FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

A Simplified Prediction Market Interface via Implicit Kelly Bets

BACHELORARBEIT

zur Erlangung des akademischen Grades

Bachelor of Science

im Rahmen des Studiums

Medieninformatik und Visual Computing

eingereicht von

Florian Klampfer
Matrikelnummer 1058208

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Ao.Univ.-Prof. Mag.rer.nat. Dipl.-Ing. Dr.techn. Rudolf Freund

Wien, 2. Jänner 2016

Florian Klampfer | Rudolf Freund

FAKULTÄT FÜR !NFORMATIK

Faculty of Informatics

A Simplified Prediction Market Interface via Implicit Kelly Bets

BACHELOR'S THESIS

submitted in partial fulfillment of the requirements for the degree of

Bachelor of Science

in

Media Informatics and Visual Computing

by

Florian Klampfer
Registration Number 1058208

to the Faculty of Informatics

at the TU Wien

Advisor: Ao.Univ.-Prof. Mag.rer.nat. Dipl.-Ing. Dr.techn. Rudolf Freund

Vienna, 2nd January, 2016

Florian Klampfer   Rudolf Freund

Erklärung zur Verfassung der Arbeit

Florian Klampfer
Weiherweg 2A, 5164 Seeham

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 2. Jänner 2016

Florian Klampfer

Abstract

This thesis introduces an algorithm to make automated bets in scoring rule-based prediction markets. The algorithm only takes users' probability estimates as input and places bets that maximize their expected rate of growth. The algorithm takes into account the price adjustment of the scoring rule and modifies the bet size accordingly, which corresponds to a proper application of the Kelly criterion. The thesis introduces the above concepts in detail and provides a derivation of the algorithm specifically for the Logarithmic Market Scoring Rule. The algorithm enables users to interact with prediction markets without in-depth knowledge of the underlying mechanisms, without risk of ruin and (Kelly-) ideal outcomes when their subjective probability estimates match the "true" probabilities of the events in question.

Contents

CHAPTER 1

Introduction

1.1 Motivation

Prediction markets, or *information-aggregation markets*, have shown remarkable accuracy in predicting the outcome of future events, like the winners of elections or the outcome of sport events [1, 30, 2]. In theory, this is possible because they allow multiple participants to pool their information into a collective forecast. They do so by buying and selling *shares* in the outcome, which are redeemable for a certain amount of money in case the event occurs, but become worthless otherwise. Through this mechanism they are incentivized to bet according to their true believes, and bettors with better information are incentivized to correct the mistakes of others.

While many people can express a subjective probability estimate, fewer know how to trade in a market. Therefore, a more accessible user interface for prediction markets would only require of users to enter their subjective probability estimate, while an algorithm figures out an appropriate amount of shares to buy or sell. This would be especially useful in play-money prediction markets [20], where users have to worry less about giving control to an algorithm. However, the question remains what an "appropriate" amount to bet is.

1.2 Problem definition

The *Kelly criterion* [14] is a classic formula for devising optimal bet sizes when being faced with a series of bets. It has been successfully applied in a variety of betting scenarios [21, 22, 26]. Since trades in prediction markets are essentially bets, using the Kelly criterion is a natural choice. In fact, formal models of prediction market activity either explicitly [3] or implicitly [31] assume Kelly bettors.

However, in markets based on *market scoring rules* – essentially automated market makers that are popular for prediction market implementations – prices change in response to each trade [8, 9, 6]. This leads users of the classic Kelly formula to bet more than is *Kelly-ideal*. This can lead them to make outright unfavorable bets in certain scenarios. A proper application of the Kelly criterion to market scoring rules would anticipate the price movement of the trade itself and lower the bet size accordingly.

Using this *non-naive* application of the formula, the Kelly-optimal amount to bet can be determined based on users' subjective probability estimate, their current wealth and the current price of the asset.

1.3 Goal

The goal of the thesis is to provide an algorithm to determine the Kelly-optimal number of shares to buy in a market scoring rule-based prediction market. Specifically, a solution for the Logarithmic Market Scoring Rule (LMSR) is provided, which is the most popular scoring rule in practice. Readers should be able to derive solutions for other scoring rules using the process outlined in this thesis.

The thesis is structured as follows. Chapter 2 will introduce markets, and prediction markets in particular. Chapter 3 will introduce market scoring rules, a solution to the shortcomings of conventional markets with respect to information-aggregation. Chapter 4 will introduce the Kelly formula. In chapter 5 the Kelly formula will be applied to market scoring rules. The last chapter will discuss various limiting factors of the solution.

CHAPTER 2

Prediction Markets

It has been known since Hayek [11] that markets, as a side effect of trade, also aggregate information. When a certain good has found additional applications, or has become more scarce for other reasons, it will be reflected in its price.

While financial markets are generally designed to facilitate the exchange of goods, a market can also be designed for the purpose of uncovering information about an asset. These are known as prediction markets or information-aggregation markets. In these *virtual* markets, instead of physical goods, contracts are being traded that have a certain payoff if a particular event occurs. All contracts related to a specific event have a common maturity date, at which point it is determined which outcome occurred (typically by the host of the prediction market) and the payouts are being distributed accordingly [30, 28]. This thesis will deal with assets that are worth either $1 when a certain state is reached or $0 otherwise, which are referred to as Arrow-Debreu securities [5].

This chapter will provide some historical context, explain traditional market structures and discuss the theory behind prediction markets.

2.1 History

"Betting markets" have been used to bet on the outcome of presidential elections from as early as 1868 until around 1940 [18]. They resurfaced in 1988 with the introduction of the University of Iowa's Iowa Electronic Markets (IEM) [7], which sparked interest in the subject again.

2.1.1 Early betting markets

Despite never being considered fully legal, the volume traded on betting markets in the 19th and 20th century was relatively high compared to prediction markets today. At times they exceeded the volume traded on stock and bond markets. While the main activity occurred in New York's Wall Street area, there were markets in other major U.S. cities as well [18].

Historical prediction markets had astonishing accuracy. Only in the 1916 election, the candidate initially favored by the market did not win, although the odds were split even by the time the polls closed [18].

It is assumed that these markets disappeared with the introduction of scientific polling. Newspapers stopped reporting on prediction markets, which were considered unethical by the general public throughout the time. In response, regulators in the 1940s finally started to limit the activity of organized betting on elections [18].

2.1.2 IEM

With a special permission for educational purposes, the IEM opened in 1988 with a maximum of $500 traded per account. The goal of the IEM was to study the behavior of markets and traders. The market was designed as a futures market. The value of the traded assets depended on the outcome of future events. In the case of the IEM, it offered contracts for each presidential candidate of the same year, paying a fixed amount if that candidate won, but nothing otherwise. Although the trade volumes were orders of magnitude smaller than earlier betting markets, the prices showed remarkable accuracy again [7].

2.2 Market mechanisms

This chapter will summarize how common auction mechanisms work and introduce critical terminology related to markets and trading.

2.2.1 Call markets

Call markets, also known as k-double auctions, were popular before the widespread availability of telecommunication technology. While outdated, they form the basis for later market mechanisms.

Call markets allow trading only at predefined times, referred to as *market clearing* times, and they only accept *limit orders*. Limit orders are orders that inform the market operator that one is willing to buy (sell) a certain security at no more (no less) than a specified price. They can be represented as triples (φ_i, q_i, b_i), where φ_i is the security being traded, q_i is the desired quantity, where negative quantities denote sell orders, and b_i denotes the limit price [28]. In case of a buy order, the limit price is also called the *bid price* and in case of a sell order it is also called *ask price*. Buy and sell orders are also referred to as *bids* and *asks*, respectively.

Table 2.1 depicts 6 example orders. Assume the market clears after receiving these orders. It contains $n = 3$ buy orders and $m = 3$ sell orders. When sorting the orders in decreasing

Nr.	Buy	Sell
1	$(A,\ 1,\ \$0.50)$	
2		$(A,\ -1,\ \$0.50)$
3	$(A,\ 2,\ \$0.40)$	
4		$(A,\ -2,\ \$0.40)$
5	$(A,\ 3,\ \$0.30)$	
6		$(A,\ -3,\ \$0.30)$

Table 2.1: Example buy and sell order sequence for asset "A".

Nr.	Rank	Order
1	p_1	$(A,\ 1,\ \$0.50)$
2	p_2	$(A,\ -1,\ \$0.50)$
3	p_3	$(A,\ 2,\ \$0.40)$
4	p_4	$(A,\ -2,\ \$0.40)$
5	p_5	$(A,\ 3,\ \$0.30)$
6	p_6	$(A,\ -3,\ \$0.30)$

Table 2.2: The orders from Table 2.1 sorted by limit price.

order of limit price, as in table 2.2, the m-th highest price is referred to as p_m. In a call market, the clearing price is set to $p_{m+1} + k(p_m - p_{m+1})$ where $k = 0.5$ [28].

In the example, this results in a market clearing price of \$0.40. As a result, order 6 will be matched against order 1 and 3. Orders 2 and 5 remain untouched as they are above/below the market clearing price. Order 4 would meet the clearing price, but there are no buyers are left.

2.2.2 Continuous double auction

The Continuous Double Auction (CDA) is the most common market structure today. Notable examples of CDA markets include the NYSE and NASDAQ.

The CDA mechanism is similar to a call market, except that the market clears every time a new order is placed. The exchange keeps an *order book* containing all unmatched buy and sell orders. As soon as a buy (sell) order is received that has a higher (lower) limit price than the lowest (highest) sell (buy) order in the book, they are matched against each other. Otherwise the order is added to the order book, waiting for future orders to be matched against.

There is a price difference between the lowest ask and the highest bid price at all times (otherwise the CDA mechanism would facilitate a trade), which is known as the *bid-ask spread*. Ideally, the bid-ask spread is small. A market is said to be *liquid* when there are high quantities of both bids and asks in the order book, which typically coincides with a low bid-ask spread.

In this market setup, the *price* is usually referred to as the midpoint between the lowest bid and the highest ask price. It can also refer to the last price at which a transaction took place. In either case, the price can only change when orders enter or leave the order book. In other words, the price changes as a side effect of trading.

