Learning Valuation Functions

Maria Florina Balcan

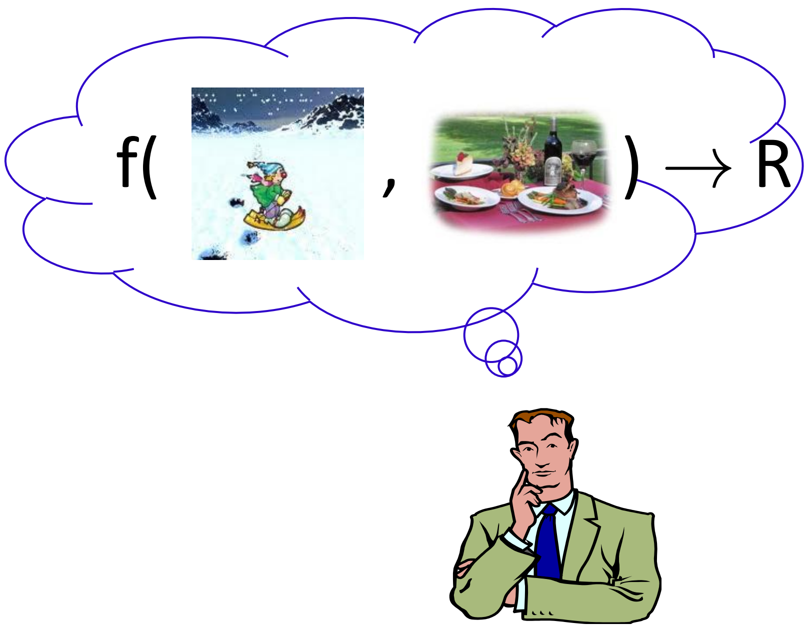
Georgia Institute of Technology

Joint with Florin Constantin, Satoru Iwata, Lei Wang

Valuation Functions

A first step in economic modeling:

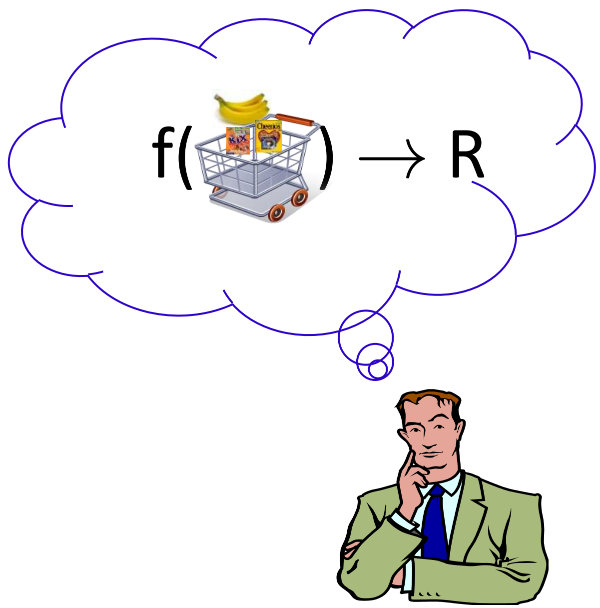
- individuals have valuation fns giving their value on different outcomes or events.



Valuation Functions

A first step in economic modeling:

- individuals have valuation fns giving their value on different outcomes or events.



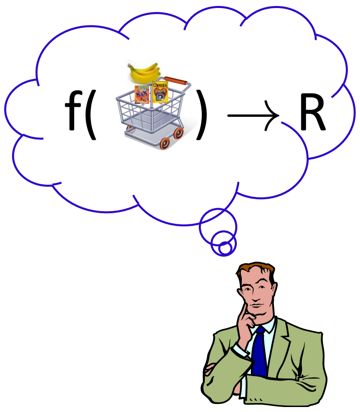
Valuation Functions

A first step in economic modeling:

- individuals have valuation fns giving their value on different outcomes or events.

Focus on combinatorial settings:

- n items, $V = \{1,2,\dots,n\}$
- $f : 2^V \to R$.



Learning Valuation Functions

This talk: learning valuation fns from past data.

- Supermarket pricing, advertising, coupons

- Web-app to find good deals

Valuation Functions

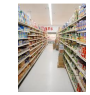
- Well-studied subclasses of subadditive valuations.

