Recitation 2

ILP and Dependency Parsing

Chaitanya Ahuja

Carnegie Mellon University

Table of contents

Introduction

Integer Linear Program (ILP) canonical form

$$\begin{aligned} \max \quad & \mathbf{c}^\intercal \mathbf{x} && (1) \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b}, && (2) \\ & \mathbf{x} \geq 0, && (3) \\ \text{integer cond.} \quad & \mathbf{x} \in \mathbb{Z}^n, && (4) \end{aligned}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{c} \in \mathbb{R}^n$ and $\mathbf{b} \in \mathbb{R}^m$

ILP: Solving for the optimal unknown x

- Solving an ILP is NP-hard unlike a LP

ILP: Solving for the optimal unknown x

- Solving an ILP is NP-hard unlike a LP
- There are 3 kinds of solutions

ILP: Solving for the optimal unknown x

- Solving an ILP is NP-hard unlike a LP
- There are 3 kinds of solutions
    - Exact - Unimodularity conditions allow for LP relaxation

ILP: Solving for the optimal unknown x

- Solving an ILP is NP-hard unlike a LP
- There are 3 kinds of solutions
  - Exact - Unimodularity conditions allow for LP relaxation
  - Exact - Cutting Plane Method, Branch and Bound Method

ILP: Solving for the optimal unknown x

- Solving an ILP is NP-hard unlike a LP
- There are 3 kinds of solutions
  - Exact - Unimodularity conditions allow for LP relaxation
  - Exact - Cutting Plane Method, Branch and Bound Method

ILP: Solving for the optimal unknown x

- Solving an ILP is NP-hard unlike a LP
- There are 3 kinds of solutions
  - Exact - Unimodularity conditions allow for LP relaxation
  - Exact - Cutting Plane Method, Branch and Bound Method
  - Heuristic - Tabu search

ILP for CRFs

Set-up

For a directed graph $G = (E, V)$, $s, t \in V$ and, $(u, v) \in E$ we have:

$$\text{Cost:} \quad c_{uv} \tag{5}$$

$$\text{Indicator Variable:} \quad x_{uv} \tag{6}$$

If (u,v) is in the minimum cost path, then x_{uv} is 1; otherwise 0.

Set-up

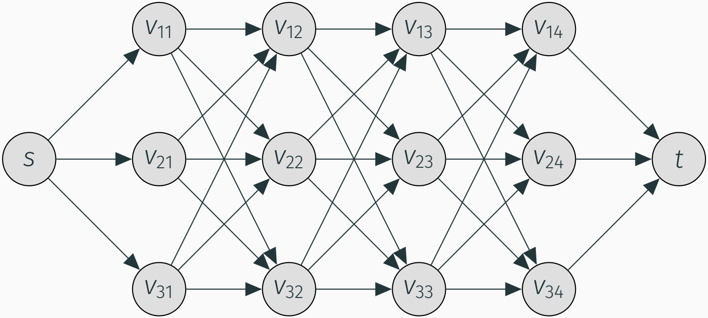

Figure 1: Graph G (Shortest Path in Red)

Set-up

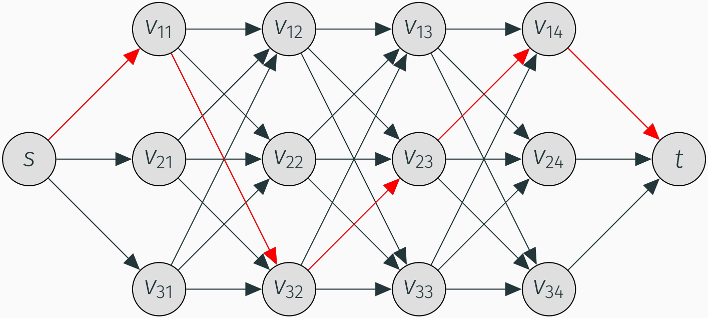

Figure 1: Graph G (Shortest Path in Red)

