# mercados preditivos

Lucas Mendes Marques Gonçalves

Universidade de São Paulo
Departamento de Ciência da Computação
Instituto de Matemática e Estatística

6 de julho de 2013

# Outline

# o problema

Diante de uma decisão em que vários agentes distintos tem informações relevantes, como extrair essa informação dos agentes ?

Temos que considerar, para essa extração, a possibilidade de

- alguns agentes terem informação superior aos demais
- alguns agentes desejarem viesar o resultado da agregação

# mercados preditivos

Uma das soluções possíveis para o problema de agregação é usar um mercado preditivo (prediction market).

Suponha que queremos estimar a temperatura global em 2020. Podemos, para isso, emitir um título. Em 2020, pagaremos por esse título a temperatura observada. Podemos tomar o preço desse título em 2013 como uma estimativa dessa temperatura futura

Mas quão boa será nossa estimativa ?

# mercados preditivos

Uma das soluções possíveis para o problema de agregação é usar um mercado preditivo (prediction market).

Suponha que queremos estimar a temperatura global em 2020. Podemos, para isso, emitir um título. Em 2020, pagaremos por esse título a temperatura observada. Podemos tomar o preço desse título em 2013 como uma estimativa dessa temperatura futura

Mas quão boa será nossa estimativa ?

# qualidade das estimativas

Alguns motivos empíricos para usar mercados preditivos:

- tem tido desempenho melhor do que outros métodos (como pesquisas de opinião ou mesmo estimativas de experts), tanto em laboratório quanto em situações reais
- tendem a convergir para uma estimativa correta mesmo quando muitos jogadores não tem informação relevante.
- tendem a convergir para uma estimativa correta mesmo quando há jogadores tentando manipular o resultado.

# a partição do conhecimento

Usaremos o seguinte modelo para a distribuição de informação entre os jogadores

- existem $\Omega$ estados possíveis do mundo ($\omega_1, \omega_2$ ...), dentre os quais apenas um ($\omega$) é verdadeiro
- cada jogador $j$ sabe distinguir entre um conjunto $X_j \subset \Omega$ e o seu complemento
- há uma distribuição a priori, de conhecimento público, sobre os estados de $\Omega$
- desejamos obter o valor esperado de uma função $f(\omega)$

# equilíbrio de expectativas racionais

No nosso modelo, podemos definir um preço de equilíbrio (rational expectations equilibrium) para cada estado $\omega_i$ possível

Assumindo que os jogadores:

- desejam maximizar seu ganho esperado
- são completamente indiferentes ao risco

Afirmamos que para cada estado $\omega_i$ possível, temos um $p^*(\omega_i)$ tal que, diante desse preço, todo jogador avalia o título exatamente nesse preço.

# equilíbrio de expectativas racionais

- em diversas situações, podemos provar não só que há o preço de equilíbrio, mas que ele revela toda a informação (i.e., que $p^*(\omega_i) \neq p^*(\omega_j)$, se $i \neq j$)
- mas em breve veremos um caso em que isso não é verdade
- além disso, será que o mercado vai efetivamente convergir para esse equilíbrio ?
- teoremas de não-negociação (no-trade theorems)

# equilíbrio de expectativas racionais

- em diversas situações, podemos provar não só que há o preço de equilíbrio, mas que ele revela toda a informação (i.e., que $p^*(\omega_i) \neq p^*(\omega_j)$, se $i \neq j$)
- mas em breve veremos um caso em que isso não é verdade
- além disso, será que o mercado vai efetivamente convergir para esse equilíbrio ?
- teoremas de não-negociação (no-trade theorems)

# equilíbrio de expectativas racionais

- em diversas situações, podemos provar não só que há o preço de equilíbrio, mas que ele revela toda a informação (i.e., que $p^*(\omega_i) \neq p^*(\omega_j)$, se $i \neq j$)
- mas em breve veremos um caso em que isso não é verdade
- além disso, será que o mercado vai efetivamente convergir para esse equilíbrio ?
- teoremas de não-negociação (no-trade theorems)

