Regression

Linear Regression: Model $P(w|X,\phi) = \mathbb{N}_w[X^T\phi, \sigma^2 I]$

Solution: Maximum Likelihood

$$\hat{\phi}, \hat{\sigma}^2 = \underset{\phi,\sigma^2}{\operatorname{argmax}}\left[-\frac{I\log(\sigma^2)}{2} - \frac{I\log(2\pi)}{2} + \frac{(w-X^T\phi)^T(w-X^T\phi)}{2\sigma^2}\right]$$

$$\Rightarrow \quad \hat{\phi} = (XX^T)^{-1}Xw$$

$$\sigma^2 = \frac{(w-X^T\phi)^T(w-X^T\phi)}{I}$$

Bayesian Regression: Model $p(\phi|X,w) = \frac{P(w|X,\phi)\,p(\phi)}{P(w)}$

Closed-form formulation:

$$P(\phi|X,w) = \mathbb{N}_\phi[\tfrac{1}{\sigma^2}A^{-1}Xw, A^{-1}] \text{ with } A = \tfrac{1}{\sigma^2}XX^T + \tfrac{1}{\sigma_p^2}I$$

Reformulation: $A^{-1} = (\tfrac{1}{\sigma^2}XX^T + \tfrac{1}{\sigma_p^2}I) = \sigma_p^2 I_D - \sigma_p^2 X(X^TX + \tfrac{\sigma^2}{\sigma_p^2}I_I)^{-1}X^T$

$D<I$ $\qquad$ $D>I$

Likelihood: $P(w|X,\phi) = \mathbb{N}_w[X^T\phi, \sigma^2 I] = \mathbb{N}_\phi[(\tfrac{1}{\sigma^2}XX^T)^{-1}\tfrac{1}{\sigma^2}Xw, (\tfrac{1}{\sigma^2}XX^T)^{-1}]$

Prior: $P(\phi) = \mathbb{N}_\phi[0, \sigma_p^2 I]$

Inference: Integrate over the parameter space

$$\begin{aligned} P(w^*|x^*,X,w) &= \int P(w^*|x^*,\phi)\,P(\phi|X,w)\,d\phi \\ &= \int \mathbb{N}_w[X^T\phi, \sigma^2 I]\,\mathbb{N}_\phi[\tfrac{1}{\sigma^2}A^{-1}Xw, A^{-1}]\,d\phi \\ &= \mathbb{N}_{w^*}[\tfrac{1}{\sigma^2}x^{*T}A^{-1}Xw, x^{*T}A^{-1}x^* + \sigma^2] \end{aligned}$$

Fitting variance:

$$\begin{aligned} P(w|X,\sigma^2) &= \int P(w|X,\phi,\sigma^2)\,P(\phi)\,d\phi \\ &= \int \mathbb{N}_w[X^T\phi, \sigma^2 I]\,\mathbb{N}_\phi[0, \sigma_p^2 I]\,d\phi \\ &= \mathbb{N}_w[0, \sigma_p^2 X^TX + \sigma^2 I] \end{aligned}$$

Non-linear regression: Model $P(w|z,\phi)$ with $z = \varphi(X)$

Solution:

$$\hat{\phi} = (ZZ^T)^{-1}Zw$$

$$\sigma^2 = \frac{(w-Z^T\phi)^T(w-Z^T\phi)}{I}$$

Gaussian Process Regression:

Model:

$$P(w^* | x^*, X, w)$$
$$= \mathbb{N}_{w^*}\left[\frac{\sigma_p^2}{\sigma^2} k[x^*, X] w - \frac{\sigma_p^2}{\sigma^2} k[x^*, X]\left(k[x^*, X] + \frac{\sigma^2}{\sigma^2} I\right)^{-1} k[X, X] w,\right.$$
$$\left.\sigma_p^2 k[x^*, x^*] - \sigma_p^2 k[x^*, X]\left(k[X, X] + \frac{\sigma^2}{\sigma_p^2} I\right)^{-1} k[X, x^*] + \sigma^2\right]$$

Sparse Linear Regression

Prior: $P(x) = \text{Stud}_x[M, \Sigma, \nu] = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sqrt{\nu\pi\sigma^2}\,\Gamma\left(\frac{\nu}{2}\right)}\left(1 + \frac{(x-\mu)^2}{\nu\sigma^2}\right)^{-\frac{\nu+1}{2}}$

Approximation:

$$P(w | X, \sigma^2) = \max_H \left[\mathbb{N}_w[0, X^T H^{-1} X + \sigma^2 I] \prod_{d=1}^{D} \text{Gam}_{h_d}[\nu/2, \nu/2]\right]$$

Learning: Learn H that maximizes above equation

Iteratively update posterior:

$$P(\phi | X, w) = \mathbb{N}_\phi\left[\frac{1}{\sigma^2} A^{-1} X w, A^{-1}\right] \text{ with } A = \frac{1}{\sigma^2} X X^T + H$$

Estimate H: $h_d^{new} = \frac{1 - h_d \Sigma_{dd} + \nu}{\mu_d^2 + \nu}$

Dual Linear Regression: Model $P(w | X, \Theta) = \mathbb{N}_w[X^T X \psi, \sigma^2 I]$

