INDR 450/550

Spring 2022

Lecture 23: Dynamic Programming

May 16, 2022

Fikri Karaesmen

Announcements

- Class Exercise at the end of lecture today. If you are participating online, please upload your document under Course Contents/Class Exercises
- HW 4 is now available (model reduction, trees, forests etc.)
- Exam on May 13.
    - Review exercises are available
    - Make sure that you also review the class exercises, the labs, and the homeworks

The Problem

- Inventory or Capacity Management under Uncertain Demand
- Matching supply (order quantity) to the random demand to minimize expected underage and overage costs
  - Short selling season
  - Decision made in advance of the season
  - No replenishments or capacity additions during the season (purchasing in advance is required)
  - Demand forecasts have considerable uncertainty
  - Items lose value significantly after the season

Empirical Risk Minimization

- The stochastic optimization formulation has recently received a lot of attention.
- We'll next look at a more recent formulation that combines predictive analytics with prescriptive analytics.

Gah-Yi Ban and Cynthia Rudin, "The Big Data Newsvendor: Practical Insights From Machine Learning", *Operations Research*, Vol. 67, pp. 90-108, 2019.

Feature based newsvendor: machine learning ideas

- Formulate and find a way to solve the following problem:

$$\min_{Q(\cdot)} R(Q(\cdot), S_n) = \frac{1}{n}\sum_{i=1}^{n} c_u\,(d_i - Q(\mathrm{x}_i))^+ + c_o(Q(\mathrm{x}_i) - d_i)^+$$

- To solve the above, we need to guess the functional form of $Q(\boldsymbol{x})$.
- Reasonable guess: a linear decision rule

$$Q(\mathrm{x}) = Q((x^1, x^2, \ldots, x^p)) = q^0 + \sum_{j=1}^{p} q^j\, x^j$$

- Note that this is supported by what we have seen before: the optimal order quantity is a linear function of the predictors and the standard deviation of the estimation error

Feature based newsvendor: machine learning ideas

- Here's the ML – optimization formulation:

$$\min_{Q(\cdot)} R(Q(\cdot), S_n) = \frac{1}{n}\sum_{i=1}^{n} c_u\,(d_i - Q(\mathrm{x}_i))^+ + c_o(Q(\mathrm{x}_i) - d_i)^+$$

$$\equiv \min_{Q=(q^1,q^2,\ldots q^p)} \frac{1}{n}\sum_{i=1}^{n} c_u\, z_i{}^+ + c_o z_i{}^-$$

$$s.t.$$

$$\begin{array}{lcll} z_i{}^+ & \geq & d_i - \left(q^0 + \sum_{j=1}^{p} q^j x_i^j\right) & i = 1,2,,\ldots,n \\ z_i{}^- & \geq & q^0 + \sum_{j=1}^{p} q^j x_i^j - d_i & i = 1,2,,\ldots,n \\ z_i{}^+, z_i{}^- & \geq & 0 & i = 1,2,,\ldots,n \end{array}$$

This is a linear program and can therefore be efficiently solved at large scale.

Feature based newsvendor: Kernel-based methods

- Ban and Rudin also consider a Kernel-based formulation based on a Nadaraya-Watson estimator.
- This is a k Nearest Neighbours approach with a sophisticated weight function.
- Consider that the k-nearest neigbours are: $d_1, d_2, \ldots, d_k$ and their corresponding weights are $w_1, w_2, \ldots, w_k$.
- The resulting problem is equivalent to solving a discrete demand newsvendor problem with a probability mass function: $w_1, w_2, \ldots, w_k$.

Feature based newsvendor: Kernel-based methods

- Assume that Friday is expected to be sunny, temperature 15 degrees.
- It is somewhat similar to the first four days but more similar to days 3 and 4.
- Assume that the weights are: $w_1 = 0.1$, $w_2 = 0.2$, $w_3 = 0.4$, $w_4 = 0.3$.

