Supervised Learning Using Artificial Prediction Markets

Adrian Barbu

Department of Statistics

Florida State University

Joint work with Nathan Lay, FSU Dept. of Scientific Computing

Overview

Main Contributions

- A mathematical theory for Artificial Prediction Markets
  - Introducing the Artificial Prediction Market
  - Equations governing the market equilibrium price.
  - Equilibrium price uniqueness.
  - Relation to existing aggregation methods:
    - Linear Aggregation
    - Logistic Regression
  - Experimental comparison with Random Forest on real and synthetic data.

Notation

Main goal: Classification

- Let $\Omega \subset R^F$ be the feature space
- K possible classes (outcomes) {1,…,K}

Supervised learning:

- Given training examples:
  - $(x_i, y_i) \in \Omega \times \{1,\ldots,K\}$
- Learn a function

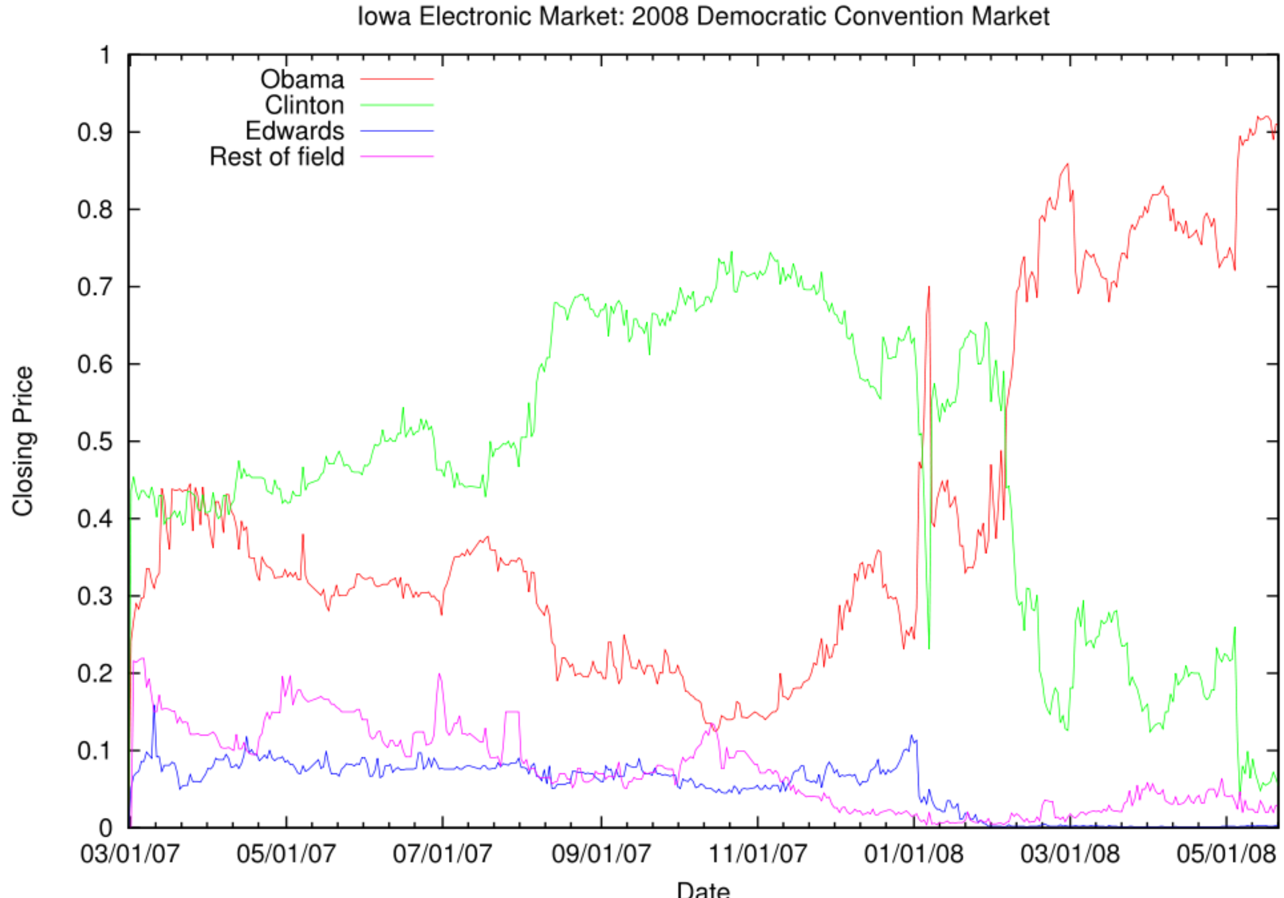
$$f(\mathbf{x}) : \Omega \to [0,1]^K, f(\mathbf{x}) = (f_1(\mathbf{x}), ..., f_K(\mathbf{x}))$$

such that $f_i(x)$ is a good approximation of $p(Y=i|x)$

Real Prediction Markets

- Forums (e.g. on the web) where contracts on future events are bought and sold.
- Contract prices are based on supply and demand.
- Contract price fuses the information possessed by the participants
  - Confident participants “put their money where their mouth is”
- Have successfully predicted outcomes of elections and sports games.
- E.g. the Iowa Electronic Market

The Iowa Electronic Market



- Market setup:
  - Contracts for each outcome are bought and sold at market price $0 < c < 1$
  - Each contract pays $1 if outcome is realized.
  - Market price of contract represents a good approximation of the probability that the corresponding event occurs