Learning: Maximize Likelihood

$$\hat{\psi}, \hat{\sigma}^2 = \underset{\psi, \sigma}{\text{argmax}}\left[\frac{I \log(2\pi)}{2} + \frac{I \log(\sigma^2)}{2} + \frac{(w - X^T X \psi)^T (w - X^T X \psi)}{2\sigma^2}\right]$$

$$\Rightarrow \hat{\psi} = (X X^T)^{-1} w$$

$$\hat{\sigma}^2 = \frac{(w - X^T X \psi)^T (w - X^T X \psi)}{I}$$

Dual Bayesian Regression:

Model: $P(\psi | X, w, \sigma^2) = \frac{P(w | X, \psi, \sigma^2) P(\psi)}{P(w | X, \sigma^2)} = \mathbb{N}_\psi\left[\frac{1}{\sigma^2} A^{-1} X^T X w, A^{-1}\right]$

With: $A = \frac{1}{\sigma^2} X^T X X^T X + \frac{1}{\sigma_p^2} I$

Relevance Vector Regression

Model $P(w | X, \Theta) = \mathbb{N}_w[X^T X \psi, \sigma^2 I]$

with $P(\psi) = \prod_{i=1}^{I} \text{Stud}_\psi[0, 1, \nu]$

Learning: same as sparse solution

Classification

Logistic Regression:

Model $P(w|X,\phi) = \text{Bern}\left[\frac{1}{1+\exp(-\phi^T X)}\right]$

$$= \left(\frac{1}{1+\exp(-\phi^T X)}\right)^{w} \left(1 - \frac{1}{1+\exp(\phi^T X)}\right)^{(1-w)}$$

Log-likelihood:

$$L = \sum_{i=1}^{I} w_i \log\left[\frac{1}{1+\exp(\phi^T x)}\right] + \sum_{i=1}^{I} (1-w_i) \log\left[\frac{\exp(\phi^T x)}{1+\exp(\phi^T x)}\right]$$

Derivatives:

$$\frac{\partial L}{\partial \phi} = -\sum_{i=1}^{I} \left(\frac{1}{1+\exp(\phi^T x_i)} - w_i\right) x_i = -\sum_{i=1}^{I} (\text{sig}[\phi^T x] - w_i) x_i$$

$$\frac{\partial^2 L}{\partial \phi^2} = -\sum_{i=1}^{I} \text{sig}[\phi^T x_i](1 - \text{sig}[\phi^T x_i]) x_i x_i^T$$

Optimization: $\theta^{t+1} = \theta^t - \lambda \left(\frac{\partial^2 L}{\partial \phi^2}\right)^{-1} \frac{\partial L}{\partial \phi}$

Bayesian Logistic Regression:

Model: $P(\phi|X,w) = \frac{P(w|X,\phi)P(\phi)}{P(w|X)}$

Likelihood: $P(w|X,\phi) = \prod_{i=1}^{I} \left(\frac{1}{1+\exp(\phi^T x_i)}\right)^{w_i} \left(\frac{\exp(\phi^T x_i)}{1+\exp(\phi^T x_i)}\right)^{(1-w_i)}$

Prior: $P(\phi) = N_\phi[0, \sigma_p^2 I]$

Learning: Find MAP solution using Newton's method

Log-posterior: $L = \sum_{i=1}^{I} \log[P(w|X,\phi)] + \log[P(\phi)]$

Derivatives:

$$\frac{\partial L}{\partial \phi} = -\sum_{i=1}^{I} (\text{sig}[\phi^T x] - w_i) x_i - \frac{\phi}{\sigma_p^2}$$

$$\frac{\partial^2 L}{\partial \phi^2} = -\sum_{i=1}^{I} \text{sig}[\phi^T x_i](1 - \text{sig}[\phi^T x_i]) x_i x_i^T - \frac{1}{\sigma_p^2}$$

Laplace Approximation: $P(\phi|X,w) \approx q(\phi) = N_\phi[\mu, \Sigma]$

Inference:

$$P(w^*|x^*,X,w) = \int P(w^*|x^*,\phi) P(\phi|X,w)\, d\phi$$

$$\approx \frac{1}{1+\exp[-\mu^T x^* / \sqrt{1+\pi(x^{*T}\Sigma x^*)/8}]}$$

Non-linear Logistic Regression

Model: $P(w|x,\phi) = \text{Bern}_w[\phi^T f(x)]$

Dual Logistic Regression:

Model: $P(w|X,\psi) = \text{Bern}_w[\text{sig}[X^T X\psi]]$

Derivatives: $\frac{\partial L}{\partial \psi} = -\sum_{i=1}^{I} (\text{sig}[a_i] - w_i) X^T x_i$

$$\frac{\partial^2 L}{\partial \psi^2} = -\sum_{i=1}^{I} \text{sig}[a_i](1-\text{sig}[a_i]) X^T x_i x_i^T X$$

Relevance Vector Classification

Model: $P(w|X) = \int P(w|X,\psi)P(\psi)\,d\psi$

$$= \int\int \prod_{i=1}^{I} \text{Bern}_{w_i}[\text{sig}[\psi^T k[X,x_i]]]\,\text{N}_\psi[0,H^{-1}]\,\text{Gam}_H[\nu/2,\nu/2]\,dH\,d\psi$$

