Naïve Bayes

These slides were assembled by Byron Boots, with only minor modifications from Eric Eaton's slides and grateful acknowledgement to the many others who made their course materials freely available online. Feel free to reuse or adapt these slides for your own academic purposes, provided that you include proper attribution.

Bayes' Rule

- Recall Baye's Rule:

$$P(\text{hypothesis} \mid \text{evidence}) = \frac{P(\text{evidence} \mid \text{hypothesis}) \times P(\text{hypothesis})}{P(\text{evidence})}$$

- Equivalently, we can write:

$$P(Y = y_k \mid X = \mathbf{x}_i) = \frac{P(Y = y_k)P(X = \mathbf{x}_i \mid Y = y_k)}{P(X = \mathbf{x}_i)}$$

where X is a random variable representing the evidence and Y is a random variable for the label (our hypothesis)

- This is actually short for:

$$P(Y = y_k \mid X = \mathbf{x}_i) = \frac{P(Y = y_k)P(X_1 = x_{i,1} \wedge \ldots \wedge X_d = x_{i,d} \mid Y = y_k)}{P(X_1 = x_{i,1} \wedge \ldots \wedge X_d = x_{i,d})}$$

where X_j denotes the random variable for the j^{th} feature

Naïve Bayes Classifier

Idea: Use the training data to estimate

$P(X \mid Y)$ and $P(Y)$.

Then, use Bayes rule to infer $P(Y|X_{\text{new}})$ for new data

---

Easy to estimate from data

Impractical, but necessary

$$P(Y = y_k \mid X = \mathbf{x}_i) = \frac{P(Y = y_k) P(X_1 = x_{i,1} \wedge \ldots \wedge X_d = x_{i,d} \mid Y = y_k)}{P(X_1 = x_{i,1} \wedge \ldots \wedge X_d = x_{i,d})}$$

Unnecessary, as it turns out

- Estimating the joint probability distribution $P(X_1, X_2, \ldots, X_d \mid Y)$ is not practical

Naïve Bayes Classifier

Problem: estimating the joint PD or CPD isn't practical

– Can severely overfit, as we saw before

However, if we make the assumption that the attributes are independent given the class label, estimation is easy!

$$P(X_1, X_2, \ldots, X_d \mid Y) = \prod_{j=1}^{d} P(X_j \mid Y)$$

- In other words, we assume all attributes are *conditionally independent* given *Y*
- Often this assumption is violated in practice, but more on that later…

Training Naïve Bayes

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting!

Sky	**Temp**	**Humid**	**Wind**	**Water**	**Forecast**	**Play?**
sunny	warm	normal	strong	warm	same	*yes*
sunny	warm	high	strong	warm	same	*yes*
rainy	cold	high	strong	warm	change	*no*
sunny	warm	high	strong	cool	change	*yes*

P(play) = ?

P(Sky = sunny | play) = ?

P(Humid = high | play) = ?

...

P(¬play) = ?

P(Sky = sunny | ¬play) = ?

P(Humid = high | ¬play) = ?

...

Training Naïve Bayes

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting!

Sky	Temp	Humid	Wind	Water	Forecast	Play?
sunny	warm	normal	strong	warm	same	*yes*
sunny	warm	high	strong	warm	same	*yes*
rainy	cold	high	strong	warm	change	*no*
sunny	warm	high	strong	cool	change	*yes*

P(play) = ?

P(Sky = sunny | play) = ?

P(Humid = high | play) = ?

...

P(¬play) = ?

P(Sky = sunny | ¬play) = ?

P(Humid = high | ¬play) = ?

...

Training Naïve Bayes

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting!

Sky	**Temp**	**Humid**	**Wind**	**Water**	**Forecast**	**Play?**
sunny	warm	normal	strong	warm	same	*yes*
sunny	warm	high	strong	warm	same	*yes*
rainy	cold	high	strong	warm	change	*no*
sunny	warm	high	strong	cool	change	*yes*

P(play) = 3/4

P(Sky = sunny | play) = ?

P(Humid = high | play) = ?

...

P(¬play) = 1/4

P(Sky = sunny | ¬play) = ?

P(Humid = high | ¬play) = ?

...

Training Naïve Bayes

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting!

Sky	Temp	Humid	Wind	Water	Forecast	Play?
sunny						*yes*
sunny						*yes*
rainy	cold	high	strong	warm	change	*no*
sunny						*yes*

P(play) = 3/4

P(¬play) = 1/4

P(Sky = sunny | play) = ?

P(Sky = sunny | ¬play) = ?

P(Humid = high | play) = ?

P(Humid = high | ¬play) = ?

...

...

Training Naïve Bayes

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting!

Sky	Temp	Humid	Wind	Water	Forecast	Play?
sunny						*yes*
sunny						*yes*
rainy	cold	high	strong	warm	change	*no*
sunny						*yes*

P(play) = 3/4

P(Sky = sunny | play) = 1

P(Humid = high | play) = ?

...

P(¬play) = 1/4

P(Sky = sunny | ¬play) = ?

P(Humid = high | ¬play) = ?

...

Training Naïve Bayes

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting!

Sky	Temp	Humid	Wind	Water	Forecast	Play?
sunny	warm	normal	strong	warm	same	*yes*
sunny	warm	high	strong	warm	same	*yes*
rainy						*no*
sunny	warm	high	strong	cool	change	*yes*

P(play) = 3/4

P(¬play) = 1/4

P(Sky = sunny | play) = 1

P(Sky = sunny | ¬play) = ?

