Generalized Matroid Secretary Problem

Sourav Chakraborty
(Indian Statistical Institute)

The Art of Selling a Car

Want to sell my car.

The Art of Selling a Car

Want to sell my car.

2 L

The Art of Selling a Car

Want to sell my car.

2 L

The Art of Selling a Car

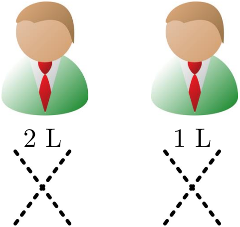
Want to sell my car.

2 L

1 L

The Art of Selling a Car

Want to sell my car.



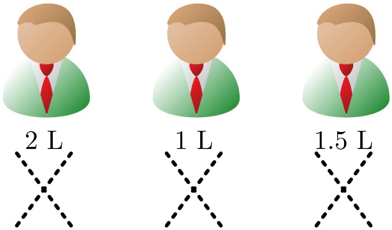
The Art of Selling a Car

Want to sell my car.

2 L 1 L 1.5 L

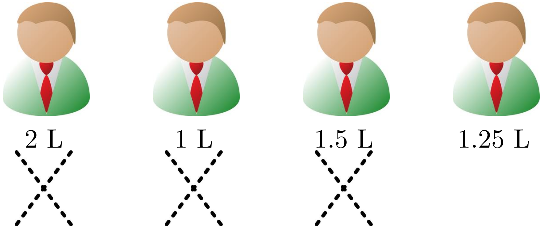
The Art of Selling a Car

Want to sell my car.



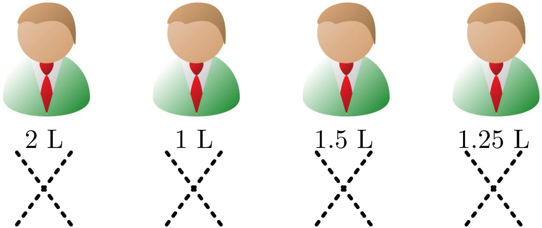
The Art of Selling a Car

Want to sell my car.



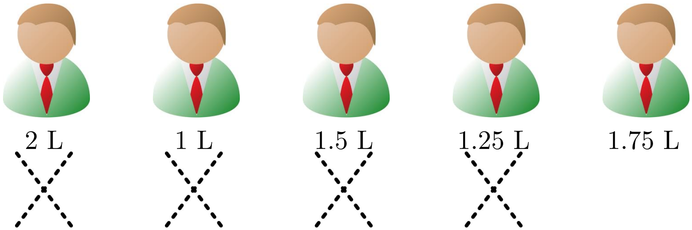
The Art of Selling a Car

Want to sell my car.



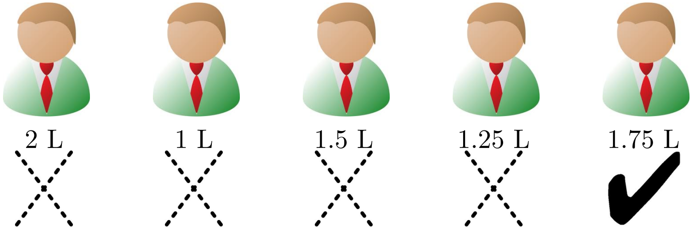
The Art of Selling a Car

Want to sell my car.



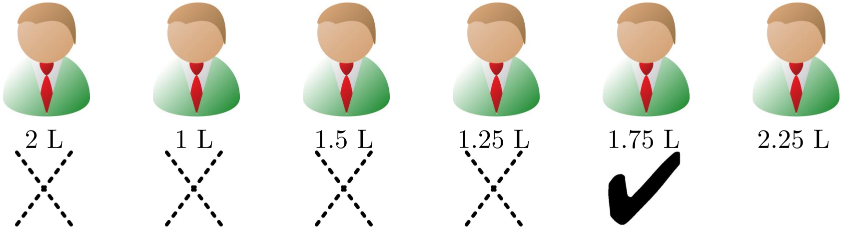
The Art of Selling a Car

Want to sell my car.



