Data-Driven Demand Learning and Dynamic Pricing Strategies in Competitive Markets

Pricing Strategies & Dynamic Programming

Rainer Schlosser, Martin Boissier, Matthias Uflacker

Hasso Plattner Institute (EPIC)

April 30, 2018

Outline

- Questions/Support: Market Simulation (Exercise)

- Today: Pricing Strategies & Duopoly Games

  Learn about Dynamic Programming

- 1st Exercise: Simulated Markets & Price Reactions

Recall: Rule-Based Price Reaction Strategies

- Idea: (1) Observe competitor prices $\vec{p}$ + (2) Adjust price a

- Examples: $$a(\vec{s}) = a^{(1)}(\vec{p}) := \max\left(c, \min_{k=1,..,K} p_k - \varepsilon\right)$$

$$a(\vec{s}) = a^{(n)}(\vec{p}) := \max_{a \in A: rank(a,\vec{p})=n} a$$

$$a(\vec{s}) = a^{(random)}(\vec{p}) := if\ U(0,1) < 0.5\ \ then\ a^{(1)}(\vec{p})\ \ else\ a^{(2)}(\vec{p})$$

$$a(\vec{s}) = a^{(2bound)}(\vec{p}) := \begin{cases} a^{(1)}(\vec{p}) & , p^{\min} \le \min_{k=1,..,K} p_k \le p^{\max} \\ p^{\max} & , else \end{cases}$$

Duopoly Example

- Assume K=2 sellers. Assume only one feature: price

- Define different price reaction strategies $a(p)$, i.e.,

  if the competitor's current price is p, we adjust our price to $a(p)$

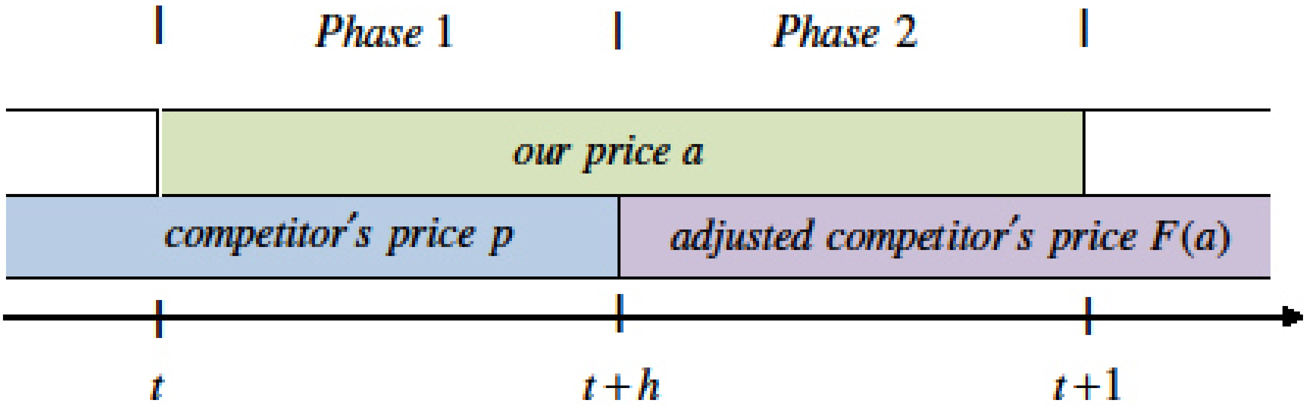
  Admissible prices are $a(p) \in \{1, 2, ..., 100\}$

- Let the competitor's response strategy be given by: $p(a) := \max(a-1, 1)$

- We adjust our prices a at times $t = 1, 2, 3, \ldots$

  The competitor adjusts his prices p at times $t = 0.5, 1.5, 2.5, \ldots$

Sequence of Events (Duopoly Example)



- In every interval $(t, t+0.5)$, $t = 0, 0.5, 1.0, \ldots$, a sale occurs with probability $1 - \min(a_t, p_t)/100$. With probability $\min(a_t, p_t)/100$ no sale takes place

Duopoly Example

- In every interval $(t, t+0.5)$, $t = 0, 0.5, 1.0, \ldots$, a sale occurs with probability $1 - \min(a_t, p_t)/100$. With probability $\min(a_t, p_t)/100$ no sale takes place

- If a sale takes place the customer chooses either our offer (k=1) or the competitor's offer (k=2) with probability $P(k, \vec{p})$ according to Approach I, where $\vec{p} = (p^{(1)}, p^{(2)}) = (a, p)$, i.e., $p^{(1)} = a$ (we) and $p^{(2)} = p$ (competitor)

- Simulate until time T=1000. Start with $a_0 = p_0 = 20$ at time $t = 0$

- Which strategy $a(p)$ performs best, i.e., maximizes expected revenues?

