A gentle introduction to the mathematics of biosurveillance: Bayes Rule and Bayes Classifiers

Andrew W. Moore

Professor
The Auton Lab
School of Computer Science
Carnegie Mellon University
http://www.autonlab.org

Associate Member
The RODS Lab
University of Pittburgh
Carnegie Mellon University
http://rods.health.pitt.edu

awm@cs.cmu.edu
412-268-7599

What we're going to do

- We will review the concept of reasoning with uncertainty
- Also known as probability
- This is a fundamental building block
- It's really going to be worth it

What we're going to do

- We will review the concept of reasoning with uncertainty
- Also known as probability
- This is a fundamental building block
- It's really going to be worth it

*(No I mean it… it **really** is going to be worth it!)*

Discrete Random Variables

- A is a Boolean-valued random variable if A denotes an event, and there is some degree of uncertainty as to whether A occurs.
- Examples
  - A = The next patient you examine is suffering from inhalational anthrax
  - A = The next patient you examine has a cough
  - A = There is an active terrorist cell in your city

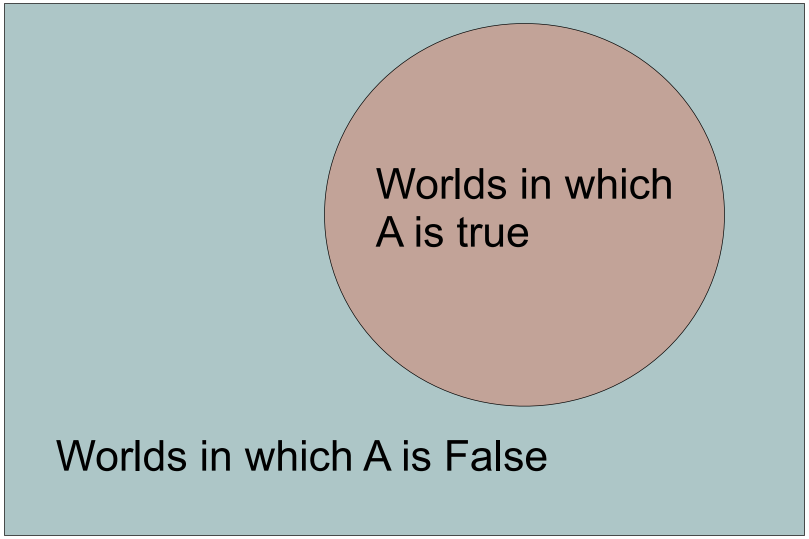
Probabilities

- We write P(A) as “the fraction of possible worlds in which A is true”
- We could at this point spend 2 hours on the philosophy of this.
- But we won’t.

Visualizing A

Event space of all possible worlds

Its area is 1

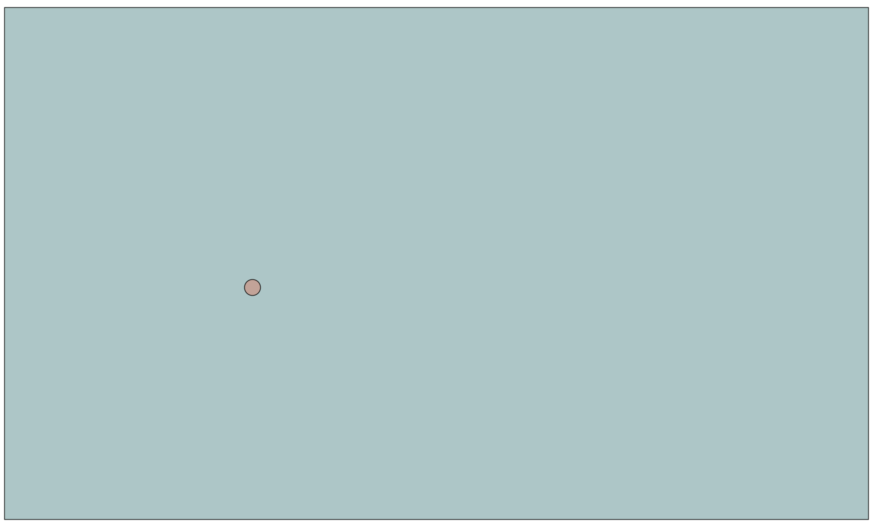


P(A) = Area of reddish oval


The
Axioms
Of
Probabili
ty

The Axioms Of Probability

- 0 <= P(A) <= 1
- P(True) = 1
- P(False) = 0
- P(A or B) = P(A) + P(B) - P(A and B)

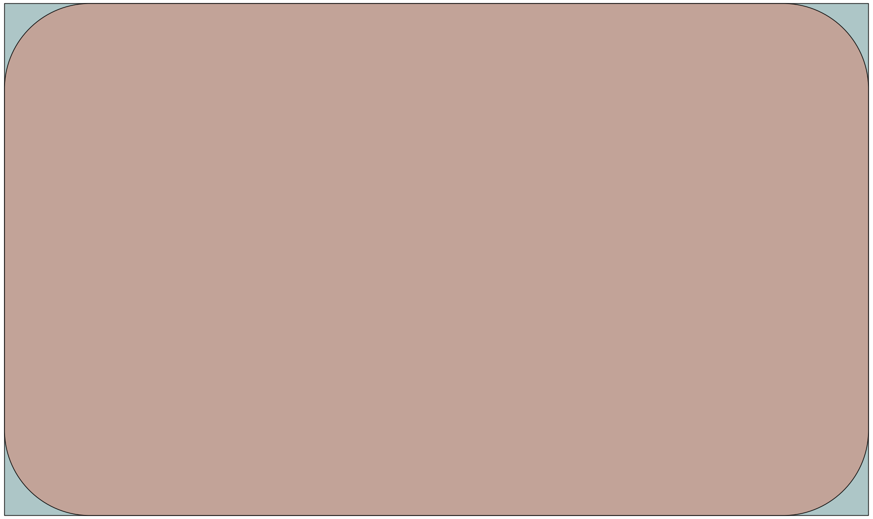


The area of A can't get any smaller than 0

And a zero area would mean no world could ever have A true

Interpreting the axioms

- 0 <= P(A) <= 1
- P(True) = 1
- P(False) = 0
- P(A or B) = P(A) + P(B) - P(A and B)

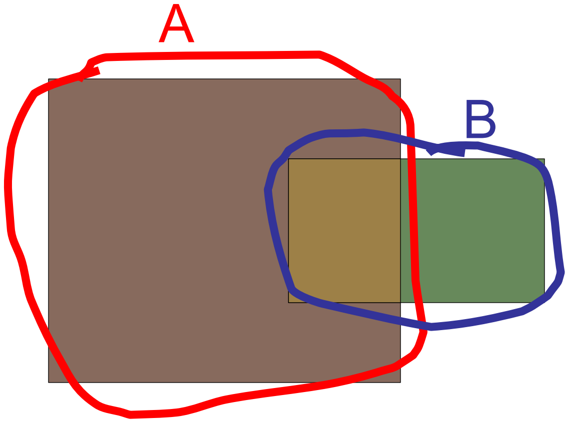


The area of A can't get any bigger than 1

And an area of 1 would mean all worlds will have A true

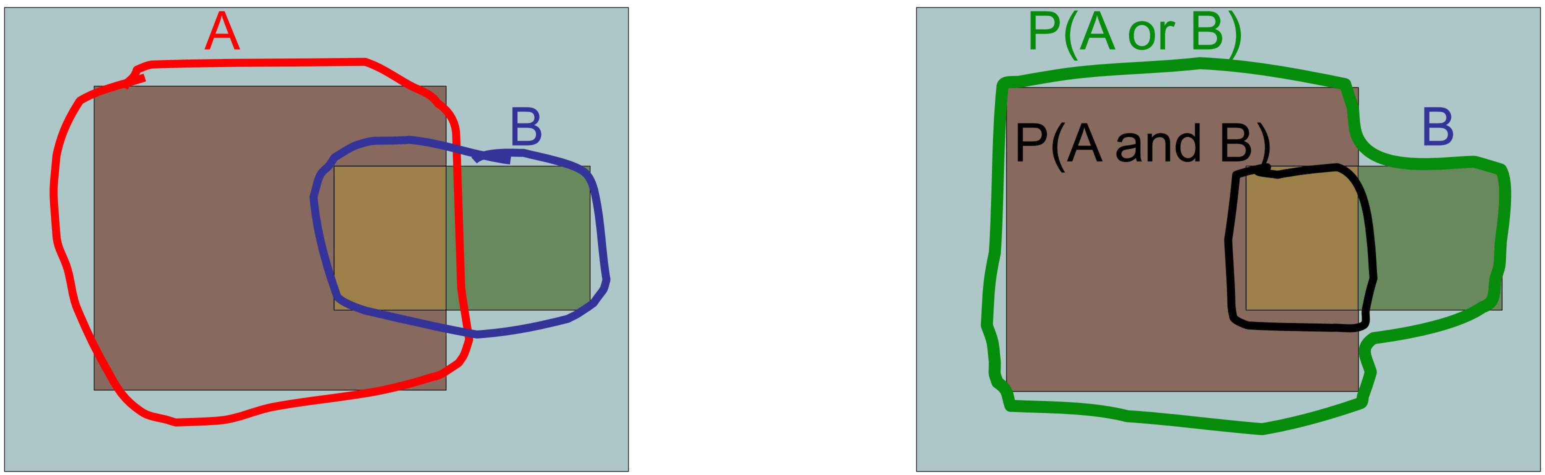
Interpreting the axioms

- $0 <= P(A) <= 1$
- $P(True) = 1$
- $P(False) = 0$
- $P(A \text{ or } B) = P(A) + P(B) - P(A \text{ and } B)$



Interpreting the axioms

- 0 <= P(A) <= 1
- P(True) = 1
- P(False) = 0
- P(A or B) = P(A) + P(B) - P(A and B)



