Information Theory and Networks

Lecture 17: Gambling with Side Information

Paul Tune
<paul.tune@adelaide.edu.au>
http://www.maths.adelaide.edu.au/matthew.roughan/Lecture_notes/InformationTheory/

School of Mathematical Sciences,
University of Adelaide

October 9, 2013

Part I

Gambling with Side Information

> A good hockey player plays where the puck is. A great hockey player plays where the puck is going to be.
>
> *Wayne Gretzky*

Section 1

More about Horse Racing

Horse Racing Redux

- Suppose you know: horse 3 is an older horse, fatigues easily
  - how has your edge changed?
  - what strategy should you employ?

Horse	Odds
1	10
2	2
3	20
4	5

Horse Racing Redux
Suppose you know: horse 3 is an older horse, fatigues easily
how has your edge changed?
what strategy should you employ?

Background

- Kelly's original paper talks about "private wire"
- AT&T's main customers were horse racing rackets
  - transmit race results from East to West Coast
  - some races allow bets up until the results
  - lag between East and West Coast in taking bets
- Mostly mob controlled
- Title change to paper to remove "unsavoury" elements

Background
Kelly's original paper talks about "private wire"
AT&T's main customers were horse racing rackets
transmit race results from East to West Coast
some races allow bets up until the results
lag between East and West Coast in taking bets
Mostly mob controlled
Title change to paper to remove "unsavoury" elements

Reinterpretation of Doubling Rate

- Write $r_i = 1/o_i$, $\mathbf{r}$ is the bookie's estimate of horse win probabilities
  - technically, this has been determined by the bettors themselves
- Recall doubling rate: $W(\mathbf{b}, \mathbf{p}) = \sum_i p_i \log b_i o_i$
- Similarly, $W(\mathbf{b}, \mathbf{p}) = D(\mathbf{p} \,||\, \mathbf{r}) - D(\mathbf{p} \,||\, \mathbf{b})$
  - comparison between estimates of the true winning distribution between the bookie and gambler
  - when does the gambler do better?
- Special case – uniform odds: $W^*(\mathbf{p}) = D(\mathbf{p} \,||\, \frac{1}{m}\mathbf{1}) = \log m - H(\mathbf{p})$

Reinterpretation of Doubling Rate
- Write $r_i = 1/o_i$, $\mathbf{r}$ is the bookie's estimate of horse win probabilities
  - technically, this has been determined by the bettors themselves
- Recall doubling rate: $W(\mathbf{b}, \mathbf{p}) = \sum_i p_i \log b_i o_i$
- Similarly, $W(\mathbf{b}, \mathbf{p}) = D(\mathbf{p} \,||\, \mathbf{r}) - D(\mathbf{p} \,||\, \mathbf{b})$
  - comparison between estimates of the true winning distribution between the bookie and gambler
  - when does the gambler do better?
- Special case – uniform odds: $W^*(\mathbf{p}) = D(\mathbf{p} \,||\, \frac{1}{m}\mathbf{1}) = \log m - H(\mathbf{p})$

Section 2

Side Information

Section 2
Side Information

Incorporating Side Information

- Based on reinterpretation, want to minimise KL divergence
  - any form of side information can provide better estimates
- Let $X \in \{1, 2, \cdots, m\}$ denote the horse that wins the race
- Consider (X, Y), where Y is the side information
  - $p(x, y) = p(y)p(x|y)$ is the joint distribution
  - betting $b(x|y) \geq 0$, $\sum_x b(x|y) = 1$
  - given $Y = y$, now want to estimate $p(x \mid y)$
  - clearly, the better the estimate, the better wealth growth rate

As an aside, you can see here how gambling is very much tied with estimation. Later on, we see how gambling, data compression and estimation are all linked.

Effect on Doubling Rate

- Unconditional doubling rate

$$W^*(X) := \max_{\mathbf{b}(x)} \sum_x p(x) \log b(x)o(x)$$

- Conditional doubling rate

$$W^*(X|Y) := \max_{\mathbf{b}(x|y)} \sum_{x,y} p(x, y) \log b(x|y)o(x)$$

- Want to find the bound on the increase $\Delta W = W^*(X|Y) - W^*(X)$
- Turns out: $\Delta W = I(X; Y)$
  - by Kelly, $b^*(x|y) = p(x|y)$
  - calculate $W^*(X|Y = y)$, then compute $W^*(X|Y)$, then take difference
- In turn, this is upper bounded by the **channel capacity**

Kelly's original paper essentially showed that the increase in the doubling rate of a gambler with a private wire is bounded above by the channel capacity of the private wire i.e. the optimal increase ΔW^* is equal to the channel capacity. More on capacity in later lectures.

Dependent Horse Races

- Side information can come from past performance of the horses
  - if horse is performing well consistently, then more likely for it to win
- For each race i, bet conditionally (fair odds)
  - $b^*(x_i|x_{i-1},\cdots,x_1) = p(x_i|x_{i-1},\cdots,x_1)$
- Let's assume fair odds (m-for-1), then after n races,

$$\frac{1}{n}E[\log S_n] = \log m - \frac{H(X_1, X_2, \cdots, X_n)}{n}$$

- Link this with entropy rate by taking $n \to \infty$

$$\lim_{n\to\infty} \frac{1}{n}E[\log S_n] + H(\mathcal{X}) = \log m$$

- Expectation can be removed if S_n is ergodic (property holds w.p. 1)

In practice, odds are, in general, subfair. In that case, remember that the highest returns come from odds that are mispriced. You have to know something the crowd doesn't know to do much better. In our parimutuel horse race table of odds example, if you know something about horse 3's advantage on the track that other people don't know, placing most of your mass on horse 3 will return the most amount of money. Kelly's criterion for subfair odds is based on a water-filling algorithm. In other words, a bet is placed only if one's edge is higher than the odds. Similar ideas can be applied to the stock market, if you ignore the efficient market hypothesis: find the mispriced stock.

