SENTIMENT ANALYSIS

OPINION MINING & EMOTION ANALYSIS

Sentiment Analysis, aka Opinion Mining

Definition: processing a text in order to identify and extract its subjective information

- Although an opinion is a belief, while a sentiment is a feeling!
- Not an easy job even for humans, two people disagree on the sentiment expressed a text!

Motivation:

- E.g., Online Marketing → Customer Feedback, ...
- E.g., the age and geographic differences in the levels of happiness were analyzed as well. *[Dodds and Danforth, 2010].*

Peter Sheridan Dodds and Christopher M Danforth. Measuring the happiness of large-scale written expression: Songs, blogs, and presidents. Journal of Happiness Studies, 11(4):441–456, 2010. DOI: 10.1007/s10902-009-9150-9. 47

DOCUMENT CLASSIFICATION

SENTIMENT ANALYSIS & DOCUMENT CLASSIFICATION

Tasks:

1. Identify the *overall polarity* of a document {positive, negative} → Binary Classes
2. Identify the *overall polarity* of a document {positive, negative, neutral} → Multiclass
3. The *polarity magnitude*, e.g., p ∈ $\boldsymbol{N}^{[1,5]}$ stars for reviews → Multiclass
4. The *polarity magnitude*, e.g., p ∈ $\boldsymbol{R}^{[1,5]}$ → Regression
5. A document may be about different topics/parts of a concept/product
    1. Aspect Detection
    2. Polarity of each aspects

"like the location but poor quality" → Hotel(location: +, quality: -)
"*food was great but the service was poor.*" → Hotel(food: +, service: -)

[Pang and Lee, 2008]. Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. Foundations and trends in information retrieval, 2(1-2):1–135, 2008. DOI: 10.1561/1500000011. 47

SENTIMENT ANALYSIS & DOCUMENT CLASSIFICATION

Tasks:

1. Identify the overall polarity of a document {positive, negative} → Binary Classes
2. Identify the overall polarity of a document {positive, negative, neutral} → Multiclass
3. The polarity magnitude, e.g., p $\in \boldsymbol{N}^{[1,5]}$ stars for reviews → Multiclass
4. The polarity magnitude, e.g., p $\in \boldsymbol{R}^{[1,5]}$ → Regression
5. A document may be about different topics/parts of a concept/product
   1. Aspect Detection
   2. Polarity of each aspects

"like the location but poor quality" → Hotel(location: +, quality: -)
"*food was great but the service was poor.*" → Hotel(food: +, service: -)

[Pang and Lee, 2008]. Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. Foundations and trends in information retrieval, 2(1-2):1–135, 2008. DOI: 10.1561/1500000011. 47

APPROACHES

lexicon-based

Avoids statistical learning altogether!

Classify documents by counting words against positive and negative sentiment word lists(Taboada et al., 2011).

- General Inquirer (Stone et al., 1966)
- LIWC (Pennebaker et al., 2007)
- MPQA Subjectivity Lexicon (Wilson et al., 2005).

E.g., the MPQA has 6885 words

- 2718 positive
- 4912 negative
- Each marked for whether it is strongly or weakly biased

Positive : admirable, beautiful, confident, dazzling, ecstatic, favor, glee, great
Negative: awful, bad, bias, catastrophe, cheat, deny, envious, foul, harsh, hate

lexicon-based

Avoids statistical learning altogether!

Classify documents by counting words against positive and negative sentiment word lists(Taboada et al., 2011).

- Irrealis mood (https://en.wikipedia.org/wiki/Irrealis_mood)

"It would be nice if you acted like you understood."

- Negation

"That's not bad for the first day."
"This is not the worst thing that can happen."
"Disturbingly good"

- Sarcasm & Irony

"That's just what I needed today!"
"Nice perfume. How long did you marinate in it?"

Learning to Classify: Classifiers

Bayesian Classifier

Naïve Bayes

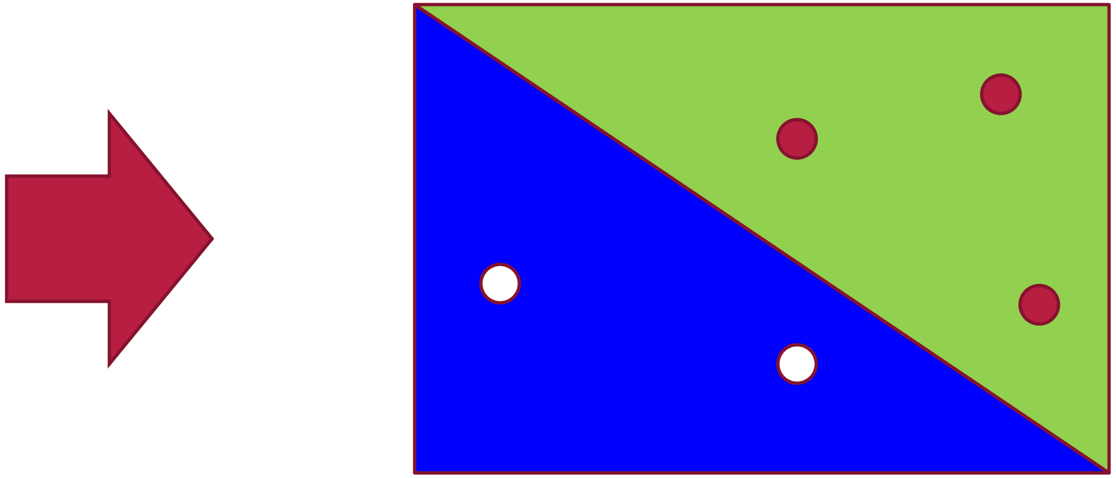
Logistic Regression (Linear)

