2: Probabilistic Models

CS1420: Machine Learning

Michael L. Littman
Spring 2018

http://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/ . Chapter 24.2, 9.3, 12.1.

Context

Talked about input: binary vector, output: bit.

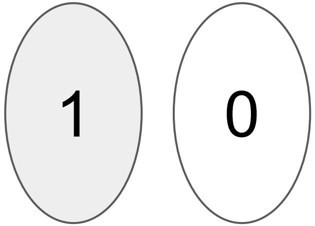
Representations: Decision trees, decision stumps, permutations.

Loss: empirical loss, entropy, gini/squared loss.

Optimizers: Exhaustive comparisons, greedy search, Hungarian matching.

Small step toward richer representations. Output: probability over bits.

Coin Learning



Like the permutation example, but even simpler.

Data:

```
1 1 0 0 0 0 0 1 0 0 1 0 1 0 1 1 1 1 0 1 0 0 1 1 1
1 0 1 1 1 0 1 1 1 0 1 1 1 1 1 0 0 1 0 0 0 1 1 1 0
0 1 0 1 1 1 0 1 1 0 0 0 1 0 0 1 1 0 0 0 0 0 1 0 0
0 0 1 1 0 1 1 0 0 1 1 0 0 1 1 1 1 0 1 1 0 0 1 1 0
```

100 coin flips of weighted coin. What’s p, its probability of “1”?

Choices for Estimators

y_i is the outcome (0 or 1) of the ith flip. What’s a good estimate for p?

A. Mode of y_is.

B. Median of y_is.

C. Mean of y_is.

D. Randomly selected y_i.

E. None of the above.

Median/Mode Estimator

It’s the most common outcome, which connects the problem to what we did in the previous unit.

Mean Estimator

Let's look at the mean: $p^{\text{hat}} = \mu = 1/m \sum_{i=1}^{m} y_i$.

Claim: $\mathrm{E}_{S\sim D}\,[p^{\text{hat}}] = p$.

That is, averaged over all possible size m samples, our estimate equals the target probability. An *unbiased estimator* is one that has this property.

Kinda cool that it does this. Not entirely obvious. (Although quite intuitive.)

Empirical Mean is Unbiased: Example

Let’s see an example for m=3:

$$
\begin{aligned}
&E_D\,[p^{\text{hat}}] \\
&= \tfrac{1}{3}\,(1\text{-}p)^2\,p + \tfrac{1}{3}\,(1\text{-}p)p(1\text{-}p) + \tfrac{2}{3}\,(1\text{-}p)\,p^2 + \tfrac{1}{3}\,p\,(1\text{-}p)^2 \\
&\qquad + \tfrac{2}{3}\,p\,(1\text{-}p)\,p + \tfrac{2}{3}\,p^2\,(1\text{-}p) + p^3 \\
&= 3\,\tfrac{1}{3}\,(1\text{-}p)^2\,p + 3\,\tfrac{2}{3}\,(1\text{-}p)\,p^2 + p^3 \\
&= p\,(\,(1\text{-}p)^2 + 2\,(1\text{-}p)\,p + p^2\,) \\
&= p\,(\,(1\text{-}p) + p\,)^2 \\
&= p
\end{aligned}
$$

S	$P(S)$	p^{hat}
0,0,0	$(1\text{-}p)^3$	0.000
0,0,1	$(1\text{-}p)^2\,p$	0.333
0,1,0	$(1\text{-}p)p(1\text{-}p)$	0.333
0,1,1	$(1\text{-}p)\,p^2$	0.667
1,0,0	$p\,(1\text{-}p)^2$	0.333
1,0,1	$p\,(1\text{-}p)\,p$	0.667
1,1,0	$p^2\,(1\text{-}p)$	0.667
1,1,1	p^3	1.000

Loss-based Perspective

We want p and p^{hat} to be similar.

- $\min |p^{\text{hat}} - p|$, $\min (p^{\text{hat}} - p)^2$
- $\min (\mathrm{E}[p^{\text{hat}}] - p)^2$
- $|p^{\text{hat}} - p| \leq \varepsilon$ with probability $1 - \delta$,

Different choices for the loss function lead to different p^{hat} choices.

- $L_S(p^{\text{hat}}) = \sum_{i=1}^{m} |y_i - p^{\text{hat}}|$
- $L_S(p^{\text{hat}}) = \sum_{i=1}^{m} (y_i - p^{\text{hat}})^2$
- $L_S(p^{\text{hat}}) = 1 - \Pr(S \mid p^{\text{hat}}) = 1 - \prod_{i=1}^{m} \Pr(y_i \mid p^{\text{hat}}) = 1 - \prod_{i=1}^{m} (p^{\text{hat}})^{yi} (1 - p^{\text{hat}})^{(1-yi)}$

Let’s go concrete to general.

Absolute Difference

$L_S(p^{\text{hat}}) = \sum_{i=1}^{m} |y_i - p^{\text{hat}}|$

What value of p^{hat} minimizes loss?

A. Mode of y_is.

B. Median of y_is.

C. Mean of y_is.

D. Randomly selected y_i.

E. None of the above.

Squared Difference

$L_S(p^{\text{hat}}) = \sum_{i=1}^{m} (y_i - p^{\text{hat}})^2$

What value of p^{hat} minimizes loss?

