Symbolic Subdifferentiation in Python (SPY)

Maurizion Caló and Jaehyun Park
EE 364B: Convex Optimization II Final Project

Subgradient-PY (SPY)

We’ve implemented a Python package Subgradient-PY (SPY) that solves convex optimization problems using subgradient methods. Some features of SPY include:

- The objective function and constraints need not be differentiable.
- Due to little space usage, SPY can be used to solve large problems.
- SPY can be combined with primal or dual decomposition techniques to develop distributed algorithms.

Subgradients

g is a subgradient of f (not necessarily differentiable) at x if

\[ f(y) \geq f(x) + g^T (y-x) \]

holds for all y. We write g ∈ ∂f(x) if g is a subgradient of f at x.

Computation of Subgradients

Computing subgradients can be done recursively, and the procedure is analogous to the chain rule for differentiable functions. For each library function, we implemented a method that computes its subgradient at a given point. Subgradients of more complicated expressions can be computed using the composition rule:

- Let f(x) = h(f₁(x),...,fₖ(x)) with h convex non-decreasing, fᵢ convex.
- Find c ∈ ∂h(f₁(x),...,fₖ(x)) and gᵢ ∈ ∂fᵢ(x).
- Then, g = c₁g₁ + c₂g₂ + ... + cₖgₖ is a subgradient of f at x.

Subgradient Methods

The subgradient method is similar to gradient descent methods for minimizing differentiable functions.

- At x(k), find g(k) ∈ ∂f(x(k)).
- Set the next point as x(k+1) := x(k) − αₖg(k), where αₖ is k-th step size.

Example Code

```python
from spy import *
x = var('x')
y = var('y')
ex = max(x + y, 2 * x - y) + huber(x, 1)
constraints = [geq(x, y), leq(norm2([x, y]), 1)]
prob = minimize(ex, constraints)
(optval, optpoint) = prob.solve()
```

Using SPY

- Declaring variables:
  One can declare a variable using the following syntax.
  ```python
  x = var('x')
  ``
- Forming expressions:
  The following line creates an expression named ex.
  ```python
  ex = abs(x - 3) + exp(x)
  ``

Expression Class

- The current system only supports scalar variables and scalar-valued expressions.
- Internally, an expression is stored as a tree.

![Expression Class Diagram]

Constraint Class

A constraint is an inequality of the form (convex) ≤ (concave) or an equality of the form (affine). Constraints specify the set of values that the variables can take.

Problem Class

A problem is a triplet of the form (minimize/maximize, objective, list of constraints). The problem class contains a solve method which solves the specified problem using subgradient methods.

Limitations and Extension Ideas

- SPY does not work well with equality constraints (yet).
- Convergence behavior depends on the step sizes, so it is up to users to give a “right” step size rule.
- Currently, SPY cannot detect if the given problem is unbounded. Extending SPY to detect unboundedness will be a great deal of work, but it will be useful.