An efficient exploration basis for learning

Barycentric spanners: Characterisation and Applications

Manu K. Gupta

TCS Innovation Labs Hyderabad, India.

Joint work with Chaitanya Amballa and Sanjay Bhat

Outline

- Motivation
  - Linear regression
  - Thompson sampling for dynamic pricing
- Current state of art
  - Approximate barycentric spanner
- Barycentric spanner characterization
  - Faster and efficient
  - Exact barycentric spanner
- Future work

Motivation

Linear Regression

- Consider a linear function $g : \mathbb{R}^d \to \mathbb{R}$, given by $g(x) = x^T \mu$
- $\mu \in \mathbb{R}^d$ is unknown.
- Given training data $(x_1, y_1), \cdots, (x_n, y_n) \in \mathcal{D} \subseteq \mathbb{R}^d \times \mathbb{R}$.
- $y_i = g(x_i) + \epsilon_i$ with $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$
- Obtain the least square estimate $\hat{\mu}$ of μ by minimizing *training* error
- Testing points are $z_1, z_2, \cdots, z_k \in \mathcal{D}$
- Mean square *testing* error: $\frac{1}{k} \sum_{i=1}^{k} \mathbb{E}(\hat{g}(z_i) - g(z_i))^2$

Testing points may be random or chosen by an adversary.

Linear Regression: Adversarial Version

- The learner chooses d training points $x_1, \cdots, x_d \in \mathcal{D}$.
- Learner can query the unknown function once at each point.
- Adversary chooses the following:
  - k and a set of testing points $z_1, \cdots, z_k$.
  - The noise variance σ_i^2 that corrupts each training point x_i
  - Noise is subject to $\sum_{i=1}^{d} \sigma_i^2 \leq \sigma^2$
- Learner minimizes the mean square testing error.
- `What are the best d training points?`

- Adversary chooses the worst case set of testing points.
- Minimize the worst case mean square testing error.
  - *Barycentric spanner*

Details of the proof

Barycentric Spanners

Consider $D \subseteq \mathbb{R}^d$, a *barycentric* spanner for D is a set of vectors $\{x_1, x_2, \cdots, x_d\} \subseteq D$, such that $\forall\, z \in D,\ \exists\, \mathbf{c} \in \mathbb{R}^d$, and

$$z = c_1 x_1 + c_2 x_2 + \cdots c_n x_n,$$

for $c_i \in [-1, 1]$ or $\|\mathbf{c}\|_\infty \leq 1$. And C-approximate barycentric spanner if $\|\mathbf{c}\|_\infty \leq C$ for some $C > 1$.

- **Applications:** Online bandit linear optimization[1,2,3], repeated decision making of approximable functions[4], John ellipsoid[5].

[1] Varsha Dani, Sham M Kakade, and Thomas P Hayes. "The price of bandit information for online optimization". In: *Advances in Neural Information Processing Systems*. 2008, pp. 345–352.

[2] Peter L Bartlett et al. "High-probability regret bounds for bandit online linear optimization". In: (2008).

[3] Varsha Dani, Thomas P Hayes, and Sham M Kakade. "Stochastic linear optimization under bandit feedback". In: (2008).

[4] Sham M Kakade, Adam Tauman Kalai, and Katrina Ligett. "Playing games with approximation algorithms". In: *SIAM Journal on Computing* 39.3 (2009), pp. 1088–1106.

[5] Sébastien Bubeck, Nicolo Cesa-Bianchi, and Sham M Kakade. "Towards minimax policies for online linear optimization with bandit feedback". In: *Conference on Learning Theory*. 2012, pp. 1–14.

Random Testing Points

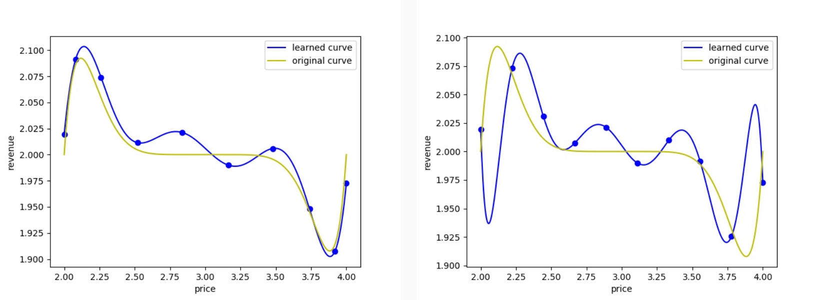


Figure 1: Linear regression with training points as barycentric spanner (left) vs equidistant points (right) in a 9th degree polynomial.