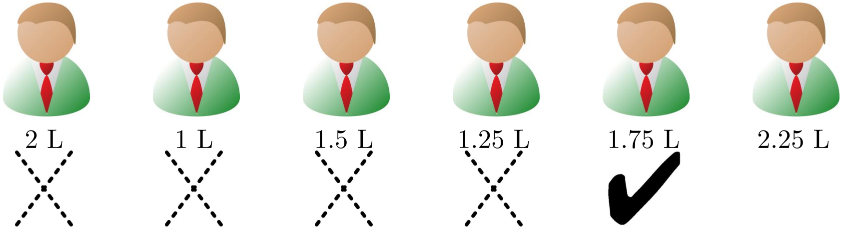
The Art of Selling a Car

Want to sell my car.



The Art of Selling a Car

Want to sell my car.



What should be my strategy?

Different Assumptions

Different Assumptions

- Are the bids chosen by an adversary or drawn from some distribution?

Different Assumptions

- Are the bids chosen by an adversary or drawn from some distribution?
- How much knowledge in advance do the seller have about the bids? (For example: do the seller know the distribution from which the bids are drawn.)

Different Assumptions

- Are the bids chosen by an adversary or drawn from some distribution?
- How much knowledge in advance do the seller have about the bids? (For example: do the seller know the distribution from which the bids are drawn.)
- Do the buyers come in a particular order or in some random order?

Different kind of assumptions

- (Pessimistic Assumption) All the bids and the ordering of the buyers are chosen by an adversary.

Different kind of assumptions

- (Pessimistic Assumption) All the bids and the ordering of the buyers are chosen by an adversary.
  For any strategy of the seller there is an adversarial strategy such that the expected return for the seller is $1/N$ of the best bid. (N is the number of bidders.)

Different kind of assumptions

- (Pessimistic Assumption) All the bids and the ordering of the buyers are chosen by an adversary.
  For any strategy of the seller there is an adversarial strategy such that the expected return for the seller is $1/N$ of the best bid. (N is the number of bidders.)
- (Optimistic Approach) All bids are drawn from a distribution (that is known to the seller) and the bidders come in a random order.

Different kind of assumptions

- (Pessimistic Assumption) All the bids and the ordering of the buyers are chosen by an adversary.
  For any strategy of the seller there is an adversarial strategy such that the expected return for the seller is $1/N$ of the best bid. (N is the number of bidders.)
- (Optimistic Approach) All bids are drawn from a distribution (that is known to the seller) and the bidders come in a random order.
  There is a simple strategy the guarantees that the expected return is at least half of the best bid.

Different kind of assumptions

- (Pessimistic Assumption) All the bids and the ordering of the buyers are chosen by an adversary.
  For any strategy of the seller there is an adversarial strategy such that the expected return for the seller is $1/N$ of the best bid. (N is the number of bidders.)
- (Optimistic Approach) All bids are drawn from a distribution (that is known to the seller) and the bidders come in a random order.
  There is a simple strategy the guarantees that the expected return is at least half of the best bid.
- (Middle Path) The bids are chosen by an adversary but the bidders come in a random order.

Different kind of assumptions

- (Pessimistic Assumption) All the bids and the ordering of the buyers are chosen by an adversary.
  For any strategy of the seller there is an adversarial strategy such that the expected return for the seller is $1/N$ of the best bid.
  (N is the number of bidders.)
- (Optimistic Approach) All bids are drawn from a distribution (that is known to the seller) and the bidders come in a random order.
  There is a simple strategy the guarantees that the expected return is at least half of the best bid.
- (Middle Path) The bids are chosen by an adversary but the bidders come in a random order.
  Secretary Problem

The Secretary Problem

The Secretary Problem

Bids: $B_1, B_2, \ldots, B_N$

The Secretary Problem

Bids: $B_1, B_2, \ldots, B_N$

- Bids are adversarially chosen but come in a random order. Bids come is a online fashion.

