CS 760: Machine Learning
Probability & Graphical Models

Fred Sala

University of Wisconsin-Madison

Nov. 4, 2021

Announcements

- **Logistics**:
  - HW 5 has been released; due Tuesday

- Class roadmap:

Thursday, Nov. 4	Graphical Models I
Tuesday, Nov. 9	Graphical Models II
Thursday, Nov. 11	Less-than-full Supervision
Tuesday, Nov. 16	Unsupervised Learning I

Outline

- **Review, SVMs, Kernels**
  - Duality, feature maps, kernel trick
- **Probability Tutorial**
  - Basics, joint probability, conditional probabilities, etc
- **Bayesian Networks**
  - Definition, examples, inference

Outline

- **Review, SVMs, Kernels**
  - Duality, feature maps, kernel trick
- **Probability Tutorial**
  - Basics, joint probability, conditional probabilities, etc
- **Bayesian Networks**
  - Definition, examples, inference

Review: Constrained Optimization & Duality

$$\min_{w} \; f(w)$$

Objective

$$g_i(w) \leq 0, \forall 1 \leq i \leq k$$

Constraints

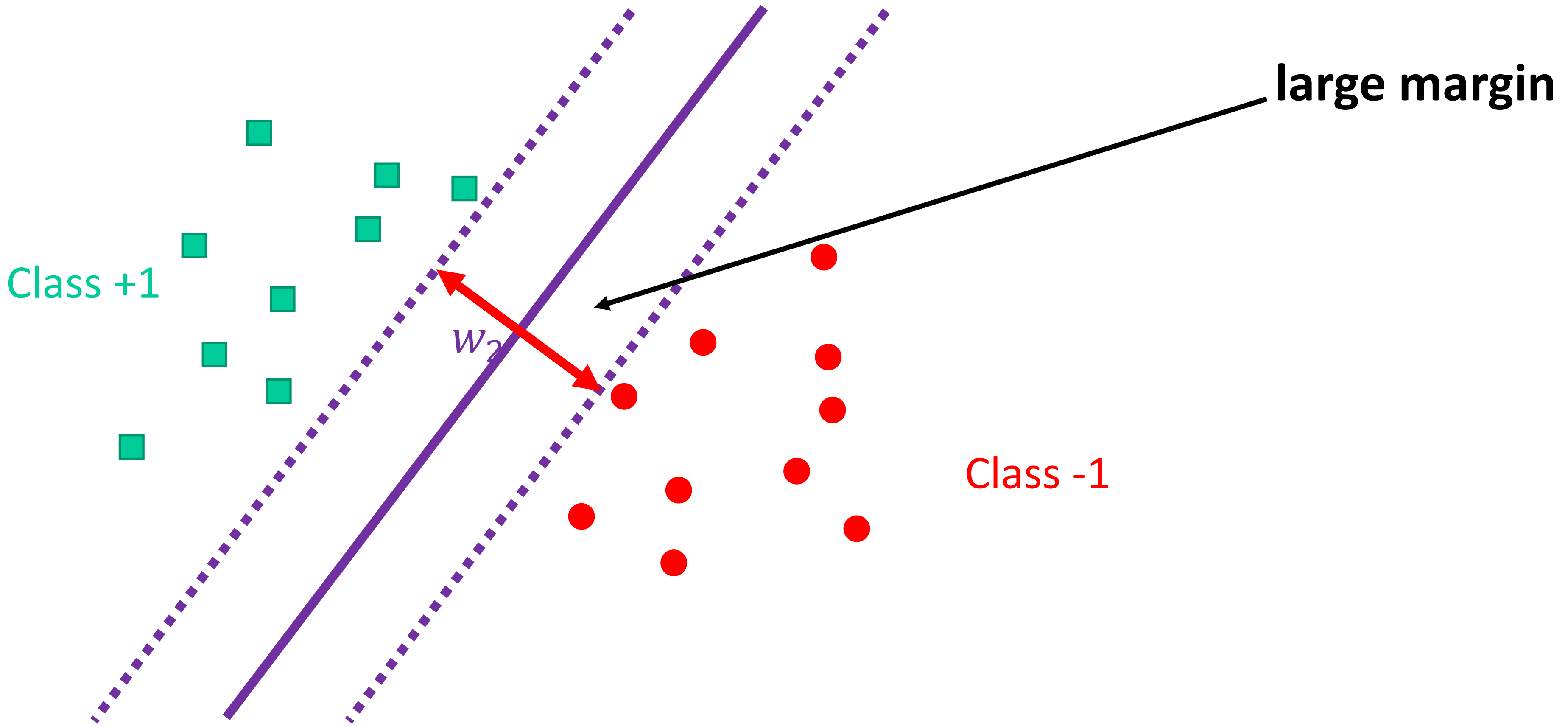
$$h_j(w) = 0, \forall 1 \leq j \leq l$$

- **Lagrangian**: $\mathcal{L}(w, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \; f(w) + \sum_i \alpha_i g_i(w) + \sum_j \beta_j h_j(w)$

- Primal problem $p^* \coloneqq \min_{w} f(w) = \min_{w} \max_{\boldsymbol{\alpha}, \boldsymbol{\beta} : \alpha_i \geq 0} \mathcal{L}(w, \boldsymbol{\alpha}, \boldsymbol{\beta})$
- Dual problem $d^* \coloneqq \max_{\boldsymbol{\alpha}, \boldsymbol{\beta} : \alpha_i \geq 0} \min_{w} \mathcal{L}(w, \boldsymbol{\alpha}, \boldsymbol{\beta})$
- Always true: $d^* \leq p^*$

Review: Apply to Training Linear Classifier

- Want: a **large margin**



Review: Support Vector Machines Goal

Define the margin to be

$$\gamma = \min_i \frac{y_i f_{w,b}(x_i)}{||w||}$$

We proved this

- If $f_{w,b}$ incorrect on some x_i, the margin is **negative**
- Fix scale: $y_{i^*}(w^T x_{i^*} + b) = 1$. Then, margin overall is $\frac{1}{||w||}$

Primal problem:

$$\min_{w,b} \frac{1}{2} ||w||^2$$

Objective: Large margin

$$y_i(w^T x_i + b) \geq 1, \forall i$$

Constraints: Correct on training data

SVM: Dual Version

- Reduces to dual problem:

$$\max_{\boldsymbol{\alpha}} \mathcal{L}(w, b, \boldsymbol{\alpha}) = \max_{\boldsymbol{\alpha}} \sum_i \alpha_i - \frac{1}{2} \sum_{ij} \alpha_i \alpha_j y_i y_j x_i^T x_j$$

$$\sum_i \alpha_i y_i = 0, \alpha_i \geq 0$$

Note: only variables are primal

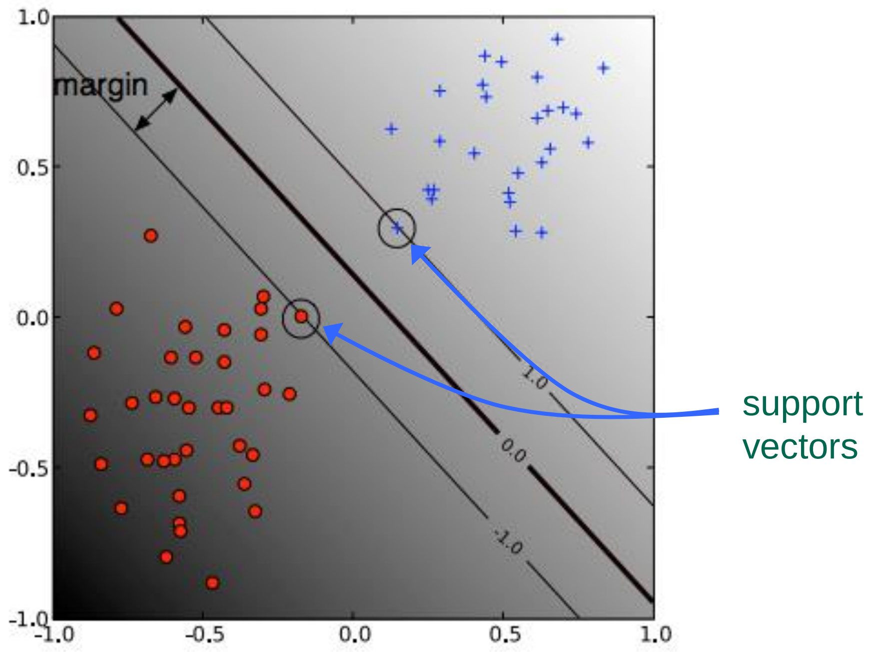
- Since $w = \sum_i \alpha_i y_i x_i$, we have $w^T x + b = \sum_i \alpha_i y_i x_i^T x + b$
- Note: only deals with data via **inner products** $x_i^T x_j$

SVM: Support Vectors

- Solution is a sparse linear combination of training instances

$$w = \sum_i \alpha_i y_i x_i$$

- Those instances with $\alpha_i > 0$ are called ***support vectors***
  - Lie on the margin boundary
- Solution does not change if we delete instances with $\alpha_i = 0$



SVM: Soft Margin

What if our data isn't linearly separable?