# equilíbrio de expectativas racionais

- em diversas situações, podemos provar não só que há o preço de equilíbrio, mas que ele revela toda a informação (i.e., que $p^*(\omega_i) \neq p^*(\omega_j)$, se $i \neq j$)
- mas em breve veremos um caso em que isso não é verdade
- além disso, será que o mercado vai efetivamente convergir para esse equilíbrio ?
- teoremas de não-negociação (no-trade theorems)

# problemas abordados no capítulo

- como prever distribuições conjuntas de variáveis binárias ?
  (i.e. como lidar com um número exponencial de títulos)
- como subsidiar um mercado preditivo para obter bons resultados ?
- que tipos de resultados podemos esperar de um mercado preditivo com um único título? Ele consegue capturar toda a informação disponível ?
  (não conseguiremos extrair toda a informação disponível aos jogadores, mesmo na mais ideal das situações)

# problemas abordados no capítulo

- como prever distribuições conjuntas de variáveis binárias ? (i.e. como lidar com um número exponencial de títulos)
- como subsidiar um mercado preditivo para obter bons resultados ?
- que tipos de resultados podemos esperar de um mercado preditivo com um único título? Ele consegue capturar toda a informação disponível ?
  (não conseguiremos extrair toda a informação disponível aos jogadores, mesmo na mais ideal das situações)

# problemas abordados no capítulo

- como prever distribuições conjuntas de variáveis binárias ? (i.e. como lidar com um número exponencial de títulos)
- como subsidiar um mercado preditivo para obter bons resultados ?
- que tipos de resultados podemos esperar de um mercado preditivo com um único título? Ele consegue capturar toda a informação disponível ?
  (não conseguiremos extrair toda a informação disponível aos jogadores, mesmo na mais ideal das situações)

# problemas abordados no capítulo

- como prever distribuições conjuntas de variáveis binárias ? (i.e. como lidar com um número exponencial de títulos)
- como subsidiar um mercado preditivo para obter bons resultados ?
- que tipos de resultados podemos esperar de um mercado preditivo com um único título? Ele consegue capturar toda a informação disponível ?
  (não conseguiremos extrair toda a informação disponível aos jogadores, mesmo na mais ideal das situações)

# problemas abordados no capítulo

- como prever distribuições conjuntas de variáveis binárias ? (i.e. como lidar com um número exponencial de títulos)
- como subsidiar um mercado preditivo para obter bons resultados ?
- que tipos de resultados podemos esperar de um mercado preditivo com um único título? Ele consegue capturar toda a informação disponível ?
  (não conseguiremos extrair toda a informação disponível aos jogadores, mesmo na mais ideal das situações)

# um mercado bem ideal

- queremos calcular uma função booleana $f(x)$, onde x é um vetor de binários
- cada jogador $j$ conhece o bit $x_j$ (há $|x|$ jogadores)
- existe uma distribuição a priori para $x$, que todo jogador conhece
- nosso título valerá $f(x)$ quando a função for revelada

# um mercado bem ideal

- a cada turno, cada jogador declara um preço $p_j$ para o título, e oferece um título
- o preço $p$, desse turno, será $\frac{\sum p_j}{|x|}$
- os jogadores com $p_j > p$ compram frações de títulos. Quando $p_j < p$, vendem
- supomos os jogadores completamente previsíveis: cada $j$ coloca $p_j$ como o seu valor esperado de $f(x)$ dados $x_j$ e os preços das rodadas anteriores
- no fim da rodada, observando o preço final, os jogadores atualizam suas crenças, eliminando os $x$ que não poderiam ocorrer, dado esse preço (e a previsibilidade dos demais jogadores)

# atualizando crenças

Quando observamos um preço $p$, podemos eliminar $x$ impossíveis: aqueles que não resultariam nesse preço.

