Prediction Market

Submitted as part of Indepedent Study for Sping 18

Sankarshan Damle
sankarshan.damle@research.iiit.ac.in

Sujit Gujar
sujit.gujar@iiit.ac.in

Abstract

Once described as "a great way to mix science with gambling" by Anna Dreber, prediction markets have gained immense popularity over the years. Predicted to be a multi-billion dollar industry by the turn of this decade, the area and has been a subject to a wide range of research.

This report describes prediction markets from the ground up, specifying the tools required to set up such a market. Further, it describes some important use cases as well as the challenges faced by such markets. It also summarizes, compares and contrasts some notable works in this field - highlighting their motivation and the novel approach used. Lastly, it analyzes PPX, an extension to [5], wherein we look on how the agents behave when the beliefs are biased towards project getting provisioned or not provisioned.

Contents

1 Introduction

Prediction markets are popular mechanism for generating probability consensus for future events. It has been estimated that the predictive analytic market would be worth $12.41 Billion by 2022 [1] A prediction market aggregates agents' opinions about events of particular importance or interest. Such an event may include predictions about the winner of the Presidential Elections, weather forecast, whether an epidemic will hit some part of the world, scientific discoveries etc. The efficiency of a prediction market lies in the fact that it relies on the collective view of many and not just one person's research. It is believed that a group of people will hold a wider range of information on the market, over single entities.

However, an agent would only be willing to share his view of a market if he is incentivized to do so. For this, prediction markets use *scoring rules*. A scoring rule is a function which measures the accuracy of probabilistic predictions, and pays an agent proportional to the difference in his prediction and the final outcome. An agent can, of course, elicit his valuation different from his true preference and hence we prefer scoring rules which maximizes an agent's utility when he elicits his true preference. Such scoring rules are called *proper*. Further, if a proper scoring rule attains it's maximum only for one particular valuation, then it is called *strictly* proper scoring rule. The logarithmic scoring rule, $S_i(\vec{r}) = a_i + blog(r_i)$, is one such example [10].

Since the scoring rule pays an agent based on the final outcome of the event, we can not set up a market for events which do not have a known outcome. Such events include annonating an image with a set of keywords, counting the number of specific items in an image etc. Such information is useful to generate datasets for machine learning algorithm, and hence their prediction is an important research question. [7] provides a decentralized, peer-to-peer prediction market for the same. In these, instead of a paying with respect to a definite outcome, agents are paid in proportion to how close there valuation is with the aggregate valuation (*relative truth*) or with respect to the valuation of a randomly chosen peer. These peer-to-peer markets can also be implemented as smart contracts [2].

1.1 The need for Prediction Markets

The prediction markets are a powerful way to turn individuals' guesses into forecasts, the results of which have often proven to be immaculately accurate. Exchanges such as BetFair and Iowa Electronic Market (IEM), use prediction markets to make forecasts of all kinds. The exchanges' US Presidential standings have turned out to be quite popular. In addition, these markets have also been used to predict almost anything - from the outcomes of sporting events to the results of business decisions.

These markets have allowed people to aggregate information without the biases that plague traditional forecasting methods - such as polls or expert analysis.

1.2 Challenges

Despite the success of prediction markets, researchers have managed to show various challenges involved in these. An obvious difficulty in a prediction market is the calculation of an agent's score for problems with no *definite outcome*. These problems can have important research implications, like the generation of data-sets for machine learning, or problems with *subjective truth*, like whether the world will end by the next century. [7] quantify these through peer-to-peer prediction markets, using labor markets.

Other challenges include free riding, getting agents to elicit their true valuation and liquidity. An agent can manipulate the outcome by sending misleading signals from inside or outside the markets. [5] provides a prediction market mechanism for crowdfunding which tackles these difficulties.

In addition, combinatorial markets increase expressiveness, by allowing an agent to trade different securities allowing for more effective information aggregation. However, the outcome space in this is exponential, and thus information aggregation becomes difficult. [6] proposes a combinatorial tournament market with polynomial time update - in a restricted setting. [10] argues that we can preserve conditional probabilities for events, under certain conditions, using the *logarithmic market scoring rule* and thus reducing the computational time for updates.

2 Prediction Markets

The prediction market is a collection of people speculating on a variety of events — exchange averages, election results, commodity prices, quarterly sales results or even such things as gross movie receipts. In this, agents trade assets whose value is tied to a particular observation, for example, which candidate wins the next presidential election, and asset prices determine a probability distribution over a set of possible outcomes.

Prediction markets require the interaction of buyers and sellers to aggregate the distribution or for price discovery. However, this requires that for every buyer there is a seller willing to trade with him. If there is no one willing to trade, then such a market has low liquidity and is undesirable. For this we make use of an *automated market maker*. An automated market maker guarantees that a market is *always available for trade*. These market maker's take the opposite of a trade, and thus can even incur losses. They also set up price for each stock using *market scoring rules* which also provides an upper bound on the market maker's potential loss.

3 Peer-to-Peer Prediction Markets

A Peer-to-Peer prediction market has the following setting:

- Each agent gets a prior signal s_i for an event w. The distributions of w and s_i *given* w are common knowledge, but w cannot be verified i.e., there is no known outcome.
- Each agent i reports his preferred s'_i.
- The market maker randomly picks a reference agent j and calculates $s^*_i = P(s'_i|s_i)$ for each agent i.
- Each agent i will be rewarded according to this s^*_i.

It has been shown that for the above setting truth telling i.e., agents reporting their true beliefs, will result in a Bayesian Nash Equilibrium. However, it won't be the only equilibrium, and the setting requires a significant number of truth tellers. Later research has shown that there are also ways to make truth telling a *unique* equilibrium in this setting.

4 Preliminaries

In this section we shall look at some tools which act as market requirements for the creation of a prediction market.

4.1 Fair Bet

A fair bet is one for which the expected value of the payoff is zero, after accounting for the cost of the bet. For example, suppose A offers to pay you $2 if a fair coin lands heads, but you must ante up $1 to play. Your expected payoff is -$1+ $0xP(tails) + $2xP(heads) = -$1 + $2x50% =$0. This is a fair bet-in the long run, if you made this bet over and over again, you would expect to break even. The fair bet eq is given by,

$$\sum_i y_i m_i(\bar{x}) = 0 \qquad [10],$$

where y_i is the change in the agent's contribution with time and $m_i(\cdot)$ is the instantaneous price of the assets.

4.2 Scoring Rules

Let, p_i be agent's belief about the probability that w_i will occur and $S_i(\vec{r})$ be the payment made to agent i if he reports distribution $\vec{r}$ if outcome is w_i, then a *proper scoring rule* is,

$$\vec{p} = \text{argmax}_{\vec{r}} \sum_i p_i S_i(\vec{r}).$$

A scoring rule is called *strictly* proper when $\vec{p}$ is unique. For example, consider the Logarithmic scoring rule, [10]

$$S_i(\vec{r}) = a_i + blog(r_i),$$

such that $r_i \leq 1$ and a_i is big enough for agents to be willing to participate. This satisfies a lot of desirable properties; for example, this is the only rule in which an agent's payoff depends only on the probability he assigned to the actual event. Thus, LSR is *local* i.e., we can look at a n-valued event as n binary events and the score remains the same. For binary outcomes, an agent's expected payment would be,

$$p(a + blog(r)) + (1 - p)(a + blog(1 - r)),$$

derivative w.r.t. r would give,

$$p - rp - r + rp = 0,$$

which implies $p = r$. Further, the second derivative is negative, thus LSR is proper.

Table 4.2 describes some popular *proper* scoring rules [10].