The Artificial Prediction Market

- A simulation of the Iowa Electronic Market:
  - Each class k = 1, ...,K corresponds to a contract type
  - Market price is a vector **c** = $(c_1,\ldots, c_K)$. We enforce $\sum c_k=1$
  - Contract for class k sells at market price $0<c_k<1$ and pays 1 if the outcome is k.
- A market participant is not a human, but a pair of:
  1. A budget (or weight) β_m
     - Based on past ability in predicting correct class
  2. A betting function

$$\phi(\mathbf{x}, \mathbf{c}) : \Omega \times [0,1]^K \rightarrow [0,1]^K$$

Betting Function

- Is the percentage of its budget a participant will allocate for each class.
- It is a function

$$\phi(\mathbf{x}, \mathbf{c}) : \Omega \times [0,1]^K \to [0,1]^K, \sum_{k=1}^{K} \phi_k(\mathbf{x}, \mathbf{c}) \le 1$$

- Depends on
  - The feature vector x
    - E.g. through a learned classifier that predicts the outcome

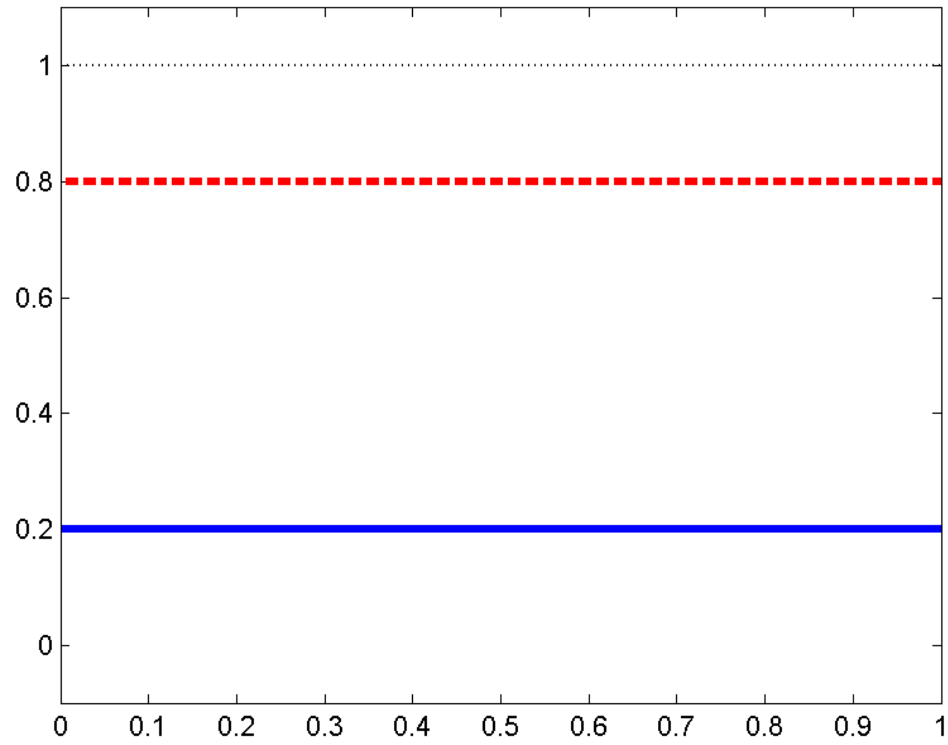
$$h : \Omega \to [0,1]^K, \sum_{k=1}^{K} h_k(\mathbf{x}) = 1$$

  - The market price c.

Constant Betting Functions

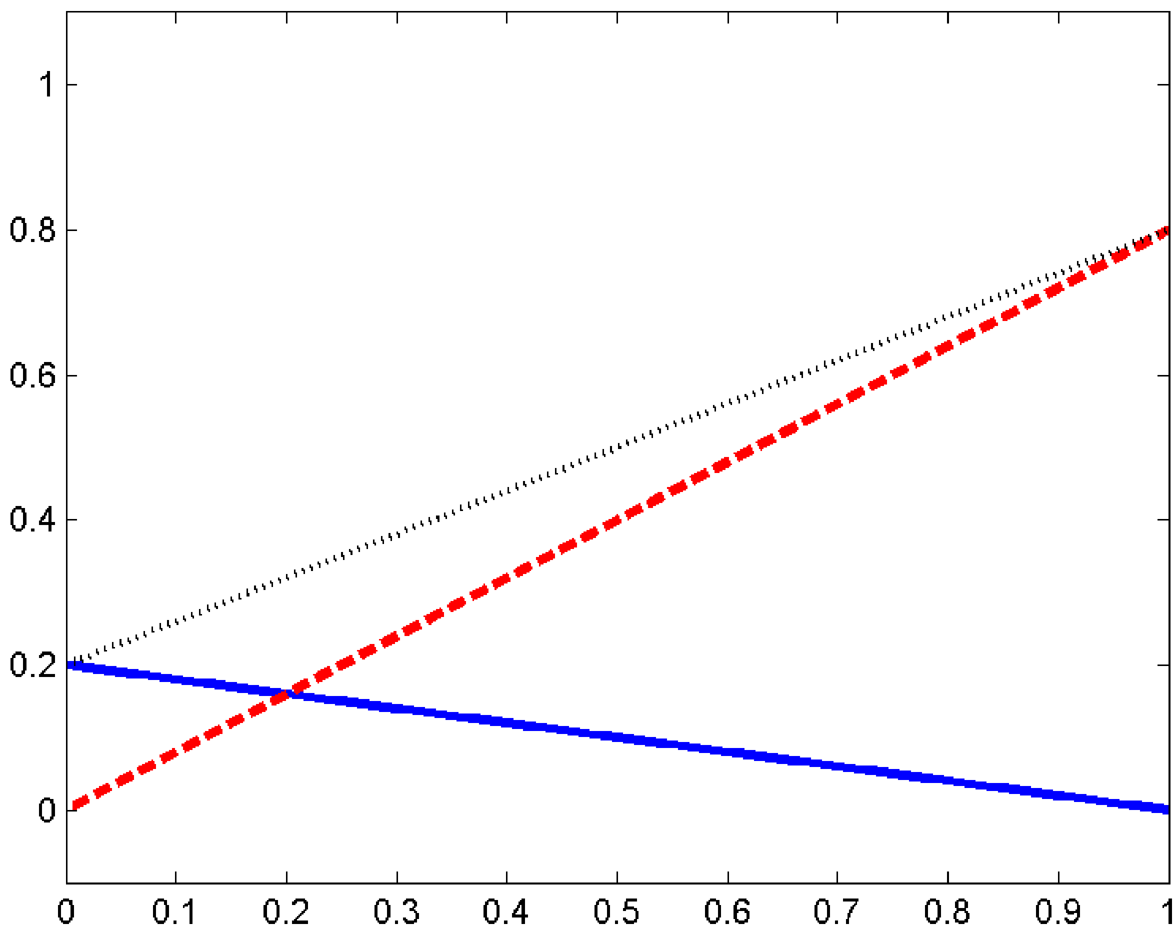
- Allocate same amount independent of the price