- Can adjust our approach by using *slack variables* (denoted by ζ_i) to tolerate errors

$$\min_{w,b,\zeta_i} \frac{1}{2}||w||^2 + C\sum_i \zeta_i$$

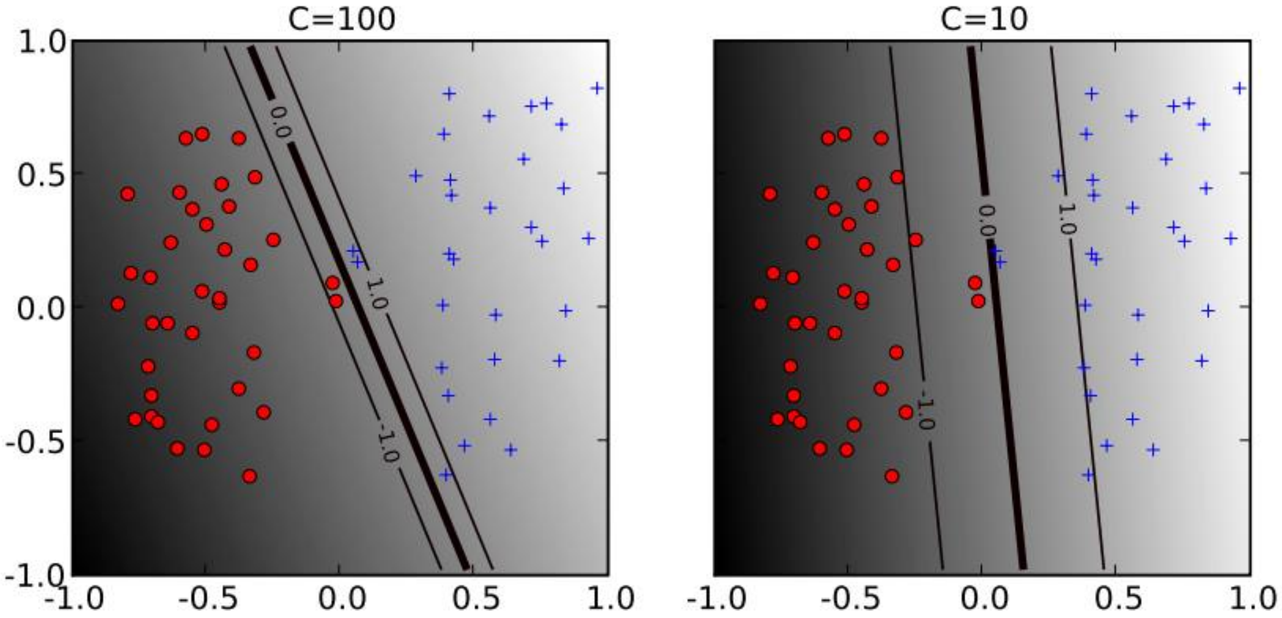
$$y_i(w^T x_i + b) \geq 1 - \zeta_i, \zeta_i \geq 0, \forall i$$

- C determines the relative importance of maximizing margin vs. minimizing slack

SVM: Soft Margin

$$\min_{w,b,\zeta_i} \frac{1}{2}||w||^2 + C\sum_i \zeta_i$$

$$y_i(w^T x_i + b) \geq 1 - \zeta_i, \zeta_i \geq 0, \forall i$$



Ben-Hur & Weston, *Methods in Molecular Biology* 2010

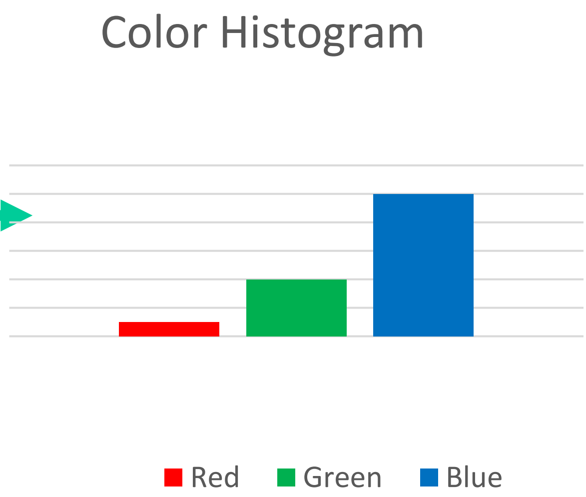
Feature Maps

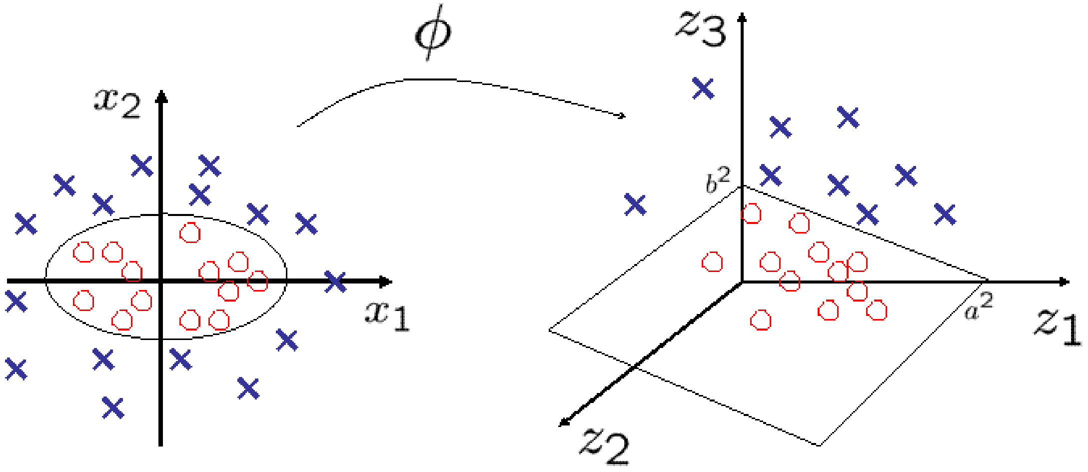
- Can take a set of features and map them into another
  - Can also construct non-linear features
  - Use these inside a linear classifier?

x

Extract features

$\phi(x)$





$$\phi : (x_1, x_2) \longrightarrow (x_1^2, \sqrt{2}x_1x_2, x_2^2)$$

$$\left(\frac{x_1}{a}\right)^2 + \left(\frac{x_2}{b}\right)^2 = 1 \longrightarrow \frac{z_1}{a^2} + \frac{z_3}{b^2} = 1$$

Feature Maps and SVMs

Want to use feature space $\{\phi(x_i)\}$ in linear classifier…

- Downside: dimension might be high (even infinite!)
- So we don't want to write down $\phi(x_i) = [0.2, 0.3, \dots]$

Recall our SVM dual form:

- Only relies on inner products $x_i^T x_j$

$$\mathcal{L}(w, b, \boldsymbol{\alpha}) = \sum_i \alpha_i - \frac{1}{2} \sum_{ij} \alpha_i \alpha_j y_i y_j x_i^T x_j$$

$$\sum_i \alpha_i y_i = 0, \alpha_i \geq 0$$

Kernel Trick

- Using SVM on the feature space $\{\phi(x_i)\}$: only need $\phi(x_i)^T \phi(x_j)$