Proper Scoring Rules	
Scoring Rule	Given By
Quadratic	$s_i = a_i + br_i - b\sum_j r_j^2/2$
Spherical	$s_i = a_i + br_i/(\sum_j r_j^2)^{1/2}$
Logarithmic	$s_i = a_i + blog(r_i)$
Power Law	$s_i = a_i + b\alpha \int_0^{r_i} \rho_i^{\alpha-2} d\rho_i - b\sum_j r_j^\alpha$

4.2.1 Market Scoring Rules

We can create a market using scoring rules as a *Market Scoring rule*. In this we have [10],

- Sequential Trading
- To update the distribution from $\vec{r}$ to $\vec{r'}$, the agent must pay the previous agent $S_i(\vec{r})$.
- Therefore, the payoff is $S_i(\vec{r'})$.
- The final $\vec{r}$ is the market's prediction.
- The market maker's loss in this case is bounded by $S_i(\vec{e}_i) - S_i(\vec{r}_0)$.

However, in this no real contracts are traded, as well as the agents can participate only once. These limitations may make the market less appealing to an agent, and thus we use *cost functions*.

4.3 Cost Function

In this, instead of trading probabilities, we trade contracts of the form *"Contract pays $1 if the event happens, and $0 otherwise"*. Such a contract are called *Arrow-Debreu* contracts. Here, $\vec{q}$ is a vector indicating the total number of shares of each type ever sold and $\vec{q}_i$ is the number of shares sold for type i. Then, to change the distribution of $\vec{q}$, through buying or selling shares, an agent must pay

$$C(\vec{q}_{new}) - C(\vec{q}_{old}),$$

where $C(\cdot)$ is the cost function. Further, the instantaneous price of a share i is

$$p_i(\vec{q}) = \frac{\partial C(\vec{q})}{\partial q_i},$$

such that $\sum_{i=1}^m p_i(\vec{q}) = 1$, where m is the total number of agents. Some of the desired properties of a cost function are [5],

- The function should be differentiable (to calculate prices)
- It should be monotonically increasing

Two popular cost functions that satisfies these properties are (for a market with binary outcomes) [5]:

$$C(\vec{q}) = b\ln(e^{q_{w_0}/b} + e^{q_{w_1}/b})$$

$$C(\vec{q}) = \frac{q_{w_0} + q_{w_1}}{2} + \frac{q_{w_0}^2 + q_{w_1}^2}{4b} - \frac{(q_{w_0} + q_{w_1})^2}{8b} - \frac{b}{2}.$$

Note: Interestingly, given $S(\cdot)$ we can calculate $C(\cdot)$ and vice versa.

As mentioned earlier, aggregating information in a combinatorial setting allows an agent to be more expressive. However, in such setting the outcome space is exponential, and thus analyzing these markets is tough. We now look at Bayesian networks that can be helpful in representing such outcome space.

4.4 Bayesian Network

This is a data structure to represent probability distributions $P(X_1, \ldots, X_n)$ especially for cases where dependencies in the distribution are sparse. This is represented as Directed acyclic graph (DAG), where each vertex correspond to a random variable and edges represent conditional (in)dependence. An important property of such a network is

$$P(X_1, \ldots, X_n) = \prod P(X_i | parents(X_i)),$$

i.e., each node X_i consists of a probability conditioned only on it's parents. In general, calculating $P(X_i = x_i)$ from this is NP-Complete. Figure 1 shows an example of a simple Bayesian Network.

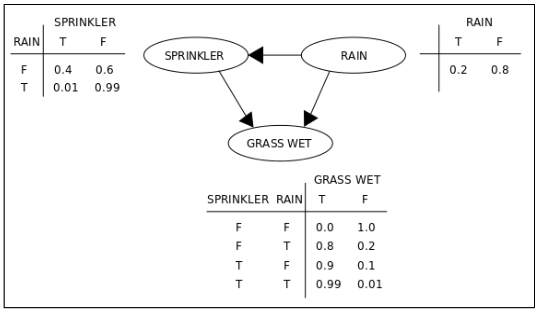

Figure 1: A simple Bayesian Network [3].

4.5 Nash Equilibrium

We'll say that a market is in NE when all agents already maximized their profits, and any further action from any agent will damage his profit. Most importantly, in a NE rational agents stop trading. A strategy profile $\psi = (\psi_1^*, \ldots, \psi_n^*)$, is said to be in Nash Equilibrium if,

$$u_i(\psi_i^*, \psi_{-i}^*; \theta_i) \geq u_i(\tilde{\psi_i}, \psi_{-i}^*; \theta_i) \qquad \forall i.$$

Nash Equilibrium in a prediction market allows us to argue about the strategy implemented by agents. Prediction Markets with a perfect Nash equilibrium are preferred.

A set of strategies also form *Subgame Perfect Nash Equilibrium*, if the agent's strategies are a Nash Equilibrium in every sub-game.

		PRISONER 2	
		Confess	Lie
PRISONER 1	Confess	-8 , -8	0 , -10
	Lie	-10 , 0	-1 , -1

Figure 2: The Prisoner's Dilemma: The strategy (Confess, Confess) is a NE, since after fixing the strategy of one, the other prisoner has no incentive to change his strategy. [4]

4.5.1 Bayesian Nash Equilibrium

Bayesian Nash Equilibrium is an extension of Nash Equilibrium and is a strategy profile $\psi = (\psi_1^*, \ldots, \psi_n^*)$ if,

$$E_{\theta_{-i}}[u_i(\psi_i^*(\theta_i), \psi_i^*(\theta_{-i})); \theta_i, \theta_{-i}] \geq E_{\theta_{-i}}[u_i(\psi_i(\theta_i), \psi_i^*(\theta_{-i})); \theta_i, \theta_{-i}].$$

Intuitively, each type of player chooses a strategy that maximizes expected utility given the actions of all types of other players and that player's beliefs about others' types.

A set of strategies also form *Subgame Perfect Bayesian Nash Equilibrium*, if the agent's strategies are a Bayesian Nash Equilibrium in every sub-game.

We will now categorize the research papers covered in this report based on the fundamental area which they cover, while stating the challenge addressed and the novel approach proposed to encounter those by them.

5 Research Paper Index

The following table summarizes the papers covered in this report-

Table Index			
Paper	**Appeared in**	**Problem Addressed**	**Novel Approach**
Prediction Market			
Logarithmic Market Scoring Rules for Modular Combinatorial Information Aggregation[10]	JPM 2002	The paper introduces a rule wherein multiple agents can share their beliefs without the need to pay each one, and which generates a single (unified) prediction	Proposes the Logarithmic Market Scoring Rule (LMRS)
Crowdfunding Public Projects with Provision Point: A Prediction Market Approach[5]	ECAI 2016	To improve the success of provision point projects, by incentivizing agents to contribute, and contribute as soon as they arrive	Uses prediction market for bonuses
Pricing Combinatorial Markets for Tournaments[6]	STOC 2008	Combinatorial prediction markets are more expressive, but their outcome space is exponential. This paper provides an algorithm which provides polynomial time update for a single-elimination tournament with restricted setting	Model the tournament as a Bayesian Network
Peer-to-Peer Prediction Market			
Incentives to Counter Bias in Human Computation[7]	HCOMP 2014	Deals with problems in labor markets including common bias, no known common answer, with existing hints etc	Proposes *Peer Truth Serum* that provides bonus that lets the workers evaluate their answers from a game-theoretic perspective
A Bayesian truth serum for subjective data[11]	American Association for the Advancement of Science 2004	Deals with the problems in subjective judgments - due to them not having a public criteria to determine judgmental truthfulness	Proposes *Bayesian Truth Serum* which assigns high scores to answers that are more common than collectively predicted or *surprisingly common*.
Crowdsourced Outcome Determination in Prediction Markets[9]	AAAI 2017	Deals with the challenges in a decentralized prediction market such as outcome ambiguity, peer prediction and conflict of interest	Proposes a mechanism using a prediction market for trading and the peer prediction mechanism "1/prior" to calculate incentives
Differential Privacy			
Constrained-Based Differential Privacy for Mobility Services [8]	AAMAS '18	Deals with the privacy leaks of sensitive user data in relation to that collected through mobile and wireless communication to make accurate prediction and mobility traces	This paper uses *diffrential privacy* to release this data for transportation systems. It proposes the idea of *Constraint Based Differential Privacy* (CBDP), which formulates the redistribution of private data as an optimization problem.