$$\phi_k(\mathbf{x}, \mathbf{c}) = h_k(\mathbf{x})$$

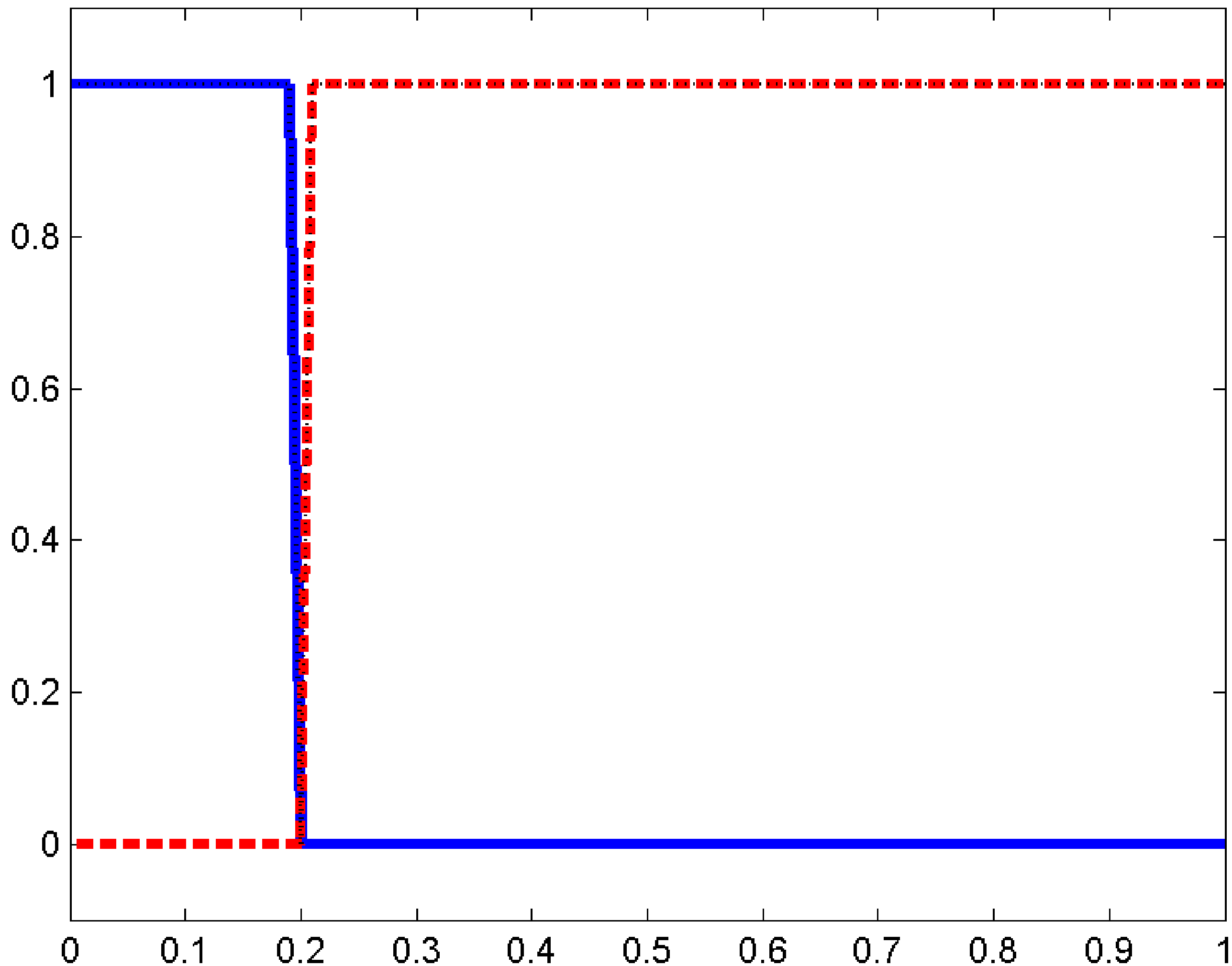


Linear Betting Functions

$$\phi_k(\mathbf{x}, \mathbf{c}) = (1 - c_k) h_k(\mathbf{x})$$



Aggressive Betting Functions



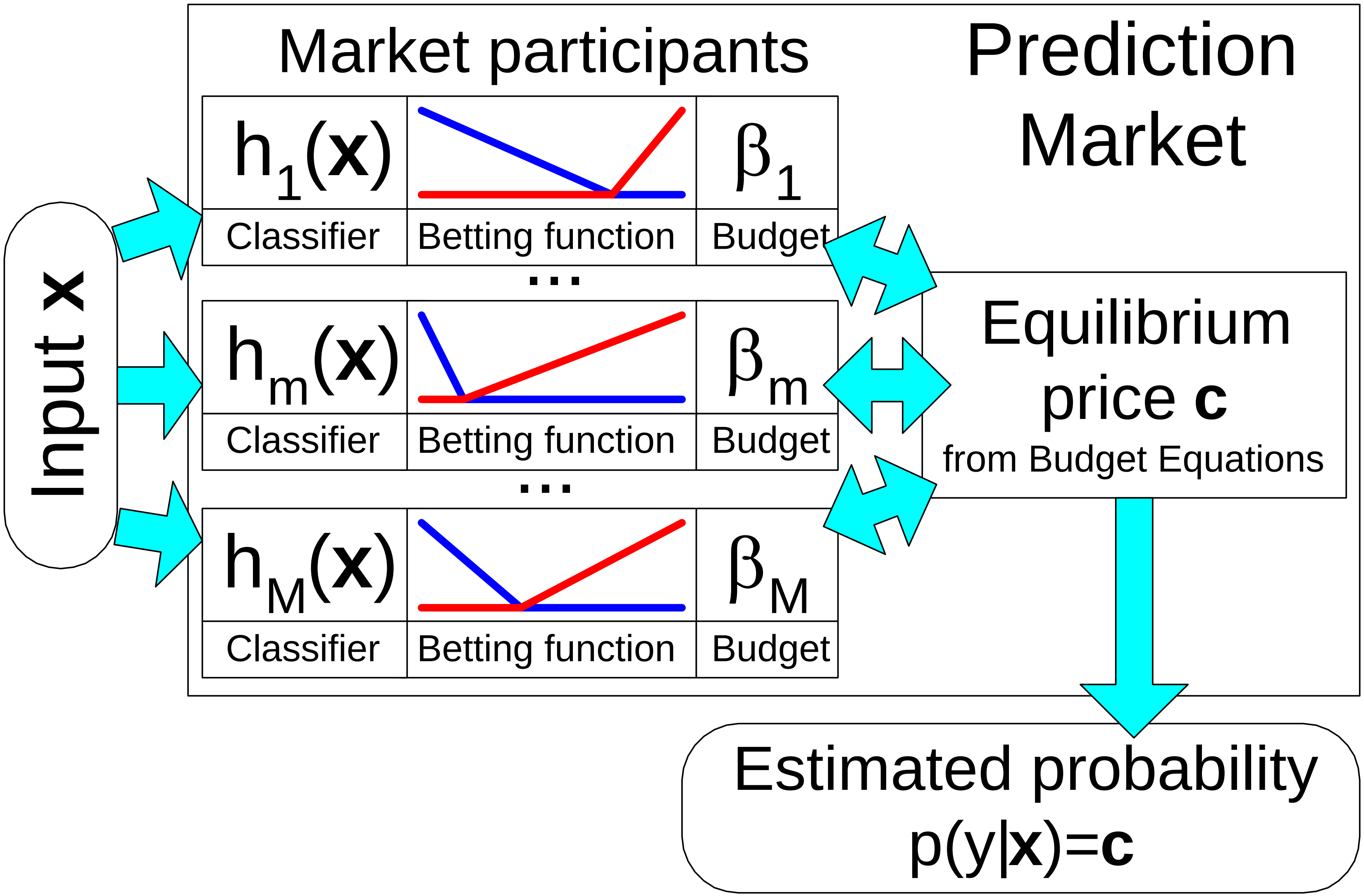
Avoiding Price Fluctuation

The Artificial Prediction Market is not a real market!

For each given observation $x \in \Omega$

- We know what each classifier will do for any market price c.
- We can use this to avoid price fluctuation:
  - Can find the equilibrium price numerically based on some equations.
  - The market is started at equilibrium price
  - All contracts are sold and bought instantly at that price.
  - The price does not change.

Artificial Prediction Market Diagram



Supervised Training of the Market

Idea: train the market participants

- For each training example (x_i, y_i) let participants bid and reward those that bought contracts for the correct outcome.
- Classifiers will get rich or poor depending on their prediction ability.
- The result is a market with trained participants.
- We will see that prediction performance is significantly better than an untrained market

Supervised Training of the Market

