Cumulative Prospect Theory

Timothy Schwieg

April 5, 2018

1 Introducton

A common criticism of Neo-Classical Economics is the structure that is placed on actions taken by agents under uncertainty. Often times we assume that agents act to optimize expected utility, where the utility is characterized by a Von-Neumann Morgenstern utility function. This has many attractive properties, but can lead to situations where the structure is too rigid to allow for behavior that is commonly observed.

One such example is lotteries. People who are generally risk-averse in areas such as insurance or other places act as if they are risk-loving in playing a lottery. One model that is able to account for this is Cumulative Prospect theory. It incorporates several important concepts not seen in Classical decision making under risk.

It incorperates reference dependence, which implies that people view things in the context of losses and gains rather than changes to their overall wealth. This is attractive for computational reasons. It also utilizes loss aversion, the notion that losses are relatively more costly than gains. The model also incorperates diminishing sensitivity, the notion that valuations are concave in losses and convex in gains. The final concept is probability weighting.

The nature of probability weighting leads to overweighting of the tails of the distributions, leading to behavior such as preferring a 0.001 chance of 5000$ to a certain gain of 5$, and prefer a certain loss of 5$ to a .001 chance of losing 5000$. As noted by Barberis **?**, there has been more sophisticated approached to the application, however this investigation will be limited to the relatively simple function forms suggested first by Kahnemann and Tversky **?**. This has been chosen due to the computationally ardous estimation method utilized.

In this paper I intend to apply Cumulative Prospect theory to small lotteries that take place within the video game Counter Strike: Global Offensive. In this game players recieve "loot boxes", which are cases that can be opened with the purchase of a key, that return a random reward based on some predefined probabilities.

A very attractive feature of this data is that both the contents of the lottery and the lotteries themselves are both traded on the Steam Community Market. While the API does not provide flawless data, there is still a large amount of data. This comes in two forms: Market history given to the last hour, and Buy and Sell Orders that have not currently been fulfilled.

I intend to test

2 Model

2.1 General Approach

Because of the nature of the data, as well as the complicated structure that Cumulative Prospect theory places upon the valuation function, it is difficult to use it within the structure used commonly in double auction literature. Almost all models of auctions specify a structure between the valuation and the price that takes the form of the valuation is the valuation for the item minus the price paid for the item at the auction. This difference cannot be reconciled with the framing effects required by Cumulative Prospect Theory.

To circumvent this issue, I use the result found in **?** which states that a continuous double auction converges quickly, in order $O(N^2)$ to a competitive market. From this, I can consider that in large double auctions, such as is the case for the purchases of the lotteries in question. From this we can conduct our analysis as if a buyer of a lottery is at a competitive market, sees the price given by the market and makes a decision choice based upon his valuation, as a function of the price.

This leads us to one of two outcomes in the market, either there is a purchase made, indicating a valuation above the price seen by the consumer, or he leaves a buy order, indicating a price he would be willing to pay. Since in a double auction where the buyer pays the seller's price, it is dominant to tell the truth I will consider these buy orders as the actual valuations of the consumer **?**.

2.2 Discrete Choice

By Following the discrete choice model, we assert that decisions made by the individual are made such that if the valuations of lottery plus ξ_l is greater than the valuation of not purchasing which is given by ξ_n where $\xi_\mathrm{j} \sim$ Gumbel. This means that the purchase is dictated by if: $V(f)+\xi_b-\xi_n > 0$. Since the difference of two Gumbel distribution is distribution logistically, and V(f) is not random. Knowing that the Logistic Distribution is in the location scale family, we may consider the entire expression as: $V_i(f) \sim logit(V(f), s)$.

However, we observe some censored realizations of V. That is, we observe valuations based on the buy orders which are non-censored realizations, but we also observe purchases which are censored. Denote the censored purchases by $d_n = 1$ and uncensored by: $d_n = 0$. The likelihood function for censored data as shown by **?** is:

$$\sum_{j=1}^{J}\sum_{n=1}^{N_j} d_{n,j}\log(\frac{1}{4s}sech^2(\frac{x_{n,j}-V_j}{2s})) + (1-d_{n,j})\log(\frac{\exp(\frac{x_{n,j}-V_j}{s})}{1+\exp(\frac{x_{n,j}-V_j}{s})}) \quad (1)$$

Where V_j is the valuation for the box of type j, $\mathrm{x_{n,j}}$ is the price paid by observation n of type j, and $\mathrm{d_{n,j}}$ is whether or not the data at observation n of type j was censored.