Looking at the orders from Table 2.1, it can be seen that using the CDA mechanism, order 1 is matched against order 2, as are orders 3 and 4, as well as orders 5 and 6. If the orders arrive in this order (assuming an empty order book) the market clears completely. It should be noted that when the orders arrive in a different order, other outcomes can be reached.

2.3 Theory

2.3.1 Marginal trader hypothesis

Intuitively, prediction markets work because traders would stand to profit from buying (selling) at prices that do not reflect their subjective estimates, thereby adjusting the price upwards (downwards), moving it towards a hypothetical consensus estimate. By mapping the prices of these securities to the $[0, 1]$ interval they can be interpreted as probability estimates [31].

The *marginal trader hypothesis* is often cited as the driving force. It proposes that a small number of well-calibrated traders ("wolves") are responsible for efficient price formation. They are being attracted to the market by "noise traders" ("sheep") who trade for any other reason than profit maximization, be it entertainment, insurance, or manipulation attempts [10]. However, this hypothesis has been called into question [16].

In any case, the designation as *information aggregation markets* should be emphasized, which makes it clear that only information that is available can be aggregated by such a structure.

2.3.2 Notation

Formally, we assume an outcome space $\mathcal{O}$ that contains n (finite) mutually exclusive and exhaustive outcomes $\{o_1, \ldots, o_n\} = \mathcal{O}$. The simplest case is a binary outcome $\{o_\top, o_\bot\} = \mathcal{O}$. $\Delta(\mathcal{O})$ denotes the set of probability distributions over $\mathcal{O}$. We assume each trader has a private belief $p \in \Delta(\mathcal{O})$. If a trader observes market prices that correspond to a $p_m \in \Delta(\mathcal{O})$ that is different from her private belief p, she can maximize her expected utility by buying or selling assets until $p_{m'} = p$, where $p_{m'}$ denotes the market prices after the trader is finished trading.

2.3.3 CDA prediction markets

Building on this model, a CDA prediction market can be created.

Unlike commodity or stock exchanges, prediction markets feature virtual goods. The CDA mechanism leaves open the question how these goods enter the market in the first place. Unlike goods in conventional markets, they can't be produced in the real-world. Luckily, they can be created in a risk-free manner for both the market operator and traders:

Assume the payoff of a security is $1. Since it is required that the outcome space is exhaustive and outcomes are mutually exclusive, the market operator can sell a bundle of securities, containing one security for each possible outcome, at $1 without risk. Since only one outcome can eventually be true, the market operator needs to pay out $1 for only one security of the bundle.

On the other hand, the buyer of the bundle doesn't face any risk either, since at least one of the securities she owns will be worth $1 at maturity. However, in order to gain a profit, she needs to sell some of the securities in the market at or above prices that reflect her subjective probability estimate. This way new contracts enter the market.

This concludes the chapter on prediction markets. In this chapter the concept of prediction markets was introduced. Additionally, critical terminology regarding markets in general, and prediction markets in particular, has been presented. The following chapter will show how the concept of prediction markets has been advanced in recent times.

CHAPTER 3

Market Scoring Rules

The last chapter introduced a CDA-based prediction market. However, the computer science research on prediction markets has shifted its focus away from traditional market designs, because they are geared towards facilitating the exchange of goods, whereas for prediction markets, designs that facilitate the sharing of information are preferable.

Specifically, the *thin-market problem* had to be addressed. When hosting prediction markets on arbitrary subjects, one can not expect the kind of popular interest that sport events and elections attract, resulting in little or no trading activity. Such markets are referred to as "thin". To turn a thin market into a "think" market, active traders are needed. Unfortunately, they are mostly interested in active markets, i.e. markets with other traders in it, resulting in a circularity. Market scoring rules are a solution to this problem.

Notational conventions In this chapter, vectors are represented as $\vec{p}$. Individual elements of vectors are referenced by their index via subscripts, e.g. p_i. Subscripts are also used to differentiate versions of the same variable, e.g. $p_{i,t}$ is the value at index i at time t. Expressions like $\sum_j p_j$ are a shorthand for the sum over all indices of the vector, while $\sum_{j \neq i} p_j$ is a shorthand for the sum over all indices except i.

3.1 Automated market makers

The thin-market problem can be overcome using market makers. These are (possibly algorithmic) market participants who are able to quote prices at all times and accept trades at these prices. They act as central actors in the market, intermediating all others.

For example, the Hollywood Stock Exchange (HSX), a popular play-money prediction market for movie box office results, uses a proprietary automated market maker. In the academic research, Hanson's market scoring rules have become popular, which are a form of automated market maker [9].

Using an automated market maker, it would be possible to run a prediction market with just a single participant. In this scenario, the participant would bet against the market maker, so that the resulting price would reflect his subjective probability estimate. While this seems counterintuitive, this is not unlike a principal asking an agent for a advice (in form of a probability distribution over a set of outcomes). In order to ensure honest reporting, the principal would like to pay the agent based on the observed outcome. One may notice that this concept shares similarities with the contracts being traded in prediction markets.

In fact, this concept has been know since 1950 [4], and has been referred to as *scoring rules* since 1971 [19]. The key contribution of Hanson [8, 9] was to modify scoring rules, so that an arbitrary number of experts can report their probability estimates, while keeping the costs to the principal fixed. These are known as *market scoring rules* and, as the name suggests, have a close relation to prediction markets.

In this chapter, scoring rules will be discussed in detail. Then, Hanson's adaptation of scoring rules to market scoring rules will be presented. An interpretation for market scoring rules as prediction markets will follow. Finally, the LMSR will be introduced.

3.2 Scoring rules

Independent of the development of prediction markets, scoring rules have long been used to receive honest probability reports from individual experts. A notable example are "Brier scores" used in weather forecasting [4].

3.2.1 Intuition

Scoring rules are useful when a party (the patron or agent) is interested in paying an expert for a report, in the form of a probability distribution, based on its accuracy. A scoring rule can be used to predetermine what the reward will be for each possible pair of expert report and observed outcome.

E.g. an expert may believe that a binary event has a 80% chance of occurring. He accepts being compensated according to a scoring rule. Based on the scoring rule, in case the event occurs, he will receive a generous compensations, while he will receive a penalty if the event does not occur. Given a "fair" scoring rule and that the "true" probability is indeed 80%, he can expect to receive a favorable reward over a larger number of such reports. The following section will establish this formally.

3.2.2 Formal model

We assume a finite, mutually exclusive and exhaustive set of outcomes $\{o_1, \ldots, o_n\} = \mathcal{O}$, not unlike those found in weather forecasting (sunny, cloudy, rain, ...). Further, we assume an expert has a private belief $\vec{p} \in \Delta(\mathcal{O})$ regarding those outcomes, where $\Delta(\mathcal{O})$ is the set of probability distributions over $\mathcal{O}$.

Experts are thought of as *risk-neutral*, *expected utility maximizing*, i.e. they will not discriminate between a guaranteed reward of \$0.80 and a 80% chance of winning \$1.00, or even a guaranteed reward of \$1.000 and a 0.1% chance of winning \$1.000.000. It should be noted that these assumptions are only made for the purpose of establishing incentive compatibility.

An expert may report a probability distribution $\vec{r} \in \Delta(\mathcal{O})$ and be rewarded a cash payment c_i if event o_i is observed. The reward is determined by a scoring rule $s = \{s_1, \ldots, s_n\}$, a sequence of scoring functions, one for each possible outcome, so that $c_i = s_i(\vec{r})$ [19]. The reward may be negative (e.g. when an expert assigns a low probability to an event that subsequently occurs).

3.2.3 Constraints

In order in incentivize the expert to report her true belief $\vec{p}$, the scoring rule has to satisfy an incentive-compatibility constraint

$$\vec{p} = \arg\max_{\vec{r}} \sum_i p_i \, s_i(\vec{r}) \tag{3.1}$$

Given this constraint, it is optimal to set $\vec{r} = \vec{p}$.

To incentivize the expert to participate at all, the scoring rule has to satisfy a rational participation constraint, i.e. the expert receives a positive reward in expectation.

$$\sum_i p_i \, s_i(\vec{p}) > 0 \tag{3.2}$$

A scoring rule that is real-valued for all $\vec{r}$ (except possibly $-\infty$ for $r_i = 0$) is said to be "regular". A scoring rule that satisfies equation 3.1 is referred to as *proper*. If there exists a unique report $\vec{r}$ that maximizes the experts expected reward, the scoring rule is said to be *strictly proper* [29]. Going forward, this thesis will assume strictly proper scoring rules.

3.2.4 Common scoring rules

Examples of proper scoring rules include the quadratic scoring rule (3.3) [4], the spherical scoring rule (3.4) and the logarithmic scoring rule (3.5) [9].

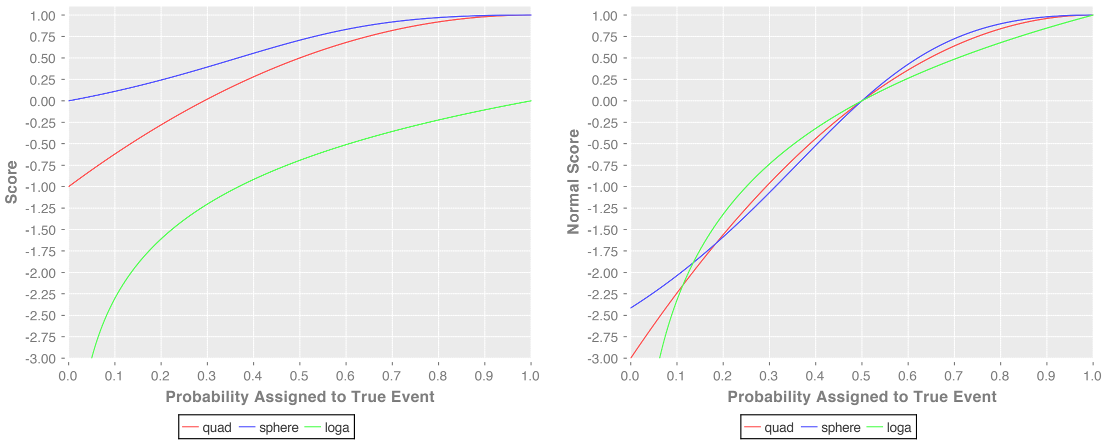
$$s_i(\vec{r}) = a_i + 2b\,r_i - b\sum_j r_j^2 \tag{3.3}$$

$$s_i(\vec{r}) = a_i + \frac{b\,r_i}{\sqrt{\sum_j r_j^2}} \tag{3.4}$$

$$s_i(\vec{r}) = a_i + b\,\log r_i \tag{3.5}$$

Here, $a_1, \ldots, a_n$ and $b > 0$ represent parameters which can be used to affine transform the curve. They are usually set to $a_i = 0\,\forall i$ and $b = 1$ for simplicity.