Prior: $P(\psi) = \prod_{i=1}^{I} \text{Stud}_{\psi_i}[0,1,\nu]$

$$= \prod_{i=1}^{I} \int \text{N}_{\psi_i}[0,\tfrac{1}{h_i}]\,\text{Gam}_{h_i}[\tfrac{\nu}{2},\tfrac{\nu}{2}]\,dh_i$$

$$= \int \text{N}_{\psi_i}[0,H^{-1}]\,\text{Gam}_{h_i}[\nu/2,\nu/2]\,dH$$

Random Forests

Splitting function: **Axis-aligned:** $h(v,\Theta) = \phi(v) \cdot \psi > \tau$
with: $\phi(v) = (x_1, x_2)$
$\psi = (0,1)/(1,0)$

Oriented line: $h(v,\Theta) = \phi(v)\ \psi > \tau$
with: $\phi(v) = (x_1, x_2)$
$\psi = (a,b) \in \mathbb{R}^2$

Conic section: $h(v,\Theta) = \phi(v)^T \psi\, \phi(v)$
with: $\phi(v) = (x_1, x_2)$
$\psi \in \mathbb{R}^{3\times 3}$

Predictor model: **Classification:** Histogram over training samples

Regression: **Constant:** $y = \text{const.}$

Polynomial: $y = \sum_{i=0}^{n} w_i x^i$

Prob. linear: $y = L_1 x + L_2$

Density: $\mathbb{N}(v; \mu_{L(v)}, \Lambda_{L(v)})$

Training: **Classification:** **IG:** $I = H(S_j) - \sum_{i\in L,R} \frac{|S_j^i|}{|S_j|} H(S_j^i)$

Entropy: $H(S) = -\sum_{c\in C} p(c) \log(p(c))$

Regression: **IG:** $I = \sum_{(x,y)\in S_j} \log(\sigma_y(x)) - \sum_{i\in R,L} \left(\sum_{(x,y)\in S_j^i} \log(\sigma_y(x)) \right)$

Entropy: $H(S) = \frac{1}{|S|} \sum_{x\in S} \int_y p(y|x) \log p(y|x)\, dy$
$= \frac{1}{|S|} \sum_{x\in S} \frac{1}{2} \log((2\pi e)^2 \sigma_y^2(x))$

CART objective: $E = \sum_{(x,y)\in S_j} (y - \bar{y}_j)^2 - \sum_{i\in R,L} \left(\sum_{(x,y)\in S_j^i} (y - y_j)^2 \right)$

Density: **IG:** $I = H(S_j) - \sum_{i\in R,L} \frac{|S_j^i|}{|S_j|} H(S_j^i)$

Entropy: $H(S) = \frac{1}{2} \log((2\pi e)^d |\Lambda(S)|)$

Ensemble model: $p(y|v) = \frac{1}{T} \sum_t^T p_t(y|v)$

Density: $p_t(v) = \frac{\pi_{L(v)}}{Z_t} \mathbb{N}(v; \mu_{L(v)}, \Lambda_{L(v)})$

with $Z_t = \int \pi_{L(v)} \mathbb{N}(v; \mu_{L(v)}, \Lambda_{L(v)})\, dv$

Convergence theorem: $\mathbb{E}_{X, A(n), Z_t}\left[\left| \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[\Theta | X, A(n), Z_t] - \mathbb{E}[\Theta | X] \right|^2 \right] \to 0$

for $k_n \to \infty,\ k_n/n \to 0,\ n \to \infty$

Maximum Margin Classification and Kernels

Classification Margin: $\gamma = \frac{\langle w, x_i \rangle + b}{\|w\|}$

Maximum Margin Classification

$$\max_{\substack{\gamma \in \mathbb{R}^+ \\ w \in \mathbb{R}^d}} \gamma \quad \text{subject to:} \quad y_i \langle w, x_i \rangle \geq \gamma$$

$$\Leftrightarrow \min_{w \in \mathbb{R}^d} \|w\|^2 \quad \text{subject to:} \quad y_i \langle w, x_i \rangle \geq 1$$

Soft-Margin Classification

$$\min_{\substack{w \in \mathbb{R}^d \\ \xi_i \in \mathbb{R}^+}} \|w\|^2 + C \sum_i \xi_i \quad \text{subject to:} \quad y_i \langle w, x_i \rangle \geq 1 - \xi_i$$

Hinge-Loss Formulation

$$\min_{w \in \mathbb{R}^d} \|w\|^2 + C \sum_{i=1}^n (1 - y_i \langle w, x_i \rangle) \quad \text{with} \quad [t]_+ = \max\{0, t\}$$

Kernel Trick

$$\min_{\substack{\alpha_i \in \mathbb{R} \\ \xi_i \in \mathbb{R}^+}} \sum_{i,j} \alpha_i \alpha_j k[x_i, x_j] + C \sum_i \xi_i \quad \text{subject to:} \quad y_i \sum_{j=1}^n \alpha_j k[x_i, x_j] \geq 1 - \xi_i$$

Dual Formulation: Dualization with Langrian multipliers

$$\min_{\alpha_i} \sum_{i,j} \alpha_i \alpha_j y_i y_i k[x_i, x_j] + C \sum_i \alpha_i \quad \text{subject to:} \quad \sum_i y_i \alpha_i = 0$$

Multiple Kernel Learning

$$\min_{\substack{v_j \in \mathcal{H}_j, \xi_i \in \mathbb{R}^+ \\ \beta_i \geq 0, \sum \beta_i = 1}} \sum_j \frac{1}{\beta_j} \|v_j\|^2_{\mathcal{H}_j} + C \sum_i \xi_i \quad \text{subject to:} \quad y_i \sum_j \langle v_j, \varphi(x_i) \rangle_{\mathcal{H}_j} \geq 1 - \xi_i$$