Goals for Today

- We want to optimally solve the duopoly example
- We have a dynamic optimization problem
- What are dynamic optimization problems?
- How to apply dynamic programming techniques?

What are Dynamic Optimization Problems?

- How to control a dynamic system over time?
- Instead of a single decision we have a sequence of decisions
- The system evolves over time according to a certain dynamic
- The decisions are supposed to be chosen such that
  a certain objective/quantity/criteria is optimized
- Find the right balance between short and long-term effects

Examples Please!

Examples

- Inventory Replenishment
- Reservoir Dam
- Drinking at a Party
- Exam Preparation
- Brand Advertising
- Used Cars
- Eating Cake

Task: Describe & Classify

- Goal/Objective
- State of the System
- Actions
- Dynamic of the System
- Revenues/Costs
- Finite/Infinite Horizon
- Stochastic Components

Classification

Example	Objective	State	Action	Dynamic	Payments
Inventory Mgmt.	min costs	#items	#order	entry-sales	order/holding
Reservoir Dam	provide power	#water	#production	rain-outflow	none
Drinking at Party*	max fun	‰	#beer	impact-rehab	fun/money
Exam Preparation*	max mark/effort	#learned	#learn	learn-forget	effort, mark
Advertising	max profits	image	#advertise	effect-forget	campaigns
Used Cars	min costs	age	replace(y/n)	aging/faults	buy/repair costs
Eating Cake*	max utility	%cake	#eat	outflow	utility

* Finite horizon

General Problem Description

- What do you want to minimize or maximize (Objective)
- Define the state of your system (State)
- Define the set of possible actions (state dependent) (Actions)
- Quantify event probabilities (state+action dependent) (Dynamics) (!!)
- Define payments (state+action+event dependent) (Payments)
- What happens at the end of the time horizon? (Final Payment)

Dynamic Pricing Scenario (Duopoly Example)

- We want to sell items in a duopoly setting with finite horizon
- We can observe the competitor’s prices and adjust our prices (for free)
- We can anticipate the competitor’s price reaction
- We know sales probabilities for various situations
- We want to maximize total expected profits

Problem Description (Duopoly Example)

- Framework: $t = 0, 1, 2, \ldots, T$ — Discrete time periods
- State: $s = p$ — Competitor's price
- Actions: $a \in A = \{1, \ldots, 100\}$ — Offer prices (for one period of time)
- Dynamic: $P(i, a, s)$ — Probability to sell *i* items at price *a*
- Payments: $R(i, a, s) = i \cdot a$ — Realized profit
- New State: $p \xrightarrow{\Gamma(i,a,s)} F(a)$ — State transition / price reaction *F*
- Initial State: $s_0 = p_0 = 20$ — Competitor's prices in $t=0$

Problem Formulation

- Find a *dynamic pricing strategy* that

  maximizes total expected (discounted) profits:

- $$\max E\left[\sum_{t=0}^{T} \underbrace{\delta^t}_{\substack{discount\\factor}} \cdot \left(\sum_{i_t \geq 0} \underbrace{P(i_t, a_t, S_t)}_{\substack{probability\ to\ sell\ i_t\ items\\ at\ price\ a_t\ in\ situation\ S_t}} \cdot \underbrace{i_t}_{sales} \cdot \underbrace{a_t}_{price} \right) \middle| \underbrace{S_0 = s_0}_{initial\ state} \right], \quad 0 < \delta \leq 1$$

- What are admissible policies? Answer: Feedback Strategies

- How to solve such problems? Answer: Dynamic Programming

Solution Approach (Dynamic Programming)

- What is the **best expected value** of having the chance to sell . . .

  "items from time t on starting in market situation s"?

- Answer: That's easy $V_t(s)$! ?????

- We have renamed the problem. Awesome. But - that's a solution approach!

- We don't know the "Value Function *V*", but *V* has to satisfy the relation

  Value (state today) = Best expected (profit today + Value (state tomorrow))

Solution Approach (Dynamic Programming)

- *Value (state today) = Best expected (profit today + Value (state tomorrow))*
- Idea: Consider the transition dynamics within one period

  What can happen during one time interval?

state today	#sales	profit	state tomorrow	probability
	0	0	$F(a)$	$P(0,a,s)$
$s=p$	1	a	$F(a)$	$P(1,a,s)$
	2	$2a$	$F(a)$	$P(2,a,s)$

with (a,p) *vs.* $(a,F(a))$

- What does that mean for the **value** of state $s=p$, i.e., $V_t(s)=V_t(p)$?