Chamamos $S$ o conjunto de $x$ que ainda são possíveis após a observação do preço, e $S_j$ o conjunto de $x$ possíveis para o jogador $j$ (que, além dos preços, conhece seu bit $x_j$)

# computando uma função

Seja $f(x_1 x_2) = x_1 \vee x_2$ (e a distribuição sobre x é uniforme)
Suponhamos $x_1 = 1$ e $x_2 = 0$

- para o jogador 1, $S_1 = \{10, 11\}$ e $P[f(x) = 1] = 1$. Assim, ele oferecerá o preço $p_1 = 1$
- para o jogador 2, $S_2 = \{10, 00\}$ e $P[f(x) = 1] = \frac{1}{2}$. Assim, ele oferecerá o preço $p_2 = \frac{1}{2}$
- teremos, então $p = \frac{3}{4}$
- no inicio da segunda rodada, temos $S = \{10, 01\}$, $S_1 = \{10\}$ e $S_2 = \{10\}$
- assim $p_1 = p_2 = 1$, $p = 1$. Esse valor se manterá para sempre

# computando uma função

Seja $f(x_1 x_2) = x_1 \vee x_2$ (e a distribuição sobre x é uniforme)
Suponhamos $x_1 = 1$ e $x_2 = 0$

- para o jogador 1, $S_1 = \{10, 11\}$ e $P[f(x) = 1] = 1$. Assim, ele oferecerá o preço $p_1 = 1$
- para o jogador 2, $S_2 = \{10, 00\}$ e $P[f(x) = 1] = \frac{1}{2}$. Assim, ele oferecerá o preço $p_2 = \frac{1}{2}$
- teremos, então $p = \frac{3}{4}$
- no inicio da segunda rodada, temos $S = \{10, 01\}$, $S_1 = \{10\}$ e $S_2 = \{10\}$
- assim $p_1 = p_2 = 1$, $p = 1$. Esse valor se manterá para sempre

# computando uma função

Seja $f(x_1 x_2) = x_1 \vee x_2$ (e a distribuição sobre x é uniforme)
Suponhamos $x_1 = 1$ e $x_2 = 0$

- para o jogador 1, $S_1 = \{10, 11\}$ e $P[f(x) = 1] = 1$. Assim, ele oferecerá o preço $p_1 = 1$
- para o jogador 2, $S_2 = \{10, 00\}$ e $P[f(x) = 1] = \frac{1}{2}$. Assim, ele oferecerá o preço $p_2 = \frac{1}{2}$
- teremos, então $p = \frac{3}{4}$
- no inicio da segunda rodada, temos $S = \{10, 01\}$, $S_1 = \{10\}$ e $S_2 = \{10\}$
- assim $p_1 = p_2 = 1$, $p = 1$. Esse valor se manterá para sempre

# computando uma função

Seja $f(x_1 x_2) = x_1 \vee x_2$ (e a distribuição sobre x é uniforme)
Suponhamos $x_1 = 1$ e $x_2 = 0$

- para o jogador 1, $S_1 = \{10, 11\}$ e $P[f(x) = 1] = 1$. Assim, ele oferecerá o preço $p_1 = 1$
- para o jogador 2, $S_2 = \{10, 00\}$ e $P[f(x) = 1] = \frac{1}{2}$. Assim, ele oferecerá o preço $p_2 = \frac{1}{2}$
- teremos, então $p = \frac{3}{4}$
- no inicio da segunda rodada, temos $S = \{10, 01\}$, $S_1 = \{10\}$ e $S_2 = \{10\}$
- assim $p_1 = p_2 = 1$, $p = 1$. Esse valor se manterá para sempre

# computando uma função

Seja $f(x_1 x_2) = x_1 \vee x_2$ (e a distribuição sobre x é uniforme)
Suponhamos $x_1 = 1$ e $x_2 = 0$