The Secretary Problem

Bids: $B_1, B_2, \ldots, B_N$

- Bids are adversarially chosen but come in a random order. Bids come is a online fashion.
- Once bid B_i is seen, the algorithm has to either REJECTS or SELECTS.

The Secretary Problem

Bids: $B_1, B_2, \ldots, B_N$

- Bids are adversarially chosen but come in a random order. Bids come is a online fashion.
- Once bid B_i is seen, the algorithm has to either REJECTS or SELECTS.
- If the algorithm REJECTS it then the bid B_i is lost forever and cannot be selected later.

The Secretary Problem

Bids: $B_1, B_2, \ldots, B_N$

- Bids are adversarially chosen but come in a random order. Bids come is a online fashion.
- Once bid B_i is seen, the algorithm has to either REJECTS or SELECTS.
- If the algorithm REJECTS it then the bid B_i is lost forever and cannot be selected later.
- If the algorithm SELECTS it then B_i is the return the algorithm get.

The Secretary Problem

Bids: $B_1, B_2, \ldots, B_N$

- Bids are adversarially chosen but come in a random order. Bids come is a online fashion.
- Once bid B_i is seen, the algorithm has to either REJECTS or SELECTS.
- If the algorithm REJECTS it then the bid B_i is lost forever and cannot be selected later.
- If the algorithm SELECTS it then B_i is the return the algorithm get.

GOAL: To maximize $\frac{\text{Expected Return}}{\max_i B_i}$.

Algorithm for Secretary Problem

Simple Algorithm

REJECT the bids $B_1, B_2, \ldots, B_{N/2}$
Let $C = \max\{B_1, \ldots, B_{N/2}\}$.
For any $i > N/2$ if B_i is at least C then ACCEPT B_i.

With probability $1/4$ the highest bid is the second half and the second highest bid is in the first half. So,

$$\frac{\text{Expected Return}}{\max_i B_i} > \frac{1}{4}.$$

[Lindley, Dynkin (1963)] showed that the competitive ratio is $1/e$.

Lets Sell Flight Tickets

Lets Sell Flight Tickets

Available seats

Lets Sell Flight Tickets

Available seats

- Chennai-London (20)

Lets Sell Flight Tickets

Available seats

- Chennai-London (20)
- London-Paris (5)
- Paris-NY (2)
- Mumbai-Paris (10)
- Chennai-Mumbai (40)
- London-NY (15)
- London-Mumbai (4)

Lets Sell Flight Tickets

Available seats

- Chennai-London (20)
- London-Paris (5)
- Paris-NY (2)
- Mumbai-Paris (10)
- Chennai-Mumbai (40)
- London-NY (15)
- London-Mumbai (4)

Bidders

B1 Chennai-London-Mumbai ($300)

Lets Sell Flight Tickets

Available seats

- Chennai-London (20)
- London-Paris (5)
- Paris-NY (2)
- Mumbai-Paris (10)
- Chennai-Mumbai (40)
- London-NY (15)
- London-Mumbai (4)

Bidders

B1 Chennai-London-Mumbai ($300) NO

Lets Sell Flight Tickets

Available seats

- Chennai-London (20)
- London-Paris (5)
- Paris-NY (2)
- Mumbai-Paris (10)
- Chennai-Mumbai (40)
- London-NY (15)
- London-Mumbai (4)

Bidders

B1 Chennai-London-Mumbai ($300) NO

B2 London-Paris ($100)

Lets Sell Flight Tickets

Available seats

- Chennai-London (20)
- London-Paris (5)
- Paris-NY (2)
- Mumbai-Paris (10)
- Chennai-Mumbai (40)
- London-NY (15)
- London-Mumbai (4)

Bidders

B1 Chennai-London-Mumbai ($300) NO

B2 London-Paris ($100) YES

Lets Sell Flight Tickets

Available seats

- Chennai-London (20)
- London-Paris (5)
- Paris-NY (2)
- Mumbai-Paris (10)
- Chennai-Mumbai (40)
- London-NY (15)
- London-Mumbai (4)

