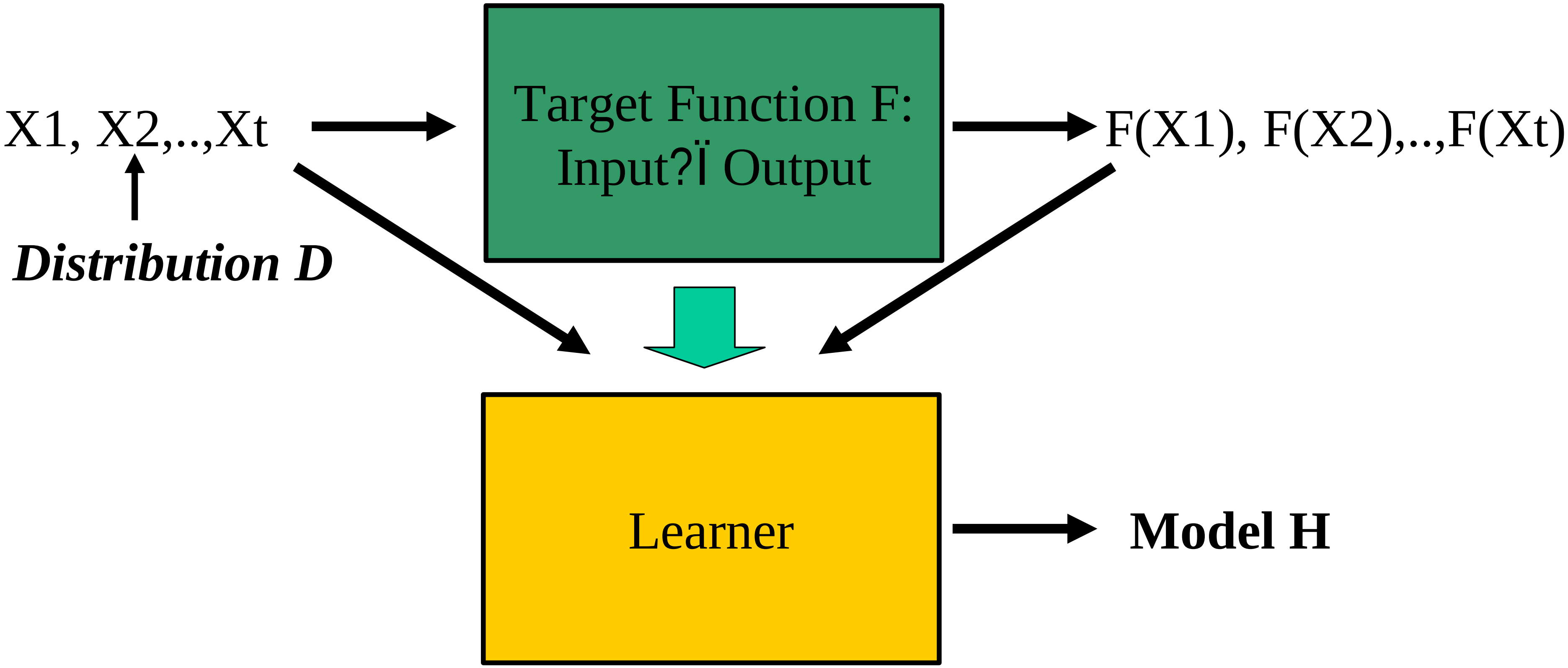
Machine Learning Paradigms for Utility Based Data Mining

Naoki Abe
Data Analytics Research
Mathematical Sciences Department
IBM T. J. Watson Research Center

Contents

(Standard) Batch Learning Model

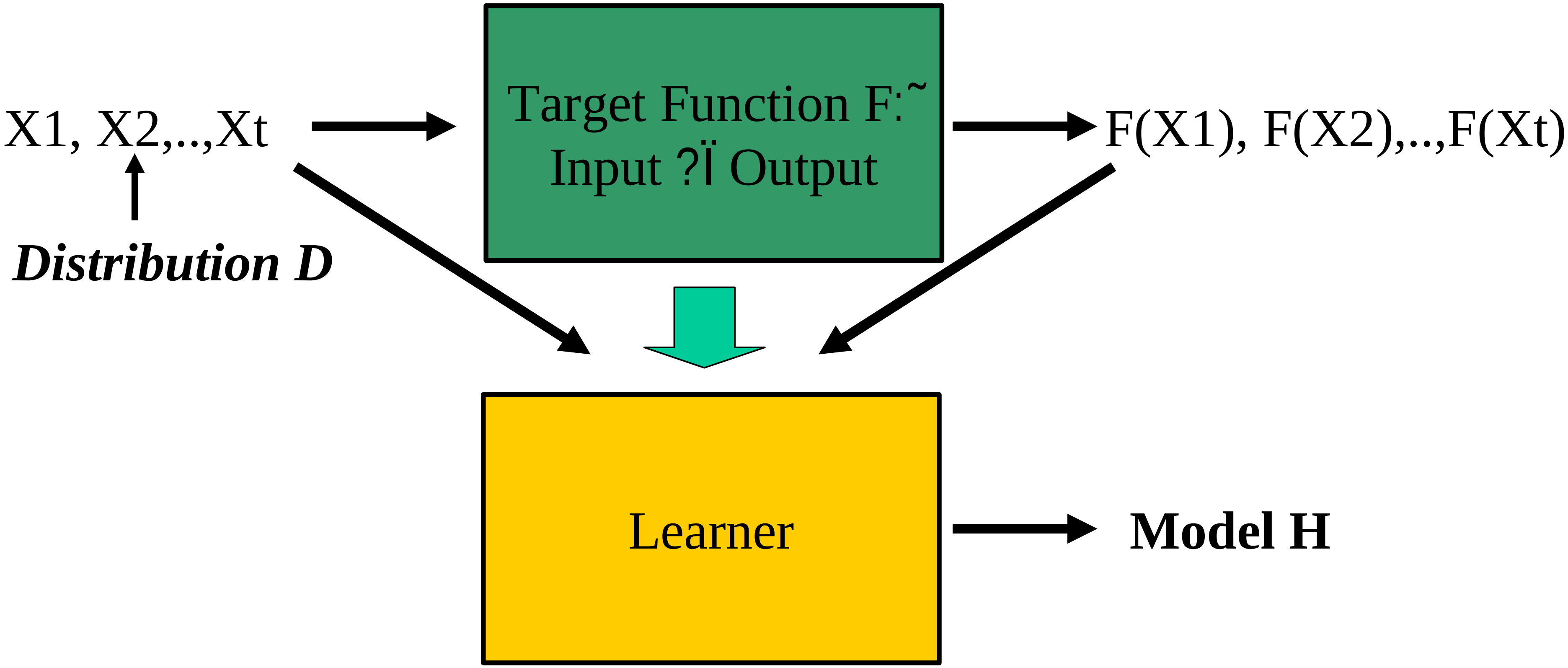


Learner's Goal: Minimize Error(H, F) for given t

e.g.) PAC-Learning Model[Valiant'84]

$$\text{PAC-Learning} \;=\; \Pr\{E_{x \approx D}[H(x) \neq F(x)] > \varepsilon\} < \delta$$

(Utility-based) Batch Learning Model



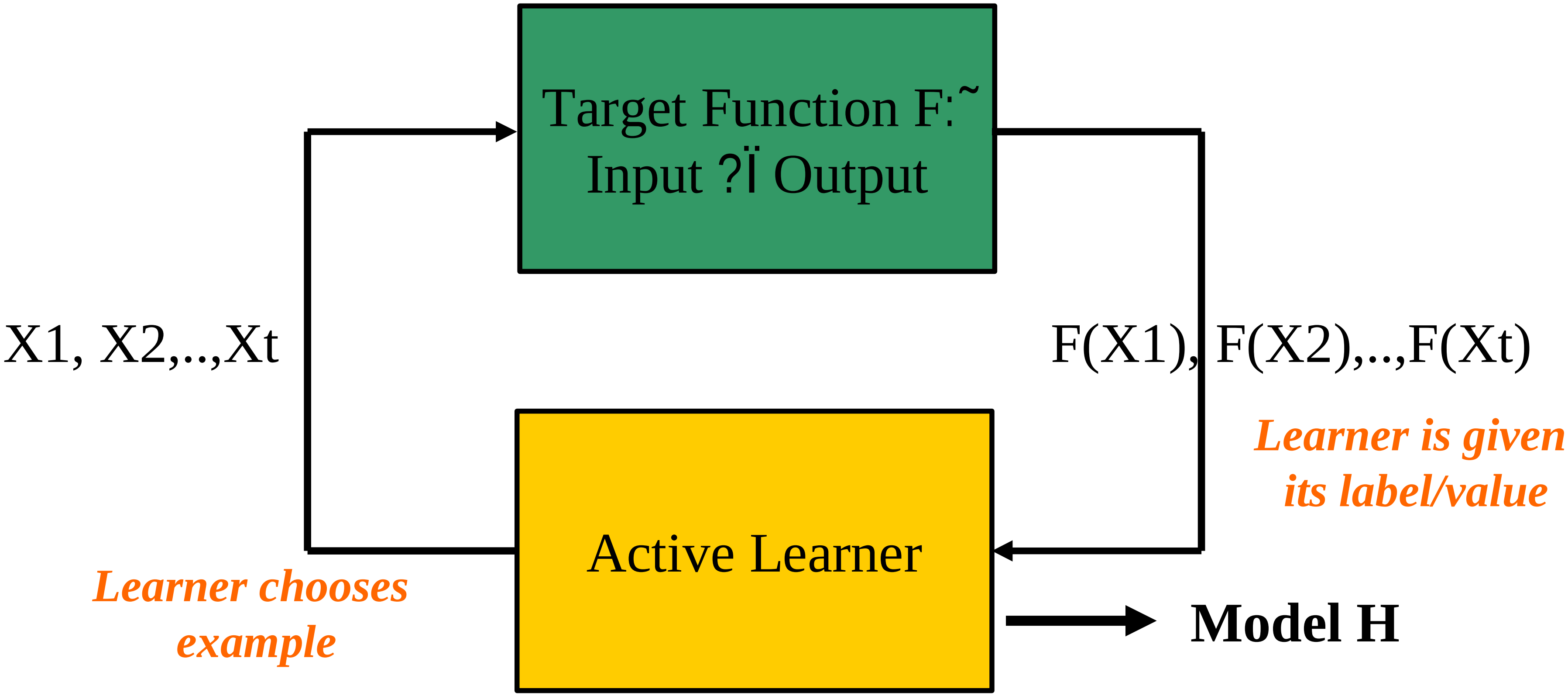
Learner's Goal: Minimize Loss(H, F) for given t

