Conditional Random Fields with Latent Variables

Mark Schmidt, June 2014
(e-mail me for references)
Outline

• Overview of General Conditional Random Fields
• Conditional Random Fields with Latent Variables
Binary Logistic Regression

Classify using \( y = \text{sign}(w^T x) \).

\[
\begin{bmatrix}
-1.7491 & 1 \\
1.1326 & 0 \\
-0.7938 & 1 \\
0.3149 & 1 \\
-0.7836 & 0 \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 & 0 \\
0 & 1 \\
0 & 1 \\
1 & 0 \\
1 & 1 \\
\end{bmatrix}
\]
**Binary Logistic Regression**

$y = [-1]$

Classify using $y = \text{sign}(w^T x)$.

$w = \begin{bmatrix} -1.7491 \\ 1.1326 \\ 0 \\ -0.7938 \\ 0.3149 \end{bmatrix}$

$x = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$
Binary Logistic Regression

\[ y = [-1] \]

Classify using \[ y = \text{sign}(w^T x) \].

\[ w = \begin{bmatrix} -1.7491 \\ 1.1326 \\ 0 \\ -0.7938 \\ 0.3149 \end{bmatrix} \]

\[ x = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \]

\( x_i \) can be continuous
Binary Logistic Regression

Classify using $y = \text{sign}(w^T x)$.

$$p(y = 1|x, w) = \frac{\exp(yw^T x)}{\exp(w^T x) + \exp(-w^T x)}$$

$$= \frac{\exp(yw^T x)}{\sum_{y'} \exp(yw^T x)}$$

$x = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$

$x_i$ can be continuous

$w = \begin{bmatrix} -1.7491 \\ 1.1326 \\ 0 \\ -0.7938 \\ 0.3149 \end{bmatrix}$

$y = [-1]$
Binary Logistic Regression

Classify using $y = \text{sign}(w^T x)$.

$$p(y = 1|x, w) = \frac{\exp(yw^T x)}{\exp(w^T x) + \exp(-w^T x)} = \frac{\exp(yw^T x)}{\sum_{y'} \exp(yw^T x)}$$

$$p(y = s|x, w) \propto \exp(sw^T x)$$

**Notes on Perceptron**

Now 2 Multi-Class

Classify using $y = \text{max}(w^T x)$.  

$$y = [-1] = \begin{cases} 1 & y \geq 0 \\ -1 & y < 0 \end{cases}$$

$w$ is a $k \times 1$ weight vector.

| 1.7491 |
| 1.1326 |
| 0 |
| -0.7938 |
| 0.3149 |

$x$ is a $k \times 1$ feature vector.

$$x = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$y$ is a $1 \times 1$ label vector.

$$y = \begin{bmatrix} -1 \end{bmatrix}$$

$w$ is a $1 \times 1$ weight vector.

$w = \begin{bmatrix} -1.7491 \\ 1.1326 \\ 0 \\ -0.7938 \\ 0.3149 \end{bmatrix}$

$(x_i$ can be continuous)
Multi-Class Logistic Regression

Now $y \in \{1, 2, 3, \ldots, S\}$.
Multi-Class Logistic Regression

\[ y = [3] \]

Now \( y \in \{1, 2, 3, \ldots, S\} \).

\[ x = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix} \]

\[ w = \begin{bmatrix} -1.7491 & 1.7411 & 0.8106 \\ 1.1326 & 0.4868 & 0.6985 \\ 0 & 1.0488 & -0.4016 \\ -0.7938 & 1.4886 & 1.2688 \\ 0.3149 & 1.2705 & -0.7836 \end{bmatrix} \]
Multi-Class Logistic Regression

Now \( y \in \{1, 2, 3, \ldots, S\} \).

Classify by maximizing \( w_s^T x_s \) over \( s \)

\[
x = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}
\]

\[
w = \begin{bmatrix} -1.7491 & 1.7411 & 0.8106 \\ 1.1326 & 0.4868 & 0.6985 \\ 0 & 1.0488 & -0.4016 \\ -0.7938 & 1.4886 & 1.2688 \\ 0.3149 & 1.2705 & -0.7836 \end{bmatrix}
\]
Multi-Class Logistic Regression

\[ y = [3] \] 

Now \( y \in \{1, 2, 3, \ldots, S\} \).