Bidders

B1 Chennai-London-Mumbai (\$300) NO

B2 London-Paris (\$100) YES

B3 Mumbai-Paris-NY (\$700)

Lets Sell Flight Tickets

Available seats

- Chennai-London (20)
- London-Paris (5)
- Paris-NY (2)
- Mumbai-Paris (10)
- Chennai-Mumbai (40)
- London-NY (15)
- London-Mumbai (4)

Bidders

B1 Chennai-London-Mumbai (\$300) NO

B2 London-Paris (\$100) YES

B3 Mumbai-Paris-NY (\$700) YES

Lets Sell Flight Tickets

Available seats

- Chennai-London (20)
- London-Paris (5)
- Paris-NY (2)
- Mumbai-Paris (10)
- Chennai-Mumbai (40)
- London-NY (15)
- London-Mumbai (4)

Bidders

B1 Chennai-London-Mumbai ($300) NO

B2 London-Paris ($100) YES

B3 Mumbai-Paris-NY ($700) YES

B4 Mumbai-Paris-NY ($800)

Lets Sell Flight Tickets

Available seats

- Chennai-London (20)
- London-Paris (5)
- Paris-NY (2)
- Mumbai-Paris (10)
- Chennai-Mumbai (40)
- London-NY (15)
- London-Mumbai (4)

Bidders

B1 Chennai-London-Mumbai ($300) NO

B2 London-Paris ($100) YES

B3 Mumbai-Paris-NY ($700) YES

B4 Mumbai-Paris-NY ($800) YES

Lets Sell Flight Tickets

Available seats

- Chennai-London (20)
- London-Paris (5)
- Paris-NY (2)
- Mumbai-Paris (10)
- Chennai-Mumbai (40)
- London-NY (15)
- London-Mumbai (4)

Bidders

B1 Chennai-London-Mumbai (\$300) NO

B2 London-Paris (\$100) YES

B3 Mumbai-Paris-NY (\$700) YES

B4 Mumbai-Paris-NY (\$800) YES

B5 Chennai-London-NY (\$950)

Lets Sell Flight Tickets

Available seats

- Chennai-London (20)
- London-Paris (5)
- Paris-NY (2)
- Mumbai-Paris (10)
- Chennai-Mumbai (40)
- London-NY (15)
- London-Mumbai (4)

Bidders

B1 Chennai-London-Mumbai (\$300) NO
B2 London-Paris (\$100) YES
B3 Mumbai-Paris-NY (\$700) YES
B4 Mumbai-Paris-NY (\$800) YES
B5 Chennai-London-NY (\$950) YES

Lets Sell Flight Tickets

Available seats

- Chennai-London (20)
- London-Paris (5)
- Paris-NY (2)
- Mumbai-Paris (10)
- Chennai-Mumbai (40)
- London-NY (15)
- London-Mumbai (4)

Bidders

B1 Chennai-London-Mumbai ($300) NO
B2 London-Paris ($100) YES
B3 Mumbai-Paris-NY ($700) YES
B4 Mumbai-Paris-NY ($800) YES
B5 Chennai-London-NY ($950) YES
B6 London-Paris-NY ($1200)
.. ....

Lets Sell Flight Tickets

Available seats

- Chennai-London (20)
- London-Paris (5)
- Paris-NY (2)
- Mumbai-Paris (10)
- Chennai-Mumbai (40)
- London-NY (15)
- London-Mumbai (4)

Bidders

B1 Chennai-London-Mumbai ($300) NO
B2 London-Paris ($100) YES
B3 Mumbai-Paris-NY ($700) YES
B4 Mumbai-Paris-NY ($800) YES
B5 Chennai-London-NY ($950) YES
B6 London-Paris-NY ($1200)
.. ....

What should be my strategy?