Bellman Equation

state today	#sales	profit	state tomorrow	probability
	0	0	$F(a)$	$P(0,a,s)$
$s=(p)$	1	a	$F(a)$	$P(1,a,s)$
	2	$2a$	$F(a)$	$P(2,a,s)$

with (a,p) *vs.* $(a,F(a))$

$$V_t(p) = \max_{a\geq 0} \Bigg\{ \underbrace{P(0,a,p)}_{probability\ not\ to\ sell} \cdot \Bigg(\underbrace{0\cdot a}_{today's\ profit} + \underbrace{\delta\cdot V_{t+1}\big(F(a)\big)}_{best\ disc.\ exp.future\ profits} \Bigg) + \underbrace{P(1,a,p)}_{probability\ to\ sell} \cdot \Bigg(\underbrace{1\cdot a}_{today's\ profit} + \underbrace{\delta\cdot V_{t+1}\big(F(a)\big)}_{best\ disc.\ exp.future\ profits} \Bigg) + \ldots \Bigg\}$$

Optimal Solution

- We finally obtain the Bellman Equation:

$$V_t(p) = \max_{a \geq 0} \left\{ \sum_{i \geq 0} \underbrace{P(i,a,p)}_{probability} \cdot \left(\underbrace{i \cdot a}_{today's\ profit} + \underbrace{\delta \cdot V_{t+1}\big(F(a)\big)}_{best\ disc.\ exp.future\ profits\ of\ new\ state} \right) \right\}$$

- Ok, but why is that interesting?

- Answer: Because $a_t^*(p) = \arg\max_{a \in A} \{...\}$ is the *optimal policy*

- Ok! Now, we just need to compute the Value Function!

How to solve the Bellman Equation?

- Using the terminal condition $V_T(p) := 0$ for time horizon T (e.g., 1000)

  We can compute the value function *recursively* $\forall t, p$:

$$V_t(p) = \max_{a \geq 0} \left\{ \sum_{i \geq 0} \underbrace{P(i,a,p)}_{probability} \cdot \left(\underbrace{i \cdot a}_{today's\ profit} + \underbrace{\delta \cdot V_{t+1}\left(F(a)\right)}_{best\ disc.\ exp. future\ profits\ of\ new\ state} \right) \right\}$$

- The optimal strategy $a_t^*(p)$, $t = 1,...,T$, $p = 1,...,100$,

  is determined by the arg max of the value function

How to solve the Bellman Equation?

- Using the terminal condition $V_T(p) := 0$ for time horizon T (e.g., 1000)

  We can compute the value function *recursively* $\forall t, p$:

$$V_t(p) = \max_{a \geq 0} \left\{ \sum_{i \geq 0} \underbrace{P(i,a,p)}_{probability} \cdot \left(\underbrace{i \cdot a}_{today's\ profit} + \underbrace{\delta \cdot V_{t+1}(F(a))}_{best\ disc.\ exp.future\ profits\ of\ new\ state} \right) \right\}$$

- The optimal strategy $a_t^*(p)$, $t = 1,...,T$, $p = 1,...,100$,

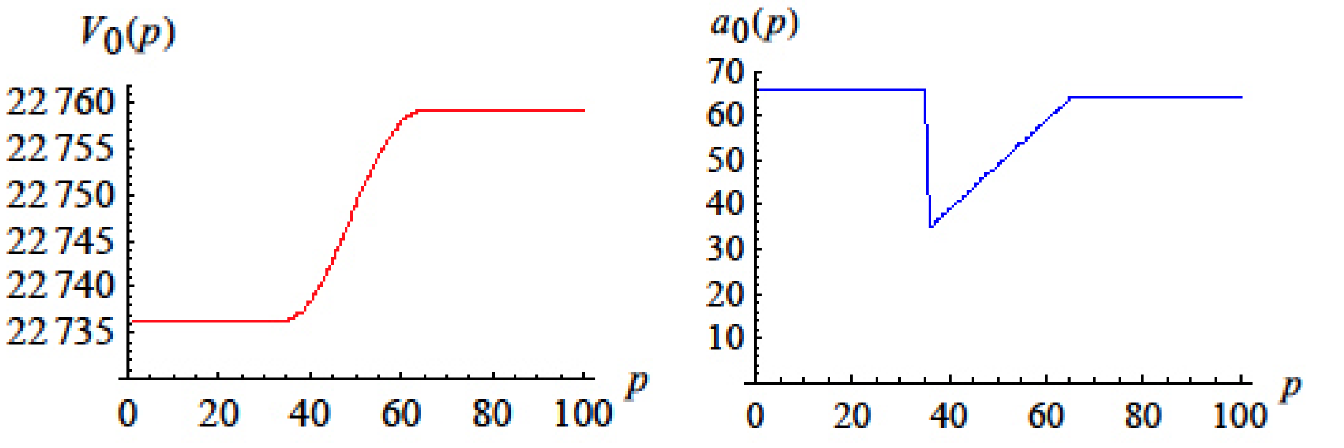
  is determined by the arg max of the value function