P(Humid = high | play) = ?

P(Humid = high | ¬play) = ?

...

...

Training Naïve Bayes

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting!

Sky	**Temp**	**Humid**	**Wind**	**Water**	**Forecast**	**Play?**
sunny	warm	normal	strong	warm	same	*yes*
sunny	warm	high	strong	warm	same	*yes*
rainy						*no*
sunny	warm	high	strong	cool	change	*yes*

P(play) = 3/4

P(¬play) = 1/4

P(Sky = sunny | play) = 1

P(Sky = sunny | ¬play) = 0

P(Humid = high | play) = ?

P(Humid = high | ¬play) = ?

...

...

Training Naïve Bayes

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting!

Sky	**Temp**	**Humid**	**Wind**	**Water**	**Forecast**	**Play?**
		normal				*yes*
		high				*yes*
rainy	cold	high	strong	warm	change	*no*
		high				*yes*

P(play) = 3/4

P(¬play) = 1/4

P(Sky = sunny | play) = 1

P(Sky = sunny | ¬play) = 0

P(Humid = high | play) = ?

P(Humid = high | ¬play) = ?

...

...

Training Naïve Bayes

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting!

Sky	Temp	Humid	Wind	Water	Forecast	Play?
		normal				*yes*
		high				*yes*
rainy	cold	high	strong	warm	change	*no*
		high				*yes*

P(play) = 3/4

P(¬play) = 1/4

P(Sky = sunny | play) = 1

P(Sky = sunny | ¬play) = 0

P(Humid = high | play) = 2/3

P(Humid = high | ¬play) = ?

...

...

Training Naïve Bayes

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting!

Sky	**Temp**	**Humid**	**Wind**	**Water**	**Forecast**	**Play?**
sunny	warm	normal	strong	warm	same	*yes*
sunny	warm	high	strong	warm	same	*yes*
		high				*no*
sunny	warm	high	strong	cool	change	*yes*

P(play) = 3/4

P(¬play) = 1/4

P(Sky = sunny | play) = 1

P(Sky = sunny | ¬play) = 0

P(Humid = high | play) = 2/3

P(Humid = high | ¬play) = ?

...

...

Training Naïve Bayes

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting!

Sky	**Temp**	**Humid**	**Wind**	**Water**	**Forecast**	**Play?**
sunny	warm	normal	strong	warm	same	*yes*
sunny	warm	high	strong	warm	same	*yes*
		high				*no*
sunny	warm	high	strong	cool	change	*yes*

P(play) = 3/4

P(Sky = sunny | play) = 1

P(Humid = high | play) = 2/3

...

P(¬play) = 1/4

P(Sky = sunny | ¬play) = 0

P(Humid = high | ¬play) = 1

...

Laplace Smoothing

- Notice that some probabilities estimated by counting might be zero
  - Possible overfitting!

- Fix by using Laplace smoothing:
  - Adds 1 to each count

$$P(X_j = v \mid Y = y_k) = \frac{\mathrm{c}_v + 1}{\displaystyle\sum_{v' \in \mathrm{values}(X_j)} \mathrm{c}_{v'} + |\mathrm{values}(X_j)|}$$

  where

  - c_v is the count of training instances with a value of v for attribute j and class label y_k
  - $|\text{values}(X_j)|$ is the number of values X_j can take on

Training Naïve Bayes with Laplace Smoothing

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting with Laplace smoothing:

Sky	Temp	Humid	Wind	Water	Forecast	Play?
sunny						*yes*
sunny						*yes*
rainy	cold	high	strong	warm	change	*no*
sunny						*yes*

P(play) = 3/4 P(¬play) = 1/4

P(Sky = sunny | play) = 4/5 P(Sky = sunny | ¬play) = ?

P(Humid = high | play) = ? P(Humid = high | ¬play) = ?

... ...

Training Naïve Bayes with Laplace Smoothing

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting with Laplace smoothing:

Sky	**Temp**	**Humid**	**Wind**	**Water**	**Forecast**	**Play?**
sunny	warm	normal	strong	warm	same	*yes*
sunny	warm	high	strong	warm	same	*yes*
rainy						*no*
sunny	warm	high	strong	cool	change	*yes*

P(play) = 3/4 P(¬play) = 1/4

P(Sky = sunny | play) = 4/5 P(Sky = sunny | ¬play) = 1/3

P(Humid = high | play) = ? P(Humid = high | ¬play) = ?

... ...

Training Naïve Bayes with Laplace Smoothing

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting with Laplace smoothing:

Sky	**Temp**	**Humid**	**Wind**	**Water**	**Forecast**	**Play?**
		normal				*yes*
		high				*yes*
rainy	cold	high	strong	warm	change	*no*
		high				*yes*

P(play) = 3/4

P(¬play) = 1/4

P(Sky = sunny | play) = 4/5

P(Sky = sunny | ¬play) = 1/3

P(Humid = high | play) = 3/5

P(Humid = high | ¬play) = ?

...

...

Training Naïve Bayes with Laplace Smoothing

Estimate $P(X_j \mid Y)$ and $P(Y)$ directly from the training data by counting with Laplace smoothing:

Sky	**Temp**	**Humid**	**Wind**	**Water**	**Forecast**	**Play?**
sunny	warm	normal	strong	warm	same	*yes*
sunny	warm	high	strong	warm	same	*yes*
		high				*no*
sunny	warm	high	strong	cool	change	*yes*

P(play) = 3/4 P(¬play) = 1/4

P(Sky = sunny | play) = 4/5 P(Sky = sunny | ¬play) = 1/3

P(Humid = high | play) = 3/5 P(Humid = high | ¬play) = 2/3

... ...