Decision Tree → Ensemble Random Forest → Boosted Tree

Neural Network (non-Linear)

Learning to Classify: Classifiers

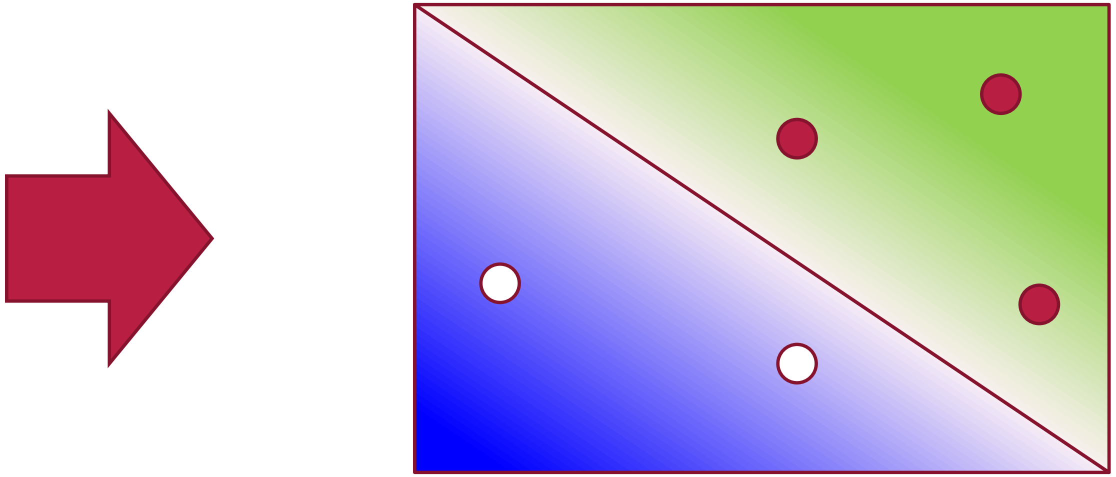
Tokens
Sentences
Documents



f: D $\longrightarrow$ {0,1} : f(x) a Boolean function

Learning to Classify: Classifiers

Tokens
Sentences
Documents

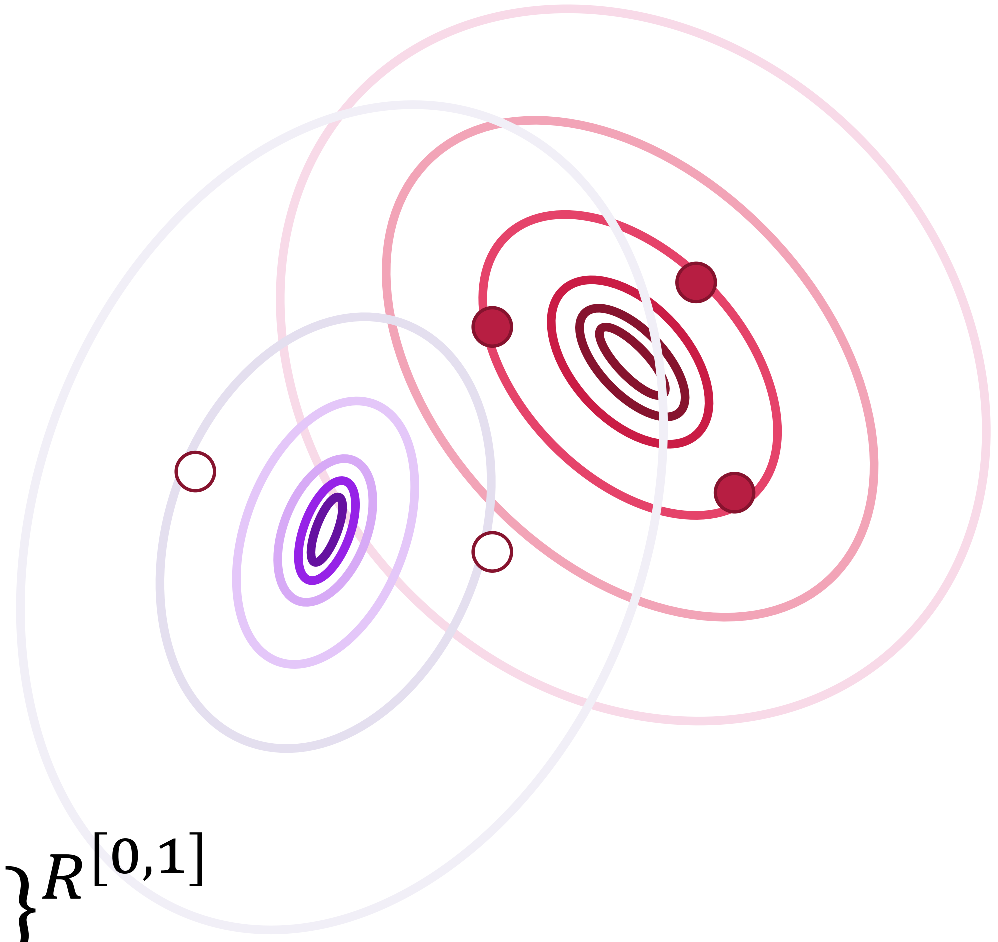


f: D $\longrightarrow$ R$^{[0,1]}$: f(x) probability value

f: D $\longrightarrow$ R$^{[-1,1]}$: f(x) confidence score

Learning to Classify: Classifiers

Tokens
Sentences
Documents



f: D $\rightarrow \{0,1\}^{R^{[0,1]}}$

mixture or overlapping classification

Bayesian Inference

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)} = \frac{P(E|H)P(H)}{\sum_{H\in\mathcal{H}} P(E|H)\,P(H)}$$

$$E: Event\ |Evidence$$
$$H: Hopothesis$$

Posterior: Evidence → Hypothesis

This is what we want to know! the probability of a hypothesis given the observed evidence