Support Vector Regression

$$\min_{\substack{w \in \mathcal{H} \\ \xi_i \in \mathbb{R}^+, \xi_i' \in \mathbb{R}^+}} \|w\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i') \quad \text{subject to:} \quad y_i - \langle w, \varphi(x_i) \rangle \leq \varepsilon + \xi_i$$

$$\langle w, \varphi(x_i) \rangle - y_i \leq \varepsilon + \xi_i'$$

Dual Support Vector Regression

$$\min_{\substack{\alpha_i \in \mathbb{R} \\ \xi_i \in \mathbb{R}^+, \xi_i' \in \mathbb{R}^+}} \sum_{i,j} \alpha_i \alpha_j k[x_i, x_j] + C \sum_i (\xi_i + \xi_i') \quad \text{subject to:} \quad y_i - \sum_j \alpha_j k[x_i, x_j] \leq \varepsilon + \xi_i$$

$$\sum_j \alpha_j k[x_i, x_j] - y_i \leq \varepsilon + \xi_i$$

Multiclass SVM

Feature embeding: $\phi(x, y) = (\llbracket y = 1 \rrbracket \phi(x), \ldots, \llbracket y = k \rrbracket \phi(x))$

$$\min_{w \in \mathbb{R}^d, \xi_i \in \mathbb{R}^+} \frac{1}{2} \|w\|^2 + \frac{C}{N} \sum_{i=1}^N \xi_i \quad \text{subject to:}$$

$$\forall_i : \forall_{y \in Y} \langle w, \phi(x_i, y_i) \rangle - \langle w, \phi(x_i, y) \rangle \geq 1 - \xi_i$$

Structured Support Vector Machine

Regularized Risk Minimization:

$$\min_{w \in \mathbb{R}^d} \lambda \|w\|^2 + \frac{C}{N} \sum_{i=1}^{N} \max_{y \in Y} [\Delta(y^n, y) + \langle w, \phi(x^n, y) \rangle - \langle w, \phi(x^n, y^n) \rangle]$$

Equivalent Formulations:

1. $$\min_{w, \xi} \frac{1}{2} \|w\|^2 + \frac{C}{N} \sum_{n=1}^{N} \xi_n$$

   subject to:

   $$\forall i : \max_{y \in Y} [\Delta(y^n, y) + \langle w, \phi(x^n, y) \rangle - \langle w, \phi(x^n, y^n) \rangle] \leq \xi_n$$

2. $$\min_{w, \xi} \frac{1}{2} \|w\|^2 + \frac{C}{N} \sum_{n=1}^{N} \xi_n$$

   subject to:

   $$\forall i : \forall y \in Y : \Delta(y^n, y) + \langle w, \phi(x^n, y) \rangle - \langle w, \phi(x^n, y^n) \rangle \leq \xi_n$$

3. $$\min_{w, \xi} \|w\|^2 + \frac{C}{N} \sum_{n=1}^{N} \xi_n$$

   subject to:

   $$\forall i : \forall y : \langle w, \delta\phi(x^n, y^n, y) \rangle \geq \Delta(y^n, y) - \xi_n$$

4. $$\min_{w, \xi} \frac{1}{2} \|w\|^2 + C\xi$$

   subject to:

   $$\forall (\hat{y}^1, \ldots, \hat{y}^n) \in Y \times \ldots \times Y : \sum_{n=1}^{N} [\Delta(y^n, \hat{y}^n) + \langle w, \phi(x^n, \hat{y}^n) \rangle - \langle w, \phi(x^n, y^n) \rangle] \leq N\xi$$

Dual Structured Support Vector Machine

$$\max_{\alpha \in \mathbb{R}_+^{N|Y|}} \sum_{\substack{n=1..N \\ y \in Y}} \alpha_{ny} \Delta(y^n, y) - \frac{1}{2} \sum_{\substack{y, \bar{y} \in Y \\ n, \bar{n} = 1..N}} \alpha_{ny} \alpha_{\bar{n}\bar{y}} \langle \delta\phi(x^n, y^n, y), \delta\phi(x^{\bar{n}}, y^{\bar{n}}, \bar{y}) \rangle$$

Kernel: Function $k: \mathcal{X} \times \mathcal{X} \to \mathbb{R}$:

1. $k[x_1, x_2] = k[x_2, x_1]$
2. Kernel matrix $K_{ij} := (k[x_i, x_j])_{ij}$ is positive (semi-)definite: $\forall t \in \mathbb{R}^n: \sum_{i,j=1}^{n} t_i K_{ij} t_j \geq 0$
   $\Leftrightarrow$ all eigenvalues $\lambda_i \geq 0$

Embedding function: $k[x, x'] = \langle \varphi(x), \varphi(x') \rangle_{\mathcal{H}}$

Kernels:

Polynomial: $k[x, x'] = (1 + \langle x, x' \rangle)^m$, $m > 0$

Gaussian: $k[x, x'] = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right)$ $\sigma^2 \sim 0.6$

Set Kernels

Averaging Kernels:

Counting: $k[D_i, D_j] = \sum_{i=1}^{n_i} \sum_{j=1}^{n_j} \delta_{d_i = d_j}$

Averaging: $k[D_i, D_j] = \frac{1}{n_i n_j} \sum_{i=1}^{n_i} \sum_{j=1}^{n_j} k[d_i, d_j]$