Using the Naïve Bayes Classifier

- Now, we have

$$P(Y = y_k \mid X = \mathbf{x}_i) = \frac{P(Y = y_k) \prod_{j=1}^{d} P(X_j = x_{i,j} \mid Y = y_k)}{\boxed{P(X = \mathbf{x}_i)}}$$

This is constant for a given instance, and so irrelevant to our prediction

– In practice, we use log-probabilities to prevent underflow

- To classify a new point **x**,

$$h(\mathbf{x}) = \arg\max_{y_k} \; P(Y = y_k) \prod_{j=1}^{d} P(X_j = \underbrace{x_j}_{j^{\text{th}} \text{ attribute value of } \mathbf{x}} \mid Y = y_k)$$

$$= \arg\max_{y_k} \; \log P(Y = y_k) + \sum_{j=1}^{d} \log P(X_j = x_j \mid Y = y_k)$$

The Naïve Bayes Classifier Algorithm

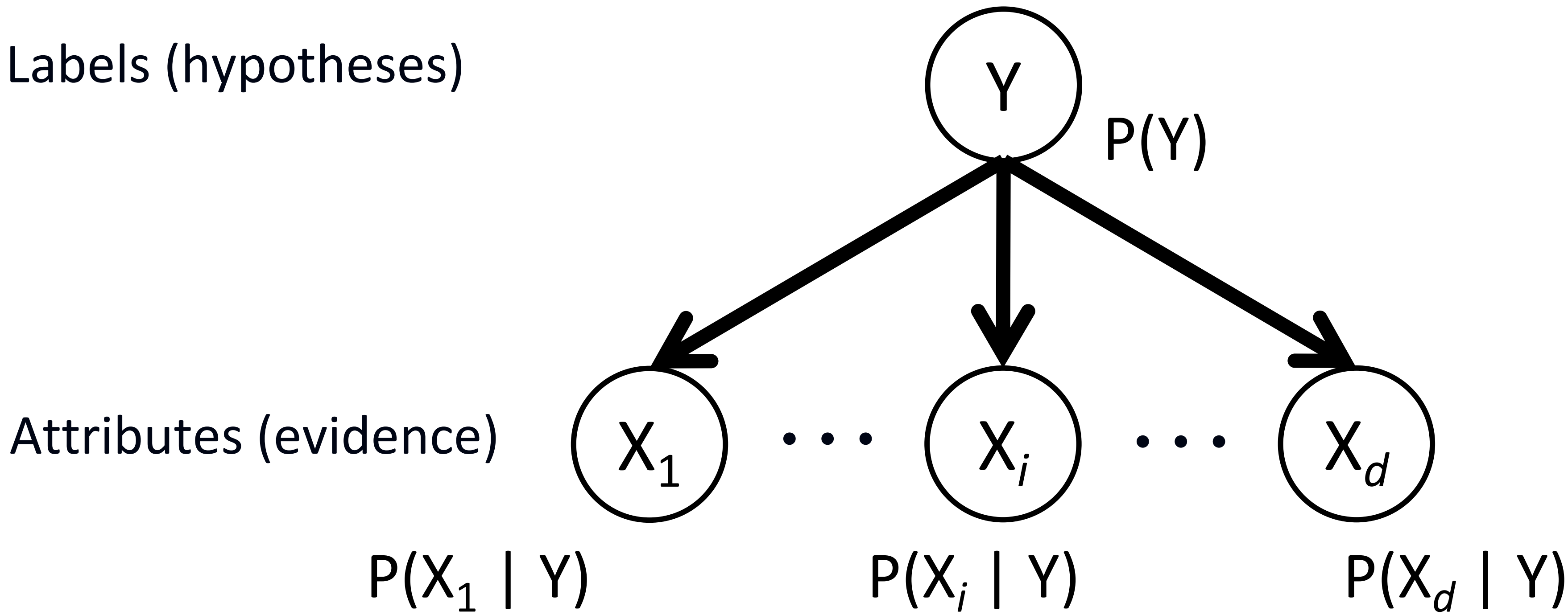
- For each class label y_k
  - Estimate $P(Y = y_k)$ from the data
  - For each value $x_{i,j}$ of each attribute X_i
    - Estimate $P(X_i = x_{i,j} \mid Y = y_k)$

- Classify a new point via:

$$h(\mathbf{x}) = \arg\max_{y_k} \; \log P(Y = y_k) + \sum_{j=1}^{d} \log P(X_j = x_j \mid Y = y_k)$$

- In practice, the independence assumption doesn't often hold true, but Naïve Bayes performs very well despite this