Additive $\subseteq$ OXS $\subseteq$ Submodular $\subseteq$ XOS $\subseteq$ Subadditive

[Sandholm'99] [Lehman-Lehman-Nisan'01]

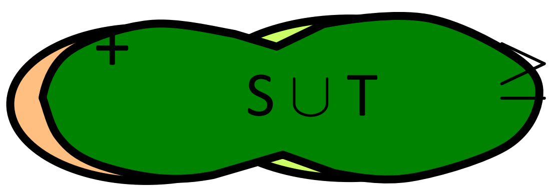
This talk

Subadditive valuations

- Ground set $V=\{1,2, \ldots, n\}$ (e.g., the items in a store) 
- For $S \subseteq V$, $f(S)$ = valuation of user for S. 

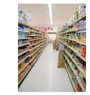
- Set-function $f : 2^V \to R$ subadditive if

  For all $S,T \subseteq V$: $f(S)+f(T) \geq f(S \cup T)$

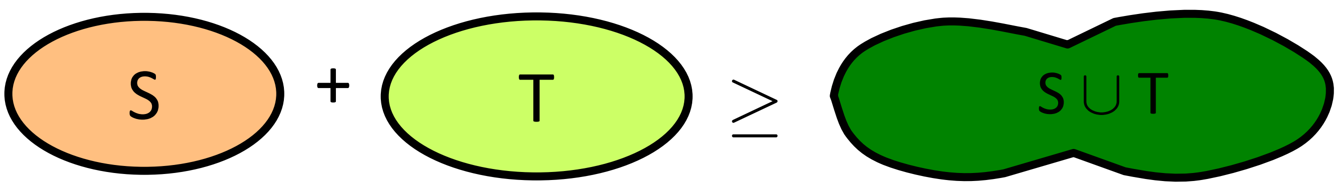


E.g.,

Subadditive valuations

- Ground set $V=\{1,2, \ldots, n\}$ (e.g., the items in a store) 
- For $S \subseteq V$, $f(S)$ = valuation of user for S. 
- Set-function $f : 2^V \rightarrow R$ subadditive if

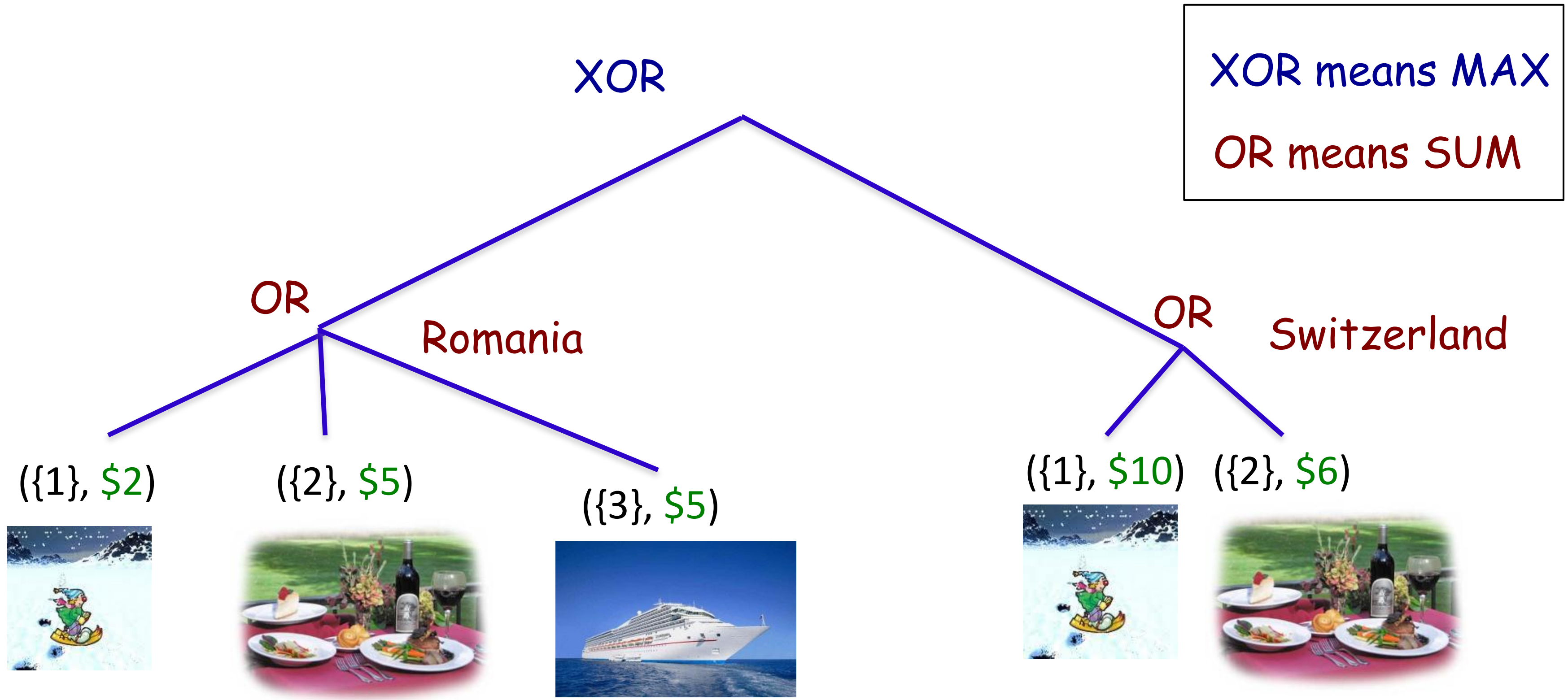
  For all $S,T \subseteq V$: $f(S)+f(T) \geq f(S \cup T)$



