Recent development in computational convex analysis

Yves Lucet

OKANAGAN

October 25, 2010

Outline

Convex Transforms

$$f^*(s) = \sup_x \langle s, x\rangle - f(x)$$

$$M_\lambda f(x) = \inf_y [f(y) + \frac{\|x-y\|^2}{2\lambda}]$$

$$h_{\mu,\lambda} f(x) = -M_\mu(-M_\lambda f(x))$$

$$\mathcal{P}_\lambda(f_0, f_1) = [(1-\lambda)M_1(f_0^*) + \lambda M_1(f_1^*)]^* - \frac{1}{2}\|\cdot\|^2$$

$$p_\mu(f_0, f_1; \lambda) = -M_{\mu+\lambda(1-\lambda)}\left(-[(1-\lambda)M_\mu f_0 + \lambda M_\mu f_1]\right)$$

$$P_\lambda(f_1, f_2)(x) = \inf_{(1-\lambda_0)y_0+\lambda y_1 = x} [(1-\lambda)f_0 + \lambda f_1 + \lambda(1-\lambda)g(y_0 - y_1)]$$

$$(f \oplus g)(x) = \inf_y [f(y) + g(x-y)]$$

OKANAGAN

Convex Operators

Core

- Addition, scalar multiplication
- Fenchel Conjugate or Moreau envelope

Composite

- Lasry-Lions double envelope
- Proximal Average

Specialized

- Fitzpatrick Functions
- Convex Envelope

Computational Convex Analysis

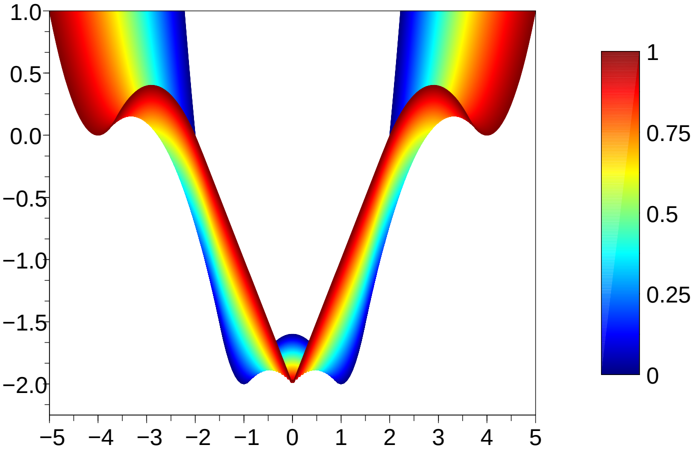
Convex Calculus

- addition, scalar multiplication
- Convex envelope co$f$
- Conjugate f^*
- Moreau envelope $M_\lambda(f)$
- Proximal average, nonconvex proximal average, Lasry-Lions double envelope, (convex) inf-convolution, etc.

Specialized transforms

- Fitzpatrick functions
- Nonconvex inf-convolution
- Kernel average

Nonconvex Extensions



Symbolic Computation

Solve $\nabla f(x) = s$ for x symbolically.

Symbolic Packages

- Symbolic Computation of Multidimensional Fenchel Conjugates, Borwein & Hamilton, 2006. SCAT (Symbolic Convex Analysis Toolkit) package in Maple.
- Symbolic Computation of Fenchel Conjugates, Bauschke & Mohrenschildt, 2006

Properties

√ Great to study examples and avoid computation errors.

X No close form exists for some functions e.g. polynomial of degree greater or equal to 6.

Fast Algorithms

Discretize: $f^*(s_j) = \max_{x_i}[s_j x_i - f(x_i)]$

Fast Algorithms

- Linear-time Legendre Transf. LLT (Lucet 96)
- Parabolic Envelope PE (Felzenszwalb & Huttenlocher 04)
- NonExpansive Prox NEP (Lucet 06)
- Parametric Legendre Transf. PLT (Hiriart-Urruty & Lucet 06)

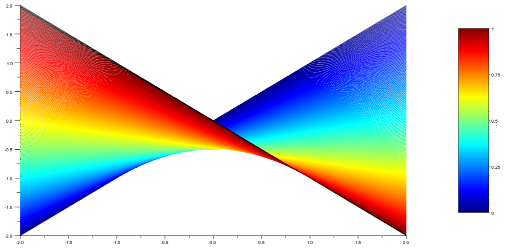
Properties

- √ Linear-time
- √ Scale linearly
- X Domain modeling and approximating quadratic functions

