36-350 Data Mining

Tom Minka

Day 3

Using probabilistic models

Categorical case

- Given batch of categorical observations
- Independent and identically distributed
- Sample proportions ≈ probabilities

$$\hat{p}(x) = \frac{\text{number of x's in batch}}{\text{size of batch}} \approx p(x)$$

- Thus infer probabilities (with error)
- Non-parametric: can model arbitrary distribution

Density estimation

- Inference problem: going beyond the sample
- Given sample, want to know about wider population or process
- Result is probability histogram or density curve

Corrected estimate

- Zero counts lead to zero probabilities
  - Not safe
- All counts should be started at 1 (or some other small value):

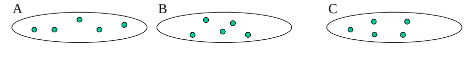
$$p(x) \approx \frac{(\text{number of x's in batch}) + 1}{(\text{size of batch}) + (\text{number of bins})}$$

Applications

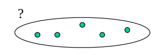
- Classification
  - Text: news articles, web pages
  - Imagery: natural scenes, face recognition
- Anomaly detection
  - Cellular phone fraud
  - UNIX intrusions

Classification problem

- Given samples of predefined populations:



- Determine most likely population of an unlabeled sample



Maximum-likelihood classification

- Estimate distribution of each population:

  $\hat{p}(x \mid \text{pop}) \qquad \text{pop} = \text{A, B, or C}$
- Given new sample, compute probability it could have arisen from A, B, or C
- **Likelihood** of each population:

$$L(\text{pop}) = p(\text{sample} \mid \text{pop})$$

- Assign sample to population with largest L

Computing L

- Let sample = $\{y_1, \ldots, y_n\}$
- Under independence assumption:

$$p(\text{sample} \mid \text{pop}) = \prod_i p(y_i \mid \text{pop})$$
$$\approx \prod_i \hat{p}(y_i \mid \text{pop})$$
$$\log L(\text{pop}) = \sum_i \log \hat{p}(y_i \mid \text{pop})$$

- If value x occurs n_x times,

$$\log L(\text{pop}) = \sum_x n_x \log \hat{p}(x \mid \text{pop})$$

Text classification

- News articles: business, politics, religion, etc.
- An article is a sample of words from a word population: business words, politics words, etc.
- Classify an article by most likely population of words it was drawn from
- Popular, successful technique

Text classification procedure

1. Collect all articles labeled “politics” into single batch of words
2. Words are categorical observations, with about 100,000 possible values
3. Compute probability histogram (100,000 bins)

Word	Prob.	Word	Prob.
the	.0619	president	.0023
to	.0332	government	.0024
…	…	advance	.00004

Text classification procedure

- To classify a document, sum over all 100,000 words:

$$\log L(\text{politics}) = \sum_w n_w \log \hat{p}(\text{w} \mid \text{politics})$$

- Independence assumption not truly satisfied
  - Doesn’t cause serious problems
  - More advanced models are possible, e.g. time series of word observations

News monitoring

- Find news articles which are predictive of a change in company stock
- Population A: accompanied by no change in stock
- Population B: accompanied by large change in stock
- Fawcett and Provost, 1999

Class priors

- Some classes are more probable than others, even before we see the sample: $p(\text{class})$
- Use Bayes' theorem: $p(\text{class} \mid \text{sample}) = \dfrac{p(\text{sample} \mid \text{class})\, p(\text{class})}{\sum_{\text{class}} p(\text{sample} \mid \text{class})\, p(\text{class})}$
- Choose most probable class
- Same as most likely class if priors are equal

Image classification procedure

- Collect all images labeled "tiger" into single batch of pixels
- RGB values are quantized into about 64 colors
- Compute probability histogram (64 bins)

Word	Prob.	Word	Prob.
green	.0619	orange	.0023
black	.0332	brown	.0024
…	…	purple	.00004