Simple addition and subtraction

These Axioms are Not to be Trifled With

- There have been attempts to do different methodologies for uncertainty
  - Fuzzy Logic
  - Three-valued logic
  - Dempster-Shafer
  - Non-monotonic reasoning

- But the axioms of probability are the only system with this property:

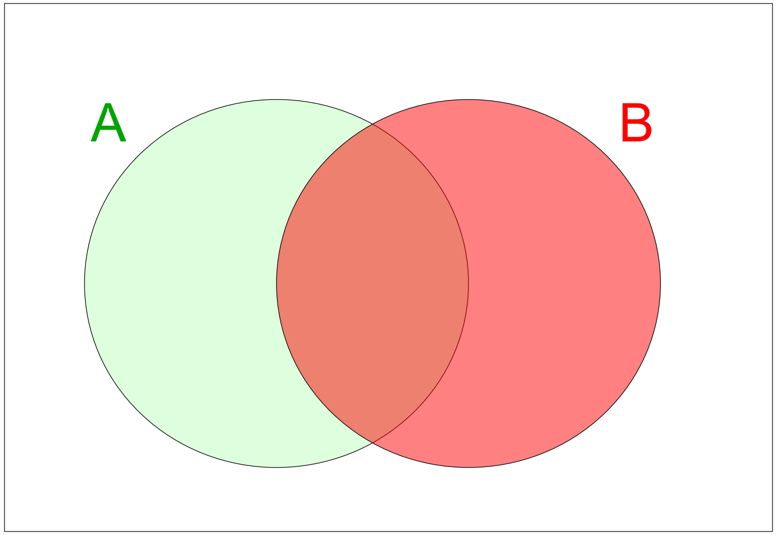
  If you gamble using them you can't be unfairly exploited by an opponent using some other system [di Finetti 1931]

Another important theorem

- $0 <= P(A) <= 1$, $P(True) = 1$, $P(False) = 0$
- $P(A \text{ or } B) = P(A) + P(B) - P(A \text{ and } B)$

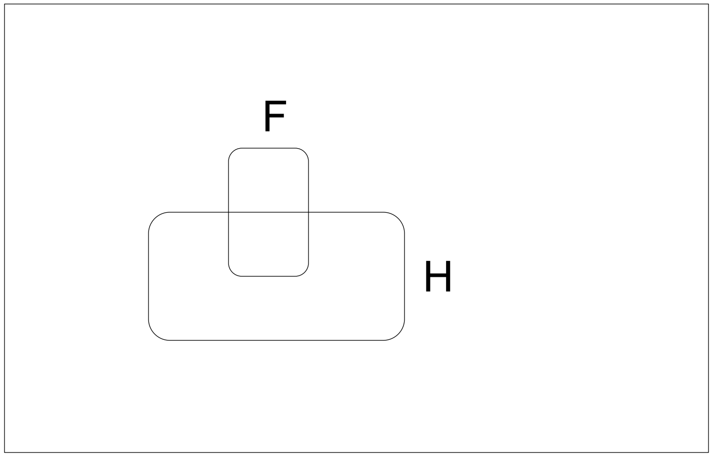
From these we can prove:

$$P(A) = P(A \text{ and } B) + P(A \text{ and not } B)$$



Conditional Probability

- P(A|B) = Fraction of worlds in which B is true that also have A true

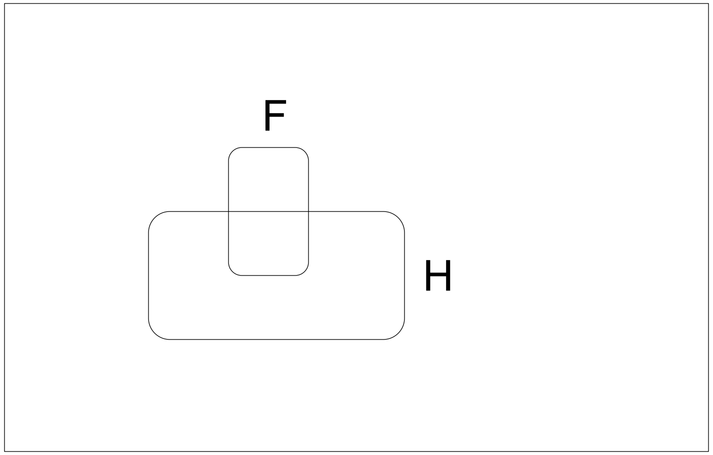


H = "Have a headache"
F = "Coming down with Flu"

P(H) = 1/10
P(F) = 1/40
P(H|F) = 1/2

"Headaches are rare and flu is rarer, but if you're coming down with 'flu there's a 50-50 chance you'll have a headache."

Conditional Probability



H = “Have a headache”
F = “Coming down with Flu”

P(H) = 1/10
P(F) = 1/40
P(H|F) = 1/2

P(H|F) = Fraction of flu-inflicted worlds in which you have a headache

$$= \frac{\text{\#worlds with flu and headache}}{\text{\#worlds with flu}}$$

$$= \frac{\text{Area of “H and F” region}}{\text{Area of “F” region}}$$

$$= \frac{P(H \text{ and } F)}{P(F)}$$

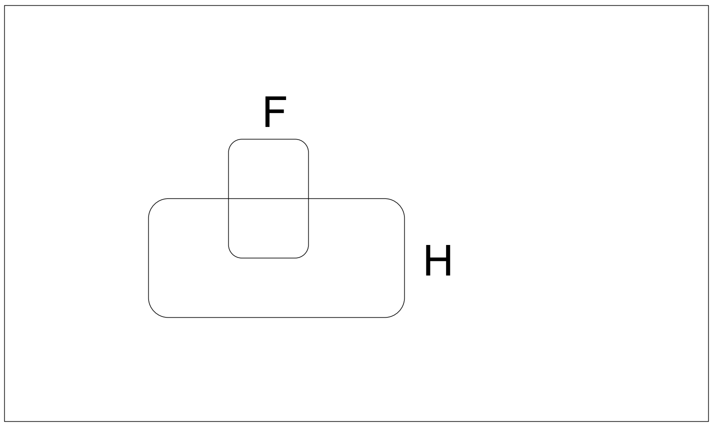
Definition of Conditional Probability

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

Corollary: The Chain Rule

$$P(A \text{ and } B) = P(A|B)\, P(B)$$

Probabilistic Inference



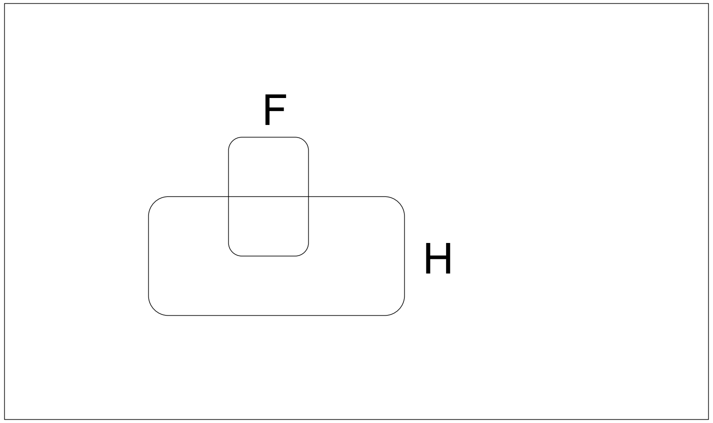
H = "Have a headache"
F = "Coming down with Flu"

P(H) = 1/10
P(F) = 1/40
P(H|F) = 1/2

One day you wake up with a headache. You think: "Drat! 50% of flus are associated with headaches so I must have a 50-50 chance of coming down with flu"

Is this reasoning good?

Probabilistic Inference



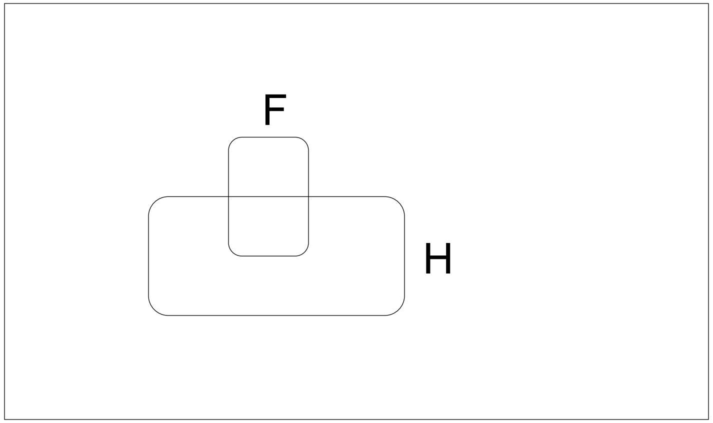
H = "Have a headache"
F = "Coming down with Flu"

P(H) = 1/10
P(F) = 1/40
P(H|F) = 1/2

P(F and H) = …

P(F|H) = …

Probabilistic Inference



H = “Have a headache”
F = “Coming down with Flu”

P(H) = 1/10
P(F) = 1/40
P(H|F) = 1/2

$$P(F \text{ and } H) = P(H \mid F) \times P(F) = \frac{1}{2} \times \frac{1}{40} = \frac{1}{80}$$

