I) Latent variables

- Motivation: semantic segmentation as finding boundary of different objects
- Segmentation is expensive → "latent variable"
- Only have class labels → y

Before, we had: \( s(x, y; w) = \langle w, p(y|x) \rangle \)

Now, consider: \( s(x, y; z; w) = \langle w, p(x, y; z) \rangle \)

as before, could predict with argmax \( z \in \mathbb{Z} \), \( y \in \mathbb{Y} \)

* CRF \( p(y|x) \) → hidden CRF

Similar to latent variable modeling with graphical model

ML → Expectation-Maximization

Analog for SVMstruct

Latent SVMstruct
\[ l(y_i, (\gamma, \xi)) \]

Generalization of structural hinge loss:

\[ f(x_i, y, \omega) \leq \max_{(\gamma, \xi)} \omega^T \langle (x_i, y, \xi) \rangle + l(y, (\gamma, \xi)) \leq \frac{\max_{(\gamma, \xi)} \omega^T \langle (x_i, y, \xi) \rangle}{\gamma^2} \geq l(y, h_\omega(x_i)) \]

Recall: sup of convex function is always convex.

If \( f \) is jointly convex function, \( f \) is convex i.e. \( \forall f(g(w)) \) where \( g \) is jointly convex in \( w \).

Here \( f(x, y, w) = u(w) - v(w) \)

where \( u \) \& \( v \) are convex functions of \( w \). "Disjoint of convex functions!"

\( \Rightarrow \) CCP procedure is used to minimize this.

CCCP procedure:

- linearize \( v(w) \) at \( w \) to get upper bound.
- minimize upper bound.
- repeat.

\[ f_c(w) = u(w) - [v(w_0) + \nabla v(w_0)^T (w - w_0)] \]

(or subgradient)

\[ w_{t+1} = \arg\min_w f_c(w) \]

Properties of this procedure:

- Like EM, descent procedure i.e. \( f(w_{t+1}) \leq f(w_t) \)

\[ f(w_t) = f_c(w_t) \leq f_c(w_{t+1}) \Rightarrow f(w_{t+1}) \leq f(w_t) \]

- Local minimum covariance dominated ascent.
\[ V(w) = \max_{\omega} \langle \omega, \phi(x, y) \rangle \]

\[ \Theta V(w_0) = \ell(x, y, \hat{z}(x, w_0)) \]

\[ \Rightarrow \beta(x, y, w) = \max_{\omega} \langle \omega, \phi(x, y) \rangle + \ell(y, \hat{z}(x, w_0)) - \min_{\omega} \langle \omega, \phi(x, y) \rangle + \text{const.} \]

\[ \text{like SVMstruct objective} \]

\[ \text{CCCP for latent SVM struct} \]

\[ \text{repeat:} \]

\[ \text{fall in } \mathcal{Z}^{(i)} \text{ for all ground truth } y^{(i)} \text{ using } w^{(i)} \]

\[ \text{dual standard SVMstruct b get } w^{(i)} \]

\[ \text{repeat} \]

**Kernels**

\[ \text{so far } \phi(x, y) = \langle \omega, \phi(x, y) \rangle \]

\[ \text{recall for with } \omega(x) = \frac{1}{n} \sum_{i=1}^{n} \alpha_i(y) \phi_i(y) \]

\[ \langle \omega, \phi(x, y) \rangle = \frac{1}{n} \sum_{i=1}^{n} \alpha_i(y) \phi_i(y) \]

\[ \langle \phi_i(y), \phi_j(y) \rangle = k((x^{(i)}, y^{(i)}), (x^{(j)}, y^{(j)})) - k(x^{(i)}, y^{(i)}; x, y) \]

\[ k((x, y); (x, y)) = k_x(x, x); k_y(y, y) \]
\[<p(x,y), \phi(x\prime,y\prime)> = \sum_{p/p} (<\phi(x_p,y_p), \phi(x\prime_p,y\prime_p)> + <\phi(y_p,y\prime_p), \phi(y\prime_p,y\prime_p)>) \]

For OCR, we had used \( \phi(x_p,y_p) = \left( \text{vec}(x_p) \cdot \begin{pmatrix} 0 & 1 \end{pmatrix} \right) - y_p \)

\[\Rightarrow K_1((x_p,y_p),(x\prime_p,y\prime_p)) = \frac{\|y_p - y\prime_p\|^2}{y_p^-} <x_p, x\prime_p> \]

But instead, could use \( \frac{\|y_p - y\prime_p\|^2}{y_p^-} \exp\left(-\frac{(x_p - x\prime_p)^2}{2\sigma^2}\right) \)

* Kernel SVM with RBF kernel is similar to nearest neighbor classifier

**Computation:**

BCFW: stored \( w \rightarrow O(d) \) for each \( i \)

\[<w_i, \phi(x_i,y_i)> \rightarrow O(d) \]

with kernel \( k \) cannot store \( w \), so maintain \( \phi(x_i,y_i) \) instead...

\[w = \sum_{i=1}^{n} \beta_i \phi(x_i) \]

\( \text{sparse} \) when using FW

often 6 iterations of BCFW, only 6 non-zero variables

\[<w, \phi(x,y)> \rightarrow O(t) \]

Kernel methods \( \rightarrow O(n^2) \),
Deep Learning

1. Plugin "deep learning" features in a structured prediction model

Example AK:

\[ u_1 \rightarrow u_2 \rightarrow u_3 \]

images:

\begin{array}{c}
\square \\
\square \\
\square \\
\end{array}

Example:

[Vu et al., ICCV 2015] "Context-aware CNNs for person head detection"

II. Recurrent Neural Network (RNN)

\[ p(y|x) = \prod_{t \in T} p(y_t | y_{t+1}, x) \]

\[ p(y_t | y_{t+1}, x) \]

\[ h_{t+1} = f(h_t, x, y_t, W) \]

\[ p(y_t | y_{1:t-1}, x) \text{ or } \exp(C(y_t) h_t) \]

\[ h_1 \quad h_2 \quad h_t \]
Pointers

- latent variable SVMstruct:

  others:

- CCCP procedure convergence rate:
  - Ian E.H. Yen, Nanyun Peng, Po-Wei Wang and Shou-de Lin, "On Convergence Rate of Concave-Convex Procedure", NIPS 2012 OPT Workshop (not considered a publication by the way)

- kernels:
  - example of early paper presenting kernels for structured SVM: Juho Rousu, Craig Saunders, Sandor Szedmak, John Shawe-Taylor, "Kernel-Based Learning of Hierarchical Multilabel Classification Models", JMLR 2006
  - for computation in BCFW, see Appendix B.5 in the usual: S. Lacoste-Julien, M. Jaggi, M. Schmidt and P. Pletscher, “Block-Coordinate Frank-Wolfe Optimization for Structural SVMs”, ICML 2013

- deep learning:
- see chapter 10 of the "Deep learning book" for RNNs
- head detection plug in example mentioned in class: Tuan-Hung Vu, Anton Osokin, and Ivan Laptev, "Context-aware CNNs for person head detection", ICCV 2015