We may maximize this likelihood to find the parameters that best fit. What remains to be decided is the form of the function V. We will have to assume a structural form for the function, and while we will be able to test between structural forms using a validation set, there will still be assumptions held.

2.3 Cumulative Prospect Theory

Since one key aspect of cumulative prospect theory is that we are more strongly motivated by losses than by gains, it is immediately obvious that the value function of the contents of the lottery will not be symmetric, and the easiest way to handle this will be to estimate two separate functions, one of which is used to evaluate losses, and one of which estimates gains. By applying a piece wise function where the loss function is used on losses, and the gain function for gains, we arrive at a continuous function (since both are zero at zero) which we can use. The question of whether or not loss aversion is displayed is one for the empirics.

Since I am following the Discrete Choice model, the decision to purchase a create will be driven by the valuations of the crate against the alternative which is buying nothing. Since Cumulative prospect theory functions by looking at deviations from a reference point, which we will use as the price of the box combined with the costs of opening it (the key).

Following the Notation of **?**, we represent the valuation of the lottery as $V(f) = \sum_{i=-m}^{n} \pi_i v(x_i)$ where v is a function that is convex in losses and concave in gains. π is a weight function that will be defined in a later section.

The question of a reference point is commonly debated, however in this application it is clear that the reference point for the crate is the cost of opening such a crate, which is the sum of the "key" that must be purchased from Valve in order to open, and the price paid for the crate. The cost of a key is constant, and marks the direct cost of participation, they are very often purchased directly before opening the crate, as when you attempt to open it, you are prompted with the price and a link to purchase a key. I believe that it is therefore reasonable to ignore all effects time play play on the problem, and treat the issue as if the consumer simply buys the key and crate at the same time as he opens it.

2.3.1 Valuations

The question of how do we measure the gains of the lottery now looms. Since each box when opened contains an item that has a particular value to the consumer who opened it, and is unobserved, the only thing that can be observed is the market price of the item over time. Identification of buyer and seller valuations, even in a simplified situation where there is only one buyer and one seller, still requires more information than we are given. As shown in **?**, for their identification strategy, all the bids in the auction are required, and for the strategy shown in **?**, there must exist exclusive covariates that shift only one trader's value distribution, which are not given by the data.

Absent an ability to identify the valuations of the specific losses and gains in the lottery, an identifying assumption will have to be made. I will represent valuations of the contents of the lottery as the weighted average of the purchases made, weighted by the quantity purchased. This is effectively the sample mean of prices purchased, which I will assume to be average valuations of the good.

The valuations of the $\mathrm{i^{th}}$ possible element of the lottery will be given by: v($\mathrm{p_i}$ - $\mathrm{p_l}$ - $\mathrm{p_{key}}$), where $\mathrm{p_i}$ is the average price of the $\mathrm{i^{th}}$ element at market, $\mathrm{p_l}$ is the price paid for at market for the lottery, and $\mathrm{p_{key}}$ is the price of the key required to open the lottery. Depending on whether or not this difference is positive or negative will result in different functions being used to evaluate the valuation. Using the specification suggested by **?**:

$$v(x) = \begin{cases} x^{\alpha} & x \geq 0 \\ -\lambda(-x)^{\alpha} & x < 0 \end{cases} \tag{2}$$

2.3.2 Probability weighting function

In Cumulative Prospect theory, the cumulative mass (distribution) function is weighted such that individuals overweight the tail probabilities. This is especially important in this model, as there are many high valued rare items, that if this part of the theory is correct, heavily influence the valuation of the box, despite their extremely low probability of occurrence.