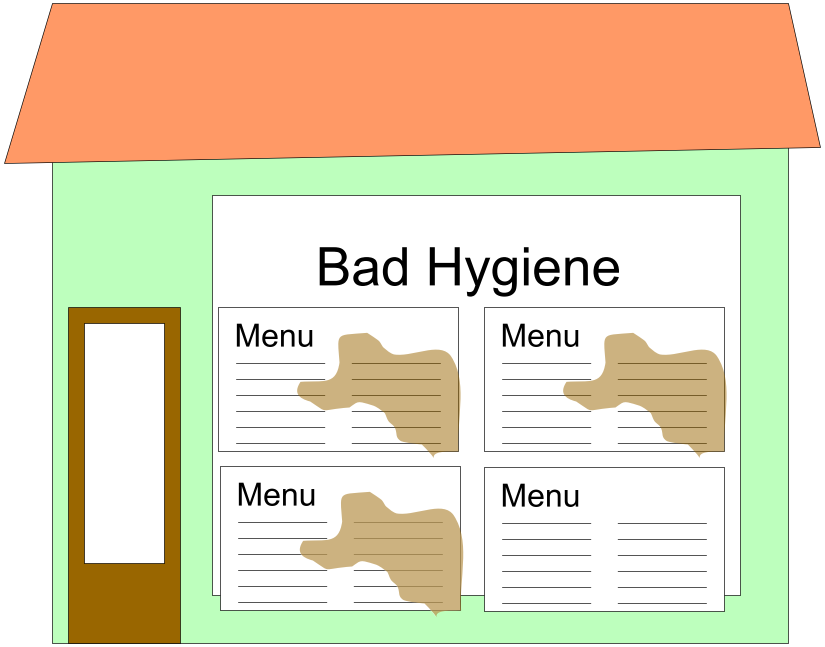
$$P(F \mid H) = \frac{P(F \text{ and } H)}{P(H)} = \frac{1/80}{1/10} = \frac{1}{8}$$

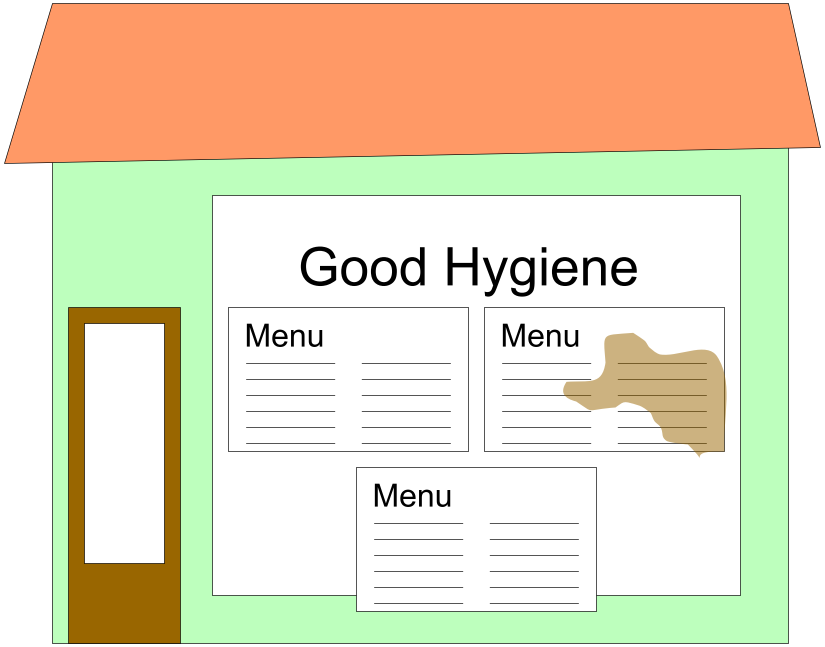
What we just did…

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

This is Bayes Rule

Bayes, Thomas (1763) An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London,* **53:370-418**





- You are a health official, deciding whether to investigate a restaurant
- You lose a dollar if you get it wrong.
- You win a dollar if you get it right
  - Half of all restaurants have bad hygiene
  - In a bad restaurant, ¾ of the menus are smudged
  - In a good restaurant, 1/3 of the menus are smudged
  - You are allowed to see a randomly chosen menu

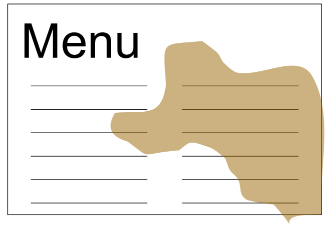
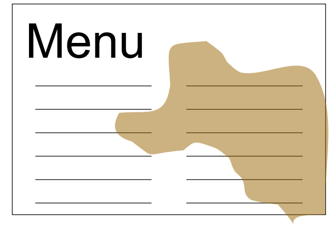
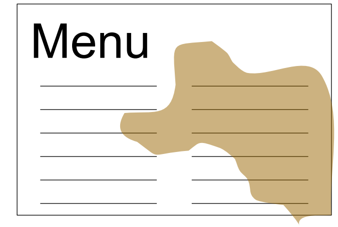
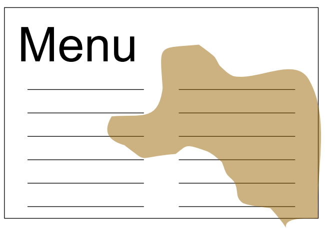
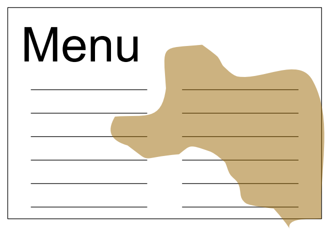
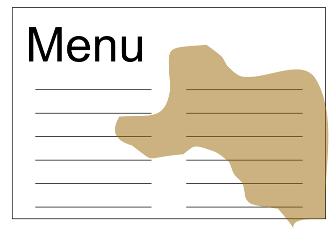
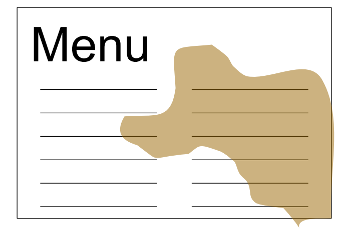
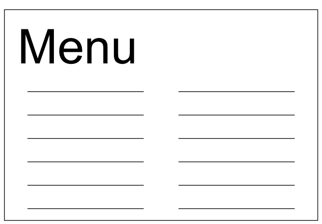
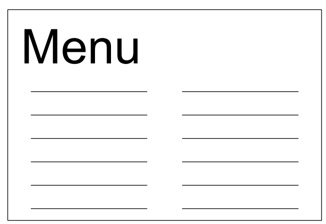
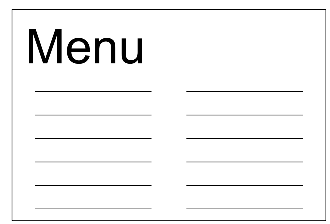
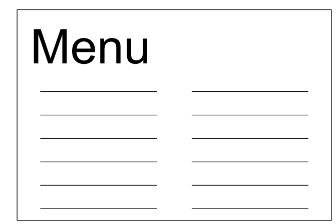
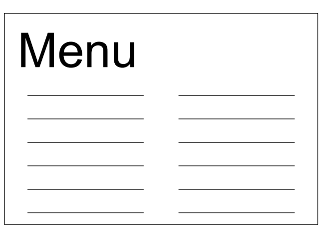
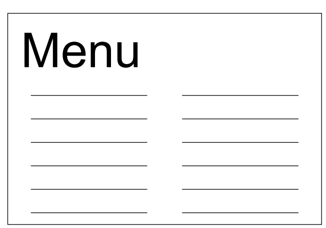
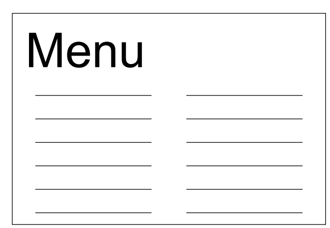
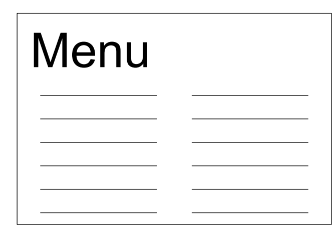
$$P(B \mid S) = \frac{P(B \text{ and } S)}{P(S)} = \frac{P(S \text{ and } B)}{P(S)}$$

$$= \frac{P(S \text{ and } B)}{P(S \text{ and } B) + P(S \text{ and not } B)}$$

$$= \frac{P(S \mid B)P(B)}{P(S \text{ and } B) + P(S \text{ and not } B)}$$

$$= \frac{P(S \mid B)P(B)}{P(S \mid B)P(B) + P(S \mid \text{not } B)P(\text{not } B)}$$

$$= \frac{\frac{3}{4} \times \frac{1}{2}}{\frac{3}{4} \times \frac{1}{2} + \frac{1}{3} \times \frac{1}{2}} = \frac{9}{13}$$


Menu

Menu

Menu

Menu

Menu

Menu

Menu

Menu

Menu

Menu

Menu

Menu

Menu

Menu

Menu

Menu

Bayesian Diagnosis

Buzzword	Meaning	In our example	Our example's value
True State	The true state of the world, which you would like to know	Is the restaurant bad?	