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)} = \frac{P(E|H)P(H)}{\sum_{H \in \mathcal{H}} P(E|H)\,P(H)}$$

$$E: Event|Evidence$$
$$H: Hopothesis$$

Prior: The Chance of Hypothesis H

is the estimate of the probability of the hypothesis H before E, the current evidence, is observed

$$P(H|E) = \frac{P(E|H)\boxed{P(H)}}{P(E)} = \frac{P(E|H)\boxed{P(H)}}{\sum_{H \in \mathcal{H}} P(E|H)\, P(H)}$$

$$E: Event|Evidence$$
$$H: Hopothesis$$

You tell me?

$$P(H|E) = \frac{\boxed{P(E|H)}P(H)}{P(E)} = \frac{\boxed{P(E|H)}P(H)}{\sum_{H \in \mathcal{H}} P(E|H)\,P(H)}$$

$$E: Event|Evidence$$
$$H: Hopothesis$$

Likelihood: Hypothesis → Evidence

$$P(H|E) = \frac{\boxed{P(E|H)}P(H)}{P(E)} = \frac{\boxed{P(E|H)}P(H)}{\sum_{H \in \mathcal{H}} P(E|H)\, P(H)}$$

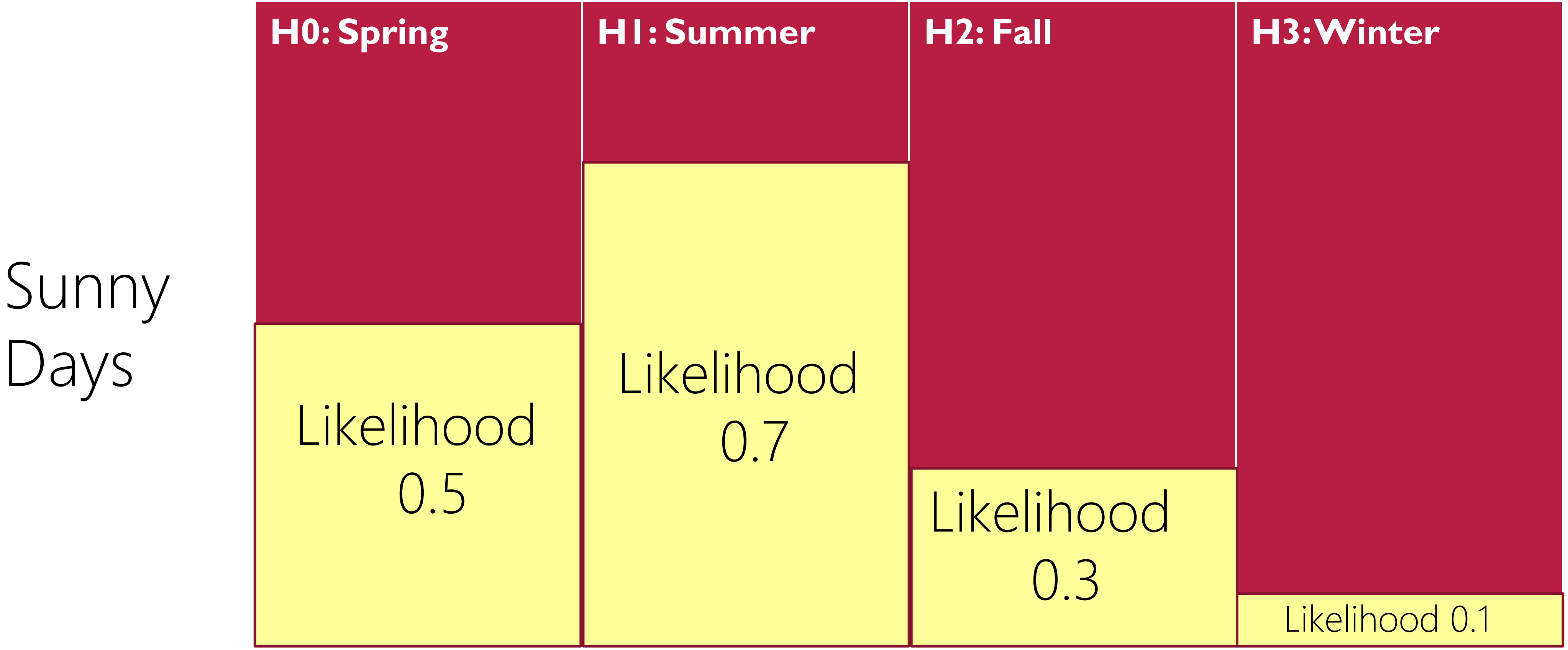
$$E: Event|Evidence$$
$$H: Hopothesis$$

Marginal Likelihood: Normalize over all H

$$P(H|E) = \frac{P(E|H)P(H)}{\boxed{P(E)}} = \frac{P(E|H)P(H)}{\boxed{\sum_{H\in\mathcal{H}} P(E|H)\,P(H)}}$$