- The proportion of the budget spent on contracts for class k at price c is $\phi_k(x,c)$
- Thus the number of contracts purchased for class k is

$$n_{km} = \frac{\beta_m \phi_{km}(\mathbf{x},\mathbf{c})}{c_k}$$

Training: For each training example (x,y), run the Market Update (x,y) i.e.:

- Find the market equilibrium price c.
- For each participant subtract from β_m the amount bet

$$\sum_{k=1}^{K} \beta_m \phi_{km}(\mathbf{x},\mathbf{c})$$

- Add to β_m the amount won n_{ym}

Market Update (x,y)

1. Compute equilibrium price **c** based on the price equations.
2. For each m=1,...,M
   - Update participant m's budget as

$$\beta_m \leftarrow \beta_m - \sum_{k=1}^{K} \beta_m \phi_{km}(\mathbf{x}, \mathbf{c}) + \frac{\beta_m \phi_{ym}(\mathbf{x}, \mathbf{c})}{c_y}$$

Budget Conservation

Main requirement:

- The total budget must remain the same after each market update, independent of the outcome y.
- This means:

$$\sum_{m=1}^{M} \sum_{k=1}^{K} \beta_m \phi_{km}(\mathbf{x}, \mathbf{c}) = \sum_{m=1}^{M} \frac{\beta_m \phi_{ym}(\mathbf{x}, \mathbf{c})}{c_y}$$

- This must hold for any y, since the market price c must depend only on x for prediction purposes.