- Non-negative: $f(S) \geq 0, \forall S \subseteq V$
- Monotone: $f(S) \leq f(T), \forall S \subseteq T$

XOS valuations

XOS : Fns that can be represented as a MAX of SUMs.



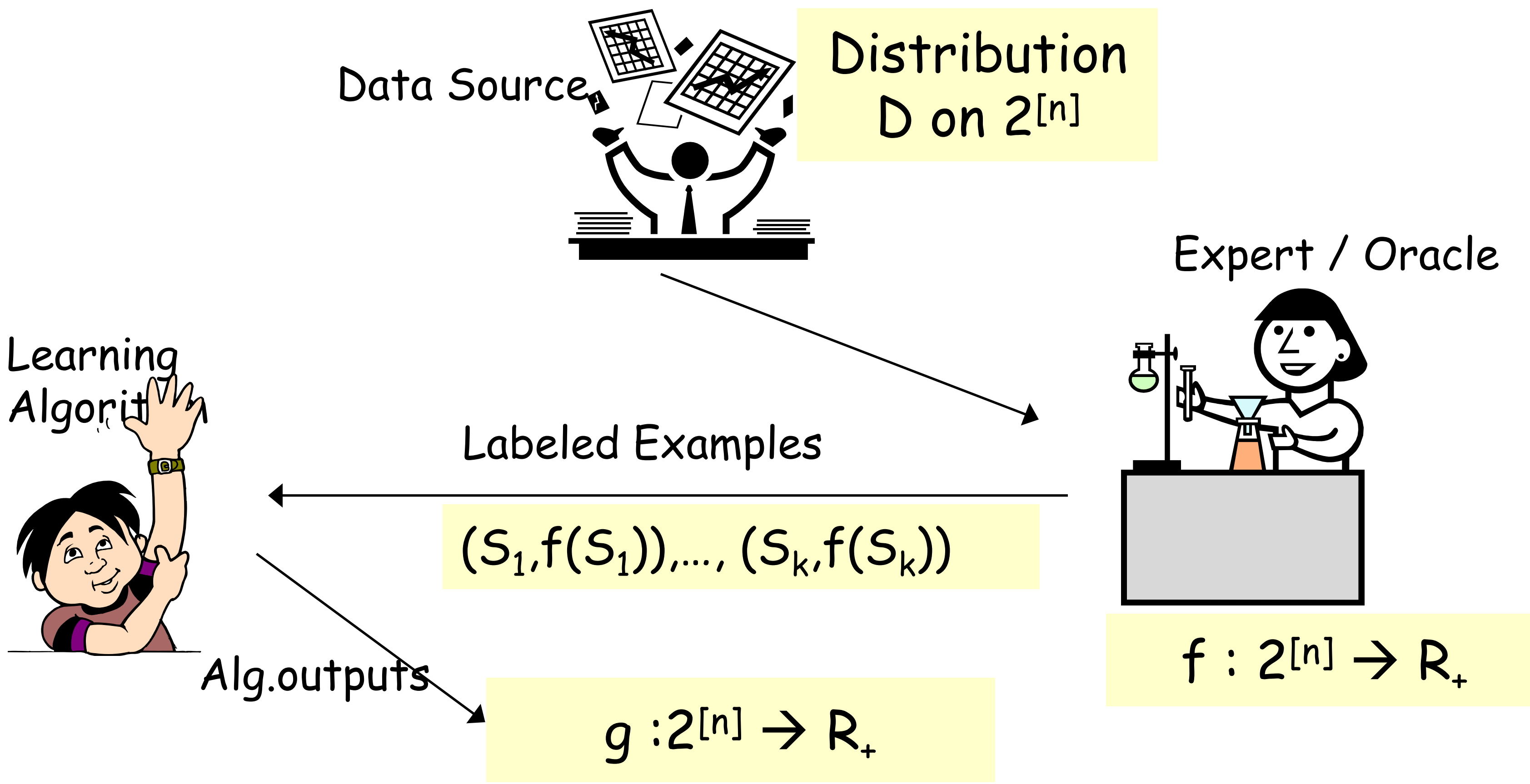
g({1,2}) = $16

g({2,3}) = $10

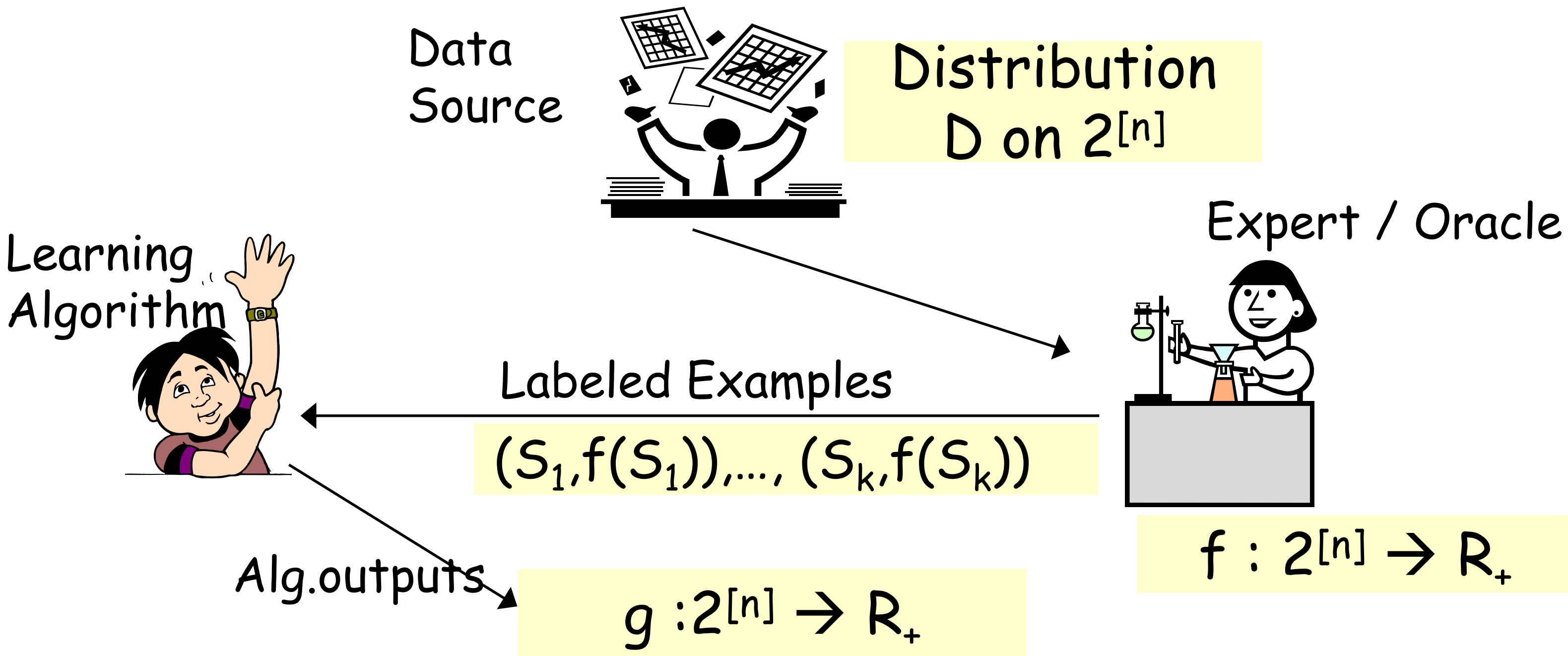
g({1,2,3}) = $16

Learning valuation functions from data.

