

Comparing Online Learning Algorithms to Stochastic Approaches for the Multi-Period Newsvendor Problem

Shawn O’Neil

Amitabh Chaudhary

Abstract

The multi-period newsvendor problem describes the dilemma of a newspaper salesman—how many papers should he purchase each day to resell, when he doesn’t know the demand? We develop approaches for this well known problem based on two machine learning algorithms: Weighted Majority of Warmuth and Littlestone, and Follow the Perturbed Leader of Kalai and Vempala. With some modified analysis, it isn’t hard to show theoretical bounds for our modified versions of these algorithms. More importantly, we test the algorithms in a variety of simulated conditions, and compare the results to those given by traditional stochastic approaches which assume more information about the demands than is typically known. Our tests indicate that such online learning algorithms can perform well in comparison to stochastic approaches, even when the stochastic approaches are given perfect information.

1 Introduction

On each morning of some sequence of days, a newspaper salesman needs to decide how many newspapers to order at a cost of c per paper, so that he can resell them for an income of r per paper. Unfortunately, every day it is unknown how many papers d will be demanded. If too many are ordered, some profits are lost on unused stock. If too few are ordered, some profits are lost due to unmet demand. The actual profit seen by a vendor who orders x items on a day with demand d is given by $r \min\{d, x\} - xc$. The papers ordered for a single day are of course only useful for that day; leftover papers cannot be sold in any later period.

This model describes a wide variety of products in industry. Fashion items and the trends they rely on are typically short lived, inducing many manufacturers to introduce new product lines every season[16]. Consumer electronics also have a short selling season due to their continuously evolving nature; cellular phones can have a lifecycle as short as six months[2]. Some vaccines such as those for influenza are only useful for a single season[6].

For many such products, due to required minimum manufacturing or processing times, the vendor must finalize his order before any demand is seen. Further, because properties of the products themselves can vary

markedly between selling periods, so too can the demand seen each period. This *demand uncertainty* is the most challenging hallmark of the newsvendor model.

A common approach taken to resolve the demand uncertainty issue is using a stochastic model for the demands; assuming, for example that for each period the demand is drawn independently from some known distribution. In using such an approach, the goal is then to choose an order amount which maximizes expected profit (see, e.g., [11]). However, such approaches are commonly inadequate, as the quality of the final result depends heavily on the quality of the assumptions made about the distribution. Given the strong uncertainty inherent in many newsvendor items, such quality is usually low. (See [21] for a lengthier discussion on the shortcomings of this approach.)

Alternate approaches to the newsvendor problem are more “adversarial” in nature. In these models, very little is assumed about the nature of the demands, and worst-case analysis is used. Typically, only a lower bound m and upper bound M on the range of possible demand values are assumed. One solution in this area develops a strategy to minimize the *maximum regret*:

$$\max_{\text{demand values}} (\text{OPT} - \text{ALG}),$$

where OPT denotes the profit of the *offline optimal* algorithm which knows the demand values, and ALG is the profit of the strategy used (see [17, 21, 22]).

Another method used to evaluate and design online algorithms for such problems is competitive ratio, where the goal is to minimize the ratio OPT/ALG in the worst case. However, one can show, using Yao’s technique, a lower bound of $\Omega(M/(mk))$ for this ratio in the single period case when $r = kc$. This bound is tight, as a simple balancing algorithm can guarantee profits of this form.

Similarly restrictive results can be found for the worst case approach to regret with respect to OPT seen above. The single period *minimax regret* solution results in a maximum regret of $c(M - m)(r - c)/r$ [21], which implies that for t periods of a newsvendor game it is possible to suffer a regret of $tc(M - m)(r - c)/r$, even for the best possible deterministic algorithm.

For these reasons, we turn away from evaluating the performance of algorithms in terms of the *dynamic* offline optimal, and consider a more realistic target: the *static* offline optimal, which we denote here by **STOPT**. **STOPT** is a weaker version of **OPT** which makes an optimal decision based on perfect knowledge of the demands, but is required to choose one single order quantity to use for all periods.

Comparing the performance of algorithms with the performance of **STOPT** has practical significance, because any bounds for an algorithm with respect to **STOPT** also hold with respect to an algorithm which makes decisions based on stationary stochastic assumptions. Much of the inventory theory literature deals with algorithms of this type[15, 11].