6 Research Paper Summary

This section summarizes the research papers mentioned earlier in brief:

6.1 Research Paper 1

Logarithmic Market Scoring Rules for Modular Combinatorial Information Aggregation[10] *The Journal of Prediction Markets, 2002*
Author *Robin Hanson*

This paper covers,

- Scoring rules
- Market Scoring rules
- Logarithmic Market Scoring Rule(LMSR)
- Distribution *equals* Cost function
- Combinatorial Markets

Preliminaries

1. *Event:* The set $\omega = \{w_1, \ldots, w_n\}$ is an event comprising of a finite set of n possible outcomes which are mutually exhaustive and exclusive.

2. *Fair-Bet:* A fair bet is one for which the expected value of the payoff is zero, after accounting for the cost of the bet. For example, suppose A offers to pay you $2 if a fair coin lands heads, but you must ante up $1 to play. Your expected payoff is -$1+ $0xP(tails) + $2xP(heads) = -$1 + $2x50% =$0. This is a fair bet-in the long run, if you made this bet over and over again, you would expect to break even. The fair bet eq is given by,

$$\sum_i y_i m_i(\bar{x}) = 0.$$

3. *Scoring rule:* Let, p_i be agent's belief about the probability that w_i will occur and $S_i(\vec{r})$ be the payment made to agent i if he reports distribution $\vec{r}$ if outcome is w_i, then a *proper scoring rule* is,

$$\vec{p} = \text{argmax}_{\vec{r}} \sum_i p_i S_i(\vec{r}).$$

A scoring rule is called *strictly* proper when $\vec{p}$ is unique.

4. *Logarithmic Scoring rule:* Given by,

$$S_i(\vec{r}) = a_i + blog(r_i),$$

such that $r_i \leq 1$ and a_i is big enough for agents to be willing to participate. This satisfies a lot of desirable properties; for example, this is the only rule in which an agent's payoff depends only on the probability he assigned to the actual event. Thus, LSR is *local* i.e., we can look at a n-valued event as n binary events and the score remains the same.

5. *LSR is proper:* For binary outcomes, an agent's expected payment would be,

$$p(a + blog(r)) + (1 - p)(a + blog(1 - r)),$$

derivative w.r.t. r would give,

$$p - rp - r + rp = 0,$$

which implies $p = r$. Further, the second derivative is negative, thus LSR is proper.

What is the problem addressed?

The paper introduces a rule wherein multiple agents can share their beliefs without the need to pay each one, and which generates a single (unified) prediction.

In addition, it has long been known that in one dimension, allowing an agent to interact with a proper scoring rules is equivalent to allowing him to choose a quantity from a continuous offer demand schedule. This paper shows how this equivalence also holds in higher dimensions.

If we use simple scoring rules we will not get a unified estimate, while simple markets do not incentivize agent contributions. Market scoring rules give incentive to agents but do not require them to find a counterparty. Thus, for a scenario where a single agent knows something about an event and the rest know that they know nothing about the event, then, these markets cannot acquire the agent's information.

What is the solution proposed?

The paper proposed a logarithmic market scoring rule. A market scoring rule is one which,

1. Starts with an initial distribution ($\vec{r_0}$), and at each step an agent reports his distribution.

2. The agent pays to the previous agent according to a scoring rule S,

$$x_i = \Delta S_i(\vec{r_t}, \vec{r_{t+1}}) = S_i(\vec{r_t}) - S_i(\vec{r_{t+1}}).$$

3. The market maker finally pays $\Delta S_i(\vec{r_t}, \vec{r_0})$.

Market makers are central actors which have public offers to buy or to sell, and update these prices in response to trades. A market scoring rule, acts like a continuous automatic market maker with which an arbitrary number of agents can have an arbitrary number of interactions, at no additional cost over that of the one last interaction. A market scoring rule gives incentive to participate and share your knowledge, increases liquidity and is easier to expand to multiple outcomes. If only one agent participates it is equivalent to a simple scoring rule. If many agents participate, it gives the same effect of a standard information market, at the cost of the payment to the last agent.

The paper proposes a logarithmic market scoring rule (LMSR), wherein $S_i(\vec{r}) = a_i + blog(r_i)$, where b is a parameter which controls liquidity and the loss of the market maker. A large b allows a trader to buy many shares at the current price without effecting the price drastically. Here, market maker's expected loss is the entropy of the initial distribution he gives, π_i, $-b\sum_i \pi_i log(\pi_i)$. If the initial distribution is uniform, then the market maker's loss is bounded by $nlogn$, where n is the number of possible outcomes.

Implementing a Market

We can think of an agent's distribution as being equal to his buying or selling of shares, and the market maker scoring rule as one which has a list of outstanding shares ($q = \{q_i\}$), an instantaneous price $\vec{p}$ and one which will accept any fair bet. The market maker's task is to extract information implicit in the trades others make with it, in order to infer a new rational price.

Logarithmic Market Scoring Rule: Let i be the outcome which finally took place. Then, the market maker pays exactly q_i\$. This payment should be equal to the payment using LSR if distribution is equal to buying or selling as stated earlier. Thus, we want a price function such that $S_i(p_i) = q_i$. Here, the price function p_i is the current distribution given by the last agent and is the inverse of the scoring rule function.

For LMSR, it's price function for an outcome i would be,

$$p_i(q) = \frac{e^{(q_i - a_i)/b}}{\sum_k e^{(q_k - a_k)/b}}.$$

This is for infinitesimal small number of shares. To find the total cost of buying some number of shares, we use the cost function

$$C(\vec{q}) = blog \sum_i e^{(q_i/b)}.$$

Logarithmic rule is MSR:

1. Let $C(\bar{q}) = log \sum_\omega e^{q_\omega}$
2. Then the current price for an ω security $p_\omega(\bar{q}) = \frac{e^{q_\omega}}{\sum_{\omega'} e^{q_{\omega'}}}$
3. An agent believes the current price should be $\bar{p}'$, and reports $\bar{q}'$, such that $\sum_{\omega'} e^{q'_{\omega'}} = k$.
4. Thus, $p'_\omega = e^{q'_\omega}/k$ or $q'_\omega = log(kp'_\omega)$.
5. The agent will pay $\sum_{\omega'} e^{q'_{\omega'}} = log(k)$ (and some constant).
6. For the realized outcome ω the agent receives $q'_\omega = log(kp'_\omega)$ (and some constant).
7. Therefore we get $log(kp'_\omega) - log(k) = log(p'_{\omega'})$ (and some constant).

LMSR is local:

Theorem: Logarithmic rule bets on A given B preserve $p(B)$, and for any event C preserve $p(C|AB)$, $p(C|A'B)$ and $p(C|B')$ and thus preserves $\mathcal{I}(\mathcal{A}, \mathcal{B}, \mathcal{C})$ and $\mathcal{I}(\mathcal{C}, \mathcal{B}, \mathcal{A})$.

That is, the market maker's inference rule assumes that a new bet on A given B in general gives it new information only about how the event A might depend on the event B, but no information on the probability of B, or on events unrelated to how A might depend on B. For such unrelated events, previous independence relations should be preserved.

When there are at least three events i, the inverse also holds; the logarithmic market scoring rule is the only local rule, even in the weak sense that a bet on A given B preserves $p(B)$. Therefore, we get the following,

Theorem: For $I \geq 3$, if $y_i = 0$ for $i \notin \{j, k\}$ implies $q_i = 0$ for $i \notin \{j, k\}$, the rule is logarithmic.

6.2 Research Paper 2

Crowdfunding Public Projects with Provision Point: A Prediction Market Approach[5] *ECAI 2016*

Authors: Praphul Chandra and Sujit Gujar and Y. Narahari

What is the problem addressed?