PLQ vs. Fast Algorithms: Prox. Avg of x and $-x$

Proximal average

$$\mathcal{P}_\lambda(f_0, f_1)(x) = (1-\lambda)(-x) + \lambda x - 2\lambda(1-\lambda)$$



PLQ vs. Fast Algorithms: Prox. Avg of x and $-x$

Fast Algorithms

$$\begin{cases} \infty & \text{if } |x| > b, \\ (1-\lambda)(-x) + \lambda x - 2\lambda(1-\lambda) & \text{if } 2(1-\lambda) - b \le x \le b - 2\lambda, \\ \frac{\lambda}{2(1-\lambda)} x^2 + \frac{\lambda+\lambda b-1}{1-\lambda} x - \frac{\lambda b(4\lambda+b-4)}{2(1-\lambda)} & \text{if } -b \le x \le 2(1-\lambda) - b, \\ \frac{1-\lambda}{2\lambda} x^2 + \frac{\lambda-b+\lambda b}{\lambda} x + \frac{b(4\lambda^2+b-\lambda b-4\lambda)}{2\lambda} & \text{if } b - 2\lambda \le x \le b \end{cases}$$

PLQ

$$\mathcal{P}(f_0, \lambda, f_1) = (1-\lambda)(-x) + \lambda x - 2\lambda(1-\lambda)$$

Piecewise Linear-Quadratic Functions

Approximate function with quadratic spline

Definition

- Domain is the intersection of linear functions
- On each piece, the function is quadratic
- Restrict to continuous functions on ri dom$f$.

Properties

- √ Closed class under convex operator
- √ Infinite domains can be modeled.
- √ Hybrid symbolic numerical algorithms running in Linear-time.
- X Restricted to univariate functions (for now).

Fast vs. PLQ

Fast

- Linear spline approximates PLQ functions
- Linear time algorithms, very fast
- Can model nonconvex PLQ functions
- scale to dimension

PLQ

- Quadratic spline models PLQ functions
- Linear time algorithms
- Can model nonconvex PLQ functions

Outline

Goebel's Graph-Matrix Calculus

Conjugate

$$s \in \partial f(x) \iff x \in \partial f^*(s)$$

$$(x, s) \in \operatorname{gph} \partial f \iff (s, x) \in \operatorname{gph} \partial f^*$$

$$\operatorname{gph} \partial (f^*) = \begin{bmatrix} 0 & \mathrm{Id} \\ \mathrm{Id} & 0 \end{bmatrix} \operatorname{gph} \partial f$$

Moreau envelope

$$\operatorname{gph} \partial M_\lambda(f) = \begin{bmatrix} \mathrm{Id} & \lambda\,\mathrm{Id} \\ 0 & \mathrm{Id} \end{bmatrix} \operatorname{gph} \partial f$$

Binary operators

$$(x,s)\in \operatorname{gph}\partial(f_1+f_2) \Leftrightarrow \exists (x_i,s_i)\in \operatorname{gph}\partial f_i \text{ such that } \begin{cases} x = x_1 = x_2, \\ s = s_1 + s_2. \end{cases}$$

$$(x_1+x_2,s)\in \operatorname{gph}\partial(f_1 \Box f_2) \Leftrightarrow (x_i,s)\in \operatorname{gph}\partial f_i.$$

$$(x,s)\in \operatorname{gph}\partial\mathcal{P}_\lambda(f_1,f_2) \Leftrightarrow \begin{cases} x = (1-\lambda)x_1 + \lambda x_2, \\ s = x_1 + s_1 - x = x_2 + s_2 - x. \end{cases}$$

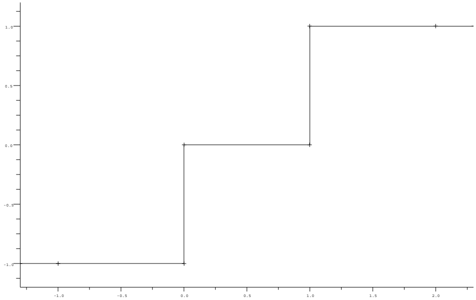
Data structure

GPH matrix

$$\begin{bmatrix} \bar{x}_0 & x_1 & \cdots & x_n & \bar{x}_{n+1} \\ \bar{s}_0 & s_1 & \cdots & s_n & \bar{s}_{n+1} \\ \bar{y}_0 & y_1 & \cdots & y_n & \bar{y}_{n+1} \end{bmatrix}.$$

Example

Subdifferential of the function $f(x) = -x$ for $x \leq 0$, $f(x) = 0$ when $0 \leq x \leq 1$, and $f(x) = x - 1$ if $x \geq 1$.