Performance measure

Root mean square error (RMSE) at testing points:

- 10 random points in the range of interest.
- 1000 equidistant points.

RMSE comparison

Training ↓ / Testing →	10 Random points [6]	1000 equidistant points
Barycentric spanner	0.0090	0.00939
Equidistant points	0.036	0.0241
10 Random points [7]	0.3169	0.2157

Table 1: RMSE at different testing points averaged over 500 seeds for 9th degree polynomial.

- Best RMSE at barycentric training points.
- Holds for different degrees.

[6] Testing points: 2.68, 3.84, 2.12, 3.96, 3.28, 3.76, 2.56, 2.76, 3.88, 2

[7] Training points: 2.72, 2.64, 2.12, 2.04, 3.44, 2.96, 2.99, 3.96, 2.24, 3.76

Dynamic pricing

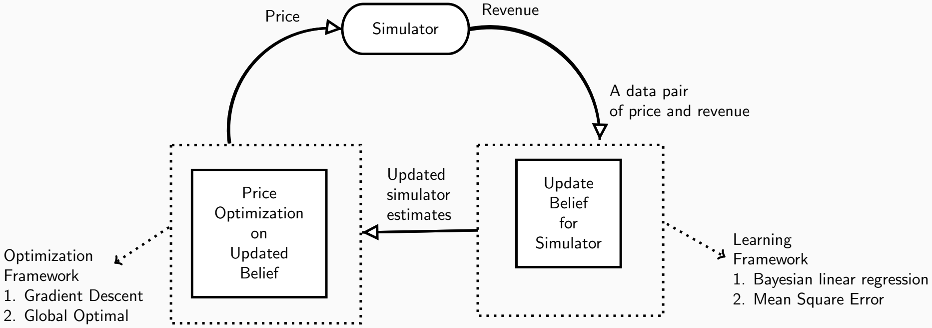
- Price for any given item on Amazon.com changes every 10 minutes[8].
- Prices subject to fluctuations depending on background algorithms.

[8] https://www.businessinsider.com/amazon-price-changes-2018-8?IR=T

Dynamic pricing

- Dynamic pricing in brick and mortar stores using electronic price tag.

Dynamic pricing



- **Goal:** Learn the optimal price in minimum number of steps
- Balance between optimization and learning.
- Use barycentric spanners for the initialization of Baysian updates.

Model Description

Consider the demand function (linear[9] in price)

$$d = \alpha - \beta p + \gamma + \epsilon, \ \textit{where } \epsilon \sim \mathcal{N}(0, \sigma^2). \tag{1}$$

$$Revenue = d \times p = w_0 + w_1 p + w_2 p^2 + \epsilon$$

The general form of revenue:

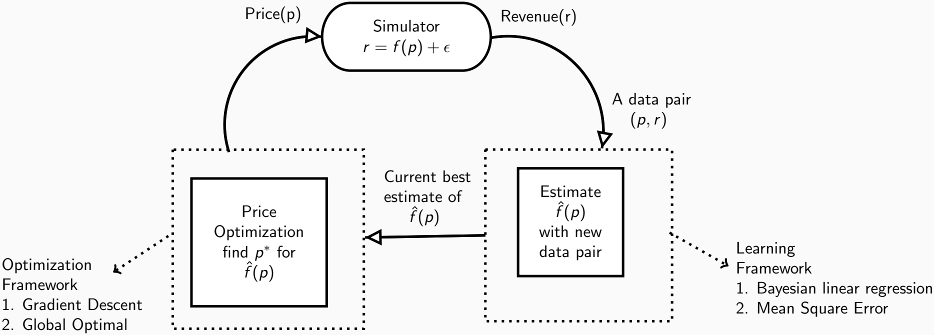
$$r = f(p) + \epsilon \tag{2}$$

where $f(p) = w_0 + w_1 p + w_2 p^2 + \cdots + w_n p^n$ is some n degree polynomial.