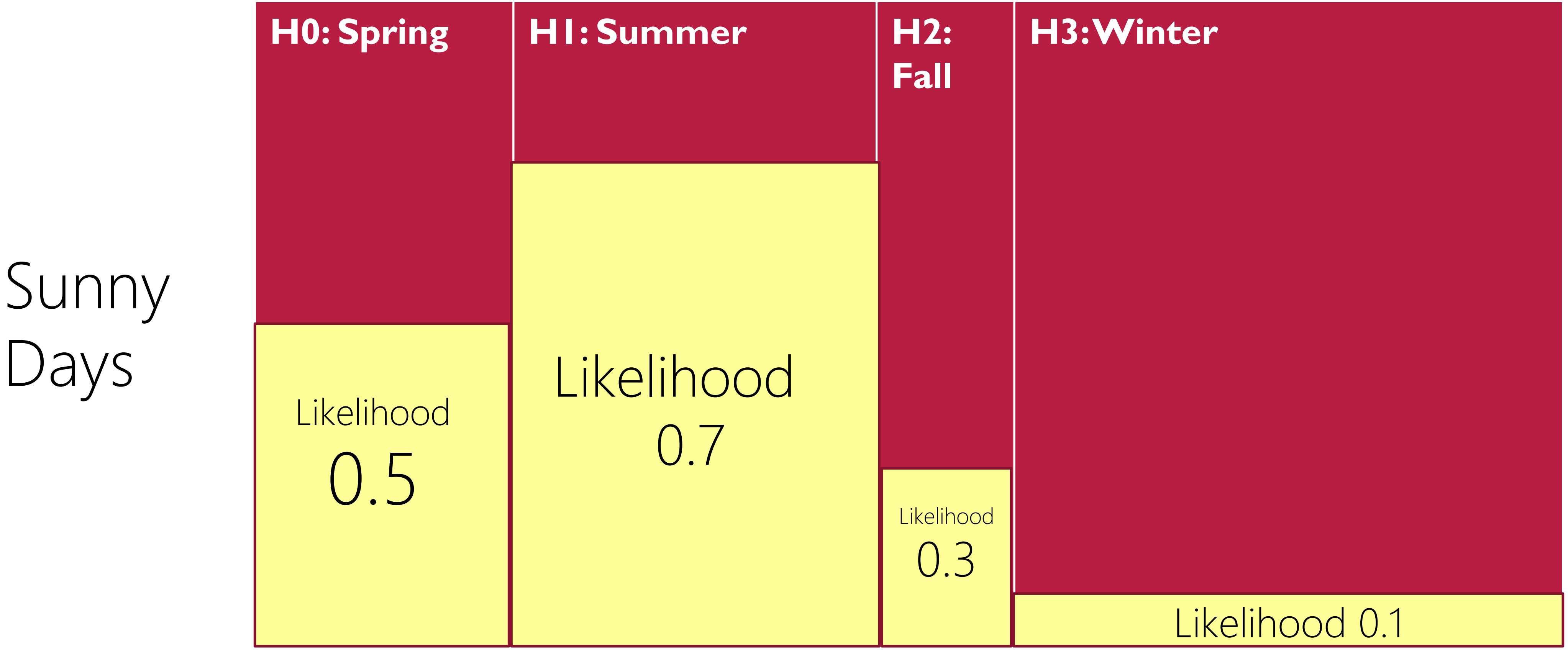
$$E: Event|Evidence$$
$$H: Hopothesis$$

Example: Hypothesis in Iran



Priors = $P(H_i)$ = 1/4

Example: Hypothesis in Canada



Priors = $P(H_i)$ → Not uniform! → $P(H_i)$ = [0.25, 0.3,0.05, 0.4]

Example: Evidence → Hypothesis

We see a sunny day in Iran! What is the season?

$P(Spring|Sunny) = ?$

$P(Summer|Sunny) = ?$

$P(Fall|Sunny) = ?$

$P(Winter|Sunny) = ?$

Example: Evidence → Hypothesis

We see a sunny day in Iran! What is the season?

$$P(\text{Spring}|\text{Sunny}) = \frac{P(\text{sunny}|H0)P(H0)}{P(s)} = \frac{P(H0)}{\text{P(sunny)}} \; 0.5$$

$$P(\text{Summer}|\text{Sunny}) = \frac{P(\text{sunny}|H1)P(H1)}{P(\text{sunny})} = \frac{P(H1)}{\text{P(sunny)}} \; 0.7$$

$$P(\text{Fall}|\text{Sunny}) = \frac{P(\text{sunny}|H2)P(H2)}{P(\text{sunny})} = \frac{P(H2)}{\text{P(sunny)}} \; 0.3$$

$$P(\text{Winter}|\text{Sunny}) = \frac{P(\text{sunny}|H3)P(H3)}{P(\text{sunny})} = \frac{P(H3)}{\text{P(sunny)}} \; 0.1$$

Maximum posterior: argmax $P(H_i|\text{Sunny})$

Example: Evidence → Hypothesis

We see a sunny day in Canada! What is the season?

$$P(\text{Spring}|\text{Sunny}) = \frac{P(\text{sunny}|H0)P(H0)}{P(s)} = \frac{1}{P(\text{sunny})}P(H0) \times \mathbf{0.5} = \frac{1}{P(\text{sunny})}0.25\times 0.5$$

$$P(\text{Summer}|\text{Sunny}) = \frac{P(\text{sunny}|H1)P(H1)}{P(\text{sunny})} = \frac{1}{P(\text{sunny})}P(H1) \times \mathbf{0.7} = \frac{1}{P(\text{sunny})}0.3\times 0.7$$

$$P(\text{Fall}|\text{Sunny}) = \frac{P(\text{sunny}|H2)P(H2)}{P(\text{sunny})} = \frac{1}{P(\text{sunny})}P(H2) \times \mathbf{0.3} = \frac{1}{P(\text{sunny})}0.05\times 0.3$$

$$P(\text{Winter}|\text{Sunny}) = \frac{P(\text{sunny}|H3)P(H3)}{P(\text{sunny})} = \frac{1}{P(\text{sunny})}P(H3) \times \mathbf{0.1} = \frac{1}{P(\text{sunny})}0.4\times 0.1$$

Maximum posterior: argmax $P(H_i|\text{Sunny})$

MAP (maximum a posteriori) vs. MLE (maximum likelihood estimation)

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)} = \frac{P(E|H)P(H)}{\sum_{H\in\mathcal{H}} P(E|H)\,P(H)}$$

$$E: Event\ |Evidence$$
$$H: Hopothesis$$

If Priors are all uniformly distributed, MAP == MLE, e.g., weights of a neural net!