Data Point	Day	Temp.	Weather	Sales
1	Monday	12	Rainy	23kg
2	Tuesday	14	Cloudy	28kg
3	Wed.	14	Sunny	35kg
4	Thu.	17	Sunny	30kg
…	…	….	…	…

$$Q^* = \min\left\{F_D(Q) \geq \frac{c_u}{c_u + c_0},\ Q \in \mathrm{Z}^+\right\}$$

Feature based newsvendor: Kernel-based methods

- But we can do better by taking the weights as a probability distribution.
- Assume that the weights are: $w_1 = 0.1, w_2 = 0.2, w_3 = 0.4, w_4 = 0.3.$

Data Point	Day	Temp.	Weather	Sales
1	Monday	12	Rainy	23kg
2	Tuesday	14	Cloudy	28kg
3	Wed.	14	Sunny	35kg
4	Thu.	17	Sunny	30kg
…	…	….	…	…

Sales	CDF
23kg	0.1
28kg	0.3
30kg	0.6
35kg	1

$$Q^* = \min\left\{F_D(Q) \geq \frac{c_u}{c_u + c_0},\ Q \in \mathrm{Z}^+\right\}$$

This time we get a different order quantity when the critical fraction changes.

Feature based newsvendor: new vs. old

- Contrast the ML approach with standard separated predictive analytics and optimization
  - In predictive analytics, we estimate the demand distribution as a function of the parameters and then solve the standard newsvendor problem.
  - In joint estimation and optimization, we do not attempt to forecast the demand but go directly for the optimal decision rule.
- Note that the predictive analytics problem is also an optimization problem: (i.e. Least squares optimization, lasso, finding the optimal tree approximation).
- The objective function of the estimation problem is not aligned with that of the operational optimization problem.
- For high margin items (c_u>> c_o), overestimation of demand might be better than underestimation of demand but regression does not take this into account.

Feature based newsvendor: new vs. old

- It turns out that the joint estimation-optimization approach may perform better than separated estimation and optimization with small and medium size data samples and a large number of predictors.
- This is subject of ongoing research.

Overview of Prediction to Prescription Path

- We would like to solve optimization problems involving random quantities with side information (conditional on some information).
  - $D|X = (x_1, x_2, \ldots, x_p)$
- Predictive methods focus on estimating the mean of the uncertain quantity: $\hat{d}(x_1, x_2, \ldots, x_p)$.
- We can use the above point estimator to solve a deterministic optimization problem but this approach does not perform well because typically the best solution uses information beyond the mean (i.e. Variance, CDF etc.).

Overview of Prediction to Prescription Path

- If we don’t have predictors or do not wish to use them (i.e. if we believe that the demand sample is iid after some simple transformation), we can use Empirical Risk Minimization (sample average approximation). Take the sample as the ‘world’, each observed point gets an equal probability.
- This works reasonably well. The resulting optimization problems are large (for large samples) but can sometimes be solved efficiently.

Overview of Prediction to Prescription Path

- With ARIMA and ordinary linear regression we can access the model residuals, we can use them to estimate the distribution of:

$$D|X = (x_1, x_2, \dots, x_p)$$

- In theory, the residuals are expected to be normally distributed, this facilitates the job. Otherwise, we need to fit a probability distribution to the residuals.

Overview of Prediction to Prescription Path

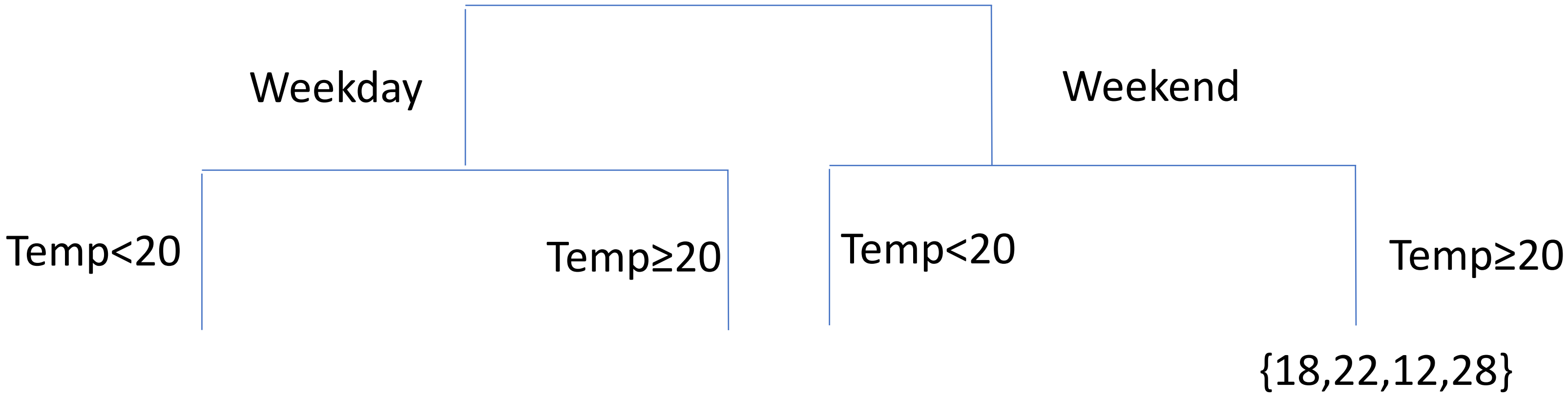
- With other methods, we need to somewhat extract an estimator for the probability distribution from the method.
- If we use KNN, then we can use each of the K nearest neigbours to constitute a discrete probability distribution.

