Machine Learning Applications: Mushrooms

Alessio Benavoli

CSIS
University of Limerick

Classifying mushrooms

Our goal is to classify mushrooms as p = poisonous, e = edible and u=unknown.

We have some input characteristic that we can use such as

1. red-color: yes, no
2. capSurface: fibrous=f,smooth=s, scaly=y

Small dataset

$EdibleOrPoisonous$	$red-color$
e	y
e	y
e	y
p	n
p	y
p	n

Based on these examples, if we see a red mushroom, what is the probability that the mushroom is poisonous (or edible)?

We apply Bayes' Rule:

$$p(edible|Red = y) = \frac{p(Red = y|edible)P(edible)}{p(Red = y)}$$

Small dataset and one feature

$EdibleOrPoisonous$	$red-color$
e	y
e	y
e	y
p	n
p	y
p	n

We can estimate the probabilities by looking at the mushrooms we have seen so far (the table above).

$$p(edible|Red=y) = \frac{p(Red=y|edible)P(edible)}{p(Red=y)}$$

$$\frac{\frac{3}{3}\cdot\frac{3}{6}}{\frac{4}{6}} = \frac{3}{4}$$

$$p(poisson|Red=y) = \frac{p(Red=y|poisson)P(poisson)}{p(Red=y)}$$

$$\frac{\frac{1}{3}\cdot\frac{3}{6}}{\frac{4}{6}} = \frac{1}{4}$$

$$p(unknown|Red=y) = 1 - p(poisson|Red=y) - p(edible|Red=y) = 0$$

Small dataset two features

$EdibleOrPoisonous$	$red-color$	$capSurface$
e	y	s
e	y	s
e	y	y
p	n	f
p	y	s
p	n	f

We can estimate the probabilities by Bayes'rule

$$
\begin{aligned}
& p(edible|Red = y, capSurface = s) \\
&= \frac{p(Red = y, capSurface = s|edible)p(edible)}{p(Red = y, capSurface = s)} \\
&= \frac{p(Red = y|edible)p(capSurface = s|edible)p(edible)}{p(Red = y, capSurface = s)}
\end{aligned}
$$

with

$$
\begin{aligned}
& p(Red = y, capSurface = s) \\
&= p(Red = y|edible)p(capSurface = s|edible)p(edible) \\
&+ p(Red = y|poisson)p(capSurface = s|poisson)p(poisson) \\
&+ p(Red = y|unknown)p(capSurface = s|unknown)p(unknown)
\end{aligned}
$$

We have assumed that the features are conditionally independent given the class.

Small dataset two features

$EdibleOrPoisonous$	$red-color$	$capSurface$
e	y	s
e	y	s
e	y	y
p	n	f
p	y	s
p	n	f

We can estimate those probabilities by MLE:

$$
\begin{aligned}
& p(edible|Red=y, capSurface=s) \\
&= \frac{p(Red=y|edible)p(capSurface=s|edible)p(edible)}{p(Red=y, capSurface=s)} \\
&= \frac{\frac{3}{3}\frac{2}{3}\frac{3}{6}}{p(Red=y, capSurface=s)}
\end{aligned}
$$

with

$$
p(Red=y, capSurface=s) = \frac{3}{3}\frac{2}{3}\frac{3}{6} + \frac{1}{3}\frac{1}{3}\frac{3}{6} + 0\cdot 0\cdot 0 = \frac{2}{6} + \frac{1}{18} = \frac{7}{18}
$$

and so

$$
p(edible|Red=y, capSurface=s) = \frac{\frac{2}{6}}{\frac{7}{18}} = \frac{6}{7}
$$

Summing up

$$p(edible|Red = y, capSurface = s) = \frac{6}{7}$$
$$p(poisonous|Red = y, capSurface = s) = \frac{1}{7}$$
$$p(unknow|Red = y, capSurface = s) = 0$$

Python

```
import numpy as np
#e->1,p->0
#y->1,n->0
#s->0,y->1,f->2
#Columns are edible,redColor,capSurface
data=np.array([
    [1 , 1 , 0],
    [1 , 1 , 0],
    [1 , 1 , 1],
    [0 , 0 , 2],
    [0 , 1 , 0],
    [0 , 0 , 2]])
```

We need to transform the data because the sklearn implementation of MultinomialNB needs 1-0 features.

	EdibleOrPoisonous	RedColor_1	RedColor_0	CapSurface_0	CapSurface_1	CapSurface_2
0	1	1.0	0.0	1.0	0.0	0.0
1	1	1.0	0.0	1.0	0.0	0.0
2	1	1.0	0.0	0.0	1.0	0.0
3	0	0.0	1.0	0.0	0.0	1.0
4	0	1.0	0.0	1.0	0.0	0.0
5	0	0.0	1.0	0.0	0.0	1.0

Python

	EdibleOrPoisonous	RedColor_1	RedColor_0	CapSurface_0	CapSurface_1	CapSurface_2
0	1	1.0	0.0	1.0	0.0	0.0
1	1	1.0	0.0	1.0	0.0	0.0
2	1	1.0	0.0	0.0	1.0	0.0
3	0	0.0	1.0	0.0	0.0	1.0
4	0	1.0	0.0	1.0	0.0	0.0
5	0	0.0	1.0	0.0	0.0	1.0

```
from sklearn.naive_bayes import MultinomialNB
clf=MultinomialNB(alpha=0,fit_prior=True)
clf.fit(X,y)
clf.predict_proba(np.array([[1,0,1,0,0]]))
>> array([[0.14285714, 0.85714286]])
```

Note that

$$0.85714286 = \frac{6}{7}$$

```
clf.predict(np.array([[1,0,1,0,0]]))
>> [1]
```

when we run "predict" then the classifier returns the class with the highest probability.