However, priors are not uniform and different, MLE is weighted by Prior.

So, MAP is more general optimization method!

Both are point estimates, i.e., give single real values as probabilities!

MAP (maximum a posteriori) vs. MLE (maximum likelihood estimation)

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)} = \frac{P(E|H)P(H)}{\sum_{H\in\mathcal{H}} P(E|H)\, P(H)}$$

$$E: Event\ |Evidence$$
$$H: Hopothesis$$

If all Ps in the above formula has probability density function:

P(a<x<b)=$\int_a^b f(x)dx$

Then we have Bayesian Inference (Advanced):

the output is not a single value but a probability density function (when H is a continuous variable) or a probability mass function (when H is a discrete variable).

Classification

Tokens
Sentences
Documents

Inputs → Events | Evidences

Classes | Labels

Outputs → Hypotheses

Bayes

Document → Classes

$$\hat{c} = \underset{c \in C}{argmax}\, P(c|d) = \underset{c \in C}{argmax} \frac{P(d|c)P(c)}{P(d)} = \underset{c \in C}{argmax}\, P(d|c)P(c)$$

Bayes

Document → Classes

$$\hat{c} = \underset{c \in C}{argmax}\, P(w_1 w_2 \ldots w_{|d|} | c) P(c) = \underset{c \in C}{argmax} \begin{bmatrix} Unigram \\ Bigram \\ Trigram \end{bmatrix} P(c)$$

$$\ldots$$

Naïve Bayes

Document → Classes

$$\hat{c} = \underset{c \in C}{argmax} P(w_1 w_2 \ldots w_{|d|} | c) P(c) = \underset{c \in C}{argmax} [unigram] \, P(c)$$

Naïve Bayes

Document → Classes

$$\hat{c} = \underset{c \in C}{argmax}\, [unigram]\, P(c) = \underset{c \in C}{argmax} \prod_{i=1}^{n} P(w_i|c)\, P(c) = \underset{c \in C}{argmax}\, P(c) \prod_{i=1}^{n} P(w_i|c)$$

Naïve Bayes

Document → Classes

$$\hat{c} = \underset{c \in C}{argmax}\, [unigram]\, P(c) = \underset{c \in C}{argmax} \prod_{i=1}^{n} P(w_i|c)\, P(c) = \underset{c \in C}{argmax}\, P(c) \prod_{i=1}^{n} P(w_i|c)$$

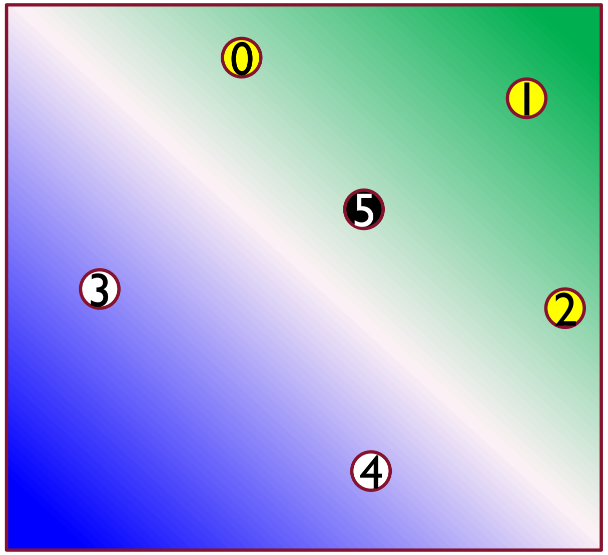
$$\hat{c} = \underset{c \in C}{argmax} \log([unigram]\, P(c)) = \underset{c \in C}{argmax}\, log \prod_{i=1}^{n} P(w_i|c)\, P(c) = \underset{c \in C}{argmax} \log P(c) + log \sum_{i=1}^{n} P(w_i|c)$$

Naïve Bayes: Example

	Documents (Sentence)	Class (Label)
Train	just plain boring	Neg (-)
	entirely predictable and lacks energy	Neg (-)
	no surprises and very few laughs	Neg (-)
	very powerful	Pos (+)
	the most fun film of the summer	Pos (+)
Test	predictable with no fun	?