e.g.) Decision Theoretic Generalization of PAC Learning*… [Haussler'92]

$$\text{Generalized-PAC-Learning} = \Pr\{E_{x \approx D}[l(H(x), F(x))] > \varepsilon\} < \delta$$

***Subsumes cost-matrix formulation of cost-sensitive learning, but not example dependent cost formulation …**

Active Learning Model



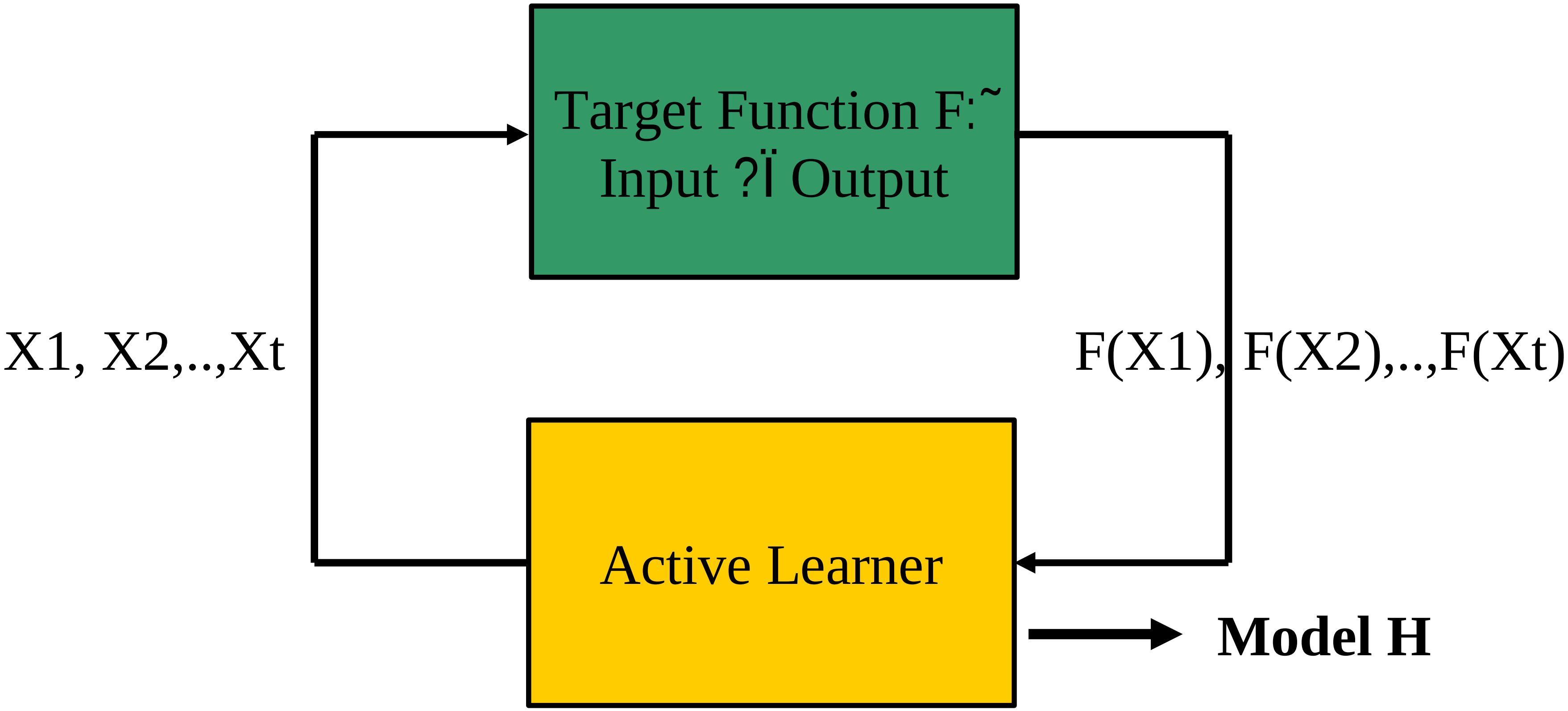
Active Learner's Goal: Minimize err(H, F) for given t

(Minimize t for given err(H,F))

e.g.) MAT-learning model [Angluin'88]:

Minimize t to achieve err(H,F)=0, assuming that F belongs to given class

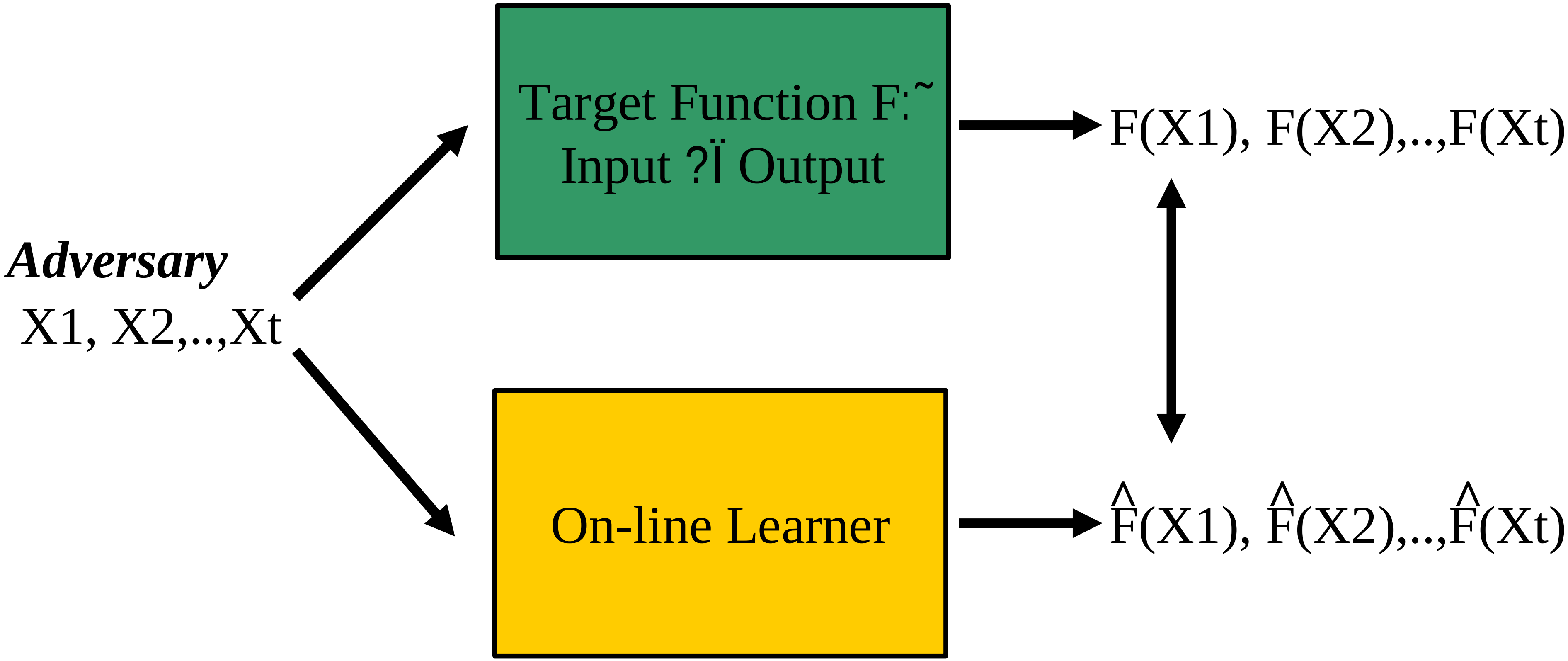
(Utility-based) Active Learning Model



<u>Active Learner’s Goal: Minimize cost(H, F) +S cost(Xi) for given t</u>

c.f.) Active feature value acquisition [Melville et al ’04, ’05]*
***Not subsumed since acquisition of individual feature values is considered**