Crowdfunding is emerging as a popular means to generate funding from citizens for public projects. The paper focuses on projects with provision point i.e., projects in which contributions must reach a predetermined threshold in order for the project to be provisioned. It proposes a mechanism that improves the success of such public projects. However, the success of such projects have been mixed. Some problems in this setting include: free riding, getting the agents to elicit their true valuation etc. The mechanisms available for such projects do not take into account the sequential nature of the agents contributions and these contributions are public.

Use Cases: Governments ask for donations for everything from dog parks to public defenders, or for crowdfunding projects can be started for areas where the government has given budget cuts (public-sector). Other cases can be start ups, medical donations etc (private-sector).

How does the work compare with previous work?

This paper focuses on discrete public projects, where agents arrive sequentially. However, earlier work restricts the agents to contribute in a fixed order (Marx and Matthew, 2000) and is not sub-game perfect equilibrium. In PPS, there is neither a pre-fixed order of contributions nor do agents contribute repeatedly and we look for sub-game perfect equilibria.

In (PPM, 1989) a project is provisioned if the funding target is met, otherwise the contributions are refunded. Hence the utility of an agent if the project is not refunded is 0. In (PPR, 2014) if the project is not provisioned, the contributions are refunded and an additional refund bonus is paid to the agents based on their contributions. In this agent's decide on their contributions without knowing other agent's contributions. Hence, this results in a simultaneous move game and the agent's can decide to not contribute until the deadline of the project. Thus, PPR is limited in sequential setting.

What is the solution proposed?

The paper proposes a protocol in which if the project is not provisioned, the contributing agents not only get their contribution refunded, but get some refund bonus (depending on their contribution) on top of it. The refunds are calculated through a prediction market approach. In addition, the protocol also takes into account the sequential nature of the contributions in a way that agents who contribute early gain higher utility and thus overcomes the serious limitation of PPR. The major contributions include:

- The paper proposes *Provision Point Mechanism with Securities* (PPS), showing that this always achieves an equilibrium in which the project always gets funded.
- Solves the free-riding problem since the agents are incentivized to contribute as soon as they arrive.
- PPS uses a complex prediction market (since the securities are awarded only when the project is not funded), and hence various versions of the protocol are possible depending on the prediction market.

Provision Point Mechanism with Securities (PPS)

PPS is a prediction market in which securities are associated with the binary outcome of the project getting funded or not i.e., in PPS $\Omega = \{\omega_0, \omega_1\}$. Every contribution x_i, at time t, towards the public project is treated as an investment in purchasing $r_i^{t_i}$ number of securities associated with the outcome ω_0. If the project is not funded, each of these securities pay out an unit amount and 0 otherwise. Thus, an agent i's utility is,

$$u_i(\psi; \theta_i) = \mathcal{I}_{\mathcal{X} \geq h^0} \times (\theta_i - x_i) + \mathcal{I}_{\mathcal{X} < h^0} \times (r_i^{t_i} - x_i),$$

and $r_i^{t_i}$ is a monotonically decreasing function of q^{t_i} which is why agents are incentivised to contribute early. The cost function needed to determine this needs to be *complex*. In a complex cost based cost function, the market maker may offer securities $< |\Omega|$ or the function may not pay a unit amount if an outcome is satisfied. In the complex cost function proposed, the securities associated with the negative event pays out, while that of the positive event does not. Also, an agent is not allowed to sell his securities. It can be shown that this cost function (C_0) is invertible.

Equilibrium in PPS

1. At equilibrium, $\mathcal{X} = h_0$.
2. At equilibrium, an agent will only contribute if his funded utility is at least as much as his unfunded utility i.e., $(r_i^* - x_i^*) \leq (\theta_i - x_i^*)$ or $r_i \leq \theta_i$.
3. An agent will contribute as soon as he arrives, thus his strategy at eq would be $\psi^* = (x_i^*, a_i)$.
4. Summing up r_i^8 from 2 gives the condition for Nash Equilibrium as $C_0^{-1}(h^0 + C_0(0)) < v$.

Individual Rationality in PPS PPS, would be individually rational if,

$$u_i(\psi_i, \psi_{-i}, \theta_i) \geq 0 \, \forall \psi_{-i},$$

$$u_i(\psi; \theta_i) = \mathcal{I}_{\mathcal{X} \geq h^0} \times (\theta_i - x_i) + \mathcal{I}_{\mathcal{X} < h^0} \times (r_i^{t_i} - x_i) \geq 0,$$

If w_0 does not occur, i.e., $\mathcal{X} \geq h^0$, $u_i(\psi; \theta_i) \geq 0$,
If w_0 does occur, i.e., $\mathcal{X} < h^0$, for every t_i,

- LMSR-PPS: satisfies condition 7 ($\frac{\partial r_i^{t_i}}{\partial x_i} > 1$), always. Therefore, $r_i^{t_i} - x_i > 0$ always, hence LMSR-PPS is IR.
- QSR-PPS: satisfies condition 7 if $b > \frac{2}{3} h_0$, thus for every such b, QSR-PPS is IR.

What does paper claim?

The paper claims to have proposed a provision point mechanisms, PPS for civic crowdfunding. PPS induces an extensive form game among the agents who arrive on the crowdfunding platform and achieves equilibria at which the project is funded. These equilibria have the desirable property that agents do not free ride but instead contribute in proportion to their true value for the project and do so as soon as they arrive.

6.3 Research Paper 3

Pricing Combinatorial Markets for Tournaments[6] *STOC 2008*
Authors: YILING CHEN, SHARAD GOEL AND DAVID PENNOCK

What is the problem addressed?

It has already been shown combinatorial markets offer a wider array of information aggregation possibilities than traditional prediction markets. This can done through LMSR market maker, but keeping the requisite distributions around is computationally hard.

This paper looks at a restricted case for this, proving it's feasibility.

Preliminaries

- *Complexity:* #P is the class of counting problems associated to decision problems in NP. For example, 2-SAT problem implies is there an assignment that satisfies such a formula, while #2-SAT implies how many such assignments exist satisfying this formula. Naturally, NP is contained in #P, since we can just count the solutions and whether the count is greater than 0 or not, to model the counting problem to a decision problem. Interestingly, 2-SAT is in P, while #2-SAT is in #P.
- *Bayesian Network:* Is a data structure to represent probability distributions $P(X_1, \ldots, X_n)$ especially for cases where dependencies in the distribution are sparse. This is represented as Directed acyclic graph (DAG), where each vertex correspond to a random variable and edges represent conditional (in)dependence. An important property of such a network is

  $$P(X_1, \ldots, X_n) = \prod P(X_i | parents(X_i)),$$

  i.e., each node X_i consists of a probability conditioned only on it's parents.

  In general, calculating $P(X_i = x_i)$ from this is NP-Complete. However, we can do better for certain topologies. In this paper, such marginal and other topologies can be calculated in $O(n)$.
- *Tournaments:* The paper uses an elimination tournament with n teams and $n-1$ games. This is arranges into a tree structure and has 2^{n-1} possible outcomes.

Computational Complexity

The pricing problem can be modeled as a #2-SAT as follows:

1. Let ϕ be a boolean formula with $S_\phi = \{\omega \text{ satisfies } \phi\}$.
2. Let c be the cost of purchasing one share of S_ϕ.
3. Model a pricing problem such that agents can place bets on S_ϕ, where $\phi = c_1 \wedge c_2 \wedge \cdots \wedge c_r$, where each c_i is a disjunction of two non-negated literal.
4. Then the pricing problem restricted to this language is #P-Hard, since #2-SAT is #P-Complete.