Notably, the Brier score mentioned earlier is an affine transformed version of the quadratic scoring rule.



(a) Raw score

(b) Normalized score

Figure 3.1: Comparison of the quadratic, spherical and logarithmic scoring rule.

Figure 3.1 provides a comparison of the quadratic, spherical and logarithmic scoring rule for a binary outcome $\mathcal{O} = \{o_\top, o_\bot\}$. Figure 3.1a shows the "raw" score of the scoring rules, i.e. the scaling parameters are set to $a_\top = a_\bot = 0$ and $b = 1$, for reporting different probability estimates for the outcome that subsequently turns out to be true.

As expected, each scoring rule assigns its highest score when reporting a probability estimate of 1 for the true outcome. It should be noted that the logarithmic scoring rule is unbounded from below, assigning a score of $-\infty$ for reporting a probability estimate of 0 for the event that turns out to be true.

For better comparison, figure 3.1b shows a normalized score for each scoring rule. This is achieved by choosing different values for $a_\top$, $a_\bot$ and b, so that reporting 1 for the correct outcome awards a score of 1, and reporting equal probabilities awards a score of 0, which

	a	b
Quadratic	-1	2
Spherical	-2.41421	3.41421
Logarithmic	1	1.4427

Table 3.1: Parameters for normalized scoring rules for binary decisions.

is thought of as the baseline prediction. Intuitively, an expert reporting the baseline does not add new information and should therefore not receive a reward.

The values for affine transforming the raw scoring rules into the normalized scoring rules are obtained by solving the following system of equations

$$s_\top\left[\left(\begin{smallmatrix}0.5\\0.5\end{smallmatrix}\right)\right] = 0$$
$$s_\top\left[\left(\begin{smallmatrix}1\\0\end{smallmatrix}\right)\right] = 1$$

for $a = a_\top = a_\bot$ and b. Table 3.1 depicts the results.

It should be noted that affine transformation leaves the incentive compatibility constraint (3.1) intact for all values of $a_1, \ldots, a_n$ and all $b > 0$.

3.3 Sequential sharing

A natural limitation of scoring rules is that a probability distribution can only be elicited from a single expert. One could ask multiple experts and use a linear combination of their reports as a collective forecast. This has the obvious disadvantage that the costs of doing so increase linearly with the number of experts. Also, it is unclear how much weight to assign to each report. One could assign equal weights, or heavier ones to reports from experts that performed well in the past. However, as it turns out, there is a better way to approach this limitation.

3.3.1 Intuition

In Chapter 2 the concept of prediction markets was introduced, which are able to elicit a collective probability distribution and letting the participants “chose their own weights” by making bets in the market. However, in order for this to work, a sufficient amount of liquidity needs to exists in the market which may not be the case in many instances.

It was Hanson’s insight that one could “share” a scoring rule sequentially between an arbitrary number of experts without increasing the costs or predefining how much weight to put on any one expert’s report. In this version of the scoring rule, every participant can report a probability distribution at any time, as long as he is willing the pay the last expert according to the rule. Naturally, reporting the same distribution as the last expert gives no reward, similar to how reporting the baseline distribution in a normalized scoring rule (as in Figure 3.1b) awards a score of 0. However, if an expert has knowledge of a more accurate probability distribution, he can net the difference between his report and the previous report. Meanwhile, the patron only needs to pay the last expert, which puts an upper bound on the cost.

3.3.2 Example

Assume the normalized logarithmic scoring rule, scaled by 10 ($a_i = 10\,\forall i$, $b = 14.427$). The current prediction for a binary event is 60%, which has been reported by the previous expert. The current expert believes the true probability is 80%. She agrees to pay the previous expert according to the rule, which is \$2.63 in case the event occurs, and −\$3.22 in case it doesn't (c.f. equation 3.5, figure 3.1b). Here, a negative amount means that she receives a payment from the previous expert.

Assuming the event occurs, the expert receives a reward of \$6.78 for her 80% prediction according to the rule. Since she still has to pay the previous expert, she nets a reward of \$4.15 = \$6.78 − \$2.63.

Assuming the event does not occur, the experts receives \$3.22 from the previous expert, but according to the rule, has to pay \$13.22 to the next expert for her implicit 20% prediction. This leaves her with −\$10 = −\$13.22 + \$3.22.

Note that it is still rational for the expert to participate, since, according to her subjective belief, losing \$10 would only happen in 20% of cases, while winning \$4.15 would happen in 80% of cases. Therefore, her expected reward is $0.8 \times \$4.15 + 0.2 \times -\$10 = \$1.32$, which is positive.

3.3.3 Costs

In a sequentially shared scoring rule, the reward c_i of an expert does not only depend on his report $\vec{r}_t$, but also the report of the previous expert $\vec{r}_{t-1}$. The reward is the difference in score between the two reports

$$c_i = \Delta s_i(\vec{r}_t, \vec{r}_{t-1}) = s_i(\vec{r}_t) - s_i(\vec{r}_{t-1}) \tag{3.6}$$

Since an expert can't change the previous expert's report, he will still maximize his score by setting $\vec{r}_t = \vec{p}$ [8].

The total cost of a market scoring rule to a patron is the sum of the costs of all reports. Assuming a total of T reports, using (3.6), the total costs is

$$\sum_{t=1}^{T} c_{i,t} = \sum_{t=1}^{T} s_i(\vec{r}_t) - s_i(\vec{r}_{t-1}) \tag{3.7}$$

$$= s_i(\vec{r}_1) - s_i(\vec{r}_0) + s_i(\vec{r}_2) - s_i(\vec{r}_1) + \cdots + s_i(\vec{r}_T) - s_i(\vec{r}_{T-1}) \tag{3.8}$$

$$= s_i(\vec{r}_T) - s_i(\vec{r}_0) \tag{3.9}$$

The intermediary terms cancel each other out, so that the cost only depends on the first and the last report. Since the patron controls the first report, effectively the cost only depends on the last report.

The maximum payout, or worst-case loss to the patron, occurs when the last expert reports 1 for the true outcome. Assuming the patron has no prior knowledge, therefore sets the first report to the uniform report, and that o_i is the true outcome, the worst-case loss is bounded by

$$s_i(\vec{1}_i) - s_i(\vec{r}_U)$$

where $\vec{r}_U$ is the uniform report, i.e. $r_i = \frac{1}{n}\,\forall i$ (with $n = |\mathcal{O}|$), and $\vec{1}_i$ is a vector with the i-th element set to 1 and all other elements set to 0.

3.3.4 Worst-case bounds

For the quadratic scoring rule (3.3), the worst case loss is bounded by

$$\begin{aligned} s_i(\vec{1}_i) - s_i(\vec{r}_U) &= a_i + 2b\,1 - b\,1^2 - \left(a_i + 2b\,\frac{1}{n} - b\,n\left(\frac{1}{n}\right)^2\right) \\ &= b\left(1 - \frac{1}{n}\right), \end{aligned}$$

for the spherical scoring rule (3.4), the worst case loss is bounded by

$$\begin{aligned} s_i(\vec{1}_i) - s_i(\vec{r}_U) &= a_i + \frac{b\,1}{\sqrt{1^2}} - \left(a_i + \frac{b\,\frac{1}{n}}{\sqrt{n\left(\frac{1}{n}\right)^2}}\right) \\ &= b\left(1 - \frac{1}{\sqrt{n}}\right), \end{aligned}$$

and for the logarithmic scoring rule (3.5), the worst case loss is bounded by

$$\begin{aligned} s_i(\vec{1}_i) - s_i(\vec{r}_U) &= a_i + b\,\log 1 - \left(a_i + b\,\log\frac{1}{n}\right) \\ &= b\,\log n. \end{aligned}$$

Notably, all market scoring rules above have bounds that depend only on the number of possible outcomes n and the parameter b.

3.3.5 Infinitesimal reports

Equation 3.7 shows that the movement from $\vec{r}_0$ to $\vec{r}_T$ can be broken up into smaller movements from $\vec{r}_t$ to $\vec{r}_{t+1}$, which are the individual reports of the experts. Similarly, experts can split their own reports into two separate reports, $\vec{r}_t$ to $\vec{r}_{t+\frac{1}{2}}$ and $\vec{r}_{t+\frac{1}{2}}$ to $\vec{r}_{t+1}$, at no additional cost. In fact, an expert can split her report into an infinite amount of reports, each moving the estimate an infinitesimal amount towards $\vec{r}_{t+1}$ [9]. This property simply follows from the fact that s_i is a function.

3.4 Scoring rules as markets

Section 3.3 showed that scoring rules can be shared among an arbitrary number of experts without additional cost to the patron, as long as each expert agrees to pay the previous expert according to the rule. However, so far it is unclear how scoring rules relate to prediction markets, where it is assumed that certain assets are being traded.

This chapter will show that sequentially shared scoring rules can be interpreted as *cost function-based automated market makers*, which enable interactions similar to prediction markets.

3.4.1 Cost function-based automated market makers

A cost function-based automated market maker is defined by a cost function C. In addition, it maintains a quantity vector $\vec{q}$ of the number of contracts sold so far, where q_i is the number sold for outcome o_i. The cost function maps the quantity vector to the total amount of money collected from all traders $\$c_{\text{total}} = C(\vec{q})$ [8].