Bayesian Diagnosis

Buzzword	Meaning	In our example	Our example's value
True State	The true state of the world, which you would like to know	Is the restaurant bad?	
Prior	Prob(true state = x)	P(Bad)	1/2

Bayesian Diagnosis

Buzzword	Meaning	In our example	Our example's value
True State	The true state of the world, which you would like to know	Is the restaurant bad?	
Prior	Prob(true state = x)	P(Bad)	1/2
Evidence	Some symptom, or other thing you can observe	Smudge	

Bayesian Diagnosis

Buzzword	Meaning	In our example	Our example's value
True State	The true state of the world, which you would like to know	Is the restaurant bad?	
Prior	Prob(true state = x)	P(Bad)	1/2
Evidence	Some symptom, or other thing you can observe		
Conditional	Probability of seeing evidence if you did know the true state	P(Smudge|Bad)	3/4
		P(Smudge|not Bad)	1/3

Bayesian Diagnosis

Buzzword	Meaning	In our example	Our example's value
True State	The true state of the world, which you would like to know	Is the restaurant bad?	
Prior	Prob(true state = x)	P(Bad)	1/2
Evidence	Some symptom, or other thing you can observe		
Conditional	Probability of seeing evidence if you did know the true state	P(Smudge|Bad)	3/4
		P(Smudge|not Bad)	1/3
Posterior	The Prob(true state = x | some evidence)	P(Bad|Smudge)	9/13

Bayesian Diagnosis

Buzzword	Meaning	In our example	Our example's value
True State	The true state of the world, which you would like to know	Is the restaurant bad?	
Prior	Prob(true state = x)	P(Bad)	1/2
Evidence	Some symptom, or other thing you can observe		
Conditional	Probability of seeing evidence if you did know the true state	P(Smudge|Bad)	3/4
		P(Smudge|not Bad)	1/3
Posterior	The Prob(true state = x | some evidence)	P(Bad|Smudge)	9/13
Inference, Diagnosis, Bayesian Reasoning	Getting the posterior from the prior and the evidence		

Bayesian Diagnosis

Buzzword	Meaning	In our example	Our example's value
True State	The true state of the world, which you would like to know	Is the restaurant bad?	
Prior	Prob(true state = x)	P(Bad)	1/2
Evidence	Some symptom, or other thing you can observe		
Conditional	Probability of seeing evidence if you did know the true state	P(Smudge|Bad)	3/4
		P(Smudge|not Bad)	1/3
Posterior	The Prob(true state = x | some evidence)	P(Bad|Smudge)	9/13
Inference, Diagnosis, Bayesian Reasoning	Getting the posterior from the prior and the evidence		
Decision theory	Combining the posterior with known costs in order to decide what to do		

Many Pieces of Evidence

Many Pieces of Evidence

Pat walks in to the surgery.

Pat is sore and has a headache but no cough

Many Pieces of Evidence

Priors

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

Conditionals

Pat walks in to the surgery.

Pat is sore and has a headache but no cough

Many Pieces of Evidence

Priors

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

Conditionals

Pat walks in to the surgery.

Pat is sore and has a headache but no cough

What is P(F | H and not C and S) ?

The Naïve Assumption

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

The Naïve Assumption

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

If I know Pat has Flu…

…and I want to know if Pat has a cough…

…it won't help me to find out whether Pat is sore

The Naïve Assumption

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

If I know Pat has Flu…

…and I want to know if Pat has a cough…

…it won't help me to find out whether Pat is sore

$$P(C \mid F \text{ and } S) = P(C \mid F)$$

$$P(C \mid F \text{ and not } S) = P(C \mid F)$$

Coughing is *explained away* by Flu

The Naïve Assumption: ***General Case***

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

If I know the true state…

…and I want to know about one of the symptoms…

…then it won't help me to find out anything about the other symptoms

$$P(\text{Symptom} \mid \text{true state and other symptoms}) = P(\text{Symptom} \mid \text{true state})$$

Other symptoms are *explained away* by the true state

The Naïve Assumption: ***General Case***

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

If I know the true state…

…and I want to know about one of the symptoms…

…then it won't help me to find o… about the other sympto…

$P(\text{Symptom}$ … $)$

… ue state)

- What are the good things about the Naïve assumption?
- What are the bad things?

Other sympto… are *explained away* by the true state

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

$$P(F \mid H \text{ and not } C \text{ and } S)$$

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

$$P(F \mid H \text{ and not } C \text{ and } S)$$

$$= \frac{P(H \text{ and not } C \text{ and } S \text{ and } F)}{P(H \text{ and not } C \text{ and } S)}$$

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

$$P(F \mid H \text{ and not } C \text{ and } S)$$

$$= \frac{P(H \text{ and not } C \text{ and } S \text{ and } F)}{P(H \text{ and not } C \text{ and } S)}$$

$$= \frac{P(H \text{ and not } C \text{ and } S \text{ and } F)}{P(H \text{ and not } C \text{ and } S \text{ and } F) + P(H \text{ and not } C \text{ and } S \text{ and not } F)}$$

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

$$P(F \mid H \text{ and not } C \text{ and } S)$$

$$= \frac{P(H \text{ and not } C \text{ and } S \text{ and } F)}{P(H \text{ and not } C \text{ and } S)}$$

$$= \frac{P(H \text{ and not } C \text{ and } S \text{ and } F)}{P(H \text{ and not } C \text{ and } S \text{ and } F) + P(H \text{ and not } C \text{ and } S \text{ and not } F)}$$

How do I get *P(H and not C and S and F)*?

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

$$P(H \text{ and not } C \text{ and } S \text{ and } F)$$

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

$$P(H \text{ and } \text{not } C \text{ and } S \text{ and } F)$$

$$= P(H \mid \text{not } C \text{ and } S \text{ and } F) \times P(\text{not } C \text{ and } S \text{ and } F)$$

Chain rule: P(■ and ■) = P(■ | ■) × P(■)

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

$$P(H \text{ and not } C \text{ and } S \text{ and } F)$$

$$= P(H \mid \text{not } C \text{ and } S \text{ and } F) \times P(\text{not } C \text{ and } S \text{ and } F)$$

$$= P(H \mid F) \times P(\text{not } C \text{ and } S \text{ and } F)$$

Naïve assumption: lack of cough and soreness have no effect on headache if I am already assuming Flu

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

$$P(H \text{ and not } C \text{ and } S \text{ and } F)$$

$$= P(H \mid \text{not } C \text{ and } S \text{ and } F) \times P(\text{not } C \text{ and } S \text{ and } F)$$

$$= P(H \mid F) \times P(\text{not } C \text{ and } S \text{ and } F)$$

$$= P(H \mid F) \times P(\text{not } C \mid S \text{ and } F) \times P(S \text{ and } F)$$

Chain rule: P(▇ and ▇) = P(▇ | ▇) × P(▇)

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

$$P(H \text{ and not } C \text{ and } S \text{ and } F)$$

$$= P(H \mid \text{not } C \text{ and } S \text{ and } F) \times P(\text{not } C \text{ and } S \text{ and } F)$$

$$= P(H \mid F) \times P(\text{not } C \text{ and } S \text{ and } F)$$

$$= P(H \mid F) \times P(\text{not } C \mid S \text{ and } F) \times P(S \text{ and } F)$$

$$= P(H \mid F) \times P(\text{not } C \mid F) \times P(S \text{ and } F)$$

Naïve assumption: Sore has no effect on Cough if I am already assuming Flu

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

$$P(H \text{ and not } C \text{ and } S \text{ and } F)$$

$$= P(H \mid \text{not } C \text{ and } S \text{ and } F) \times P(\text{not } C \text{ and } S \text{ and } F)$$

$$= P(H \mid F) \times P(\text{not } C \text{ and } S \text{ and } F)$$

$$= P(H \mid F) \times P(\text{not } C \mid S \text{ and } F) \times P(S \text{ and } F)$$

$$= P(H \mid F) \times P(\text{not } C \mid F) \times P(S \text{ and } F)$$

$$= P(H \mid F) \times P(\text{not } C \mid F) \times P(S \mid F) \times P(F)$$

Chain rule: P(and) = P(|) × P( )

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

$$\boxed{P(H \text{ and not } C \text{ and } S \text{ and } F)}$$

$$= P(H \mid \text{not } C \text{ and } S \text{ and } F) \times P(\text{not } C \text{ and } S \text{ and } F)$$

$$= P(H \mid F) \times P(\text{not } C \text{ and } S \text{ and } F)$$

$$= P(H \mid F) \times P(\text{not } C \mid S \text{ and } F) \times P(S \text{ and } F)$$

$$= P(H \mid F) \times P(\text{not } C \mid F) \times P(S \text{ and } F)$$

$$\boxed{= P(H \mid F) \times P(\text{not } C \mid F) \times P(S \mid F) \times P(F)}$$

$$= \frac{1}{2} \times \left(1 - \frac{2}{3}\right) \times \frac{3}{4} \times \frac{1}{40} = \frac{1}{320}$$

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

$$P(F \mid H \text{ and not } C \text{ and } S)$$

$$= \frac{P(H \text{ and not } C \text{ and } S \text{ and } F)}{P(H \text{ and not } C \text{ and } S)}$$

$$= \frac{P(H \text{ and not } C \text{ and } S \text{ and } F)}{P(H \text{ and not } C \text{ and } S \text{ and } F) + P(H \text{ and not } C \text{ and } S \text{ and not } F)}$$

P(Flu)	=	1/40	P(Not Flu)	=	39/40
P(Headache | Flu)	=	1/2	P(Headache | not Flu)	=	7 / 78
P(Cough | Flu)	=	2/3	P(Cough | not Flu)	=	1/6
P(Sore | Flu)	=	3/4	P(Sore | not Flu)	=	1/3

$$P(H \text{ and not } C \text{ and } S \text{ and not } F) =$$

$$= P(H \mid \text{not } C \text{ and } S \text{ and not } F) \times P(\text{not } C \text{ and } S \text{ and not } F)$$

$$= P(H \mid \text{not } F) \times P(\text{not } C \text{ and } S \text{ and not } F)$$

$$= P(H \mid \text{not } F) \times P(\text{not } C \mid S \text{ and not } F) \times P(S \text{ and not } F)$$

$$= P(H \mid \text{not } F) \times P(\text{not } C \mid \text{not } F) \times P(S \text{ and not } F)$$

$$= P(H \mid \text{not } F) \times P(\text{not } C \mid \text{not } F) \times P(S \mid \text{not } F) \times P(\text{not } F)$$

$$= \frac{7}{78} \times \left(1 - \frac{1}{6}\right) \times \frac{1}{3} \times \frac{39}{40} = \frac{7}{288}$$

P(Flu)	= 1/40	P(Not Flu)	= 39/40
P(Headache | Flu)	= 1/2	P(Headache | not Flu)	= 7 / 78
P(Cough | Flu)	= 2/3	P(Cough | not Flu)	= 1/6
P(Sore | Flu)	= 3/4	P(Sore | not Flu)	= 1/3

$$P(F \mid H \text{ and not } C \text{ and } S)$$

$$= \frac{P(H \text{ and not } C \text{ and } S \text{ and } F)}{P(H \text{ and not } C \text{ and } S)}$$

$$= \frac{\overbrace{P(H \text{ and not } C \text{ and } S \text{ and } F)}^{\frac{1}{320}}}{\underbrace{P(H \text{ and not } C \text{ and } S \text{ and } F)}_{\frac{1}{320}} + \underbrace{P(H \text{ and not } C \text{ and } S \text{ and not } F)}_{\frac{7}{288}}}$$