- Motivation for learning an unknown polynomial
- Barycentric spanner for polynomial feature set $D_n = \{\mathbf{p} := (1, p, p^2, \cdots, p^n), p \in [p_{min}, p_{max}]\}$ for $n \geq 1$.

[9] N Bora Keskin and Assaf Zeevi. "Dynamic pricing with an unknown demand model: Asymptotically optimal semi-myopic policies". In: *Operations Research* 62.5 (2014), pp. 1142–1167.

Model Flow



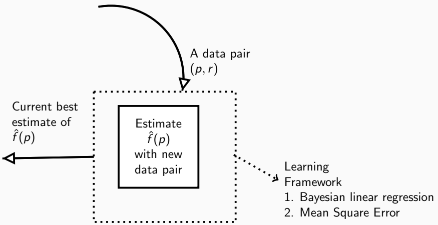
- Optimization framework
  - Global optimal: By using the roots of a polynomial
  - Gradient descent: Might get stuck in local optima

Learning Framework: Bayesian Linear Regression

- Assume a conjugate prior for weight vectors $\mathbf{w}$ of $f(p)$.
- Let prior $\mathbf{w} \sim \mathcal{N}(\mu, \mathbf{A})$ and error $\epsilon \sim \mathcal{N}(0, \sigma^2)$
- Using Bayes theorem, posterior turns out to be multivariate Gaussian with the following updates.

$$A_{n+1}^{-1} = A_n^{-1} + \frac{\mathbf{p_{n+1}} \mathbf{p_{n+1}}^T}{\sigma^2}$$

$$A_{n+1}^{-1} \mu_{n+1} = A_n^{-1} \mu_n + \frac{r_{n+1} \mathbf{p_{n+1}}}{\sigma^2}$$



where $\mathbf{p_i} = [1, p, p^2, \cdots, p^n]$ in ith iteration.

Performance

$$\text{Regret } R_t = \hat{\mu}^T p_{best}^* - simulator(p_t^*).$$

Thompson Sampling for dynamic pricing

Algorithm 1: BLR-TS(n, $\hat{\mu}$)

Initialization: Set $D = \{\mathbf{p} := (1, p, p^2, \cdots, p^n), p \in [0,1]\}$;

1. Find barycentric spanner $b_1, b_2, \cdots, b_n$ for D;
2. Set $A_0^{-1} = \sum_{i=1}^{n} b_i b_i^T$ and sample $\mathbf{w}_0 \sim \mathcal{N}(\mu_0 = 0, A_0)$;
3. Set $f_0(p) = \mathbf{w}_0^T \mathbf{p}$ and find $p_0^* = \arg\max_{0 \le p \le 1} f_0(p)$;
4. Set $\mathbf{p_0} = [1, p_0^*, (p_0^*)^2, \cdots (p_0^*)^n]$, $t = 0$ and $C_t = 0$;

while $t \le$ *total_iterations* **do**

Simulation: $r_t \leftarrow$ simulator(p_t^*);

Learning: $A_{t+1}^{-1} = A_t^{-1} + \frac{\mathbf{p_t}\mathbf{p_t}^T}{\sigma^2}, A_{t+1}^{-1}\mu_{t+1} = A_t^{-1}\mu_t + \frac{r_t \mathbf{p_t}}{\sigma^2}$;

Sampling: $\mathbf{w}_{t+1} \sim \mathcal{N}(\mu_{t+1}, A_{t+1})$ and set $f_{t+1}(p) = \mathbf{w}_{t+1}^T \mathbf{p}$;

Optimization: Find $p_{t+1}^* = \arg\max_{0 \le p \le 1} f_{t+1}(p)$;

Regret: $R_t = \hat{\mu}^T p_{best}^* - simulator(p_t^*)$, $C_t = C_t + R_t$;

Set $t \leftarrow t + 1$;

end

Performance comparison

- Consider a fourth degree revenue curve $r = -p^4 + 22p^3 - 165p^2 + 480p - 150 + \epsilon$.
- Two choices for initialization:
  1. $A_0^{-1} = I$.
  2. A_0^{-1} is obtained using barycentric spanner as in Algorithm.
- Cumulative regrets averaged over 10 sample path.