The Naïve Bayes Graphical Model

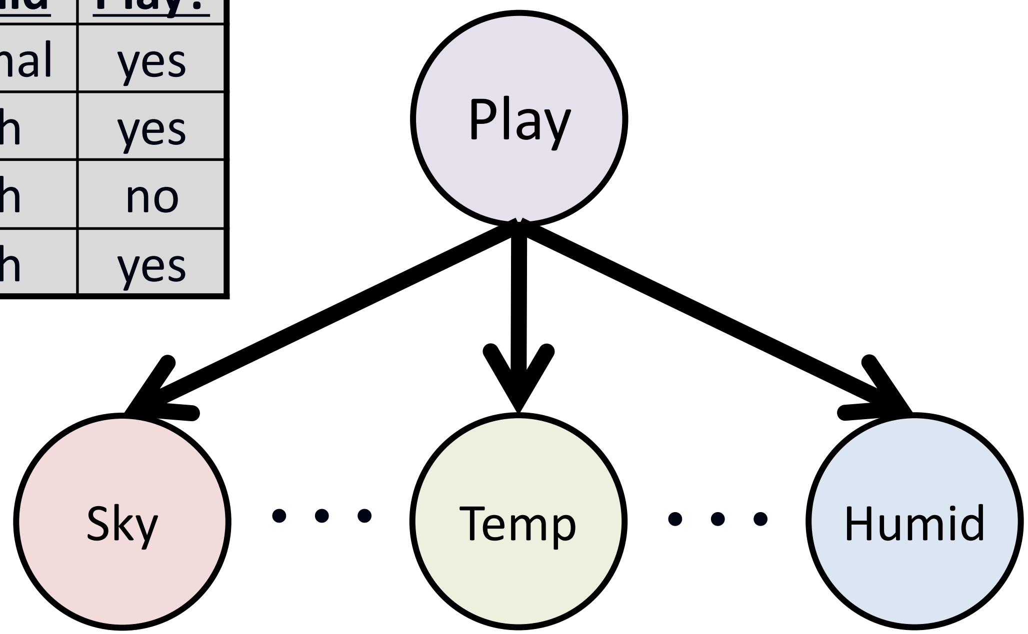


- Nodes denote random variables
- Edges denote dependency
- Each node has an associated conditional probability table (CPT), conditioned upon its parents

Example NB Graphical Model

Data:

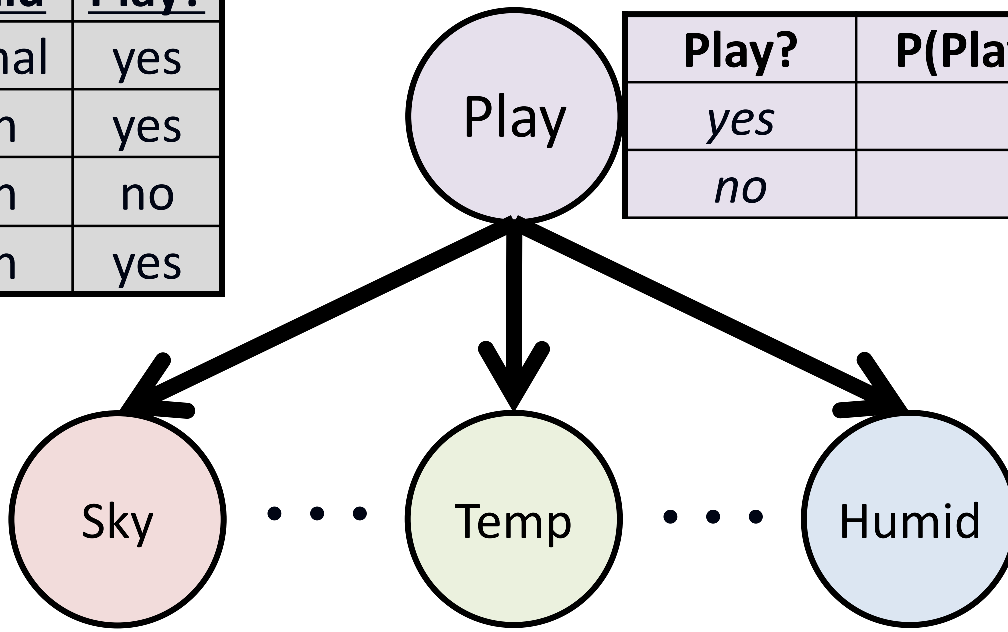
Sky	Temp	Humid	Play?
sunny	warm	normal	yes
sunny	warm	high	yes
rainy	cold	high	no
sunny	warm	high	yes



Example NB Graphical Model

Data:

Sky	Temp	Humid	Play?
sunny	warm	normal	yes
sunny	warm	high	yes
rainy	cold	high	no
sunny	warm	high	yes



