Solving Very Large or Infinite Problems: Constraint Generation

Optimization - 10725
Carlos Guestrin
Carnegie Mellon University

February 6th, 2008

Weighted Least-Squares

- Least-squares regression problem:
  - Basis functions:
  - Find coefficients:
- Some points are more important than others:
  - Weighted least-squares:

Robust Least Squares*

- Weighted least squares:
- Test set distribution may be different from training set!
  - Must reweigh according to likelihood ratio:
- But what is the test set distribution???
- Don't want to commit!
  - Pick worst case weights!
  - Robust LS:

* many other optimization problems are called robust least-squares… :)

Optimization of Robust LS

- Robust LS problem:
- For each set of weights, must solve weighted least squares:
- How do we find worst case weights?
  - Option B : guess weights, solve least squares, tweak weights,…

Equivalent optimization problem

- Robust LS:
- Pushing min **w** into constraint:
- Non-linear constraint, give up!

Minimum over **w** as infinite constraints

- Non-linear min constraint:
- Infinite constraint set:
- Great! Had a non-linear constraint, now all I have are infinite constraints, for each alpha!

Constraints for one alpha, help with other alphas

- Suppose you have α_0, and introduce a constraint for some coefficients $\mathbf{w}_0$:
- Constraint also upper bound for other weights α:
- Linear constraint! Cool!

A geometric view

- We have an infinite number of linear constraints, many are irrelevant
  - Set of constraints forms a convex set*
- Linear program with one constraint per w
  - Still infinite…

* There is better machinery to understand this, but more when we talk about convexity

Suppose we use a subset of the constraints

- What if we use a finite number of constraints
  - Set of constraints at a finite set of coefficients Ω
- Can solve with any LP solver!
- But, solution with subset of constraints may not be a solution to original problem
  - Fewer constraints, solution may be infeasible, value of LP too high…

Active constraints

- Original LP with infinite constraints:
- How many variables?
- How many active constraints at optimal solution?
- So, if we knew set of active constraints at optimal solution Ω^*
  - Could discard all other constraints

Active Constraints at Optimal Point

- Original problem:
- If we knew set of active constraints at optimal solution Ω^*
  - Could discard all other constraints
  - Solution will be feasible with respect to original problem
- Consider some set of constraints Ω:
  - Too few, infeasible solution:
  - Just right, feasible solution:

Constraint Generation

- Start with some finite set of constraints Ω
  - Solve LP, obtain α
- Check is (ε,α) is feasible for infinite constraints:
  - If feasible, done!
  - Otherwise, add a constraint that makes (ε,α) infeasible:
- But how do we find which constraint to add???

Separation Oracle for Robust LS

- Original problem:
- Is (ε,α) feasible?
  - infeasibility ➔ ε too high for this particular α
- What's the smallest possible ε?
- Standard weighted LS!
  - If result is ε, then we are done!
  - Otherwise found a violated constraint

Constraint Generation: The General Case

- Given an LP with (possibly infinitely) many constraints:
- Start with some subset of the constraints
- Solve LP to find a solution with new subset of the constraints:
- Separation oracle:
  - If x is feasible:
  - If x is infeasible:
- Add violated constraint to set
- (It is also possible to remove (some or all) inactive constraints, in addition to adding violated constraints)
  - Makes LP solver step faster
  - But requires more outer loop iterations
  - Trade-off is application specific

Are we there yet?

- When do we stop?
- Solve with infinite set of constraints:
  - Obtain ($\varepsilon_{OPT},\alpha_{OPT}$)
- Solve with constraints Ω
  - Obtain (ε,α)
- Optimizing subset of constraints, same objective
  -
- If we get any feasible point with infinite constraints
  - E.g.,
- Bound on how far we are from optimal solution:

Bound on optimal solution - General case

- Problem with many constraints:
- Some relaxation:
  - E.g., only subset of constraints
- If you can obtain some feasible point for the original problem:
- Bound on the optimal solution:

Practicalities of Constraint Generation

- Constraint generation converges in a finite number of iterations if the original set is finite
  - Can't guarantee fast rate, similar to simplex algorithm
  - Infinite case: will get arbitrarily close, but not necessarily to the optimum
- Idea of using relaxations to obtain bounds is very useful in general
  - E.g., useful in duality (more later in the semester)
- Separation oracle:
  - Must find some violated constraint
  - If we find most violated constraint, usually faster
  - Also very useful for proving that LPs can be solved in polytime (ellipsoid algorithm, more later)
- Constraint generation is extremely useful in practice
  - Often, e.g., robust LS, we have a poly-time separation oracle, even if there are exponentially or infinitely many constraints
  - Even if polynomially many constraints, a fast oracle can make constraint generation faster than using a standard solver
- Constraint generation can be useful for solving general convex problems, not just LP
- Remember: most LP solvers allow you to start from previous solution
  - (the one found with fewer constraints)
  - Make sure you do this, otherwise approach will be much much much slower