- In Pseudo-Code:

```
param V{t in 0..T,p in A}:= if t<T then max {a in A}
sum{i in I} P[i,a,p] * ( i*a + delta * V[t+1,F[a]] );
```

Optimal Pricing Strategy of the Duopoly Example

Best expected profit | Optimal pricing strategy



Infinite Horizon Problem

- $$\max E\left[\left.\sum_{t=0}^{\infty}\underbrace{\delta^t}_{\substack{discount\\ factor}}\cdot\left(\sum_{i_t\geq 0}\underbrace{P(i_t,a_t,S_t)}_{\substack{probability\ to\ sell\ i_t\\ at\ price\ a_t\ in\ situation\ S_t}}\cdot\underbrace{i_t}_{sales}\cdot\underbrace{a_t}_{price}\right)\right|\underbrace{S_0=s_0}_{initial\ state}\right],\ \underline{0<\delta<1}$$

- Will the value function be time-dependent?

- Will the optimal price reaction strategy be time-dependent?

- How does the Bellman equation look like?

Solution of the Infinite Horizon Problem

- $$V^*(p) = \max_{a \geq 0} \left\{ \sum_{i \geq 0} \underbrace{P(i,a,p)}_{probability} \cdot \left(\underbrace{i \cdot a}_{today's\ profit} + \underbrace{\delta \cdot V^*\big(F(a)\big)}_{disc.\ exp. future\ profits\ of\ new\ state} \right) \right\}$$

- **Approximate solution**: Finite horizon approach (value iteration)

- For "large" T the values $V_0(p)$ converge to the exact values $V^*(p)$

- The optimal policy $a^*(p)$, $p = 1,...,100$, is determined by

  the arg max of the last iteration step, i.e., $a_0(p)$

Exact Solution of the Infinite Horizon Problem

- $$V^*(p) = \max_{a \geq 0} \left\{ \sum_{i \geq 0} \underbrace{P(i,a,p)}_{probability} \cdot \left(\underbrace{i \cdot a}_{today's\ profit} + \underbrace{\delta \cdot V^*(F(a))}_{disc.\ exp.future\ profits\ of\ new\ state} \right) \right\},\ p \in A$$

- We have to solve a system of nonlinear equations

- Solvers can be applied, e.g, MINOS (see NEOS Solver)

- In Pseudo-Code:

```
subject to NB {p in A}:  V[p] = max {a in A}
sum{i in I} P[i,a,p] * ( i*a + delta * V[F[a]] ); solve;
```

Questions?

- State
- Action
- Events
- Dynamics & State Transitions
- Recursive Solution Principle

- General concept applicable to other problems

1st Exercise – Simulated Markets & Price Reactions

- Study our market simulation notebook
- Modify the arrival stream of interested customers
- Modify the customer behaviour (scoring weights)
- Modify the competitors’ reaction strategies
- Bonus (Dynamic Programming): Solve duopoly example with finite horizon

Overview

2	April 24	Customer Behavior
3	April 30	Pricing Strategies, 1st Homework (market simulation)
4	**May 8**	**Demand Estimation, 2nd Homework (demand learning)**
5	May 15	Introduction Price Wars Platform
6	May 22	Warm up Platform Exercise (in Groups)
7	May 29	Dynamic Pricing Challenge / Projects
8	June 5	no Meeting
9	June 12	Workshop / Group Meetings
10	June 19	Presentations (First Results)
11	June 26	Workshop / Group Meetings
12	July 3	Workshop / Group Meetings
13	July 10	no Meeting
14	**July 17**	**Presentations (Final Results), Feedback, Documentation (Aug/Sep)**