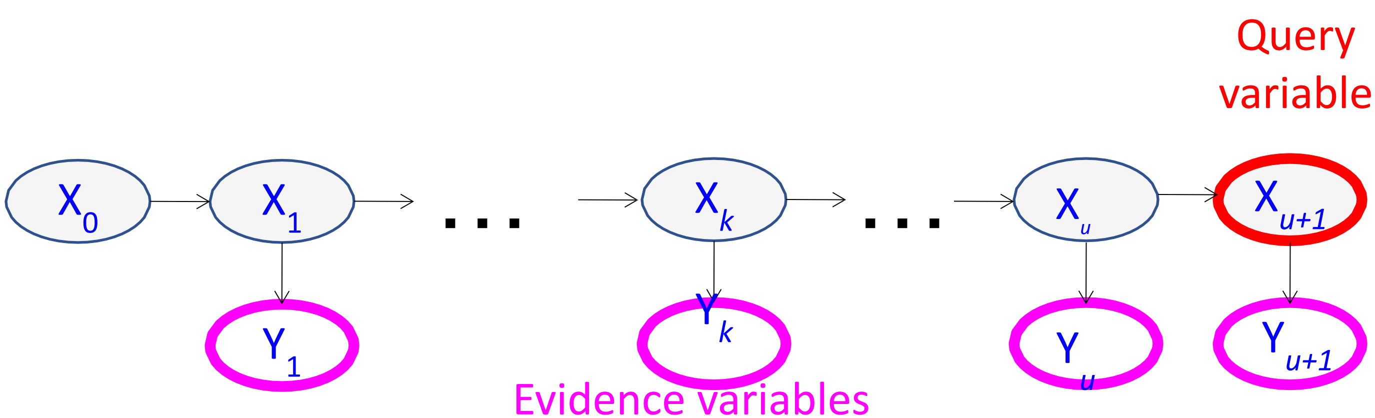
Bayes' Theorem

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

$$P(X_{0:u+1} | Y_{0:u+1}) = P(X_0) \prod_{i=1}^{u+1} P(X_i | X_{i-1}) P(Y_i | X_i)$$