Set-up

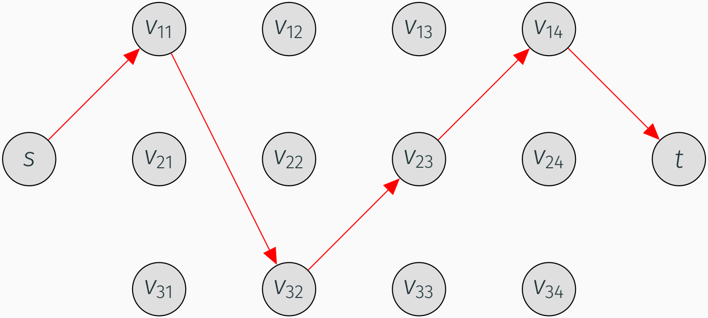

Figure 1: Graph G (Shortest Path in Red)

$x_{v_{11}v_{32}} = 1$;
but $x_{v_{11}v_{12}} = 0$ and $x_{v_{11}v_{22}} = 0$.

More Definitions

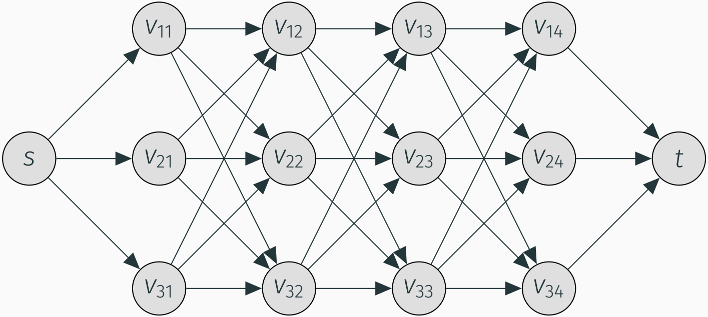

Figure 2: Graph G

(10)

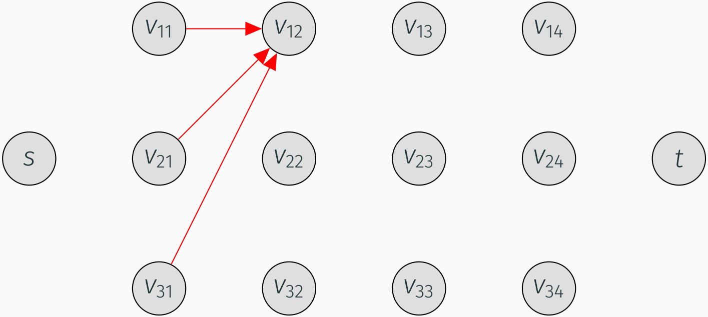

Figure 2: Graph G

$$V^{-}(v) \text{ :nodes connected by inward edges} \tag{7}$$

$$V^{-}(v_{12}) = \{v_{11}, v_{21}, v_{31}\} \tag{8}$$

$$\tag{10}$$

More Definitions

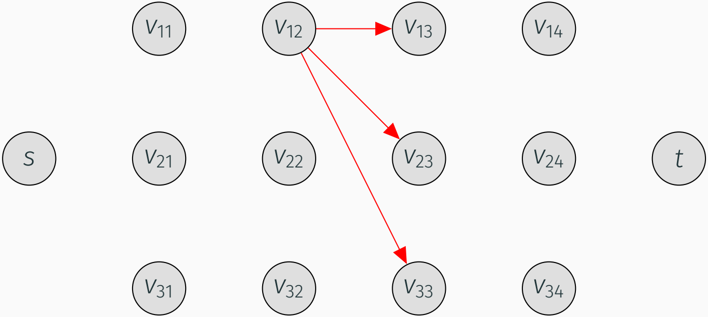

Figure 2: Graph G

$$V^{-}(v) \text{ :nodes connected by inward edges} \tag{7}$$

$$V^{-}(v_{12}) = \{v_{11}, v_{21}, v_{31}\} \tag{8}$$

$$V^{+}(v) \text{ :nodes connected by outward edges} \tag{9}$$

$$V^{+}(v_{12}) = \{v_{13}, v_{23}, v_{33}\} \tag{10}$$

ILP Constraints

$$\min \sum_{(u,v)\in E} c_{uv} x_{uv} \tag{11}$$

(14)