Betting Languages

Due to the computational complexity of the problem we model the pricing problem in a restricted setting by allowing bets of only a certain type. This paper discusses the following two betting languages,

1. *Betting Language 1:* This allows bets of the form "Team i will win game k", "Team i will win game k given that they make it to that game" and "Team i beats team j given that they face off". Such a language can be modeled as a Bayesian network arranged like a tournament tree, with each nodes representing the outcome of a game and edges directed opposite to their causality. For this setting the paper proposes the following:
   - *Theorem 1:* The distribution $\tilde{P}$ that results from executing the order $O = (g_j, t_j, \Delta b)$, interpreted as buying Δb shares of the outcome in which team g_j wins game t_j, is also represented by a Bayesian network with the same structure, and only the distributions of g_j and its ancestors are affected.
   - *Theorem 2:* The distribution $\tilde{P}$ that results from executing the order $O = (g_{j_1}, g_{j_2}, t_{j_1}, t_{j_2}, \Delta b)$, interpreted as buying Δb shares of the outcome in which team t_{j_1} wins game g_{j_1} where g_{j_1} is the parent of g_{j_2}, is also represented by a Bayesian network with the same structure, and only the distributions of g_{j_2} and its ancestors are affected.
   - To preserve the Bayesian network for the bet "Team i will win game k given that they make it to that game", denoted as $A|B$, agents buy AB and short sell $\bar{A}B$. The Bayesian network is preserved for AB through the above theorem, while for $\bar{A}B$, trade in the sequence $A_1B, \ldots, A_{i-1}B, A_{i+1}B, \ldots, A_nB,$. Each trade in the sequence preserves the network through the above theorem.
   - To preserve the Bayesian network for the bet "Team i beats team j given that they face off", there will always be a unique game k in which they could face off. Let $A = \{X_k = i\}$ and $B = \{X_{j_1} = i, X_{j_2} = j\}$ where X_{j_1} and X_{j_2} are the children of X_k. Now, $AB = \{X_k = i, X_{j_2} = j\}$ and $\bar{A}B = \{X_k = j, X_{j_2} = i\}$. Theorem 2 allows us to trade in both of these joint events, and these can consequently construct the conditional asset.

   The paper then shows that for this betting language, update any node in this network will require to update $O(n^2)$ parameters. Also, it has also been shown that each such update can be done in linear time for such a topology. Thus, the total complexity becomes $O(n^3)$ i.e., in polynomial time.

This language an however lead to unexpected dependencies in the market-derived distribution and hence we look at a further restricted case.

2. *Betting Language 2:* This language restrict the type of bet only to "Team i beats team j given that they face off". For this the network can be modeled similar to the previous case with the edges being directed to the direction of causality. This can be modeled in polynomial time without the dependency problem of the first betting language. Any update in the distribution preserves the Bayesian network as shown by the following theorem,

   Theorem 3: The distribution $\tilde{P}$ that results from executing the order $O = (g_j, t_j, t'_j, \Delta b)$, interpreted as buying Δb shares of the outcome in which team t_j wins game g_j conditioned on the probability that t_j and t'_j play game g_j, is also represented by a Bayesian network with the same structure, and only the distributions of g_j.

   Since each trade in this language requires updating only a single parameter of the Bayesian network, and since that update can be performed in O(n) steps, an update phase is only $O(n)$.

Working Example

Consider the following example of a tournament with 4 teams $\{1, 2, 3, 4\}$ and games $\{X_1, X_2, X_3\}$ where X_1 is final game. *At the start there are no outstanding shares,* and the outcome space is as follows,

$$\omega_1 = (1,3,1) \quad \omega_2 = (1,3,3) \quad \omega_3 = (1,4,1) \quad \omega_4 = (1,4,4)$$

$$\omega_5 = (2,3,2) \quad \omega_6 = (2,3,3) \quad \omega_7 = (2,4,2) \quad \omega_8 = (2,4,4)$$

. Now, domain of X, be represented by $D(X)$, then

$$D(X_1) = \{1,2,3,4\}; D(X_2) = \{1,2\}; D(X_3) = \{3,4\}.$$

Note: In this example, game 3 means the final game and so on.

- Δb shares on team 1 to win game 3: The outcomes ω_1 and ω_3 will get Δb shares while the rest of the shares remain the same. The change in the probability distribution ($\tilde{P}$) after this bet is as follows

$$\tilde{P}(X_3 = 3) = \tilde{P}(X_3 = 4) = \frac{e^{\Delta} + 3}{2e^{\Delta} + 6}$$

$$\tilde{P}(X_2 = 1) = \frac{2e^{\Delta} + 2}{2e^{\Delta} + 6}; \tilde{P}(X_2 = 2) = \frac{2}{2e^{\Delta} + 6}$$

$$\tilde{P}(X_1 = 1) = \frac{2e^{\Delta}}{2e^{\Delta} + 6};$$

$$\tilde{P}(X_3 = 2) = \tilde{P}(X_3 = 3) = \tilde{P}(X_3 = 4) = \frac{2}{2e^{\Delta} + 6}$$

  The probabilities must be updated for all 4 parameters. The change in the conditional probabilites are as follows

$$\tilde{P}(X_2 = 1|X_1 = 1) = 1; \tilde{P}(X_2 = 2|X_1 = 1) = 0.$$

  Similarly for $X_1 = 2$.

$$\tilde{P}(X_2 = 1|X_1 = 3) = \tilde{P}(X_2 = 2|X_1 = 3) = 1/2.$$

  Similarly for $X_1 = 4$.

$$\tilde{P}(X_3 = 3 \text{ or } 4|X_1 = 1) = \frac{e^{\Delta}}{2e^{\Delta}} = 1/2.$$

  Similarly for $X_1 = \{2, 3, 4\}$. All these conditional probabilities do not change as consistent with Theorem 1. For each $x_i \in D(X_1) = \{1, 2, 3, 4\}$, X_3 and X_2 will have to update at all $n/2$ probabilities, and hence we get that for each such node the set of parameters to be maintained are of the order $O(n^2)$.

- Δb shares on team 3 to win game 3 given that they make it to that game: The outcome space for team 3 reaching game 3 is ω_1, ω_2, ω_5 and ω_6 while the space for team 3 winning is ω_2 and ω_6 and hence these outcomes will get extra Δb shares. From theorem 2,

$$\tilde{P}(X_2 = 1|X_1 = 1) = 1; \tilde{P}(X_2 = 2|X_1 = 1) = 0$$

$$\tilde{P}(X_2 = 1|X_1 = 2) = 0; \tilde{P}(X_2 = 2|X_1 = 2) = 1$$

$$\tilde{P}(X_2 = 1 \text{ or } 2|X_1 = 3) = \frac{e^{\Delta}}{2e^{\Delta}} = 1/2.$$

  Again, the conditional probabilities will not change for any combination this time consistent with theorem 2 and since $|D(X_1)| = n$ and $|D(X_2)| = |D(X_3)| = n/2$, so update will require $O(n^2)$ for each node.

Working Example 2

Consider the same scenario as above, with betting language 2. Thus, we can only bet on the language "Team i beats team j given that they face off".

Note: For this language, the Bayesian network changes dependencies i.e., for the previous example, X_1 was the parent of X_2 and X_3 while in this X_2 and X_3 are

- Δb shares on Team 1 to beat Team 3 given that they face off : For this the outcome space will reduce to ω_1 and ω_2 and outcome ω_1 will get Δb shares. In this scenario, Theorem 3 shows that only the probability distribution of the node g_j (this is the game where Team i and j face off) will change. It is trivial to see that such a game will be unique in a tournament. The conditional probabilities now become,

$$\tilde{P}(X_1|X_2, X_3) = \tilde{P}(X_1|X_2)\tilde{P}(X_1|X_3)$$

  since X_2 and X_3 are independent. In addition, for team 1 and team 3 to face each other in X_1, $X_2 = 1$ and $X_3 = 3$. Now,

$$\tilde{P}(X_1 = 1|X_2, X_3) = \frac{e^{\Delta}}{e^{\Delta} + 1}$$

$$\tilde{P}(X_1 = 2|X_2, X_3) = 0$$

$$\tilde{P}(X_1 = 3|X_2, X_3) = \frac{1}{e^{\Delta} + 1}$$

$$\tilde{P}(X_1 = 4|X_2, X_3) = 0$$

  Only the distribution of X_1 will change, and unlike the previous case the node will only consider values in $D(X_1)$, so the update is of the order $O(n)$.