We look at adaptations of two Expert Advice algorithms: Weighted Majority, developed by Littlestone and Warmuth[14], and Follow the Perturbed Leader, developed by Kalai and Vempala[12].

In the expert advice problem, the algorithm designer is given access to n experts, each of whom make a prediction for each period, and suffer some cost for incorrect predictions. The goal is to design an algorithm that makes its own predictions based on the experts' advice, and yet does not suffer much more cost than the best performing expert in hindsight.

In our setting, we use naive experts which make fixed predictions in the range $[m, M]$, and the cost they suffer in each period is the regret (difference in profit) from the dynamic offline **OPT**. Adapting the Weighted Majority algorithm to the non linear profit function of the newsvendor problem requires some careful attention if one wants to show theoretical performance bounds, whereas Follow the Perturbed Leader is a more straightforward implementation. Details of the algorithms' operation and theoretical performance bounds in this setting can be found in the appendices.

2 Goals of This Paper

In Section 4, we'll give overviews of the operation of three algorithms, two based on Weighted Majority variants which we call **WMN** and **WMNS**, and one based on Follow the Perturbed Leader which we call **FPL**. Each of these algorithms takes parameters which are chosen by the experimenter as input, which affect their operation and the performance bounds they achieve.

The primary interest of this paper, then, is to empirically evaluate the performance of these algorithms and compare the results to those generated by **STOPT** as well as more traditional stochastic approaches. Each of the stochastic solutions takes as input the assumptions made by the experimenter about the mean and standard deviation of the input distribution.

Further, the specifics of the problem instance itself may lead to interesting observations about all of the solutions specified. For instance, we know that the relationship of r and c can make a large difference on the performance of the minimax regret solution; does this ratio also affect the performance of other approaches we are going to test? Do certain types of input distributions favor one approach over the other?

Given such a large number of possible experimental variables, we are forced to select those which we believe will be most interesting, and design experiments using simulated data which are most likely to highlight the advantages and deficiencies of the different approaches.

3 Related Work

The Newsvendor Problem The origins of the newsvendor problem can be traced as far back as Edgeworth's 1888 paper[10] in which the author considers how much money a bank should keep in reserve to satisfy customer withdrawal demands, with high probability. If the demand distribution and the first two moments are assumed known (normal, log-normal, and Poisson are common), then it can be shown that the expected profit is maximized at x , where $\phi(x) = (r - c)/r$ and $\phi(\cdot)$ is the cumulative probability density function for the distribution. Gallego's lecture notes[11] as well as the book by Porteus[15] have useful overviews. When only the mean and standard deviation are known, Scarf's results[18] give the optimal stocking quantity which maximizes the expected profit assuming the worst case distribution with those two moments (a *maxi-min* approach). In some situations this solution prescribes ordering no items at all.

Among worst-case analyses, one of the earliest uses of the minimax regret criterion for *decision making under uncertainty* was introduced by Savage[17]. Applying the techniques to the newsvendor problem, Vairaktarakis describes adversarial solutions for several performance criteria in the setting of multiple item types per period and a budget constraint[21]. Bertsimas and Thiele give solutions for several variants of the newsvendor problem which optimize the order quantity based on historical data[3]. The solutions discussed take into account risk preferences by "trimming," or ignoring, historical data which leads to overly optimistic predictions.

Learning from Experts Weighted Majority is a very adaptable machine learning algorithm developed by Littlestone and Warmuth[14]. There are several versions of the weighted majority algorithm, including discrete, continuous, and randomized. Each consults the predictions of experts, and seeks to minimize the regret (in terms of prediction mistakes) with respect to the best

expert in the pool.

Weighted Majority and variations thereof have been applied to a wide variety of areas including online portfolio selection[8, 7] and robust option pricing[9]. Other variants include the WINNOWER algorithm also developed by Littlestone[13], which has been applied to such areas as predicting user actions on the world wide web[1].

Follow the Perturbed leader is a general algorithm for online decision making which is also applicable to the learning from experts problem. It's creators, Kalai and Vempala[12], apply the algorithm to such problems as online shortest paths[20] and the tree update problem[19].