Betting Sequentially vs. Once-off

- Consider a card game: *red and black*
  - a deck of 52 cards, 26 red, 26 black
  - gambler places bets on whether the next card is red or black
  - payout: 2-for-1 (fair for equally probably red/black cards)
- Play this sequentially
  - what are the proportions we should bet? (hint: use past information)
- Play this once-off for all $\binom{52}{26}$ sequences
  - proportional betting allocates $1/\binom{52}{26}$ wealth on each sequence
- Both schemes are equivalent: why?

$$S_{52}^* = \frac{2^{52}}{\binom{52}{26}} = 9.08$$

- Return does not depend on actual sequence: sequences are typical (c.f. AEP)

Part II

Data Compression and Gambling

Part II

Data Compression and Gambling

Gambling-Based Compression

- Consider $X_1, X_2, \cdots, X_n$ a sequence of binary random variables to compress
- Gambling allocations are $b(x_{k+1}|x_1, x_2, \cdots, x_k) \geq 0$ with

$$\sum_{x_{k+1}} b(x_{k+1}|x_1, x_2, \cdots, x_k) = 1$$

- Odds: uniform 2-for-1
- Wealth:

$$S_n = 2^n \prod_{k=1}^{n} b(x_{k+1}|x_1, x_2, \cdots, x_k) = 2^n b(x_1, x_2, \cdots, x_n)$$

- **Idea**: use $b(x_1, x_2, \cdots, x_n)$ as a proxy for $p(x_1, x_2, \cdots, x_n)$, if S_n is maximised, then have log-optimal and best compression

Gambling-Based Compression

- Consider $X_1, X_2, \cdots, X_n$ a sequence of binary random variables to compress
- Gambling allocations are $b(x_{k+1}|x_1, x_2, \cdots, x_k) \geq 0$ with

$$\sum_{x_{k+1}} b(x_{k+1}|x_1, x_2, \cdots, x_k) = 1$$

- Odds: uniform 2-for-1
- Wealth:

$$S_n = 2^n \prod_{k=1}^{n} b(x_{k+1}|x_1, x_2, \cdots, x_k) = 2^n b(x_1, x_2, \cdots, x_n)$$

- **Idea**: use $b(x_1, x_2, \cdots, x_n)$ as a proxy for $p(x_1, x_2, \cdots, x_n)$, if S_n is maximised, then have log-optimal and best compression

Algorithm: Encoding

- Assumption: both encoder and decoder knows n
- **Encoding**:
  - arrange 2^n sequences lexicographically
  - sees $x(n)$, calculate wealth $S_n(x'(n))$ for all $x'(n) \leq x(n)$
  - compute $F(x(n)) = \sum_{x'(n) \leq x(n)} 2^{-n} S_n(x'(n))$, where $F(x(n)) \in [0, 1]$
  - express $F(x(n))$ in binary decimal to $k = \lceil n - \log S_n(x(n)) \rceil$ accuracy
  - codeword of $F(x(n))$: $.c_1 c_2 \cdots c_k$
  - the sequence $c(k) = (c_1, c_2, \cdots, c_k)$ is transmitted to the decoder

Notice the similarity of the algorithm to arithmetic coding. Here, we want our bets $b(x_1, x_2, \cdots, x_n)$ be a close estimate to the underlying distribution.

Algorithm: Decoding

- **Decoding**:
  - computes all $S_n(x'(n))$ for all 2^n sequences exactly; knows $F(x'(n))$ for any $x'(n)$
  - calculate $F(x'(n))$ in lexicographical ordering until first time output exceeds $.c(k)$: determines index
  - size of $2^{-n}S(x(n))$ ensures uniqueness: no other $x'(n)$ will have this wealth value
- Bits required: k, bits saved: $n - k = \lfloor \log(S_n(x(n))) \rfloor$
- With proportional gambling, $S_n(x(n)) = 2^n p(x(n))$, so $E[k] \leq H(X_1, X_2, \cdots, X_n) + 1$

Estimating Entropy of English

- Use the algorithm to estimate the entropy per letter of English
- Odds: 27-for-1 (including space, but no punctuations)
- Wealth: $S_n = (27)^n b(x_1, x_2, \cdots, x_n)$
- After n rounds of betting

$$E\left[\frac{1}{n}\log S_n\right] \leq \log 27 - H(\mathcal{X})$$

- Assuming English is ergodic, $\hat{H}(\mathcal{X}) = \log 27 - \frac{1}{n}\log S_n$ converges to $H(\mathcal{X})$ w.p. 1
- Example for "Jefferson the Virginian" gives 1.34 bits per letter

Further reading I

Thomas M. Cover and Joy A. Thomas, *Elements of information theory*, John Wiley and Sons, 1991.