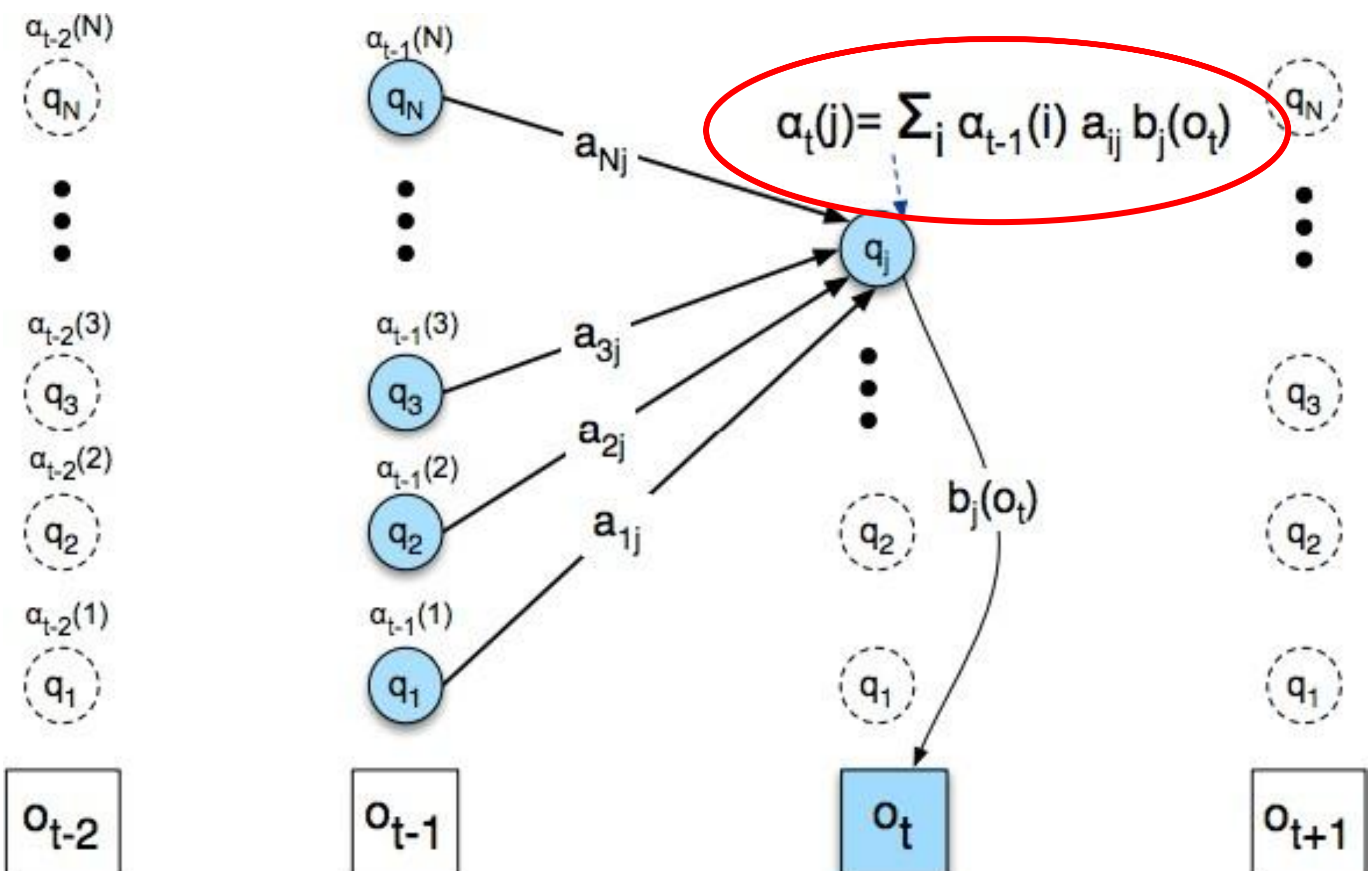
HMM inference tasks

- **Filtering:** what is the distribution over the current state X_t given all the evidence so far, $\mathbf{Y}_{1:t}$? (example: is it currently raining?)



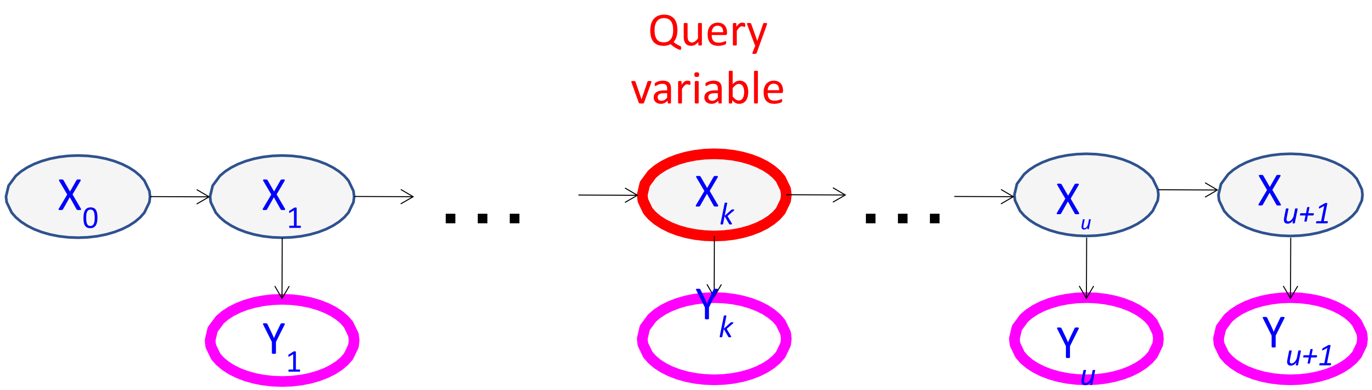
Forward algorithm

$\alpha_{t-1}(i)$	the **previous forward path probability** from the previous time step
a_{ij}	the **transition probability** from previous state q_i to current state q_j
$b_j(o_t)$	the **state observation likelihood** of the observation symbol o_t given the current state j