A. Mode of y_is.

B. Median of y_is.

C. Mean of y_is.

D. Randomly selected y_i.

E. None of the above.

Maximum Likelihood

$L_S(p^{\text{hat}}) = 1-\Pr(S|p^{\text{hat}}) = 1- \prod_{i=1}^{m} \Pr(y_i \mid p^{\text{hat}}) = 1- \prod_{i=1}^{m} (p^{\text{hat}})^{yi} (1-p^{\text{hat}})^{(1-yi)}$

What value of p^{hat} minimizes loss?

A. Mode of y_is.

B. Median of y_is.

C. Mean of y_is.

D. Randomly selected y_i.

E. None of the above.

Absolute Difference

$L_S(p^{\text{hat}}) = \sum_{i=1}^{m} |y_i - p^{\text{hat}}|$

0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

What value of p^{hat} minimizes loss?

Absolute Difference

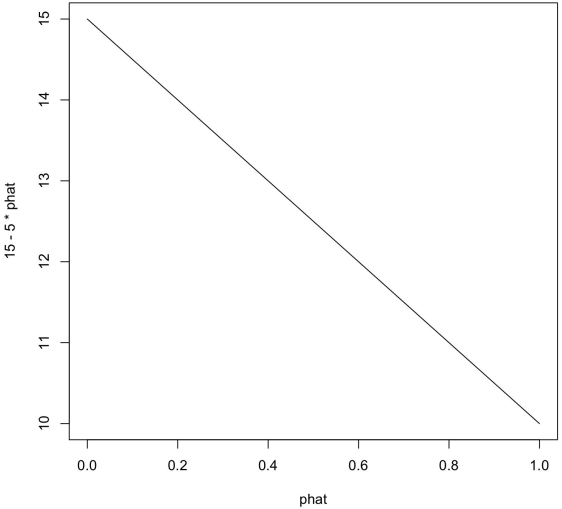
$$L_S(p^{\text{hat}}) = \sum_{i=1}^{m} |y_i - p^{\text{hat}}|$$

Must be minimized in [0,1]. Why?

0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

$$15\ (1 - p^{\text{hat}}) + 10\ p^{\text{hat}} = \ 15 - 5\ p^{\text{hat}}$$

Always minimized by most common value.



Squared Difference

$L_S(p^{\text{hat}}) = \sum_{i=1}^{m} (y_i - p^{\text{hat}})^2$

0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

What value of p^{hat} minimizes loss?

Squared Difference

$L_S(p^{\text{hat}}) = \sum_{i=1}^{m} (y_i - p^{\text{hat}})^2$

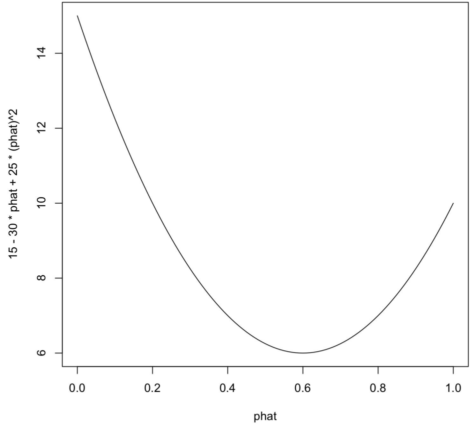
0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

$15\ (1 - p^{\text{hat}})^2 + 10\ (p^{\text{hat}})^2 = 15\ (1 - 2\, p^{\text{hat}} + (p^{\text{hat}})^2) + 10\ (p^{\text{hat}})^2 = 15 - 30\, p^{\text{hat}} + 25\ (p^{\text{hat}})^2$

Where is the derivative zero?

$D(15 - 30\, p^{\text{hat}} + 25\ (p^{\text{hat}})^2) = -30 + 50\, p^{\text{hat}} = 0$

$p^{\text{hat}} = \frac{3}{5}$



Always minimized by the mean!

Maximum Likelihood

$$L_S(p^{\text{hat}}) = 1 - \mathrm{Pr}(S|p^{\text{hat}}) = 1 - \prod_{i=1}^{m} \mathrm{Pr}(y_i \mid p^{\text{hat}}) = 1 - \prod_{i=1}^{m} (p^{\text{hat}})^{yi} (1 - p^{\text{hat}})^{(1-yi)}$$

What value of p^{hat} minimizes loss?

What makes the data most likely?