After trading has stopped and o_i is the true outcome, the market maker needs to pay out \$1 to each trader that holds the corresponding asset, which is a total of $\$1 \times q_i$. However, it gets to keep c_{total}, the amount that was collected from the traders in exchange for all assets, including the ones for o_i.

In order to achieve bounded worst-case loss, the market maker needs to buy and sell all contracts at a price so that the difference between q_i and c_{total} stays within that bound for all possible outcomes.

This is best illustrated via the price function, which is the first-order derivative of the cost function [6].

$$p(\vec{q}) = \frac{\partial C}{\partial \vec{q}} \tag{3.10}$$

The price function maps the quantity vector to the *risk-neutral* price of each asset, which is the price at which the algorithm is indifferent to either buying or selling an infinitesimal amount of the asset (these market makers have zero bid-ask spread). It can be interpreted as the current price of the asset and is also called the instantaneous price [9, 6].

After buying or selling each infinitesimal amount, the market maker quotes a new risk-neural price, which is determined by the price function. It should be noted that each larger trade can be broken up into smaller trades (cf. section 3.3.5). Buying or selling a larger, non-infinitesimal amount of any asset, then, is the integral of the price function over the distance from the current quantity vector to the quantity vector after the trade [6].

If a trader wants to buy/sell a bundle of assets $\Delta\vec{q} = \vec{q}' - \vec{q}$, where $\vec{q}$ is the current quantity vector and $\vec{q}'$ is the quantity vector after the trade, the price of the bundle is determined by

$$\int_{\vec{q}}^{\vec{q}'} p(\vec{x})\, d\vec{x} = C(\vec{q}') - C(\vec{q}) \tag{3.11}$$

which follows directly from the definition of the price function.

For example, buying two assets of the "true" outcome and selling one for the "false" outcome $\Delta\vec{q} = \left(\begin{smallmatrix} 2 \\ -1 \end{smallmatrix}\right)$ in a binary prediction market, where 20 "true" and 18 "false" assets have already been sold ($\vec{q} = \left(\begin{smallmatrix} 20 \\ 18 \end{smallmatrix}\right)$), costs $\$\, C[\left(\begin{smallmatrix} 22 \\ 17 \end{smallmatrix}\right)] - C[\left(\begin{smallmatrix} 20 \\ 18 \end{smallmatrix}\right)]$.

The exact amount, how much the prices change in response to trades, and what the worst-case loss of the market maker is, depends on the choice of the cost function C. This will be discussed in the following section.

3.4.2 Equivalence

It is possible to construct a cost function-based automated market maker from any market scoring rule [6]. The equivalence can be established as follows:

- In a market scoring rule, an expert who has private probability estimate $\vec{p}'$ changes the estimate from $\vec{p}$ to $\vec{p}'$ and expects a reward of $\$\Delta s_i(\vec{p}', \vec{p}) = s_i(\vec{p}') - s_i(\vec{p})$ when outcome o_i occurs.
- When trading with a cost function-based automated market maker, a trader who has a private probability estimate $\vec{p}'$ buys/sells a bundle of assets $\Delta\vec{q}$ from/to the market maker, so that prices change to $\vec{p}'$. She expects a profit of $\$(q_i' - q_i) - (C(\vec{q}') - C(\vec{q}))$ for outcome o_i, which is the payoff of \$1 per share times the number of shares, minus the costs of buying those shares according to the market maker's cost function.

Following this rationale, the equivalent cost function for a given market scoring rule can be obtained by equating the payoffs under both scenarios for all i and finding a C, so that

$$s_i(\vec{p}') - s_i(\vec{p}) = (q_i' - q_i) - (C(\vec{q}') - C(\vec{q})) \tag{3.12}$$

under consideration of $\vec{p} = \frac{\partial C}{\partial \vec{q}}$ and $\sum_i p_i = 1$.

Chen and Pennock [6] show that the following system of equations establishes an unambiguous equivalence between market scoring rules and cost function-based market makers:

$$\begin{cases} s_i(\vec{p}) = q_i - C \quad \forall i \\ \sum_i p_i = 1 \\ p_i = \dfrac{\partial C}{\partial q_i} \end{cases} \tag{3.13}$$

3.5 The Logarithmic Market Scoring Rule

The LMSR is the *de facto* standard market scoring rule in practice, as it is easy to understand and implement. Much of this thesis will focus on the LMSR for this reason.

3.5.1 Cost and price function

The logarithmic scoring rule is defined as $s_i = b \log(r_i)$ (setting $a_i = 0$). According to [6] the corresponding cost function is

$$C(\vec{q}) = b \log\left(\sum_j e^{\frac{q_j}{b}}\right) \tag{3.14}$$

and the corresponding price function, which is obtained by taking the partial derivative, is

$$p_i(\vec{q}) = \frac{e^{\frac{q_i}{b}}}{\sum_j e^{\frac{q_j}{b}}} \tag{3.15}$$

The equivalence between the cost function and the market scoring rule can be shown via (3.13):

$$\begin{aligned} s_i(\vec{p}) &= q_i - C(\vec{q}) \\ b \log(p_i) &= q_i - b \log\left(\sum_j e^{\frac{q_j}{b}}\right) \\ b \log\left(\frac{e^{\frac{q_i}{b}}}{\sum_j e^{\frac{q_j}{b}}}\right) &= q_i - b \log\left(\sum_j e^{\frac{q_j}{b}}\right) \end{aligned}$$

$$b \log e^{\frac{q_i}{b}} - b \log \left(\sum_j e^{\frac{q_j}{b}} \right) = q_i - b \log \left(\sum_j e^{\frac{q_j}{b}} \right)$$

$$b \, \frac{q_i}{b} \, \log e = q_i$$

$$q_i = q_i$$

This shows that, given equal initial conditions, an expert reporting a $\vec{p}\,'$ to a LMSR receives the exact same reward as a trader buying shares $\Delta\vec{q}$, so to that prices correspond to $\vec{p}\,'$, when trading with an automated market maker with cost function (3.14).

3.5.2 Liquidity Parameter

The b parameter in equation (3.14) and (3.15) plays an important role when trading. It is also called the *liquidity parameter*, as it defines how much prices move in response to trades. A larger b is generally preferred by traders, as it allows them to buy/sell more shares at a similar price level. However, from the patron's perspective, a larger b means a higher worst-case loss, as has been established in section 3.3.4. For the LMSR, the worst-case loss is defined by $b \log n$.

Figure 3.2 shows the price of an asset in a binary prediction market $\vec{q} = \left(\begin{smallmatrix} x \\ 0 \end{smallmatrix}\right)$ in response to the number of shares being bought with $b = 10$, $b = 25$, and $b = 100$. The worst-case losses are \$6.93, \$17.33, and \$69.31, respectively.

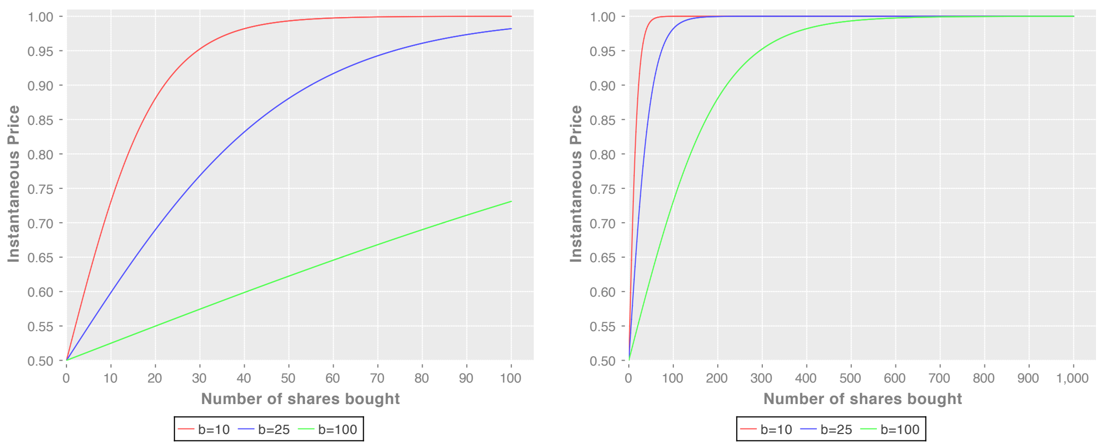


Figure 3.2: Price of an asset in response to the number of outstanding shares.

One may object that predefining the liquidity parameter is impossible, as it requires to know the popularity of a given market in advance. Othman and Pennock [17] provide a solution that lets market scoring rules dynamically adjust their liquidity. For the sake of simplicity, this thesis will continue with the assumption that b is known and constant.

This concludes the chapter on market scoring rules. The following chapter will introduce the Kelly criterion.

CHAPTER 4

The Kelly Criterion

The Kelly criterion is a classic formula to devise the fraction of one's wealth to commit to each bet in a series of independent bets in order to maximize one's expected rate of growth. In other words, given certain conditions, no other fraction would lead to a better outcome over an infinite number of bets [15].

In practice, *fractional* Kelly betting, i.e. betting a fraction of full Kelly, is popular since it gives leeway to estimation errors and reduces variance [15].

This chapter is structured as follows: Section 1 will introduce the classic Kelly formula, and section 2 will show how the formula is derived and how it achieves its goal of maximizing expected rate of growth.

Notational Convention In this chapter, upper-case letters refer exclusively to random variables. $\mathbb{E}$ is the expectation operator and Pr is a probability measure. E.g.: $\mathbb{E}[X]$ is the expected value of random variable $X \sim f(x)$, with $\Pr[X \leq x] = F(x)$.

4.1 Formula

This section will introduce basic betting terminology and the original Kelly formula for two-outcome bets.

4.1.1 Betting odds and implied probability

The Kelly criterion is usually explained in terms of fractional odds, which are expressed as "b to 1". Such a bet means that a bettor receives a payout of \$$b$ for every dollar staked.

For example, a "2 to 1" bet will pay \$2 for every dollar staked. In case of a ten dollar bet, the change in wealth for the bettor would be +\$20 in case of a win and −\$10 otherwise.