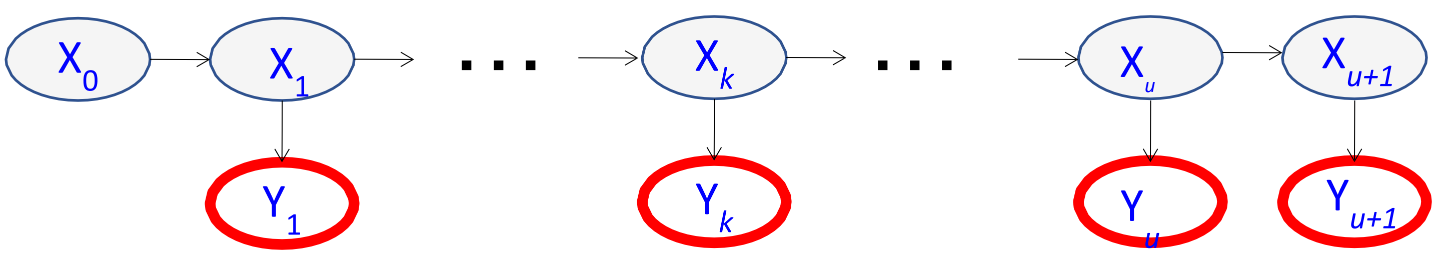
HMM inference tasks

- **Filtering:** what is the distribution over the current state X_t given all the evidence so far, $\mathbf{Y}_{1:t}$?
- **Smoothing:** what is the distribution of some state X_k (k<t) given the entire observation sequence $\mathbf{Y}_{1:t}$? (example: did it rain on Sunday?)



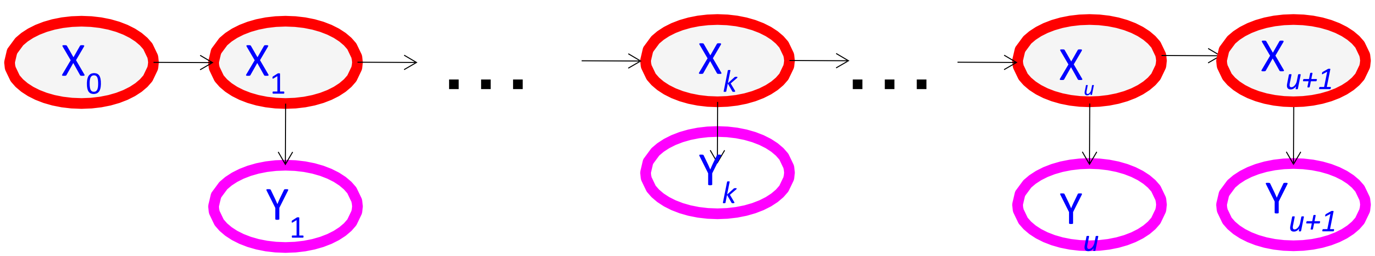
HMM inference tasks

- **Filtering:** what is the distribution over the current state X_t given all the evidence so far, $\mathbf{Y}_{1:t}$?
- **Smoothing:** what is the distribution of some state X_k (k<t) given the entire observation sequence $\mathbf{Y}_{1:t}$?
- **Evaluation:** compute the probability of a given observation sequence $\mathbf{Y}_{1:t}$



HMM inference tasks

- **Filtering:** what is the distribution over the current state X_t given all the evidence so far, $\mathbf{Y}_{1:t}$
- **Smoothing:** what is the distribution of some state X_k (k<t) given the entire observation sequence $\mathbf{Y}_{1:t}$?
- **Evaluation:** compute the probability of a given observation sequence $\mathbf{Y}_{1:t}$
- **Decoding:** what is the most likely state sequence $\mathbf{X}_{0:t}$ given the observation sequence $\mathbf{Y}_{1:t}$? (example: what's the weather every day?)



HMM Learning and Inference

- **Inference tasks**
  - **Filtering:** what is the distribution over the current state X_t given all the evidence so far, $\mathbf{Y}_{1:t}$
  - **Smoothing:** what is the distribution of some state X_k ($k<t$) given the entire observation sequence $\mathbf{Y}_{1:t}$?
  - **Evaluation:** compute the probability of a given observation sequence $\mathbf{Y}_{1:t}$
  - **Decoding:** what is the most likely state sequence $\mathbf{X}_{0:t}$ given the observation sequence $\mathbf{Y}_{1:t}$?
- **Learning**
  - Given a training sample of sequences, learn the model parameters (transition and emission probabilities)

HMM inference tasks

- **Filtering:** what is the distribution over the current state X_t given all the evidence so far, $\mathbf{Y}_{1:t}$
- **Smoothing:** what is the distribution of some state X_k ($k<t$) given the entire observation sequence $\mathbf{Y}_{1:t}$?
- **Evaluation:** compute the probability of a given observation sequence $\mathbf{Y}_{1:t}$
- **Decoding:** what is the most likely state sequence $\mathbf{X}_{0:t}$ given the observation sequence $\mathbf{Y}_{1:t}$? (example: what's the weather every day?)