CAS CS 440
Lec 22

Supervised Learning IV - Naive Bayes

1. Our Example Data
    a.

Outlook	Temperature	Humidity	Windy?	Play Outside?
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

    b. Play Outside? is the label (ground truth)
2. Conversion into Numeric Representation
    a.

Outlook	Temperature	Humidity	Windy?	Play Outside?
0	2	1	0	0
0	2	1	1	0
1	2	1	0	1
2	1	1	0	1
2	0	0	0	1
2	0	0	1	0
1	0	0	1	1
0	1	1	0	0
0	0	0	0	1
2	1	0	0	1
0	1	0	1	1
1	1	1	1	1
1	2	0	0	1
2	1	1	1	0

    b. Change to machine familiar format
3. Modeling the Data
    a. Last time we assumed we wanted to learn a decision tree
        i. Biased the model with axis-parallel cuts

b. This time:

$$\hat{y} = \underset{y \in Y}{\operatorname{argmax}} \Pr[Y = y \mid x]$$

$$= \underset{y \in Y}{\operatorname{argmax}} \frac{\Pr[x \mid Y = y]\Pr[Y = y]}{\Pr[x]}$$

$$= \underset{y \in Y}{\operatorname{argmax}} \Pr[x \mid Y = y]\Pr[Y = y]$$

i. Y-hat is the output

ii. Calculate the probability of class 1 and class 0 to find out which class is more likely by applying bayes rule

iii. The denominator does not depend on the value of y, so we can factor it out

c. Learn probabilistic model

4. The Hard Part

a. Need to learn two things:

i. Pr[Y]

1.

Play Outside? = y	Pr[Y=y]
0	5/14
1	9/14

2. Easy: just count!

ii. Pr[x | Y]

1. Hard, need to learn a joint distribution:

a. $\Pr[x_{Outlook}, x_{Temp}, x_{Humidity}, x_{Windy} \mid Y]$

b. In general:

i. $\Pr[x \mid y] = \Pr[f_1, f_2, f_3, \ldots, f_m \mid Y]$

ii. Figuring this is difficult → counting them will take a lot of time

b. What should we do?

i. Naïve Bayes part:

1. Assume features are conditionally independent

a. $\Pr[f_1, f_2, f_3, \ldots, f_m \mid Y] = \Pr[f_1 \mid Y]\Pr[f_2 \mid Y]\Pr[f_3 \mid Y]\ldots\Pr[f_m \mid Y]$

b. We assume the features are independent when in reality it is not necessary the case

5. Naive Bayes

a. Since we assumed conditional independence:

i. Only need to focus on one feature at a time!

ii. Much easier!

1. $\Pr\left[x_{Outlook} \middle| Y\right]$
   $\Pr[x_{Humidity} | Y]$
   $\Pr[x_{Temp} | Y]$
   $\Pr[x_{Windy} | Y]$
2. Now we do not need superior amount of data
3.

Outlook	Pr[Outlook = x | Y = 0]	Pr[Outlook = x | Y = 1]
0	3/5	2/9
1	0/5	4/9
2	2/5	3/9

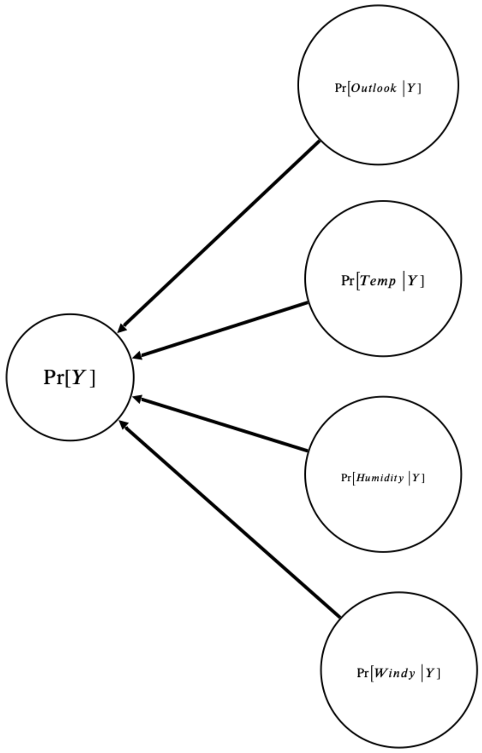
4.

Humidity	Pr[Humidity = x | Y = 0]	Pr[Humidity = x | Y = 1]
0	2/5	6/9
1	3/5	3/9

b. Be careful!
   i. Don't want 0 probs!
      1. Smooth the distribution → 0s are too aggressive and make things small value but not 0
         a. Smoothing refers to the fact that I should not believe the training data 100%
      2. If there are 0s, Naive Bayes is overfitting → caring too much about the data collected

6. Naive Bayes Visually
   a.

Outlook	Temperature	Humidity	Windy?	Play Outside?
0	2	1	0	0
0	2	1	1	0
1	2	1	0	1
2	1	1	0	1
2	0	0	0	1
2	0	0	1	0
1	0	0	1	1
0	1	1	0	0
0	0	0	0	1
2	1	0	0	1
0	1	0	1	1
1	1	1	1	1
1	2	0	0	1
2	1	1	1	0

b. 

    i. Instance of Bayesian network

7. How to Make Predictions

    a.

function predict(x) returns int // class label
  best_prob = -inf
  best_class = -1
  for y ∈ Y do
    $$\Pr\left[Y = y \middle| x\right] = \Pr[\mathrm{Y} = \mathrm{y}]\prod_{f \in F} \Pr[f = x_f \mid Y = y]$$
    if $\Pr\left[Y = y \middle| x\right]$ > best_prob then
      best_prob = $\Pr\left[Y = y \middle| x\right]$
      best_class = y
    end if

  end for
  return best_class

8. Continuous Features
    a. The node no longer contains a pmf
    b. That's ok!
        i. Parameterize with a pdf

1. For instance, assume a Gaussian (or another pdf)
2. Learn the parameters of the pdf from the feature data!

c. When making predictions:
   i. You have the feature value: how likely is it to be drawn from the pdf!