When a bookie is offering "b to 1" bets, he is making an implicit prediction about the event. This is referred to as *implied probability.* The implied probability of the event in the example above is $\frac{1}{1+2} = 0.3\overline{3}$. In general, the implied probability p_i of a "b to 1" bet is given by

$$p_i = \frac{1}{1+b} \tag{4.1}$$

If the implied probability is indeed the true probability of the event, the expected gain of both the bettor and the bookie is 0. Any deviation of the implied probability from the true probability favors either the bettor or the bookie. It is such deviations that the Kelly criterion is concerned with.

4.1.2 Classic formula

The classic Kelly formula is defined as follows: Given a series of "b to 1" bets, the fraction f^* of wealth w to bet each time is

$$f^* = \frac{bp - (1-p)}{b} \tag{4.2}$$

where p is the "true" probability of the event.

It assumes that the bets are favorable to the bettor. In other words, the bettor has better knowledge of the true probability p than is implied by b. It is said that the bettor has an "edge". In lieu of such an advantage, i.e. $p = p_i$, Kelly advises not to bet (one can see that $f^* = 0$ in this case).

It should be noted that favorable bets alone are no guarantee for long-term gain. Trivially, if the bettor sets $f = 1$, i.e. he bets his entire wealth every time, he faces near-certain ruin after a small number of bets. Conversely, if he sets $f = 0$, the possibility of losing is avoided, but he has also failed to profit from his advantage. The Kelly fraction f^* is a reasonable, and arguably optimal, compromise between the two.

4.2 Derivation

The following derivation of the Kelly criterion is based on [13], which is in turn a modified version of [23] and [24]. It is extended with insights from [27].

4.2.1 Rate of growth

Assume a bettor with initial wealth w_0. After each bet, the bettor's wealth is multiplied by a random variable X. Thus, his wealth W_n after n bets can be modeled as

$$W_n = w_0 \, X_1 X_2 \cdots X_n \tag{4.3}$$

Using basic operations and the law of large numbers, it can be said that

$$\begin{aligned} W_n &= w_0\, X_1 \cdots X_n \\ &= e^{\log(w_0\, X_1 \cdots X_n)} \\ &= e^{\log(w_0)+\log(X_1)+\cdots+\log(X_n)} \\ &= w_0\, e^{\log(X_1)+\cdots+\log(X_n)} \\ &\approx w_0\, e^{\mathbb{E}[\log(X)]\, n} \qquad (4.4) \end{aligned}$$

The last step is valid by the law of large numbers, which (loosely) states that a sum of n independent samples of a random variable Y is approximately equal to $n \cdot \mathbb{E}[Y]$.

Equation 4.4 can be contrasted with the exponential growth formula, which is

$$x(t) = x_0\, e^{gt} \qquad (4.5)$$

where g is the exponential growth factor. This comparison shows that in the case of repeated gambles the growth factor is equal to $\mathbb{E}[\log(X)]$.

Solving (4.4) for $\mathbb{E}[\log(X)]$, the growth factor over n bets, which is denoted as G_n, can be defined as

$$\begin{aligned} G_n &:= \mathbb{E}\left[\log(X)\right] \\ &= \log\left(\frac{W_n}{w_0}\right)^{\frac{1}{n}} \qquad (4.6) \\ &= \frac{1}{n}\log(W_n) - \frac{1}{n}\log(w_0) \qquad (4.7) \end{aligned}$$

It should be noted that G_n is a random variable because it depends on W_n, which is, in turn, a random variable.

As mentioned in the introduction, Kelly suggests to maximize the expectation of this variable, i.e. to maximize $\mathbb{E}[G_n]$ [14]. It was claimed earlier that maximizing the expected rate of growth is equivalent to maximizing the expected logarithm of wealth. This can be shown to be true as follows:

Building on (4.7) and by the linearity of expectations,

$$\begin{aligned} \mathbb{E}[G_n] &= \mathbb{E}\left[\frac{1}{n}\log(W_n) - \frac{1}{n}\log(w_0)\right] \\ &= \frac{1}{n}\,\mathbb{E}[\log(W_n)] - \frac{1}{n}\log(w_0) \end{aligned} \tag{4.8}$$

Since w_0 and n are fixed, $\mathbb{E}[G_n]$ is at its maximum when $\mathbb{E}[\log(W_n)]$ is at its maximum.

4.2.2 Choosing a fraction to bet

This section will show how a bettor can maximize his expected rate of growth by choosing a fraction f of his wealth to bet.

Generally, a bet pays b units per unit wagered for a successful bet, and costs a units per unit wagered when unsuccessful (typically $a = 1$, i.e. the amount wagered is lost). The stochastic multiplier X, as seen in 4.3, can be defined based on this description as

$$X(\omega) = \begin{cases} 1 + bf & \omega = \text{win} \\ 1 - af & \omega = \text{lose} \end{cases} \tag{4.9}$$

where $0 \leq f \leq 1$ is a fraction and $\omega \in \{\text{win}, \text{lose}\}$. Further, $\Pr(\{\text{win}\}) = p$ is the probability of winning and $\Pr(\{\text{lose}\}) = 1 - p$ is the probability of losing.

Now assume a bettor with wealth w_0 accepts n such bets, S_n of which succeed and F_n of which fail, so that $S_n + F_n = n$. Using the definition of X and (4.3), his wealth W_n after n bets is

$$W_n = w_0\,(1 + bf)^{S_n}(1 - af)^{F_n} \tag{4.10}$$

Inserting this result in (4.6), the bettor's exponential growth rate G_n can be expressed as a function of f.

$$\begin{aligned} G_n(f) &= \log\left[\frac{w_0\,(1+bf)^{S_n}(1-af)^{F_n}}{w_0}\right]^{\frac{1}{n}} \\ &= \frac{S_n}{n}\log(1+bf) + \frac{F_n}{n}\log(1-af) \end{aligned} \tag{4.11}$$

As discussed earlier, Kelly suggests to maximize $\mathbb{E}[G_n(f)]$. Since $\mathbb{E}[S_n] = np$ and $\mathbb{E}[L_n] = n(1 - p)$, a function $g(f)$ can be defined as

$$
\begin{aligned}
g(f) &:= \mathbb{E}[G_n(f)] \\
&= \mathbb{E}\left[\frac{S_n}{n}\log\left(1+bf\right) + \frac{F_n}{n}\log\left(1-af\right)\right] \\
&= p\log\left(1+bf\right) + (1-p)\log\left(1-af\right) \qquad (4.12)
\end{aligned}
$$

Since f is the only parameter under the bettor's influence, she will want to choose f so that $g(f)$ is maximal,

$$f^* = \arg\max_f g(f) \qquad (4.13)$$

This point can be found by solving $g'(f^*) = 0$, where g' is the derivative of g with respect to f. It is

$$g'(f) = \frac{bp}{1+bf} + \frac{a(1-p)}{1-af} \qquad (4.14)$$

and consequently f^* is given by

$$
\begin{aligned}
\frac{bp}{1+bf^*} + \frac{a(1-p)}{1-af^*} &= 0 \\
\implies f^* &= \frac{bp - a(1-p)}{ab} \qquad (4.15)
\end{aligned}
$$

With $a = 1$, this is equal to (4.2), the classic Kelly formula as stated earlier.

4.2.3 Visualization

Figure 4.1 shows an example curve of $g(f)$ with values of $a = 1$, $b = 1$ (implied probability $p_i = 0.5$) and true probability $p = 0.7$. The highest expected rate of growth is achieved when betting 40% per bet. Despite the advantage, betting more than $\approx 72\%$ of one's wealth results in a negative growth rate.

Figures 4.2a to 4.2f show example histories over 25 bets for a bettor betting according to Kelly in the example above (resulting in a fraction of 0.4 per bet). As can be seen, the end results are generally favorable, but are subject to variability and depend a lot on the outcomes of earlier bets. As figure 4.2e shows, having an advantage and betting according to Kelly is still no guarantee for success, as it can only diminish the role of luck, not prevent it.

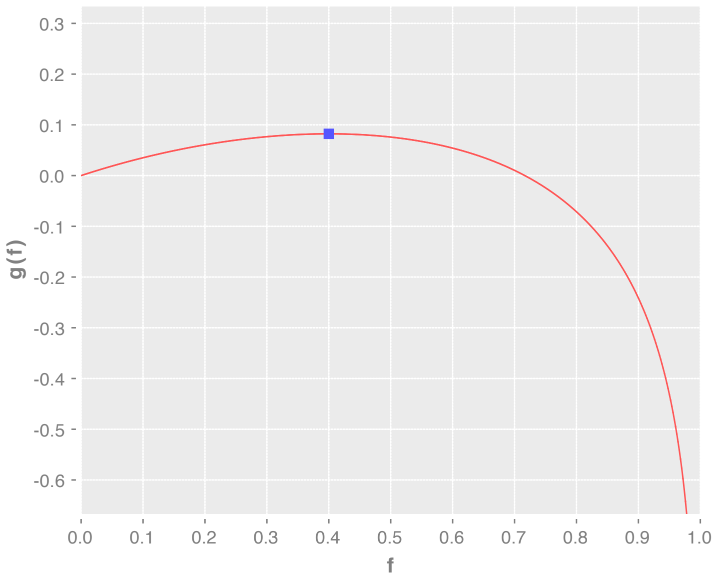


Figure 4.1: Expected rate of growth in relation to fraction of wealth being bet, for iterated even-money bets when the true probability of winning is 70%.

Figure 4.2: Example histories of 25 even-money bets when the true probability of winning is 70%, risking a fixed fraction of 0.4 (Kelly) per bet.

CHAPTER 5

Application of Kelly to Market Scoring Rules

This chapter will apply the Kelly criterion to trading in prediction markets based on cost function-based automated market makers. As discussed in section 3.4.1, these automated market makers change their prices in response to trading, so a naive application of the Kelly criterion will necessarily result in suboptimal results. This may have little consequence when the amounts traded are small in relation to the liquidity parameter, but can lead to substantially higher amounts being bet otherwise.

The following sections will provide a naive solution, a "capped" solution, and the exact solution to the problem. Then, section 5.5 will run simulations and discuss the results.