Graph Calculus

Conjugate

Given $G = [x; s; y]$ GPH matrix representing f, f^* has GPH matrix

$$\begin{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} * \begin{bmatrix} x \\ s \end{bmatrix} \\ s.*x - y \end{bmatrix} = \begin{bmatrix} s \\ x \\ s.*x - y \end{bmatrix}$$

Moreau envelope

$M_\lambda(f)$ has GPH matrix

$$\begin{bmatrix} \begin{bmatrix} 1 & \lambda \\ 0 & 1 \end{bmatrix} * \begin{bmatrix} x \\ s \end{bmatrix} \\ y + \frac{\lambda}{2} s.*s \end{bmatrix} = \begin{bmatrix} x + \lambda s \\ s \\ y + \frac{\lambda}{2} s.*s \end{bmatrix}$$

GPH for Proximal Average

f_1, f_2 convex PLQ functions with GPH matrix $G_1 = [x_1; s_1; y_1]$ and $G_2 = [x_2; s_2; y_2]$. Set $\lambda_1 = 1 - \lambda$, $\lambda_2 = \lambda$ then $\mathcal{P}_\lambda(f_1, f_2)$ admits the GPH matrix $G = [x; s; y]$ where $x = \lambda_1 x_1 + \lambda_2 P x_1$, $s = x_1 + s_1 - x$,

$$y = \lambda_1 (y_1 + \frac{1}{2} x_1 .* x_1) + \lambda_2 (y_{Px_1} + \frac{1}{2} P x_1 .* P x_1) - \frac{1}{2} x .* x$$

with

$$P = (\mathrm{Id} + \partial f_2)^{-1} (\mathrm{Id} + \partial f_1),$$

and $y_{Px_1} = f_2(Px_1)$.

Piecewise quadratic function vs. piecewise linear subdifferential

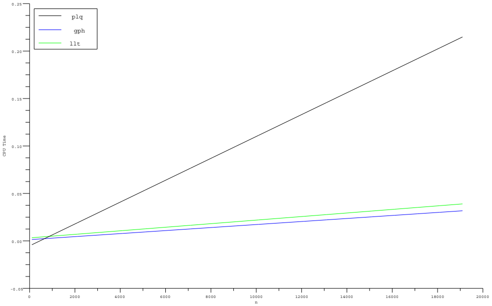
PLQ

- Model PLQ functions as piecewise quadratic polynomial
- Linear time algorithms
- Can model nonconvex PLQ functions

GPH

- Model convex PLQ functions
- Store the graph of the subdifferential as a finite set of points
- Linear time algorithms reducing to matrix multiplications
- Same advantages as PLQ algorithms with computation time of Fast algorithms.

PLQ vs. GPH


plq
gph
llt
CPU Time
n
0.25
0.20
0.15
0.10
0.05
0.00
-0.05
0
2000
4000
6000
8000
10000
12000
14000
16000
18000
20000

Outline

OKANAGAN

Convex envelope

$$M_\lambda f(x) = \frac{\|x\|^2}{2\lambda} - \frac{1}{\lambda}\left(\frac{\|\cdot\|^2}{2} + \lambda f\right)^*(x),$$
$$f^*(s) = \frac{\|s\|^2}{2} - \lambda M_\lambda\left(\frac{1}{\lambda} f - \frac{\|\cdot\|^2}{2\lambda}\right)(s),$$

Key idea

1. compute the convex envelope
2. apply convex operators

For Fast algorithms (linear spline), use Beneath-Beyond algorithm.
For GPH algorithms, convert to PLQ format.