On-line Learning Model

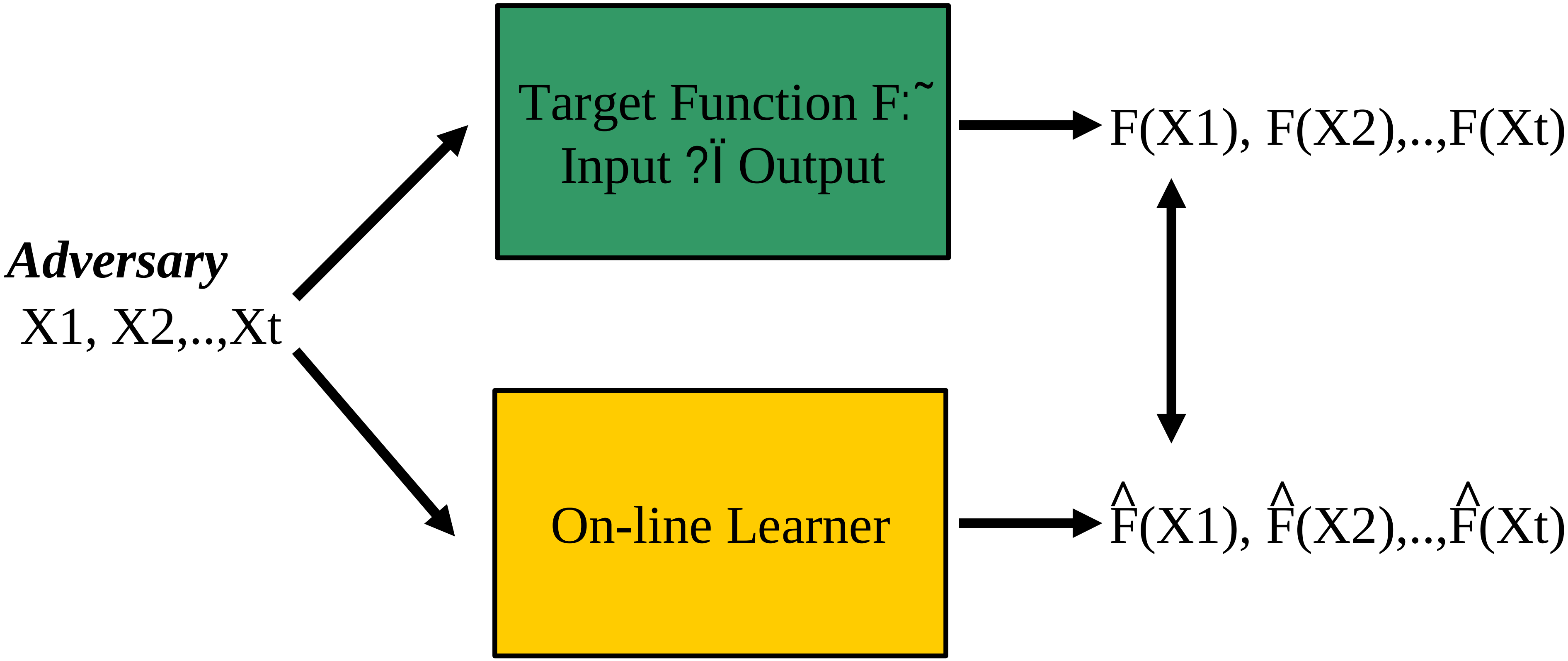


<u>On-line Learner's Goal: Minimize Cum. Error Σ err(F̂(Xi),F(Xi))</u>

e.g.) Mistake Bound Model [Littlestone '88], Expert Model [Cesa-Bianchi et al 97]

Minimize the worst-case $\sum_{i=1}^{t} | \hat{F}(x_i) - F(x_i) |$

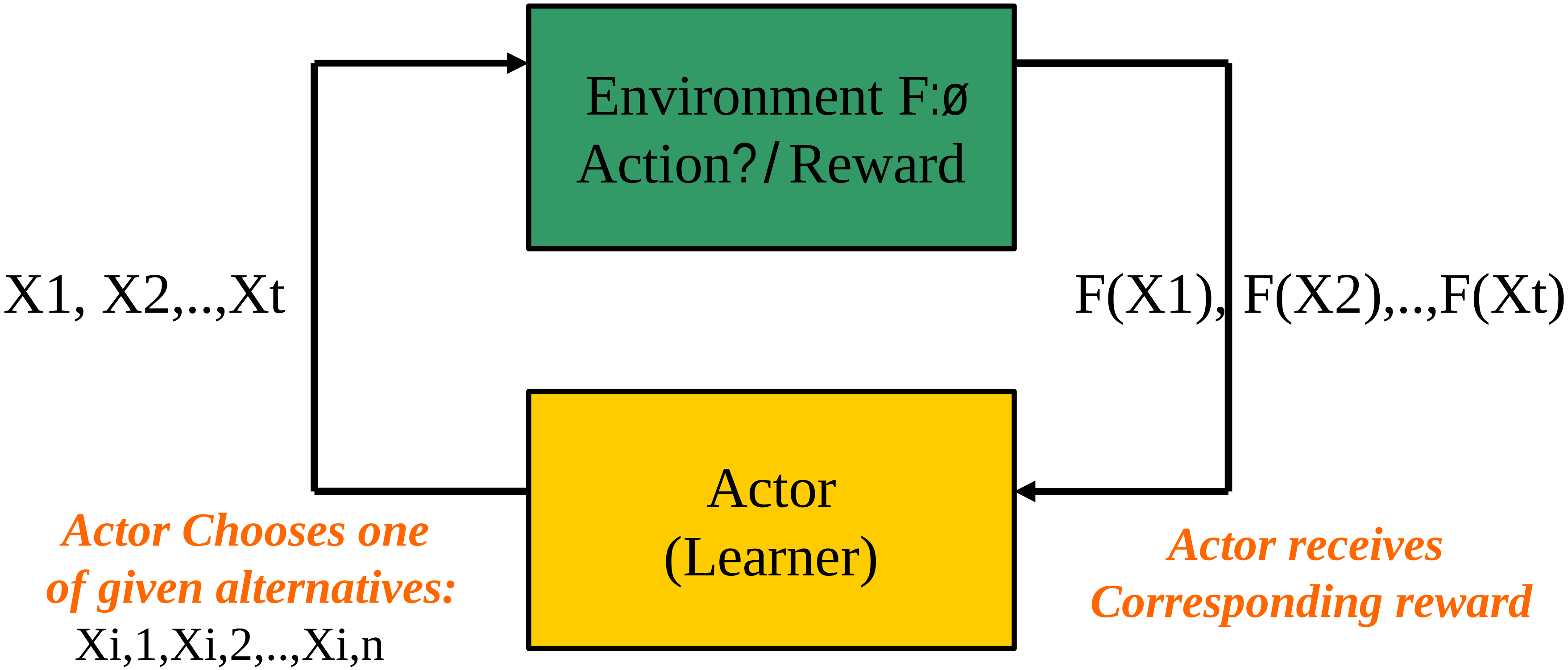
(Utility-based) On-line Learning Model



<u>On-line Learner’s Goal: MinimizeS· Loss(F(Xi),$\hat{F}$(Xi))</u>

e.g.) On-line loss bound model [Yamanishi ’91]

On-line Active Learning
(Associative Reinforcement Learning*)