5.1 Naive application

A naive application of the Kelly criterion to market scoring rules will not consider the price adjustment of the scoring rule. Technically, this approach isn't an application of the Kelly criterion, since it fails to maximize expected rate of growth, but it is a natural starting point.

5.1.1 Basics

The classic Kelly formula (4.2) as presented earlier assumes bets of the form "b to 1". As has been discussed in chapter 2, prediction markets do not offer bets of this form. Instead, an asset that might pay \$1 is bought at a certain market price p_m. The relationship between the two is established through

$$b = \frac{1 - p_m}{p_m} \tag{5.1}$$

as in [3].

The original Kelly formula can thus be restated as

$$f^* = \frac{bp - (1-p)}{b} = \frac{p - p_m}{1 - p_m} \tag{5.2}$$

In the context of the Kelly criterion, p denotes the "true" probability of a successful gamble. In the context of prediction markets, it denotes a trader's subjective belief.

5.1.2 Inverse bets

Following the interpretation of Kelly as a "criterion", it states that one should not take a gamble when $f^* < 0$. Conversely, one should not buy an asset in a prediction market, when its price p_m is higher than one's subjective probability estimate p. Looking at equation 5.2, it can been seen that f^* is negative when $p_m > p$.

However, in a prediction market one can always take the opposite side of a bet by (short-) selling the asset, or buying a fraction $\frac{1}{n-1}$ of each other asset. In the case of a binary prediction market this means buying the opposite asset. In any case, since all p_i sum to 1, the costs of doing so are always $\overline{p_m} = (1 - p_m)$.

In other words, there is always an inverse bet with odds "$\overline{b}$ to 1" being offered by the market. Analogous to (5.1), the relationship between $\overline{b}$ and p_m is

$$\overline{b} = \frac{1 - \overline{p_m}}{\overline{p_m}} = \frac{p_m}{1 - p_m} \tag{5.3}$$

which leads to an inverse Kelly fraction $\overline{f^*}$ of

$$\overline{f^*} = -\frac{\overline{b}\overline{p} - (1 - \overline{p})}{\overline{b}} = \frac{p - p_m}{p_m} \tag{5.4}$$

where $\overline{p} = 1 - p$ is the expert's subjective belief that the event in question will not occur. This fraction is always negative (assuming $p_m > p$) to denote that the fraction should be used to (short-) sell the asset.

Combining (5.2) and (5.4), it can thus be said that the ideal fraction f_c^* to bet in a prediction market is

$$f_c^* = \begin{cases} \dfrac{p - p_m}{1 - p_m} & p_m < p \\[2ex] \dfrac{p - p_m}{p_m} & \text{otherwise} \end{cases} \tag{5.5}$$

where a negative value means that the fraction should be used to bet against the event. It should be noted that $f_c^* = 0$ when $p_m = p$.

5.1.3 Number of shares

Assume a trader with wealth w has a subjective belief $p_{s,i}$ about *one* particular outcome $o_i \in \mathcal{O}$. She might be interested in the exact number Δq_i of shares to buy according to Kelly, when being offered to buy at a market price of $p_{m,i}$. The number is determined by the total money to spend (the Kelly fraction times wealth) over the price per share. According to (5.5),

$$\Delta q_i = \frac{f_c^* w}{p_{m,i}} = \begin{cases} w \dfrac{p_{s,i} - p_{m,i}}{p_{m,i} - {p_{m,i}}^2} & p_{m,i} < p_{s,i} \\[2ex] w \dfrac{p_{s,i} - p_{m,i}}{{p_{m,i}}^2} & \text{otherwise} \end{cases} \tag{5.6}$$

Again, a negative quantity denotes a sell orders.

Going forward, this will be referred to as the “naive” application of the Kelly criterion to prediction markets. The reasons are discussed in the next section.

5.1.4 Price adjustment

So far, it was assumed that an arbitrary amount of shares can be purchased at the fixed price of $p_{m,i}$. However, this is not the case. In a market scoring rule, only a infinitesimal amount can be purchased at $p_{m,i}$, then the market maker adjusts the price. Section 3.5.2 discussed this property for the LMSR and figure 3.2 showed the price response at different levels of the liquidity parameter b.

The problem is as follows: Kelly suggests to bet a certain fraction of wealth at a certain price, but the trade itself moves the price to a less favorable level. Naturally, Kelly suggest a smaller fraction to bet when being offered a less favorable bet. The market maker keeps adjusting and so does Kelly, until a point is reached where it is no longer rational to bet, either because the price moved beyond the trader’s subjective probability or the Kelly limit is reached. In other words, due to the nature of the automated market maker, the correct Kelly fraction will always be lower than then one suggested by (5.6), except for infinitesimal amounts.

The following section will describe a crude solution to this problem.

5.2 Capped application

A straightforward solution to the problem of adjusting prices is to prevent the algorithm from buying shares at a price above the user’s subjective probability estimate.

Since the solution is more procedural than analytical in nature, it is expressed in pseudocode. Algorithm 5.1 summarizes the approach.

Basically, the algorithm starts with the result of the naive application of the Kelly criterion. However, it includes a “sanity check” (line 3) to determine whether any portion

Algorithm 5.1: Capped application of the Kelly criterion.

Input: A LMSR with quantity vector $\vec{q}$ and liquidity parameter b, a trader with wealth w and subjective belief p regarding outcome o_i

Output: The number of shares Δq_i to buy according to the capped application

$\Delta q_i \leftarrow$ *result form naive application of Kelly*;
$p_m \leftarrow$ *projected market price after buying* Δq_i *shares*;
if $p_m > p$ **then**
  return *number of shares s.t.* $p_m = p$
else
  return Δq_i;
end

of the shares would be bought at an outright unfavorable price. This is the case when the market price after the trade would be above the trader's subjective probability estimate. In this case, the number of shares is capped at the point where $p_m = p$.

Needless to say, this is not an actual application of the Kelly criterion. However, it is comparatively easy to implement and leads to decent results as section 5.5 will show.

Specifically for the LMSR, the number of shares to buy so that $p_m = p$ can be determined by solving the price function for q_i:

$$
\begin{aligned}
p_i &= \frac{e^{q_i/b}}{e^{q_i/b} + \sum_{j \neq i} e^{q_j/b}} \\
\implies q_i &= b \log\left(\frac{p_i}{1-p_i} \sum_{j \neq i} e^{q_j/b}\right)
\end{aligned}
$$

5.3 Non-naive application

The approaches discussed so far, the naive application of the Kelly criterion and the capped application, have taken recourse to the Kelly formula, but they haven't been actual applications of the Kelly criterion. The reason is that they both fail to maximize the user's expected rate of growth, or equally, the expected logarithm of wealth. Maximizing the expected rate of growth is the central idea behind Kelly, which isn't bound to any particular formula.

However, only a slight adjustment to equation 5.6 is necessary to determine the correct Kelly amount to bet in a market scoring rule-based prediction market: The market price $p_{m,i}$ has to be substituted with the price function $p_i(\cdot)$. This leads to

$$\Delta q_i = \begin{cases} w \dfrac{p_{s,i} - p_i(\vec{q} + \Delta q_i \,\vec{1}_i)}{p_i(\vec{q} + \Delta q_i \,\vec{1}_i) - p_i(\vec{q} + \Delta q_i \,\vec{1}_i)^2} & p_{m,i} < p_{s,i} \\ w \dfrac{p_{s,i} - p_i(\vec{q} + \Delta q_i \,\vec{1}_i)}{p_i(\vec{q} + \Delta q_i \,\vec{1}_i)^2} & \text{otherwise} \end{cases} \tag{5.7}$$

where $\vec{1}_i$ is a vector with value 1 at index i and 0 for all other indices, and $\vec{q}$ is the quantity vector of the market maker.

In order to determine the amount of shares to buy, this equation must be solved for Δq_i. Unfortunately, unlike in case of equation 5.6, Δq_i can't be computed directly, as it appears multiple times in the equation. However, an approximation can be found via numerical methods.

5.4 Solution for the LSMR

This section will derive a solution of (5.7) for the LMSR.

5.4.1 Complete equation

The LMSR's price function is given by equation 3.15. Together with definition (5.7) one arrives at the slightly unwieldy

$$\Delta q_i = \begin{cases} w \dfrac{p_{s,i} - \frac{e^{\frac{q_i+\Delta q_i}{b}}}{e^{\frac{q_i+\Delta q_i}{b}} + \sum_{j\neq i} e^{\frac{q_j}{b}}}}{\frac{e^{\frac{q_i+\Delta q_i}{b}}}{e^{\frac{q_i+\Delta q_i}{b}} + \sum_{j\neq i} e^{\frac{q_j}{b}}} - \left(\frac{e^{\frac{q_i+\Delta q_i}{b}}}{e^{\frac{q_i+\Delta q_i}{b}} + \sum_{j\neq i} e^{\frac{q_j}{b}}}\right)^2} & p_{m,i} < p_{s,i} \\ w \dfrac{p_{s,i} - \frac{e^{\frac{q_i+\Delta q_i}{b}}}{e^{\frac{q_i+\Delta q_i}{b}} + \sum_{j\neq i} e^{\frac{q_j}{b}}}}{\left(\frac{e^{\frac{q_i+\Delta q_i}{b}}}{e^{\frac{q_i+\Delta q_i}{b}} + \sum_{j\neq i} e^{\frac{q_j}{b}}}\right)^2} & \text{otherwise} \end{cases} \tag{5.8}$$

It can be brought to a more manageable form using the following identities

$$x := \frac{q_i + \Delta q_i}{b} \tag{5.9}$$

$$r := \sum_{j\neq i} e^{\frac{q_j}{b}} \tag{5.10}$$

as well as using p as a shorthand for $p_{s,i}$. It should not be confused with the price function $p(\cdot)$, though.

$$bx - q_i = \begin{cases} w \dfrac{p - \frac{e^x}{e^x+r}}{\frac{e^x}{e^x+r} - \left(\frac{e^x}{e^x+r}\right)^2} & p_{m,i} < p_{s,i} \\ w \dfrac{p - \frac{e^x}{e^x+r}}{\left(\frac{e^x}{e^x+r}\right)^2} & \text{otherwise} \end{cases} \tag{5.11}$$

5.4.2 Approximation via Newton

A classic method for solving equations that do not have a closed form expression is Newton's method, where the solution is approximated by iterating the following expression over $n = 0, 1, 2, \ldots$ until the desired precision is reached

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{5.12}$$

f' denotes the derivative of f with respect to x.