4 Algorithms

For these experiments, we implement the following algorithms as described:

STOPT This approach is given perfect information about the demand sequence, and chooses the single order quantity to use for all periods which maximizes the overall profit (and thus also minimizes the total regret). As Bertsimas and Thiele discuss[3], the static offline optimal choice is the $[t - t(c/r)]^{th}$ order statistic of the demand sequence.

NORMAL This stochastic solution assumes the demands will be drawn from a known normal distribution, and maximizes the expected profit. This approach prescribes ordering the amount $\mu + \sigma\phi^{-1}((r-c)/r)$, where $\phi^{-1}(\cdot)$ is the inverse of the standard normal cumulative distribution function[11].

SCARF This stochastic solution is described in Scarf's original paper[18] as well as in [11]. The solution maximizes the expected profit for the worst case distribution (a *maximin* approach in the stochastic sense) with first and second moments μ and σ . The order quantity is prescribed to be $\mu + \frac{\sigma}{2}(\sqrt{(r-c)/c} - \sqrt{c/(r-c)})$ if $c(1 + \sigma^2/\mu^2) < r$, and 0 otherwise.

MINIMAX This is the *minimax regret* approach mentioned in Section 1. Described by [21], the algorithm orders the quantity $(M(r-c) + mc)/r$ for every period, which minimizes the maximum possible regret from the optimal for each period. As such, it also minimizes the maximum possible regret for the whole sequence.

The solution works by balancing the regret suffered by the two worst case possibilities: the demand being m or M . Because of this, its order never changes (as long as the range $[m, M]$ doesn't change), and is very pessimistic in nature.

WMN We develop this algorithm (Weighted Majority Newsvendor) as an adaptation of the Weighted Majority algorithm of Littlestone and Warmuth[14]. The algorithm takes two parameters: n , the number of "experts" to consult, and $\beta \in (0, 1]$, the weight adjustment parameter. Essentially, we divide up the range $[m, M]$ into n buckets, and have expert i predict the minimax regret order quantity for the i^{th} bucket. Buckets and experts are set up so that each bucket/expert pair has the same minimax regret.

As per the standard operation of Weighted Majority, each expert is given an initial weight of 1. After each round, we decrease each expert i 's weight by some factor F , where F depends on β and the regret that expert would have suffered on the demand seen using its prediction. If an expert is often wrong, its weight will be decreased faster than others. This punishment happens faster overall with smaller β 's.

The amount ordered by WMN in a given period is the weighted average of all experts. The intuition is that wherever the static optimal choice is, it must fall in one of the n buckets, and thus one of our experts will be close to this static optimal choice. Further, because experts' weights are decreased according to how poorly they do, the algorithm is able to learn where the static optimal choice is after a few periods, and even adapt to changing inputs over time.

Adapting the analysis of Weighted Majority to the non linear newsvendor profit function requires special care to ensure bounds similar to that of Weighted Majority can still be given. In Appendix A, we give a detailed description of WMN and a proof of the following theorem:

THEOREM 4.1. *The total regret experienced by WMN for a t period newsvendor game with per item cost c , per item revenue r , and all demands within $[m, M]$ satisfies*

$$\begin{aligned} & \text{WMN}_{TotalRegret} \\ & \leq \frac{\mathbb{C} \ln(n)}{1 - \beta} + \frac{\ln\left(\frac{1}{\beta}\right) c(M - m)(r - c)t}{nr(1 - \beta)} \\ & \quad + \frac{\ln\left(\frac{1}{\beta}\right) \text{STOPT}_{TotalRegret}}{1 - \beta} \end{aligned}$$

where $\mathbb{C} = \max\{(M - m)(r - c), (M - m)c\}$ is the maximum possible single period regret, n is the number of buckets used by WMN, and β is the update parameter used.

WMNS WMNS, for Weighted Majority Newsvendor Shifting, is based on the "shifting target" version of the standard Weighted Majority algorithm. Here, if the input sequence can be decomposed into subsequences such

that for each subsequence a particular expert does very well, then WMNS will do nearly as well for that subsequence. WMNS needs no information about how many shifts there will be, or when they will be. For example, if for the first third of the sequence all demands are near m , WMNS will initially adjust the weights of the experts so that it is ordering near m as well. If the sequence shifts so that demands are then drawn from near M , WMNS will adjust the weights quickly (quicker than WMN) so that the order quantities will match.