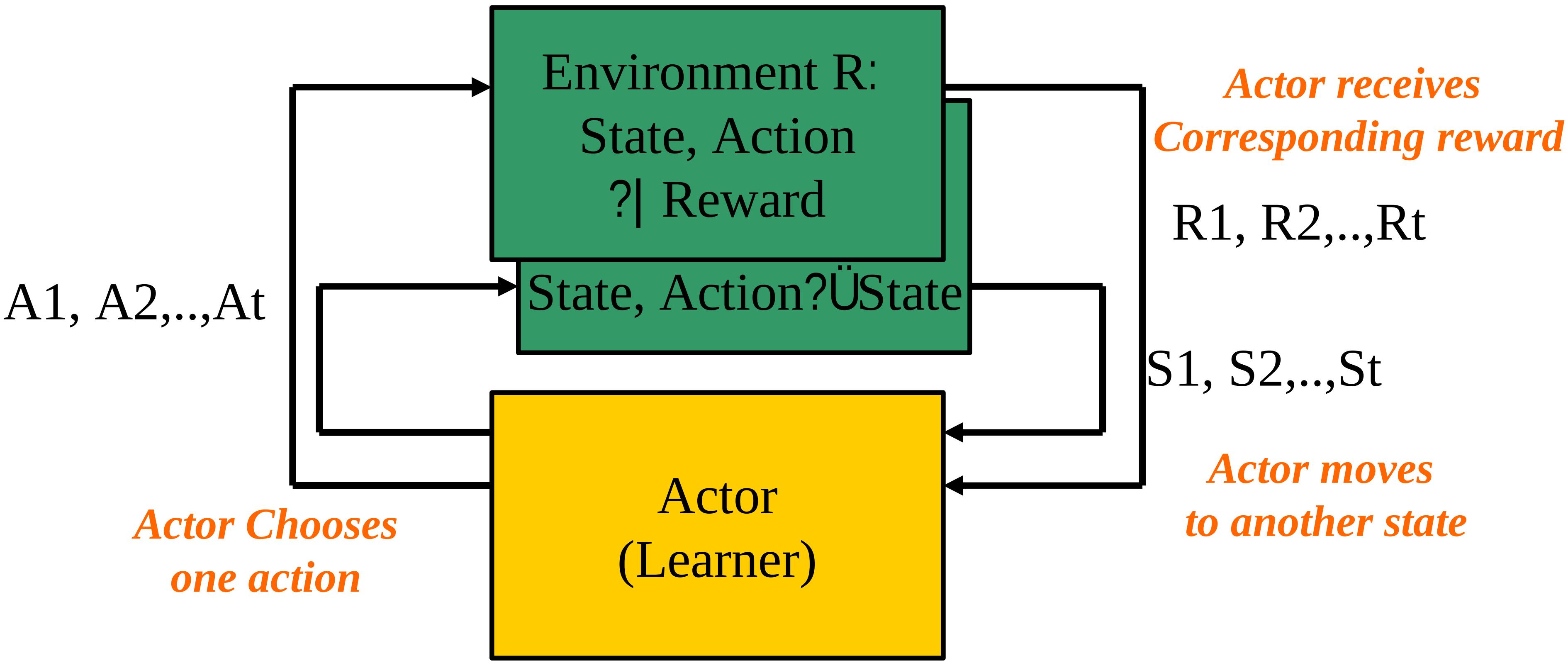


Actor's Goal: Maximize Cumulative Rewards SŠF(Xi)
(F(xi) can incorporate cost(xi): this is already a utility-based model !)

e.g.) Bandit Problem [BF'85], Associative Reinforcement Learning [Kaelbling'94]
Apple Tasting [Helmbold et al'92], Lob-Pass [Abe&Takeuchi'93]
Linear Function Evaluation [Long 97, Abe&Long 99, ABL'03]
***Also known as "Reinforcement Learning with Immediate Rewards"**

Reinforcement Learning

Markov Decision Processes



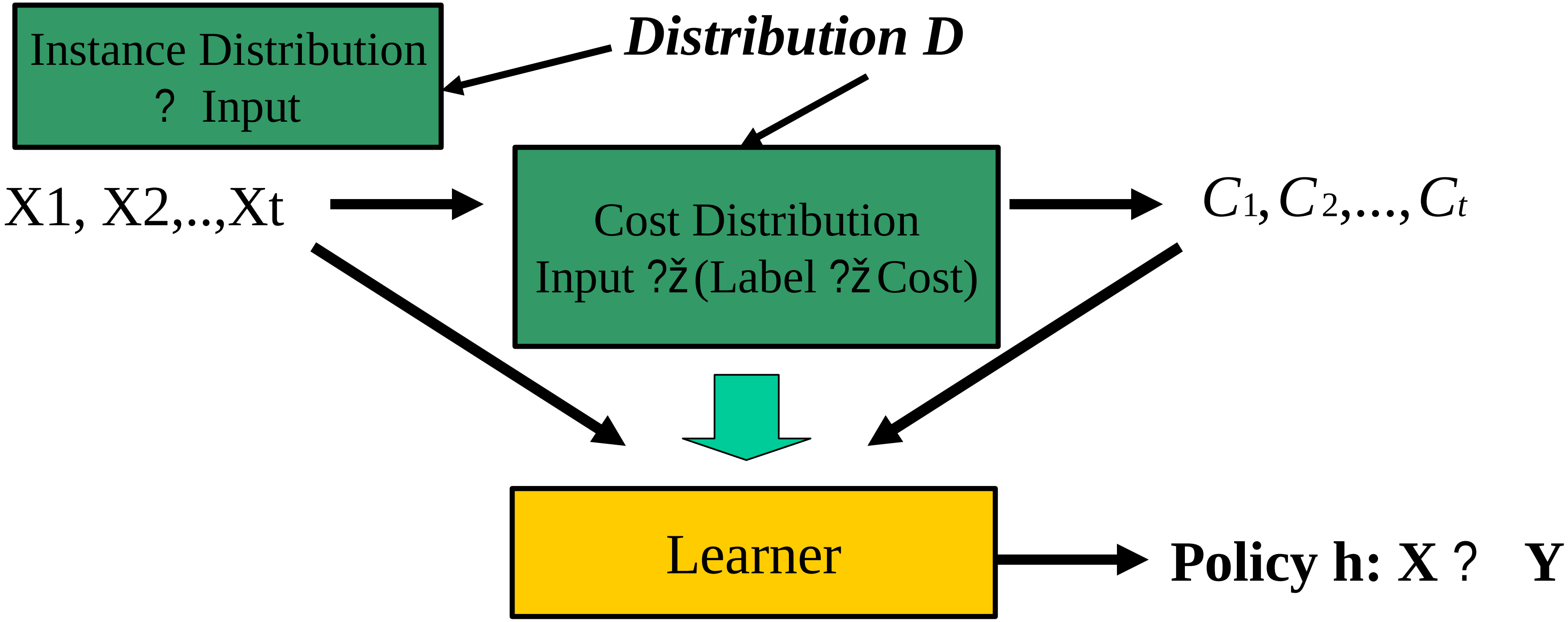
Actor's Goal: Maximize Cumulative RewardsSá**Ri (or** Sá? ¡**Ri)**

e.g.) Reinforcement Learning for Active Model Selection [KG'05]
Pruning improves cost-sensitive learning [B-Z,D'02]

Contents

- Learning Models and UBDM
  - Learning Models
  - Utility-based Versions
- *Case Studies*
  - Example-dependent Cost-sensitive Learning
  - One-Benefit Cost-Sensitive Learning
  - On-line Active Learning
  - Batch vs. On-line Reinforcement Learning
- Applications
- Discussions

Example Dependent Cost-Sensitive Learning [ZE'01,ZLA'03]

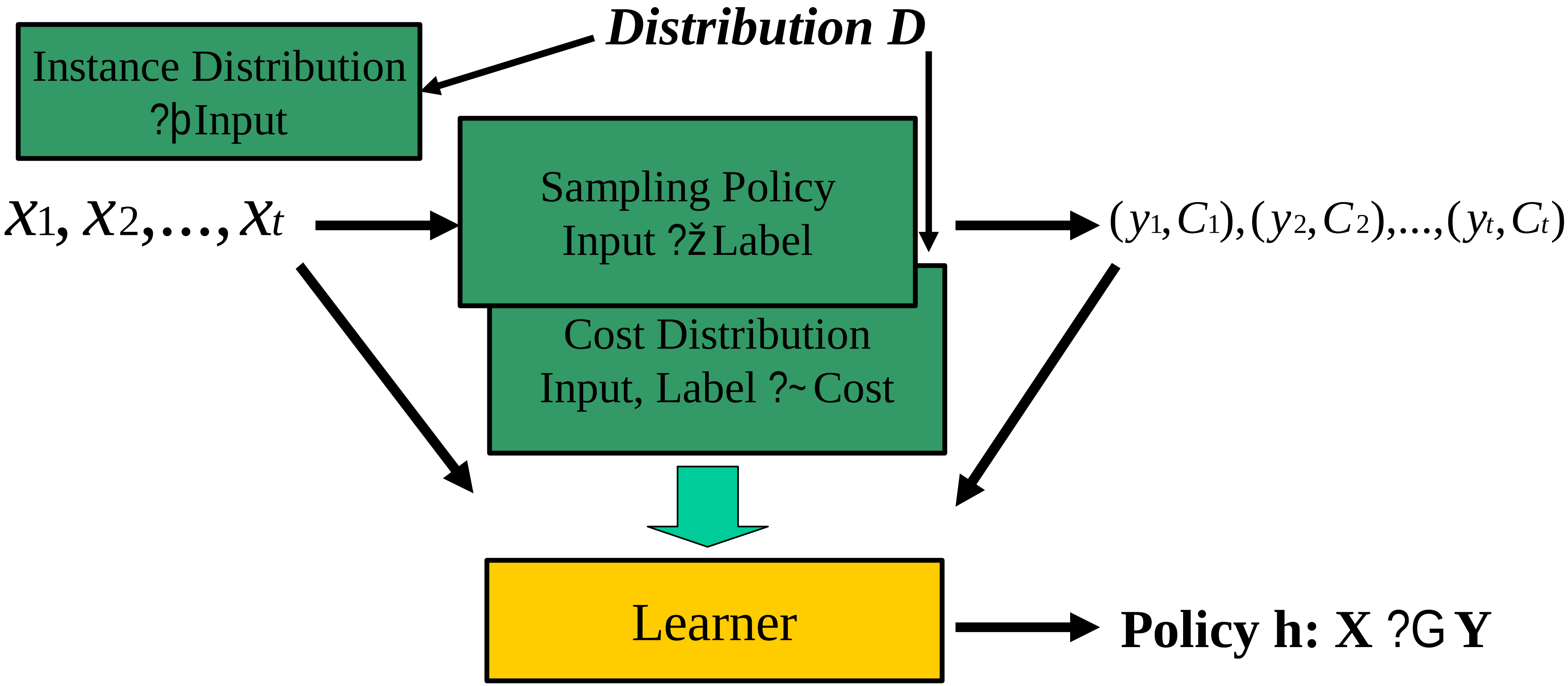


PAC Cost-sensitive Learning… [ZLA'03]

$$\Pr\{E_{x,y,c \approx D}[c \cdot I(h(x) \neq y)] - \min_{f \in H}\{Cost(f)\} > \varepsilon\} < \delta$$