- Conclusion: no need to design $\phi(\cdot)$, only need to design

$$k(x_i, x_j) = \phi(x_i)^T \phi(x_j)$$

Kernel Matrix

Feature Maps

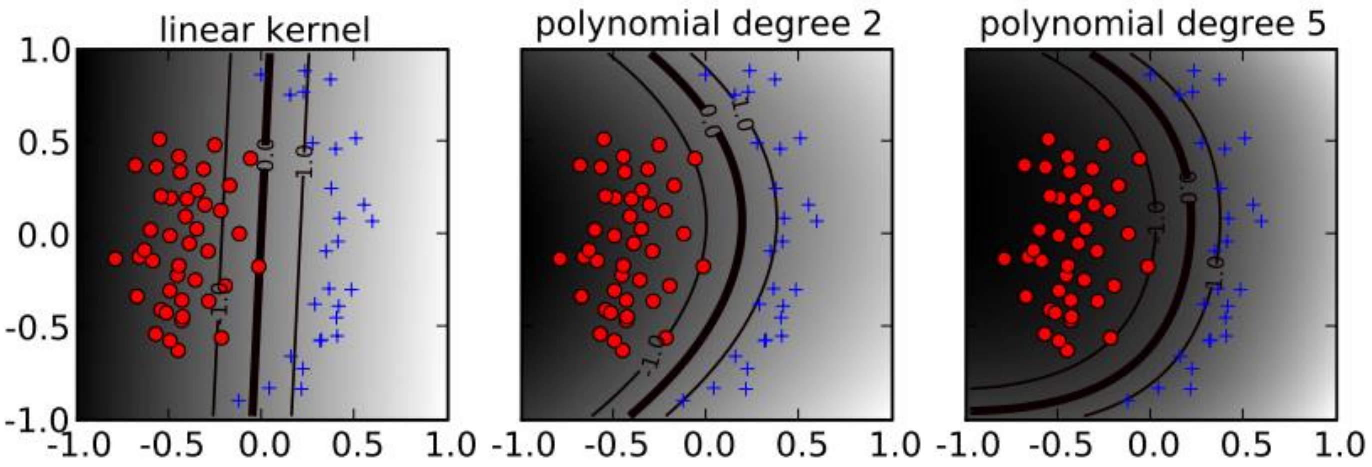
Kernel Types: Polynomial

- Fix degree d and constant c:

$$k(x, x') = (x^T x' + c)^d$$

- What are $\phi(x)$?
- Expand the expression to get $\phi(x)$

$$\forall \mathbf{x}, \mathbf{x}' \in \mathbb{R}^2, \quad K(\mathbf{x}, \mathbf{x}') = (x_1 x_1' + x_2 x_2' + c)^2 = \begin{bmatrix} x_1^2 \\ x_2^2 \\ \sqrt{2}\, x_1 x_2 \\ \sqrt{2c}\, x_1 \\ \sqrt{2c}\, x_2 \\ c \end{bmatrix} \cdot \begin{bmatrix} {x'}_1^2 \\ {x'}_2^2 \\ \sqrt{2}\, x_1' x_2' \\ \sqrt{2c}\, x_1' \\ \sqrt{2c}\, x_2' \\ c \end{bmatrix}$$



Ben-Hur & Weston, *Methods in Molecular Biology* 2010

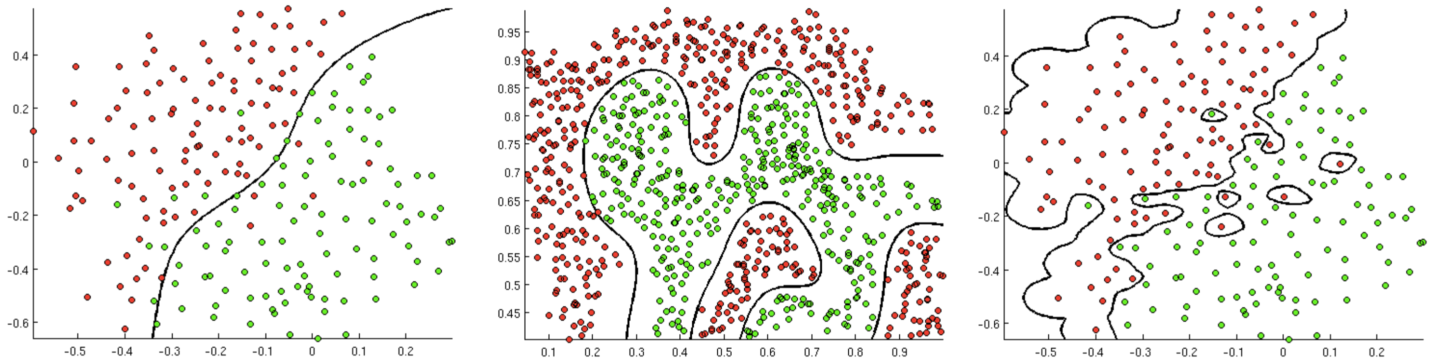
Kernel Types: Gaussian/RBF

- Fix bandwidth σ:

$$k(x, x') = \exp(-\left\|x - x'\right\|^2 / 2\sigma^2)$$

  - Also called radial basis function (RBF) kernels

$\gamma = 10$ $\gamma = 100$ $\gamma = 1000$



$$k(x, x') = \exp(-\gamma \left\|x - x'\right\|^2)$$

Theory of Kernels

- Part of a deep mathematical theory
- With some conditions, any kernel yields a feature map:
  - Theorem: $k(x, x')$ has expansion

$$k(x, x') = \sum_{i}^{+\infty} a_i \phi_i(x) \phi_i(x')$$

Feature Maps

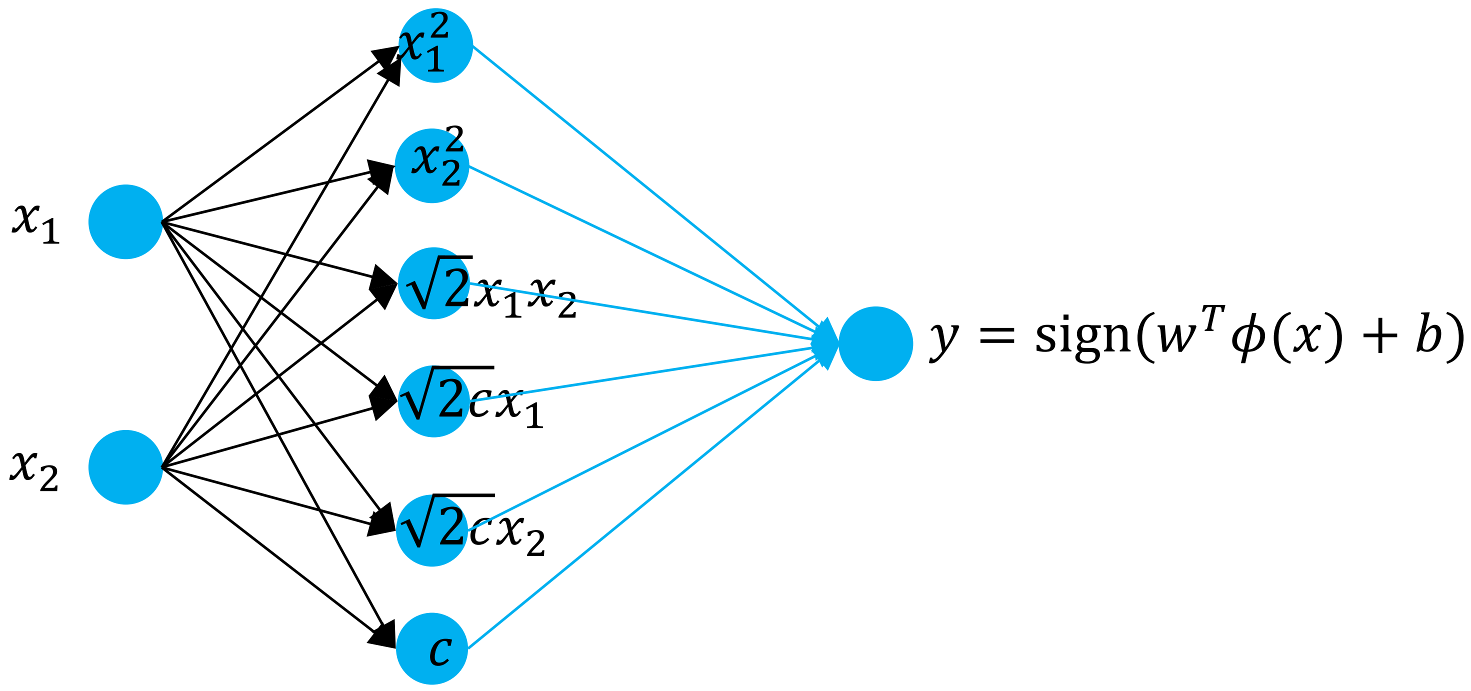
for nonnegative a_i's, if and only if for any function $c(x)$,

$$\int\int c(x)c(x')k(x, x')dxdx' \geq 0$$

- Given certain requirements/conditions, can construct a bunch of new kernels from existing ones

Kernel Methods VS Neural Networks

- Can think of our kernel SVM approach as fixing a layer of a neural network



SVM Review

- Can find globally optimal solutions: convex optimization
  - No local minima (unlike training general NNs)
- Can train primal or dual
  - Dual: relies on **support vectors**; enables use of **kernels**
- Variety of pre-existing optimization techniques
- Kernels: allow non-linear decision boundaries
  - And to represent all sorts of new data (strings, trees)
  - High-dimensional representations, but can use kernel trick to avoid explicitly computing feature maps
  - Good performance! Sometimes close to DNNs