= 0.1139 (11% chance of Flu, given symptoms)

Building A Bayes Classifier

Priors

P(Flu)	= 1/40	P(Not Flu)	= 39/40
P(Headache | Flu)	= 1/2	P(Headache | not Flu)	= 7 / 78
P(Cough | Flu)	= 2/3	P(Cough | not Flu)	= 1/6
P(Sore | Flu)	= 3/4	P(Sore | not Flu)	= 1/3

Conditionals

The General Case

Building a naïve Bayesian Classifier

Assume:

- True state has *N* possible values: *1, 2, 3 .. N*
- There are *K* symptoms called $Symptom_1$, $Symptom_2$, … $Symptom_K$
- $Symptom_i$ has M_i possible values: *1, 2, .. M_i*

P(State=1) = ___		P(State=2) = ___	…	P(State=N) = ___
P(Sym_1=1 | State=1) = ___		P(Sym_1=1 | State=2) = ___	…	P(Sym_1=1 | State=N) = ___
P(Sym_1=2 | State=1) = ___		P(Sym_1=2 | State=2) = ___	…	P(Sym_1=2 | State=N) = ___
: :	:	: :		: :
P(Sym_1=M_1 | State=1) = ___		P(Sym_1=M_1 | State=2) = ___	…	P(Sym_1=M_1 | State=N) = ___
P(Sym_2=1 | State=1) = ___		P(Sym_2=1 | State=2) = ___	…	P(Sym_2=1 | State=N) = ___
P(Sym_2=2 | State=1) = ___		P(Sym_2=2 | State=2) = ___	…	P(Sym_2=2 | State=N) = ___
: :	:	: :		: :
P(Sym_2=M_2 | State=1) = ___		P(Sym_2=M_2 | State=2) = ___	…	P(Sym_2=M_2 | State=N) = ___
:		:		:
P(Sym_K=1 | State=1) = ___		P(Sym_K=1 | State=2) = ___	…	P(Sym_K=1 | State=N) = ___
P(Sym_K=2 | State=1) = ___		P(Sym_K=2 | State=2) = ___	…	P(Sym_K=2 | State=N) = ___
: :	:	: :		: :
P(Sym_K=M_K | State=1) = ___		P(Sym_K=M_1 | State=2) = ___	…	P(Sym_K=M_1 | State=N) = ___

Building a naïve Bayesian Classifier

Assume:

- True state has *N* values: *1, 2, 3 .. N*
- There are *K* symptoms called $Symptom_1$, $Symptom_2$, … $Symptom_K$
- $Symptom_i$ has M_i values: *1, 2, .. M_i*

P(State=1) = ___		P(State=2) = ___	…	P(State=N) = ___
$P(Sym_1=1 \mid State=1)$ = ___		$P(Sym_1=1 \mid State=2)$ = ___	…	$P(Sym_1=1 \mid State=N)$ = ___
$P(Sym_1=2 \mid State=1)$ = ___		$P(Sym_1=2 \mid State=2)$ = ___	…	$P(Sym_1=2 \mid State=N)$ = ___
: :	:	: :		: :
$P(Sym_1=M_1 \mid State=1)$ = ___		$\boxed{P(Sym_1=M_1 \mid State=2) = \_\_\_}$	…	$P(Sym_1=M_1 \mid State=N)$ = ___
$P(Sym_2=1 \mid State=1)$ = ___		P(Sy … $\mid$ State=2) = ___	…	$P(Sym_2=1 \mid State=N)$ = ___
$P(Sym_2=2 \mid State=1)$ = ___		P(… State=2) = ___	…	$P(Sym_2=2 \mid State=N)$ = ___
:		:		: :
P(Sym_2=M …		…) = ___	…	$P(Sym_2=M_2 \mid State=N)$ = ___
				:
P(Sym_K=1 …		= ___	…	$P(Sym_K=1 \mid State=N)$ = ___
$P(Sym_K=2 \mid State=1)$ = ___		$P(Sym_K=2 \mid State=2)$ = ___	…	$P(Sym_K=2 \mid State=N)$ = ___
: :	:	: :		: :
$P(Sym_K=M_K \mid State=1)$ = ___		$P(Sym_K=M_1 \mid State=2)$ = ___	…	$P(Sym_K=M_1 \mid State=N)$ = ___

Example:
P(Anemic | Liver Cancer) = 0.21

P(State=1) = ___		P(State=2) = ___	…	P(State=N) = ___
P(Sym$_1$=1 | State=1) = ___		P(Sym$_1$=1 | State=2) = ___	…	P(Sym$_1$=1 | State=N) = ___
P(Sym$_1$=2 | State=1) = ___		P(Sym$_1$=2 | State=2) = ___	…	P(Sym$_1$=2 | State=N) = ___
: :	:	: :		: :
P(Sym$_1$=M$_1$ | State=1) = ___		P(Sym$_1$=M$_1$ | State=2) = ___	…	P(Sym$_1$=M$_1$ | State=N) = ___
P(Sym$_2$=1 | State=1) = ___		P(Sym$_2$=1 | State=2) = ___	…	P(Sym$_2$=1 | State=N) = ___
P(Sym$_2$=2 | State=1) = ___		P(Sym$_2$=2 | State=2) = ___	…	P(Sym$_2$=2 | State=N) = ___
: :	:	: :		: :
P(Sym$_2$=M$_2$ | State=1) = ___		P(Sym$_2$=M$_2$ | State=2) = ___	…	P(Sym$_2$=M$_2$ | State=N) = ___
:		:		:
P(Sym$_K$=1 | State=1) = ___		P(Sym$_K$=1 | State=2) = ___	…	P(Sym$_K$=1 | State=N) = ___
P(Sym$_K$=2 | State=1) =		P(Sym$_K$=2 | State=2) =		P(Sym$_K$=2 | State=N) =
:				
P				

$$P(\text{state} = Y \mid \text{symp}_1 = X_1 \text{ and } \text{symp}_2 = X_2 \text{ and } \boxed{?}\, \text{symp}_n = X_n)$$

$$= \frac{P(\text{symp}_1 = X_1 \text{ and } \text{symp}_2 = X_2 \text{ and } \boxed{?}\, \text{symp}_n = X_n \text{ and state} = Y)}{P(\text{symp}_1 = X_1 \text{ and } \text{symp}_2 = X_2 \text{ and } \boxed{?}\, \text{symp}_n = X_n)}$$

$$= \frac{P(\text{symp}_1 = X_1 \text{ and } \text{symp}_2 = X_2 \text{ and } \boxed{?}\, \text{symp}_n = X_n \text{ and state} = Y)}{\sum_Z P(\text{symp}_1 = X_1 \text{ and } \text{symp}_2 = X_2 \text{ and } \boxed{?}\, \text{symp}_n = X_n \text{ and state} = Z)}$$

$$= \frac{\left[\prod_{i=1}^{n} P(\text{symp}_i = X_i \mid \text{state} = Y)\right] P(\text{state} = Y)}{\sum_Z \left[\prod_{i=1}^{n} P(\text{symp}_i = X_i \mid \text{state} = Z)\right] P(\text{state} = Z)}$$

P(State=1) = ___		P(State=2) = ___	…	P(State=N) = ___
$P(Sym_1=1 \mid State=1)$ = ___		$P(Sym_1=1 \mid State=2)$ = ___	…	$P(Sym_1=1 \mid State=N)$ = ___
$P(Sym_1=2 \mid State=1)$ = ___		$P(Sym_1=2 \mid State=2)$ = ___	…	$P(Sym_1=2 \mid State=N)$ = ___
: :	:	: :		: :
$P(Sym_1=M_1 \mid State=1)$ = ___		$P(Sym_1=M_1 \mid State=2)$ = ___	…	$P(Sym_1=M_1 \mid State=N)$ = ___
$P(Sym_2=1 \mid State=1)$ = ___		$P(Sym_2=1 \mid State=2)$ = ___	…	$P(Sym_2=1 \mid State=N)$ = ___
$P(Sym_2=2 \mid State=1)$ = ___		$P(Sym_2=2 \mid State=2)$ = ___	…	$P(Sym_2=2 \mid State=N)$ = ___
: :	:	: :		: :
$P(Sym_2=M_2 \mid State=1)$ = ___		$P(Sym_2=M_2 \mid State=2)$ = ___	…	$P(Sym_2=M_2 \mid State=N)$ = ___
:		:		:
$P(Sym_K=1 \mid State=1)$ = ___		$P(Sym_K=1 \mid State=2)$ = ___	…	$P(Sym_K=1 \mid State=N)$ = ___
$P(Sym_K=2 \mid State=1)$ =		$P(Sym_K=2 \mid State=2)$ =		$P(Sym_K=2 \mid State=N)$ =

$$P(\text{state} = Y \mid \text{symp}_1 = X_1 \text{ and } \text{symp}_2 = X_2 \ldots$$

$$= \frac{\left[\prod \ldots X_i \mid \text{state} = Y)\right] P(\text{state} = Y)}{\sum_Z \left[\prod_{i=1}^{n} P(\text{symp}_i = X_i \mid \text{state} = Z)\right] P(\text{state} = Z)}$$

Coming Soon: How this is used in Practical Biosurveillance

Also coming soon: Bringing time and space into this kind of reasoning. And how to not be naïve.