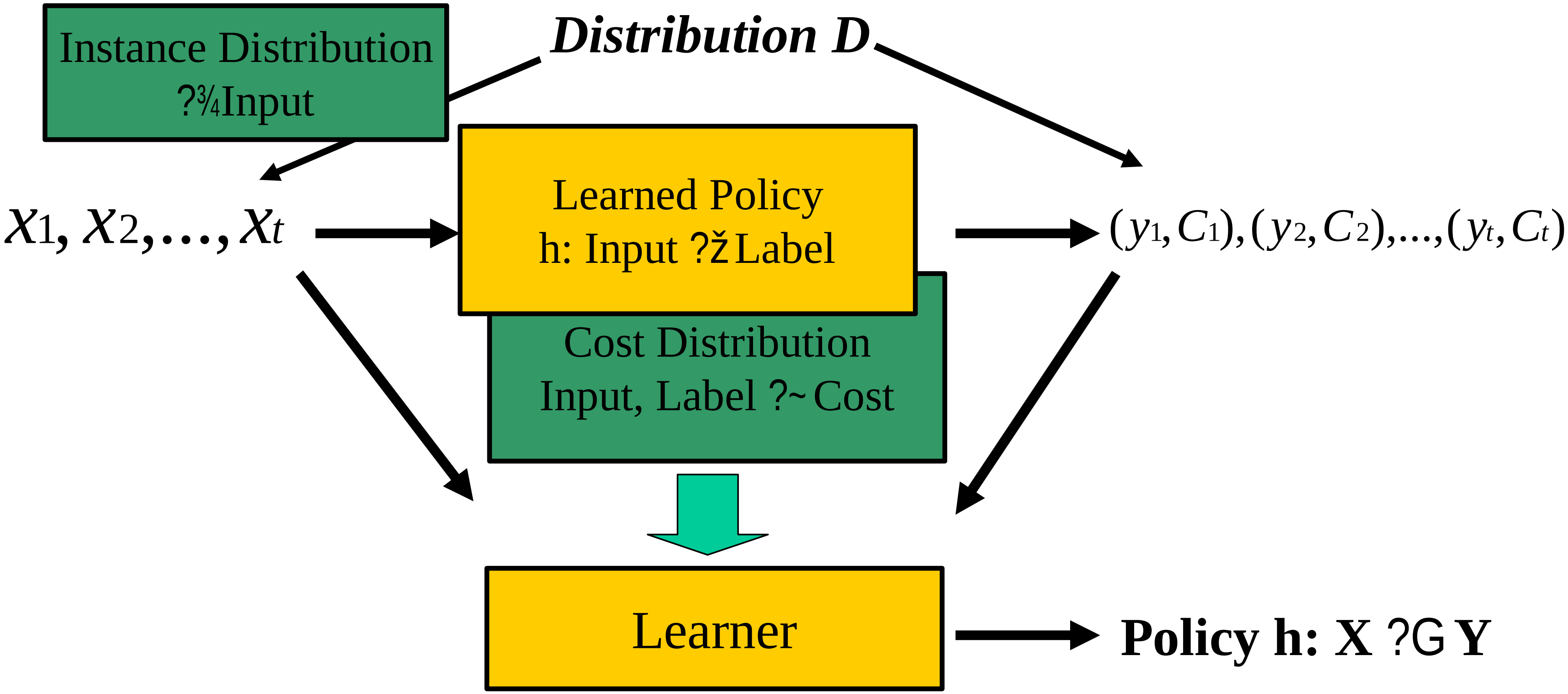
- **A key property of this model is that the learner must learn the utility-function from data**
- **Distributional modeling has let to simple but effective method with theoretical guarantee**
- **The full cost knowledge model works for 2-class or cost-matrix formulations, but…**

One Benefit (Cost-Sensitive) Learning [Zadrozny'03,'05]



***Key property is that the learner gets to observe the utility corresponding only to the action (option/decision) it took…**

One Benefit Cost-Sensitive Learning [Zadrozny’03,’05]



<u>Learner’s Goal: Minimize Cost(h) w.r.t. D?h</u>

***Key property is that the learner gets to observe the utility corresponding only to the action (option/decision) it took…**

***Another key property is that sampling policy and learned policy differ**

An Example On-line Active Learning Model: Linear Probabilistic Concept Evaluation

[Abe and Long ’99]

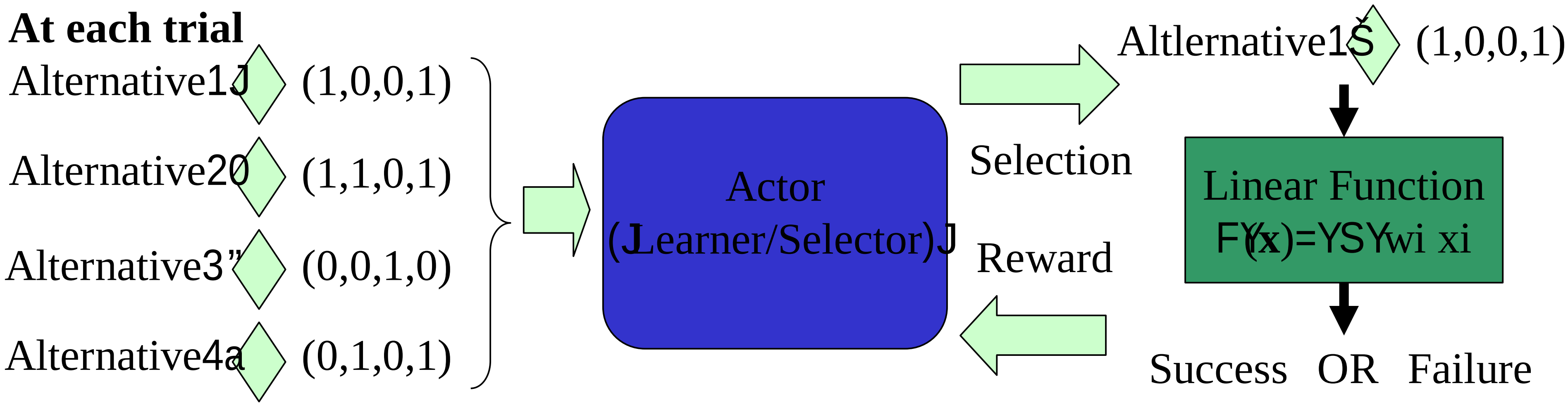
– Select one from a number of alternatives

– Success probability =Linear Function(Attributes)

– Performance Evaluation for Learner/Selector

E(Regret)=ãEã(ãOptimal Rewards)ã- ãEã(ãCumulative Rewards)ã

If you knew function F



Actor’s Goal: Maximize Total Rewards!õ

An Example On-line Learning/Selection Method [AL'99]

- **<u>Strategy</u>** Ap
  - Learning:j Widrow-Hoff Update with Step Size aj =j $1/t^{1/2}$
  - Selection:
    - Explore: Select J (?ðI*) with prob. ?ð1/|$\hat{F}$(I*)-$\hat{F}$(J)|
    - Exploit: Otherwise select I* with max estimated success probability

Performance Analysis

Bounds on Worst Case Expected Regrets

<u>Theorem</u> [AL'99]

- Upper Bound on Expected Regret
  - Learning Strategy A
    - Expected Regret $= / O(t^{3/4} n^{1/2})$

- Lower Bound on Expected Regret
  - Expected Regret of any Learner $=+O+(t^{3/4} n^{1/4})$

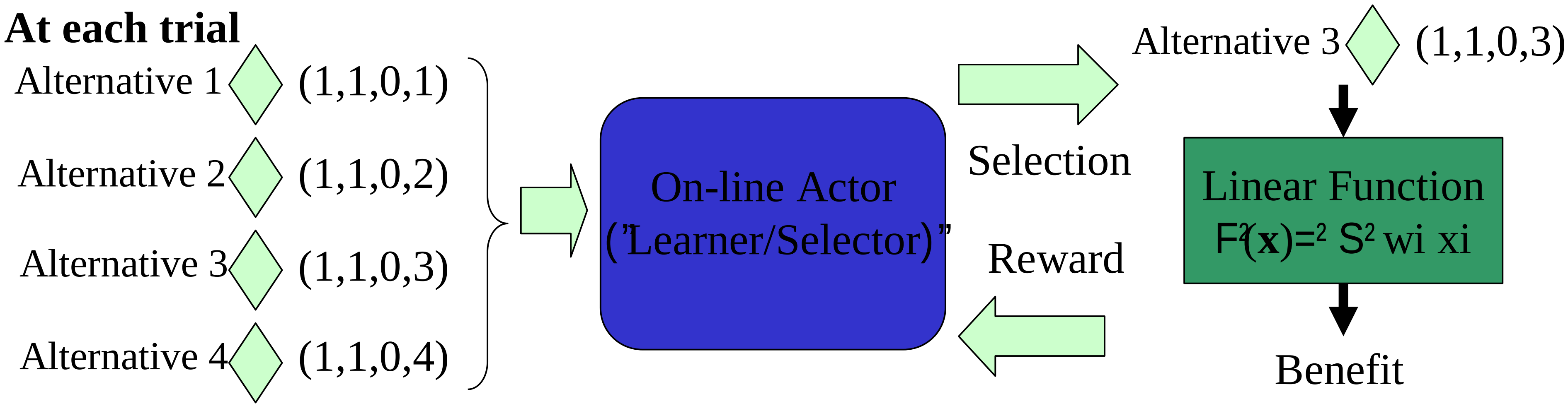
Expected regret of Strategy A is asymptotically optimal as function of t!

