Modified Gauss-Newton Algorithms under Noise

Krishna Pillutla, Vincent Roulet, Sham Kakade, Zaid Harchaoui

Google Research   HARVARD UNIVERSITY   UNIVERSITY of WASHINGTON

Setting

Consider the finite sum composite problem

$$F(w) = \frac{1}{n}\sum_{i=1}^{n} f \circ \phi_i(w) \qquad (1)$$

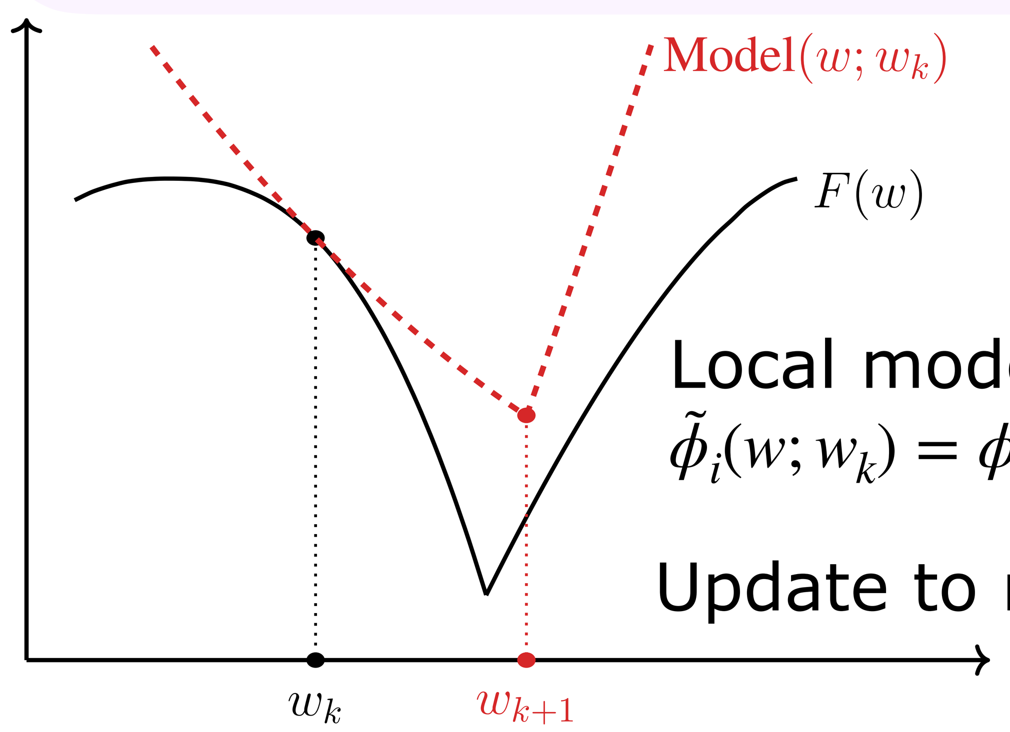
with $f : \mathbb{R}^m \to \mathbb{R}$ Lipschitz and convex (loss), $\phi_i : \mathbb{R}^d \to \mathbb{R}^m$ smooth and non-convex (predictor)

Examples:

- Robust Regression: $f = \|\cdot\|_2$
- Classification: $f =$ multi-class hinge loss