Passive Supervised Learning



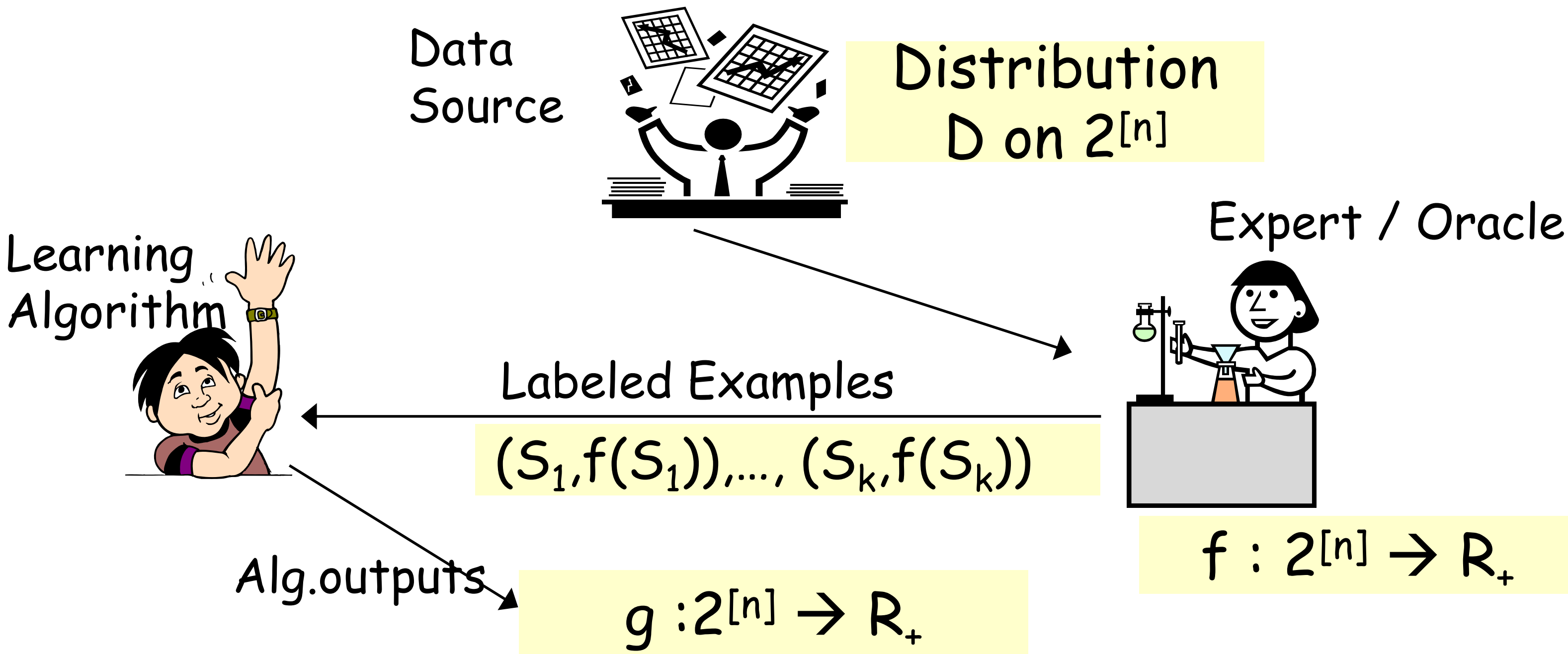
PMAC model for learning real valued functions



- Algo sees $(S_1,f(S_1)),\ldots,(S_k,f(S_k))$, S_i i.i.d. from D, produces g.
- With probability $\geq 1-\delta$ we have $\Pr_S[g(S) \leq f(S) \leq \alpha\, g(S)] \geq 1-\epsilon$

Probably Mostly Approximately Correct [Balcan-Harvey'11]

PMAC model for learning real valued functions



- Algo sees $(S_1,f(S_1)),\ldots,(S_k,f(S_k))$, S_i i.i.d. from D, produces g.
- With probability $\geq 1-\delta$ we have $\Pr_S[g(S) \leq f(S) \leq \alpha\, g(S)] \geq 1-\epsilon$
  - Compared to $E_S\{(f(S)-g(S))^2\}$, aligns better with the optimization literature.
  - Allows fine-grained control of errors: distinguishes between low error on most of the distrib & high error on a few points vs moderately high error everywhere.

Learning XOS, subadditive valuations

Theorem: (Our general upper bound)

Efficient alg. for PMAC-learning XOS fns with approx. factor $\alpha = O(n^{1/2})$ and subadditive fns with $\alpha = O(\log n \cdot n^{1/2})$.

Improves over [Badanidiyuru-Dobzinski-Fu- Kleinberg-Nisan-Roughgarden'12] and [Balcan-Harvey'11] .