Price Equations

Theorem.

The total budget $\sum \beta_m$ *is conserved after the* Market Update(x,y), *independent of the outcome y, if and only if there exists* $n \in R^+$ *such that*

$$\sum_{m=1}^{M} \beta_m \phi_{km}(\mathbf{x}, \mathbf{c}) = n c_k, \qquad \forall k = 1, \ldots, K$$

- These are the equations that govern the market price c, together with

$$\sum_{k=1}^{K} c_k = 1$$

Constant Betting is Linear Aggregation

- In the case of constant betting functions

$$\phi_k(\mathbf{x}, \mathbf{c}) = h_k(\mathbf{x})$$

the budget equations become

$$\sum_{m=1}^{M} \beta_m h_{km}(\mathbf{x}) = nc_k, \quad \forall k = 1, ..., K$$

- Can prove that $n = \|\beta\|_1 = \sum_{m=1}^{M} \beta_m$
- We obtain linear aggregation of classifiers

$$\mathbf{c} = \frac{1}{\|\beta\|_1} \sum_{m=1}^{M} \beta_m h_m(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m h_m(\mathbf{x})$$

existent in Adaboost, Random Forest, etc

Constant Betting Update Rule

- We obtain a new online learning rule for linear aggregation:

$$\beta_m \leftarrow \beta_m(1-\eta)) + \eta\|\beta\|_1 \frac{\beta_m h_{ym}(\mathbf{x})}{\sum_{i=1}^{M} \beta_i h_{yi}(\mathbf{x})}$$

- Bunea & Nobel, 2008 introduced online linear aggregation with exponential weights, different from this rule.

Price Uniqueness

- Reasonable assumption:
  - Betting functions should be monotonically decreasing i.e. if contract price is higher, invest less.

Theorem (Monotonic Betting Functions).

If all betting functions $\phi_{km}(x,c_k)$, $m = 1, ...,M$, $k = 1, ...,K$ are continuous and monotonically decreasing, then for each Market Update(x,y) *there is a unique price $c=(c_1,...,c_K)$ such that the total budget $|\beta|_1$ is conserved.*

- It holds for the constant, linear and aggressive betting functions.

Two Class Formulation

- Write c=(1−c,c), then the budget is conserved if and only if

$$c\sum_{m=1}^{M}\beta_m\phi_{0m}(\mathbf{x},c)=(1-c)\sum_{m=1}^{M}\beta_m\phi_{1m}(\mathbf{x},c)$$

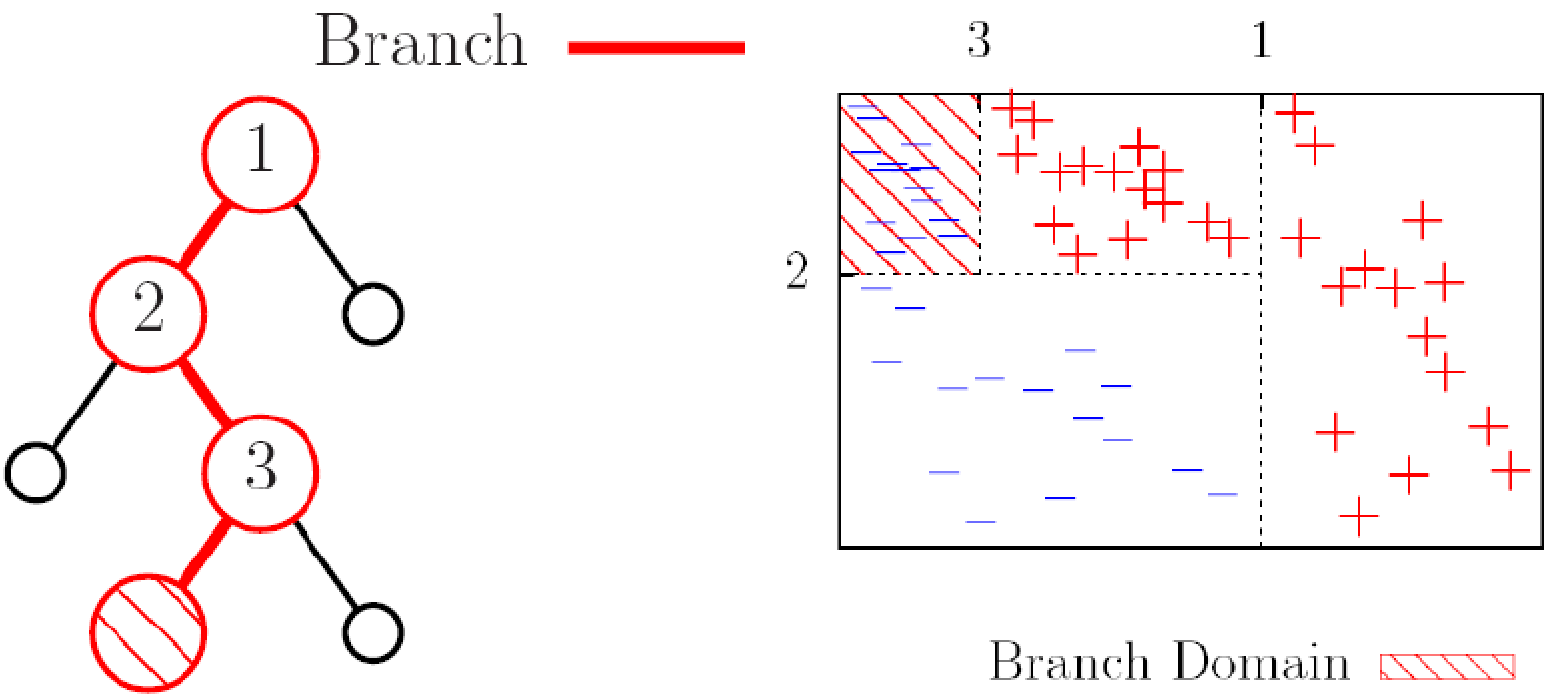
- This again has a unique solution that can be found easily by the bisection method.