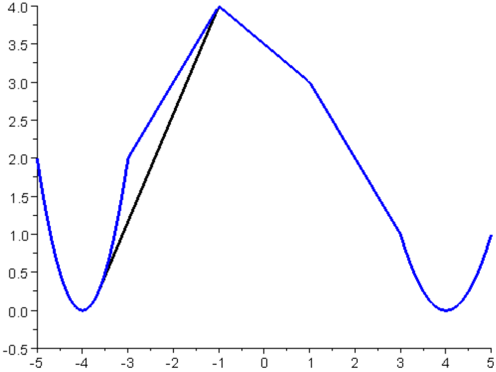
Convex envelope of the maximum

Decompose $f = \min f_i$ where $f_i(x) = f(x) + I_{[x_i, x_{i+1}]}$ to get

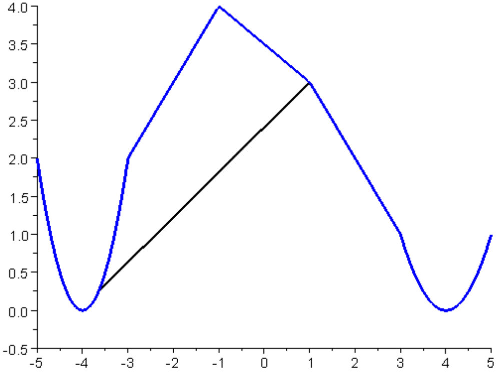
$$\operatorname{co} f = \operatorname{co} \min_i f_i = \operatorname{co} \min_i \operatorname{co} f_i = [\max_i (\operatorname{co} f_i)^*]^*.$$

1. Split f into f_i
2. Compute $\operatorname{co} f_i$ for each $i = 0, \ldots, n$ by replacing any f_i with $a_i < 0$ with the line going through $(x_i, f_i(x_i))$ and $(x_{i+1}, f_i(x_{i+1}))$; any f_i with $a_i \geq 0$ is convex so $\operatorname{co} f_i = f_i$
3. Compute $f_i^* = (\operatorname{co} f_i)^*$ using the PLQ conjugate algorithm for convex functions
4. Compute $\max f_i^*$ directly (it is a convex function)
5. Compute the conjugate of $\max f_i^*$ to obtain $\operatorname{co} f$

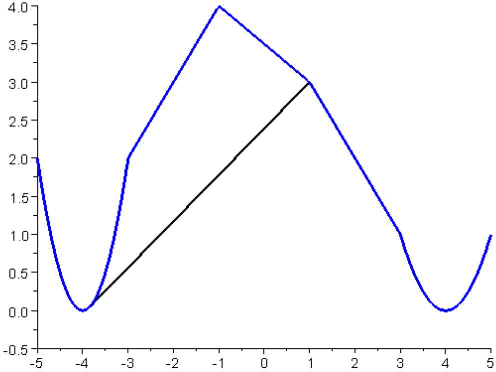
Convex envelope by extending the Beneath-Beyond algorithm



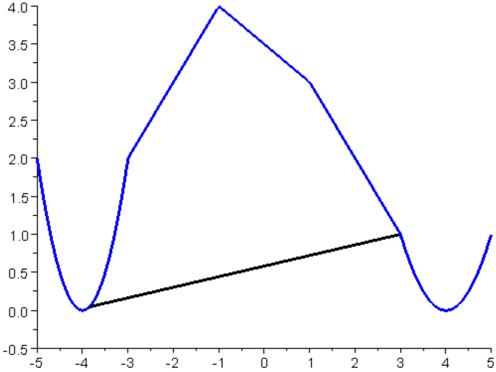
Convex envelope by extending the Beneath-Beyond algorithm



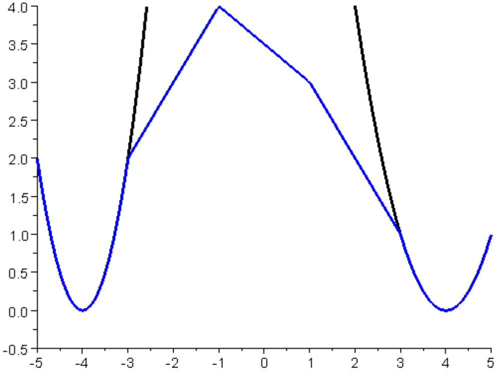
Convex envelope by extending the Beneath-Beyond algorithm



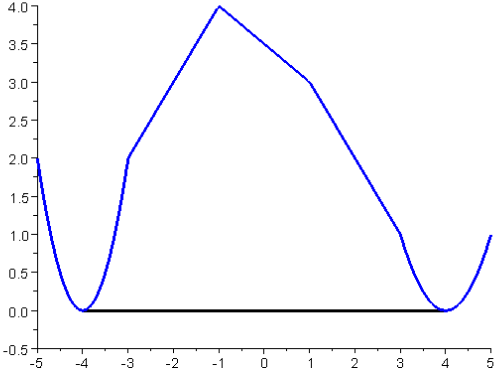
Convex envelope by extending the Beneath-Beyond algorithm