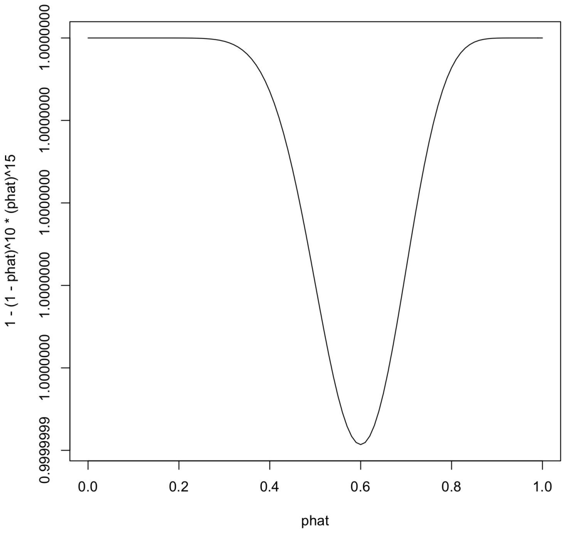
Maximum Likelihood

$$L_S(p^{\text{hat}}) = 1-\Pr(S|p^{\text{hat}}) = 1- \prod_{i=1}^{m} \Pr(y_i \mid p^{\text{hat}}) = 1- \prod_{i=1}^{m} (p^{\text{hat}})^{yi} (1-p^{\text{hat}})^{(1-yi)}$$

0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

$$1- (p^{\text{hat}})^{15} \ (1-p^{\text{hat}})^{10}$$

Where is the derivative zero?



Maximum Likelihood

$$D(1-\ (p^{\text{hat}})^{15}\ \ (1-p^{\text{hat}})^{10}) = D((p^{\text{hat}})^{15}\ \ (1-p^{\text{hat}})^{10})$$

$$= (p^{\text{hat}})^{15}\ D((1-p^{\text{hat}})^{10}) + D((p^{\text{hat}})^{15})\ (1-p^{\text{hat}})^{10})$$

$$= (p^{\text{hat}})^{15}\ 10\ (1-p^{\text{hat}})^{9}\ D(1-p^{\text{hat}}) + 15\ (p^{\text{hat}})^{14}\ (1-p^{\text{hat}})^{10}$$

$$= -(p^{\text{hat}})^{15}\ 10\ (1-p^{\text{hat}})^{9} + 15\ (p^{\text{hat}})^{14}\ (1-p^{\text{hat}})^{10} \quad = 0$$

$$(p^{\text{hat}})^{15}\ 10\ (1-p^{\text{hat}})^{9} = 15\ (p^{\text{hat}})^{14}\ (1-p^{\text{hat}})^{10}$$

$$(p^{\text{hat}})\ 10\ \ = 15\ (1-p^{\text{hat}})$$

$$10\ (p^{\text{hat}}) = 15-15p^{\text{hat}}$$

$$25\ (p^{\text{hat}}) = 15$$

$$p^{\text{hat}} = 3/5$$

Also always minimized by the mean!

Hoeffding’s Inequality (Bernoulli case)

Let p be a probability and p^{hat} be the estimate of that probability given m samples.

For any value ε: $\Pr(|p^{\text{hat}} - p| > \varepsilon) \leq 2e^{-2\varepsilon 2\ m}$. (Proof in book, appendix: B4.)

We set δ so as $2\ e^{-2\varepsilon 2\ m} \leq \delta$, so we are sure the true probability is even smaller. Solving for m, that means:

$\log(2\ e^{-2\varepsilon 2\ m}) \leq \log(\delta)$
$-2\ \varepsilon^2\ m \leq \log(\delta/2)$
$m \geq \log(\delta/2)\ /\ (-2\ \varepsilon^2)$
$m \geq \log(2/\delta)/\ (2\ \varepsilon^2)$ Compare to: $m \geq \log(1/□)/\varepsilon$

Summary

Mean:

- unbiased estimate
- minimizes squared error
- maximum likelihood
- PAC-like bound

Median/mode:

- minimizes absolute difference
- maximizes probability match

A Use Case

Two dungeons and dragons dice.

We choose the green die with probability b^g (and red otherwise). Red produces value x with probability b^{xr}. Green produces value x with probability b^{xg}.

```
def roll(b): return(sum((np.arange(len(b))+1)*np.random.multinomial(1,b)))
s = ()
for i in range(1000):
 if random.random() <= bg:
  s = s + ("g", roll(bxg))
 else:
  s = s + ("r", roll(bxr))
```