Specialization

- In Boosting and Random Forrest, all classifiers are aggregated for any observation $x \in \Omega$.
- The Market participants can be specialized
  - A participant can predict very well on a subregion of Ω.
  - It will not bet on any x outside its region.
  - For each observation, a different subset of classifiers could participate in betting
  - Example: a leaf node of a random tree

Decision Tree Rules as Specialized Classifiers

- Decision tree rules (leaves) can perfectly classify training data in their specialized domain.



Related Work in Economics

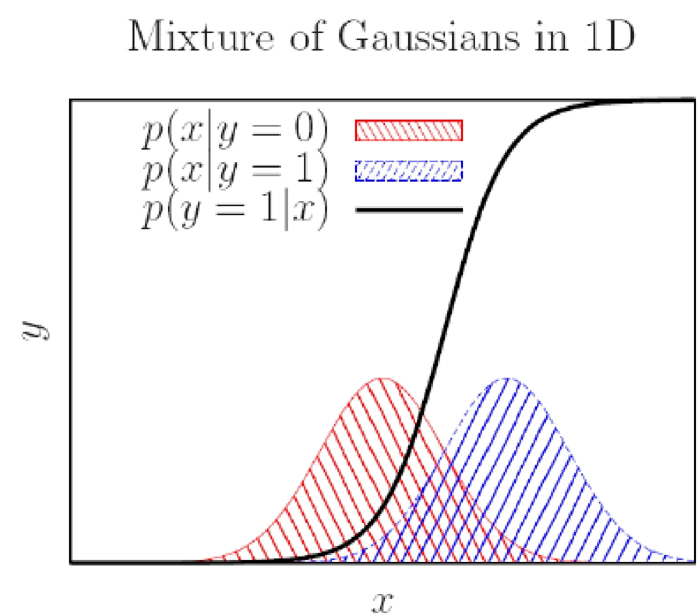
- Extensive recent work in Economics.
  - Plott'03, Manski'06, Perols'09 study the information fusion capability of the market.
  - Plott'03, Perols'09, use the parimutuel betting mechanisms, not the Iowa Market
  - None of them uses a supervised approach or specialization
  - All focus on two-class problems
  - Perols'09 evaluates on real datasets but participants are not trained (have equal budgets).
- We will see that training the participants significantly improves market accuracy.

Related Work in Statistics and Machine Learning

- Specialization is a sort of reject rule (Chow'70, Tortorella'04)
  - But for each participant
  - Not for the aggregated classifier
  - An overall reject rule can be obtained from the individual reject rules
- Delegated Classifiers (Ferri'04)
  - Two classifiers with disjoint specialization domains
  - First classifier decides on easy instances
  - Second classifier decides on the rest
- Rule Ensemble (Friedman'08) combines leaves of random trees with linear aggregation.

Results on Synthetic Data

- Two-class datasets coming from two 100D Gaussians.
  - True probability p(y = 1|x) can be computed analytically.
  - Evaluated for 50 Bayes error increments from 0.01 to 0.5.
  - Gaussian centers placed so that desired Bayes error is obtained
  - For each Bayes error, 100 datasets of size 200 were created.
- Totally 5000 datasets.



Evaluation Details

- 50 random trees were trained for each dataset.
- The tree branches were used as the market participants.
- Market Update was run on each data set
- Betting functions were multiplied by $\eta = 0.1$ to limit the maximum bet.

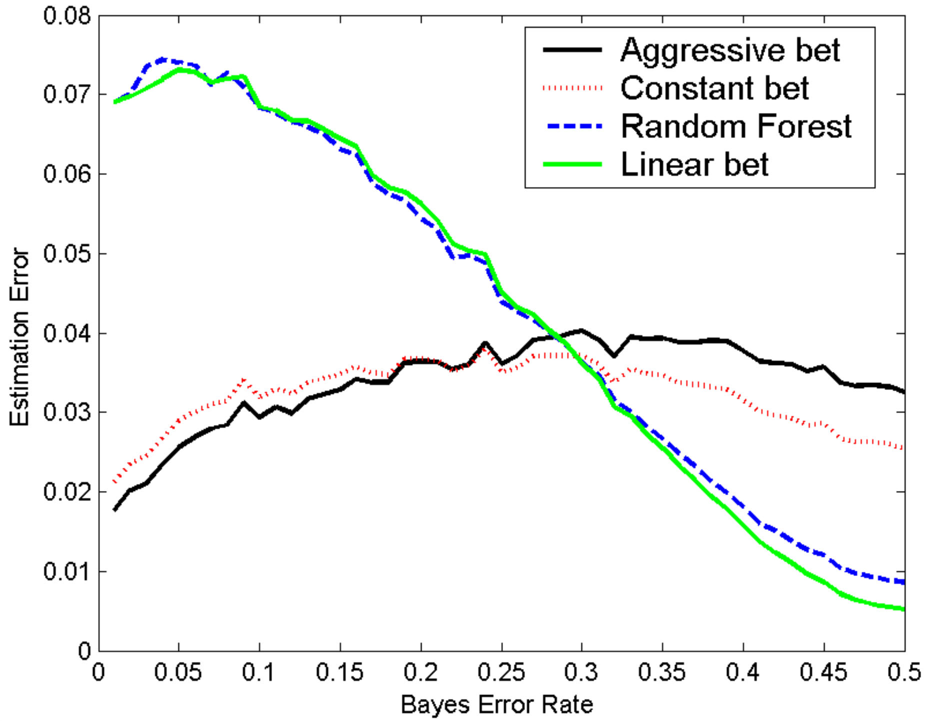
Markets evaluated:

1. Random Forest = Constant betting with equal budgets
2. Trained Constant Betting
3. Trained Linear Betting
4. Trained Aggressive Betting

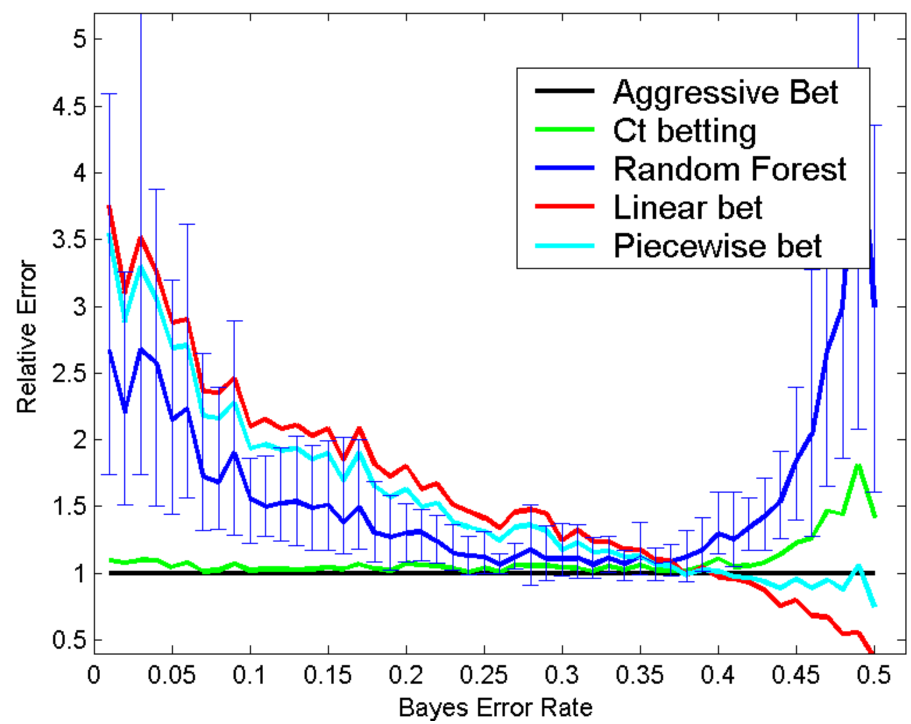
Two Evaluations:

- Probability Estimation Error as $E[||p^* - \hat{p}||_2]$ approximated with a sample of size 1000.
- Misclassification error on a sample of size 1000