Again, I will use the specification suggested by **?**, and use the cumulative transformation function of:

$$w(P) = \frac{P^{\delta}}{(P^{\delta} + (1-P)^{\delta})^{\frac{1}{\delta}}} \tag{3}$$

To define the decision weights π_{i}, we must first order the prospects of the lottery in ascending order of gains. the weight π_{i} then is defined by:

$$\pi_i = w(\sum_{j=-m}^{i} P(x_j)) - w(\sum_{j=-m}^{i-1} P(x_j)) \tag{4}$$

2.3.3 The Valuation function

From these, we can create a valuation function for an individual facing a particular lottery:

$$V_i = \begin{cases} (w(\sum_{j=-m}^{i} P(x_j)) - w(\sum_{j=-m}^{i-1} P(x_j)))(p_i - p_l - p_{key})^{\alpha} & (p_i - p_l - p_{key}) \geq 0 \\ -\lambda(w(\sum_{j=-m}^{i} P(x_j)) - w(\sum_{j=-m}^{i-1} P(x_j)))(p_l + p_{key} - p_i)^{\alpha} & (p_i - p_l - p_{key}) < 0 \end{cases} \tag{5}$$

3 Data

3.1 The Data

The data is pricing data of items on the Steam Community Market for the game Counter Strike: Global Offensive. Players in game earn items random that they can sell on the market or open themselves. However most rare items are earned via opening of dropped "loot boxes" that are then opened by players via purchasing of a key. These boxes can be earned by playing or received randomly from players who are watching games of professionals play. The probabilities of the drops are not known or even estimated well, as they change depending on many factors including time.

However, once a box has been obtained, the probability of receiving an item is well documented as required by Chinese Law. Each item has a certain grade of rarity, for example the Ak-47 Redline has a rarity level of Classified which means that there is a 3.2% chance of receiving a Classified item in the crate. All Classified items contained in the crate have the same probability of being dropped by the crate.

However there are many variants of each item. Each item has a quality ascribed to it, the float of the item. This describes the wear on the item, and is distributed uniformly on the interval 0-1. On the market the items are split into intervals: Battle-scared, well-worn, field-tested, minimal wear and factory new. Each quality is a separate listing on the market with a separate price. In addition to each item having a quality type there is also a 10% chance of each item being labeled as StatTrak, which also distinguishes the value of a weapon. This means that each item has 10 possible different variations all with different probabilities of being obtained. Some rare items, usually knives and gloves may have more or less variants, but the amount and probabilities are known, and can easily be determined by checking if there is a market history for the item.

The probabilities for each condition are as follows:

Float	Condition
0.00 - 0.07	Factory New
0.07 - 0.15	Minimal Wear
0.15 - 0.38	Field-Tested
0.38 - 0.45	Well-Worn
0.45 - 1.00	Battle-Scarred

Each item has a 10% chance of being StatTrak if that item has statTrak enabled. float values are distributed uniformly, making the probability calculations easy.

However the rarity of a skin also controls its probability of being dropped in a particular lottery. These rarities are set by Valve, and are specified for each crate. They rank from gold (very rare) to blue (not very rare) The probabilities of getting an item of a rarity is given as follows:

Probability	Rarity
.0026	Special (gold)
.0064	Covert (red)
.032	Pink (Classified)
.1598	Purple (Restricted)
.7992	Blue (Mil-spec)

In each box there are several items of each rarity, each one is equally likely to be found when the lottery is explored.

Each box contains some subset of these items that is known, and the market value of each item at a particular time period is also known, so the expected value, or any other modified version of a valuation of the lottery can easily be calculated.

3.2 Sources of the Data

The data has been mined from the steam community market api, which provides a purchase history for every item on the market, down to the hour for the last thirty days and daily for the rest of the lifetime of the item. It does not provide a record of every purchase, just the quantity sold in that time period as well as the median price they were sold at. Obviously this is less than ideal, but I believe it will cause less problems than the inaccuracies introduced by the market only working in one cent intervals.

Also available is current buy and sell orders for each item. If a potential buyer wishes to buy on this market, he may either select a box directly and purchase from a particular seller, or he may put forward a buy order, which he stipulates a price, and as soon as a seller puts an item up for sale below that price, it is sold to the buyer, and he is charged the seller's price. This gives the valuations of people who have not yet obtained the item directly. However, it does not appear that there is a history available for these items. In some ways this is beneficial because it would impossible to determine the differences between buy orders that were fulfilled and buy orders that were removed because of changes in the prices of underlying assets. I have decided to treat all outstanding buy orders as valuations in the final time period that simply are below the market price. I will not consider the case that there are buy orders placed and forgotten about.