Naïve Bayes: Example

	Documents (Sentence)	Class (Label)
Train	S_0: just plain boring	Neg (-)
	S_1: entirely predictable and lacks energy	Neg (-)
	S_2: no surprises and very few laughs	Neg (-)
	S_3: very powerful	Pos (+)
	S_4: the most fun film of the summer	Pos (+)
Test	S_5: predictable with no fun	= max(P(- | Test), P(+ | Test))

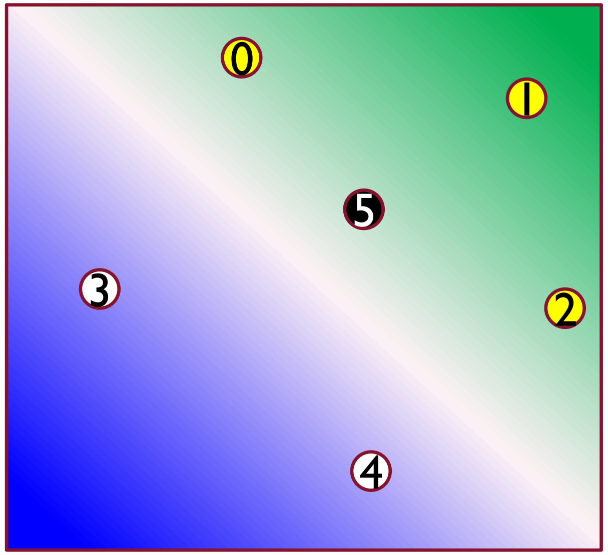


$$P(-|\text{ Test}) = \frac{1}{P(Test)} P(\text{Test}|-)\, P(-) \propto P(-)\, P(\text{predictable}|-) P(\text{with}|-)\, P(\text{no}|-)\, P(\text{fun}|-) = ?$$

$$P(+|\text{ Test}) = \frac{1}{P(Test)} P(\text{Test}|+)\, P(+) \propto P(+)\, P(\text{predictable}|+) P(\text{with}|+)\, P(\text{no}|+)\, P(\text{fun}|+) = ?$$

Naïve Bayes: Example

	Documents (Sentence)	Class (Label)
Train	S_0: just plain boring	Neg (-)
	S_1: entirely predictable and lacks energy	Neg (-)
	S_2: no surprises and very few laughs	Neg (-)
	S_3: very powerful	Pos (+)
	S_4: the most fun film of the summer	Pos (+)
Test	S_5: predictable with no fun	= max(P(- | Test), P(+ | Test))



$$P(-) = \frac{3}{5}$$

$$P(+) = \frac{2}{5}$$

$$P(-|\text{ Test}) = \frac{1}{P(Test)} P(\text{Test}|-)\, P(-) \propto P(-)\, P(\text{predictable}|-) P(\text{with}|-)\, P(\text{no}|-)\, P(\text{fun}|-) = ?$$

$$P(+|\text{ Test}) = \frac{1}{P(Test)} P(\text{Test}|+)\, P(+) \propto P(+)\, P(\text{predictable}|+) P(\text{with}|+)\, P(\text{no}|+)\, P(\text{fun}|+) = ?$$

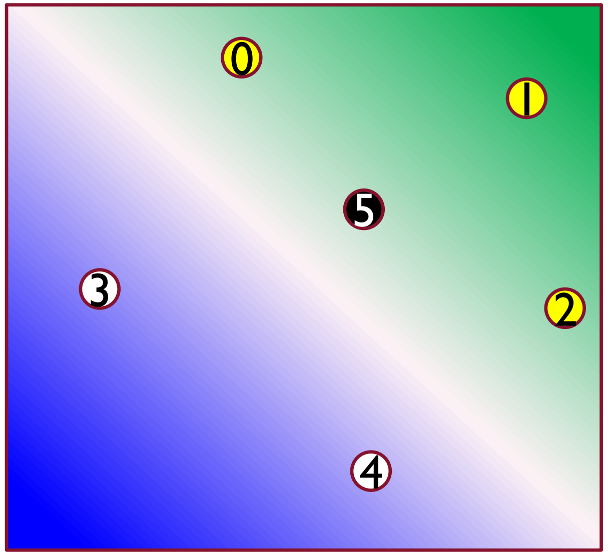
Neg (-) : just plain boring entirely predictable and lacks energy no surprises and very few laughs
Pos (+) : very powerful the most fun film of the summer

$$P(\text{predictable}|-) = \frac{\#preditable}{\#total\ tokens\ in\ (-)} = \frac{1}{14};\ P(\text{with}|-) = \frac{\#with}{\#total\ tokens\ in\ (-)} = \frac{0}{14};\ P(\text{no}|-) = \frac{\#no}{\#total\ tokens\ in\ (-)} = \frac{1}{14};\ P(\text{fun}|-) = \frac{\#fun}{\#total\ tokens\ in\ (-)} = \frac{0}{14}$$

$$P(\text{predictable}|+) = \frac{\#preditable}{\#total\ tokens\ in\ (+)} = \frac{0}{9};\ P(\text{with}|+) = \frac{\#with}{\#total\ tokens\ in\ (+)} = \frac{0}{9};\ P(\text{no}|+) = \frac{\#no}{\#total\ tokens\ in\ (+)} = \frac{0}{9};\ P(\text{fun}|+) = \frac{\#fun}{\#total\ tokens\ in\ (+)} = \frac{1}{9}$$

Naïve Bayes + Smoothing: Example

	Documents (Sentence)	Class (Label)
Train	S_0: just plain boring	Neg (-)
	S_1: entirely predictable and lacks energy	Neg (-)
	S_2: no surprises and very few laughs	Neg (-)
	S_3: very powerful	Pos (+)
	S_4: the most fun film of the summer	Pos (+)
Test	S_5: predictable with no fun	= max(P(- | Test), P(+ | Test))



$$P(-) = \frac{3}{5}$$

$$P(+) = \frac{2}{5}$$

$$P(-|\text{ Test}) = \frac{1}{P(Test)} P(\text{Test}|-)\, P(-) \propto P(-)\, P(\text{predictable}|-) P(\text{with}|-)\, P(\text{no}|-)\, P(\text{fun}|-) = ?$$

$$P(+|\text{ Test}) = \frac{1}{P(Test)} P(\text{Test}|+)\, P(+) \propto P(+)\, P(\text{predictable}|+) P(\text{with}|+)\, P(\text{no}|+)\, P(\text{fun}|+) = ?$$