One-Benefit Cost-Sensitive Learning [Zadrozny ’05] as On-line Active Learning

“One-Benefit Cost-Sensitive Learning” [Z’05] could be thought of as a “batch” version of on-line active learning

- Each alternative consists of the common x-vector and a variable y-label
- Alternative Vectors:

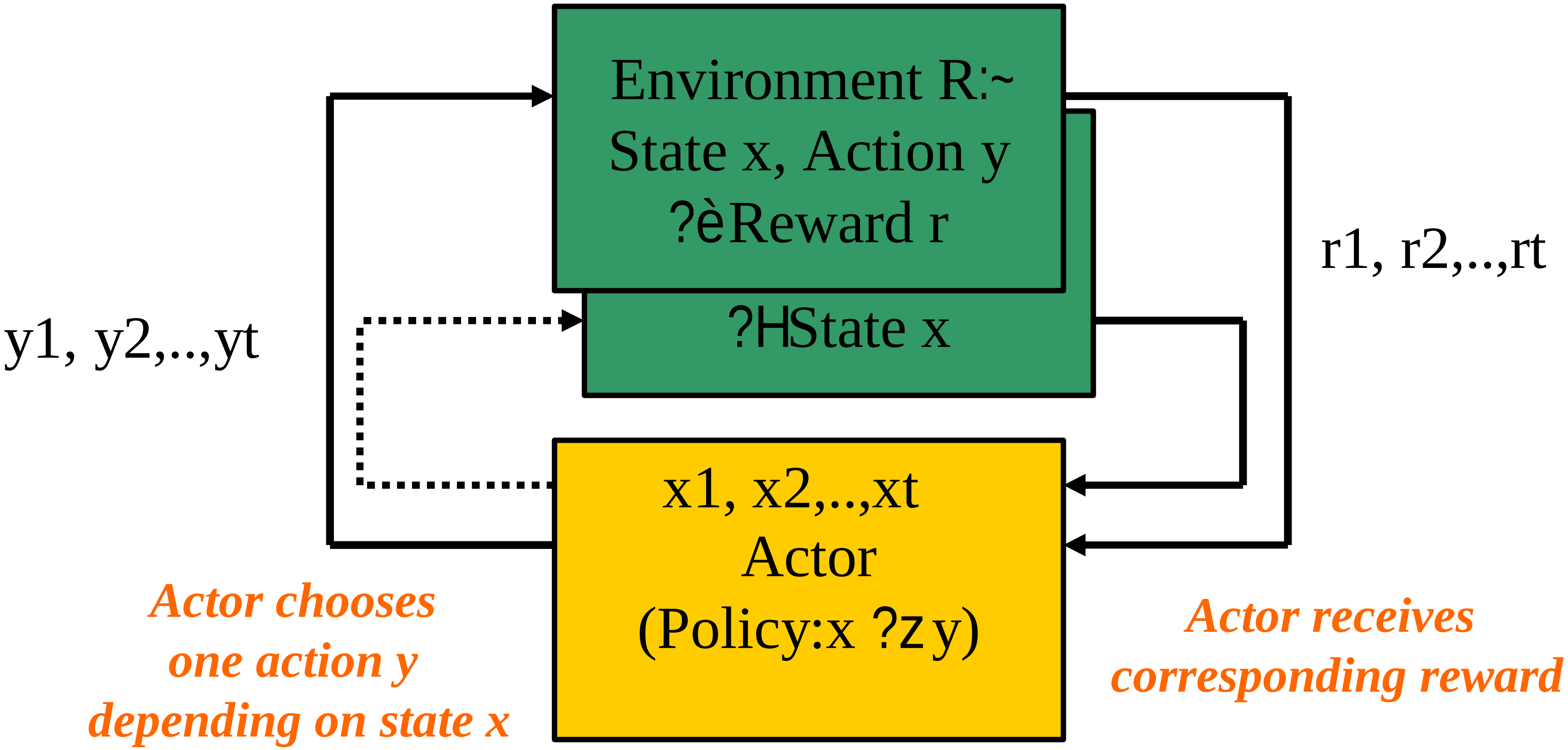
  $(X^{.3}Y1), (X^{.3}Y2), (X^{.3}Y3),\ldots, (X^{.3}Yk)$



Actor’s Goal: Maximize Total Benefits!õ

One-Benefit (Cost-Sensitive) Learning [Z’05] as Batch Random-Transition Reinforcement Learning*

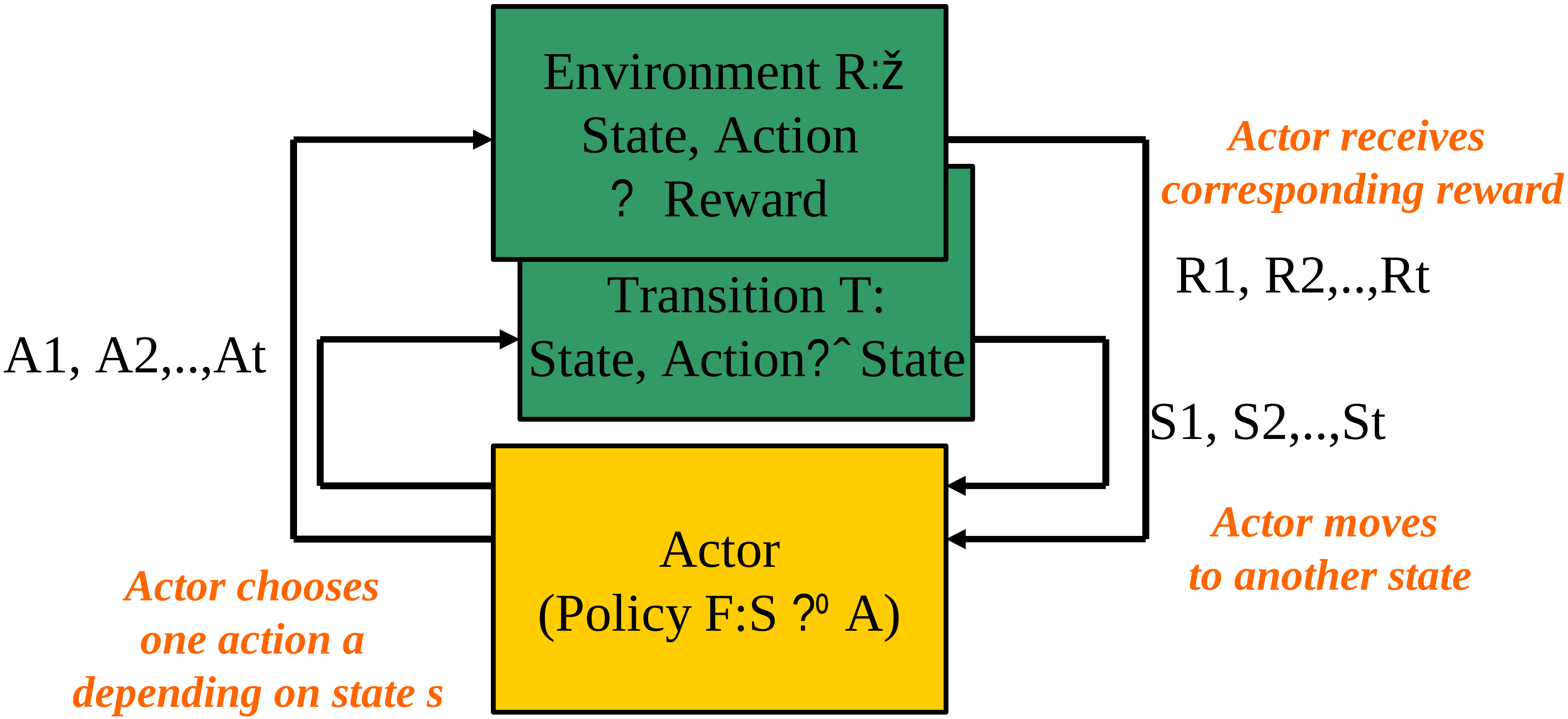
*Called “Policy Mining” in Zadrozny’s thesis [’03]



On-line Learner’s Goal: Maximize Cumulative Rewards S”ri

Batch Learner’s Goal: Find policy F s.t. expected reward E_D[R(x,F(x))] is maximized, given data generated w.r.t. sampling policy P(y|x)

On-line v.s. Batch Reinforcement Learning



On-line learner's Goal: Maximize Cumulative Rewards ScRi

Batch Learner's Goal: Find policy F s.t. expected reward E_T[R(s,F(s))] is maximized, given data generated w.r.t. sampling policy P(a|s)

Contents

- Learning Models and Utility
  - Learning Models
  - Utility-based Versions
- Case Studies
  - Example-dependent Cost-sensitive Learning
  - One-Benefit Cost-Sensitive Learning
  - On-line Active Learning
  - Batch vs. On-line Reinforcement Learning
- *Applications*
- Discussions

Internet Banner Ad Targeting [LNKAK’98,AN’98]