- para o jogador 1, $S_1 = \{10, 11\}$ e $P[f(x) = 1] = 1$. Assim, ele oferecerá o preço $p_1 = 1$
- para o jogador 2, $S_2 = \{10, 00\}$ e $P[f(x) = 1] = \frac{1}{2}$. Assim, ele oferecerá o preço $p_2 = \frac{1}{2}$
- teremos, então $p = \frac{3}{4}$
- no inicio da segunda rodada, temos $S = \{10, 01\}$, $S_1 = \{10\}$ e $S_2 = \{10\}$
- assim $p_1 = p_2 = 1$, $p = 1$. Esse valor se manterá para sempre

# uma função mais difícil

Seja $f(x_1 x_2) = x_1 \oplus x_2$ (e a distribuição sobre x é uniforme)

- independente do valor de $x_1$, o jogador 1 calcula a probabilidade como $\frac{1}{2}$ (e o jogador 2 também)
- teremos, então $p = p_1 = p_2 = \frac{1}{2}$
- nenhum jogador ganhou nenhuma informação. Seus conjuntos $S_i$ permanecem
- assim, o preço se manterá para sempre

# uma função mais difícil

Seja $f(x_1 x_2) = x_1 \oplus x_2$ (e a distribuição sobre x é uniforme)

- independente do valor de $x_1$, o jogador 1 calcula a probabilidade como $\frac{1}{2}$ (e o jogador 2 também)
- teremos, então $p = p_1 = p_2 = \frac{1}{2}$
- nenhum jogador ganhou nenhuma informação. Seus conjuntos $S_i$ permanecem
- assim, o preço se manterá para sempre

# uma função mais difícil

Seja $f(x_1 x_2) = x_1 \oplus x_2$ (e a distribuição sobre x é uniforme)

- independente do valor de $x_1$, o jogador 1 calcula a probabilidade como $\frac{1}{2}$ (e o jogador 2 também)
- teremos, então $p = p_1 = p_2 = \frac{1}{2}$
- nenhum jogador ganhou nenhuma informação. Seus conjuntos $S_j$ permanecem
- assim, o preço se manterá para sempre

# uma função mais difícil

Seja $f(x_1 x_2) = x_1 \oplus x_2$ (e a distribuição sobre x é uniforme)

- independente do valor de $x_1$, o jogador 1 calcula a probabilidade como $\frac{1}{2}$ (e o jogador 2 também)
- teremos, então $p = p_1 = p_2 = \frac{1}{2}$
- nenhum jogador ganhou nenhuma informação. Seus conjuntos $S_j$ permanecem
- assim, o preço se manterá para sempre

# convergência

Denotemos $S_j^1$ o conhecimento inicial de $i$, $S_j^2 \subseteq S_j^1$ seu conhecimento no inicio da segunda rodada, e assim sucessivamente. Definamos $S^1, S^2 \ldots$ os conhecimentos comuns a todos em cada rodada

Como só há uma quantidade finita de possíveis $S_j^i$, é necessário que haja um $S_j$ final para cada $j$, que denotaremos $S_j^\infty$. $S^\infty$ é definido de forma análoga

# convergência

**Teorema**

*Em um mercado como o descrito, existe um preço final $p^\infty$ tal que*

- $P[f(x)|S^\infty] = p^\infty$
- $P[f(x)|S_j^\infty] = p^\infty$

(note que essas probabilidades correspondem a esperanças do valor de $f(x)$)

# caracterização das funções f(x)

Uma função $f(x)$ é dita uma função limiar ponderada (weighted threshold function) se existem constantes reais $w_0, w_1, w_2 \dots w_n$ tais que

$$f(x) = 1 \iff w_0 + \sum_{j=1}^{n} w_j x_j \geq 1$$

**Teorema**

*Se $f$ é uma função limiar ponderada, então, independente da distribuição de probabilidade a priori, $p^\infty = f(x)$*

**Teorema**

*Se $f$ não pode ser expressa como função limiar ponderada, existe alguma distribuição de probabilidade a priori tal que $p^\infty \neq f(x)$*

**Teorema**

*Se $f$ é uma função limiar ponderada, então, independente da distribuição de probabilidade a priori, $p^\infty = f(x)$*