Classify by maximizing \( w_s^T x_s \) over \( s \)

\[ x = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix} \]

\[ w = \begin{bmatrix} -1.7491 & 1.7411 \\ 1.1326 & 0.4868 \\ 0 & 1.0488 \\ -0.7938 & 1.4886 \\ 0.3149 & 1.2705 \end{bmatrix} \]

This is \( w_3 \)
Multi-Class Logistic Regression

$y = [3]$ \hspace{1cm} \text{Now } y \in \{1, 2, 3, \ldots, S\}.$

Classify by maximizing $w_s^T x$ over $s$

$w = \begin{bmatrix}
-1.7491 & 1.7411 \\
1.1326 & 0.4868 \\
0 & 1.0488 \\
-0.7938 & 1.4886 \\
0.3149 & 1.2705
\end{bmatrix}$

Usually we use the same features across classes.

This is $w_3$
Multi-Class Logistic Regression

\[ y = [3] \]

Now \( y \in \{1, 2, 3, \ldots, S\} \).

Classify by maximizing \( w_s^T x \) over \( s \)

\[
w = \begin{bmatrix}
-1.7491 & 1.7411 \\
1.1326 & 0.4868 \\
0 & 1.0488 \\
-0.7938 & 1.4886 \\
0.3149 & 1.2705
\end{bmatrix}
\]

Usually we use the same features across classes.

\[
p(y = s | x, w) \propto \exp(w_s^T x)
\]

This is \( w_3 \)
Multi-Class Logistic Regression

Now $y = [3]$  

Now $y \in \{1, 2, 3, \ldots, S\}$.  

Classify by maximizing $w_s^T x$ over $s$

Usually we use the same features across classes.

$$x = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

$$y = [3]$$

$$w = \begin{bmatrix} -1.7491 & 1.7411 & 0.8106 \\ 1.1326 & 0.4868 & 0.6985 \\ 0 & 1.0488 & -0.4016 \\ -0.7938 & 1.4886 & 1.2688 \\ 0.3149 & 1.2705 & -0.7836 \end{bmatrix}$$

$p(y = s | x, w) \propto \exp(w_s^T x)$

For ordered classes, use ordinal logistic regression.

This is $w_3$
Multi-Label Logistic Regression

We now have multiple labels $y_n$

This is $w_3 \cdot p(y = s | x, w)$

Usually we use the same features across classes.

For ordered classes, use ordinal logistic regression.

3 Multi-Task

We now have multiple labels $y_n$
Multi-Label Logistic Regression

We now have multiple labels $y_n$

$$p(y = s|x, w) \propto \prod_{n=1}^{N} \exp(w_{n,s}^T x_n)$$

Challenges:
share information across the $w_n$ model in correlations in the $y_n$
This is $w_3 \p(y = s | x, w_n)$

Usually we use the same features across classes.

For ordered classes, use ordinal logistic regression.

### Multi-Task

We now have multiple labels $y_n$.

Challenges:

- share information across the model in correlations in the $y_n$

### Conditional Random Fields

CRFs model correlation in the $y_n$.

$$p(y = s | x, w) \propto \exp(w^T s x)$$
Conditional Random Fields

CRFs model correlation in the $y_n$

Can have special potentials on start/end
CRFs model correlation in the $y_n$

Can have special potentials on start/end
We often tie parameters (but can have node/edge types)
Conditional Random Fields

CRFs model correlation in the $y_n$

Can have special potentials on start/end

We often tie parameters (but can have node/edge types)

Could also share information through regularization
Conditional Random Fields

CRFs model correlation in the $y_n$

Can have special potentials on start/end

Edges can depend on the features

We often tie parameters (but can have node/edge types)

Could also share information through regularization
Conditional Random Fields

CRFs model correlations in the $y_n$

Can have special potentials on start/end

Edges can depend on the features

We often tie parameters (but can have node/edge types)

We can have global features

Could also share information through regularization
Conditional Random Fields

CRFs model correlation in the $y_n$

$p(y = s | x, w) \propto \prod_{n=1}^{N} \exp(w^T_{s_n} x_n) \prod_{n=0}^{N} \exp(v_{s_n, s_{n+1}})$
General Conditional Random Fields

We can have any graph structure on the $y_n$

$$p(y = s|x, w) \propto \prod_{i \in N} \exp(w_{si}^T x_i) \prod_{i,j \in E} \exp(v_{si,sj})$$
General Conditional Random Fields