Break & Quiz

Outline

- **Review, SVMs, Kernels**
  - Duality, feature maps, kernel trick
- **Probability Tutorial**
  - Basics, joint probability, conditional probabilities, etc
- **Bayesian Networks**
  - Definition, examples, inference

Probability Tutorial: Outcomes & Events

- Outcomes: possible results of an **experiment**
- **Events:** subsets of outcomes we're interested in

Ex:

$$\Omega = \underbrace{\{1, 2, 3, 4, 5, 6\}}_{\text{outcomes}}$$

$$\mathcal{F} = \underbrace{\{\emptyset, \{1\}, \{2\}, \ldots, \{1, 2\}, \ldots, \Omega\}}_{\text{events}}$$

Probability Tutorial: Outcomes & Events

- Event space can be smaller:

$$\mathcal{F} = \underbrace{\{\emptyset, \{1,3,5\}, \{2,4,6\}, \Omega\}}_{\text{events}}$$

- Two components always in it!

$$\emptyset, \Omega$$

Probability Tutorial: Sigma Fields

- F is a "sigma algebra".
  - Follows certain rules:
    - Everything in it (saw this already)
    - If A is in F, so is A^c
    - Closed under countable unions

Probability Tutorial: Probability Spaces

- Now we need a way to produce probabilities of events, so introduce a function

$$P : \mathcal{F} \to [0, 1]$$

- Has certain properties, which we'll see in a second.
- Overall, we get a probability space

$$(\Omega, \mathcal{F}, P)$$

Probability Tutorial: Probability Sapces

- We have outcomes and events and probabilities
- I.e.,

$$\text{For } E \in \mathcal{F},\ P(E) \in [0,1]$$

Back to our example:

$$\mathcal{F} = \underbrace{\{\emptyset, \{1,3,5\}, \{2,4,6\}, \Omega\}}_{\text{events}}$$

$$P(\{1,3,5\}) = 0.2, P(\{2,4,6\}) = 0.8$$

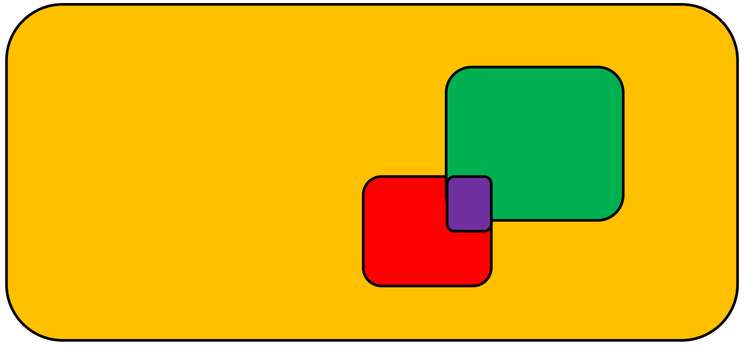
Basics: **Axioms**

- Rules for probability:
  - For all events $E \in \mathcal{F}, P(E) \geq 0$
  - Always, $P(\emptyset) = 0, P(\Omega) = 1$
  - For disjoint events,

$$P(E_1 \cup E_2) = P(E_1) + P(E_2)$$

- Easy to derive other laws. Ex: non-disjoint events

$$P(E_1 \cup E_2) = P(E_1) + P(E_2) - P(E_1 \cap E_2)$$

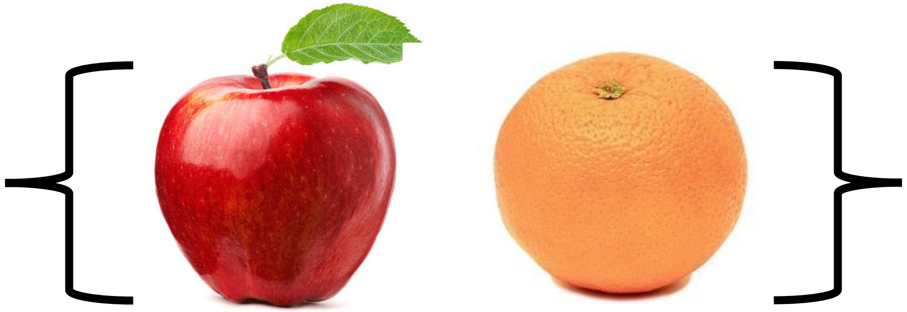


Basics: **Random Variables**

- Really, functions
- Map outcomes to real values

$$X : \Omega \to \mathbb{R}$$

- Why?
  - So far, everything is a set.
  - Hard to work with!
  - Real values are easy to work with



- One requirement, "F measurable". For any *c*,

$$\{\omega : X(\omega) \leq c\} \in \mathcal{F}$$

Basics: **CDF & PDF**

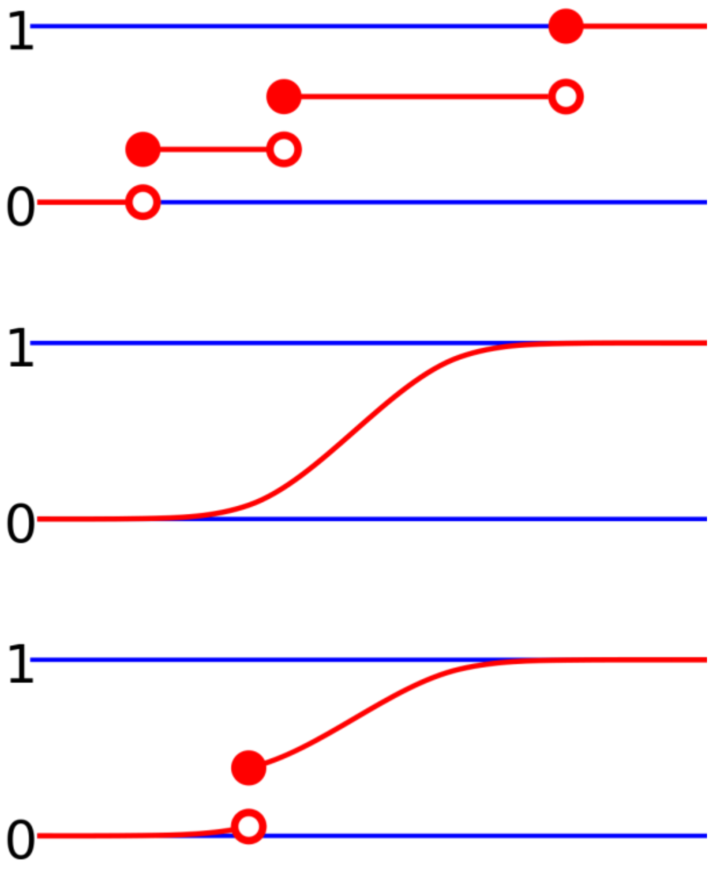
- Can still work with probabilities:

$$P(X = 3) := P(\{\omega : X(\omega) = 3\})$$

- Cumulative Distribution Func. (CDF)

$$F_X(x) := P(X \leq x)$$

- Density / mass function $\quad p_X(x)$
  - Doesn't always exist!



Wiki CDF

Basics: **Expectation** & **Variance**

- Another advantage of RVs are ``summaries''
- Expectation:
  - The "average" $E[X] = \sum_a a \times P(x = a)$
- Variance: $Var[X] = E[(X - E[X])^2]$
  - A measure of spread
- Raw moments: $E[X], E[X^2], E[X^3], \ldots$
- Note: also don't always exist…
  - **Ex**: Cauchy distribution

Basics: **Expectation** Properties

- Expectation has very useful properties…
  - Linearity: $$E[\sum_i a_i X_i] = \sum_i a_i E[X_i]$$
  - Independence not required!
  - Hat check problem:
    - There is a dinner party where n people check their hats. The hats are mixed up during dinner, so that afterward each man receives a random hat. In particular, each person gets their own hat with probability 1/n. What is the expected number of people who get their own hat?

Basics: **Joint Distributions**

- Move from one variable to several
- Joint distribution

$$P(X = a, Y = b)$$

- Or more variables.

$$P(X_1 = x_1, X_2 = x_2, \ldots, X_k = x_k)$$

Basics: **Marginal** Probability

- Given a joint distribution

$$P(X = a, Y = b)$$

  - Get the distribution in just one variable:

$$P(X = a) = \sum_b P(X = a, Y = b)$$

  - This is the "marginal" distribution.