(prova em sala. As definições e fatos a seguir serão úteis)

$$P(f(y) = 1 | y \in S^\infty) = p^\infty$$

$$P(f(y) = 1 | y \in S^\infty \wedge y_j = x_j) = p^\infty$$

$$L_j^+ = P(y_j = 1 | y \in S^\infty, f(y) = 1)$$

$$L_j^- = P(y_j = 1 | y \in S^\infty, f(y) = 0)$$

$$L^+ = \sum_{i=1}^{n} w_j L_j^+ \; ; L^- = \sum_{i=1}^{n} w_j L_j^-$$

**Teorema**

*Se $f$ não pode ser expressa como função limiar ponderada, existe alguma distribuição de probabilidade a priori tal que $p^\infty \neq f(x)$*

(prova em sala. As definições e fatos a seguir serão úteis)

$$H^+ = \text{fecho}(f^{-1}(1))$$
$$H^- = \text{fecho}(f^{-1}(0))$$

Se não há função limiar que expresse $f$, $H^+$ e $H^-$ se intersectam. Tome $x^*$ nessa intersecção

# outro exemplo de função não agregável

Uma outra função não agregável é $(x_1 \vee x_2) \wedge (x_3 \vee x_4)$

(prova em sala, usando o teorema)

# tempo de convergência

**Teorema**

*Se f é uma função limiar ponderada com $|x| = n$, independente da distribuição a priori, após n turnos o equilíbrio $p^\infty$ já terá sido atingido*

**Teorema**

*Há uma função limiar que, com $|x| = 2k$, pode levar k turnos para convergir para $p^\infty$*

# mercados preditivos combinatórios

Desejamos saber distribuições conjuntas sobre várias variáveis aleatórias binárias
ex: o preço da gasolina vai baixar e o do carvão aumentar ?

O tamanho do espaço de estados $\Omega$ é exponencial no tamanho do conjunto de eventos $\epsilon$ que desejamos considerar

Permitiremos aos agentes comprar ordens compostas: Títulos $S_f$ que rendem 1 se a função booleana f for verdadeira, e 0 caso contrário

# complicações

Permitiremos aos agentes comprar ordens compostas: Títulos $S_f$ que rendem 1 se a função booleana f for verdadeira, e 0 caso contrário

Isso introduz dois problemas:

- como permitir que dois agentes de fato façam suas operações, quando é improvável que eles queiram negociar o mesmo $S_f$ (problema de mercado magro, ou "thin market")
- como fazer isso de maneira computacionalmente tratável

Acabaremos resolvendo o problema do mercado magro, mas nossa solução não será tratável

# complicações

Permitiremos aos agentes comprar ordens compostas: Títulos $S_f$ que rendem 1 se a função booleana f for verdadeira, e 0 caso contrário

Isso introduz dois problemas:

- como permitir que dois agentes de fato façam suas operações, quando é improvável que eles queiram negociar o mesmo $S_f$ (problema de mercado magro, ou "thin market")
- como fazer isso de maneira computacionalmente tratável

Acabaremos resolvendo o problema do mercado magro, mas nossa solução não será tratável

# complicações

Permitiremos aos agentes comprar ordens compostas: Títulos $S_f$ que rendem 1 se a função booleana f for verdadeira, e 0 caso contrário

Isso introduz dois problemas:

- como permitir que dois agentes de fato façam suas operações, quando é improvável que eles queiram negociar o mesmo $S_f$ (problema de mercado magro, ou ”thin market”)
- como fazer isso de maneira computacionalmente tratável

Acabaremos resolvendo o problema do mercado magro, mas nossa solução não será tratável

# complicações

Permitiremos aos agentes comprar ordens compostas: Títulos $S_f$ que rendem 1 se a função booleana f for verdadeira, e 0 caso contrário
Isso introduz dois problemas:

- como permitir que dois agentes de fato façam suas operações, quando é improvável que eles queiram negociar o mesmo $S_f$ (problema de mercado magro, ou "thin market")
- como fazer isso de maneira computacionalmente tratável

Acabaremos resolvendo o problema do mercado magro, mas nossa solução não será tratável

# o problema de matching

Uma ”solução” seria simplesmente vender a quem quer que peça o título $S_f$ que ele quiser. Porém, dessa forma, o controlador do mercado está se expondo a muito risco.