Conclusion

- You will hear lots of “Bayesian” this and “conditional probability” that this week.
- It’s simple: don’t let wooly academic types trick you into thinking it is fancy.
- You should know:
  - What are: Bayesian Reasoning, Conditional Probabilities, Priors, Posteriors.
  - Appreciate how conditional probabilities are manipulated.
  - Why the Naïve Bayes Assumption is Good.
  - Why the Naïve Bayes Assumption is Evil.

Text mining

- Motivation: an enormous (and growing!) supply of rich data
- Most of the available text data is unstructured…
- Some of it is semi-structured:
  - Header entries (title, authors' names, section titles, keyword lists, etc.)
  - Running text bodies (main body, abstract, summary, etc.)
- Natural Language Processing (NLP)
- Text Information Retrieval

Text processing

- Natural Language Processing:
  - Automated understanding of text is a very very very challenging Artificial Intelligence problem
  - Aims on extracting *semantic contents* of the processed documents
  - Involves extensive research into semantics, grammar, automated reasoning, …
  - Several factors making it tough for a computer include:
    - *Polysemy* (the same word having several different meanings)
    - *Synonymy* (several different ways to describe the same thing)

Text processing

- Text Information Retrieval:
  - Search through collections of documents in order to find objects:
    - relevant to a specific query
    - similar to a specific document
  - For practical reasons, the text documents are parameterized
  - Terminology:
    - *Documents* (*text data units: books, articles, paragraphs, other chunks such as email messages, ...*)
    - *Terms* (*specific words, word pairs, phrases*)

Text Information Retrieval

- Typically, the text databases are parametrized with a *document-term matrix*
- Each row of the matrix corresponds to one of the documents
- Each column corresponds to a different term

Shortness of breath

Difficulty breathing

Rash on neck

Sore neck and difficulty breathing

Just plain ugly

Text Information Retrieval

- Typically, the text databases are parametrized with a ***document-term matrix***
- Each row of the matrix corresponds to one of the documents
- Each column corresponds to a different term

	breath	difficulty	just	neck	plain	rash	shor t	sore	ugly
Shortness of breath	1	0	0	0	0	0	1	0	0
Difficulty breathing	1	1	0	0	0	0	0	0	0
Rash on neck	0	0	0	1	0	1	0	0	0
Sore neck and difficulty breathing	1	1	0	1	0	0	0	1	0
Just plain ugly	0	0	1	0	1	0	0	0	1

Parametrization
for Text Information Retrieval

- Depending on the particular method of parametrization the matrix entries may be:
  - binary
    (telling whether a term *Tj* is present in the document *Di* or not)
  - counts (frequencies)
    (total number of repetitions of a term *Tj* in *Di*)
  - weighted frequencies
    (see the slide following the next)

Typical applications of Text IR

- Document indexing and classification
  (e.g. library systems)

- Search engines
  (e.g. the Web)

- Extraction of information from textual sources
  (e.g. profiling of personal records, consumer complaint processing)

Typical applications of Text IR

- Document indexing and classification
  (e.g. library systems)

- Search engines
  (e.g. the Web)

- Extraction of information from textual sources
  (e.g. profiling of personal records, consumer complaint processing)

Building a naïve Bayesian Classifier

Assume:

- True state has N values: $1, 2, 3 .. N$
- There are K symptoms called $Symptom_1$, $Symptom_2$, … $Symptom_K$
- $Symptom_i$ has M_i values: $1, 2, .. M_i$

$P(State=1)$ = ___		$P(State=2)$ = ___	…	$P(State=N)$ = ___
$P(Sym_1=1 \mid State=1)$ = ___		$P(Sym_1=1 \mid State=2)$ = ___	…	$P(Sym_1=1 \mid State=N)$ = ___
$P(Sym_1=2 \mid State=1)$ = ___		$P(Sym_1=2 \mid State=2)$ = ___	…	$P(Sym_1=2 \mid State=N)$ = ___
: :	:	: :		: :
$P(Sym_1=M_1 \mid State=1)$ = ___		$P(Sym_1=M_1 \mid State=2)$ = ___	…	$P(Sym_1=M_1 \mid State=N)$ = ___
$P(Sym_2=1 \mid State=1)$ = ___		$P(Sym_2=1 \mid State=2)$ = ___	…	$P(Sym_2=1 \mid State=N)$ = ___
$P(Sym_2=2 \mid State=1)$ = ___		$P(Sym_2=2 \mid State=2)$ = ___	…	$P(Sym_2=2 \mid State=N)$ = ___
: :	:	: :		: :
$P(Sym_2=M_2 \mid State=1)$ = ___		$P(Sym_2=M_2 \mid State=2)$ = ___	…	$P(Sym_2=M_2 \mid State=N)$ = ___
:		:		:
$P(Sym_K=1 \mid State=1)$ = ___		$P(Sym_K=1 \mid State=2)$ = ___	…	$P(Sym_K=1 \mid State=N)$ = ___
$P(Sym_K=2 \mid State=1)$ = ___		$P(Sym_K=2 \mid State=2)$ = ___	…	$P(Sym_K=2 \mid State=N)$ = ___
: :	:	: :		: :
$P(Sym_K=M_K \mid State=1)$ = ___		$P(Sym_K=M_1 \mid State=2)$ = ___	…	$P(Sym_K=M_1 \mid State=N)$ = ___

Building a naïve Bayesian Classifier

Assume:

- prodrome has N values: GI, Respiratory, Constitutional … N
- There are K symptoms called $Symptom_1$, $Symptom_2$, … $Symptom_K$
- $Symptom_i$ has M_i values: $1, 2, .. M_i$

P(State=1) = ___		P(State=2) = ___	…	P(State=N) = ___
P(Sym_1=1 | State=1) = ___		P(Sym_1=1 | State=2) = ___	…	P(Sym_1=1 | State=N) = ___
P(Sym_1=2 | State=1) = ___		P(Sym_1=2 | State=2) = ___	…	P(Sym_1=2 | State=N) = ___
: :	:	: :		: :
P(Sym_1=M_1 | State=1) = ___		P(Sym_1=M_1 | State=2) = ___	…	P(Sym_1=M_1 | State=N) = ___
P(Sym_2=1 | State=1) = ___		P(Sym_2=1 | State=2) = ___	…	P(Sym_2=1 | State=N) = ___
P(Sym_2=2 | State=1) = ___		P(Sym_2=2 | State=2) = ___	…	P(Sym_2=2 | State=N) = ___
: :	:	: :		: :
P(Sym_2=M_2 | State=1) = ___		P(Sym_2=M_2 | State=2) = ___	…	P(Sym_2=M_2 | State=N) = ___
:		:		:
P(Sym_K=1 | State=1) = ___		P(Sym_K=1 | State=2) = ___	…	P(Sym_K=1 | State=N) = ___
P(Sym_K=2 | State=1) = ___		P(Sym_K=2 | State=2) = ___	…	P(Sym_K=2 | State=N) = ___
: :	:	: :		: :
P(Sym_K=M_K | State=1) = ___		P(Sym_K=M_1 | State=2) = ___	…	P(Sym_K=M_1 | State=N) = ___

Building a naïve Bayesian Classifier

Assume:

GI, Respiratory, Constitutional …

- prodrome has N values: 1, 2, 3 .. N
- There are K s words s called word$_1$ m_1, S word$_2$ m_2, … S word$_K$ m_K
- $Symptom_i$ has M_i values: $1, 2, .. M_i$

P(State=1) = ___		P(State=2) = ___	…	P(State=N) = ___
$P(Sym_1=1 \mid State=1)$ = ___		$P(Sym_1=1 \mid State=2)$ = ___	…	$P(Sym_1=1 \mid State=N)$ = ___
$P(Sym_1=2 \mid State=1)$ = ___		$P(Sym_1=2 \mid State=2)$ = ___	…	$P(Sym_1=2 \mid State=N)$ = ___
: :	:	: :		: :
$P(Sym_1=M_1 \mid State=1)$ = ___		$P(Sym_1=M_1 \mid State=2)$ = ___	…	$P(Sym_1=M_1 \mid State=N)$ = ___
$P(Sym_2=1 \mid State=1)$ = ___		$P(Sym_2=1 \mid State=2)$ = ___	…	$P(Sym_2=1 \mid State=N)$ = ___
$P(Sym_2=2 \mid State=1)$ = ___		$P(Sym_2=2 \mid State=2)$ = ___	…	$P(Sym_2=2 \mid State=N)$ = ___
: :	:	: :		: :
$P(Sym_2=M_2 \mid State=1)$ = ___		$P(Sym_2=M_2 \mid State=2)$ = ___	…	$P(Sym_2=M_2 \mid State=N)$ = ___
:		:		:
$P(Sym_K=1 \mid State=1)$ = ___		$P(Sym_K=1 \mid State=2)$ = ___	…	$P(Sym_K=1 \mid State=N)$ = ___
$P(Sym_K=2 \mid State=1)$ = ___		$P(Sym_K=2 \mid State=2)$ = ___	…	$P(Sym_K=2 \mid State=N)$ = ___
: :	:	: :		: :
$P(Sym_K=M_K \mid State=1)$ = ___		$P(Sym_K=M_1 \mid State=2)$ = ___	…	$P(Sym_K=M_1 \mid State=N)$ = ___

Building a naïve Bayesian Classifier

Assume:

- prodrome has N values: GI, Respiratory, Constitutional …
- There are K words called word_1, word_2, … word_K
- word_i is either present or absent_i