We can have any graph structure on the $y_n$

$p(y = s|x, w) \propto \prod_{i \in N} \exp(w_{s_i}^T x_i) \prod_{i,j \in E} \exp(v_{s_i, s_j})$
General Conditional Random Fields

Tasks involving states $s$:

- Decoding: $\arg\max_s p(y = s|x, w)$
- Inference: $\sum_s p(y = s|x, w)$ and $\sum_{s|s_i=c} p(y = s|x, w)$
- Sampling: generate $s \sim p(y = s|x, w)$

For chain structured data:

- Decode using Viterbi
- Inference using Forward-Backward
- Sampling using Forward-Filter, Backward-Sample
General Conditional Random Fields

Exact methods:

- Cutset conditioning
- Super nodes
- Junction tree
- Graph cuts (for decoding of binary associative)
General Conditional Random Fields

Exact methods:

- Cutset conditioning
- Super nodes
- Junction tree
- Graph cuts (for decoding of binary associative)

Exponential in tree-width
General Conditional Random Fields

Exact methods:

- Cutset conditioning
- Super nodes
- Junction tree
- Graph cuts (for decoding of binary associative)

Approximate methods:

- Decode using local search
- Inference using variational
- Sample using MCMC
General Conditional Random Fields

Exact methods:
- Cutset conditioning
- Super nodes
- Junction tree
- Graph cuts (for decoding of binary associative)

Approximate methods:
- Decode using local search
- Inference using variational
- Sample using MCMC

Use one task to perform the other:
- Inference with sampling: counting
- Inference with decoding: Viterbi approximation
- Decoding with inference: max-product
- Decoding with sampling: simulated annealing
- Sampling with inference: variational MCMC
- Sampling with decoding: herding
General Conditional Random Fields

Estimation methods to find $w$:

- Inference: maximum likelihood and regularized maximum likelihood
- Decoding: perceptron and max-margin Markov networks
- Sampling: contrastive divergence and stochastic maximum likelihood
- None: pseudo-likelihood and composite likelihoods
Higher-Order CRF

Add 2nd- or higher-order dependencies
Higher-Order CRF

Add 2nd- or higher-order dependencies
Higher-Order CRF

Add 2nd- or higher-order dependencies

Inference with super nodes:
- 1st-order: $O(\mathcal{N}\mathcal{S}^2)$
- 2nd-order: $O(\frac{\mathcal{N}}{2}\mathcal{S}^4)$
- ith-order: $O(\frac{\mathcal{N}}{i}\mathcal{S}^{2i})$
Dynamic CRFs

Track multiple variables with repeated structure
Dynamic CRFs

Track multiple variables with repeated structure
Dynamic CRFs

Track multiple variables with repeated structure

6 Higher-Order CRFs

Inference with super nodes:

- 1st-order: $O(N S^2)$
- 2nd-order: $O(N^2 S^4)$
- ith-order: $O(N^i S^{2i})$
Dynamic CRFs

Track multiple variables with repeated structure

Inference with super-nodes
Semi-Markov CRF

Add dependency on length of segment

6 Higher-Order CRF
Add 2nd- or higher-order dependencies

Inference with super nodes:

• 1st-order: $O(\text{NS}^2)$
• 2nd-order: $O(\text{N}^2\text{S}^4)$
• $i$th-order: $O(\text{N}^i\text{S}^{2i})$

7 Dynamic CRFs
Track multiple variables with repeated structure

Inference with super-nodes

8 Semi-Markov CRFs
Add dependency on length of segment

Small number of other tricks are possible: sentence must have one verb
Semi-Markov CRF

Add dependency on length of segment

\[ y_1, s, e \]
\[ y_2, s, e \]
\[ y_3, s, e \]
\[ y_4, s, e \]
\[ y_5, s, e \]
\[ y_6, s, e \]

\[ x_1 \]
\[ x_2 \]
\[ x_3 \]
\[ x_4 \]
\[ x_5 \]
\[ x_6 \]
Semi-Markov CRF

Add dependency on length of segment

Can also have small number of global dependencies:
‘at least one verb’
Skip-Chain CRF