3.3 Treatment of the data

There are aproximately 11,000 items mined through the procedure followed by the script `BuildData.py`. This data must be organized so that it can be used effectively. First a hierachical file structure was created by `MoveFiles.py`, this sorted each item by its type, skin, and finally quality. However, To this end, I created text files that contained the contents of each lottery that was to be examined, and then aggregated this data using `rarity.py`. This aggregation included the different varieties of items included in each box (condition and StatTrak), controlling for availibility of items by searching for them in the file structure.

Once each individual lottery had the available items, the actual price data that had been mined by `BuildData.py` could be applied. Using the script `CreateData.py`, for each transaction of the lottery that was recorded during the period where there is hourly data, the price and quantity was noted, as well as the most current price of each item contained in the lottery. This data was combined with the probabilities of each item being drawn, as well an indicator variable for whether or not this data was censored or not. All data that was drawn from the market purchases is considered censored, and buy order data was considered uncensored. This data is finally saved in csv format for each lottery, containing probability data, price history of the box as well as the probablity of obtaining each item in the lottery.

4 Computations

4.1 Problems

There are two problems that prevent this problem from being calculated easiy, the first is the presense of censored data, which make estimation difficult by providing a non-convex liklihood function. This problems is further complicated by Cumulative Prospect theory, which specifies that consumers valuations are convex in gains, and concave in losses, and a weighted linear combination of these valuations which is in general not a convex optimization problem. Since we are attempting to estimate the shape of this function, there is no manner in which this problem can be couched as a convex optimization problem. This means that our problem lies in the NP-Hard class, and it will be very difficult to apply all of our data to the problem.

4.2 The approach

Since there is such a computation load, the programming language julia will be used. I chose to use the mathematical programming interface JuMP **?**, as it allows for easy interfacing to solvers. The solver Ipopt was used for all calculations becuase of the non-convex nature of the problem **?**.

4.3 Calculations

Because of the large amount of data and complicated operations applied to the problem, automatic differentiation was unable to produce derivative values for even moderate portions of the data. As a result, we will have to provide derivatives of the likihood function. This will have to be broken down into parts due to the complex nature of the problem.

There are four parameters of interest to the problem, so we need only to calculate the gradient for the four parameters. It can also be noted that s is only present in the liklihood function, not in the functional forms provided for Cumulative Prospect theory, and that the other paramters only enter the model in the Valuation of the lottery, not in the distribution or density function that make up the liklihood function. Using the chain rule we can see that:

$$\frac{\partial L}{\partial \alpha} = \frac{\partial L}{\partial V}\frac{\partial V}{\partial \alpha} \quad \frac{\partial L}{\partial \lambda} = \frac{\partial L}{\partial V}\frac{\partial V}{\partial \lambda} \quad \frac{\partial L}{\partial \delta} = \frac{\partial L}{\partial V}\frac{\partial V}{\partial \delta}$$

4.3.1 Liklihood Function Derivatives

Our liklihood function is given by:

$$\sum_{j=1}^{J}\sum_{n=1}^{N_j} d_{n,j}\log(\frac{1}{4s}sech^2(\frac{x_{n,j}-V_j}{2s})) + (1-d_{n,j})\log(\frac{\exp(\frac{x_{n,j}-V_j}{s})}{1+\exp(\frac{x_{n,j}-V_j}{s})})$$

Taking the derivative with respect to s gives us:

$$\frac{\partial L}{\partial s} = \sum_{j=1}^{J}\sum_{n=1}^{N_j} d_{n,j}\frac{-1}{s^2}\left(2s - \frac{1}{2}(v-x)\tanh\left(\frac{v-x}{2s}\right)\right) \\ +(1-d_{n,j})\frac{(v-x)\,e^{\frac{1}{s}(v-x)}}{s^2\left(e^{\frac{1}{s}(v-x)}+1\right)}$$