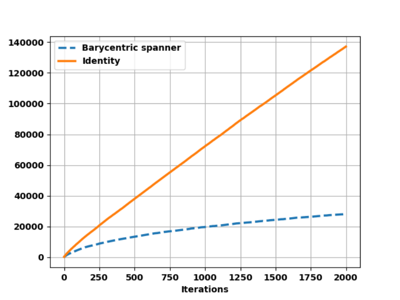


Figure 2: Regret comparison for identity vs barycentric spanner initialization.

Barycentric spanner leads to much lower regret.

Robustness check - I

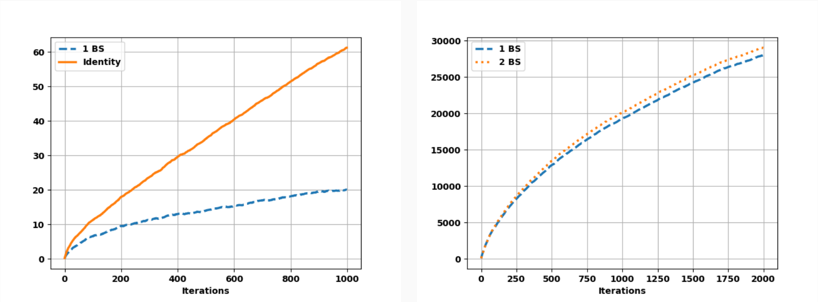


Figure 3: Regret comparison for a 2nd degree polynomial (left) and for 2-approximate barycentric spanner (right).

- Different polynomial degrees.
- Approximate barycentric spanner.

Robustness check - II

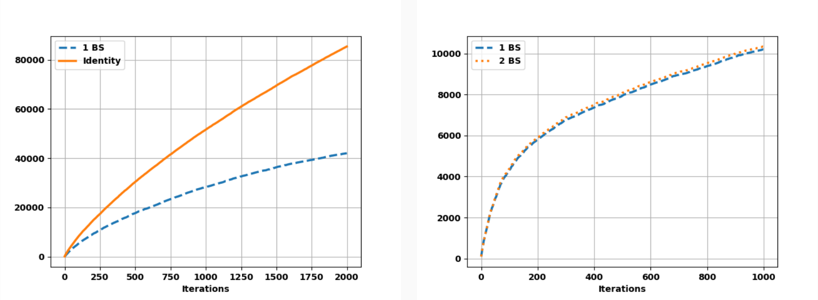


Figure 4: Regret comparison for learning a 4nd degree polynomial with 7th degree model (left) and for radial basis function (right) $100 * e^{-(x-5)^2}$.

- Different degree for the polynomial vs model.
- Non-polynomial models - radial basis function.

Other learning methods

- Iterated least square (ILS) by minimizing the mean square error.
- Constrained iterated least square (CILS)[10].
    - In each period t, CILS with threshold parameter k charges the price:

$$p_t = \begin{cases} \bar{p}_{t-1} + sgn(\delta_t) k t^{-1/4} & \text{if } |\delta_t| < k t^{-1/4} \\ \text{ILS price} & \text{otherwise.} \end{cases}$$

    - CILS induces exploration.

[10] N Bora Keskin and Assaf Zeevi. "Dynamic pricing with an unknown demand model: Asymptotically optimal semi-myopic policies". In: *Operations Research* 62.5 (2014), pp. 1142–1167.

Expected regret comparison

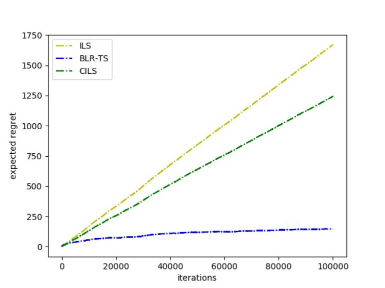


Figure 5: Expected regret under different learning methods

- Thompson sampling results in the best performance.
- Drawback: sampling despite learning the true optimal.