Costs

- Different classification errors may have different costs
- E.g. classifying nuclear reactor as "stable" when it isn't
- Cost of saying A when truth is B: $C(A \mid B)$
- Choose class which minimizes

$$C(A \mid \text{sample}) = \sum_B C(A \mid B)\, p(B \mid \text{sample})$$

Image classification procedure

- To classify an image, sum over all 64 colors:

$$\log L(\text{tiger}) = \sum_c n_c \log \hat{p}(c \mid \text{tiger})$$

- Independence assumption not satisfied
  - More complex image models possible

Image classification

- Stock photos: tigers, flowers, boats, etc.
- An image is a sample of pixels from a pixel population: tiger pixels, flower pixels, etc.
- Populations overlap, but emphasize different colors
- "Color histogram classification"
- Simple, effective

Labeled images

Tiger

Tiger

Flower

Flower

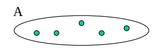
Unlabeled images

More complex image model

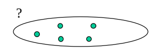
- Schneiderman & Kanade (2000)
- An image is a set of sub-images sampled from a population of sub-images
- One histogram bin for every possible sub-image, after quantization (about 6,561 bins)
- Requires huge amounts of labeled data

Anomaly detection

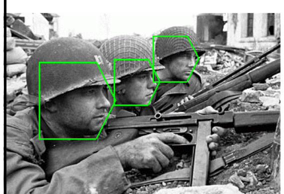
- Given a sample from a population:



- Determine if an unlabeled sample is likely to be from the same population



Face, car detection



Solution

- Estimate distribution of the population:
  $\hat{p}(x \mid \mathrm{A})$
- Given new sample, compute probability it could have arisen from A:
  $p(\text{sample} \mid \text{pop})$
- If probability is too small, the sample is anomalous: $p(\text{sample} \mid \text{pop}) < t$

Machine learning methods

- Often based on simple statistical models (or equivalent)
- Tend to ignore inference issues, proper estimation, model checking
- Main issues are computation, object representation

Choosing the threshold

- Low threshold = missed anomalies
- High threshold = false positives
- Generally t is set as high as tolerable
- Resampling training set gives expected number of false positives

Applications of anomaly detection

- Similarity
  - Retrieving similar documents, images
  - Query by example
- Dissimilarity
  - Activity monitoring, surveillance
  - Fraud detection
  - Computer intrusions

Potentially frauded account

Time	Day	Length	From	To	Fraud?
10am	Mon	13m	NY	CT	
3pm	Fri	5m	NY	NY	
1pm	Tue	9m	NY	CT	
2am	Wed	35s	MA	NY	Y
9pm	Thu	24s	MA	MA	Y

Cellular cloning fraud

- Cellphones continually broadcast their serial number and customer ID, without encryption
- Inexpensive equipment can catch these numbers and program a second phone to use them
- Free, untraceable calls!
- Even PINs are unencrypted

Caller profiling

- Make categorical variable x ranging over (time, location) combinations
- Compute probability histogram of x for each customer:

(Time, Location)	Prob.
(9am, NY)	.12
(5pm, NY)	.09
(10pm, NY)	.01
(11pm, MA)	.001

Catching fraud

- Classification doesn't work
  - Bandit population isn't distinct from legitimate population
  - An unusual call for you is typical for me
- Must spot differences from a customer's profile
- Individual calls are not enough evidence
  - Must use batches

Fraud detection

- Compute probability of today's calls:

$$p(\text{calls} \mid \text{profile}) = \prod p(\text{call } i \mid \text{profile})$$

- Flag account if $p(\text{calls} \mid \text{profile}) < t$
- Choose t based on size of fraud dept.
- Can also incorporate potential cost of fraud
- Calls are not really independent

UNIX intrusion

- Prior to ssh, telnet had same problems as cellphones
- Security holes allow crackers to log in as legitimate users
- Must spot differences from user's profile
  - Which commands are used

Recurring problem

- Most applications require reducing the number of bins (quantization)
  - Words, colors, times, locations, UNIX commands
- For computational as well as estimation reasons
- What is best way to reduce?

Catching intrusion

- Make categorical variable ranging over UNIX commands
- Compute probability histogram for each user:

Command	Prob.
gs	.03
gcc	.005
kill	.0001
ps	.0001

Catching intrusion

- Each login session is sequence of commands
- For each session, compute

$$p(\text{commands} \mid \text{profile}) = \prod p(\text{command } i \mid \text{profile})$$

- Fawcett and Provost, 1999