Probability Estimation Evaluation

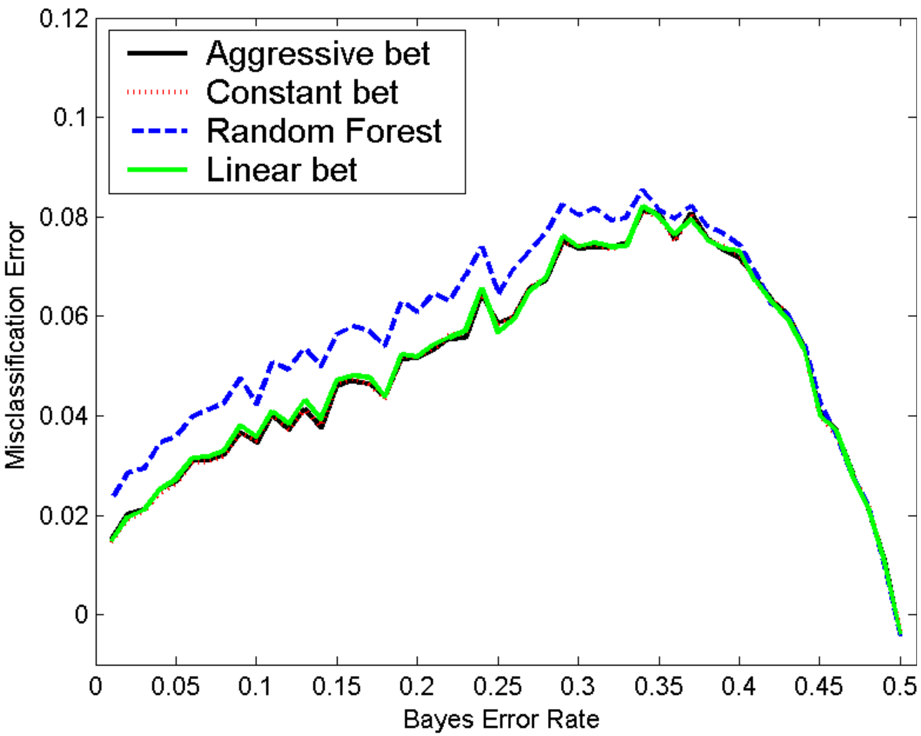


Conditional probability estimation error vs problem difficulty

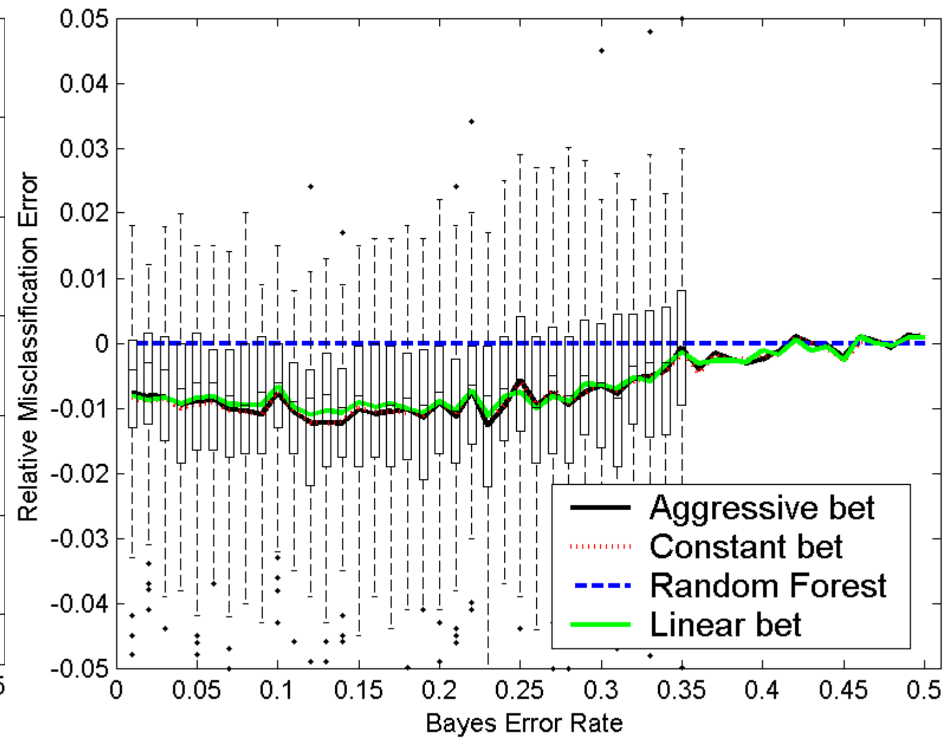


Relative estimation error vs problem difficulty

Probability Estimation Evaluation



Misclassification error minus Bayes error vs problem difficulty



Relative misclassification error vs problem difficulty

Real Data Results

- 21 datasets from the UC Irvine Machine Learning repository
  - Many are small (≈ 200 examples).
  - Training and test sets are randomly subsampled, 90% for training and 10% for testing.
  - Exceptions are satimage and poker datasets with test sets of size 2000 and 10^6 respectively
- All results are averaged over 100 runs.
- Significance comparison tests (α<0.01):
  - Mean differences from RF results from Breiman'01
  - Paired t-tests with our RF implementation

Results on UCI Data

Table 1. Misclassification errors in percent (%) for 21 UCI datasets from the UC Irvine Repository. The markets evaluated are Random Forest (RF), and Constant (CB), Linear (LB) and Aggressive (AB) betting.

Data	Train Size	Test Size	Feat.	Cls	ADB	RFB	RF	CB	LB	AB
cancer	699	–	9	2	3.2	2.9	3.0	2.9	2.9	2.9
sonar	208	–	60	2	15.6	15.9	14.8	14.1	14.3	14.1
vowel	990	–	10	11	4.1	3.4	3.3	3.1 •	3.2	3.1 •
diabetes	768	–	8	2	26.6	24.2	23.4	23.4	23.4	23.5
ecoli	336	–	8	8	14.8	12.8	13.1	13.0	13.0	13.1
german	1000	–	20	2	23.5	24.4	23.7	23.7	23.6	23.7
glass	214	–	9	6	22.0	20.6	20.0	20.1	20.1	20.2
ionosphere	351	–	34	2	6.4	7.1	**5.8**	**5.7**	**5.7**	**5.7**
letter-recognition	20000	–	16	26	3.4	3.5	3.3	3.2 •	3.2 •	3.2 •
satimage	4435	2000	36	6	8.8	8.6	***8.8***	8.6 •	8.7 •	8.6 •
image	2310	–	19	7	1.6	2.1	**1.8**	**1.6 •**	**1.6 •**	**1.6 •**
vehicle	846	–	18	4	23.2	25.8	24.8	**24.5**	**24.6**	**24.5**
voting-records	435	–	16	2	4.8	4.1	3.0	3.0	3.0	3.0
car	1728	–	6	4	–	–	2.4	1.2 •	1.4 •	1.2 •
poker	25010	1000000	10	10	–	–	38.0	35.7 •	36.0 •	35.7 •
cylinder-bands	540	–	39	2	–	–	20.3	20.2	20.1	20.0
yeast	1484	–	9	10	–	–	35.9	35.8	35.7	35.8
magic	19020	–	10	2	–	–	12.0	11.7 •	11.8 •	11.8 •
king-rook-vs-king	28056	–	6	18	–	–	21.6	11.0 •	11.8 •	11.0 •
connect-4	67557	–	42	3	–	–	19.9	16.7 •	16.9 •	16.7 •
splice-junction-gene	3190	–	59	3	–	–	4.9	4.6 •	4.6	4.6 •

- ADB and RFB are Adaboost and Random Forest from Breiman'01
- CB and AB perform best and significantly outperform RF in many cases
- Trained markets never performed significantly worse than RF

Conclusion

A theory for Artificial Prediction Markets based on the Iowa Electronic Market:

- Online, supervised training of participants by updating their budgets.
- Price equations that guarantee total budget conservation after each budget update.
- Equilibrium price is unique under some mild assumptions.
- Specialized participants are fused very well by the market.
- Significantly outperforms Random Forest in many cases, in both prediction and probability estimation.