Convex envelope by extending the Beneath-Beyond algorithm



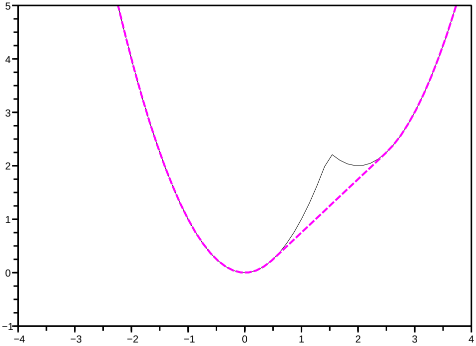
Convex envelope by extending the Beneath-Beyond algorithm



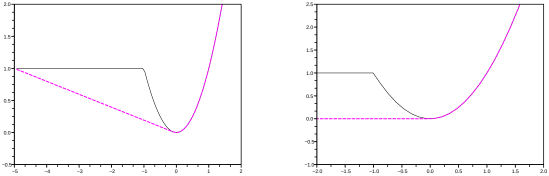
Convex envelope of 2 quadratic functions

- L-L case
- Q-L
- L-Q
- Q-Q

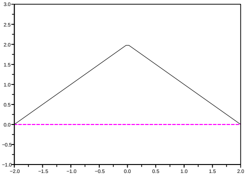
Q-Q case

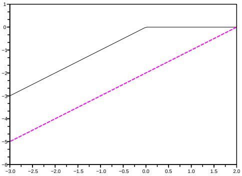


L-Q case

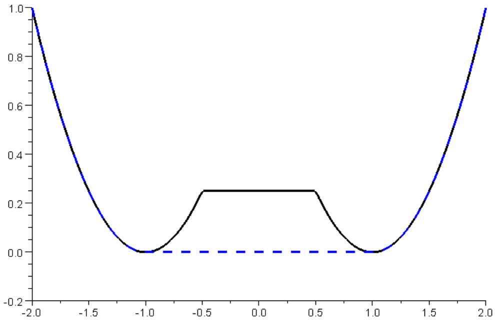

2.0
1.5
1.0
0.5
0.0
−0.5
−5
−4
−3
−2
−1
0
1
2
2.5
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0

L-L case


3.0
2.5
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0


1
0
−1
−2
−3
−4
−5
−6
−3.0 −2.5 −2.0 −1.5 −1.0 −0.5 0.0 0.5 1.0 1.5 2.0

Q-L-Q case



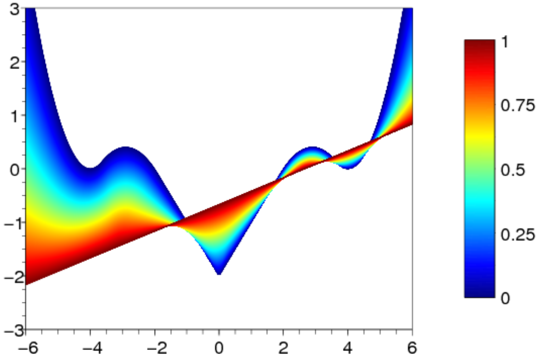
Performance

Complexity

- plq_coSplit has quadratic complexity
- plq_coDirect has linear complexity

But with linear spline, plq_coSplit shows linear complexity experimentally.

Example



Outline

Conclusion

Current status

- CCA toolbox fairly complete for univariate functions
- Missing algorithms: nonconvex inf-convolution, kernel average
- Bivariate functions coming soon

Conclusion

Summary

- Y. Lucet, *What Shape is your Conjugate? A Survey of CCA and its Applications*, SIAM OPT 2009
- B. Gardiner & Y. Lucet, *Graph-Matrix Calculus for Computational Convex Analysis*, Special Issue on Fixed-Points operators, 2010
- B. Gardiner & Y. Lucet, *Convex Hull Algorithms for Piecewise Linear-Quadratic Functions in Computational Convex Analysis*, Set-Valued and Variational Analysis, 2010, 1-16
- CA^2: CCA numerical library (GPL) http://people.ok.ubc.ca/~ylucet/CCA/

UBC Okanagan Campus

- Home to OCANA Optimization Group:
  H. Bauschke, W. Hare, Y. Lucet, S. Wang
  Mathematics, Computer Science, Engineering
- Three Optimization Research Laboratories!
- Located in beautiful Kelowna (Beaches, Wineries, Skiing, ...)!
- Graduate Funding Available!