P(State=1) = ___		P(State=2) = ___	…	P(State=N) = ___
P(Sym_1=1 | State=1) = ___		P(Sym_1=1 | State=2) = ___	…	P(Sym_1=1 | State=N) = ___
P(Sym_1=2 | State=1) = ___		P(Sym_1=2 | State=2) = ___	…	P(Sym_1=2 | State=N) = ___
: :	:	: :		: :
P(Sym_1=M_1 | State=1) = ___		P(Sym_1=M_1 | State=2) = ___	…	P(Sym_1=M_1 | State=N) = ___
P(Sym_2=1 | State=1) = ___		P(Sym_2=1 | State=2) = ___	…	P(Sym_2=1 | State=N) = ___
P(Sym_2=2 | State=1) = ___		P(Sym_2=2 | State=2) = ___	…	P(Sym_2=2 | State=N) = ___
: :	:	: :		: :
P(Sym_2=M_2 | State=1) = ___		P(Sym_2=M_2 | State=2) = ___	…	P(Sym_2=M_2 | State=N) = ___
:		:		:
P(Sym_K=1 | State=1) = ___		P(Sym_K=1 | State=2) = ___	…	P(Sym_K=1 | State=N) = ___
P(Sym_K=2 | State=1) = ___		P(Sym_K=2 | State=2) = ___	…	P(Sym_K=2 | State=N) = ___
: :	:	: :		: :
P(Sym_K=M_K | State=1) = ___		P(Sym_K=M_1 | State=2) = ___	…	P(Sym_K=M_1 | State=N) = ___

Building a naïve Bayesian Classifier

Assume:

- prodrome has N values: GI, Respiratory, Constitutional …
- There are K words called $word_1$, $word_2$, … $word_K$
- $word_i$ is either present or absent$_i$

P(Prod'm=GI) = ___		P(Prod'm=respir) = ___	…	P(Prod'm=const) = ___
P(angry | Prod'm=GI) = ___		P(angry | Prod'm=respir) = ___	…	P(angry | Prod'm=const) = ___
P(~angry | Prod'm=GI) = ___		P(~angry | Prod'm=respir) = ___	…	P(~angry | Prod'm=const) = ___
P(blood | Prod'm=GI) = ___		P(blood | Prod'm=respir) = ___	…	P(blood | Prod'm=const) = ___
P(~blood | Prod'm=GI) = ___		P(~blood | Prod'm=respir) = ___	…	P(~blood | Prod'm=const) = ___
:		:		:
P(vomit | Prod'm=GI) = ___		P(vomit | Prod'm=respir) = ___	…	P(vomit | Prod'm=const) = ___
P(~vomit | Prod'm=GI) = ___		P(~vomit |Prod'm=respir) = ___	…	P(~vomit | Prod'm=const) = ___

Building a naïve Bayesian Classifier

Assume:

- prodrome has N values: GI, Respiratory, Constitutional …
- There are K words called word$_1$, word$_2$, … word$_K$
- word$_i$ has is either present or absent$_i$

P(Prod'm=GI) = ___	P(Prod'm=respir) = ___	…	P(Prod'm=const) = ___
P(angry | Prod'm=GI) = ___	P(angry | Prod'm=respir) = ___	…	P(angry | Prod'm=const) = ___
P(~angry | Prod'm=GI) = ___	P(~angry | Prod'm=respir) = ___	…	P(~angry | Prod'm=const) = ___
P(blood | Prod'm=GI) = ___	P(blood | Prod'm=respir) = ___	…	P(blood | Prod'm=const) = ___
P(~blood | Prod'm=GI) = ___	P(ood | n=respir) = ___	…	P(~blood | Prod'm=const) = ___
:			:
P(vomit | Prod'm=GI) =	mit | Prod'm=respir) = ___	…	P(vomit | Prod'm=const) = ___
P(~vomit | Prod'm=GI)	vomit |Prod'm=respir) = ___	…	P(~vomit | Prod'm=const) = ___

Example:
Prob(Chief Complaint contains “Blood” | Prodrome = Respiratory) = 0.003

Building a naïve Bayesian Classifier

Assume:

prodrome

GI, Respiratory, Constitutional …

- … has N values: … N
- There are K s… words …s called word$_1$ … m_1, S … word … S word$_K$ … m_K
- S word$_i$ … $_i$ has … is eith…

Q: Where do these numbers come from?

P(Proc…			…d'm=const) = ___
P(angry…	…d'm=respir) = ___	…	P(angry | Prod'm=const) = ___
P(~angr…) = ___	P(~angry | Prod'm=respir) = ___	…	P(~angry | Prod'm=const) = ___
P(blood | Prod'm=GI) = ___	P(b… | Prod'm=respir) = ___	…	P(blood | Prod'm=const) = ___
P(~blood | Prod'm=GI) = ___	…od | …=respir) = ___	…	P(~blood | Prod'm=const) = ___
:			:
P(vomit | Prod'm=GI) = …	…mit | Prod'm=respir) = ___	…	P(vomit | Prod'm=const) = ___
P(~vomit | Prod'm=GI)…	…vomit |Prod'm=respir) = ___	…	P(~vomit | Prod'm=const) = ___

Example:
Prob(Chief Complaint contains "Blood" | Prodrome = Respiratory) = 0.003

Building a naïve Bayesian Classifier

Assume:

- prodrome has N values: GI, Respiratory, Constitutional …
- There are K s words s called word$_1$ m_1, S … word$_K$ m_K
- S word$_i$ $_i$ has

Q: Where do these numbers come from?

A: Learn them from expert-labeled data

P(Prod			d'm=const) = ___
P(angry	Prod'm=respir) = ___	…	P(angry | Prod'm=const) = ___
P(~angr) = ___	P(~angry | Prod'm=respir) = ___	…	P(~angry | Prod'm=const) = ___
P(blood | Prod'm=GI)			d | Prod'm=const) = ___
P(~blood			d | Prod'm=const) = ___
			:
P(vomit |	nit | Prod'm=respir) = ___	…	P(vomit | Prod'm=const) = ___
P(~vomit | Prod'm=GI)	vomit |Prod'm=respir) = ___	…	P(~vomit | Prod'm=const) = ___

Example:
Prob(Chief Complaint contains “Blood” | Prodrome = Respiratory) = 0.003

Learning a Bayesian Classifier

1. Before deployment of classifier,

	breath	difficulty	just	neck	plain	rash	shor t	sore	ugly
Shortness of breath	1	0	0	0	0	0	1	0	0
Difficulty breathing	1	1	0	0	0	0	0	0	0
Rash on neck	0	0	0	1	0	1	0	0	0
Sore neck and difficulty breathing	1	1	0	1	0	0	0	1	0
Just plain ugly	0	0	1	0	1	0	0	0	1

Learning a Bayesian Classifier

1. Before deployment of classifier, get labeled training data

	EXPERT SAYS	breath	difficulty	just	neck	plain	rash	shor t	sore	ugly
Shortness of breath	Resp	1	0	0	0	0	0	1	0	0
Difficulty breathing	Resp	1	1	0	0	0	0	0	0	0
Rash on neck	Rash	0	0	0	1	0	1	0	0	0
Sore neck and difficulty breathing	Resp	1	1	0	1	0	0	0	1	0
Just plain ugly	Other	0	0	1	0	1	0	0	0	1

Learning a Bayesian Classifier

1. Before deployment of classifier, get labeled training data
2. Learn parameters (conditionals, and priors)

	EXPERT SAYS	breath	difficulty	just	neck	plain	rash	short	sore	ugly
Shortness of breath	Resp	1	0	0	0	0	0	1	0	0
Difficulty breathing	Resp	1	1	0	0	0	0	0	0	0
Rash on neck	Rash	0	0	0	1	0	1	0	0	0
Sore neck and difficulty breathing	Resp	1	1	0	1	0	0	0	1	0
Just plain ugly	Other	0	0	1	0	1	0	0	0	1

P(Prod'm=GI) = ___		P(Prod'm=respir) = ___	…	P(Prod'm=const) = ___
P(angry | Prod'm=GI) = ___		P(angry | Prod'm=respir) = ___	…	P(angry | Prod'm=const) = ___
P(~angry | Prod'm=GI) = ___		P(~angry | Prod'm=respir) = ___	…	P(~angry | Prod'm=const) = ___
P(blood | Prod'm=GI) = ___		P(blood | Prod'm=respir) = ___	…	P(blood | Prod'm=const) = ___
P(~blood | Prod'm=GI) = ___		P(~blood | Prod'm=respir) = ___	…	P(~blood | Prod'm=const) = ___
:		:		:
P(vomit | Prod'm=GI) = ___		P(vomit | Prod'm=respir) = ___	…	P(vomit | Prod'm=const) = ___
P(~vomit | Prod'm=GI) = ___		P(~vomit |Prod'm=respir) = ___	…	P(~vomit | Prod'm=const) = ___

Learning a Bayesian Classifier

1. Before deployment of classifier, get labeled training data
2. Learn parameters (conditionals, and priors)

	EXPERT SAYS	breath	difficulty	just	neck	plain	rash	short	sore	ugly
Shortness of breath	Resp	1	0	0	0	0	0	1	0	0
Difficulty breathing	Resp	1	1	0	0	0	0	0	0	0
Rash on neck	Rash	0	0	0	1	0	1	0	0	0
Sore neck and difficulty breathing	Resp	1	1	0	1	0	0	0	1	0
Just plain ugly	Other	0	0	1	0	1	0	0	0	1

P(Prod'm=GI) = ___	P(Prod'm=respir) = ___	...	P(Prod'm=const) = ___
P(angry | Prod'm=	P(angry | Prod'm=respir) = ___	...	P(angry | Prod'm=const) = ___
P(~angry | Prod'	P(~angry | Prod'm=respir) = ___	...	P(~angry | Prod'm=const) = ___
P(blood | Pro	P(blood | Prod'm=respir) = ___	...	P(blood | Prod'm=const) = ___
P(~blood | P	P(~blood | Prod'm=respir) = ___	...	P(~blood | Prod'm=const) = ___
	:		:
P(vomit	omit | Prod'm=respir) = ___	...	P(vomit | Prod'm=const) = ___
P(~vom	Prod'm=respir) = ___	...	P(~vomit | Prod'm=const) = ___

$$P(\text{breath} = 1 \mid \text{prodrome} = \text{Resp}) = \frac{\text{num "resp" training records containing "breath"}}{\text{num "resp" training records}}$$