Python

	EdibleOrPoisonous	RedColor_1	RedColor_0	CapSurface_0	CapSurface_1	CapSurface_2
0	1	1.0	0.0	1.0	0.0	0.0
1	1	1.0	0.0	1.0	0.0	0.0
2	1	1.0	0.0	0.0	1.0	0.0
3	0	0.0	1.0	0.0	0.0	1.0
4	0	1.0	0.0	1.0	0.0	0.0
5	0	0.0	1.0	0.0	0.0	1.0

```
from sklearn.naive_bayes import MultinomialNB
clf=MultinomialNB(alpha=0,fit_prior=True)
clf.fit(X,y)
clf.predict_proba(np.array([[1,0,0,1,0]]))
>> array([[0,1]])
```

It seems that this is not right. The classifier returns “edible” with certainty although we have only seen one case for “CapSurface=scaly”

Regularisation

	EdibleOrPoisonous	RedColor_1	RedColor_0	CapSurface_0	CapSurface_1	CapSurface_2
0	1	1.0	0.0	1.0	0.0	0.0
1	1	1.0	0.0	1.0	0.0	0.0
2	1	1.0	0.0	0.0	1.0	0.0
3	0	0.0	1.0	0.0	0.0	1.0
4	0	1.0	0.0	1.0	0.0	0.0
5	0	0.0	1.0	0.0	0.0	1.0

```
from sklearn.naive_bayes import MultinomialNB
clf=MultinomialNB(alpha=1,fit_prior=True)
clf.fit(X,y)
clf.predict_proba(np.array([[1,0,0,1,0]]))
>> array([[0.2,0.8]])
```

Now the MLE estimator of the probability of the feature value given the class value is regularised, i.e.,

$$p(CapSurface = scaly|pois.) = \frac{n_{CapSurface=scaly,pois.} + \alpha}{n_{poisonous} + m\alpha} = \frac{0+1}{3+5} = \frac{1}{8}$$

where m is the number of features ($m = 5$ in the example).

Regularisation

	EdibleOrPoisonous	RedColor_1	RedColor_0	CapSurface_0	CapSurface_1	CapSurface_2
0	1	1.0	0.0	1.0	0.0	0.0
1	1	1.0	0.0	1.0	0.0	0.0
2	1	1.0	0.0	0.0	1.0	0.0
3	0	0.0	1.0	0.0	0.0	1.0
4	0	1.0	0.0	1.0	0.0	0.0
5	0	0.0	1.0	0.0	0.0	1.0

$$p(edible|Red = y, CapSurface = s)$$

$$= \frac{p(Red = y|edible)p(CapSurface = s|edible)p(edible)}{p(Red = y, CapSurface = s))}$$

$$= \frac{\frac{3+1}{3+5}\frac{1+1}{3+5}\frac{3}{6}}{\frac{3+1}{3+5}\frac{1+1}{3+5}\frac{3}{6} + \frac{1+1}{3+5}\frac{1}{3+5}\frac{3}{6}} = \frac{4}{5} = 0.8$$

Classifying mushrooms

Our goal is to classify mushrooms as p = poisonous, e = edible and u=unknown.

We have some input characteristic that we can use such as

1. capShape: bell=b,conical=c,convex=x,flat=f, knobbed=k,sunken=s
2. capSurface: fibrous=f,grooves=g,scaly=y,smooth=s
3. cap-color: brown=n,buff=b,cinnamon=c,gray=g,green=r, pink=p,purple=u,red=e,white=w,yellow=y
4. bruises?: bruises=t,no=f
5. odor: almond=a,anise=l,creosote=c,fishy=y,foul=f, musty=m,none=n,pungent=p,spicy=s
6. gill-attachment: attached=a,descending=d,free=f,notched=n
7. gill-spacing: close=c,crowded=w,distant=d
8. gill-size: broad=b,narrow=n
9. gill-color: black=k,brown=n,buff=b,chocolate=h,gray=g, green=r,orange=o,pink=p,purple=u,red=e, white=w,yellow=y
10. stalk-shape: enlarging=e,tapering=t
11. stalk-root: bulbous=b,club=c,cup=u,equal=e, rhizomorphs=z,rooted=r,missing=?
12. stalk-surface-above-ring: fibrous=f,scaly=y,silky=k,smooth=s
13. stalk-surface-below-ring: fibrous=f,scaly=y,silky=k,smooth=s
14. stalk-color-above-ring: brown=n,buff=b,cinnamon=c,gray=g,orange=o, pink=p,red=e,white=w,yellow=y
15. stalk-color-below-ring: brown=n,buff=b,cinnamon=c,gray=g,orange=o, pink=p,red=e,white=w,yellow=y
16. veil-type: partial=p,universal=u 17. veil-color: brown=n,orange=o,white=w,yellow=y
18. ring-number: none=n,one=o,two=t
19. ring-type: cobwebby=c,evanescent=e,flaring=f,large=l, none=n,pendant=p,sheathing=s,zone=z
20. spore-print-color: black=k,brown=n,buff=b,chocolate=h,green=r, orange=o,purple=u,white=w,yellow=y
21. population: abundant=a,clustered=c,numerous=n, scattered=s,several=v,solitary=y
22. habitat: grasses=g,leaves=l,meadows=m,paths=p, urban=u,waste=w,woods=d

Classifying mushrooms

See notebook