Stopping Criterion

- Include a stopping criterion for sampling.
- Stopping criterion based on moving average of previous prices.
- BLR with a stopping criterion (partial greedy) gives the best regret.
- Parameter settings: Linear demand model[11]

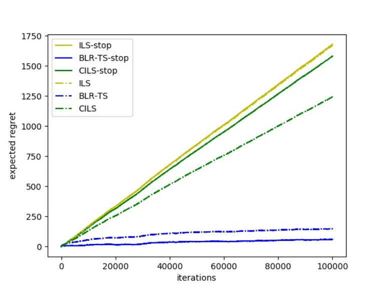


Figure 6: Expected regret under different learning methods

[11] N Bora Keskin and Assaf Zeevi. "Dynamic pricing with an unknown demand model: Asymptotically optimal semi-myopic policies". In: *Operations Research* 62.5 (2014), pp. 1142–1167.

Characterization of barycentric spanner

State of art

- The only known algorithm[12] to compute C-approximate barycentric spanner for a general compact set $D \subset \mathbb{R}^d$
- $O(d^2 \log_C d)$ calls to an optimization oracle for performing linear optimization over D.
- Complexity diverges as C approaches 1.
- We are interested in the polynomial feature set D_n for some $n \geq 1$:

$$D_n := \{[1, p, p^2, \cdots p^n] : p \in [p_{\min}, p_{\max}]\}$$

Main result

Two different characterizations of barycentric spanner for set D_n.

[12] Baruch Awerbuch and Robert Kleinberg. "Online linear optimization and adaptive routing". In: *Journal of Computer and System Sciences* 74.1 (2008), pp. 97–114.

Characterization of barycentric spanner

Suppose $\mathbf{p} \in \mathbb{R}^{n+1}$ is such that $p_{\min} \leq p_1 \leq \cdots \leq p_{n+1} \leq p_{\max}$ and $f_n(p) = [1, p, p^2, \ldots, p^n]^{\mathrm{T}}$. Then, the following are equivalent:

1. The set $\{f_n(p_1), \ldots, f_n(p_{n+1})\} \subset D_n$ is a barycentric spanner for D_n.
2. The vector $\mathbf{p}$ satisfies $p_{\min} = p_1 < p_2 \cdots < p_{n+1} = p_{\max}$ and

$$\sum_{\substack{1 \leq j \leq n+1, \\ j \neq i}} \frac{1}{p_i - p_j} = 0, \; i = 2, \ldots, n. \tag{3}$$

First characterization

Barycentric spanner for set D_n as a set of non-linear equations.

The following are equivalent:

1. The set $\{f_n(p_1), \ldots, f_n(p_{n+1})\} \subset D_n$ is a barycentric spanner for D_n.
2. The vector $\mathbf{p}$ is the unique global solution of the optimization problem

$$\max_{\substack{\mathbf{w} \in \mathbb{R}^{n+1} \\ p_{\min} = w_1 < \cdots < w_{n+1} = p_{\max}}} \ln|\det V(\mathbf{w})|. \tag{4}$$

where $V(\mathbf{w}) := [f_n(w_1), \ldots, f_n(w_{n+1})]$ is the $(n+1) \times (n+1)$ Vandermonde matrix formed from the elements of $\mathbf{w}$.

Second characterization

Barycentric spanner for set D_n as a convex optimization problem.

Proof of characterization.

Major Steps

- The determinant of the Vandermonde matrix $V(\mathbf{p})$ equals

$$det(V(\mathbf{p})) = \prod_{1 \leq i < j \leq n+1} (p_j - p_i). \tag{5}$$

- First order conditions for optimization.
- If $p_i \neq p_j \ \forall \ i \neq j$, Then $c_1, \ldots, c_{n+1} \in \mathbb{R}$ satisfy

$$c_1 f_n(p_1) + \cdots + c_{n+1} f_n(p_{n+1}) = f_n(s) \tag{6}$$

iff c_i is the ith Lagrange basis polynomial $\{p_1, p_2, \ldots, p_{n+1}\}$ given by

$$l_i(s, \mathbf{p}) := \frac{\prod_{j \neq i} (s - p_j)}{\prod_{j \neq i} (p_i - p_j)} \tag{7}$$

Nature of the solution

- The set D_n has a unique barycentric spanner.
- Can be found by either of the following methods:
  - Non-linear equations.
  - Convex optimization problem.
- Symmetric solution.
  - Can be exploited for faster computation.
- Shift and scale property.
  - Only need to compute the barycentric spanner for canonical case i.e interval [0,1].