Learning a Bayesian Classifier

1. Before depl
2. Learn para

$$P(\text{prodrome} = \text{Resp}) = \frac{\text{num "resp" training records}}{\text{total num training records}}$$

Shortness of breath
Difficulty breathing
Rash on neck
Sore neck and difficulty breathing
Just plain ugly

ERT SAYS	breath	difficulty	just	neck	plain	rash	short	sore	ugly
sp	1	0	0	0	0	0	1	0	0
esp	1	1	0	0	0	0	0	0	0
Rash	0	0	0	1	0	1	0	0	0
Resp	1	1	0	1	0	0	0	1	0
Other	0	0	1	0	1	0	0	0	1

P(Prod'm=GI) = ___	P(Prod'm=respir) = ___	…	P(Prod'm=const) = ___
P(angry | Prod'm=GI) = ___	P(angry | Prod'm=respir) = ___	…	P(angry | Prod'm=const) = ___
P(~angry | Prod' = ___	P(~angry | Prod'm=respir) = ___	…	P(~angry | Prod'm=const) = ___
P(blood | Pro	P(blood | Prod'm=respir) = ___	…	P(blood | Prod'm=const) = ___
P(~blood | P	P(~blood | Prod'm=respir) = ___	…	P(~blood | Prod'm=const) = ___
	:		:
P(vomit	vomit | Prod'm=respir) = ___	…	P(vomit | Prod'm=const) = ___
P(~vo	Prod'm=respir) = ___	…	P(~vomit | Prod'm=const) = ___

$$P(\text{breath} = 1 \mid \text{prodrome} = \text{Resp}) = \frac{\text{num "resp" training records containing "breath"}}{\text{num "resp" training records}}$$

Learning a Bayesian Classifier

1. Before deployment of classifier, get labeled training data
2. Learn parameters (conditionals, and priors)

	EXPERT SAYS	breath	difficulty	just	neck	plain	rash	short	sore	ugly
Shortness of breath	Resp	1	0	0	0	0	0	1	0	0
Difficulty breathing	Resp	1	1	0	0	0	0	0	0	0
Rash on neck	Rash	0	0	0	1	0	1	0	0	0
Sore neck and difficulty breathing	Resp	1	1	0	1	0	0	0	1	0
Just plain ugly	Other	0	0	1	0	1	0	0	0	1

P(Prod'm=GI) = ___		P(Prod'm=respir) = ___	…	P(Prod'm=const) = ___
P(angry | Prod'm=GI) = ___		P(angry | Prod'm=respir) = ___	…	P(angry | Prod'm=const) = ___
P(~angry | Prod'm=GI) = ___		P(~angry | Prod'm=respir) = ___	…	P(~angry | Prod'm=const) = ___
P(blood | Prod'm=GI) = ___		P(blood | Prod'm=respir) = ___	…	P(blood | Prod'm=const) = ___
P(~blood | Prod'm=GI) = ___		P(~blood | Prod'm=respir) = ___	…	P(~blood | Prod'm=const) = ___
:		:		:
P(vomit | Prod'm=GI) = ___		P(vomit | Prod'm=respir) = ___	…	P(vomit | Prod'm=const) = ___
P(~vomit | Prod'm=GI) = ___		P(~vomit |Prod'm=respir) = ___	…	P(~vomit | Prod'm=const) = ___

Learning a Bayesian Classifier

1. Before deployment of classifier, get labeled training data
2. Learn parameters (conditionals, and priors)
3. During deployment, apply classifier

	EXPERT SAYS	breath	difficulty	just	neck	plain	rash	short	sore	ugly
Shortness of breath	Resp	1	0	0	0	0	0	1	0	0
Difficulty breathing	Resp	1	1	0	0	0	0	0	0	0
Rash on neck	Rash	0	0	0	1	0	1	0	0	0
Sore neck and difficulty breathing	Resp	1	1	0	1	0	0	0	1	0
Just plain ugly	Other	0	0	1	0	1	0	0	0	1

P(Prod'm=GI) = ___		P(Prod'm=respir) = ___	...	P(Prod'm=const) = ___
P(angry | Prod'm=GI) = ___		P(angry | Prod'm=respir) = ___	...	P(angry | Prod'm=const) = ___
P(~angry | Prod'm=GI) = ___		P(~angry | Prod'm=respir) = ___	...	P(~angry | Prod'm=const) = ___
P(blood | Prod'm=GI) = ___		P(blood | Prod'm=respir) = ___	...	P(blood | Prod'm=const) = ___
P(~blood | Prod'm=GI) = ___		P(~blood | Prod'm=respir) = ___	...	P(~blood | Prod'm=const) = ___
:		:		:
P(vomit | Prod'm=GI) = ___		P(vomit | Prod'm=respir) = ___	...	P(vomit | Prod'm=const) = ___
P(~vomit | Prod'm=GI) = ___		P(~vomit |Prod'm=respir) = ___	...	P(~vomit | Prod'm=const) = ___

Learning a Bayesian Classifier

1. Before deployment of classifier, get labeled training data
2. Learn parameters (conditionals, and priors)
3. During deployment, apply classifier

	EXPERT SAYS	breath	difficulty	just	neck	plain	rash	short	sore	ugly
Shortness of breath	Resp	1	0	0	0	0	0	1	0	0
Difficulty breathing	Resp	1	1	0	0	0	0	0	0	0
Rash on neck	Rash	0	0	0	1	0	1	0	0	0
Sore neck and difficulty breathing	Resp	1	1	0	1	0	0	0	1	0
Just plain ugly	Other	0	0	1	0	1	0	0	0	1

P(Prod'm=GI) = ___		P(Prod'm=respir) = ___	…	P(Prod'm=const) = ___
P(angry | Prod'm=GI) = ___		P(angry | Prod'm=respir) = ___	…	P(angry | Prod'm=const) = ___
P(~angry | Prod'm=GI) = ___		P(~angry | Prod'm=respir) = ___	…	P(~angry | Prod'm=const) = ___
P(blood | Prod'm=GI) = ___		P(blood | Prod'm=respir) = ___	…	P(blood | Prod'm=const) = ___
P(~blood | Prod'm=GI) = ___		P(~blood | Prod'm=respir) = ___	…	P(~blood | Prod'm=const) = ___
:		:		:
P(vomit | Prod'm=GI) = ___		P(vomit | Prod'm=respir) = ___	…	P(vomit | Prod'm=const) = ___
P(~vomit | Prod'm=GI) = ___		P(~vomit |Prod'm=respir) = ___	…	P(~vomit | Prod'm=const) = ___

New Chief Complaint: “Just sore breath”

Learning a Bayesian Classifier

1. Before deployment of classifier, get labeled training data
2. Learn parameters (conditionals, and priors)
3. During deployment, apply classifier

	EXPERT SAYS	breath	difficulty	just	neck	plain	rash	short	sore	ugly
Shortness of breath	Resp	1	0	0	0	0	0	1	0	0
Difficulty breathing	Resp	1	1	0	0	0	0	0	0	0
Rash on neck	Rash	0	0	0	1	0	1	0	0	0
Sore neck and difficulty breathing	Resp	1	1	0	1	0	0	0	1	0
Just plain ugly	Other	0	0	1	0	1	0	0	0	1

P(Prod'm=GI) = ___		P(Prod'm=respir) = ___	...	P(Prod'm=const) = ___
P(angry | Prod'm=GI) = ___		P(angry | Prod'm=respir) = ___	...	P(angry | Prod'm=const) = ___
P(~angry | Prod'm=GI) = ___		P(~angry | Prod'm=respir) = ___	...	P(~angry | Prod'm=const) = ___
P(blood | Prod'm=GI) = ___		P(blood | Prod'm=respir) = ___	...	P(blood | Prod'm=const) = ___
P(~blood | Prod'm=GI) = ___		P(~blood | Prod'm=respir) = ___	...	P(~blood | Prod'm=const) = ___
:		:		:
P(vomit | Prod'm=GI) = ___		P(vomit | Prod'm=respir) = ___	...	P(vomit | Prod'm=const) = ___
P(~vomit | Prod'm=GI) = ___		P(~vomit |Prod'm=respir) = ___	...	P(~vomit | Prod'm=const) = ___

New Chief Complaint: "Just sore breath"

$$P(\text{prodrome} = \text{GI} \mid \text{breath} = 1, \text{difficulty} = 0, \text{just} = 1, \ldots)$$

$$= \frac{P(\text{breath} = 1 \mid \text{prod} = \text{GI}) \times P(\text{difficulty} = 0 \mid \text{prod} = \text{GI}) \ldots \times P(\text{state} = GI\ )}{\sum_Z P(\text{breath} = 1 \mid \text{prod} = \text{Z}) \times P(\text{difficulty} = 0 \mid \text{prod} = \text{Z}) \ldots \times P(\text{state} = Z\ )}$$

CoCo Performance (AUC scores)

- Botulism 0.78
- rash, 0.91
- neurological 0.92
- hemorrhagic, 0.93;
- constitutional 0.93
- gastrointestinal 0.95
- other, 0.96
- respiratory 0.96

Conclusion

- Automated text extraction is increasingly important
- There is a very wide world of text extraction outside Biosurveillance
- The field has changed very fast, even in the past three years.
- Warning, although Bayes Classifiers are simplest to implement, Logistic Regression or other discriminative methods often learn more accurately. Consider using off the shelf methods, such as William Cohen's successful "minor third" open-source libraries: http://minorthird.sourceforge.net/
- Real systems (including CoCo) have many ingenious special-case improvements.

Discussion

1. What new data sources should we apply algorithms to?
    1. EG Self-reporting?
2. What are related surveillance problems to which these kinds of algorithms can be applied?
3. Where are the gaps in the current algorithms world?
4. Are there other spatial problems out there?
5. Could new or pre-existing algorithms help in the period after an outbreak is detected?
6. Other comments about favorite tools of the trade.