Data

```
(g, 15) (g, 11) (r,  8) (g, 18) (g,  2) (r, 15) (g, 20) (r, 15) (g, 12) (g, 11) (r,  7) (g, 10) (r,  2)
(g, 17) (r, 19) (g,  4) (r, 20) (g,  9) (g, 11) (r,  2) (g,  9) (g,  3) (g, 20) (r, 13) (g,  9) (g, 20)
(r,  9) (r, 15) (g,  5) (g,  9) (g,  4) (g,  6) (r,  5) (r,  5) (g,  7) (g, 11) (r,  8) (r, 15) (g,  4)
(g, 14) (g, 15) (r, 20) (g, 13) (g,  6) (g, 20) (g,  6) (g,  4) (r, 15) (g,  4) (r,  7) (g, 20) (r,  2)
(g, 18) (g, 13) (g, 15) (g, 13) (g, 12) (r, 13) (g,  9) (g, 12) (r, 19) (g,  5) (r,  8) (g, 11) (g,  5)
(r,  8) (g,  5) (r,  7) (g, 15) (r, 20) (r, 15) (r,  2) (g, 17) (g,  9) (r, 14) (g, 11) (g,  4) (r, 13)
(g,  5) (g, 12) (g, 12) (r,  8) (g,  6) (g, 15) (r,  7) (g, 12) (r,  7) (r,  9) (r, 14) (g,  5) (g, 18)
(g, 20) (r,  9) (g,  5) (g,  4) (g, 19) (r, 20) (g, 14) (g,  9) (r, 15) (g,  5) (r, 20) (g,  2) (r, 19)
(g, 19) (r, 15) (g, 18) (g, 18) (r, 10) (g, 18) (g, 18) (g, 12) (g,  5) (g,  5) (g,  1) (g, 20) (g,  4)
(r,  7) (g,  5) (g, 20) (r, 15) (g, 16) (r, 15) (r, 12) (g,  2) (g,  1) (r,  7) (g, 16) (g, 12) (g, 11)
(g, 13) (g, 17) (g,  6) (g,  3) (g,  2) (g, 18) (g,  2) (g, 15) (r,  7) (g,  2) (r, 13) (r,  8) (g, 12)
(r,  8) (g, 13) (g,  5) (r,  2) (g, 12) (r, 20) (g,  3) (g, 20) (g, 14) (g,  8) (g, 19) (r,  8) (g, 11)
(g, 20) (g, 20) (g,  5) (g,  6) (r, 20) (g,  5) (r, 14) (g, 13) (g, 11) (g, 10) (g,  4) (g,  5) (g,  5)
(g, 12) (g,  9) (g, 13) (g, 16) (g,  9) (r, 19) (g, 11) (g, 16) (g,  2) (g,  1) (g, 19) (g,  6) (g,  9)
(r,  5) (r,  3) (g, 10) (g, 11) (r,  4) (r,  7) (g,  5) (r,  2) (g,  2) (r,  3) (g, 16) (g,  1) (r, 20)
(g,  6) (r,  5) (g,  5) (r,  4) (g, 13) (g, 19) (g, 16) (g,  5) (g,  1) (r, 20) (r, 20) (g, 12) (g, 11)
(g,  9) (r,  9) (r, 13) (g,  1) (g,  5) (g, 16) (g, 16) (g, 15) (r, 10) (g,  9) (g,  4) (g, 15) (g, 20)
(g, 20) (g,  4) (r, 13) (g, 10) (r,  2) (r, 17) (g, 16) (g, 19) (g, 19) (r,  9) (g,  9) (g, 16) (g, 13)
(g,  9) (g, 13) (g, 17) (r,  2) (g, 12) (r, 19) (g, 19) (r,  8) (r, 19) (g,  6) (g, 16) (g, 12) (g, 17)
```

Goal: What Die Was Rolled?

Want a classifier that can predict the die from the roll.

Summary data:

```
Side:  [ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]
Green: [39, 61, 20, 41, 49, 25, 10,  5, 57,  8, 54, 57, 26, 11, 35, 19, 15, 48, 46, 65]
Red:   [ 4, 31,  6, 19, 16,  3, 45, 24, 26,  6,  0, 18, 17, 11, 27,  1,  1,  1, 34, 19]
```

Predict die: 16? 8? **15**?

A: red, B: green, C: huh?, D: depends on the weather, E: yes

Approach 1: ERM

ERM on hypothesis class $h(x) \in \{g,r\}$.

h: `[ g,  g,  g,  g,  g,  g,  r,  r,  g,  g,  g,  g,  g,  r,  g,  g,  g,  g,  g,  g]`

Empirical risk: 255.

How many hypotheses?

Approach 1: ERM

ERM on hypothesis class $h(x) \in \{\mathrm{g,r}\}$.

h: `[ g,  g,  g,  g,  g,  g,  r,  r,  g,  g,  g,  g,  g,  r,  g,  g,  g,  g,  g,  g]`

Empirical risk: 255.

How many hypotheses? $|H| = 2^{20}$

Turns out (see HW) we can get by with data on the order of $\log(|H|)$, so, good.

Oh, also, can be optimized efficiently!

Approach 2: Maximize Answer Probability

Estimate probabilities: $b^g = 0.691$. `sum(g)/(sum(g)+sum(r)+0.0)`

$x_g =$ 0.056, 0.088, 0.029, 0.059, 0.071, 0.036, 0.014, 0.007, 0.082, 0.012,
0.078, 0.082, 0.038, 0.016, 0.050, 0.021, 0.029, 0.069, 0.061, 0.100

$x_r =$ 0.013, 0.100, 0.019, 0.061, 0.052, 0.010, 0.146, 0.078, 0.084, 0.019,
0.000, 0.140, 0.055, 0.036, 0.005, 0.003, 0.003, 0.003, 0.111, 0.060

`np.array(g)/(sum(g)+0.0), np.array(r)/(sum(r)+0.0)`

Given input x, return y with maximum $\Pr(y|x)$.