Taking the derivative with respect to v gives us:

$$\frac{\partial V}{\partial V} = \sum_{j=1}^{J}\sum_{n=1}^{N_j} d_{n,j} - \frac{1}{2s}\tanh\left(\frac{v-x}{2s}\right) - (1-d_{n,j})\frac{e^{\frac{1}{s}(v-x)}}{s\left(e^{\frac{1}{s}(v-x)}+1\right)}$$

4.3.2 Valuation Function Derivatives

As we can see from 5, the valuation function is a peicewise function, and it is not immedietly obvious that it is differentiable. However it can be rewritten by partitioning the x values based upon whether or not they are gains or losses. That is whether or not $p_i - p_l - p_{key}$ is positive or negative. We will consider the set S to be the set of x values for which $p_i - p_l - p_{key}$ is negative, and the set $\bar{S}$ to be the set of x values for which $p_i - p_l - p_{key}$ is weakly positive.

We may now write our valuation function as

$$V_i = \sum_{i \in S} \pi_i (p_i - p_l - p_k ey)^\alpha - \sum_{i \in \bar{S}} \pi_i \lambda (p_l + p_{key} - p_i)^\alpha \tag{6}$$

Since $p_i - p_l - p_{key}$ is independent of α, δ, λ the sign does not change at any point during the calculations, so this summation will be along the same sets for all iterations, and this is a continously differentiable function in α, δ, λ.

We may now take the derivatives with respect to the parameters:

$$\frac{\partial V_i}{\partial \alpha} = \sum_{i \in S} \pi_i (p_i - p_l - p_{key})^\alpha \log(p_i - p_l - p_{key}) - \sum_{i \in \bar{S}} \pi_i \lambda (p_l + p_{key} - p_i)^\alpha \log(p_l + p_{key} - p_i)$$

$$\frac{\partial V_i}{\partial \lambda} = - \sum_{i \in \bar{S}} \pi_i (p_l + p_{key} - p_i)^\alpha$$

$$\frac{\partial V_i}{\partial \delta} = \sum_{i \in S} \frac{\partial \pi_i}{\partial \delta} (p_i - p_l - p_k ey)^\alpha - \sum_{i \in \bar{S}} \frac{\partial \pi_i}{\partial \delta} \lambda (p_l + p_{key} - p_i)^\alpha$$

4.3.3 Probability Weighting Function Derivatives

Since it is known that $\pi_i(x, \delta) = w(\sum_{j=-m}^{i} P(x_j), \delta) - w(\sum_{j=-m}^{i-1} P(x_j), \delta)$, then

$$\frac{\partial \pi_i}{\partial \delta}(x, \delta) = \frac{\partial w}{\partial \delta}(\sum_{j=-m}^{i} P(x_j)) - \frac{\partial w}{\partial \delta}(\sum_{j=-m}^{i-1} P(x_j))$$

$$\frac{\partial w}{\partial \delta}(p, \delta) = \frac{p^d}{\delta^2} \left(p^\delta + (1-p)^\delta\right)^{-\frac{1}{\delta}(\delta+2)} \left(\delta^2 \left(p^\delta + (1-p)^\delta\right)^{\frac{1}{\delta}(\delta+1)} \log(p) + \left(p^\delta + (1-p)^\delta\right)^{\frac{1}{\delta}} \right.$$
$$\left. \left(-\delta \left(p^\delta \log(p) + (-p+1)^\delta \log(-p+1)\right) + \left(p^\delta + (-p+1)^\delta\right) \log\left(p^\delta + (-p+1)^\delta\right)\right)\right)$$

As we can see, the complexity of this problem quickly becomes staggering, and for the large amount of data gathered, it will become extremely difficult to reach estimates for large samples.

4.4 Further Computational Considerations

Many of these lotteries are trading at the minimum possible price that the market allows, and as such there are no outstanding buy orders for these lotteries.

As a result of this, I will attempt to limit myself to lotteries that have outstanding buy orders, reducing the set of possible lotteries to examine to: Chroma Case, Operation Hydra Case, Operation Vanguard Case, Huntsman Weapon Case, CS:GO Weapon Case, CS:GO Weapon Case 2, CS:GO Weapon Case 3, and eSports 2013 Case.

5 Results

5.1

6 Reading list