6.4 Research Paper 4

Incentives to Counter Bias in Human Computation [7] *Proceedings of the Second AAAI Conference on Human Computing HCOMP '14*
Authors: Boi Faltings & Pearl Pu & Bao Duy Tran

What is the problem addressed?

For problems that Machine Learning cannot yet solve, online labor markets are used. Such problems include assessing labeling images or subjective content. However for many tasks, especially subjective ones such as annotating an image with a set of keywords, the validity of the answer is hard to establish. This is problematic because workers have an incentive to minimize the effort they spend on the task, and tend to use heuristic problem solving strategies that involve little or no effort on the task itself. These heuristic strategies can be countered by increasing the number of workers. However, this is not possible when the heuristic strategy has a common bias that can be mistaken for the true value. These problems arises for many strategies such as:

1. When a worker has carried out several instances of a task that all had the same answer, and thus develops a bias towards that answer.
2. When there is a known most likely answer.
3. When a hint is provided by the task itself that the workers use as a basis for bias.

What is the solution proposed?

Increasing the number of workers does not work for such biases. Previous results use statistical techniques to eliminate this bias. However, it is clearly better to motivate workers to provide unbiased answers rather than eliminate them.

This paper proposed a scheme *Peer Truth Serum*, that provides bonus that lets the workers evaluate their answers from a game-theoretic perspective. In this, workers are motivated to reveal their truthful answers, since it results in a Nash Equilibrium whenever the workers prior is same as a publicly available distribution.

Preliminaries

1. The workers are incentivized through bonuses. Generally, incentives are decided by comparing them with the answers of an experienced human resource expert i.e., workers are rewarded for agreeing or punished for disagreeing. This has been shown to have a significant effect on accuracy.
2. However, to discourage the tendency to report common answers that are most likely to be matched by others, it is important to give higher rewards to surprising answers.
3. A worker chooses any one of the following strategies:
   - *cooperate:* invest effort atmost γ to find the true answer x and report it truthfully.
   - *deceive:* invest effort to find the true answer x but report a different answer y.
   - *heuristic:* not effort; report an answer as per a heuristic strategy.

Peer Truth Serum

1. *Proposition 1:* Provided that the payoff of the best cooperate strategy is more than γ higher than the best deceive strategy, no heuristic strategy can be optimal.
2. PTS pays a worker a bonus if whenever her answer matches another worker. The bonus is defined as:

   $$f(a_i, a_j, R) = \frac{c}{R(a_i)} \qquad \text{if } a_i = a_j,$$

   and 0 otherwise, where a_j is a selected answer of a peer, a_i is the worker's answer and R is publicly available distribution.
3. *Proposition 2:* Whenever the agents' prior belief $Pr(x)$ is equal to the publicly available distribution $R(x)$, the Peer Truth Serum makes truthful reporting a Nash Equilibrium.

Experimental Analysis

1. *Requirements:* The task must be tedious with a verifiable answer as well as must be possible to induce a bias in it.
2. *The Task:* Workers were asked to count the number of visible cameras/camcorders/ binoculars/phones in an image of a crowd.
3. *Different Schemes:*
   - No bonus: Consensus gave a mean error of 1.0667 without any bias. For a bias way off the actual answer, 60, the mean error was 5.6316, while for the bias with the correct answer, 34, the mean error was 2.9434. This is explained by the fact that anchoring perturbs answers from purely random errors. The answer distribution was shifted towards the prime value, with the peak not being at the prime itself.

     Result 1: Priming with an incorrect value decreases accuracy while degradation is not significant for priming with an accurate value.
   - Peer Truth Serum: The model was described to the workers with the following statement: "We will pay you $0.01, and a bonus if your response matches that of another worker on the

same image within +/- 1 tolerance. The bonus is as follows: (i) \$0.01 if the matching answer is within +/-1 of the current average count. (ii) \$0.06 if the matching answer is something different."

Result 2: The Peer Truth Serum counters priming better than having no bonus.

- Peer Confirmation: The statement for this model is "We will pay you \$0.01, and a bonus of \$0.04 if your response matches that of another worker on the same image within +/-1 tolerance."

  Result 3: The Peer Truth Serum corrects priming better than other bonus schemes.

4. The following table summarizes the results:

Bonus	Priming	Average Error	t-test
no bonus	none	1.0667	
	60	5.6316	p = 0.0266
	34	2.9434	p = 0.2092
vague	none	2.2500	
	60	6.6563	p = 0.0810
	34	9.0984	p = 0.0032
peer conf.	none	0.3492	
	60	3.3429	p = 0.0554
	34	2.4194	p = 0.1496
peer conf. PTS	none	0.3492	
	60	0.8000	p = 0.4036
	34	2.1667	p = 0.2145

Figure 3: Comparison of error between treatments without and with priming to an incorrect value (60) and a correct value (34).

6.5 Research Paper 5

A Bayesian truth serum for subjective data[11]
Authors: Drazen Prelec

What is the problem addressed?

For questions with *subjective truth* such as "will you buy the new iPhone X?" or *Will you vote for in the next Presidential Election?*, one can not have a definite answer. Therefore, tradition methods such as scoring rules can not be directly applied in such markets. Assigning, greater weightage to majority answer will also not always work - since it induces a bias, and a lazy agent will choose to simply choose the most popular answer rather than elicit his true preference.

What is the solution proposed?

This paper proposes a survey scoring method, called *Bayesian Truth Serum*, that provides truth-telling incentives for respondents answering multiple-choice questions. In this, respondents supply not only their own answers, but also percentage estimates of other's answers. The BTS formula then assigns high scores to answers that are *surprisingly common*.

Bayesian Truth Serum

The intuition behind this scheme is that if people truly hold a particular belief, they are more likely to think that others agree or have had similar experiences i.e., *you're your best estimator* and your estimation reveals you.

In this, each respondent is required to submit

1. an endorsement of an answer to an $m-$ multiple choice question $x_k^r \in \{0, 1\}$, which indicates whether respondent r has endorsed answer $k \in \{1, \dots, m\}$
2. a prediction $(y_1^r, \dots, y_m^r)$ of the sample distribution of endorsements

The BTS score is then defined relative to the reported sample averages i.e., by $\bar{x}_k$ which is the fraction of respondents endorsing answer k and $\bar{y}_k$ which is the geometric average of endorsement predictions for answer k. Then, the total BTS score for some respondent r, for endorsement $(x_1^r, \dots, x_m^r)$ and prediction $(y_1^r, \dots, y_m^r)$ is given by:

BTS score = Information Score + Prediction Score,

i.e.,

$$u^r = \sum_k x_k^r log\frac{\bar{x}_k}{\bar{y}_k} + \alpha \sum_k \bar{x}_k log\frac{y_k^r}{\bar{x}_k},$$

where α is a constant which fine tunes the weight given to the prediction error. The BTS score is symmetric, and zero sum if $\alpha = 1$. In this, the information score is the measurement for *surprisingly common* answers, while the prediction score measures the *prediction accuracy*.

With this BTS score, the paper proves the following theorems:
Theorem 1: Truth-telling is a Nash equilibrium for any $\alpha > 0$: Truth-telling maximizes expected total score of every respondent who believes that others are responding truthfully.
Theorem 2: Expected equilibrium information-scores are non-negative, and attain a maximum for all respondents in the truth-telling equilibrium.
Theorem 3: For $\alpha = 1$, the game is zero-sum.