$\Pr(g|15) =$ ___ % (nearest integer)

Using Bayes Rule

Pr(x|y) = Pr(y|x) Pr(x)/Pr(y)

Pr(15|g) = 0.050, Pr(15|r) = 0.005, Pr(g) = 0.8, Pr(r) = 1-Pr(g) = 0.2

Pr(15) = Pr(15 and g) + Pr(15 and r) = Pr(15 | g) Pr(g) + Pr(15 | r) Pr(r) = 0.004+0.001 = 0.005

Pr(g|15) = Pr(15|g) Pr(g) / Pr(15) = .004/.0041 = 0.9756 (so, 98%)

Pr(15|g)Pr(g) = .05 .8 = .0400

Pr(15|r)Pr(r) = .005 .2 = .0010

Pr(15) = Pr(15 & r) + Pr(15 & g)

They Are The Same

Side: `[ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]`

Green: `[39, 61, 20, 41, 49, 25, 10,  5, 57,  8, 54, 57, 26, 11, 35, 19, 15, 48, 46, 65]`

Red: `[ 4, 31,  6, 19, 16,  3, 45, 24, 26,  6,  0, 18, 17, 11, 27,  1,  1,  1, 34, 19]`

Let $c(x,y)$ be the number of times input x appeared with label y in the sample.

Approach 1: $h^{\text{hat}}(x) = \text{argmax}_y\ c(x,y)$

Approach 2: $h^{\text{hat}}(x) = \text{argmax}_y\ \Pr(y|x) = \text{argmax}_y\ \Pr(x|y)\Pr(y) / \Pr(x)$
$= \text{argmax}_y\ \Pr(x|y)\Pr(y) = \text{argmax}_y\ \Pr(x\ \&\ y) = \text{argmax}_y\ (c(x,y) / (\sum_{x'y'} c(x',y'))$
$= \text{argmax}_y\ c(x,y)$

Naive Bayes: Coping With Scaling Issues

But, these approaches don’t work for a 2^{100} sided die. Too many possibilities.

More constrained way to map $\{0,1\}^k$ to probabilities.

Similar to die case: Represent $\Pr(y)$ and $\Pr(x|y)$. But, x is a bit vector, now. Instead of producing it all at once, we’ll do each bit separately: $b^{iy} = \Pr(x^i|y)$, $b^y = \Pr(y)$.

Instead of a distribution over 8-bit vectors, we have 3 1-bit distributions.

Probability of x=001 for y=0?

	$\Pr(y)$	x^1	x^2	x^3
y=0	0.6	0.2	0.7	0.1
y=1	0.4	0.4	0.2	0.4

Application of Naive Bayes

Figure out if it’s raining by looking out the window

	Pr(y)	*sky bright?*	*ground wet?*	*see drops?*	*umbrellas?*	*coats?*
Not raining?	0.8	0.6	0.2	0.1	0.1	0.7
Rain?	0.2	0.2	0.9	0.8	0.6	0.8

Maximum Likelihood Estimation for Naive Bayes

The beautiful thing about Naive Bayes is the parameters are trivial to estimate.

Let $c(i,y)$ be number of times in the data we see feature i on when the label is y.
Let $c(y)$ be the number of times in the data we see label y.

b^{iy} stands for $\Pr(x^i|y)$. Estimate via $c(i,y)/c(y)$.

b^y stands for $\Pr(y)$. Estimate via $c(y)/\sum_{y'} c(y')$.

Running time: $O(mk)$.

$b^{3,0} = ?$

$h^{\text{hat}}(10001) = ?$

```
10011: 1
10100: 0
01011: 0
10100: 0
01011: 1
10101: 1
10101: 0
10011: 0
```

$c(i,y)$:

	1	**2**	**3**	**4**	**5**
0	4	1	3	2	3
1	2	1	1	2	3

$c(0)$: 5
$c(1)$: 3

Naive Bayes Example: Title to Series

The One with George Stephanopoulos	I'm Going to the Beach with Josh and His Friends!	Truth or Dick	My Mentor	Kimmy Goes to a Hotel!
The One with the Butt	Josh Has No Idea Where I Am!	Green Eyed Dick	My Best Friend's Mistake	Kimmy Finds Her Mom!
The One with the Blackout	Paula Needs to Get Over Josh!	Dick Like Me	My Fifteen Minutes	Kimmy Goes on a Date!
The One Where the Monkey Gets Away	Josh and I Work on a Case!	Dick's First Birthday	My Blind Date	Kimmy Goes on a Playdate!
The One with the East German Laundry Detergent	My Mom, Greg's Mom and Josh's Sweet Dance Moves!	Assault With a Deadly Dick	My Student	Kimmy Goes Roller Skating!
The One with the Evil Orthodontist	Josh and I Go to Los Angeles!	Angry Dick	My Old Lady	Kimmy Meets a Drunk Lady!
The One with Mrs. Bing	Why Is Josh in a Bad Mood?	Selfish Dick	My Super Ego	Kimmy Goes to a Play!
The One with the Monkey	Josh Is Going to Hawaii!	Dick, Smoker	My Occurrence	Kimmy Goes to Her Happy Place!
The One Where Nana Dies Twice	Josh Just Happens to Live Here!	Lonely Dick	My Drug Buddy	Kimmy Walks Into a Bar!
The Pilot	I'm Back at Camp with Josh!	Brains and Eggs	My Sacrificial Clam	Kimmy Meets a Celebrity!