Basics: **Marginal** Probability

$$P(X=a)=\sum_b P(X=a,Y=b)$$

	Sunny	Cloudy	Rainy
hot	150/365	40/365	5/365
cold	50/365	60/365	60/365

$$[P(\mathrm{hot}),P(\mathrm{cold})]=[\tfrac{195}{365},\tfrac{170}{365}]$$

Independence

- Independence for a set of events $A_1, \dots, A_k$

$$P(A_{i_1} A_{i_2} \cdots A_{i_j}) = P(A_{i_1}) P(A_{i_2}) \cdots P(A_{i_j})$$

for all the $i_1, \dots, i_j$ combinations

- Why useful? Dramatically reduces the complexity
- Collapses joint into **product** of marginals
  - Note sometimes we have only pair-wise, etc independence

Uncorrelatedness

- For random variables, uncorrelated means

$$E[XY] = E[X]E[Y]$$

Note: weaker than independence.

- Independence implies uncorrelated (easy to see)
- Other way around: usually false (but not always).
- If X,Y independent, functions are not correlated:

$$E[f(X)f(Y)] = E[f(X)]E[f(Y)]$$

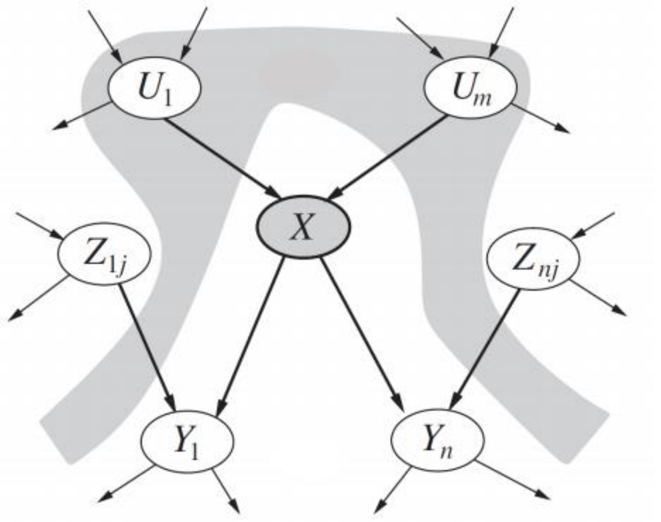
Conditional Probability

- For when we know something,

$$P(X = a|Y = b) = \frac{P(X = a, Y = b)}{P(Y = b)}$$

- Leads to **conditional independence**

$$P(X, Y|Z) = P(X|Z)P(Y|Z)$$



Credit: Devin Soni

Chain Rule

- Apply repeatedly,

$$P(A_1, A_2, \ldots, A_n)$$
$$= P(A_1)P(A_2|A_1)P(A_3|A_2, A_1)\ldots P(A_n|A_{n-1}, \ldots, A_1)$$

- Note: still big!
  - If some **conditional independence**, can factor!
  - Leads to **probabilistic graphical models (this lecture)**

Law of Total Probability

- Partition the sample space into disjoint $B_1, \ldots, B_k$
- Then,

$$P(A) = \sum_i P(A|B_i)P(B_i)$$

- Useful way to control A via conditional probabilities.
  - **Example**: there are 5 red and 2 green balls in an urn. A random ball is selected and replaced by a ball of the other color; then a second ball is drawn. What is the probability the second ball is red?

Review: Bayesian Inference

- Conditional Prob. & Bayes:

$$P(H|E_1, E_2, \ldots, E_n) = \frac{P(E_1, \ldots, E_n|H)P(H)}{P(E_1, E_2, \ldots, E_n)}$$

- Has more evidence.
  - Likelihood is hard---but **conditional independence assumption**

$$P(H|E_1, E_2, \ldots, E_n) = \frac{P(E_1|H)P(E_2|H)\cdots, P(E_n|H)P(H)}{P(E_1, E_2, \ldots, E_n)}$$

Random Vectors & Covariance

- Recall variance: $\mathbb{E}[(X - E[X])^2]$
- Now, for a **random vector** (same as joint of *d* RVs)
  - Note: size *d x d*. All variables are centered

$$\Sigma = \begin{bmatrix} \mathbb{E}[(X_1 - \mathbb{E}[X_1])^2] & \dots & [(X_1 - \mathbb{E}[X_1])((X_n - \mathbb{E}[X_n])] \\ \vdots & \vdots & \vdots \\ [(X_n - \mathbb{E}[X_n])((X_1 - \mathbb{E}[X_1])] & \dots & \mathbb{E}[(X_n - \mathbb{E}[X_n])^2] \end{bmatrix}$$

Cross-variance

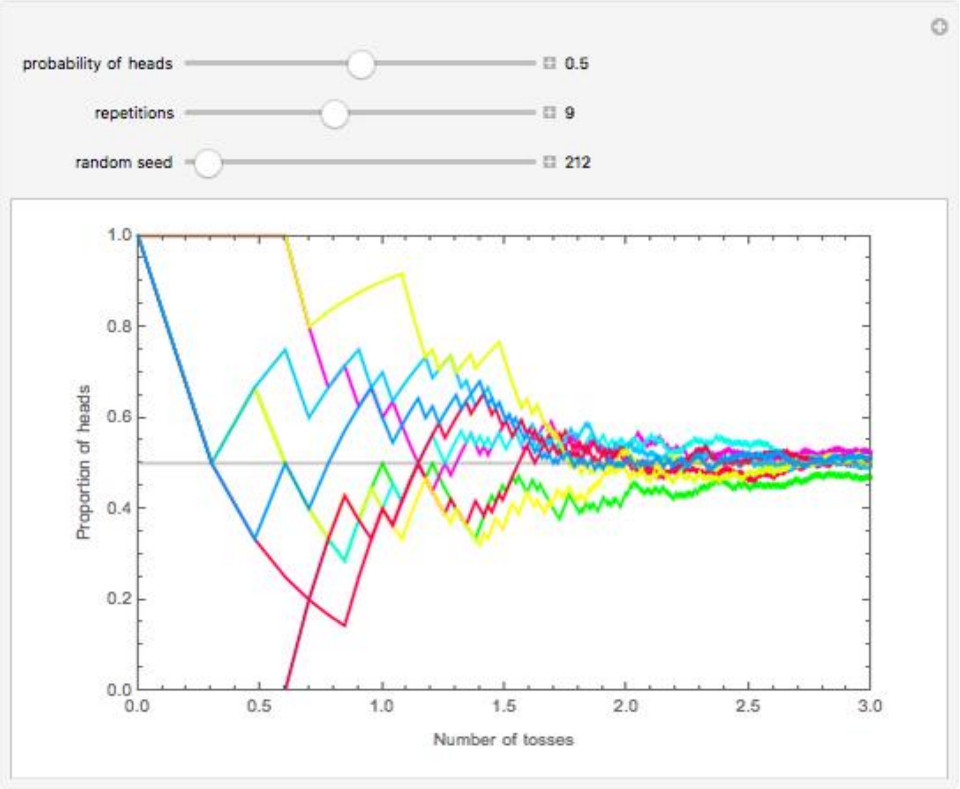
Diagonals: Scalar Variance

Estimation Theory

- How do we know that the sample mean is a good estimate of the true mean?
  - Concentration inequalities

$$P(|\mathbb{E}[X] - \hat{\mathbb{E}}[X]| \geq t) \leq \exp(-2nt^2)$$

  - Law of large numbers
  - Central limit theorems, etc.



Wolfram Demo

Break & Quiz

Outline

- **Review, SVMs, Kernels**
  - Duality, feature maps, kernel trick
- **Probability Tutorial**
  - Basics, joint probability, conditional probabilities, etc
- **Bayesian Networks**
  - Definition, examples, inference

Bayesian Networks Example

- Consider the following 5 binary random variables:

  B = a burglary occurs at the house

  E = an earthquake occurs at the house

  A = the alarm goes off

  J = John calls to report the alarm

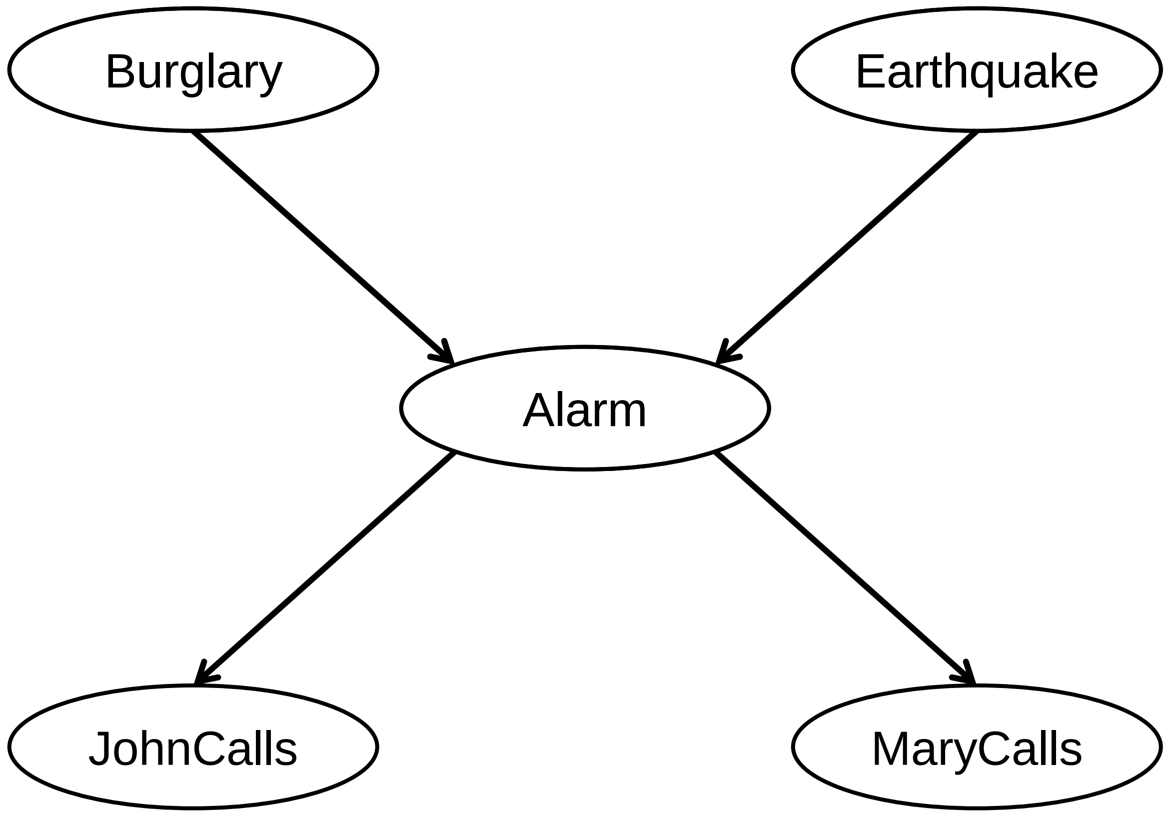
  M = Mary calls to report the alarm

- Suppose Burglary or Earthquake can trigger Alarm, and Alarm can trigger John’s call or Mary’s call

- Now we want to answer queries like what is $P(B \mid M, J)$?

Bayesian Networks Example

- Set up a network that shows how random variables influence others:

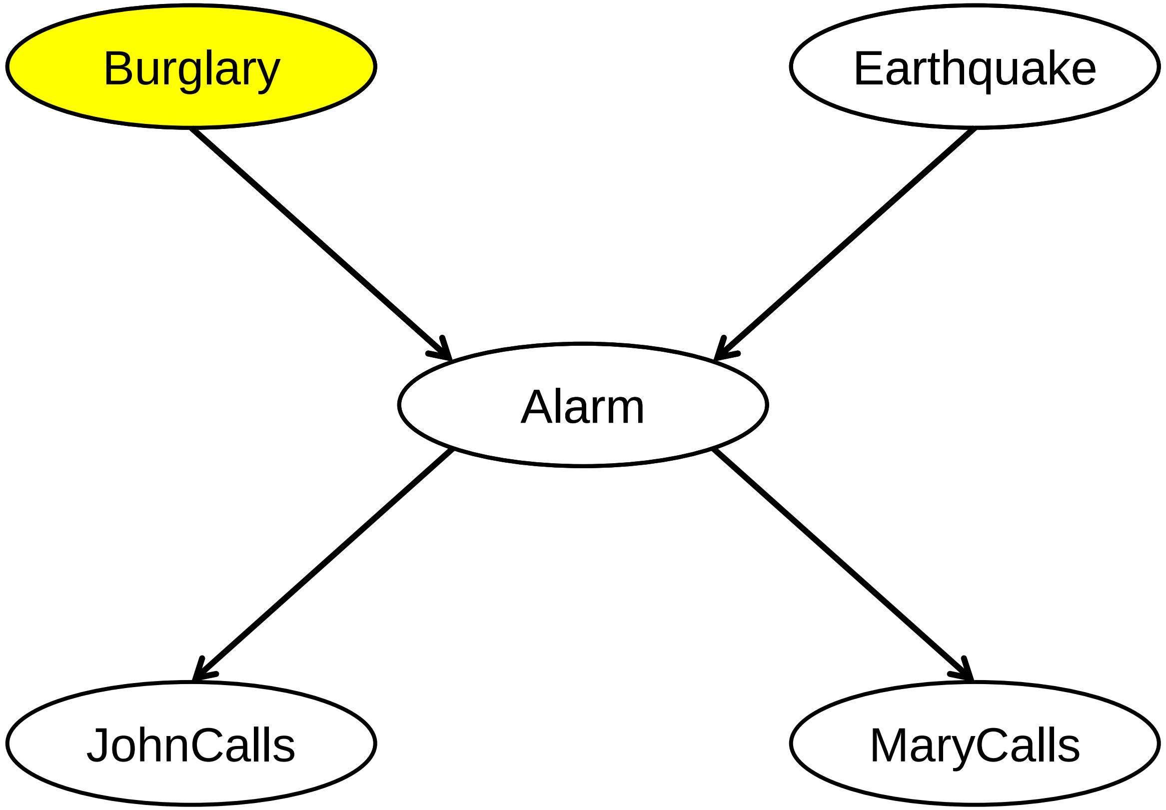


Bayesian Networks Example

- Set up a network that shows how random variables influence others:

P (*B*)

t	f
0.001	0.999

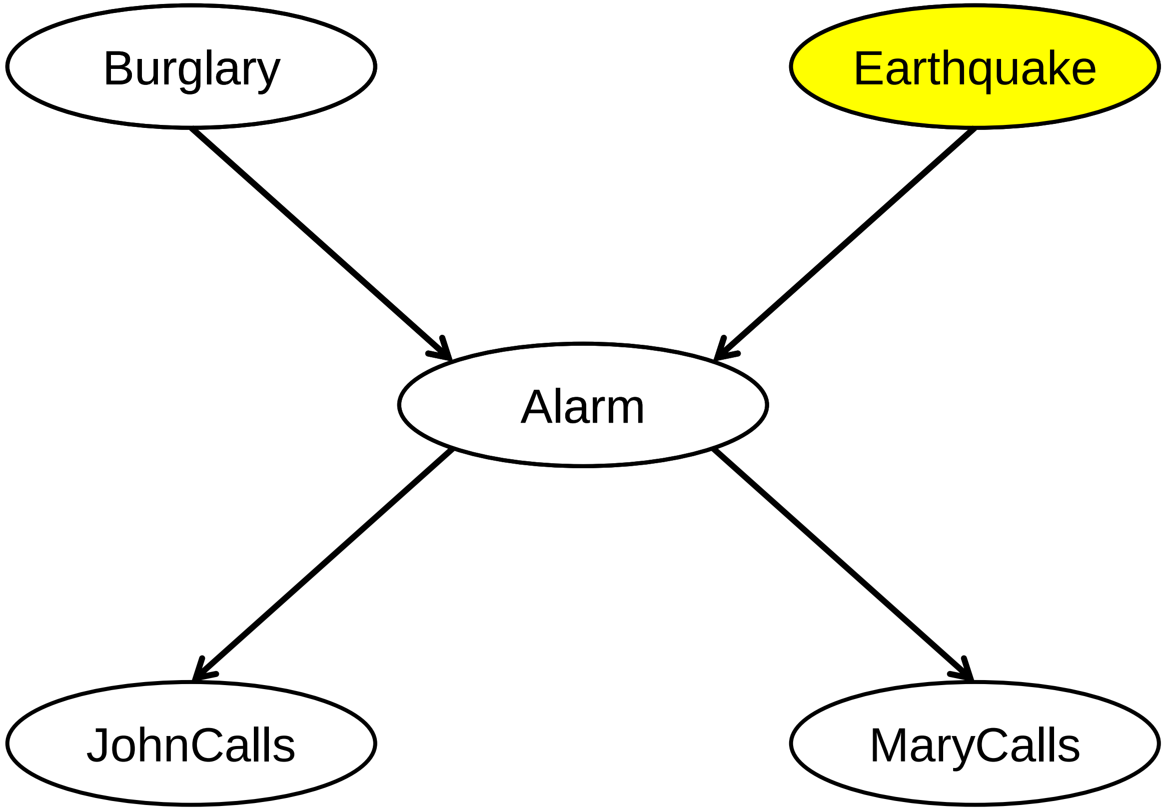


Bayesian Networks Example

- Set up a network that shows how random variables influence others:

P (*B*)

t	f
0.001	0.999



P (*E*)