Conclusion: The best strategy is for the respondent to tell the truth and that a respondent's preference will win to the extent that it is more popular than collectively estimated.

6.6 Research Paper 6

Constrained-Based Differential Privacy for Mobility Services[8] *AAMAS '18*
Authors: Ferdinando Fioretto, Chansoo Lee and Pascal Van Hentenryck

What is the problem addressed?

Mobile and wireless communications present everywhere, have the potential to revolutionize transportation systems by making accurate predictions and mobility traces. However, such data sets also pose security threats since these can reveal highly sensitive information of individual agents.

What is the solution proposed?

This paper uses *diffrential privacy* to release this data for transportation systems, showing that existing systems do not provide the desired fidelity. It proposes the idea of *Constraint Based Differential Privacy* (CBDP), which formulates the redistribution of private data as an optimization problem. It's other properties include,

- It is a constant factor away from optimality
- If the constraints capture categorical features, it runs in polynomial time

The paper also presents a case study of a On-Demand Multimodal Transit System with CBDP.

Preliminaries

The following tools are required for the design of CBDP,

- *Differential Privacy:* Intuitively, it is the privacy loss that results to individuals when their private information is used in the creation of a data product. We say, a randomized mechanism $\mathcal{M} : \mathcal{D} \to \mathcal{R}$, is a ϵ-differential private mechanism if for any $S \subset \mathcal{R}$ and any two inputs $D_1, D_2 \in \mathcal{D}$, differing in at most 1 data item, we have

  $$Pr[\mathcal{M}(D_1) \in S] \leq exp(\epsilon)Pr[\mathcal{D} \in S],$$

  where the probability is calculated over the coin tosses of $\mathcal{M}$. The parameter ϵ is the privacy budget of the mechanism.

  In addition, for CBDP, we will require the $\mathcal{M}$ to satisfy the following theorems,

  Theorem 1 *Composition Theorem Let $\mathcal{M}_i : \mathcal{D} \to \mathcal{R}_i$ be an ϵ_i-differential private mechanism. Their composition $\mathcal{M}(D) = (\mathcal{M}_1(D), \ldots, \mathcal{M}_k(D))$ is $\sum_{i=1}^{k} \epsilon_i$-differentially private.*

  Theorem 2 *Post Processing Immunity Let $\mathcal{M}_i : \mathcal{D} \to \mathcal{R}_i$ be an ϵ_i-differential private mechanism and $g : \mathcal{R} \to \mathcal{R}'$ be an arbitrary mapping. The mechanism $g \cdot \mathcal{M}$ is a ϵ-differential private mechanism.*

  In private data analysis, agents interact with the data set through *queries*. A *numeric* query is a function from a data set $D \in \mathcal{D}$ to a result set $\mathcal{R} \in \mathbf{R}^n$. The *sensitivity* of a query is the maximum of $||Q(D_1) - Q(D_2)||$ over every D_1 and D_2, and is denoted by Δ_Q. For example, *counting* queries, which return the number of data points satisfying a predicate, have sensitivity 1.

  Theorem 3 *Laplace Mechanism Let $Q : \mathcal{D} \to \mathcal{R}$ be a numerical query. The Laplace mechanism defined as $M_{Lap}(D; Q, \epsilon) = Q(D)+z$ where $z \in \mathcal{R}$ is a vector of i.i.d. samples drawn from the Laplace distribution with scaling factor Δ_f/ϵ, is a ϵ-differential private mechanism.*

- *On-Demand Multimodal Transit System:* On-Demand multimodal transit systems (ODMTS) jointly address congestion and the first/last mile problem that plagues many transit systems. They combine high-frequency buses (or light-rail) between hubs for high-density corridors, with on-demand shuttles to bring riders from their origins to the hubs and from the hubs to their destinations. On-demand shuttles also perform direct trips between origins and destinations. Bus routes are fixed, while the shuttles are dispatched and routed dynamically, maximizing ride sharing while preserving short waiting and transit times for passengers. Riders are picked up and dropped off at virtual vehicle stops (also called locations for simplicity), which include the hubs.

- *The Differential Privacy Challenge:* For previous differential privacy study on OBMTS, the Laplace mechanism on batched query outputs a real vector with many negative numbers, which are semantically meaningless. These are rounded off to its closest non-negative number, which induced a significant bias towards positive values.

- *Notations:*

Data	Notation
Set of Non-negative integers	$\mathbf{N}_0$
Data universe	$\mathcal{U}$
Multi set of elements in $\mathcal{U}$	D
Set of Locations	L
Time period	T
$\mathcal{U}$ definition	$\mathcal{U} = L \times L \times T$
Origin and Destination	o and d
Elements in $\mathcal{U}$	$u = (o, d, t)$

The CBDP Mechanism

1. *Feature and Feature Queries:* A feature is the partition of the universe $\mathcal{U}$ and the size of the feature is the number of elements in the feature. A feature $\mathbf{F}'$ is a sub-feature of $\mathbf{F}$, denoted by $\mathbf{F}' \prec \mathbf{F}$.

2. *The Input:* The mechanism takes as input a data set D and a collection of features $\mathcal{F} = \{\mathbf{F}_1, \ldots, \mathbf{F}_k\}$ where $\mathbf{F}_i = \{d_{i1}, \ldots, d_{in_i}\} \quad \forall i$. The paper also assumes that the first feature is the set of singletons i.e., $\mathbf{F}_1 = \{\{u\} : u \in \mathcal{U}\}$ and the second feature consists of the entire universe i.e., $\mathbf{F}_2 = \{\mathcal{U}\}$.

3. *The CBDF Mechanism:* The mechanism first applies the Laplace mechanism with privacy parameter $\frac{\epsilon}{k}$ to each feature query. The resulting counts are then post-processed by the following optimization algorithm,

$$
\begin{aligned}
&\textbf{minimize: } \|\mathbf{x} - \tilde{\mathbf{c}}\|^2_{2,\mathbf{w}} = \sum_{i=1}^{k} \frac{1}{n_i} \sum_{j=1}^{n_i} (x_{ij} - \tilde{c}_{ij})^2 && \text{(O1)}\\
&\textbf{subject to:} && \\
&\qquad \forall i', i : \mathbf{F}_{i'} \prec \mathbf{F}_i, j \in [n_i] : x_{ij} = \sum_{l: \mathbf{d}_{i'l} \subseteq \mathbf{d}_{ij}} x_{i'l} && \text{(O2)}\\
&\qquad \forall i, j : x_{ij} \geq 0. && \text{(O3)}
\end{aligned}
$$

Figure 4: The CBDP Post-Processing Step.

4. Intuitively, the CBDP algorithm can be thought as redistributing the noise introduced by the Laplace mechanism to obtain a consistent data set. The post-processing step searches for a feasible solution that satisfies all the feature constraints and is as close as possible to the Laplacian counts.

5. Some observations,

   (a) By using a weighted L2-norm, CBDP ensures that the sums of the terms for each partition are of the same order of magnitude.

   (b) The features are not hierachical: They can capture fundamentally different aspects of the problem structure.

   (c) The non-negativity constraints ensure that only non-nengative post-processed counts are generated, contrary to the Laplacian mechanism.

Theoretical Properties of CBDP

Theorem 4 *CBDP achieves ϵ-differential privacy.*

Proof: Since each feature partitions the universe, each feature query is a counting query with sensitivity 1. Thus each count is ϵ/k-differential private by Theorem 3. The combination of each count is ϵ-differential private by Theorem 1. The result follows from Theorem 2.

Theorem 5 *A $\delta-$ solution to the optimization model from Figure 1 can be obtained in time polynomial in the size of the universe, the number of features, and $1/\delta$.*

Proof: Since the number of variables and constraints in the optimization model are bounded by a polynomial in the size of the universe and the number of features, and that the optimization model is convex.

6.7 Research Paper 7

Crowdsourced Outcome Determination in Prediction Markets [9] *AAAI 2017*
Authors: Rupert Freeman, Sebastien Lahaie and David M. Pennock

What is the problem addressed?