Naive Bayes Example: Title to Series

Friends	**Crazy Ex-Girlfriend**	**Third Rock from the Sun**	**Scrubs**	**Unbreakable Kimmy Schmidt**
The One with George Stephanopoulos	I'm Going to the Beach with Josh and His Friends!	Truth or Dick	My Mentor	Kimmy Goes to a Hotel!
The One with the Butt	Josh Has No Idea Where I Am!	Green Eyed Dick	My Best Friend's Mistake	Kimmy Finds Her Mom!
The One with the Blackout	Paula Needs to Get Over Josh!	Dick Like Me	My Fifteen Minutes	Kimmy Goes on a Date!
The One Where the Monkey Gets Away	Josh and I Work on a Case!	Dick's First Birthday	My Blind Date	Kimmy Goes on a Playdate!
The One with the East German Laundry Detergent	My Mom, Greg's Mom and Josh's Sweet Dance Moves!	Assault With a Deadly Dick	My Student	Kimmy Goes Roller Skating!
The One with the Evil Orthodontist	Josh and I Go to Los Angeles!	Angry Dick	My Old Lady	Kimmy Meets a Drunk Lady!
The One with Mrs. Bing	Why Is Josh in a Bad Mood?	Selfish Dick	My Super Ego	Kimmy Goes to a Play!
The One with the Monkey	Josh Is Going to Hawaii!	Dick, Smoker	My Occurrence	Kimmy Goes to Her Happy Place!
The One Where Nana Dies Twice	Josh Just Happens to Live Here!	Lonely Dick	My Drug Buddy	Kimmy Walks Into a Bar!
The Pilot	I'm Back at Camp with Josh!	Brains and Eggs	My Sacrificial Clam	Kimmy Meets a Celebrity!

Naive Bayes Data

Counted up the fraction of times each of a set of words or punctuation appeared in a title.

Hedged away from 0 and 1 because multiplying by zero can be fatal: sometimes called *Laplace smoothing*.

Learning is simple and fast: estimate probabilities by counting.

	Friends	Crazy Ex-Girlfriend	Third Rock from the Sun	Scrubs	Unbreakable Kimmy Schmidt
!	0.01	0.90	0.01	0.01	0.99
a	0.10	0.01	0.01	0.10	0.70
dick	0.01	0.01	0.90	0.01	0.01
first	0.01	0.01	0.10	0.01	0.01
josh	0.01	0.99	0.01	0.01	0.01
kimmy	0.01	0.01	0.01	0.01	0.99
my	0.01	0.10	0.01	0.99	0.01
with	0.70	0.20	0.10	0.01	0.01

Using Naive Bayes

My First Thanksgiving with Josh! `10011011`

Invert the probabilities in the rows that correspond to missing features.

Take product down columns.

Multiply by priors (uniform here).

Pick biggest.

		Friends	Crazy Ex-Girlfriend	Third Rock from the Sun	Scrubs	Unbreakable Kimmy Schmidt
!	**1**	0.01	0.90	0.01	0.01	0.99
a	**0**	0.90	0.99	0.99	0.90	0.30
dick	**0**	0.99	0.99	0.10	0.99	0.99
first	**1**	0.01	0.01	0.10	0.01	0.01
josh	**1**	0.01	0.99	0.01	0.01	0.01
kimmy	**0**	0.99	0.99	0.99	0.99	0.01
my	**1**	0.01	0.10	0.01	0.99	0.01
with	**1**	0.70	0.20	0.10	0.01	0.01
		6.2e-09	1.7e-04	9.8e-10	8.7e-09	9.7e-11

Logistic Regression

Another way to produce probabilities from bit vectors is a much studied approach called *logistic regression*.

Like Naive Bayes, we have a parameter for each attribute that can be thought of as a way of weighting the evidence from that attribute when predicting the output.

However, the parameters themselves are no longer probabilities.

Logistic function (a sigmoid)

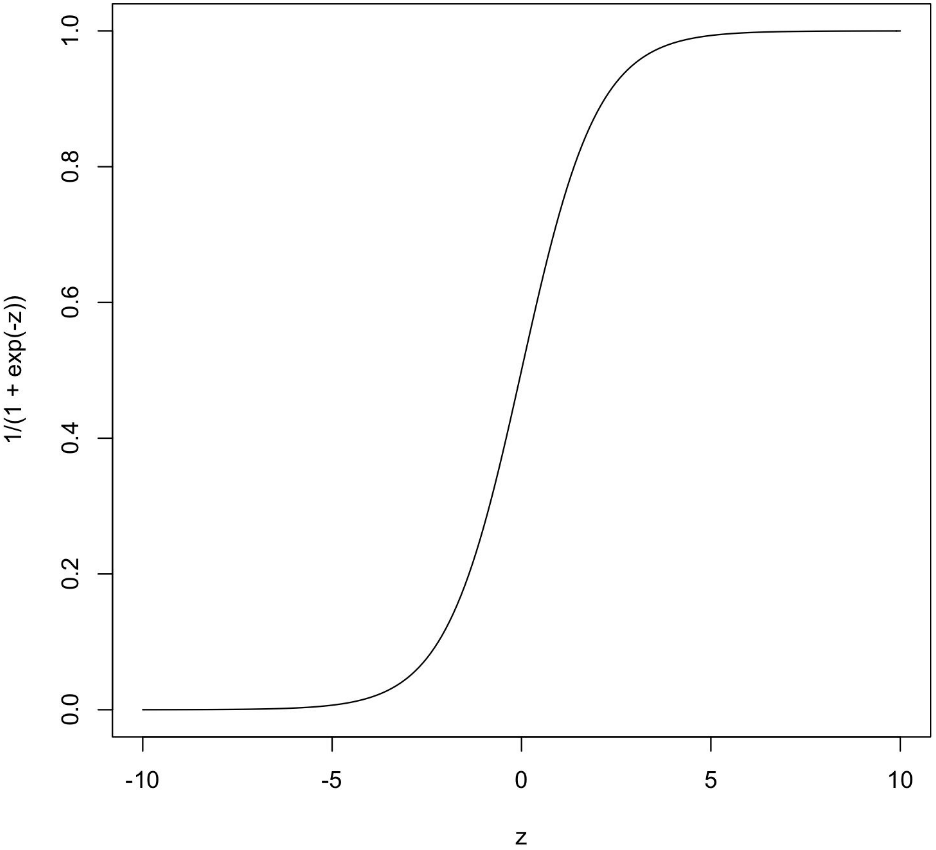
$\varphi_{\mathrm{sig}}(z) = 1/\,(1 + e^{-z})$.