Harmonic: $k[D_i, D_j] = \left(\prod_{i=1}^{n_i} \prod_{j=1}^{n_j} k[d_i, d_j] \right)^{\frac{1}{n_i n_j}}$

Matching Kernel: $k[D_i, D_j] = \frac{1}{2}\left[\hat{k}(D_i, D_j) + \hat{k}(D_j, D_i) \right]$

with: $\hat{k}[D_i, D_j] = \frac{1}{n_i} \sum_{i=1}^{n_i} \max k[d_i, d_j]$

Pyramid Match kernel:

$$k_\Delta(\underbrace{\Psi(X), \Psi(Y)}_{\text{histogram pyramids}}) = \sum_{i=0}^{L} \underbrace{\frac{1}{2^i}}_{\text{difficulty}} \underbrace{\left(I(H_i(X), H_i(Y)) - I(H_{i-1}(X), H_{i-1}(Y)) \right)}_{\text{newly matched pairs}}$$

Histogram Kernels

Unnormalized:

Linear kernel: $k[h, h'] = \sum_j h_j h'_j$

Polynomial kernel: $k[h, h'] = (c + \sum_j h_j h'_j)^m$

Gaussian kernel: $k[h, h'] = \exp\left(-\frac{1}{\gamma} \sum_j \|h_j - h'_j\|^2\right)$

Normalized:

Histogram Intersection: $k[h, h'] = \sum_j \min(h_j, h'_j)$

Bhattacharya: $k[h, h'] = \sum_j \sqrt{h_j h'_j}$

Symmetric KLD: $k[h, h'] = \exp\left(-\frac{1}{2}(KL(h|h') + KL(h'|h))\right)$

χ^2 kernel: $k[h, h'] = \exp\left(-\frac{1}{\gamma} \chi^2(h, h')\right)$

with: $\chi^2(h, h') = \sum_j \frac{(h_j - h'_j)^2}{h_j + h'_j}$

Neural Networks

McCullough-Pitts: $f(x,\theta) = \sum_{d=1}^{D} \theta_d x_d + \theta_0$

Fully-connected Layer: $f(x,\theta) = \sum_{n=0}^{N-1} \sum_{d=1}^{D} \theta_d \phi(x_n) + \theta_0$

Activation functions:

ReLU: $f(y) = \max(0,y)$

GELU: $f(y) = y \cdot P(Y \le y) \approx 0.5y\left(1 + \tanh\left[\sqrt{2/\pi}\left(x + 0.044715y^3\right)\right]\right)$

Sigmoid: $f(y) = \frac{1}{1+\exp(-y)}$

Softmax: $f(y) = \frac{e^y}{\sum_y e^y}$

Backpropagation:

Error: $E = \frac{1}{2}\sum_{n=1}^{N} (\hat{y}_n - y_n)^2$

Weight change: $\Delta w_{ij} = \eta \cdot \delta_j \cdot z_i$

Output Neuron: $\Delta w_{ij} = -\eta \frac{\partial E}{\partial z_j} \cdot \frac{\partial z_j}{\partial a_j} \cdot \frac{\partial a_j}{\partial w_{ij}} = \eta(\hat{y}_j - y_j) \cdot f'(a_j) \cdot z_j$

$\delta_j = (\hat{y}_j - y_j) \cdot f'(a_j)$

Hidden Neuron: $\Delta w_{ij} = -\eta\left(\sum_{k=1}^{k} \delta_k \cdot \frac{\partial\left(\sum_{h=1}^{H} z_h w_{hk}\right)}{\partial w_{jk}}\right) \frac{\partial z_i}{\partial a_j} \frac{\partial a_j}{\partial w_{ij}}$

$= \eta\left(\sum_{k=1}^{k} \delta_k w_{jk}\right) f'(a_j) z_j$

Bayesian NN: $R(\theta) = \frac{1}{N}\sum_{n=0}^{N} L(y_n - f(x_n)) + \lambda\|\theta\|^2$

Convolution: $(I * k)[x,y] = \sum_a \sum_b K[a,b] \cdot I[x-a, y-b]$

Segmentation Refinement

Energy function: $E(x) = \sum_i \psi_u(x_i) + \sum_{i<j} \psi_p(x_i, y_i)$

Unary: Inverse likelihood of segmentation

Pairwise: $\theta_{ij}(x_i,x_j) = \mu(x_i,x_j)\left[w_1 \exp\left(-\frac{\|p_i - p_j\|^2}{2\theta_\alpha^2} - \frac{\|I_i - I_j\|^2}{2\theta_\beta^2}\right) + w_2 \exp\left(-\frac{\|p_i - p_j\|^2}{2\theta_\gamma^2}\right)\right]$

Variational Lower Bound:

$\ln P(D) = \ln \int Q(\theta) \frac{P(D,\theta)}{Q(\theta)} d\theta \ge \int Q(\theta) \ln P(D|\theta) d\theta + \int Q(\theta) \ln \frac{1}{Q(\theta)} d\theta$