Generalized Matroid Problem

Generalized Matroid Problem

$(\mathcal{U}, \mathcal{I})$ is a matroid. $\mathcal{U}$ is a universe (all possible itineraries). $\mathcal{I}$ is the set of independent sets (allowed combinations of the elements of the universe).

$$\text{Bids: } (U_1, B_1), (U_2, B_2), \ldots, (U_N, B_N) \in (\mathcal{U}, \mathbb{R}^+)$$

Generalized Matroid Problem

$(\mathcal{U}, \mathcal{I})$ is a matroid. $\mathcal{U}$ is a universe (all possible itineraries). $\mathcal{I}$ is the set of independent sets (allowed combinations of the elements of the universe).

$$\text{Bids: } (U_1, B_1), (U_2, B_2), \ldots, (U_N, B_N) \in (\mathcal{U}, \mathbb{R}^+)$$

- Bids are adversarially chosen but come in a random order. Bids come is a online fashion.
- Once bid (U_i, B_i) is seen, the algorithm has to either REJECTS or SELECTS.

Generalized Matroid Problem

$(\mathcal{U}, \mathcal{I})$ is a matroid. $\mathcal{U}$ is a universe (all possible itineraries). $\mathcal{I}$ is the set of independent sets (allowed combinations of the elements of the universe).

$$\text{Bids: } (U_1, B_1), (U_2, B_2), \ldots, (U_N, B_N) \in (\mathcal{U}, \mathbb{R}^+)$$

- Bids are adversarially chosen but come in a random order. Bids come is a online fashion.
- Once bid (U_i, B_i) is seen, the algorithm has to either REJECTS or SELECTS.
- If the algorithm REJECTS it then the bid (U_i, B_i) is lost forever.

Generalized Matroid Problem

$(\mathcal{U}, \mathcal{I})$ is a matroid. $\mathcal{U}$ is a universe (all possible itineraries). $\mathcal{I}$ is the set of independent sets (allowed combinations of the elements of the universe).

$$\text{Bids: } (U_1, B_1), (U_2, B_2), \ldots, (U_N, B_N) \in (\mathcal{U}, \mathbb{R}^+)$$

- Bids are adversarially chosen but come in a random order. Bids come is a online fashion.
- Once bid (U_i, B_i) is seen, the algorithm has to either REJECTS or SELECTS.
- If the algorithm REJECTS it then the bid (U_i, B_i) is lost forever.
- If the algorithm SELECTS (U_i, B_i) then it adds U_i to its set S of selected items. S must be in $\mathcal{I}$.

Generalized Matroid Problem

$(\mathcal{U}, \mathcal{I})$ is a matroid. $\mathcal{U}$ is a universe (all possible itineraries). $\mathcal{I}$ is the set of independent sets (allowed combinations of the elements of the universe).

$$\text{Bids: } (U_1, B_1), (U_2, B_2), \ldots, (U_N, B_N) \in (\mathcal{U}, \mathbb{R}^+)$$

- Bids are adversarially chosen but come in a random order. Bids come is a online fashion.
- Once bid (U_i, B_i) is seen, the algorithm has to either REJECTS or SELECTS.
- If the algorithm REJECTS it then the bid (U_i, B_i) is lost forever.
- If the algorithm SELECTS (U_i, B_i) then it adds U_i to its set S of selected items. S must be in $\mathcal{I}$.
- The final return is the weight of S. Where, weight of a set S is $\sum_{i:U_i \in S} B_i$.

Generalized Matroid Problem

$(\mathcal{U}, \mathcal{I})$ is a matroid. $\mathcal{U}$ is a universe (all possible itineraries). $\mathcal{I}$ is the set of independent sets (allowed combinations of the elements of the universe).

$$\text{Bids: } (U_1, B_1), (U_2, B_2), \ldots, (U_N, B_N) \in (\mathcal{U}, \mathbb{R}^+)$$

- Bids are adversarially chosen but come in a random order. Bids come is a online fashion.
- Once bid (U_i, B_i) is seen, the algorithm has to either REJECTS or SELECTS.
- If the algorithm REJECTS it then the bid (U_i, B_i) is lost forever.
- If the algorithm SELECTS (U_i, B_i) then it adds U_i to its set S of selected items. S must be in $\mathcal{I}$.
- The final return is the weight of S. Where, weight of a set S is $\sum_{i:U_i \in S} B_i$.

$$\text{GOAL: To maximize } \frac{\text{Expected Return}}{\max_{S\in\mathcal{I}} \text{Weight of S}}.$$

Current Status

- Introduced by Babaioff et al (2007).