Given input x, we're going to scale it by a set of weights w, then put it through the logistic function to get the probability of y=1. (Weird jump #1.)

We write $<w,x> = \sum_i w^i\, x^i$.

So, $h_w(x) = 1/\,(1 + e^{-<w,x>})$ is the predicted probability of y=1.

More weight, more certainty.



Loss: How Wrong?

$h_w(x) = 1/\ (1 + e^{-<w,x>})$ is the probability of output 1. Bad if big when y=0.

$1 - h_w(x) = 1 - 1/\ (1 + e^{-<w,x>})$ is the probability of output 0. Bad if big when y=1.

$$
\begin{aligned}
&1 - h_w(x) = 1 - 1/\ (1 + e^{-<w,x>})\\
&= (1 + e^{-<w,x>})\ /\ (1 + e^{-<w,x>}) - 1/\ (1 + e^{-<w,x>})\\
&= (1 + e^{-<w,x>} - 1)\ /\ (1 + e^{-<w,x>})\\
&= e^{-<w,x>}\ /\ (1 + e^{-<w,x>})\\
&= e^{-<w,x>}/e^{-<w,x>}\ /\ (1/e^{-<w,x>} + e^{-<w,x>}/e^{-<w,x>})\\
&= 1\ /\ (1/e^{-<w,x>} + 1)\\
&= 1\ /\ (1 + 1/e^{-<w,x>})\\
&= 1\ /\ (1 + e^{<w,x>})
\end{aligned}
$$

Loss: How Wrong?

$h_w(x) = 1/\,(1 + e^{-<w,x>})$ is the probability of output 1. Bad if big when y=0.

$1 - h_w(x) = 1/\,(1 + e^{<w,x>})$ is the probability of output 0. Bad if big when y=1.

Combining:

$1/\,(1 + e^{(2y\text{-}1)<w,x>})$ is the probability of bad output.

That's monotonic with: $1 + e^{\text{-}(2y\text{-}1)<w,x>}$

And that's monotonic with: $\log(1 + e^{\text{-}(2y\text{-}1)<w,x>})$

Make this quantity the loss: (Weird jump #2.)

y	multi-plier	$2y$-1
0	-1	-1
1	1	1

Minimizing the Loss

$L_S(w) = 1/m \sum_{(x,y)\in S} \log(1+e^{-(2y-1)<w,x>})$.

Want w that minimizes this quantity. Take derivatives for each component of w.

$D_{wi}(L_S(w)) = D_{wi}(1/m \sum_{(x,y)\in S} \log(1+e^{-(2y-1)<w,x>}))$
$= 1/m \sum_{(x,y)\in S} D_{wi}(\log(1+e^{-(2y-1)<w,x>}))$
$= 1/m \sum_{(x,y)\in S} 1/(1+e^{-(2y-1)<w,x>})\ D_{wi}(1+e^{-(2y-1)<w,x>})$
$= 1/m \sum_{(x,y)\in S} 1/(1+e^{-(2y-1)<w,x>})\ D_{wi}(e^{-(2y-1)<w,x>})$
$= 1/m \sum_{(x,y)\in S} 1/(1+e^{-(2y-1)<w,x>})\ (e^{-(2y-1)<w,x>})D_{wi}(-(2y-1)<w,x>)$
$= -1/m \sum_{(x,y)\in S} e^{-(2y-1)<w,x>}/(1+e^{-(2y-1)<w,x>})\ (2y-1)D_{wi}(<w,x>)$
$= -1/m \sum_{(x,y)\in S} e^{-(2y-1)<w,x>}/(1+e^{-(2y-1)<w,x>})\ (2y-1)x_i$
Can’t just set it to zero this time.