9. Decision Boundary
   a. Consider binary classification
      i. Argument extends to higher dims
   b.
$$\Pr\left[c=1\,\middle|\,\vec{x}\right] = \frac{\Pr\left[\vec{x}\,\middle|\,c=1\right]\Pr\left[c=1\right]}{\Pr\left[\vec{x}\,\middle|\,c=1\right]\Pr\left[c=1\right]+\Pr\left[\vec{x}\,\middle|\,c=0\right]\Pr\left[c=0\right]}$$
$$= \frac{1}{1+\frac{\Pr\left[\vec{x}\,\middle|\,c=0\right]\Pr\left[c=0\right]}{\Pr\left[\vec{x}\,\middle|\,c=1\right]\Pr\left[c=1\right]}}$$
$$= \frac{1}{1+e^{-\log\left(\frac{\Pr\left[\vec{x}\,\middle|\,c=1\right]\Pr\left[c=1\right]}{\Pr\left[\vec{x}\,\middle|\,c=0\right]\Pr\left[c=0\right]}\right)}}$$
$$= \frac{1}{1+e^{-\log\left(\frac{\Pr\left[\vec{x}\,\middle|\,c=1\right]}{\Pr\left[\vec{x}\,\middle|\,c=0\right]}\right)-\log\left(\frac{\Pr\left[c=1\right]}{\Pr\left[c=0\right]}\right)}}$$
   c. $$\sigma(z) = \frac{1}{1+e^{-z}}$$ → sigmoid function
   d.
$$= \sigma\left(\log\left(\frac{\Pr\left[\vec{x}\,\middle|\,c=1\right]}{\Pr\left[\vec{x}\,\middle|\,c=0\right]}\right)+\log\left(\frac{\Pr\left[c=1\right]}{\Pr\left[c=0\right]}\right)\right)$$
$$= \sigma\left(\log\left(\frac{\Pr\left[c=1\right]}{\Pr\left[c=0\right]}\right)+\sum_{x_i}\log\left(\frac{\Pr\left[x_i\,\middle|\,c=1\right]}{\Pr\left[x_i\,\middle|\,c=0\right]}\right)\right)$$
   e. This is not a linear equation in x
      i. In general, Naïve Bayes is not a linear classifier!
      ii. The decision boundary is not a line
   f. What if $\Pr[x_i|c]$ is from the exponential family?
      i. Gaussian
      ii. Exponential
      iii. Bernoulli
      iv. Dirichlet
      v. Poisson
      vi. …

10. When Pr[xi|c] is from the exponential family

a. General formula for distribution in exponential family

$$\Pr[x_i|c] = \underbrace{h_i(x_i)}_{\text{amplitude}}\, e^{\underbrace{\vec{w}_i^{cT}\phi_i(x_i)}_{\text{Linear comb. of features}} - \underbrace{f(\vec{w}_i^{c})}_{\text{offset}}}$$

$$\begin{aligned}
\Pr\left[c=1\,\middle|\,\vec{x}\right] &= \sigma\left(\log\left(\frac{\Pr[c=1]}{\Pr[c=0]}\right) + \sum_{x_i}\log\left(\frac{\Pr[x_i|c=1]}{\Pr[x_i|c=0]}\right)\right)\\
&= \sigma\left(\log\left(\frac{\Pr[c=1]}{\Pr[c=0]}\right) + \sum_{x_i}\log\left(\frac{h_i(x_i)e^{\vec{w}_i^{1T}\phi_i(x_i)-f(\vec{w}_i^{1})}}{h_i(x_i)e^{\vec{w}_i^{0T}\phi_i(x_i)-f(\vec{w}_i^{0})}}\right)\right)\\
&= \sigma\left(\log\left(\frac{\Pr[c=1]}{\Pr[c=0]}\right) + \sum_{x_i}\left[\log\left(\frac{h_i(x_i)}{h_i(x_i)}\right) + \log\left(e^{\vec{w}_i^{1T}\phi_i(x_i)-f(\vec{w}_i^{1})-\left(\vec{w}_i^{0T}\phi_i(x_i)-f(\vec{w}_i^{0})\right)}\right)\right]\right)\\
&= \sigma\left(\underbrace{\log\left(\frac{\Pr[c=1]}{\Pr[c=0]}\right) + \sum_{x_i}\left[-f(\vec{w}_i^{1}) + f(\vec{w}_i^{0})\right]}_{\text{Constant offset } b} + \sum_{x_i}\underbrace{\left(\vec{w}_i^{1T} - \vec{w}_i^{0T}\right)}_{\text{Weights } \vec{w}}\phi_i(x_i)\right)
\end{aligned}$$

   i. Fi takes the ith feature and it turns it into bunch of other features in the new coordinate space

b. $$= \sigma\left(b + \sum_{x_i}\vec{w}^{T}\phi_i(x_i)\right)$$

c. When Pr[xi|c] is exponential
   i. Naïve Bayes is a linear classifier
      1. Not linear in x
      2. Linear in $\phi(\vec{x})$
         a. Linear in the new feature space
      3. Naive Bayes has internal representation of the points provided (more human like) → it is linear in its internal space
      4. Asymptotically Logistic Regression!
         a. Softmax Regression for k > 2 classes