Prox-linear/ Modified Gauss-Newton Method

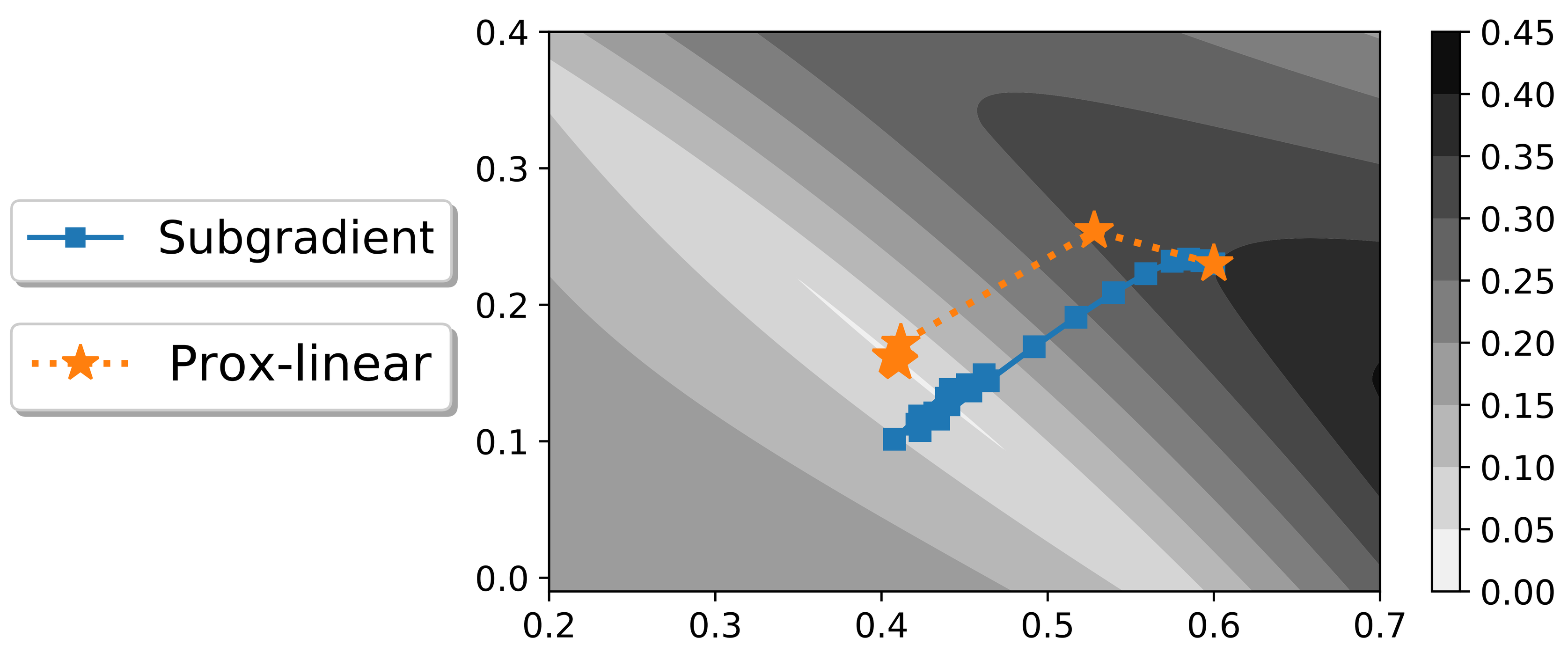
[Burke, 1985]



Local model by linearization
$\tilde{\phi}_i(w; w_k) = \phi_i(w_k) + \nabla \phi_i(w_k)^\top (w - w_k)$

Update to minimize the model

$$w_{k+1} = \arg\min_w \frac{1}{n}\sum_{i=1}^{n} f\big(\tilde{\phi}_i(w; w_k)\big) + \frac{\kappa}{2}\|w - w_k\|_2^2$$



Quadratic local convergence

Proposition: If f is ℓ-Lipschitz and μ-sharp, $\phi = (\phi_1; \cdots; \phi_n)$ is L-smooth and $\sigma_{\min}(\nabla \phi(w)^\top) \geq \nu > 0$, then $F(w_k) \to F^\star$ globally.

If $F(w_k) - F^\star \leq R$, then for all $t \geq k$:

$$F(w_{t+1}) - F^\star \leq \frac{1}{2R^2}\big(F(w_t) - F^\star\big)^2$$

$$\text{where} \quad R = \frac{\mu^2 \nu^2}{L \ell n^{3/2}}$$

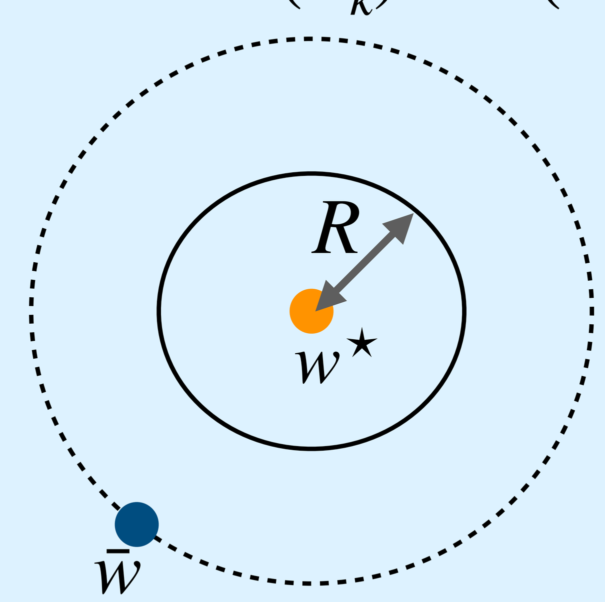
Statistical Trade-offs

Setting: $y_i = \psi(x_i; \bar{w}) + \xi_i$, where $\xi_i \sim \mathcal{N}(0, \sigma^2 I_m)$

Consider problem (1) with $\phi_i(w) = \psi(x_i, w)$

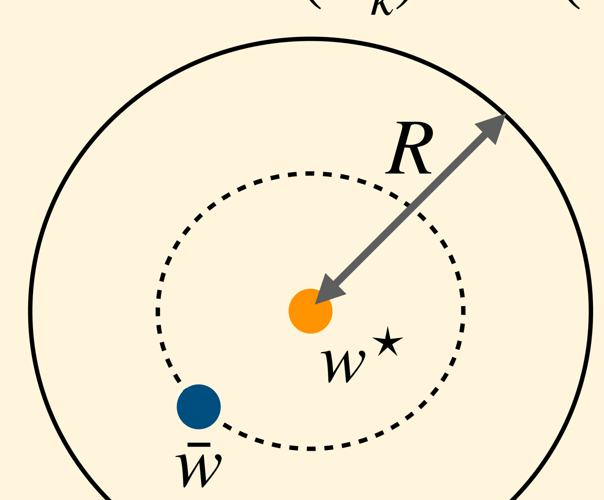
Proposition: Let R denote the radius of quadratic convergence and $\exists w^\star$ s.t. $y_i = \psi(x_i, w^\star)$ for all i. Let w_k be the first iterate enjoying quadratic convergence. Then w.h.p.,