Theorem: (Our general lower bound)

No algorithm can PMAC learn the class of XOS/subadditive fns with an approx. factor $\tilde{o}(n^{1/2})$.

Similar to [Badanidiyuru-Dobzinski-Fu- Kleinberg-Nisan-Roughgarden'12] and much simpler than [Balcan-Harvey'11] for submodular fns.

Theorem: XOS with Polynomial number of XOR trees

$O(n^{\epsilon})$ approximation in time $O(n^{1/\epsilon})$.

Lower Bound for XOS valuations

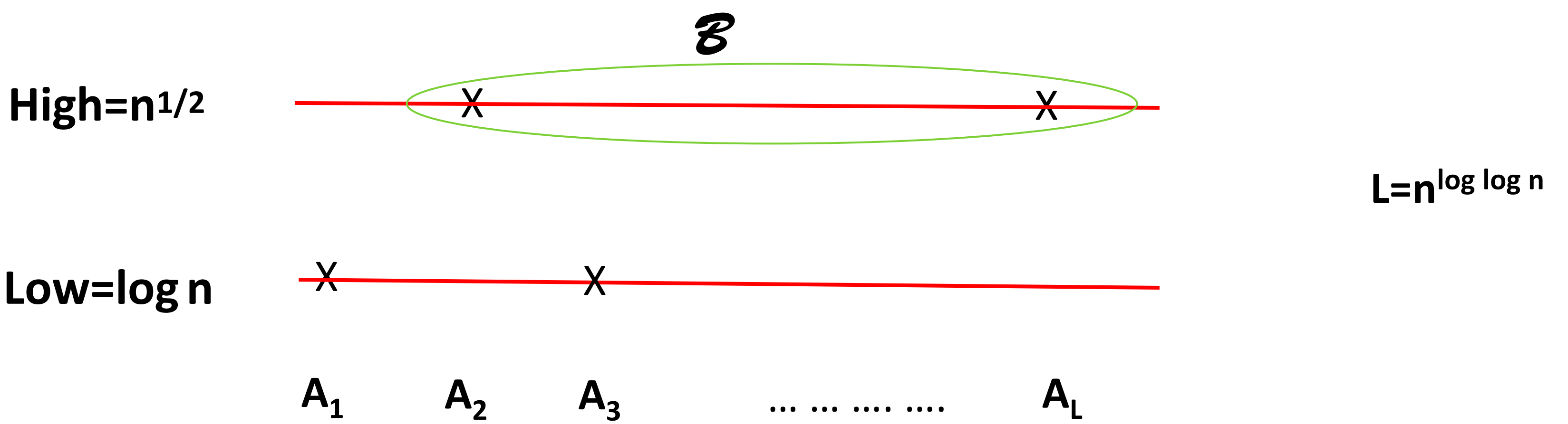
Theorem: No algorithm can PMAC learn the class of XOS valuations with an approx. factor $\tilde{o}(n^{1/2})$.

Main Idea:

There exist $A_1, \ldots, A_L$, $L=n^{\log \log n}$ s.t.:

(i) $|A\_i| \approx n^{1/2}$

(ii) $|A\_i \cap A\_j| \leq \log n$



Lower Bound for XOS valuations

Theorem: No algorithm can PMAC learn the class of XOS valuations with an approx. factor $\tilde{o}(n^{1/2})$.

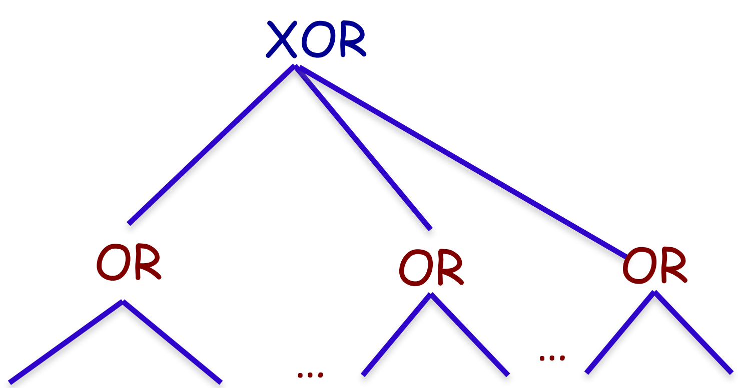
Main Idea:

There exist $A_1, \ldots, A_L$, $L=n^{\log \log n}$ s.t.:

(i) $|A\_i| \approx n^{1/2}$

(ii) $|A\_i \cap A\_j| \leq \log n$

For each A_i in $\mathcal{B}$, add an OR tree with leaves elements in A_i



A_i in $\mathcal{B}$

(i) $f(A_i)=|A_i|$, A_i in $\mathcal{B}$

(ii) $f(A_i) \leq \log n$, A_i not in $\mathcal{B}$

Lower Bound for XOS valuations

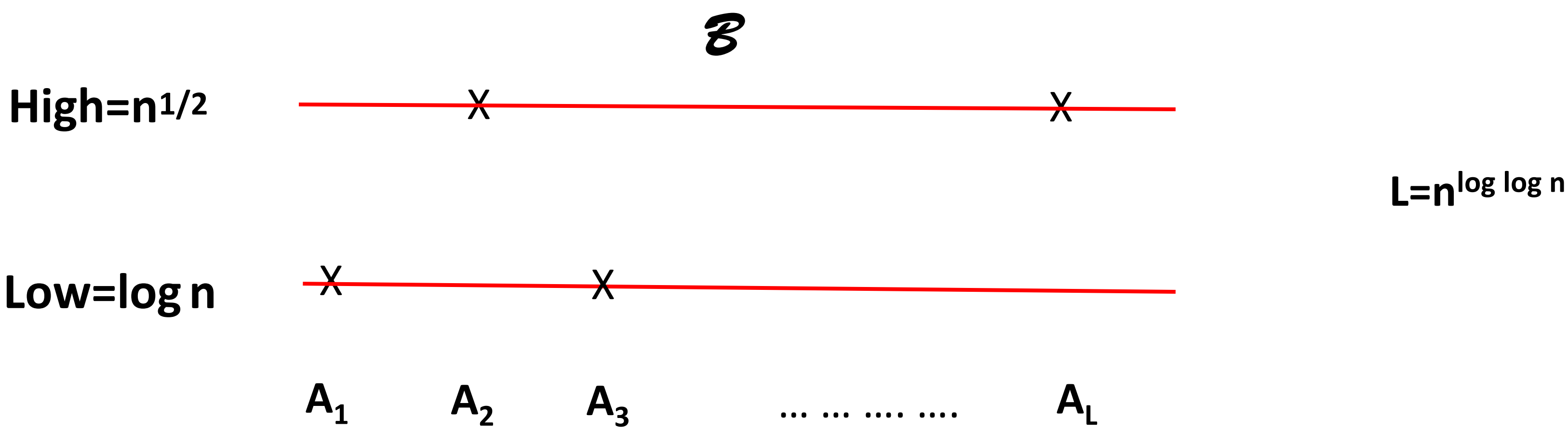
Theorem: No algorithm can PMAC learn the class of XOS valuations with an approx. factor $\tilde{o}(n^{1/2})$.

Main Idea:

There exist $A_1, \ldots, A_L$, $L=n^{\log \log n}$ s.t.:

(i) $|A\_i| \approx n^{1/2}$

(ii) $|A\_i \cap A\_j| \leq \log n$



General Upper Bound

Theorem: Efficient alg for PMAC-learning XOS fns with approx. factor $\alpha = O(n^{1/2})$ and subadditive fns with approx. factor $\alpha = O(\log n \cdot n^{1/2})$.

Main Ideas:

- **Claim**: f XOS approx. within $n^{1/2}$ by $\sqrt{\text{linear function}}$
- Set-function $f : 2^V \rightarrow R$ fractionally subadditive if

  For all $T \subseteq V$: $f(T) \leq \sum_S \{\lambda_S f(S)\}$ whenever

  $\lambda_S \geq 0$, $\sum_{S:\, s \in S} \lambda_S \geq 1$, for any s in T.
- $f : 2^V \rightarrow R$ fractionally subadditive iff XOS [Feige'06].

General Upper Bound

Theorem: Efficient alg for PMAC-learning XOS fns with approx. factor $\alpha=O(n^{1/2})$ and subadditive fns with approx. factor $\alpha=O(\log n \cdot n^{1/2})$.