$= \ln P(D) - KL(Q(\theta) \| P(\theta|D))$

Generative Models

Markov Random Fields

General potential function: $P(w) = \frac{1}{Z}\prod_{j=1}^{J}\phi_j[w_{c_j}]$

MAP inference: $\hat{w}_{1\dots N} = \underset{1\dots N}{\operatorname{argmin}}\left[\sum_{n=1}^{N} U_n(w_n) + \sum_{(m,n)\in C} P_{mn}(w_m, w_n)\right]$

Binary Pairwise MRF: $P_\Theta(x) = \frac{1}{Z(\Theta)}\exp\left(\sum_{ij\in E} x_i x_j \Theta_{ij} + \sum_{i\in V} x_i \Theta_i\right)$

Log-likelihood: $L(\Theta) = \frac{1}{N}\sum_{n=1}^{N} \log P_\Theta(x^{(n)})$

Derivative: $\frac{\partial L(\Theta)}{\partial \Theta_{ij}} = \frac{1}{N}\sum_n [x_i^{(n)} x_j^{(n)}] - \sum_x [x_i x_j P_\Theta(x)] = \mathbb{E}_{P_{data}}[x_i x_j] - \mathbb{E}_{P_\Theta}[x_i x_j]$

Restricted Boltzmann Machines

Energy function: $E(v, h; \Theta) = -\sum_{ij} w_{ij} v_i h_j - \sum_i b_i v_i - \sum_j a_j h_j$

Joint distribution: $P_\Theta(v, h) = \frac{1}{Z(\Theta)}\exp(-E(v, h; \Theta))$

$$= \frac{1}{Z(\Theta)}\prod_{ij} e^{w_{ij} v_i h_j}\prod_i e^{b_i v_i}\prod_j e^{a_j h_j}$$

with: $Z(\Theta) = \sum_{h,v}\exp(-E(v, h; \Theta))$

Generative distribution: $P_\Theta(v) = \sum_h P_\Theta(v, h)$

$$= \frac{1}{Z(\Theta)}\prod_i e^{b_i v_i}\prod_j (1 + \exp(a_j + \sum_i w_{ij} v_i))$$

Model hidden variables: $P(h|v) = \prod_j P(h_j|v)$

with: $P(h_j = 1|v) = \frac{1}{1 + \exp(-\sum_i w_{ij} v_i - a_j)}$

Model visible variables: $P(v|h) = \prod_i P(v_i|h)$

with: $P(v_i = 1|h) = \frac{1}{1 + \exp(-\sum_j w_{ij} h_j - b_i)}$

Maximum Likelihood: $L(\Theta) = \frac{1}{N}\sum_{n=1}^{N} \log P_\Theta(v^{(n)}) - \frac{\lambda}{N}\|W\|_F^2$

Derivative:

$$\frac{\partial L(\Theta)}{\partial w_{ij}} = \frac{1}{N}\sum_{n=1}^{N}\frac{\partial}{\partial w_{ij}}\log\left(\sum_h \exp[v^{(n)T} W h + a^T h + b^T v^{(n)}]\right) - \frac{\partial}{\partial w_{ij}}\log Z(\Theta) - \frac{2\lambda}{N} w_{ij}$$

$$= \mathbb{E}_{P_{data}}[v_i h_j] - \mathbb{E}_{P_\Theta}[v_i h_j] - \frac{2\lambda}{N} w_{ij}$$

Weight update: $\Delta w_{ij} = \mathbb{E}_{P_{data}}[v_i h_j] - \mathbb{E}_{P_1}[v_i h_j]$

Gaussian-Bernoulli RBM

Energy function: $E(v, h; \Theta) = \sum_i \frac{(v_i - b_i)^2}{2\sigma_i^2} - \sum_{ij} w_{ij} h_j \frac{v_i}{\sigma_i} - \sum_j a_j h_j$

Model image: $P(v_i = x|h) = \frac{1}{\sqrt{2\pi}\sigma_i}\exp\left(\frac{(x_i - b_i - \sigma_i \sum_j w_{ij} h_j)^2}{2\sigma_i^2}\right)$

Model hidden variables: $P(h_j = 1|v) = \frac{1}{1 + \exp(-\sum_i w_{ij}\frac{v_i}{\sigma_i} - a_j)}$

Generative Adversarial Network

Discriminator Loss:

$$L(\theta) = \sum_j -\log[1-\text{sig}[f[g[z_j,\theta],\phi]]] - \sum_i \log[\text{sig}[f[x_i,\phi]]]$$

Generator Loss: $L(\theta) = \sum_j \log[1-\text{sig}[f[g[z_j,\theta],\phi]]$

Jenson-Shannon divergence:

$$D_{JS}[P(x^*)\|P(x)] = \frac{1}{2}\int P(x^*)\log\left[\frac{2P(x^*)}{P(x^*)+P(x)}\right]dx^* + \frac{1}{2}\int P(x)\log\left[\frac{2P(x)}{P(x)+P(x^*)}\right]dx$$

Wasserstein GAN

Lipschitz continuous: $|D(x_2)-D(x_1)| \leq |x_2-x_1|$

Lipschitz penalty: $R_{LP} = \mathbb{E}_{(\hat{x},h)\sim P_{\hat{x},h}}[\max(0, \|\nabla_{\hat{x},h} D(\hat{x},h)\|_2 - 1)^2]$

Wasserstein distance: $D_w[P(x)\|q(x)] = \min_{\mathbb{P}}\left[\sum_{i,j}\mathbb{P}_{ij}|i-j|\right]$

such that: $\sum_j \mathbb{P}_{ij} = P(x=i)$

$\sum_i \mathbb{P}_{ij} = q(x=j)$

$\mathbb{P}_{ij} \geq 0$

Solution:

Primal: $D_w[P(x),q(x)] = \min_{\pi[:,:]} \iint \pi[x_1,x_2]\|x_1-x_2\|dx_1dx_2$

Dual: $D_w[P(x),q(x^*)] = \max_{f[x]}\left[\int P(x)f[x]dx - \int P(x^*)f[x]dx\right]$

subject to: $\|f[z_1]-f[z_2]\| \leq \beta\|z_1-z_2\|$

Diffusion Models

Forward process: $q(x_t|x_{t-1}) = \mathbb{N}(x_t; \sqrt{1-\beta_t}\,x_{t-1}, \beta_t I)$

$$q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1})$$

Closed-form: $q(x_t|x_0) = \mathbb{N}(x_t; \sqrt{\bar{\alpha}_t}\,x_0, (1-\bar{\alpha}_t)I)$

with $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$

Sampling: $x_t = \sqrt{\bar{\alpha}_t}\,x_0 + \sqrt{1-\bar{\alpha}_t}\,\varepsilon$

Backward process: $p_\theta(x_{t-1}|x_t) = \mathbb{N}(x_{t-1}; \mu_\theta(x_t,t), \sigma_t^2 I)$

$$p_\theta(x_{0:T}) = p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)$$

Conditional:

$$p(z_{t-1}|z_t,x) = \mathbb{N}_{z_{t-1}}\left[\frac{(1-\alpha_{t-1})}{1-\alpha_t}\sqrt{1-\beta_t}\,z_t + \frac{\sqrt{\alpha_{t-1}}\beta_t}{1-\alpha_t}x_i,\ \frac{\beta_t(1-\alpha_{t-1})}{1-\alpha_t}\cdot I\right]$$

Loss: $L(\phi) = \sum_{i=1}^{I}\sum_{t=1}^{T}\|g_t[\sqrt{\alpha_t}x_i + \sqrt{1-\alpha_t}\cdot\varepsilon_{it}, \phi_t] - \varepsilon_{it}\|^2$

Gaussian Processes

Linear Latent Variable Model: $y_i = Wx_i + \varepsilon_i$ with $\varepsilon_i \sim \mathcal{N}[0, \sigma^2 I]$

Probabilistic PCA: $p(Y|W) = \prod_{i=1}^{n} \mathcal{N}(y_i | 0, WW^T + \sigma^2 I)$

Maximum likelihood solution:

For: $\log p(Y|W) = -\frac{n}{2}\log|C| - \frac{1}{2}tr(C^{-1}YY) + const.$

with: $C = WW^T + \sigma^2 I$

$\Rightarrow W = U_q L R^T$, $L = (\Lambda_q - \sigma^2 I)^{\frac{1}{2}}$

with U_q: first q principal eigenvectors of $n^{-1}Y^TY$

Dual Probabilistic PCA: $p(Y|W) = \prod_{j=1}^{p} \mathcal{N}(y_j | 0, XX^T + \sigma^2 I)$

Maximum likelihood solution:

For: $\log p(Y|W) = -\frac{p}{2}\log|K| - \frac{1}{2}tr(K^{-1}YY) + const.$

with: $K = XX^T + \sigma^2 I$

$\Rightarrow W = U'_q L R^T$, $L = (\Lambda_q - \sigma^2 I)^{\frac{1}{2}}$

with U'_q: first q principal eigenvectors of $p^{-1}Y^TY$

Equivalence of solutions: $U_q = Y^T U'_q \Lambda_q^{-\frac{1}{2}}$

Non-linear Latent Variable Models: $p(Y|X) = \prod_{j=1}^{p} \mathcal{N}(y_j | 0, K)$

K: non-linear kernel

Character Animation: $\underset{X,q}{\operatorname{argmin}} \left(\frac{\|W(y - f(x))\|^2}{2\sigma^2(x)} + \frac{D}{2}\ln\sigma^2(x) + \frac{1}{2}\|x\|^2 \right)$

such that: $C(q) = 0$

Continuous Dimensionality Reduction:

Prior: $\alpha \sum_{i=1}^{p} \phi(s_i)$ with: $\phi(s_i, r) = |s_i|^r$ s_i: eigenvalues

Optimization: $\min_{Y,\Theta} p(Y|X,\Theta)$ s.t $\forall i$ $s_i > 0$, $E(Y) - E(X) = 0$

with: $E(X) = \sum_i s_i^2$

Choice: $Q = \underset{i}{\operatorname{argmax}} \frac{s_i}{s_{i+1} + \varepsilon}$ with $\varepsilon \ll 1$ and $s_1 > s_2 > \ldots > s_D$

Shared-Private Factorization

Optimization: $L = L_{data} + L_{ortho} + L_{dim} + L_{energy}$

$L_{ortho} = \alpha \sum_i (\|X^T z^{(i)}\|_F^2 + \sum_{j \neq i} \|(z^{(i)})^T z^{(j)}\|_F^2)$

Multilinear Models

Style-Content Separation: $y = \sum_{i,j} w_{ij} a_i b_j + \varepsilon$

Multi-linear Analysis: $y = \sum_{i,j,k,\ldots} w_{ijk\ldots} a_i b_j c_k \ldots + \varepsilon$