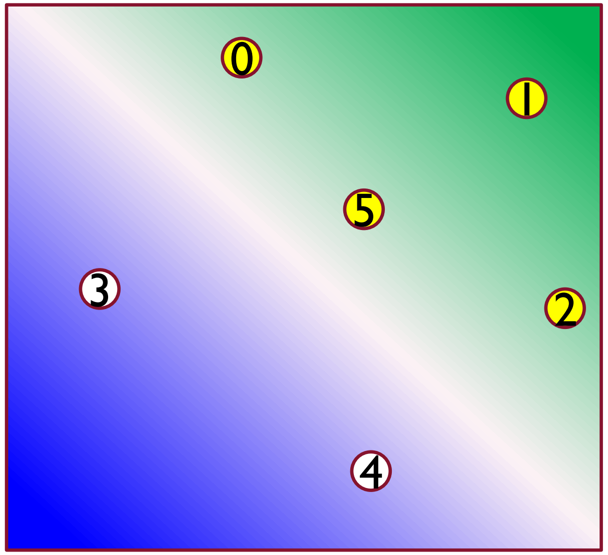
Neg (-) : just plain boring entirely predictable and lacks energy no surprises and very few laughs
Pos (+) : very powerful the most fun film of the summer

$$P(\text{predictable}|-) = \frac{\#preditable}{\#total\ tokens\ in\ (-)} = \frac{1+1}{14+23};\ P(\text{with}|-) = \frac{\#with}{\#total\ tokens\ in\ (-)} = \frac{0+1}{14+23};\ P(\text{no}|-) = \frac{\#no}{\#total\ tokens\ in\ (-)} = \frac{1+1}{14+23};\ P(\text{fun}|-) = \frac{\#fun}{\#total\ tokens\ in\ (-)} = \frac{0+1}{14+23}$$

$$P(\text{predictable}|+) = \frac{\#preditable}{\#total\ tokens\ in\ (+)} = \frac{0+1}{9+23};\ P(\text{with}|+) = \frac{\#with}{\#total\ tokens\ in\ (+)} = \frac{0+1}{9+23};\ P(\text{no}|+) = \frac{\#no}{\#total\ tokens\ in\ (+)} = \frac{0+1}{9+23};\ P(\text{fun}|+) = \frac{\#fun}{\#total\ tokens\ in\ (+)} = \frac{1+1}{9+23}$$

Naïve Bayes + Smoothing: Example

	Documents (Sentence)	Class (Label)
Train	S_0: just plain boring	Neg (-)
	S_1: entirely predictable and lacks energy	Neg (-)
	S_2: no surprises and very few laughs	Neg (-)
	S_3: very powerful	Pos (+)
	S_4: the most fun film of the summer	Pos (+)
Test	S_5: predictable with no fun	= max(P(- | Test), P(+ | Test))



$$P(-) = \frac{3}{5}$$

$$P(+) = \frac{2}{5}$$

$$P(-|\text{ Test}) = \frac{1}{P(Test)} P(\text{Test}|-)\ P(-) \propto P(-)\ P(\text{predictable}|-)P(\text{with}|-)\ P(\text{no}|-)\ P(\text{fun}|-) = \frac{3}{5}\times\frac{2}{37}\times\frac{1}{37}\times\frac{2}{37}\times\frac{1}{37} = 0.00000128057$$

$$P(+|\text{ Test}) = \frac{1}{P(Test)} P(\text{Test}|+)\ P(+) \propto P(+)\ P(\text{predictable}|+)P(\text{with}|+)\ P(\text{no}|+)\ P(\text{fun}|+) = \frac{2}{5}\times\frac{1}{32}\times\frac{1}{32}\times\frac{1}{32}\times\frac{2}{32} = 0.00000762939453$$

Neg (-) : just plain boring entirely predictable and lacks energy no surprises and very few laughs
Pos (+) : very powerful the most fun film of the summer

$$P(\text{predictable}|-) = \frac{\#preditable}{\#total\ tokens\ in\ (-)} = \frac{2}{37};\ P(\text{with}|-) = \frac{\#with}{\#total\ tokens\ in\ (-)} = \frac{1}{37};\ P(\text{no}|-) = \frac{\#no}{\#total\ tokens\ in\ (-)} = \frac{2}{37};\ P(\text{fun}|-) = \frac{\#fun}{\#total\ tokens\ in\ (-)} = \frac{1}{37}$$

$$P(\text{predictable}|+) = \frac{\#preditable}{\#total\ tokens\ in\ (+)} = \frac{1}{32};\ P(\text{with}|+) = \frac{\#with}{\#total\ tokens\ in\ (+)} = \frac{1}{32};\ P(\text{no}|+) = \frac{\#no}{\#total\ tokens\ in\ (+)} = \frac{1}{32};\ P(\text{fun}|+) = \frac{\#fun}{\#total\ tokens\ in\ (+)} = \frac{2}{32}$$

DOCUMENT CLASSIFICATION ⇒ Sentiment

Per document, whether a word occurs or not seems to matter more than its frequency
word counts → word occurrence

This variant is called binary NB.