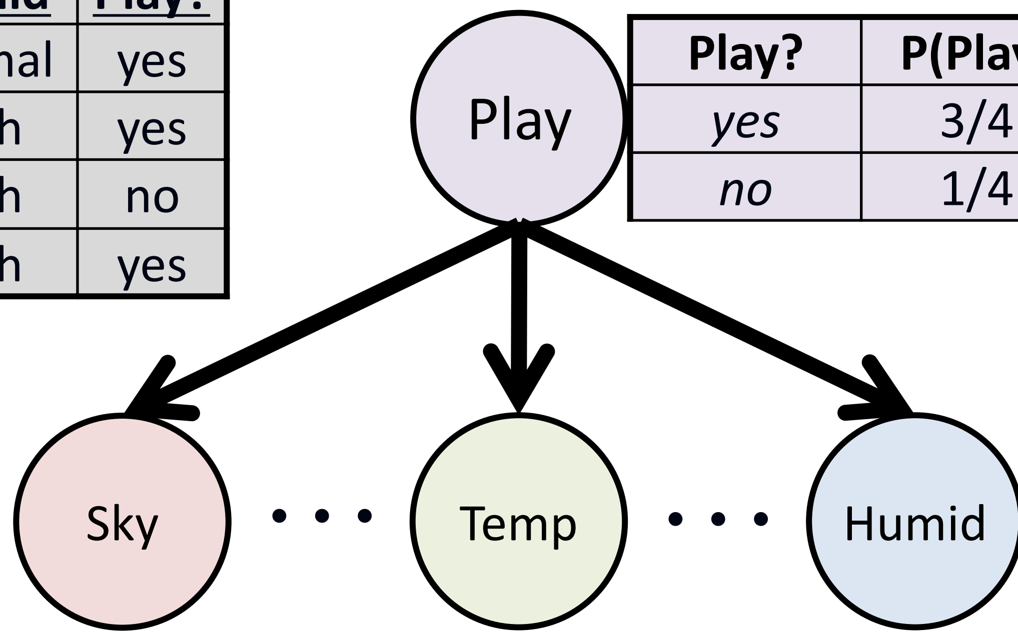
Play?	P(Play)
yes	
no	

Example NB Graphical Model

Data:

Sky	**Temp**	**Humid**	**Play?**
sunny	warm	normal	yes
sunny	warm	high	yes
rainy	cold	high	no
sunny	warm	high	yes

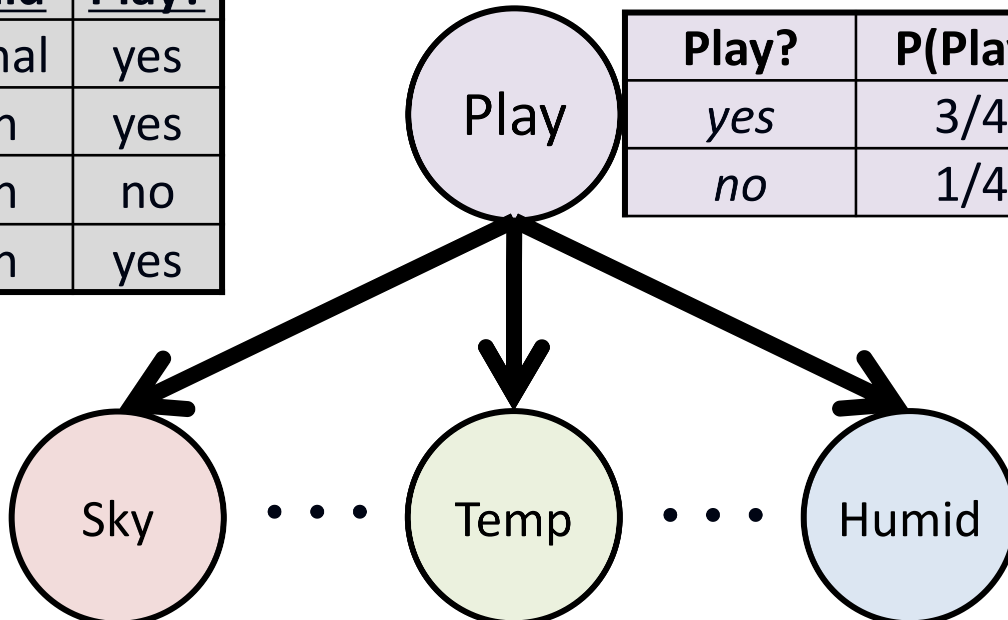
Play?	**P(Play)**
yes	3/4
no	1/4



Example NB Graphical Model

Data:

Sky	Temp	Humid	Play?
sunny	warm	normal	yes
sunny	warm	high	yes
rainy	cold	high	no
sunny	warm	high	yes



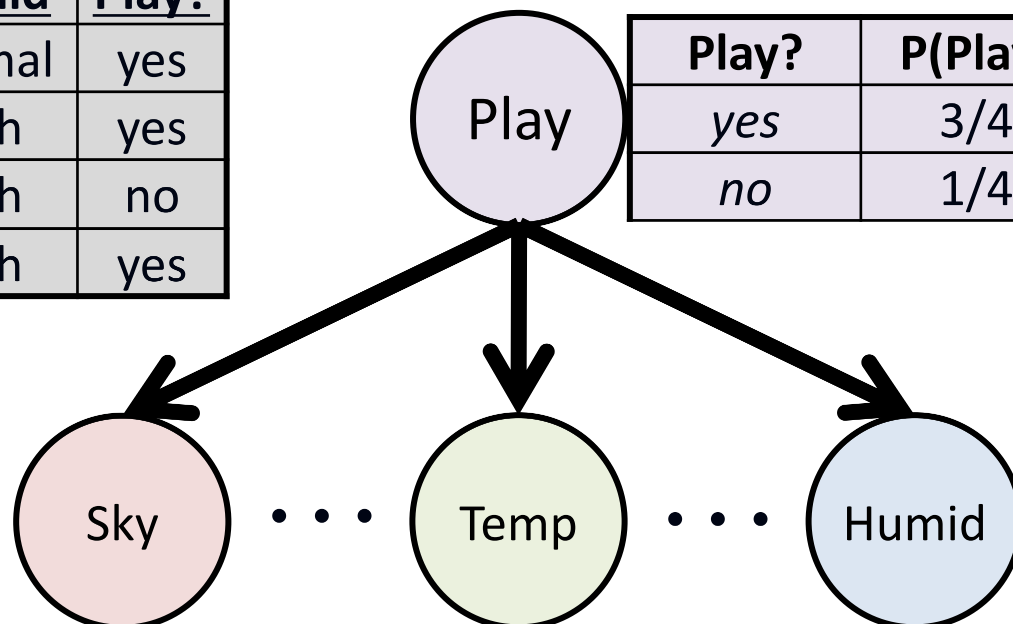
Play?	P(Play)
yes	3/4
no	1/4

Sky	Play?	P(Sky | Play)
sunny	*yes*	
rainy	*yes*	
sunny	*no*	
rainy	*no*	

Example NB Graphical Model

Data:

Sky	**Temp**	**Humid**	**Play?**
sunny	warm	normal	yes
sunny	warm	high	yes
rainy	cold	high	no
sunny	warm	high	yes



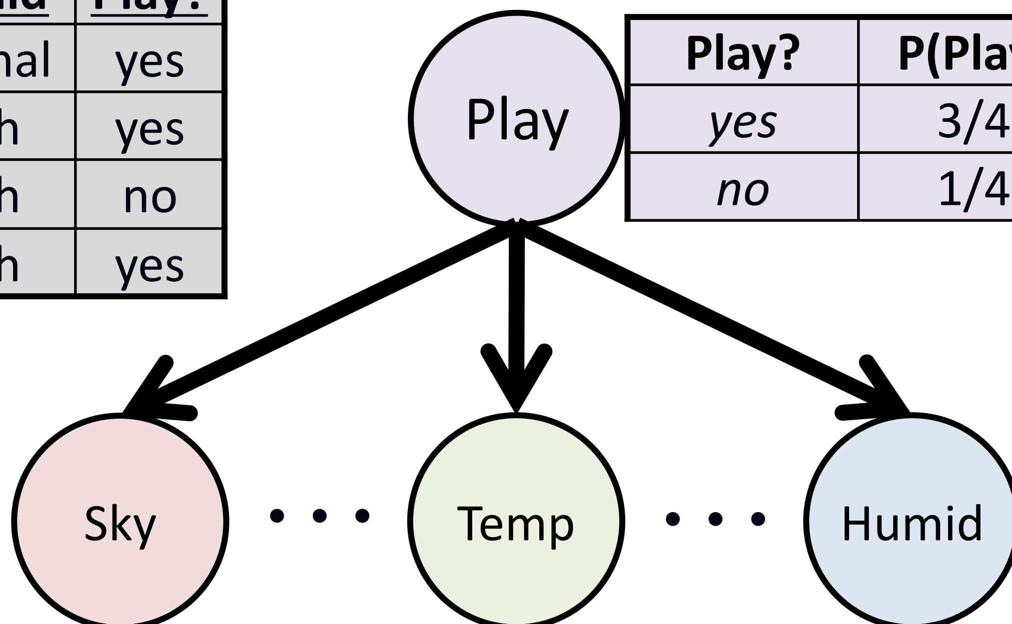
Play?	**P(Play)**
yes	3/4
no	1/4

Sky	**Play?**	**P(Sky | Play)**
sunny	*yes*	4/5
rainy	*yes*	1/5
sunny	*no*	1/3
rainy	*no*	2/3

Example NB Graphical Model

Data:

Sky	Temp	Humid	Play?
sunny	warm	normal	yes
sunny	warm	high	yes
rainy	cold	high	no
sunny	warm	high	yes



Play?	P(Play)
yes	3/4
no	1/4

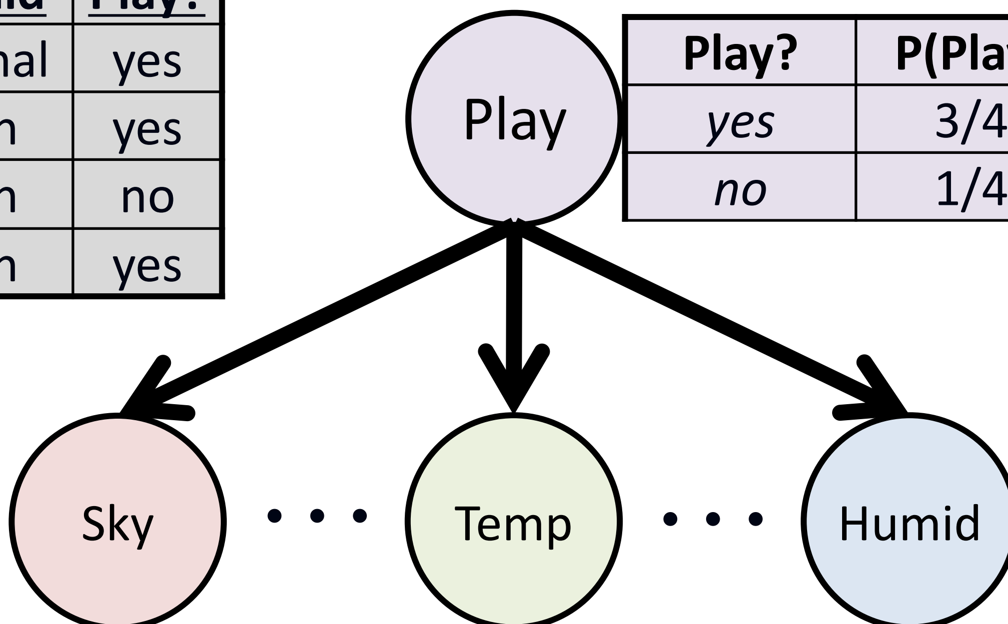
Sky	Play?	P(Sky | Play)
sunny	*yes*	4/5
rainy	*yes*	1/5
sunny	*no*	1/3
rainy	*no*	2/3

Temp	Play?	P(Temp | Play)
warm	*yes*	
cold	*yes*	
warm	*no*	
cold	*no*	

Example NB Graphical Model

Data:

Sky	Temp	Humid	Play?
sunny	warm	normal	yes
sunny	warm	high	yes
rainy	cold	high	no
sunny	warm	high	yes



Play?	P(Play)
yes	3/4
no	1/4

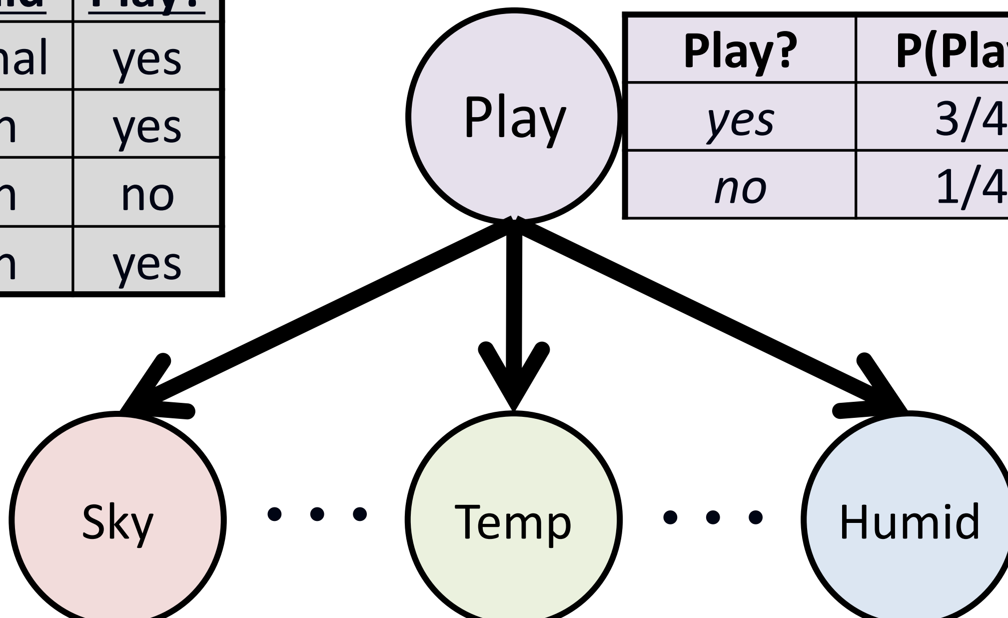
Sky	Play?	P(Sky | Play)
sunny	*yes*	4/5
rainy	*yes*	1/5
sunny	*no*	1/3
rainy	*no*	2/3

Temp	Play?	P(Temp | Play)
warm	*yes*	4/5
cold	*yes*	1/5
warm	*no*	1/3
cold	*no*	2/3

Example NB Graphical Model

Data:

Sky	**Temp**	**Humid**	**Play?**
sunny	warm	normal	yes
sunny	warm	high	yes
rainy	cold	high	no
sunny	warm	high	yes



Play?	**P(Play)**
yes	3/4
no	1/4

Sky	**Play?**	**P(Sky | Play)**
sunny	*yes*	4/5
rainy	*yes*	1/5
sunny	*no*	1/3
rainy	*no*	2/3

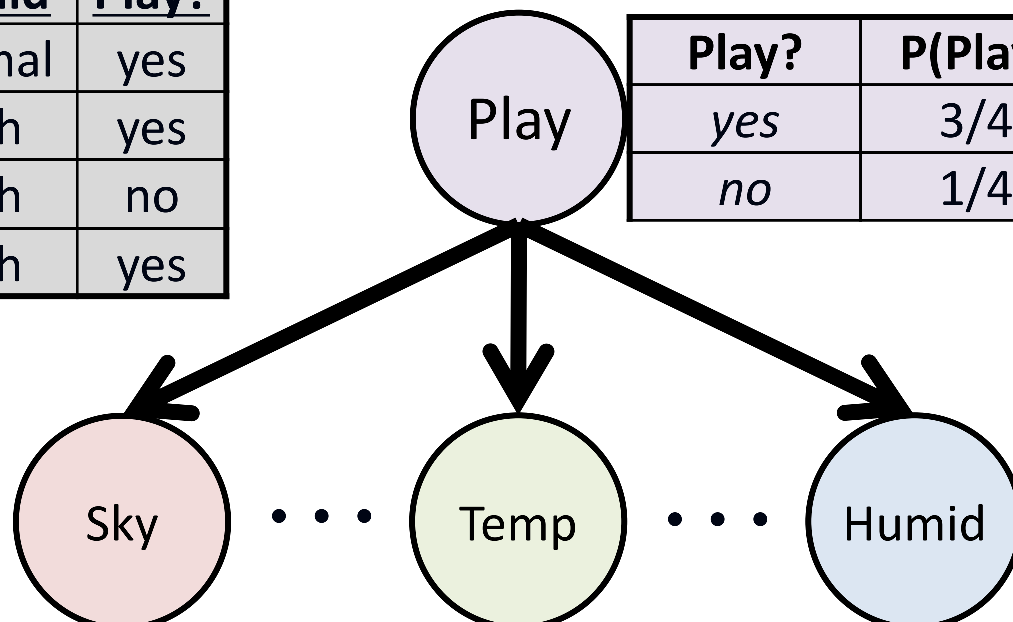
Temp	**Play?**	**P(Temp | Play)**
warm	*yes*	4/5
cold	*yes*	1/5
warm	*no*	1/3
cold	*no*	2/3