This ability comes from WMNS’s use of a weight limiting factor $\delta \in (0, 1]$, so that no expert’s weight will be less than δ times the average weight. When a new expert starts doing significantly better, the old best expert’s weight is decreased to below the new expert’s weight more rapidly, as the new expert’s weight is guaranteed not to be too low in relation.

THEOREM 4.2. *The total regret experienced by WMNS for a t period newsvendor game with per item cost c , per item revenue r , and all demands within $[m, M]$ satisfies*

$$\begin{aligned} & \text{WMNS}_{\text{TotalRegret}} \\ & \leq \frac{k\mathbb{C} \ln\left(\frac{n}{\beta\delta}\right)}{(1-\beta)(1-\delta)} + \frac{\ln\left(\frac{1}{\beta}\right) c(M-m)(r-c)t}{nr(1-\beta)(1-\delta)} \\ & \quad + \frac{\ln\left(\frac{1}{\beta}\right) \text{SSTOPT}_{\text{TotalRegret}}}{(1-\beta)(1-\delta)} \end{aligned}$$

where $\mathbb{C} = \max\{(M-m)(r-c), (M-m)c\}$ is the maximum possible single period regret, n is the number of buckets used by WMNS, β is the update parameter used, and δ is the weight limiting parameter used. SSTOPT is allowed to use a static optimal choice for k subsequences (i.e., is allowed to change order values $k-1$ times, see below).

Details of WMNS’s operation and proof of the above bounds are given in Appendix B.

SSTOPT This “optimal,” which makes its decisions based on the entire sequence, is a slightly stronger version of STOPT, which is allowed to change its order quantity exactly $k-1$ times during the sequence.

FPL Similar to WMN, FPL is a randomized algorithm based upon the Follow the Perturbed Leader approach developed by Kalai and Vempala[12]. As a general algorithm it is well suited to making decisions a number of times, when one wants to minimize the total cost in relation to the best single decision for all periods. Here, decisions will be of the form “use expert i ’s prediction,” where the experts again predict minimax

values in buckets which divide the $[m, M]$ range. FPL as we use it takes two parameters, n , for the number of experts/buckets, and ϵ , which affects the final cost bound in relation to the best static decision.

THEOREM 4.3. *The total regret experienced by FPL for a t period newsvendor game with per item cost c , per item revenue r , and all demands within $[m, M]$ satisfies*

$$\begin{aligned} & E[\text{FPL}_{\text{TotalRegret}}] \\ & \leq \frac{4\mathbb{C}(1+\ln(n))}{\epsilon} + \frac{(1+\epsilon)c(M-m)(r-c)t}{nr} \\ & \quad + (1+\epsilon)\text{STOPT}_{\text{TotalRegret}} \end{aligned}$$

where $\mathbb{C} = \max\{(M-m)(r-c), (M-m)c\}$ is the maximum possible single period regret, n is the number of buckets used by FPL, and ϵ is the randomness parameter used.

Details of the algorithm and proof of the bounds it gives in our application appear in Appendix C.

5 Experiments

In order to evaluate the online learning algorithms for the newsvendor problem, we run them on simulated demand sequences comparing the total regret suffered by each approach to the regret suffered by the stochastic algorithms SCARF and NORMAL, as well as MINIMAX and STOPT.

Unless otherwise noted, all experiments consist of 100 demand newsvendor sequences, and each data point represents the average of 100 such trials. Thus, data points in the following figures typically represent the average total regret of various approaches on newsvendor sequences of length 100. Also, due to space limitations, we won’t experiment with the affect of the upper and lower demand bounds $[m, M]$; we’ll instead fix these bounds to $[10, 100]$ for all tests. Whenever a normal distribution is used, we restrict it to this range by resampling if a demand falls outside the range, and we further restrict all demands to be integers.

5.1 Algorithm Parameters

5.1.1 β , ϵ , and μ For this first batch of tests, we investigate the performance of our three machine learning approaches while varying some of the parameters they accept as input. WMN and WMNS use β as a weight adjustment parameter: the smaller β is, the quicker expert weights are adjusted downward. WMNS also uses a “weight limiting” parameter δ , which we hold constant at 0.3 for these tests.

FPL uses the parameter ϵ , which affects the amount of “randomness” used in deciding which expert to