Overview of Prediction to Prescription Path

- If we use a tree based approach, we can use the observations under the same leaf to generate a discrete probability distribution.
- With random forests, we can use weighted averages over the leaves of each tree in the forest.

Overview of Prediction to Prescription Path

- Consider the ice-cream sales example and the following tree:



- If the critical fractile is 0.8, what is the optimal order quantity for a Saturday with weather forecast 24 degrees?

Overview of Prediction to Prescription Path

- Joint estimation and optimization seems to perform really well (Ban and Rudin (2019))

Dynamic Programming – Reinforcement Learning

- Not much time left but let’s do a focused introduction.
- We’ll take a look at a specific problem that arises in dynamic capacity control in revenue management.
- This defines the scope of the next four lectures:
  - Finite state finite horizon stochastic dynamic program (Markov Decision Process) and their analysis
  - The need for approximations and some ideas

Capacity Allocation in Revenue Management

- Q units of inventory are available and they have to be sold before a strict due-date. This is appropriate for airlines, trains, hotels, concerts, theater etc.
- There are multiple classes of customers who are willing to pay different prices
- Service industry has found ways of segmenting such classes
  - Student discounts
  - Early reservation discounts
  - Little changes in service (extra luggage, flexiibility in cancelation etc.)
- This generates the following capacity control problem when a demand from a lower class shows up, should we sell the item or reject the demand to keep the item for a potential future customer that might pay a higher price.

Capacity Allocation in Revenue Management

- We'll look at a model which simplifies some of the realistic issues but captures the essence of the dynamic problem (sell today versus reserve for later under a deadline and random demand arrivals).
- Two classes of customers arriving independently and randomly over time. Class 1 customers purchase the item price p_1 and Class 2 purchase at p_2 ($p_1 > p_2$).
- We are in discrete time and items perish at time T.
- Our options at a given period t, are to accept or reject the demands from classes 1 and 2. We would like to do this to maximize the expected revenue from sales.

Capacity Allocation in Revenue Management

- We divide the planning horizon into small time segments such that there can be at most one booking request in the segment.
- Let us also assume each booking request is for a single item (seat).
- We'll count time backwards (number of periods until the deadline (i.e. time of flight)). Assume that there are T periods remaining until the deadline.
- In a period, at most one arrival (class 1, class 2 or no arrivals).
- q_{it}: probability that a request from class i takes place with t periods remaining (i=1,2,..k)
- q_{0t}:probability that no demand arrives with t periods remaining
- $\sum_{i=0}^{k} q_{it} = 1 \; \forall t, t = 0,1,2,\dots,T$

Some formality: stochastic DPs

- To solve the previous problem, we'll set up a stochastic dynamic optimization problem.
- The goal is to find a way of to control the capacity over time by accepting/rejecting demands in order to maximize the revenue.
- **State:** The number of items available for sales
- **Actions:** Accept or Reject Demands from Classes 1 and 2
- **Policy:** A complete rule that tells us which action take at any time t for all states (inventory levels) x.
- Find the optimal policy: a rule that enables to maximize the expected revenue starting from some time t with a state x.