Humid	**Play?**	**P(Humid | Play)**
high	*yes*	
norm	*yes*	
high	*no*	
norm	*no*	

Example NB Graphical Model

Data:

Sky	**Temp**	**Humid**	**Play?**
sunny	warm	normal	yes
sunny	warm	high	yes
rainy	cold	high	no
sunny	warm	high	yes



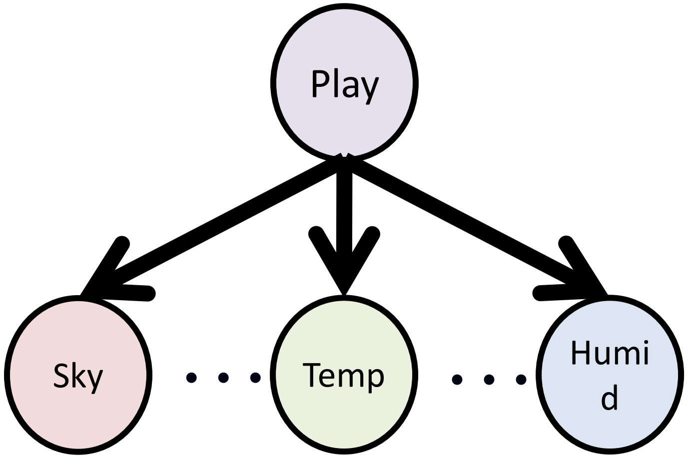
Play?	**P(Play)**
yes	3/4
no	1/4

Sky	**Play?**	**P(Sky | Play)**
sunny	*yes*	4/5
rainy	*yes*	1/5
sunny	*no*	1/3
rainy	*no*	2/3

Temp	**Play?**	**P(Temp | Play)**
warm	*yes*	4/5
cold	*yes*	1/5
warm	*no*	1/3
cold	*no*	2/3

Humid	**Play?**	**P(Humid | Play)**
high	*yes*	3/5
norm	*yes*	2/5
high	*no*	2/3
norm	*no*	1/3

Example Using NB for Classification



Play?	P(Play)
yes	3/4
no	1/4

Temp	Play?	P(Temp | Play)
warm	*yes*	4/5
cold	*yes*	1/5
warm	*no*	1/3
cold	*no*	2/3

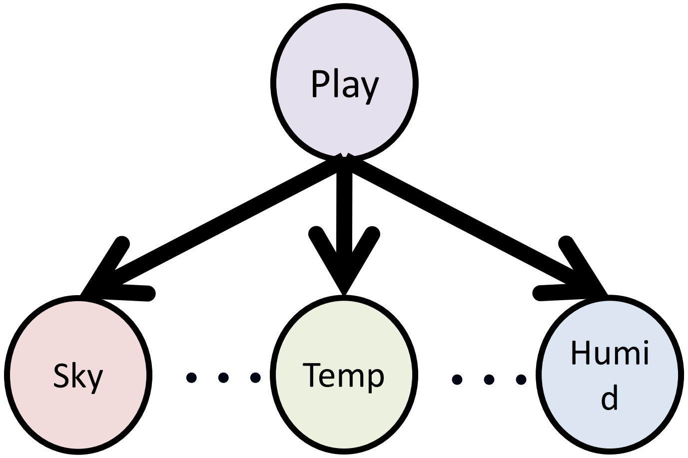
Sky	Play?	P(Sky | Play)
sunny	*yes*	4/5
rainy	*yes*	1/5
sunny	*no*	1/3
rainy	*no*	2/3

Humid	Play?	P(Humid | Play)
high	*yes*	3/5
norm	*yes*	2/5
high	*no*	2/3
norm	*no*	1/3

$$h(\mathbf{x}) = \arg\max_{y_k} \; \log P(Y = y_k) + \sum_{j=1}^{d} \log P(X_j = x_j \mid Y = y_k)$$

Goal: Predict label for **x** = (rainy, warm, normal)

Example Using NB for Classification



Play?	P(Play)
yes	3/4
no	1/4

Temp	Play?	P(Temp | Play)
warm	*yes*	4/5
cold	*yes*	1/5
warm	*no*	1/3
cold	*no*	2/3

Sky	Play?	P(Sky | Play)
sunny	*yes*	4/5
rainy	*yes*	1/5
sunny	*no*	1/3
rainy	*no*	2/3

Humid	Play?	P(Humid | Play)
high	*yes*	3/5
norm	*yes*	2/5
high	*no*	2/3
norm	*no*	1/3

Predict label for:

x = (rainy, warm, normal)

$$P(\text{play} \mid \mathbf{x}) \propto \log P(\text{play}) + \log P(\text{rainy} \mid \text{play}) + \log P(\text{warm} \mid \text{play}) + \log P(\text{normal} \mid \text{play})$$

$$\propto \log 3/4 + \log 1/5 + \log 4/5 + \log 2/5 = -1.319$$

predict PLAY

$$P(\neg\text{play} \mid \mathbf{x}) \propto \log P(\neg\text{play}) + \log P(\text{rainy} \mid \neg\text{play}) + \log P(\text{warm} \mid \neg\text{play}) + \log P(\text{normal} \mid \neg\text{play})$$

$$\propto \log 1/4 + \log 2/3 + \log 1/3 + \log 1/3 = -1.732$$

Naïve Bayes Summary

Advantages:

- Fast to train (single scan through data)
- Fast to classify
- Not sensitive to irrelevant features
- Can handle real-valued data (e.g. fit Gaussian for each attribute)
- Handles streaming data well

Disadvantages:

- Assumes independence of features