Four original documents:

- – it was pathetic the worst part was the boxing scenes
- – no plot twists or great scenes
- \+ and satire and great plot twists
- \+ great scenes great film

After per-document binarization:

- – it was pathetic the worst part boxing scenes
- – no plot twists or great scenes
- \+ and satire great plot twists
- \+ great scenes film

	NB Counts		Binary Counts	
	+	−	+	−
and	2	0	1	0
boxing	0	1	0	1
film	1	0	1	0
great	3	1	2	1
it	0	1	0	1
no	0	1	0	1
or	0	1	0	1
part	0	1	0	1
pathetic	0	1	0	1
plot	1	1	1	1
satire	1	0	1	0
scenes	1	2	1	2
the	0	2	0	1
twists	1	1	1	1
was	0	2	0	1
worst	0	1	0	1

Figure 4.3 An example of binarization for the binary naive Bayes algorithm.

DOCUMENT CLASSIFICATION ⇒ Sentiment

Negation: during text normalization, prepend the prefix NOT to every word after a token of logical negation (n’t, not, no, never) until the next punctuation mark.

“didn't like this movie , but I” ➔ *“didn't NOT_like NOT_this NOT_movie , but I”*

Statistically,

- Words like *NOT_like*, *NOT_recommend* more in negative
- Words like *NOT_bored*, *NOT_dismiss* more in positive

Use of parsing to deal more accurately with the scope relationship between negation words and the predicates they modify!

Naïve Bayes as LM

See Bengio's LM!
See Mikolov's Word2Vec!

Naïve Bayes as LM

Classify the context into the class of next token!

$$P(w_1 w_2 \ldots wi\ \_\ _1 | w_i) = argmax\ P(wi | w_1 w_2 \ldots wi\ \_\ _1)$$

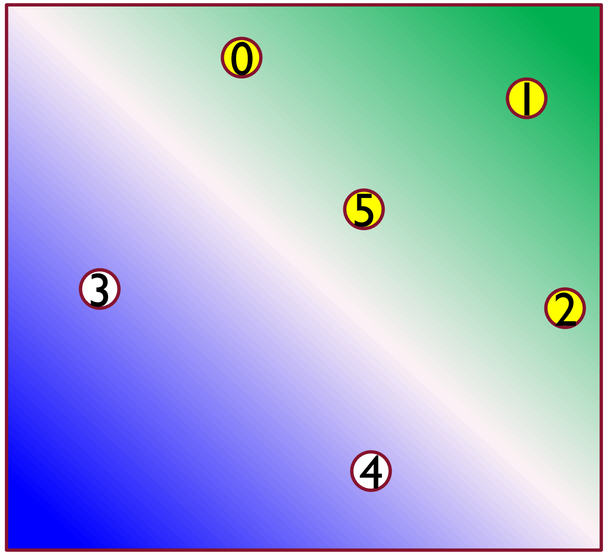
Input Representation

Tokens
Sentences
Documents

Vector Representation
e.g., t-d, t-t, tf-idf, w2v

Naïve Bayes

	Documents (Sentence)	Class (Label)
Train	S_0: just plain boring	Neg (-)
	S_1: entirely predictable and lacks energy	Neg (-)
	S_2: no surprises and very few laughs	Neg (-)
	S_3: very powerful	Pos (+)
	S_4: the most fun film of the summer	Pos (+)
Test	S_5: predictable with no fun	= max(P(- | Test), P(+ | Test))



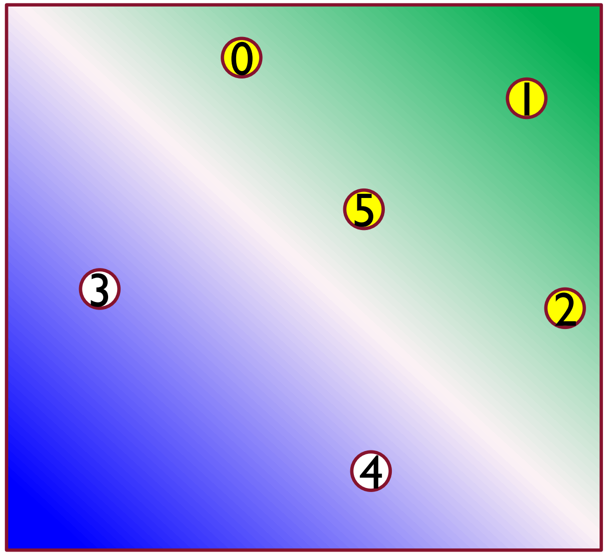
$$P(-) = \frac{3}{5}$$

$$P(+) = \frac{2}{5}$$

	just	plain	boring	entirely	...										the	of	very	fun	...	Class (Label)
Train	1	1	1	0	0	0	0	0	..	..	..	..	0	0	0	0	0	0	0	Neg (-)
	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	Neg (-)
	0	0	0	0	0	0	0	0	0	0	0	1	1	..	..	..	1	0	0	Neg (-)
	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	Pos (+)
	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	1	0	Pos (+)
Test	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	= max(P(- | Test), P(+ | Test))

Naïve Bayes: Features

	Documents (Sentence)	Class (Label)
Train	S_0: just plain boring	Neg (-)
	S_1: entirely predictable and lacks energy	Neg (-)
	S_2: no surprises and very few laughs	Neg (-)
	S_3: very powerful	Pos (+)
	S_4: the most fun film of the summer	Pos (+)
Test	S_5: predictable with no fun	= max(P(- | Test), P(+ | Test))



$$P(-) = \frac{3}{5}$$

$$P(+) = \frac{2}{5}$$

	F1	F2	F3	F4	F5, …,														F_N	Class (Label)
Train	1	1	1	0	0	0	0	0	..	..	..	..	0	0	0	0	0	0	0	Neg (-)
	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	Neg (-)
	0	0	0	0	0	0	0	0	0	0	0	1	1	..	..	..	1	0	0	Neg (-)
	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	Pos (+)
	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	1	0	Pos (+)
Test	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	= max(P(- | Test), P(+ | Test))