Main Ideas:

- **Claim**: f fractionally subad. approx. within $n^{1/2}$ by $\sqrt{\text{linear function}}$
  - $f(T)= \max_{x \in P(f)} \{x(T)\}$, where $P(f)=\{x \geq 0: x(S) \leq f(S), \forall S \subseteq [n]\}$
  - John's ellipsoid theorem for symmetric convex bodies implies $\exists\, \mathcal{E}$ such that $\mathcal{E}$ contains P(f) and $(1/n^{1/2})\,\mathcal{E}$ is contained in P(f)
  - Define $g(T)= \max\_\{x \in 1/n^{1/2}\, \mathcal{E}\}\{x(T)\}$. So $g(S) \leq f(S) \leq n^{1/2} g(S)$
  - $\mathcal{E}$ is axis alligned, so g is $\sqrt{\text{linear function}}$

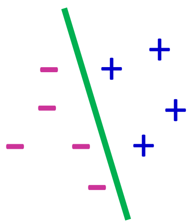
General Upper Bound

Theorem: Efficient alg for PMAC-learning XOS fns with approx. factor α=O($n^{1/2}$) and subadditive fns with approx. factor α=O(log n · $n^{1/2}$).

Main Ideas: $g^2(S) \le f(S) \le n\, g^2(S)$ where $g(S) = \left(w \cdot \chi(S)\right)^{\{\frac{1}{2}\}}$

features

- Labeled examples ((χ(S), f^2(S)), +) and ((χ(S), n · f^2(S)), -) linearly separable in R^{n+1}.



- Idea: reduction to learning a linear separator.

  Problem: data not i.i.d.

  Solution: create a related distib. P. Sample S from D; flip a coin. If heads add ((χ(S), f^2(S)), +) . Else add ((χ(S), n · f^2(S)), -).

  - Claim: A linear separator with low error on P induces a linear function with an approx. factor of $n^{1/2}$ on the original data.

General Upper Bound

Theorem: Efficient alg for PMAC-learning XOS fns with approx. factor $\alpha = O(n^{1/2})$ and subadditive fns with approx. factor $\alpha = O(\log n \cdot n^{1/2})$.

Main Ideas:

Input: $(S_1, f(S_1)) \ldots, (S_m, f(S_m))$

- For each S_i, flip a coin.
  - If heads add $((\chi(S), f^2(S_i)), +)$.
  - Else add $((\chi(S), n\, f^2(S_i)), -)$.
- Learn a linear separator $u=(w,-z)$ in R^{n+1}.

Output: $g(S)=1/(n+1)^{1/2}\; w \cdot \chi(S)$

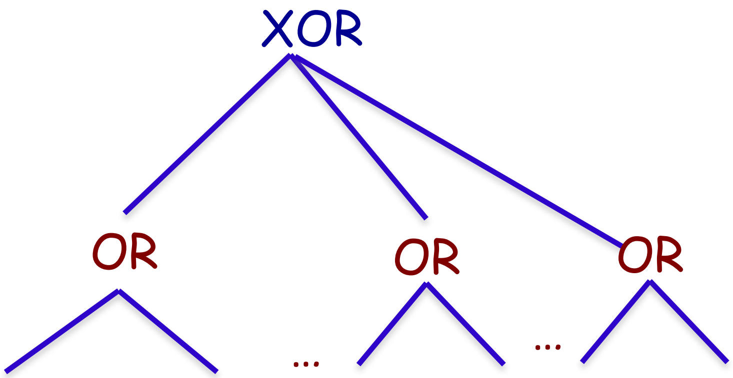
A subadditive function is within $\log n$ factor of a XOS function.

XOS: Target dependent Upper Bound

Theorem: (Polynomial number of XOR trees)
$O(n^{\epsilon})$ approximation in time $O(n^{1/\epsilon})$.

Highlights importance of complexity of the target function.

Main Proof Idea:



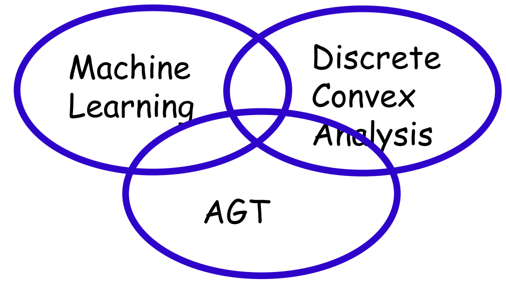
$f(S) = \max_{j=1..R} \{k_j(S)\}$

where $k_j(S) = w_j \cdot \chi(S)$

- $g(S)=(1/R)\, \mathrm{sum}_j \{(k_j(S))^L\}$ satisfies $g(S) \leq f(S)^L \leq R\, g(S)$
- Reduction to learning a linear separator over L-tuples.

Conclusions

Learnability of important classes of valuation functions (OXS, XOS, subadditive).



Open Questions

- Better bounds for XOS functions with polynomial number of XOR trees

- Analyze learnability of other interesting classes of valuations functions