Current Status

- Introduced by Babaioff et al (2007).
- They also gave a strategy that has expected return at least $1/\log d$ of the OPT, where d is the dimension of the matroid (the size of the biggest set in $\mathcal{I}$).

Current Status

- Introduced by Babaioff et al (2007).
- They also gave a strategy that has expected return at least $1/\log d$ of the OPT, where d is the dimension of the matroid (the size of the biggest set in $\mathcal{I}$).
- Under various restrictions on the matroid structure or the bids better strategies have been designed. [Survey: Babaioff-Immorlica-Kempe-Klienberg *Online auctions and generalized secretary problems* (2008)]

Current Status

- Introduced by Babaioff et al (2007).
- They also gave a strategy that has expected return at least $1/\log d$ of the OPT, where d is the dimension of the matroid (the size of the biggest set in $\mathcal{I}$).
- Under various restrictions on the matroid structure or the bids better strategies have been designed. [Survey: Babaioff-Immorlica-Kempe-Klienberg *Online auctions and generalized secretary problems* (2008)]
- Even for general matroid the conjecture is constant competitive ratio.

Current Status

- Introduced by Babaioff et al (2007).
- They also gave a strategy that has expected return at least $1/\log d$ of the OPT, where d is the dimension of the matroid (the size of the biggest set in $\mathcal{I}$).
- Under various restrictions on the matroid structure or the bids better strategies have been designed. [Survey: Babaioff-Immorlica-Kempe-Klienberg *Online auctions and generalized secretary problems* (2008)]
- Even for general matroid the conjecture is constant competitive ratio.

Theorem (Chakraborty-Lachish 2012)

For the general matroid secretary problem there is a strategy such that the expected return is at least $1/\sqrt{\log d}$ of the OPT.

Current Status

Improvement of the Competitive Ratio

- $O(\log d)$ by Babaioff-Immorlica-Kempe-Klienberg 2008
- $O(\sqrt{\log d})$ by Chakraborty-Lachish 2012
- $O(\log \log d)$ by Lachish 2014
- $O(\log \log d)$ by Moran-Svensson-Zenklusen 2015

Better (even constant competitive ratio) algrotihms are known for special matroids.

Formal statement of the Problem - Matroid be a vector space

Formal statement of the Problem - Matroid be a vector space

V is a vector space of dimension d.

$$\text{Bids: } (v_1, w(v_1)), (v_2, w(v_2)), \ldots, (v_N, w(v_N)) \in (\mathcal{U}, \mathbb{R}^+)$$

Formal statement of the Problem - Matroid be a vector space

V is a vector space of dimension d.

$$\text{Bids: } (v_1, w(v_1)), (v_2, w(v_2)), \ldots, (v_N, w(v_N)) \in (\mathcal{U}, \mathbb{R}^+)$$

- Bids are adversarially chosen but come in a random order.
- Once bid $(v_i, w(v_i))$ is seen, the algorithm has to either REJECTS or SELECTS.
- If the algorithm REJECTS it then the bid $(v_i, w(v_i))$ is lost.
- If the algorithm SELECTS $(v_i, w(v_i))$ then it adds v_i to its set S of selected items. S must be an independent set.
- The final return is the weight of S. Where, weight of a set S is $\sum_{i:v_i \in S} w(v_i)$

Formal statement of the Problem - Matroid be a vector space

V is a vector space of dimension d.

$$\text{Bids: } (v_1, w(v_1)), (v_2, w(v_2)), \ldots, (v_N, w(v_N)) \in (\mathcal{U}, \mathbb{R}^+)$$