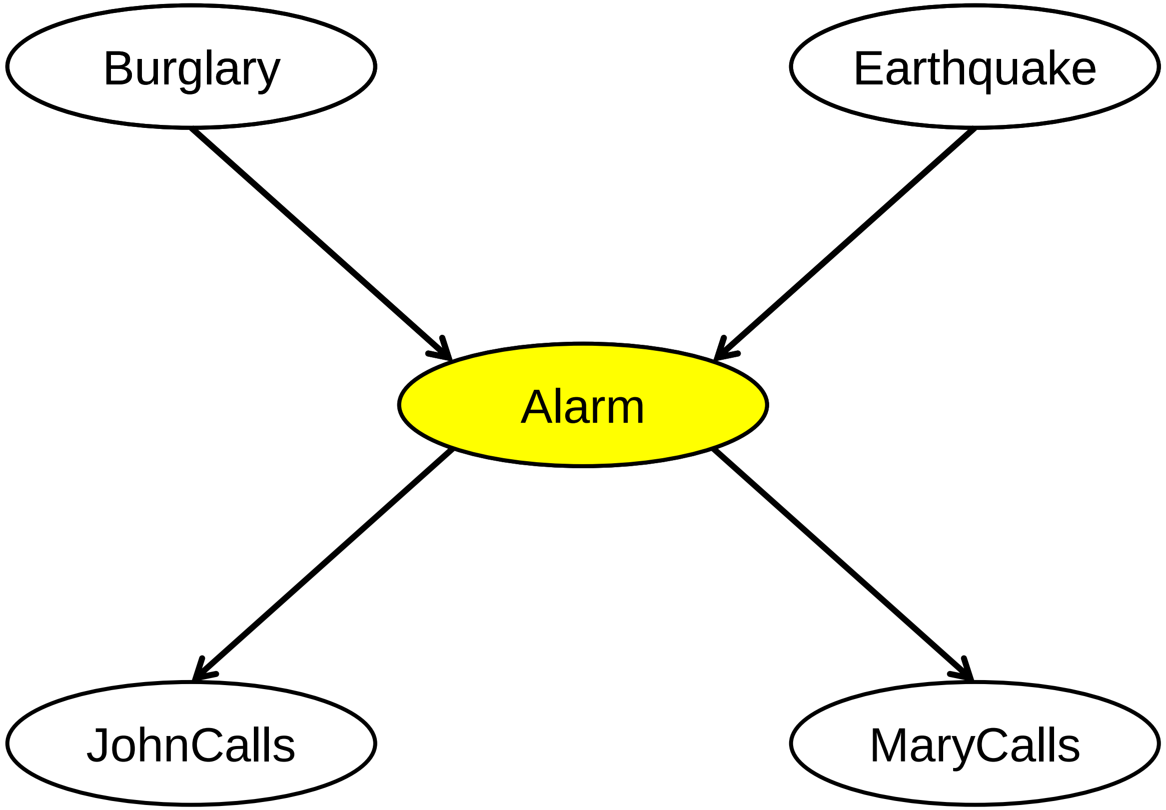
t	f
0.001	0.999

Bayesian Networks Example

- Set up a network that shows how random variables influence others:

P (B)

t	f
0.001	0.999



P (E)

t	f
0.001	0.999

P (A | B, E)

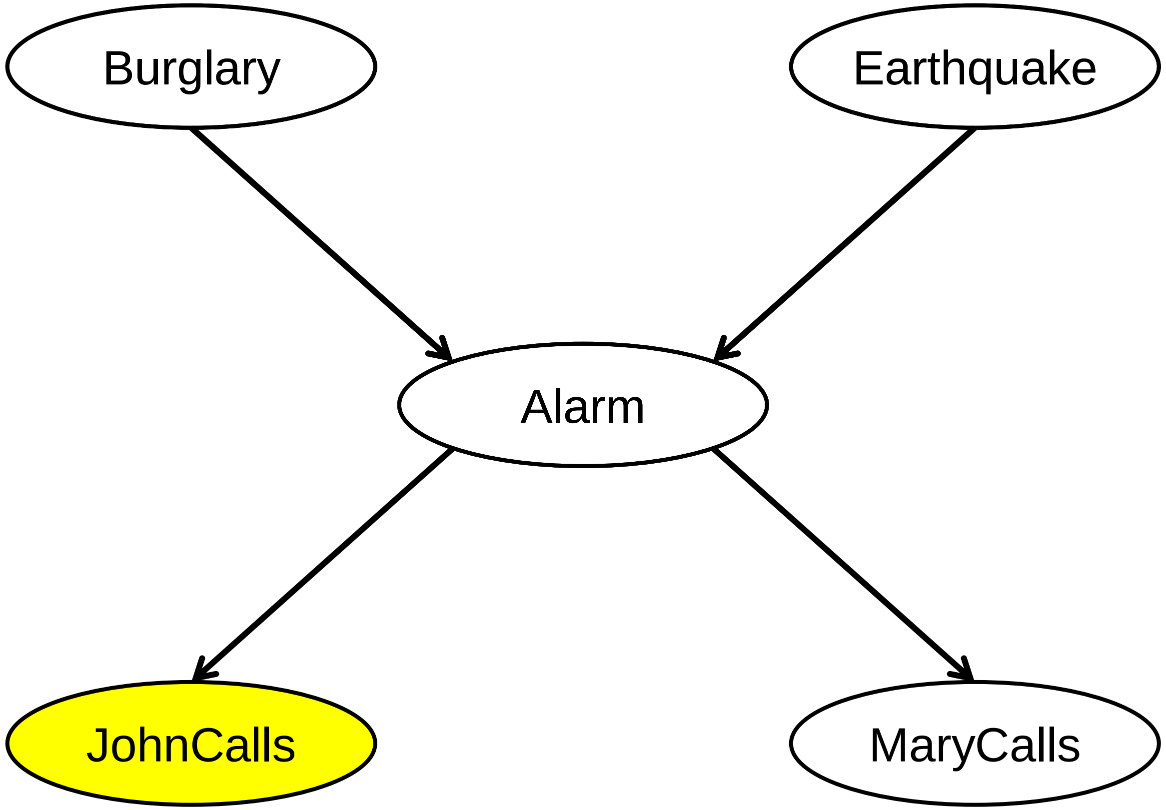
B	*E*	t	f
t	t	0.95	0.05
t	f	0.94	0.06
f	t	0.29	0.71
f	f	0.001	0.999

Bayesian Networks Example

- Set up a network that shows how random variables influence others:

P (*B*)

t	f
0.001	0.999



P (*E*)

t	f
0.001	0.999

P (*A* / *B, E*)

B	*E*	t	f
t	t	0.95	0.05
t	f	0.94	0.06
f	t	0.29	0.71
f	f	0.001	0.999

P (*J* / *A*)

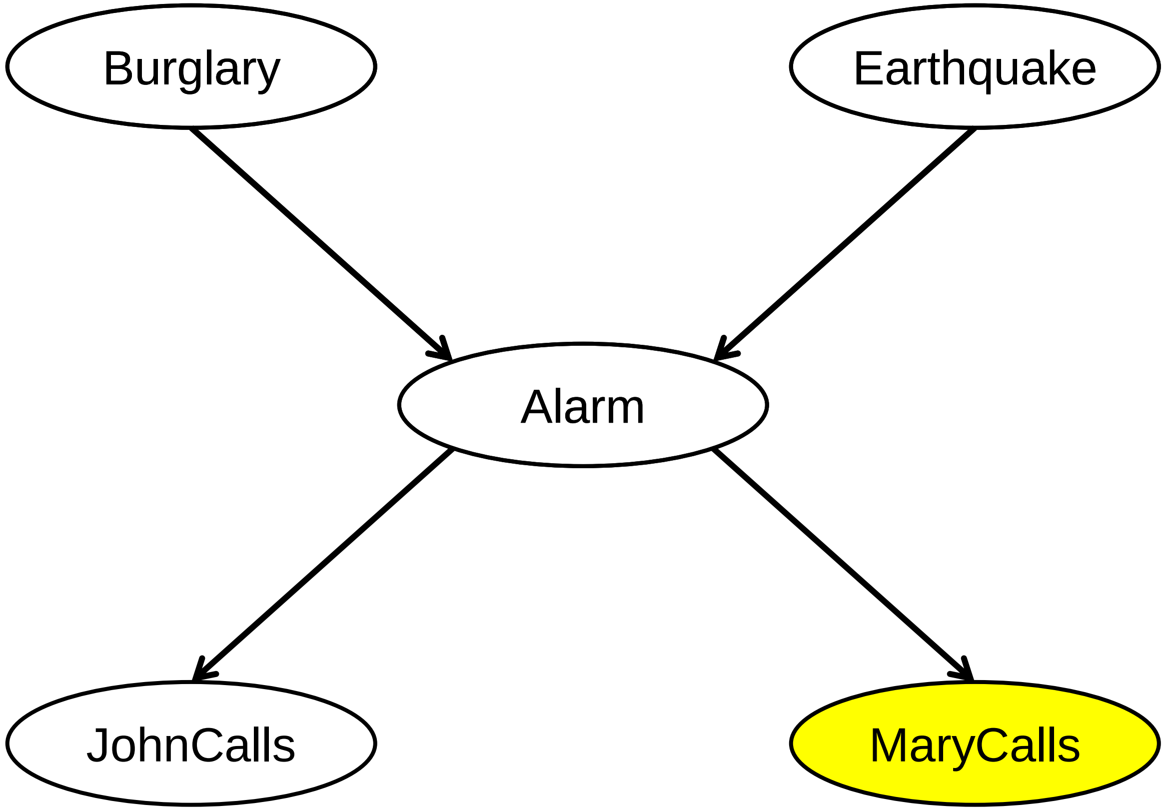
A	t	f
t	0.9	0.1
f	0.05	0.95

Bayesian Networks Example

- Set up a network that shows how random variables influence others:

P (B)

t	f
0.001	0.999



P (E)

t	f
0.001	0.999

P (A | B, E)

B	*E*	t	f
t	t	0.95	0.05
t	f	0.94	0.06
f	t	0.29	0.71
f	f	0.001	0.999

P (J | A)

A	t	f
t	0.9	0.1
f	0.05	0.95

P (M | A)

A	t	f
t	0.7	0.3
f	0.01	0.99

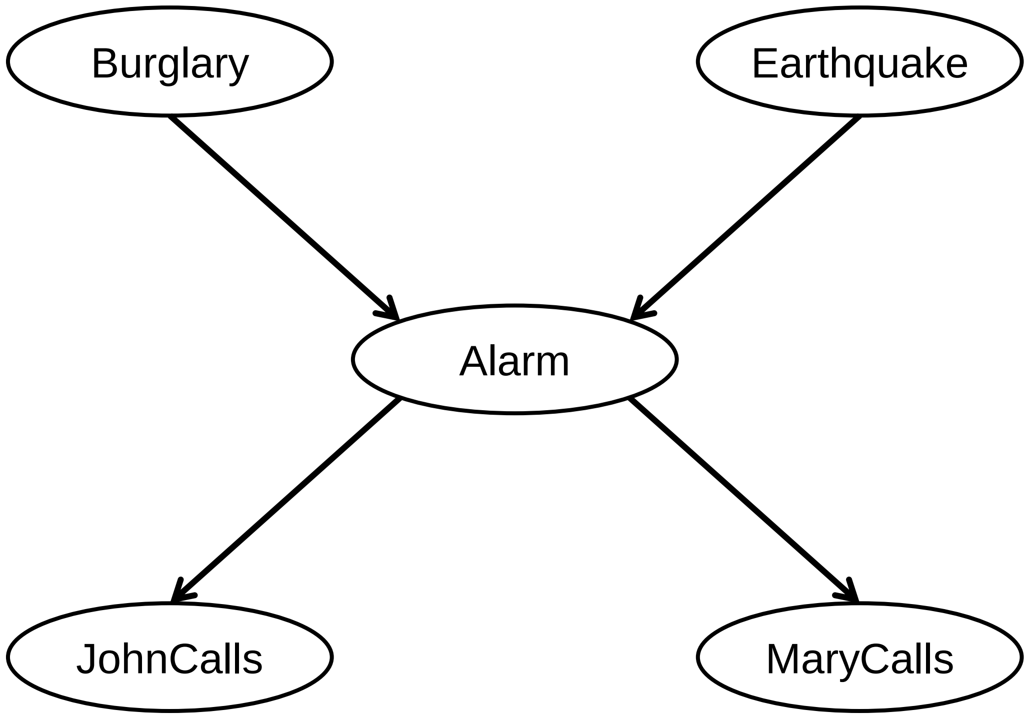
Bayesian Networks: Definition

- A BN consists of a **Directed Acyclic Graph (DAG**) and a set of **conditional probability distributions**
- The DAG:
  - each node denotes a random variable
  - each edge from X to Y represents that X *directly influences* Y
  - (formally: each variable X is independent of its non-descendants given its parents)
  - **Each CPD: represents** $P(X \mid Parents(X))$

$$p(x_1, \ldots, x_d) = \prod_{v \in V} p(x_v | x_{\mathrm{pa}(v)})$$

Bayesian Networks: Parameter Counting

- Parameter reduction: a standard representation of the joint distribution for the Alarm example has $2^5 = 32$ parameters
- the BN representation of this distribution has 20 parameters



$$
\begin{aligned}
&P(B, E, A, J, M) \\
&= P(B) \\
&\times P(E) \\
&\times P(A \mid B, E) \\
&\times P(J \mid A) \\
&\times P(M \mid A)
\end{aligned}
$$

Inference in Bayesian Networks

Given: values for some variables in the network (*evidence*), and a set of ***query*** variables

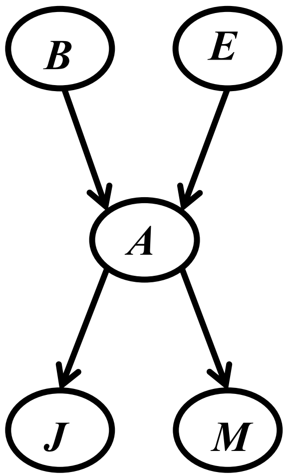
Do: compute the posterior distribution over the query variables

- variables that are neither evidence variables nor query variables are ***hidden*** variables
- the BN representation is flexible enough that any set can be the evidence variables and any set can be the query variables

Inference by Enumeration

- Let a denote A=true, and $\neg a$ denote A=false
- Suppose we're given the query: $P(b \mid j, m)$

  "probability the house is being burglarized given that John and Mary both called"
- From the graph structure we can first compute:



$$P(b, j, m) = \sum_{e,\neg e} \sum_{a,\neg a} P(b)P(E)P(A \mid b, E)P(j \mid A)P(m \mid A)$$

sum over possible values for E and A variables (e, $\neg e$, a, $\neg a$)

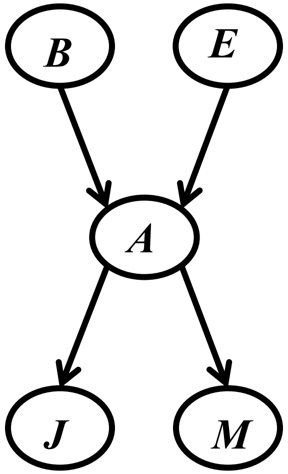
Inference by Enumeration

$$P(b,j,m)=\sum_{e,\neg e}\sum_{a,\neg a}P(b)P(E)P(A\mid b,E)P(j\mid A)P(m\mid A)$$

$$=P(b)\sum_{e,\neg e}\sum_{a,\neg a}P(E)P(A\mid b,E)P(j\mid A)P(m\mid A)$$

P(B)
0.001

P(E)
0.001



B	***E***	***P(A)***
t	**t**	**0.95**
t	**f**	**0.94**
f	**t**	**0.29**
f	**f**	**0.00 1**

A	***P(J)***
t	**0.9**
f	**0.05**

A	***P(M)***
t	**0.7**
f	**0.01**

$B \quad E \quad A \quad J \quad M$

$$\begin{aligned}=0.001\times(&0.001\times0.95\times0.9\times0.7+ && e,\ a\\ &0.001\times0.05\times0.05\times0.01+ && e,\ \neg a\\ &0.999\times0.94\times0.9\times0.7+ && \neg e,\ a\\ &0.999\times0.06\times0.05\times0.01) && \neg\, e,\ \neg\, a\end{aligned}$$

Inference by Enumeration

- Next do equivalent calculation for $P(\neg b,\ j,\ m)$ and determine $P(b \mid j,\ m)$

$$P(b \mid j,m) = \frac{P(b, j,m)}{P(j,m)} = \frac{P(b, j,m)}{P(b, j,m) + P(\neg b, j,m)}$$

So: exact method, but can be intractably hard.

- Some cases: efficient
- Approximate inference sometimes available

Thanks Everyone!

Some of the slides in these lectures have been adapted/borrowed from materials developed by Mark Craven, David Page, Jude Shavlik, Tom Mitchell, Nina Balcan, Elad Hazan, Tom Dietterich, Pedro Domingos, Jerry Zhu, Yingyu Liang, Volodymyr Kuleshov, Fei-Fei Li, Justin Johnson, Serena Yeung, Pieter Abbeel, Peter Chen, Jonathan Ho, Aravind Srinivas