ILP Constraints

$$\min \sum_{(u,v)\in E} c_{uv} x_{uv} \tag{11}$$

$$\text{subject to: } \sum_{u\in V^-(v)} x_{uv} - \sum_{w\in V^+(v)} x_{vw} = 0, \quad \forall v \in V - \{s,t\} \tag{12}$$

(14)

ILP Constraints

$$\min \sum_{(u,v)\in E} c_{uv} x_{uv} \tag{11}$$

$$\text{subject to: } \sum_{u\in V^-(v)} x_{uv} - \sum_{w\in V^+(v)} x_{vw} = 0, \quad \forall v \in V - \{s,t\} \tag{12}$$

$$\text{starting node: } \sum_{u\in V^-(s)} x_{us} - \sum_{w\in V^+(s)} x_{sw} = -1 \tag{13}$$

$$\tag{14}$$

ILP Constraints

$$\min \sum_{(u,v)\in E} c_{uv}x_{uv} \tag{11}$$

$$\text{subject to: } \sum_{u\in V^-(v)} x_{uv} - \sum_{w\in V^+(v)} x_{vw} = 0, \quad \forall v \in V - \{s,t\} \tag{12}$$

$$\text{starting node: } \sum_{u\in V^-(s)} x_{us} - \sum_{w\in V^+(s)} x_{sw} = -1 \tag{13}$$

$$\text{terminating node: } \sum_{u\in V^-(t)} x_{ut} - \sum_{w\in V^+(t)} x_{tw} = 1 \tag{14}$$

We are ready!! Let's write an ILP for CRFs [1]

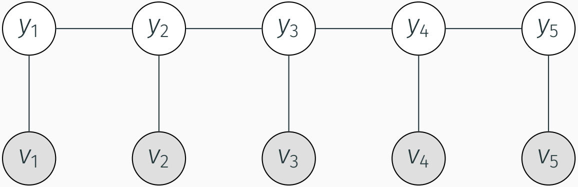

Figure 3: CRF

[1] Roth, Dan, and Wen-tau Yih. "Integer linear programming inference for conditional random fields." Proceedings of the 22nd international conference on Machine learning. ACM, 2005.

We are ready!! Let's write an ILP for CRFs [1]

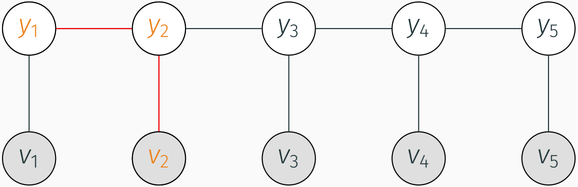

Figure 3: CRF

[1] Roth, Dan, and Wen-tau Yih. "Integer linear programming inference for conditional random fields." Proceedings of the 22nd international conference on Machine learning. ACM, 2005.

We are ready!! Let's write an ILP for CRFs [1]

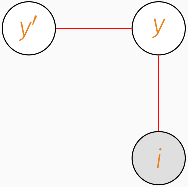

Figure 3: CRF

where, $y', y \in \{0, 1, \ldots, m-1\}$ and $x \in \{0, 1, \ldots n-1\}$

[1] Roth, Dan, and Wen-tau Yih. "Integer linear programming inference for conditional random fields." Proceedings of the 22nd international conference on Machine learning. ACM, 2005.

We are ready!! Let's write an ILP for CRFs [1]

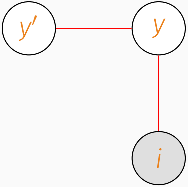

Figure 3: CRF

$$\text{where, } y', y \in \{0, 1, \ldots, m-1\} \text{ and } x \in \{0, 1, \ldots n-1\}$$

$$x_{i,y'y} = 1$$

[1] Roth, Dan, and Wen-tau Yih. "Integer linear programming inference for conditional random fields." Proceedings of the 22nd international conference on Machine learning. ACM, 2005.

Example 1

Force the label of token i to be 0.

Example 1

Force the label of token i to be 0.

$$\sum_{0 \leq y \leq m-1} x_{i,y0} = 1,$$

Example 2

If label a appears, then label b must also appear.