Encourage repeated words to receive the same label
Skip-Chain CRF

Encourage repeated words to receive the same label
Outline

• Overview of General Conditional Random Fields

• Conditional Random Fields with Latent Variables
**Missing Labels**

\[
p(y_{1:2} = s_{1:2}, h = s_3, y_{4:N} = s_{4:N} | x, w) \propto \prod_{n=1}^{N} \exp(w^T s_n x_n) \prod_{n=0}^{N} \exp(v_{s_n, s_{n+1}}) = \frac{f(y, h)}{\sum_{y', h} f(y', h)}
\]
Missing Labels

\[ p(y_{1:2} = s_{1:2}, h = s_3, y_{4:N} = s_{4:N} | x, w) \propto \prod_{n=1}^{N} \exp(w_{s_n}^T x_n) \prod_{n=0}^{N} \exp(v_{s_n, s_{n+1}}) = \frac{f(y, h)}{\sum_{y', h} f(y', h)} \]

\[ p(y | x, w) = \sum_{h} p(y, h | x, w) = \frac{\sum_{h} f(y, h')}{\sum_{y', h'} f(y', h')} \]
\[ p(y_{1:2} = s_{1:2}, h = s_3, y_{4:N} = s_{4:N} | x, w) \propto \prod_{n=1}^{N} \exp(w_{sn}^T x_n) \prod_{n=0}^{N} \exp(v_{sn, sn+1}) = \frac{f(y, h)}{\sum_{y', h} f(y', h)} \]

\[ p(y | x, w) = \sum_{h} p(y, h | x, w) = \frac{\sum_{h} f(y, h')}{\sum_{y', h'} f(y', h')} \]

If all variables hidden, cancels out

Numerator leads to non-convex optimization
Latent Logistic Regression

Latent logistic: class variables have unknown sub-classes
Latent logistic: class variables have unknown sub-classes

### Latent Logistic Regression

- **y**: Has unknown sub-classes
- **h**: $h \in \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$
  - Class 1
  - Class 2
  - Class 3

Inference with super-nodes:
- If all variables hidden, cancels out.
- Numerator leads to non-convex optimization.

Mathematical Formulation:
\[
p(y|h, x, w) = \exp \left( \sum_{n=1}^{N} \left( s_n \cdot h_n \cdot f(y_n|h_n, x, w) \right) \right)
\]
Latent Logistic Regression

Latent logistic: class variables have unknown sub-classes

\[ p(y = s|x, w) \propto \sum_{h \in s} \exp(w_h^T x) \]
Hidden CRF
Hidden CRF

An HMM with a supervised label

\[
p(y = s | x, w) \propto \exp \left( \sum_{i=1}^{v} v_{i,s} g_i \left( w^T_{i} x \right) \right)
\]
Latent Dynamic CRF
Latent Dynamic CRF

Hidden dependency structure among sub-classes
Latent Logistic Regression and Neural Networks

Latent logistic: class variables have unknown sub-classes

\[ p(y = s | x, w) \propto \sum_{h \in s} \exp(w_h^T x) \]
Latent Logistic Regression and Neural Networks

Latent logistic: class variables have unknown sub-classes

\[ p(y = s|x, w) \propto \sum_{h \in s} \exp(w_h^T x) \]

\( h \in \{1, 2, 3, 4, 5, 6, 7, 8, 9\} \)

Class 1 Class 2 Class 3

Neural network: combine non-linear transformations to binary variables

\( g_i \in [0, 1] \)
Latent Logistic Regression and Neural Networks

Latent logistic: class variables have unknown sub-classes

Neural network: combine non-linear transformations to binary variables

\[
p(y = s | x, w) \propto \sum_{h \in s} \exp(w_h^T x)
\]

\[
p(y = s | x, w) \propto \exp \left( \sum_{i=1}^{H} v_{i,s} g_i(w_i^T x) \right)
\]
Hidden-Unit CRF,
Conditional Neural Field (CNF)
Hidden-Unit CRF, Conditional Neural Field (CNF)

A standard CRF where we learn the features
Related to earlier support vector random fields
Latent Dynamic CNF

\[
\begin{align*}
\gamma_1 & \rightarrow h_1 \rightarrow x_1 \\
\gamma_2 & \rightarrow h_2 \rightarrow x_2 \\
\gamma_3 & \rightarrow h_3 \rightarrow x_3 \\
\gamma_4 & \rightarrow h_4 \rightarrow x_4 \\
\gamma_5 & \rightarrow h_5 \rightarrow x_5 \\
\gamma_6 & \rightarrow h_6 \rightarrow x_6
\end{align*}
\]