Empirical Comparison of Run Times

n	A-K[13]			Non linear equations		Convex optimization	
	C=1	C=2	C=5	Full	Reduced	Full	Reduced
2	0.097	0.097	0.097	0.0002	0.00003	0.0209	0.0154
5	4.537	0.372	0.372	0.0007	0.0004	0.0713	0.0478
10	35.185	2.891	2.698	0.0081	0.0025	0.2296	0.1517
13	53.752	5.537	5.467	0.0158	0.0038	0.3678	0.2087
15	65.656	8.163	7.937	0.0316	0.0081	0.4853	0.2691
20	115.45	19.13	18.93	0.0793	0.0241	0.9198	0.4967
22	NA	NA	NA	0.1073	0.0351	1.1172	0.6056

Table 2: Time in seconds for computing a barycentric spanner

- A-K algorithm is implemented in an efficient way.

[13] Baruch Awerbuch and Robert Kleinberg. "Online linear optimization and adaptive routing". In: *Journal of Computer and System Sciences* 74.1 (2008), pp. 97–114.

Time to compute barycentric spanner for higher degree

n	Non linear equations		Convex optimization	
	Full	Reduced	Full	Reduced
25	0.206	0.068	1.933	0.804
30	0.415	0.099	2.126	1.278
45	2.305	0.377	4.656	2.527
60	6.985	1.534	9.618	5.975
80	24.676	3.299	15.636	8.196

Table 3: Time in seconds to compute barycentric spanner for higher degrees n.

Scalability of our approach where A-K algorithm fails.

Summary

- A characterization of the barycentric spanner.
  - Univariate polynomial feature set.
- Motivation
  - Linear regression.
  - Dynamic pricing.
- Properties of barycentric spanner
  - Symmetry and affine transformation.
- Usage of the barycentric spanner for initializing covariance updates in Thompson sampling.

Work in progress

- Extend the results on finding barycentric spanner for multi-variate polynomials.
- Dynamic pricing.
    - Include additional features such as seasonality etc.
    - To cater non-stationary demand.
    - Theoretical guarantee in terms of bounds.
    - Try bandit algorithms.
    - Performance of explore and commit algorithm
- Connection with volumetric spanner.

Thank you!

manukumar.gupta@tcs.com

Proof of adversarial linear regression

$$\frac{1}{k}\sum_{i=1}^{k}\mathbb{E}(\hat{g}(z_i)-g(z_i))^2=\frac{\sigma^2}{k}\|X^{-1}Z\|_F^2 \tag{8}$$

$$\frac{1}{k}\sum_{i=1}^{k}\mathbb{E}(\hat{g}(z_i)-g(z_i))^2=\frac{1}{k}\sum_{j=1}^{k}\left[\sigma_1^2(e_1^TX^{-1}z_j)^2+\cdots+\sigma_d^2(e_1^TX^{-1}z_j)^2\right] \tag{9}$$

The adversary can ensure the worst case mean-square error for a given choice of X by setting $k=1$.

Back to Linear Regression.

Proof of barycentric spanner characterization

Let $s \in [p_{\min}, p_{\max}]$ and suppose $p_1, \ldots, p_{n+1} \in [p_{\min}, p_{\max}]$ are such that $p_i \neq p_j$ for all $i \neq j$. Then $c_1, \ldots, c_{n+1} \in$ satisfy

$$c_1 f_n(p_1) + \cdots + c_{n+1} f_n(p_{n+1}) = f_n(s) \tag{10}$$

if and only if $c_i = l_i(s, \mathbf{p})$ for each $i = 1, \ldots, n+1$, where $\mathbf{p} = [p_1, \ldots, p_{n+1}]^{\mathrm{T}}$, and $l_i(\cdot, \mathbf{p})$ is the ith Lagrange basis polynomial for the points $\{p_1, p_2, \ldots, p_{n+1}\}$ given by

$$l_i(s, \mathbf{p}) := \frac{\prod_{j \neq i} (s - p_j)}{\prod_{j \neq i} (p_i - p_j)} \tag{11}$$

$$\left.\frac{\partial l_1(s, \mathbf{p})}{\partial s}\right|_{s = p_1} = \sum_{j \neq 1} \frac{1}{p_1 - p_j} < 0. \tag{12}$$

Back to BS characterization.