Example 2

If label a appears, then label b must also appear.

$$\sum_{0 \le y \le m-1} x_{i,ya} \le \sum_{\substack{0 \le y \le m-1 \\ 0 \le i \le n-1}} x_{i,yb}$$

$$\forall i \in \{0, 1, \ldots, n-1\}$$

Example 3

Force every sequence to have atleast one segment of interest.

Example 3

Force every sequence to have atleast one segment of interest.

$$\sum_{\substack{0 \leq i \leq n-1 \\ 0 \leq y \leq m-1}} x_{i,y0} \leq n - 1,$$

if 0 is the label O

Example 4

If a token $a \in \mathcal{A}$ is assigned label l, then all the tokens in $\mathcal{A}$ have to be l. Assuming $\mathcal{A}$ ranges from tokens p to q

Example 4

If a token $a \in \mathcal{A}$ is assigned label l, then all the tokens in $\mathcal{A}$ have to be l. Assuming $\mathcal{A}$ ranges from tokens p to q

$$v_{i,y} = \sum_{0 \leq y' \leq m-1} x_{i,y'y}$$

Example 4

If a token $a \in \mathcal{A}$ is assigned label l, then all the tokens in $\mathcal{A}$ have to be l. Assuming $\mathcal{A}$ ranges from tokens p to q

$$v_{i,y} = \sum_{0 \leq y' \leq m-1} x_{i,y'y}$$

Is $v_{i,y}$ a binary variable?

Example 4

If a token $a \in \mathcal{A}$ is assigned label l, then all the tokens in $\mathcal{A}$ have to be l. Assuming $\mathcal{A}$ ranges from tokens p to q

$$v_{i,y} = \sum_{0 \leq y' \leq m-1} x_{i,y'y}$$

Is $v_{i,y}$ a binary variable?

Yes, and indicates whether token i is assigned label y.

Example 4

If a token $a \in \mathcal{A}$ is assigned label l, then all the tokens in $\mathcal{A}$ have to be l. Assuming $\mathcal{A}$ ranges from tokens p to q

$$v_{i,y} = \sum_{0 \leq y' \leq m-1} x_{i,y'y}$$

Is $v_{i,y}$ a binary variable?

Yes, and indicates whether token i is assigned label y.

$$(q-p)v_{p,l} = \sum_{p=1 \leq i \leq q} v_{i,l}$$

Dependency Parsing

Basics

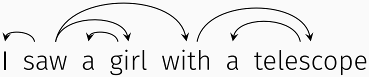

Figure 4: Dependency Parse

Ambiguous Parses

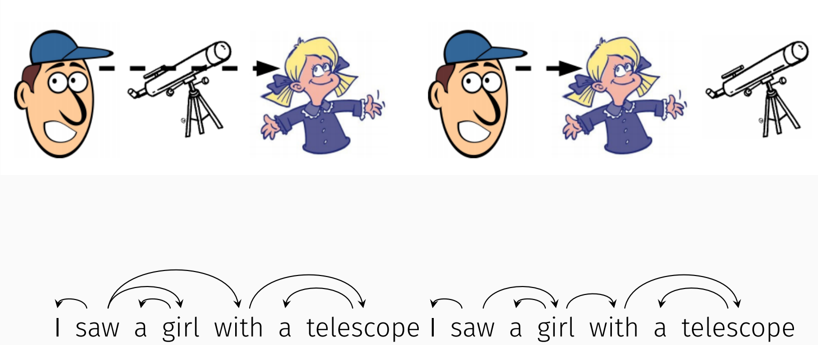

Figure 5: Multiple Dependency Parses[2]

[2] Example taken from Neubig's slides

Dependency Parse as an ILP

A dependency parse

- is a directed acyclic graph.

Dependency Parse as an ILP

A dependency parse

- is a directed acyclic graph.
- has each node with exactly one parent (except the root node).

Dependency Parse as an ILP

A dependency parse

- is a directed acyclic graph.
- has each node with exactly one parent (except the root node).
- Optimising a parser is equivalent to finding the maximum spanning tree.

Questions ?