Assuming $p_{m,i} < p_{s,i}$, the following statements are valid regarding (5.11).

$$\begin{aligned} bx - q_i &= w \frac{p - \frac{e^x}{e^x+r}}{\frac{e^x}{e^x+r} - \left(\frac{e^x}{e^x+r}\right)^2} \\ \implies \quad \frac{bx - q_i}{w} &= \frac{p - \frac{e^x}{e^x+r}}{\frac{e^x}{e^x+r}\left(1 - \frac{e^x}{e^x+r}\right)} \\ \implies \quad 0 &= bx - q_i - w\left[r^{-1}e^x(p-1) + re^{-x}p + 2p - 1\right] \end{aligned}$$

Conversely, assuming $p_{m,i} \geq p_{s,i}$, the following statements can be made.

$$\begin{aligned} bx - q_i &= w \frac{p - \frac{e^x}{e^x+r}}{\left(\frac{e^x}{e^x+r}\right)^2} \\ \implies \quad \frac{bx - q_i}{w} &= \frac{p - \frac{e^x}{e^x+r}}{\left(\frac{e^x}{e^x+r}\right)^2} \\ \implies \quad 0 &= bx - q_i - w\left[re^{-x}(2p-1) + r^2e^{-2x}p + p - 1\right] \end{aligned}$$

Using these results, a function f that has a root that is the solution to equation 5.8 can be defined, under consideration of identities (5.9) and (5.10):

$$f(x) := \begin{cases} bx - q_i - w\left[r^{-1}e^{x}(p-1) + re^{-x}p + 2p - 1\right] & p_{m,i} < p \\ bx - q_i - w\left[re^{-x}(2p-1) + r^2e^{-2x}p + p - 1\right] & \text{otherwise} \end{cases} \tag{5.13}$$

As (5.12) shows, Newton's method requires that f is differentiable. This is the case with (5.13):

$$f'(x) = \begin{cases} b - wr^{-1}e^{x}(p-1) + wre^{-x}p & p_{m,i} < p \\ b + wre^{-x}(2p-1) + 2wr^2e^{-2x}p & \text{otherwise} \end{cases} \tag{5.14}$$

x can be approximated with arbitrary precision using the following iteration sequence:

$$x_{n+1} = \begin{cases} x_n - \dfrac{bx_n - q_i - w\left[r^{-1}e^{x_n}(p-1) + re^{-x_n}p + 2p - 1\right]}{b - w\left[r^{-1}e^{x_n}(p-1) - re^{-x_n}p\right]} & p_{m,i} < p \\ x_n - \dfrac{bx_n - q_i - w\left[re^{-x_n}(2p-1) + r^2e^{-2x_n}p + p - 1\right]}{b + w\left[re^{-x_n}(2p-1) + 2r^2e^{-2x_n}p\right]} & \text{otherwise} \end{cases} \tag{5.15}$$

Finally, in order to start iterating, an initial guess x_0 is necessary. Since the goal is to improve on the naive application of the Kelly criterion, it is a natural choice to determine x_0.

5.4.3 Algorithm

Algorithm 5.2 summarizes the ideas of this chapter in terms of basic operations.

Lines 10 and 11 as well as 14 and 15 deserve special attention. These are assigning functions to f and f' respectively, which is a common practice in functional programming. This is not a necessity, but simplifies the notation of the algorithm. The functions are being evaluated at line 18 and 19.

Lines 1-6 compute an intermediary term $r = \sum_{j \neq i} e^{q_j/b}$. Line 7 computes the instantaneous price of asset i using r, which is equivalent to evaluating equation 3.15.

Lines 9 and 13 are the naive application of the Kelly criterion as in (5.6), which is used as initial approximation.

Finally, lines 18-20 approximate x using Newton's method until the desired precision ε is reached.

This concludes the derivation of the non-naive Kelly criterion for the LMSR. The following section will compare the naive, capped, and non-naive version of the algorithm under various conditions.

Algorithm 5.2: The non-naive kelly criterion for the LMSR

Input: A LMSR with quantity vector $\vec{q}$ and liquidity parameter b, a trader with wealth w and subjective belief p regarding outcome o_i, a desired precision ε

Output: The number of shares Δq_i to buy according to Kelly

$r \leftarrow 0$;
for $j \leftarrow 1$ **to** $|\vec{q}|$ **do**
    if $j \neq i$ **then**
        $r \leftarrow r + e^{q_j/b}$;
    end
end
$p_m \leftarrow e^{q_i/b}/(e^{q_i/b} + r)$;
if $p_m < p$ **then**
    $\Delta q_i \leftarrow w(p - p_m)/(p_m - {p_m}^2)$;
    $f \leftarrow x \mapsto bx - q_i - w\left[r^{-1}e^x(p-1) + re^{-x}p + 2p - 1\right]$;
    $f' \leftarrow x \mapsto b - w\left[r^{-1}e^x(p-1) - re^{-x}p\right]$;
else
    $\Delta q_i \leftarrow w(p - p_m)/{p_m}^2$;
    $f \leftarrow x \mapsto bx - q_i - w\left[re^{-x}(2p-1) + r^2e^{-2x}p + p - 1\right]$;
    $f' \leftarrow x \mapsto b + w\left[re^{-x}(2p-1) + 2r^2e^{-2x}p\right]$;
end
$x \leftarrow (q_i + \Delta q_i)/b$;
while $|f(x)| > \varepsilon/b$ **do**
    $x \leftarrow x - f(x)/f'(x)$;
end
$\Delta q_i \leftarrow bx - q_i$;
return Δq_i;

5.5 Comparison

This section will compare the approaches discussed in this chapter. Simulations will be run to create example histories of the three algorithms discussed in this chapter: the naive application of the Kelly criterion, the capped application and the non-naive application.

It should be noted that these graphs are for illustrative purposes only. The previous sections already established that the non-naive application is the only true application of the Kelly criterion. However, it is of interest to show how the approaches differ under various circumstances.

The simulations are set up as follows: For each simulation, a series of 50 events with fixed outcomes is assumed. At each point, shares are bought according to the three approaches from three separate logarithmic market scoring rules, each with a liquidity parameter of $b = 100$. The bettors each start with an initial wealth $w = 1$ and it is assumed that they know the true probability of each outcome, but not the outcome itself. Figures 5.1a to

5.1f show six such example histories.

As can be seen, the differences between the three strategies are negligible while $w \ll b$. However, as w approaches and surpasses b, the strategies diverge.

While the naive application is prone to betting too much and facing ruin as a consequence, the capped application generally outperforms the non-naive application. This is expected, as higher risks naturally result in higher rewards. However, just as with the naive application, the capped application may face ruin, as is the case in example history 5.1f. As figure 5.1c shows, the non-naive application performs better when being faced with a higher number of losses.

While these graphs can't confirm properties of the non-naive strategy, they can certainly disconfirm claims relating to either the naive or capped approach being without possibility of ruin, because such outcomes have been observed. Hence, the non-naive application is preferable. However, for a practical implementation of a betting system as described in the introduction, many more problems would have to be addressed. These will be discussed in the following chapter.

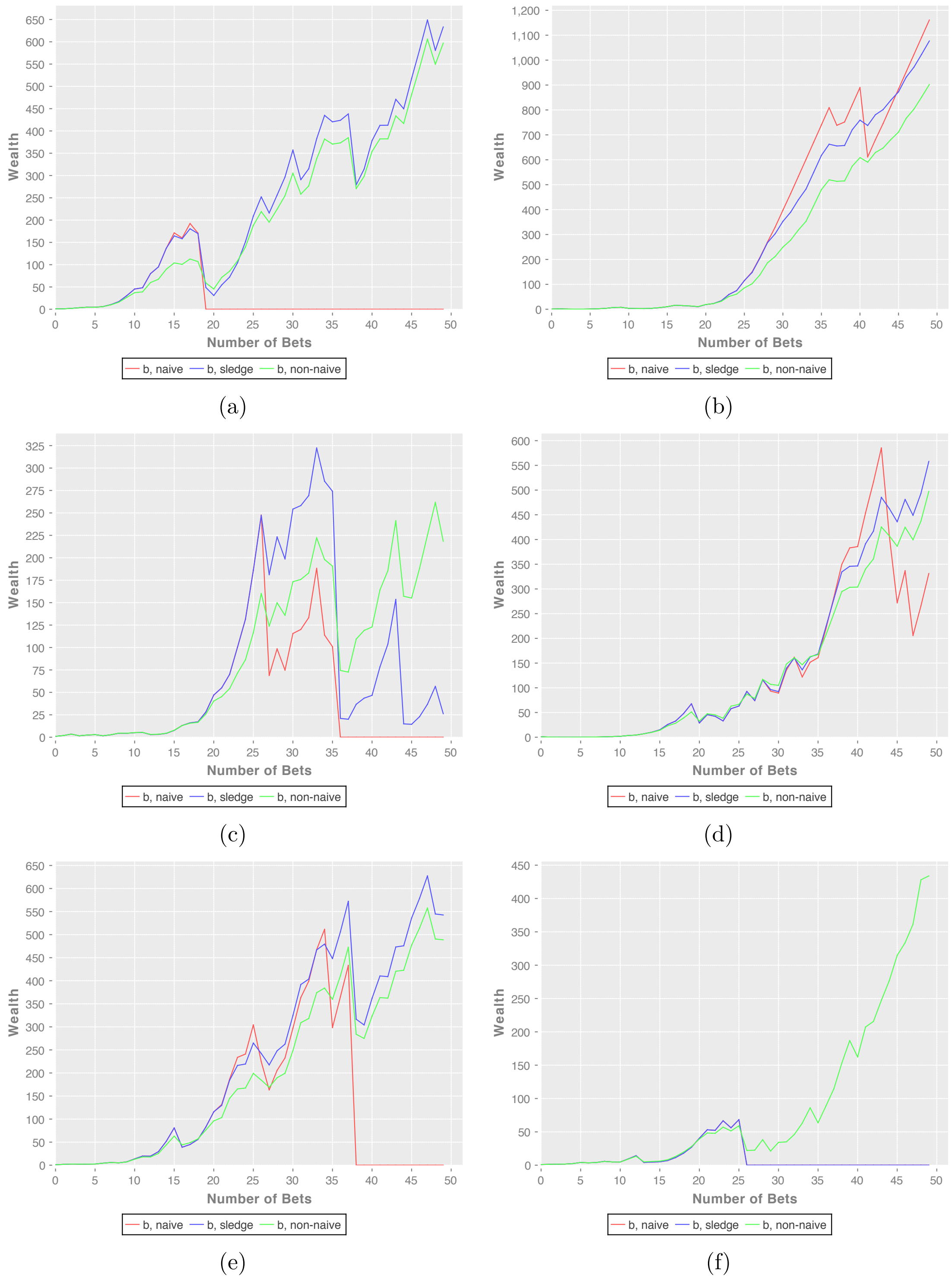


Figure 5.1: Example histories of the naive, capped and non-naive approach.