- Learn Fit Between Ads and Keywords/Pages
- Display a Toyota Ad on keyword ‘drive’
- Display a Disney Ad on animation page
- **The Goal is to maximize the total click-through’s**

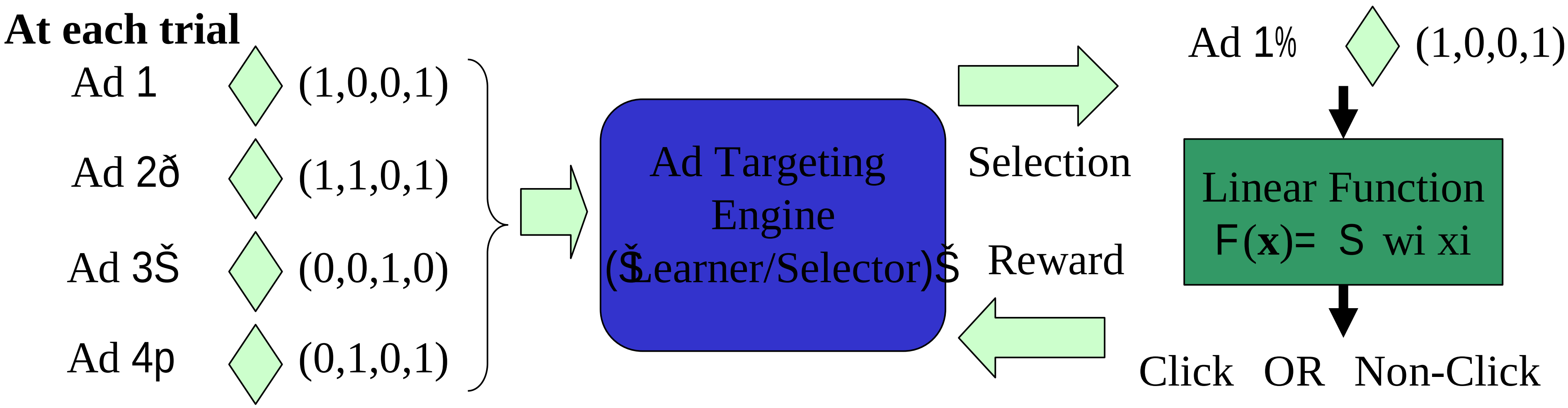
Car Ad

Search Keyword ‘drive’

A Solution with On-line Active Learning

- Represent Click-through Rates as Linear Function of Ad/User Attribute Vectors
- Ad/User Attribute Vector =

  (A1·ÈU1, A2·ÈU1, A1·ÈU2, A2·ÈU2)



Ad Targeter’s Goal: Maximize Total Click-throughs!

- ***Key issue is the Exploration Exploitation Trade-Off !***

A Simpler Solution Using
Gittins Index for Bandit Problem

Gittins Index

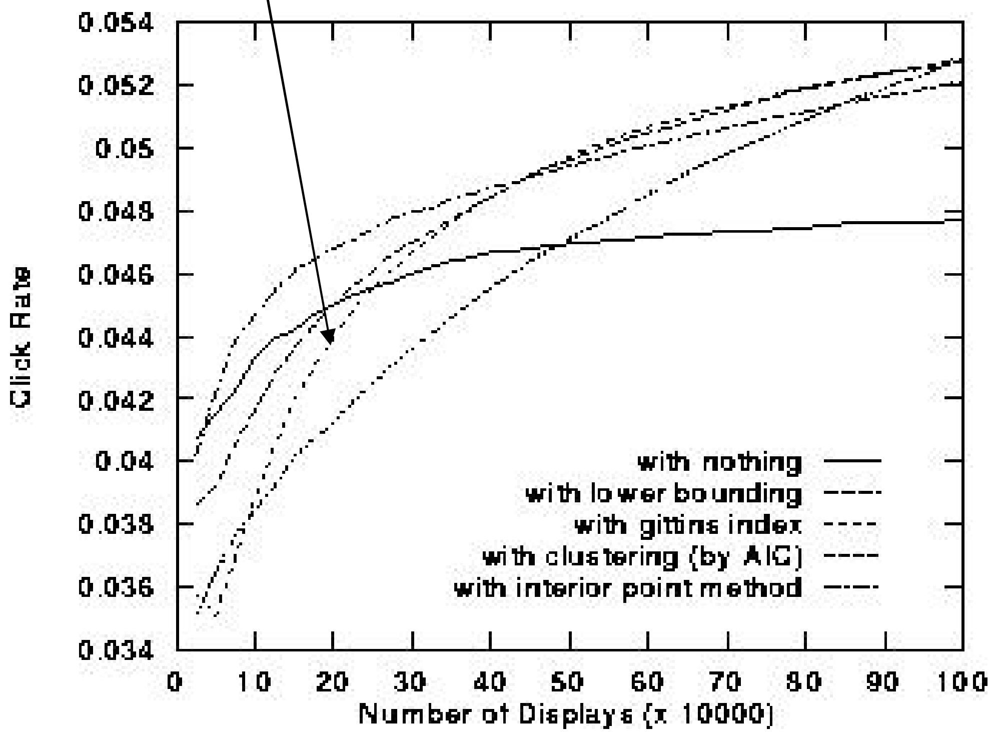
#non-clicks

#clicks

ßá a	0	1	2	3	4	5	6
0á	0. 84	0.91	0.94	0.95	0.96	0.96	0.97
1	0.53	0.71	**0.78**	0.82	0.85	0.87	0.88
2Š	0.37	**0.56**	0.66	0.71	**0.75**	0.78	0.80
3í	0.28	0.46	0.56	0.62	0.67	0.71	**0.74**
4ï	0.22	0.39	**0.48**	0.55	0.60	0.64	0.68
5¨	0.17	0.33	0.43	0.49	0.55	0.59	0.62
6¸	0.15	0.29	0.38	**0.45**	0.50	**0.54**	0.58

Empirical Results [AN'98]

(LP with Gittins modification)



G(a2,ß2) = p such that

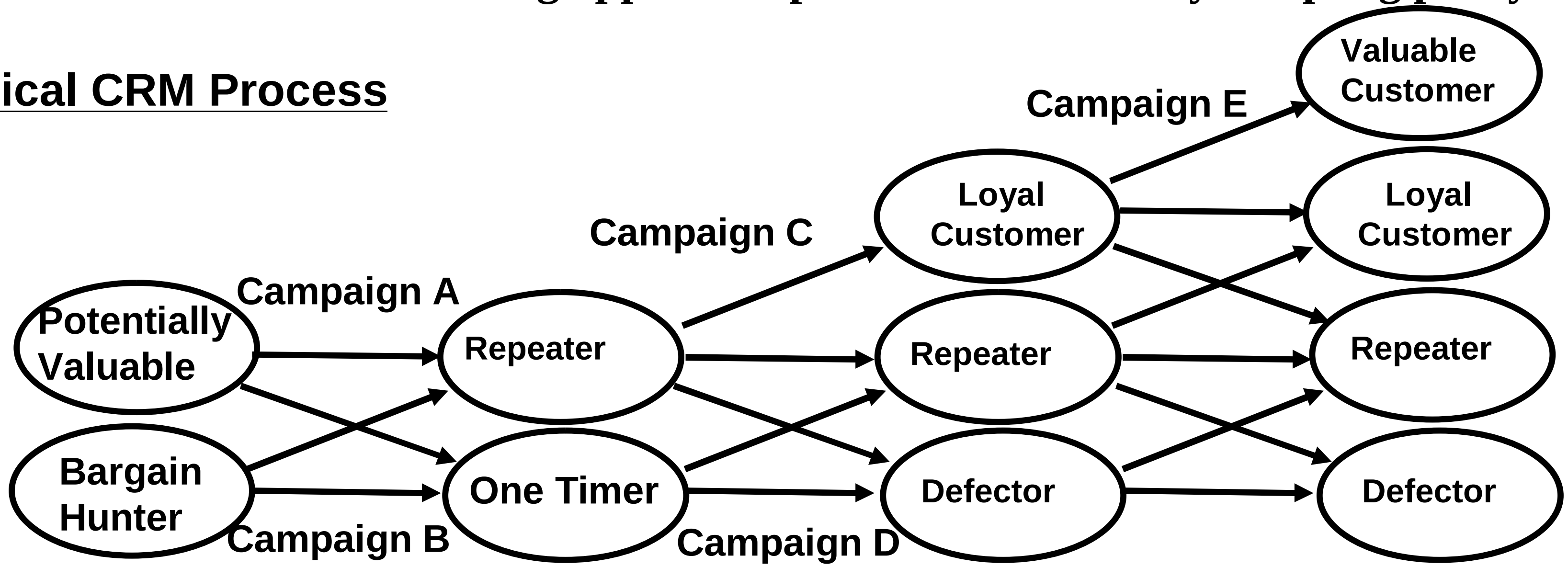
discounted cumulative reward of p = discounted cumulative reward of (a2,ß2)

i.e. pÂ / 1ì?ì ?ì = a™ / aà+àßà (áìà+à?àRà(àà+à1à?àßà?àpà)à)à+à ß£ / aÂ+ÂßÂ ?áRà(àà?àßà+à1à?àpà)à

Maximizing Customer Lifetime Value by Batch Reinforcement Learning [PAZ…'02,AVAS'04]