- Bids are adversarially chosen but come in a random order.
- Once bid $(v_i, w(v_i))$ is seen, the algorithm has to either REJECTS or SELECTS.
- If the algorithm REJECTS it then the bid $(v_i, w(v_i))$ is lost.
- If the algorithm SELECTS $(v_i, w(v_i))$ then it adds v_i to its set S of selected items. S must be an independent set.
- The final return is the weight of S. Where, weight of a set S is $\sum_{i:v_i \in S} w(v_i)$

$$\text{GOAL: To maximize } \frac{\text{Expected Return}}{\max_{S \in \mathcal{I}} \text{Weight of S}}.$$

Sampling and Selection Phase

Sampling and Selection Phase

V is a vector space of dimension d.

$$\text{Bids: } (v_1, w(v_1)), (v_2, w(v_2)), \ldots, (v_N, w(v_N)) \in (\mathcal{U}, \mathbb{R}^+)$$

Sampling Phase : Reject $(v_1, w(v_1)), (v_2, w(v_2)), \ldots, (v_{N/2}, w(v_{N/2}))$, but record all the data.

Selection Phase : Based on the data from the Sampling Phase decide which of the $(v_{N/2+1}, w(v_{N/2+1})), \ldots, (v_N, w(v_N))$ to select.

Simplifying the Problem

By loosing a constant in the competitive ratio we can assume the following:

Simplifying the Problem

By loosing a constant in the competitive ratio we can assume the following:

A1 Let M be the maximum weight of all the vectors. M is known to us.

Simplifying the Problem

By loosing a constant in the competitive ratio we can assume the following:

- A1 Let M be the maximum weight of all the vectors. M is known to us.
- A2 For all vectors v_i the weight $w(v_i)$ is one of $\{OPT, \frac{OPT}{2}, \frac{OPT}{4}, \ldots, \frac{OPT}{2^{\log d}}\}$.

Simplifying the Problem

By loosing a constant in the competitive ratio we can assume the following:

- A1 Let M be the maximum weight of all the vectors. M is known to us.
- A2 For all vectors v_i the weight $w(v_i)$ is one of $\{OPT, \frac{OPT}{2}, \frac{OPT}{4}, \ldots, \frac{OPT}{2^{\log d}}\}$.
- A3 Let L_i be the set of vectors in $(v_1, w(v_1)), \ldots, (v_{N/2}, w(v_{N/2}))$ that have weight $OPT/2^i$.

Simplifying the Problem

By loosing a constant in the competitive ratio we can assume the following:

- A1 Let M be the maximum weight of all the vectors. M is known to us.
- A2 For all vectors v_i the weight $w(v_i)$ is one of $\{OPT, \frac{OPT}{2}, \frac{OPT}{4}, \dots, \frac{OPT}{2^{\log d}}\}$.
- A3 Let L_i be the set of vectors in $(v_1, w(v_1)), \dots, (v_{N/2}, w(v_{N/2}))$ that have weight $OPT/2^i$.
- A3' Let R_i be the set of vectors in $(v_{N/2+1}, w(v_{N/2+1})), \dots, (v_N, w(v_N))$ that have weight $OPT/2^i$.

What is the OPT?

A1 Let $M < OPT/\sqrt{\log d}$ be the maximum weight of all the vectors. M is known to us.

A2 For all vectors v_i the weight $w(v_i)$ is one of $\{OPT, \frac{OPT}{2}, \frac{OPT}{4}, \ldots, \frac{OPT}{2^{\log d}}\}$.

A3 Let L_i be the set of vectors in $(v_1, w(v_1)), \ldots, (v_{N/2}, w(v_{N/2}))$ that have weight $OPT/2^i$.

A3' Let R_i be the set of vectors in $(v_{N/2+1}, w(v_{N/2+1})), \ldots, (v_N, w(v_N))$ that have weight $OPT/2^i$.

What is the OPT?

A1 Let $M < OPT/\sqrt{\log d}$ be the maximum weight of all the vectors. M is known to us.