Ao invés disso, tentaremos atender aos pedidos de forma que o controlador do mercado não possa ter prejuízo, não importa quais sejam os valores reais das variáveis binárias

Poderíamos só aceitar pedidos pareados (i.e., alguém pediu $S_f$, e outra pessoa $S_{\bar{f}}$), mas a estrutura do problema nos permite aceitar mais

# quanto ganha um agente

Consideremos uma ordem (compra ou venda de títulos) i, onde um agente está comprando (ou vendendo) um título $X_{f_i}$ por um preço $p_i$

Se i for uma compra, o lucro do agente será $1 - p_i$, se $f_i = 1$, e $-p_i$ caso contrário

Se i for uma venda, o lucro do agente será $p_i - 1$, se $f_i = 1$, e $p_i$ caso contrário

Representemos os possíveis lucros desse investimento em um vetor $\Upsilon_i$ indexado pelo espaço $\Omega = 2^{\epsilon}$

# exemplo

Consideremos um mercado, onde $\epsilon$ é o conjunto de duas variáveis binárias $X_1$ e $X_2$

- foi emitido um pedido de compra para $S_{X_1 \wedge X_2}$, ao preço 0.4
- foi emitido um pedido de venda para $S_{X_1}$, ao preço 0.3

Chamando o primeiro pedido de "1", temos
$\Upsilon_1 = (0.6, -0.4, -0, 4, -0.4)$
(respectivamente, os payoffs de 11, 10, 01 e 00)
Analogamente, para o segundo pedido,
$\Upsilon_2 = (-0.7, -0.7, 0.3, 0.3)$

# exemplo

$$\Upsilon_1 = (0.6, -0.4, -0,4, -0.4)$$

$$\Upsilon_2 = (-0.7, -0.7, 0.3, 0.3)$$

Observemos que, para esses dois pedidos, não existe estado do mundo em que os jogadores, juntos, tenham saído ganhando:
$\Upsilon_1 + \Upsilon_2 = (-0.1, -1.1, -0.1, -0.1)$
O controlador pode usar a estrutura lógica do problema para aceitar pedidos não pareados!

# exemplo

$$\Upsilon_1 = (0.6, -0.4, -0,4, -0.4)$$

$$\Upsilon_2 = (-0.7, -0.7, 0.3, 0.3)$$

Observemos que, para esses dois pedidos, não existe estado do mundo em que os jogadores, juntos, tenham saído ganhando:
$\Upsilon_1 + \Upsilon_2 = (-0.1, -1.1, -0.1, -0.1)$
O controlador pode usar a estrutura lógica do problema para aceitar pedidos não pareados!

# quanto perde o controlador

O ganho total dos agentes será o prejuízo do controlador do mercado.
Nosso objetivo, então, será garantir que $\sum \Upsilon_i \leq 0$, ou algo análogo, coordenada a coordenada

## Definição (matching divisível)

*Dado um conjunto de pedidos $\Upsilon_i$ , um matching divisível é um conjunto de $\alpha_i$ tais que:*

- $\sum \Upsilon_i \alpha_i \leq 0$
- $0 < \alpha_i < 1$
- *ao menos um $\alpha_i$ é não nulo*

# complexidade computacional

**Teorema**

*O problema de determinar, dadas n ordens, se há matching divisível é polinomial no caso de* $|\epsilon| = O(\log(n))$

**Teorema**

*O problema de determinar, dadas n ordens, se há matching divisível é co-NP-completo no caso geral*

# referências

- Joan Feigenbaum, Lance Fortnow, David M. Pennock, and Rahul Sami.
  Computation in a distributed information market, 2004.
- Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani.
  *Algorithmic Game Theory*.
  Cambridge University Press, New York, NY, USA, 2007.
  ISBN 0521872820.