Some formality: stochastic DPs

- State: inventory level x at time t.
- Action $a(x,t) = (a_1, a_2)$ where $a_i \in \{\mathrm{A},\mathrm{R}\}$.
  - Example: a(5,1) = (A,A) -> accept demand from Class 1 and Class 2 if we have 5 items with one period remaining until the deadline.
- Policy: **μ**: A complete set of actions for all x and for all t.
  - Example for all x and for all t accept all demands from class 1 and reject all demands from class 2: $a(x,t) = (A, R)$ for all x and for all t .
- Optimal policy: **μ*** a policy that maximizes the expected profit

Policy Evaluation

- It turns out that if a policy is specified, we can compute the expected revenue it generates starting from any x and t.
- Let's take as an example **μ'**: $a(x,t) = (A, R)$ for all x and for all t .
- Let's denote by $w_t(x)$ the expected revenue that is generated using policy μ' starting from some x and t.
- We'll write a recursion to compute $w_t(x)$.
- Note that $w_t(0) = 0$ for all t .
- (if there is no inventory left to sell, the future revenue is zero).
- And $w_0(x) = 0$ for all x (if there is no time left to sell, the future revenue is also zero).

Policy Evaluation

- We can now compute $w_1(x)$:

$$w_1(x) = q_1(p_1 + w_0(x-1)) + q_2 w_0(x) + q_3 w_0(x) \text{ for all } x \geq 1$$

This enables us to compute $w_1(x)$ for all x. Then, we can compute:

$$w_2(x) = q_1(p_1 + w_1(x-1)) + q_2 w_1(x) + q_3 w_1(x) \text{ for all } x \geq 1$$

We can therefore recursively compute:

$$w_T(x) = q_1(p_1 + w_{T-1}(x-1)) + q_2 w_{T-1}(x) + q_3 w_{T-1}(x) \text{ for all } x \geq 1$$

Policy Evaluation to Optimization

- We can now therefore compute the expected revenue from any policy.
- But this is not a tool for finding the optimal policy, there are too many policies to evaluate even for this simple example.

- Bellman's principle of dynamic programming is about the following insight: we can find the optimal policy for the entire problem (T periods) by combining the optimal policies for subproblems that find the optimal policies starting from start from t=1,2,..T-1.
  - if you find yourself in state x with t periods remaining it does not matter how you got there, the best that you can do is to maximize the expected reward over the remaining horizon.

Policy Evaluation to Optimization

Let $v_1(x)$ be the maximum expected revenue with one period remaining and x items available , we can write

$$\begin{aligned} v_1(x) &= q_1 \max\{(p_1 + v_0(x-1)), v_0(x)\} \\ &\quad + q_2 \max\{(p_2 + v_0(x-1)), v_0(x)\} \\ &\quad + q_3 v_0(x) \end{aligned}$$

Recall that $v_0(x) = 0$ for all $x \geq 0$. We can therefore extract the optimal action with 1 period to go: $a(x, 1) = (A, A)$. It is optimal to sell to both classes with 1 period to go.

Policy Optimization

- But now we can do the same for $v_2(x)$.
  - For $0 \leq x \leq Q$ We have:

$$\begin{aligned} v_2(x) &= q_{12}\max\{p_1 + v_1(x-1), v_1(x)\} + q_{22}\max\{p_2 + v_1(x-1), v_1(x)\} + \ldots \\ &\quad + q_{k2}\max\{p_k + v_1(x-1), v_1(x)\} + q_{02}v_1(x) \end{aligned}$$

- Going backwards, for t periods remaining, we have for $0 \leq x \leq Q$:

$$\begin{aligned} v_t(x) &= q_{1t}\max\{p_1 + v_{t-1}(x-1), v_{t-1}(x)\} + q_{2t}\max\{p_2 + v_{t-1}(x-1), v_{t-1}(x)\} + \ldots \\ &\quad + q_{kt}\max\{p_k + v_{t-1}(x-1), v_{t-1}(x)\} + q_{0t}v_{t-1}(x) \end{aligned}$$

- This can be computed if $v_{t-1}(x)$ has already been computed for all x.

Policy Optimization

- To extract the optimal actions a_1 and a_2 at time t for state x we note that:

$$a_1 = A \text{ if } p_1 + v_{t-1}(x-1) \geq v_{t-1}(x), \text{ and } a_1 = R; \text{otherwise}$$

$$a_2 = A \text{ if } p_2 + v_{t-1}(x-1) \geq v_{t-1}(x), \text{ and } a_1 = R; \text{otherwise}$$

- Note that since $p_1 > p_2$, if it is optimal to sell to class 2 at t and x then it is also optimal to sell to class 1.

We can compute both the expected optimal profit and the corresponding optimal policy from the same recursion.