Non-linear Basis Function: $y = \sum_{ij} w_{ij} a_i \phi_j(b) + \varepsilon$

Multi (non)-linear Models:

GP-LVM: $y = \sum_j w_j \phi_j(x) + \varepsilon = w^T \phi(x) + \varepsilon$

with: $E[y, y'] = \phi(x)^T \phi(x') + \beta^{-1}\delta$

$= k[x, x'] + \beta^{-1}\delta$

Multifactor Gaussian Process: $y = \sum_{ijk\ldots} w_{ijk\ldots} \phi_i^{(1)} \phi_j^{(1)} \phi_k^{(1)} + \varepsilon$

with: $E[y, y'] = \prod_i \phi^{(i)T} \phi^{(i)} + \beta^{-1}\delta$

$= \prod_i k_i[x^{(i)}, x^{(i)'}] + \beta^{-1}\delta$

Continuous Character Control

Prior: $\ln p(X) = w_c \sum_{ij} \ln K_{ij}^d$

Graph diffusion kernel: $K_{ij}^d = \exp(\beta H)$ with: $H = -T^{-1/2} L T^{-1/2}$

$T_{ii} = \sum_j w(x_i, x_j)$ → diagonal matrix and $w(x_i, x_j) = \|x_i - x_j\|^{-p}$

$$L_{ij} = \begin{cases} \sum_k w(x_i, x_k) & \text{if } i = j \\ -w(x_i, x_j) & \text{else} \end{cases}$$

Structured Learning

KL Divergence: $KL_{cond}(p\|d)(x) = \sum_{y\in Y} d(y|x) \log \frac{d(y|x)}{p(y|x,w)}$

Expected KL: $KL_{tot}(p\|d) = \mathbb{E}_{x\sim d(x)}[KL_{cond}(p\|d)(x)]$

$$= \sum_{x\in X}\sum_{y\in Y} d(x,y) \log \frac{d(y|x)}{p(y|x,w)}$$

Learning Techniques:

Maximum Likelihood:

$$w^* = \underset{w\in\mathbb{R}^d}{\operatorname{argmax}}\, p(y_1 \dots y_N | x_1,\dots,x_N,w)$$

$$= \underset{w\in\mathbb{R}^d}{\operatorname{argmax}} \prod_{n=1}^{N} p(y_n|x_n,w)$$

$$= \underset{w\in\mathbb{R}^d}{\operatorname{argmin}} -\sum_{n=1}^{N} \log p(y_n|x_n,w)$$

Best Approximation:

$$w^* = \underset{w\in\mathbb{R}^d}{\operatorname{argmin}} \sum_{x\in X}\sum_{y\in Y} d(x,y) \log \frac{d(y|x)}{p(y|x,w)}$$

$$= \underset{w\in\mathbb{R}^d}{\operatorname{argmin}} -\sum_{y\in Y} d(x,y) \log p(y|x,w)$$

$$= \underset{w\in\mathbb{R}^d}{\operatorname{argmin}} -\sum_{n=1}^{N} \log p(y_n|x_n,w)$$

MAP Estimation: $w^* = \underset{w\in\mathbb{R}^d}{\operatorname{argmax}}\, p(w|D) = \underset{w\in\mathbb{R}^d}{\operatorname{argmin}} [-\log p(w|D)]$

$$= \underset{w\in\mathbb{R}^d}{\operatorname{argmin}} \Big[-\log p(w) - \sum_{n=1}^{N} \log p(y_n|x_n,w) + \log p(y_n|x_n)\Big]$$

$$= \underset{w\in\mathbb{R}^d}{\operatorname{argmin}} \Big[-\log p(w) - \sum_{n=1}^{N} \log p(y_n|x_n,w)\Big]$$

Prior: $p(w) = \text{const.} \Rightarrow w^* = \underset{w\in\mathbb{R}^d}{\operatorname{argmin}} \Big[-\sum_{n=1}^{N} \log p(y_n|x_n,w) + \text{const.}\Big]$

$p(w) = \text{const.} \cdot e^{-\frac{1}{2\sigma^2}\|w\|^2}$

$$\Rightarrow w^* = \underset{w\in\mathbb{R}^d}{\operatorname{argmin}} \Big[\frac{1}{2\sigma^2}\|w\|^2 - \sum_{n=1}^{N} \log p(y_n|x_n,w) + \text{const.}\Big]$$

CRF Learning

Model: $p(y|x,w) = \frac{1}{Z(x,w)} \exp(\langle w, \phi(x,y)\rangle)$

Likelihood: $L(w) = \frac{1}{2\sigma^2}\|w\|^2 - \sum_{n=1}^{N} \Big[\langle w, \phi(x_n,y_n)\rangle - \log \sum_{y\in Y} e^{\langle w,\phi(x_n,y)\rangle}\Big]$

Gradient: $\nabla_w L(w) = \frac{1}{\sigma^2} w - \sum_{n=1}^{N} [\phi(x_n,y_n) - \mathbb{E}_{y\sim p(y|x_n,w)}[\phi(x_n,y)]]$

Hessian: $\Delta L(w) = \frac{1}{\sigma^2} \mathrm{Id}_{D\times D} + \sum_{n=1}^{N} \big(\mathbb{E}_{y\sim p(y|x_n,w)}[\phi(x_n,y)]\big)\big(\mathbb{E}_{y\sim p(y|x_n,w)}[\phi(x_n,y)]\big)^T$