1 Primal-Dual method

Suppose we want to solve an optimization problem that is formulated as a “covering” integer program:

\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad Ax \geq b, \\
& \quad x \geq 0, \\
& \quad x \text{ binary.}
\end{align*}
\]

Its LP relaxation is obtained by dropping the integrality requirement: we call this the primal LP. Then it has an associated “dual” LP which is as follows:

\[
\begin{align*}
\max & \quad b^T y \\
\text{s.t.} & \quad y^T A \leq c^T, \\
& \quad y \geq 0.
\end{align*}
\]

Each constraint in primal gives rise to a variable in dual (\(y\) in this case), and each variable in primal gives rise to a constraint in dual.

The primal-dual method is an approach that incrementally constructs an integer primal and fractional dual solution. The method ensures that the cost of the primal solution is at most some factor \(\alpha\) times the dual solution, which implies an approximation ratio of \(\alpha\).

Start with an empty primal solution \(F = \emptyset\) and dual solution \(y = 0\).

while \(F\) is not feasible do
  Increase \(y\) in a suitable way until the dual constraint gets tight for some \(i\).
  Add element \(i\) to the primal solution, i.e. \(F \leftarrow F \cup i\).

Often we also need to modify the final solution \(F\) (in some simple way) to prove a good approximation ratio. One particularly useful aspect of this approach is that it does not rely on actually solving an LP: so primal-dual algorithms are often faster than LP-rounding algorithms.

2 \(k\)-Sparse Set Cover

The \(k\)-sparse set cover problem is defined as follows. We are given a set of elements \(E\) and \(n\) subsets \(S_i \subseteq E\). There is cost \(c_i \geq 0\) for each subset \(S_i\). Let \(k\) denote the maximum number of sets...
containing any element, i.e. $k = \max_{e \in E} |i : e \in S_i|$. The goal is to select some subsets to cover the set $E$ with a minimum cost.

**Remark 2.1** In vertex-cover problem, $k = 2$.

**Remark 2.2** In homework 3, a $k$-approximation algorithm is developed via LP rounding. Here, we will provide another $k$-approximation algorithm via Primal-Dual approach without solving the corresponding LP directly.

Recall that we can formulate the LP relaxation of the $k$-sparse set cover problem as follows, where $x_i$ denotes whether set $S_i \subseteq E$ is selected or not.

$$
(P) \quad \min \sum_{i=1}^{n} c_i x_i \\
\text{s.t.} \quad \sum_{i : e \in S_i} x_i \geq 1, \quad \forall e \in E, \\
\quad x \geq 0.
$$

By assigning variable $y_e$ to each of the constraint, the dual problem of the above (P) is

$$
(D) \quad \max \sum_{e \in E} y_e \\
\text{s.t.} \quad \sum_{e \in S_i} y_e \leq c_i, \quad \forall i \in [n], \\
\quad y \geq 0.
$$

Let $P^*$ denote the optimal primal objective value, $OPT$ is the optimal set cover cost. The following observation is a restatement of weak duality.

**Observation 2.1** Any dual feasible solution has an objective value upper bounded by the optimal primal objective, i.e., $D \leq P^* \leq OPT$.

Now we formally state the primal-dual algorithm for $k$-sparse set cover problem.

**Algorithm 1** Primal-Dual algorithm for $k$-sparse set cover problem

1: Start with primal feasible solution $F = \emptyset$ and dual feasible solution $y = 0$
2: **while** $F$ is not feasible **do**
3: \hspace{1em} Pick an element $e$ that is not covered by $F$
4: \hspace{1em} Increase the dual variable $y_e$ as much as possible while maintaining dual feasibility. This process stops when $\sum_{e \in S_i} y_e = c_i$ for some set $S_i$.
5: \hspace{1em} Add set $S_i$ to $F$, i.e., $F \leftarrow F \cup \{i\}$

The main technical result is the following theorem.

**Theorem 2.1** The above algorithm is a $k$-approximation algorithm for $k$-sparse set cover problem

To analyze the algorithm, we first show the following critical lemma.

**Lemma 2.1** $\text{Cost}(F) \leq k \sum_{e \in E} y_e$, where $\text{Cost}(F)$ denotes the primal cost computed by the algorithm and $y_e$ is the dual solution found by the algorithm.
Proof: Since for each $S_i$ picked by the algorithm, we have $c_i = \sum_{e \in S_i} y_e$ holds until the end of the algorithm, we have

$$\text{Cost}(F) = \sum_{i \in F} c_i = \sum_{i \in F} \sum_{e \in S_i} y_e = \sum_{e \in E} y_e | i \in F : e \in S_i | \leq k \sum_{e \in S_i} y_e,$$

where the third equality holds by interchanging the order of summation. The last inequality is by definition of the $k$-sparse set cover instance.

Since the dual solution computed by the algorithm is always feasible, we have

$$\text{Cost}(F) \leq k \sum_{e \in E} y_e \leq kP^* \leq k \cdot \text{OPT}$$

by applying Observation 2.1 and Lemma 2.1.

3 Steiner Forest Problem

Given a graph $G = (V, E)$ along with cost $c_e \geq 0$ on each edge $e \in E$. Let $\{s_i, t_i\}_{i=1}^k$ be pairs of terminals which are given as input. The Steiner forest problem is to select some edges from graph $G$ such that $s_i$ is connected to $t_i$ for each $i \in [k]$ while minimizing the total cost.

The main theorem is stated as follows.