CHAPTER 6

Conclusion

This thesis introduced an algorithm for the non-naive application of the Kelly criterion for market scoring rules, specifically the LMSR. It allows users to specify their subjective probability estimate for one of the possible outcomes. The algorithm then computes the exact number of shares to buy or sell according to the Kelly criterion, and under consideration of the price change of the trade itself. The algorithm can be used to enable a simplified user interface for prediction markets, one that does not ask users to make trades themselves.

The remainder of this chapter will discuss the technical and practical limitations of the algorithm, and if possible provide recommendations on how these could be addressed.

6.1 Technical limitations

This section will discuss various technical limitations of the algorithm, as opposed the the more open-ended considerations discussed in the next section.

6.1.1 Multi-outcome markets

The algorithm is applicable to prediction markets that offer one degree of freedom, i.e. binary prediction markets. However, for markets with three or more possible outcomes, a direct application of the method is not always possible (it is useable as long as the user wants to bet on no more than one outcome).

Suppose a user has subjective estimates for more than one outcome, which are different from current market prices. Applying the method one asset at a time will result in different trades for different permutations. This is due to each trade affecting the prices of all assets. The method could be generalized, so that it allows the user to report a complete probability distribution, which would require to solve a more complex system of equations.

6.1.2 Changing prices

As with all markets, prices change continuously. This can lead to situations where a bettor bought shares at one point, and is being offered a more favorable bet later. A subsequent application of the Kelly criterion results in a higher fraction assigned to the bet than when being offered the more favorable bet from the beginning. This is known as "Proebsting's paradox" [25].

However, the resolution is straightforward (in fact there is no paradox): The goal of Kelly is always to maximize the expectation of growth, so the bettor has to reverse the previous bet, then calculate the new fraction based on the new condition. In the context of market scoring rules, this is possible at no additional cost.

This solution could be interesting in combination with an implementation of limit orders [12].

6.2 Practical considerations

Other than the strictly technical limitations discussed in the previous section, there are number of practical considerations that are not easily modeled mathematically, and generally don't have an obvious answer.

6.2.1 Time value

The algorithm is missing a notion of time value. Unlike in the simulation, the outcomes of real events aren't immediately known, rendering shares that are closer to their maturity date more valuable. It would be in the user's interest to lower the bet sizes for trades with maturity dates further in the future.

6.2.2 Order matters

Related to the time value is the order in which users input their estimates. Generally, the algorithm will assign higher bet sizes to the earlier predictions, as the wealth of the user shrinks with each bet.

6.2.3 Fractional Kelly

In practice it is not recommended to bet the full Kelly amount, instead it should be considered an upper bound that must not be crossed. A more forgiving version of the algorithm would only bet a fraction of the amount recommended by Kelly. This gives some leeway to estimation errors, which can't be fully avoided in practice.

6.2.4 Differing utility functions

As mentioned earlier, Kelly maximizes the expected log utility of money. This is only one of an infinite amount of possible utility functions, any of which may better fit the profile of a particular user. Kelly is considered ideal because it achieves the highest possible wealth in the asymptote. However, since users live in the pre-asymptote this might be of limited interest. An improved version of the algorithm might be based on a different utility function.

6.2.5 Unsophisticated users

Obviously, Kelly can only produce good outcomes if the user has an edge. Users' probability estimates might be honest, but they needn't be correct. The proposed method does not guarantee gains if users are not well calibrated, it merely abstracts away the intricacies of trading. As a result, unsophisticated traders will lose influence over time (though Kelly prevents complete ruin) and might do so faster than if asked to make bets manually.

It should be noted that this is not necessarily problematic and could even be desired. After all, at least in some theoretical models, the evolutionary pressures on bad actors are supposed the be the reason for the accuracy of the market.

List of Figures

List of Tables

List of Algorithms

Acronyms

CDA Continuous Double Auction. 5–7, 9

HSX Hollywood Stock Exchange. 9

IEM Iowa Electronic Markets. 3, 4

LMSR Logarithmic Market Scoring Rule. 2, 10, 18, 19, 31–33, 35, 36, 39, 45

Bibliography

[1] Kenneth J Arrow, Robert Forsythe, Michael Gorham, Robert Hahn, Robin Hanson, John O Ledyard, Saul Levmore, Robert Litan, Paul Milgrom, Forrest D Nelson, et al. The promise of prediction markets. *SCIENCE-NEW YORK THEN WASHINGTON-*, 320(5878):877, 2008.

[2] Joyce Berg, Forrest Nelson, and Thomas Rietz. Accuracy and forecast standard error of prediction markets. *Tippie College of Business Administration, University of Iowa*, 2003.

[3] Alina Beygelzimer, John Langford, and David M Pennock. Learning performance of prediction markets with kelly bettors. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 3*, pages 1317–1318. International Foundation for Autonomous Agents and Multiagent Systems, 2012.

[4] Glenn W Brier. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3, 1950.

[5] Yiling Chen, Ian A Kash, Michael Ruberry, and Victor Shnayder. Eliciting predictions and recommendations for decision making. *ACM Transactions on Economics and Computation*, 2(2):6, 2014.

[6] Yiling Chen and David M Pennock. A utility framework for bounded-loss market makers. *arXiv preprint arXiv:1206.5252*, 2012.

[7] Paul Gomme. Iowa electronic markets. *Federal Reserve Bank of Cleveland*, 2003.

[8] Robin Hanson. Combinatorial information market design. *Information Systems Frontiers*, 5(1):107–119, 2003.

[9] Robin Hanson. Logarithmic market scoring rules for modular combinatorial information aggregation. *The Journal of Prediction Markets*, 1(1):3–15, 2012.

[10] Robin Hanson and Ryan Oprea. Manipulators increase information market accuracy. *George Mason University*, 2004.

[11] Friedrich August Hayek. The use of knowledge in society. *The American economic review*, pages 519–530, 1945.

[12] Hoda Heidari, Sébastien Lahaie, David M Pennock, and Jennifer Wortman Vaughan. Integrating market makers, limit orders, and continuous trade in prediction markets. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, pages 583–600. ACM, 2015.

[13] Jane Hung. Betting with the kelly criterion. 2010.

[14] John Kelly. A new interpretation of information rate. *IRE Transactions on Information Theory*, 2(3):185–189, 1956.

[15] Leonard C MacLean, Edward O Thorp, and William T Ziemba. Good and bad properties of the kelly criterion. *Risk*, 20(2):1, 2010.

[16] Abraham Othman. Zero-intelligence agents in prediction markets. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 2*, pages 879–886. International Foundation for Autonomous Agents and Multiagent Systems, 2008.

[17] Abraham Othman, David M Pennock, Daniel M Reeves, and Tuomas Sandholm. A practical liquidity-sensitive automated market maker. *ACM Transactions on Economics and Computation*, 1(3):14, 2013.

[18] Paul Rhode and Koleman Strumpf. Historical prediction markets: Wagering on presidential elections. *Journal of Economic Perspectives*, 18(2):127–42, 2004.

[19] Leonard J Savage. Elicitation of personal probabilities and expectations. *Journal of the American Statistical Association*, 66(336):783–801, 1971.

[20] Emile Servan-Schreiber, Justin Wolfers, David M Pennock, and Brian Galebach. Prediction markets: Does money matter? *Electronic markets*, 14(3):243–251, 2004.

[21] Edward O Thorp. Fortune's formula: the game of blackjack. *Notices of the American Mathematical Society*, 7(7):935–936, 1960.

[22] Edward O Thorp. Beat the dealer. Random, 1966.

[23] Edward O Thorp. Optimal gambling systems for favorable games. *Revue de l'Institut International de Statistique*, pages 273–293, 1969.

[24] Edward O Thorp. The kelly criterion in blackjack, sports betting, and the stock market. *Finding the Edge: Mathematical Analysis of Casino Games*, 1(6), 1998.

[25] Edward O Thorp. Understanding the kelly criterion. *The Kelly Capital Growth Investment Criterion: Theory and Practice. World Scientific Press, Singapore*, 2010.

[26] Edward O Thorp and Sheen T Kassouf. Beat the market. *New York: Random*, 1967.

[27] Ben Tilly. Kelly criterion in detail. https://web.archive.org/web/20160718060755/http://www.elem.com/~btilly/kelly-criterion/, 2013.

[28] Leighton Vaughan Williams. *Prediction markets: Theory and applications*, volume 66. Routledge, 2011.

[29] Robert L Winkler. Scoring rules and the evaluation of probability assessors. *Journal of the American Statistical Association*, 64(327):1073–1078, 1969.

[30] Justin Wolfers and Eric Zitzewitz. Prediction markets. *The Journal of Economic Perspectives*, 18(2):107–126, 2004.

[31] Justin Wolfers and Eric Zitzewitz. Interpreting prediction market prices as probabilities. Technical report, National Bureau of Economic Research, 2006.