It’s Convex

If we take the derivative again...

$$
\begin{aligned}
&D_{wi}(-1/m \textstyle\sum_{(x,y)\in S} e^{-(2y-1)\langle w,x\rangle}/(1+e^{-(2y-1)\langle w,x\rangle})\ (2y-1)x_i) \\
&= -(2y-1)x_i\,/m \textstyle\sum_{(x,y)\in S} D_{wi}(e^{-(2y-1)\langle w,x\rangle}/(1+e^{-(2y-1)\langle w,x\rangle})) \\
&= -(2y-1)x_i\,/m \textstyle\sum_{(x,y)\in S}\ e^{-(2y-1)\langle w,x\rangle}/(1+e^{-(2y-1)\langle w,x\rangle})^2\ D_{wi}(-(2y-1)\langle w,x\rangle) \\
&= (2y-1)^2\, x_i^{\ 2}/m \textstyle\sum_{(x,y)\in S}\ e^{-(2y-1)\langle w,x\rangle}/(1+e^{-(2y-1)\langle w,x\rangle})^2
\end{aligned}
$$

All quantities are positive. So, loss is convex in every parameter.

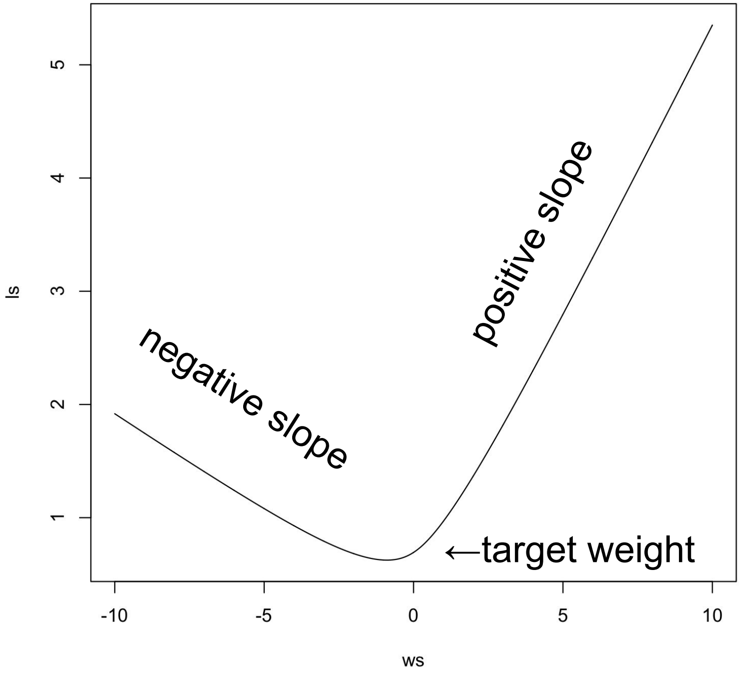
That helps...

Convex function

Our function has a minimum and the derivatives point to it.

$w_i \leftarrow w_i - \alpha\, D_{wi}(L_S(w))$

If we set α small enough so we don’t repeatedly overshoot, and update each of the weights,



Putting it Together

We can adjust the weights that define the logistic regression function so that it minimizes loss (and therefore fits the sample well). The weights that come out can then be used to classify new instances.

Homework Related Details

It is straightforward to extend these algorithms beyond binary output to multiclass output. Whereas, $h_w(x) = 1/\,(1 + e^{-<w,x>})$ is the predicted probability of $y=1$ for the two-class case, the new formula becomes $h_w(x) = e^{-<wi,x>}/\,\sum_j(e^{-<wj,x>})$ for the predicted probability of $y=i$, where w_i is a set of weights for class i. Note that, in the two-class case, for $y=1$, we have

$h_w(x) = e^{-<w1,x>}/\,\sum_j(e^{-<wj,x>}) = e^{-<w1,x>}/\,(e^{-<w1,x>}+e^{-<w2,x>}) = 1/\,(1+e^{-<w2,x>-<w1,x>}) = 1/$
$(1+e^{-<w,x>})$

Stochastic gradient descent is like gradient descent only it uses some randomness to avoid certain kinds of bad cases (kind of like quicksort$^{\text{TM}}$).

Loss Function in Multiclass Case

The loss for a given input/output pair is *cross entropy*: $L((x, y)) = -\sum_i y^i \log(v^i)$, where v is the output of softmax given input x, and y is the label, written out as a ("one hot") Boolean vector (y^i)=1 iff class i is the label, otherwise 0).

Concentration Inequalities

https://en.wikipedia.org/wiki/Hoeffding%27s_inequality

Bernouli case:

$$\mathbb{P}((p-\epsilon)n \leq H(n) \leq (p+\varepsilon)n) \geq 1-2\exp(-2\varepsilon^2 n).$$

General case:

$$\mathbb{P}\left(\left|\overline{X}-\mathrm{E}\left[\overline{X}\right]\right| \geq t\right) \leq 2\exp\left(-\frac{2n^2t^2}{\sum_{i=1}^n (b_i-a_i)^2}\right)$$

https://en.wikipedia.org/wiki/Azuma%27s_inequality

$$P(|X_N - X_0| \geq t) \leq 2\exp\left(\frac{-t^2}{2\sum_{k=1}^N c_k^2}\right).$$

https://en.wikipedia.org/wiki/Bernstein_inequalities_(probability_theory)

$$\mathbb{P}\left(\sum_{i=1}^n X_i > t\right) \leq \exp\left(-\frac{\frac{1}{2}t^2}{\sum \mathbb{E}\left[X_j^2\right] + \frac{1}{3}Mt}\right)$$

https://en.wikipedia.org/wiki/Doob_martingale#McDiarmid's_inequality