A2 For all vectors v_i the weight $w(v_i)$ is one of $\{OPT, \frac{OPT}{2}, \frac{OPT}{4}, \ldots, \frac{OPT}{2^{\log d}}\}$.

A3 Let L_i be the set of vectors in $(v_1, w(v_1)), \ldots, (v_{N/2}, w(v_{N/2}))$ that have weight $OPT/2^i$.

A3' Let R_i be the set of vectors in $(v_{N/2+1}, w(v_{N/2+1})), \ldots, (v_N, w(v_N))$ that have weight $OPT/2^i$.

What is the OPT?

- A1 Let $M < OPT/\sqrt{\log d}$ be the maximum weight of all the vectors. M is known to us.
- A2 For all vectors v_i the weight $w(v_i)$ is one of $\{OPT, \frac{OPT}{2}, \frac{OPT}{4}, \ldots, \frac{OPT}{2^{\log d}}\}$.
- A3 Let L_i be the set of vectors in $(v_1, w(v_1)), \ldots, (v_{N/2}, w(v_{N/2}))$ that have weight $OPT/2^i$.
- A3' Let R_i be the set of vectors in $(v_{N/2+1}, w(v_{N/2+1})), \ldots, (v_N, w(v_N))$ that have weight $OPT/2^i$.

What is the OPT?

- A1 Let $M < OPT/\sqrt{\log d}$ be the maximum weight of all the vectors. M is known to us.
- A2 For all vectors v_i the weight $w(v_i)$ is one of $\{OPT, \frac{OPT}{2}, \frac{OPT}{4}, \ldots, \frac{OPT}{2^{\log d}}\}$.
- A3 Let L_i be the set of vectors in $(v_1, w(v_1)), \ldots, (v_{N/2}, w(v_{N/2}))$ that have weight $OPT/2^i$.
- A3' Let R_i be the set of vectors in $(v_{N/2+1}, w(v_{N/2+1})), \ldots, (v_N, w(v_N))$ that have weight $OPT/2^i$.

For all i the best offline algo on the first half will choose $d_i \triangleq dim(L_1 \cup L_2 \cup \cdots \cup L_i) - dim(L_1 \cup L_2 \cup \cdots \cup L_{i-1})$ number of vectors from layer L_i.

Main Hurdle: Measure on the interaction between layers

Let us only plan to SELECT elements from L_i and L_j.

Main Hurdle: Measure on the interaction between layers

Let us only plan to SELECT elements from L_i and L_j.
[Simple Case]: Let the $span(L_i)$ is disjoint from $span(L_j)$.
Then if we greedily select vectors from L_i and L_j then our return from L_i and L_j is

$$dim(L_i)\frac{OPT}{2^i} + dim(L_j)\frac{OPT}{2^j}.$$

Main Hurdle: Measure on the interaction between layers

Let us only plan to SELECT elements from L_i and L_j.
[Simple Case]: Let the $span(L_i)$ is disjoint from $span(L_j)$.
Then if we greedily select vectors from L_i and L_j then our return from L_i and L_j is

$$dim(L_i)\frac{OPT}{2^i} + dim(L_j)\frac{OPT}{2^j}.$$

[Hard Case]: If $span(L_i)$ is not disjoint from $span(L_j)$.
Thus selecting vectors from L_i can obstruct selecting vectors from L_j.

Main Hurdle: Measure on the interaction between layers

Let us only plan to SELECT elements from L_i and L_j.
[Simple Case]: Let the $span(L_i)$ is disjoint from $span(L_j)$.
Then if we greedily select vectors from L_i and L_j then our return from L_i and L_j is

$$dim(L_i)\frac{OPT}{2^i} + dim(L_j)\frac{OPT}{2^j}.$$

[Hard Case]: If $span(L_i)$ is not disjoint from $span(L_j)$.
Thus selecting vectors from L_i can obstruct selecting vectors from L_j.

So we have to understand how the layers are disrupting each other.