- **Model CRM process using "Markov Decision Process"(MDP)**
  - Customer is in some "state" (his/her attributes) at any point in time
  - Enterprise's action will move customer into another state
  - Enterprise's goal is to take sequence of actions to guide customer's path to maximize customer's life time value
- **Produce optimized targeting rules as a policy**
  - If customer is in state "s", then take marketing action "a"
  - Customer state "s" represented by customer attribute vector computed from data
- **Batch Reinforcement Learning applied on past data collected by sampling policy**

<u>**Typical CRM Process**</u>



Bias Correction in Evaluation

- ***Key Challenge is the Bias Correction due to Batch Learning***:
  - Need to evaluate new policy using data collected by existing (sampling) policy
- Solution: Use bias-corrected estimation of "policy advantage" using data collected by sampling policy
- Definition of policy advantage:
  - (Discrete Time) Advantage

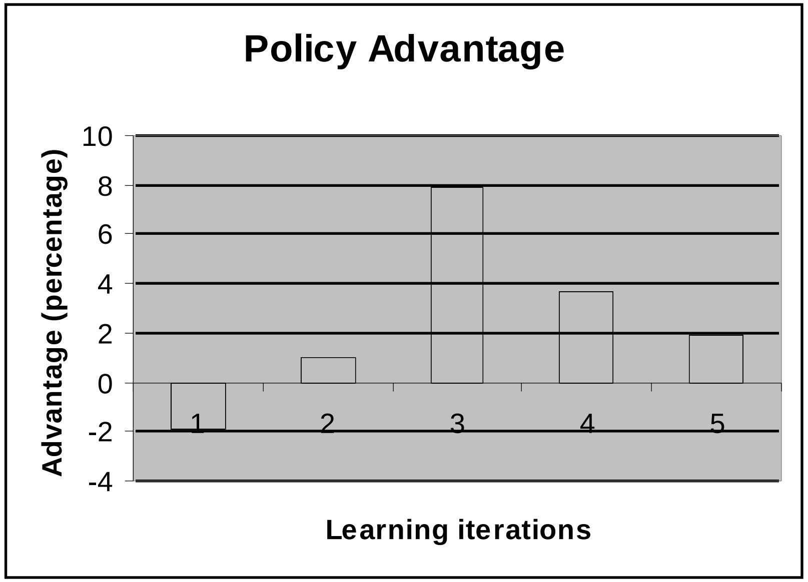
$$\mathbf{A}_{\text{pá}}\mathbf{(s,a) := Q}_{\text{pá}}\mathbf{\ (s,a) - max}_{\mathbf{a}'}\mathbf{\ Q}_{\text{pá}}\mathbf{\ (s,a')}$$

  - Policy Advantage

$$\mathbf{A}_{\mathbf{s}\sim\text{pž}}\mathbf{(}\text{p?}\mathbf{'):= E}_{\text{pá}}\mathbf{\ [E}_{\mathbf{a}\sim\text{pž}'}\mathbf{\ [A}_{\text{pá}}\mathbf{(s,a)]]}$$

- Estimating policy advantage with bias corrected sampling

$$\mathbf{A}_{\mathbf{s}\sim\text{pp}}\mathbf{(}\text{p}^2\mathbf{\ '):= E}_{\text{pá}}\mathbf{\ [(}\text{p}^2\mathbf{\ '(a|s)/\ }\text{p}^2\mathbf{\ (a|s))\ [A}_{\text{pá}}\mathbf{(s,a)]]}$$

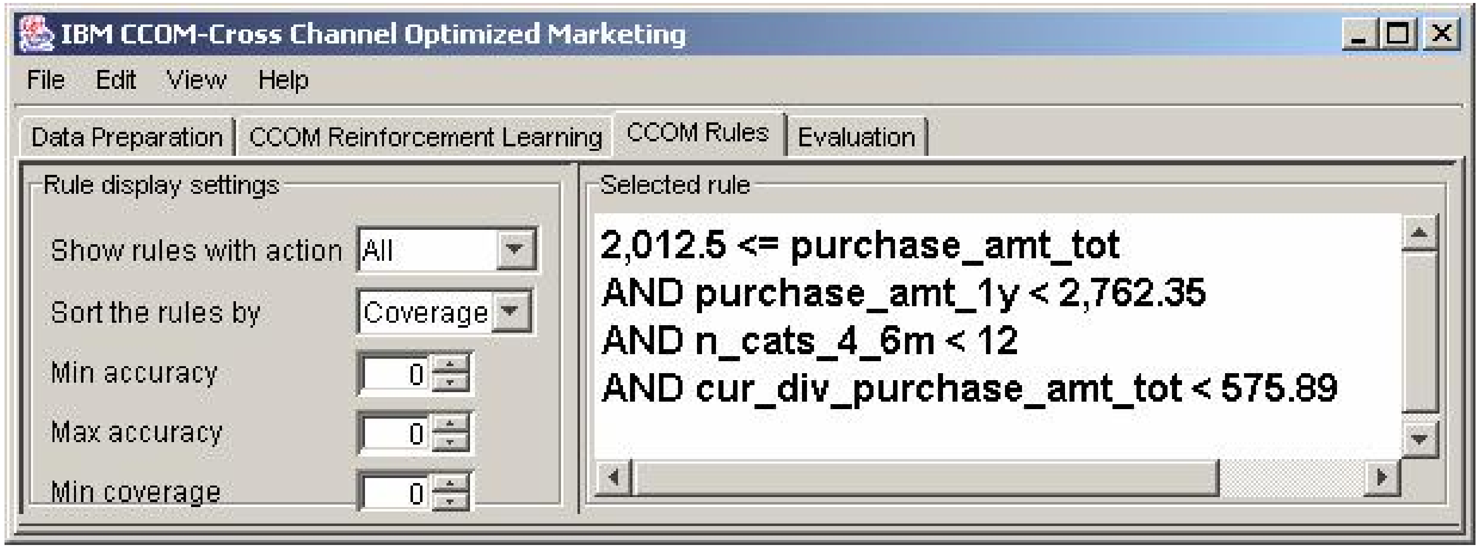


Policy advantage over actual policy of Saks Fifth Avenue data

An Example Rule (that addresses Exploration-Exploitation Trade-off)

This rule suggests that the enterprise wait until it has seen enough of the customer's behavior to judge that he or she is not interested in a given product group … i.e. it invests in the customer until it knows it is not worth it

If



then don't mail

- **Interpretation: If a customer has spent significantly in the past and yet has not spent much in the current division (product group) then don't mail**

Contents

- Learning Models and UBDM
  - Learning Models
  - Utility-based Versions
- Concrete Examples
  - Example-dependent Cost-sensitive Learning
  - On-line Active Learning
  - One-Benefit Cost-Sensitive Learning
  - Batch vs. On-line Reinforcement Learning
- Applications
- *Discussions*

Discussions

- Machine Learning Paradigms vs. Utility-based Data Mining
  - Practical considerations lead to refinement and extension of existing learning models (Details matter !)

- Utility-based Data Mining as
  - “On-line” Reinforcement Learning and special cases thereof ?
  - “Batch” Reinforcement Learning and special cases thereof?

- Issues
  - “On-line”: Exploration v.s. Exploitation Trade-off
  - “Batch”: Bias Correction
  - Combining the two (!)

References

Classic Learning Models in Computational Learning Theory

L. G. Valiant, ‘A theory of the Learnable’, Communications of the ACM, 1984", pp.1134-1142.

D. Haussler, ‘Decision theoretic generalizations of the PAC model for neural net and other learning applications’ Information and Computation, 100(1), 78—150, 1992.

D. Angluin, "Queries and concept learning", Machine Learning, vol. 2, 319--342, 1987.

N. Littlestone, ‘Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm’, Machine Learning, 2:285--318, 1988.

N. Cesa-Bianchi et al, ‘How to use expert advice’, Journal of the ACM, 44(3):427-485, May 1997.

Online Active Learning

L. P. Kaelbling: Associative Reinforcement Learning: Functions in k-DNF. Machine Learning 15(3): 279-298 (1994)

D. A. Berry, B. Fristedt, Bandit Problems: Sequential Allocation of Experiments. Chapman & Hall, London, 1985.

N. Abe, A. Biermann, and P. Long, ‘Reinforcement Learning with Immediate Rewards and Linear Hypotheses,’ Algorithmica, 37, 263-293, 2003.

J. Takeuchi, N. Abe and S. Amari, ‘The Lob-Pass Problem’, Journal of Computer and System Sciences, 61(3), 2000

Cost-sensitive Learning and Economic Learning

B. Zadrozny, One-Benefit Learning: Cost-Sensitive Learning with Restricted Cost Information, this volume.

B. Zadrozny and C. Elkan. Learning and Making Decisions When Costs and Probabilities are Both Unknown, KDD’01.

P. Melville et al, Economical active feature-value acquisition through expected utility estimation, this volume.

F. Provost, ‘Toward Economic Machine Learning and Utility-based Data Mining’, this volume.

Applications

N. Abe & A. Nakamura, ‘Learning to Optimally Schedule Banner Ads..’ ICML’99

E. Pednault, et al, Sequential Cost Sensitive Decision Making with Reinforcement Learning , KDD’02.

N. Abe, N. Verma, C. Apte and R. Schroko, ‘Cross Channel Optimized Marketing by Reinforcement Learning’, KDD’04.