Prediction markets have become a popular tool for information aggregation. However, a key challenge in implementing and scaling prediction markets is the question of outcome determination (i.e., closing markets for events). Since a centralized prediction market, limits the scope of what can be predicted in the hands of the market maker, this paper looks at decentralized prediction markets. In this the markets are closed by consensus among a group of arbiters rather than by a center. Some of the problem in such a market include:

1. *Outcome ambiguity:* At the time the market closes, it might be unreasonable to assign a binary value to the event outcome due to lack of clarity in the outcome.

2. *Peer Prediction:* For the credibility of the market, it is essential that arbiters are incentivized to truthfully report their opinion as to the realized outcome.

3. *Conflict of Interest:* Even if arbiters can be incentivized to report truthfully in isolation, there is no way to prevent them also having a stake in the market.

What is the solution proposed?

The paper proposes a market mechanism which tackles the above mentioned challenges as:

1. *Outcome ambiguity:* Outcomes in this mechanism are determined by the fraction of arbiters that report an event to have occurred. This also guarantees that every market is well-defined, by having traders explicitly trade on their expectations of the arbiter reports, not the actual event.

2. *Peer Prediction:* Incentives are distributed through an existing peer prediction mechanism, the 1/prior mechanism, with a technical change.

3. *Conflict of Interest:* The paper shows that in this mechanism, as long as each agent is responsible for a limited fraction of trading, and questions are clear enough, realistic trading fees can fully subsidize truthful reporting on the part of the arbiters.

Preliminaries

1. *Notation:* Let N be the set of agents, $A \subset N$ be a small set of distinct and verifiable arbiters with $m = |A|$. Let X be a binary event of some realized outcome in $\{0, 1\}$.

2. *Prediction Markets:* This paper considers a prediction market via a Market Scoring Rule, where the underlying scoring rule is strictly proper. Under this market structure, agents trade shares of a security with a centralized market maker, who commits to quoting a buy and sell price for the security at any time.

3. *Peer Prediction:* Peer prediction mechanisms are designed to truthfully elicit private information from a pool of agents via a reward structure that takes advantage of information correlation across agents. After the realization of X, each arbiter receives either a positive or negative signal x_i, denoted as $x_i = 1$ and $x_i = 0$, respectively. Now, let μ be the probability that an agent receives a positive signal. Let u_1 (resp. μ_0) be the probability that given i receives a positive (resp. negative) signal, another agent also receives the same signal. These probabilities are specific to the nature of the event X.

   This paper uses the 1/prior mechanism in which every agent is first asked their signal report $\hat{x}_i$. Then every agent i is randomly paired with an agent $j(j \neq i)$ and paid,

$$u(\hat{x}_i, \hat{x}_j) = \begin{cases} k\mu & \text{if } \hat{x}_i = \hat{x}_j = 0 \\ k(1-\mu) & \text{if } \hat{x}_i = \hat{x}_j = 1 \\ 0 & \text{if } \hat{x}_i \neq \hat{x}_j \end{cases}$$

   where k is some constant that can be freely adjusted. In this, truthfully reporting $\hat{x}_i = x_i$ is an equilibrium provided that $\mu_1 \geq \mu \geq \mu_0$.

The Mechanism

The following describes the mechanism,

1. *Market Stage*
   (a) A prediction market is set up for an event X using a market scoring rule.
   (b) Agents trade in the market. For a security bought at price p, a trading fee of fp is charged, and for a security sold at price p, a fee of $f(1-p)$ is charged.
   (c) The market closes, trading stops.

2. *Arbitration Stage*
   (a) Each arbiter i receives a signal $x_i \in \{0, 1\}$ and reports an outcome $\hat{x}_i \in \{0, 1\}$.
   (b) Each arbiter i is assigned a peer arbiter $j \neq i$ and paid according to the 1/prior with midpoint mechanism.
   (c) The outcome of the market, and the payoff of each share sold, is set to the fraction of arbiters that report $\hat{x}_i = 1$.

Mechanism Analysis

Lemma 6 *Let n_i be the number of securities held by arbiter i. Then truthfully reporting $\hat{x}_i = x_i$, is a best response for arbiter i given that all other arbiters report truthfully, if and only if*

$$k \geq \frac{2|n_i|}{m\delta}.$$

This characterization requires an upper bound on the number of securities that any single agent owns. Therefore, the paper looks at it in terms of size of the fee f, and the amount of money that any single arbiter spends in the market, B. For a fixed fee, let q^- and q^+ be the number of the two outstanding secu+rities. Then,

Lemma 7 *For fixed percentage fee f, the number of outstanding securities lies in the interval $[q^-, q^+]$.*

Lemma 2 gives the minimum and maximum number of outstanding securities at any time. Now, let $\phi_b^-(B) = C_b^-(B + C_b(q^+)) - q^+$ (similarly ϕ_b^+). Then,

Corollary 7.1 *At the end of the market stage, an agent i with budget B holds $n_i \in [\phi_b^-(B), \phi_b^+(B)]$.*

Also,

Corollary 7.2 *For an agent that spends at most B dollars in a market with trading fee f and infinite liquidity, n_i lies in the range $\left[\frac{-B(1+f)}{f}, \frac{B(1+f)}{f}\right]$.*

We can combine Corollaries 2.1 and 2.2 with Lemma 1 to determine the minimum payment that guarantees truthful reporting in the arbitration phase. This gives us **Theorem 1**.

Theorem 2 gives bound which the mechanism should generate in order to pay the arbiter without any sponsor as,

Theorem 8 *The mechanism generates enough fee revenue to pay the arbiters without requiring any outside subsidy if*

$$fM \geq \frac{2max\{|\phi_b^-(B)|, |\phi_b^+(B)|\}}{\delta}$$

.

Appendix

PPX

PPX is an extension of PPS [5], which is a provision point mechanism for *crowdfunding*. PPS assumes no prior beliefs for an agent with respect to the project being provisioned or not, and hence is envisioned as a more general framework than PPS. PPX can handle biases in believes about the project getting provisioned.

PPS assumes that each agent has belief that with exactly 1/2 the project will be provisioned, while in real life this may be different. PPX incorporates these beliefs by categorizing agents into two types : those that believe that the project will be provisioned (with some probability $1/2 + \epsilon$) as *positive belief* agents, and those that do not believe that the project will be provisioned (with some probability $1/2 + \epsilon$) as *negative belief* agents. The analysis for such a mechanism is challenging.

References

[1] https://www.marketsandmarkets.com/PressReleases/predictive-analytics.asp.

[2] https://medium.com/byteball/making-p2p-great-again-episode-iii-prediction-markets-f40d49c0abab.

[3] https://images.google.com.

[4] https://www.quora.com/How-do-I-prove-Nash-equilibrium.

[5] P. Chandra, S. Gujar, and Y. Narahari. Crowdfunding public projects with provision point: A prediction market approach. In *ECAI*, pages 778–786, 2016.

[6] Y. Chen, S. Goel, and D. M. Pennock. Pricing combinatorial markets for tournaments. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 305–314. ACM, 2008.

[7] B. Faltings, R. Jurca, P. Pu, and B. D. Tran. Incentives to counter bias in human computation. In *Second AAAI conference on human computation and crowdsourcing*, 2014.

[8] F. Fioretto, C. Lee, and P. Van Hentenryck. Constrained-based differential privacy for mobility services.

[9] R. Freeman, S. Lahaie, and D. M. Pennock. Crowdsourced outcome determination in prediction markets. In *AAAI*, pages 523–529, 2017.

[10] R. Hanson. Logarithmic markets coring rules for modular combinatorial information aggregation. *The Journal of Prediction Markets*, 1(1):3–15, 2012.

[11] D. Prelec. A bayesian truth serum for subjective data. *science*, 306(5695):462–466, 2004.