**Theorem 3.1** There is a 2-approximation algorithm for Steiner forest problem.

**Remark 3.1** Agrawal, Klein, and Ravi (1995) first give a 2-approximation algorithm for the Steiner forest problem. Goemans and Williamson (1995) proposed a general primal-dual framework which can be applied to various problems including Steiner forest problem.

**Definition 3.1** We say a vertex set $S \in 2^V$ is active if and only if $|S \cap \{s_i, t_i\}| = 1$ for some $i \in [k]$. Let $A \subseteq 2^V$ be the collection of all active vertex sets.

We first provide an LP relaxation to the Steiner forest problem. Let $x_e$ denote whether an edge $e$ is selected. Consider the following LP, where $\delta S = \{(u, v) \in E : u \in S, v \notin S\}$ denotes the edges at the boundary of $S$.

\[
\begin{align*}
(P) \quad \min & \quad \sum_{e \in E} c_e x_e \\
\text{s.t.} & \quad \sum_{e \in \delta S} x_e \geq 1, \quad \forall S \in A, \\
& \quad x \geq 0.
\end{align*}
\]
By assigning variable $y_S$ to each of the constraint, the dual problem of the above (P) is

$$(D) \quad \text{max} \quad \sum_{S \in A} y_S$$

$$\text{s.t.} \quad \sum_{S \in A, e \in \delta S} y_S \leq c_e, \quad \forall e \in E,$$

$$y \geq 0.$$ 

Now we formally state the primal-dual algorithm for Steiner forest problem.

**Algorithm 2** Primal-Dual algorithm for Steiner forest problem

1: Start with primal feasible solution $F = \emptyset \subseteq E$ and dual feasible solution $y = 0$
2: while $F$ is not feasible do
3: Let $B$ denotes the connected components in $F$ and $C \subseteq B$ consists of active components.
4: Increase the dual solution $y_S$ for each $S \in C$ until $c_e = \sum_{S \in A, e \in \delta S} y_S$ for some edge $e \in E$.
5: Add edge $e$ to $F$, i.e., $F \leftarrow F \cup \{e\}$
6: Output union of $s_i$-$t_i$ paths in $F$ (denoted by $R$)

**Observation 3.1** The solution $R$ is a feasible Steiner forest and $y$ is a feasible dual solution.

To analyze the algorithm, we need to show the following critical lemma.

**Lemma 3.1** $\text{cost}(R) = \sum_{e \in R} c_e \leq 2 \sum_{S \in A} y_S$.

**Proof:** Since for each $e \in R$, the algorithm guarantees $c_e = \sum_{S \in A, e \in \delta S} y_S$, therefore,

$$\text{cost}(R) = \sum_{e \in R} c_e = \sum_{e \in R} \sum_{S \in A, e \in \delta S} y_S = \sum_{S \in A} y_S \cdot |R \cap \delta S| \triangleq P(y)$$

Let $D(y) \triangleq \sum_{S \in A} y_S$, It suffices to show that $\frac{dP}{dt} \leq 2 \frac{dD}{dt}$, where time $t$ is ticking whenever dual solution $y$ increases. Note that $\frac{dy_S}{dt} = 1$ when $S$ is active and $\frac{dy_S}{dt} = 0$ otherwise. We conclude that $\frac{dD}{dt} = \sum_{S \in A} \frac{dy_S}{dt} = |C|$ and $\frac{dP}{dt} = \sum_{S \in A} \frac{dy_S}{dt} |\delta S \cap R| = \sum_{S \in C} |\delta S \cap R|$. Define an auxiliary graph $H$ that treats each connected component in $B$ as a single node and includes all edges in $R$ that are added after time $t$. Note that for any $S \in B$, we have $\text{deg}_H(S) = |\delta S \cap R|$. Also, $H$ is a forest because $F$ is a forest where each $S \in B$ is connected internally.

We now have several claims:

**Claim 3.1** The nodes of zero degree in $H$ must be inactive connected components.

Otherwise, the solution $R$ must be infeasible since it contains active connected component without an edge going out. Let $B' = B \setminus \{S \in B : \text{deg}_H(S) = 0\}$ and $I = B' \setminus C$. Note that $H$ is a forest on the nodes $B'$ and all these nodes have degree at least one.

**Claim 3.2** Every leaf node in $H$ must be active.
Otherwise, suppose an inactive leaf node $S_I \in \cal I$ has only one edge $e$ connected to a connected component $S \in \cal B'$. Then the edge $e$ will not be used in any $s_i$-$t_i$ path of $F$: any such path must cross $S_I$ exactly once which is not possible as $S_I$ is inactive. This contradicts with the fact that $e \in R$.

Since $H$ has no cycles, we must have $\sum_{S \in \cal B'} \deg_H(S) \leq 2|\cal B'|$. Note that Claim 3.2 implies $\sum_{S \in \cal I} \deg_H(S) \geq 2|\cal I|$, we have

$$\frac{dP}{dt} \leq \sum_{S \in \cal C} \deg_H(S) = \sum_{S \in \cal B'} \deg_H(S) - \sum_{S \in \cal I} \deg_H(S) \leq 2|\cal B'| - 2|\cal I| = 2|\cal C| = 2 \frac{dD}{dt}.$$ 

Thus, the lemma is proved. \qed

Finally, according to Lemma 3.1 and the feasibility of both the primal and dual solution, we have $\text{cost}(R) \leq 2 \sum_{S \in \cal A} y_S \leq 2P^* \leq 2OPT$, which completes the proof of Theorem 3.1.

References