(1) if $\sigma > \dfrac{R}{m^{1/2} - m^{1/4}}$ then $F(w_k) < F(\bar{w})$



σ large $\Longrightarrow$ quadratic convergence is not active

(2) if $\sigma < \dfrac{R}{m^{1/2} + m^{1/4}}$ then $F(w_k) > F(\bar{w})$



σ tiny $\Longrightarrow$ quadratic convergence is active

Experiments

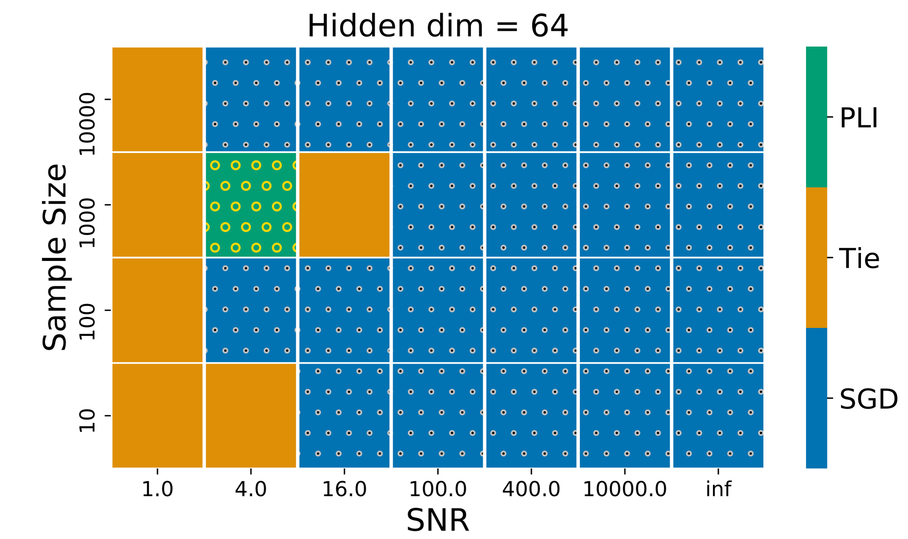
Robust multi-output regression

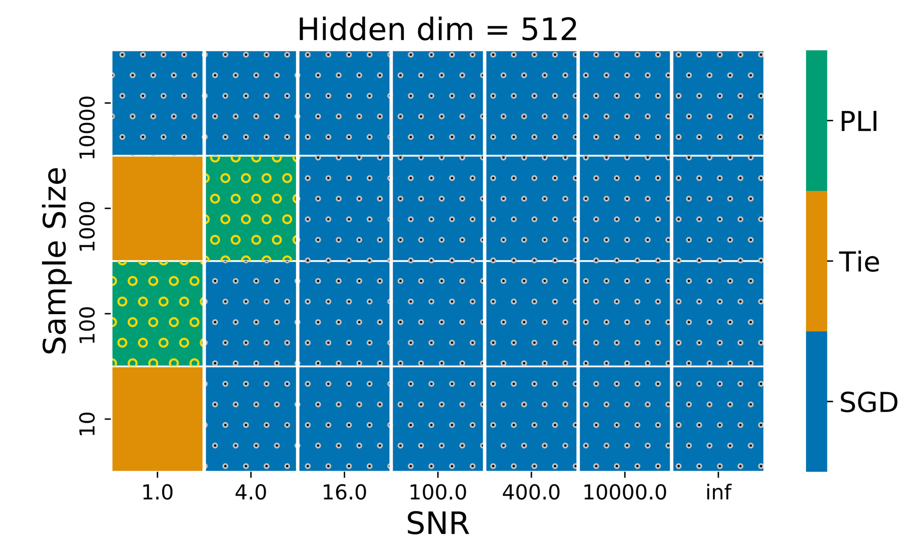
Input: $x \in \mathbb{R}^p$

Output: $y \in \mathbb{R}^m$ (same as the statistical setting)

Loss: $f = \|\cdot\|_2$

Model: 2-layer MLP





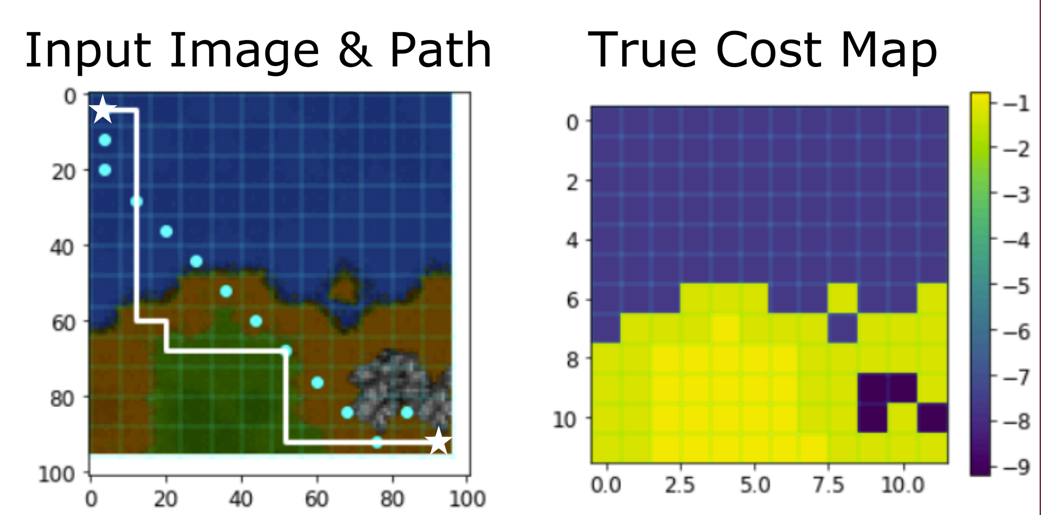
Path planning (structured prediction)

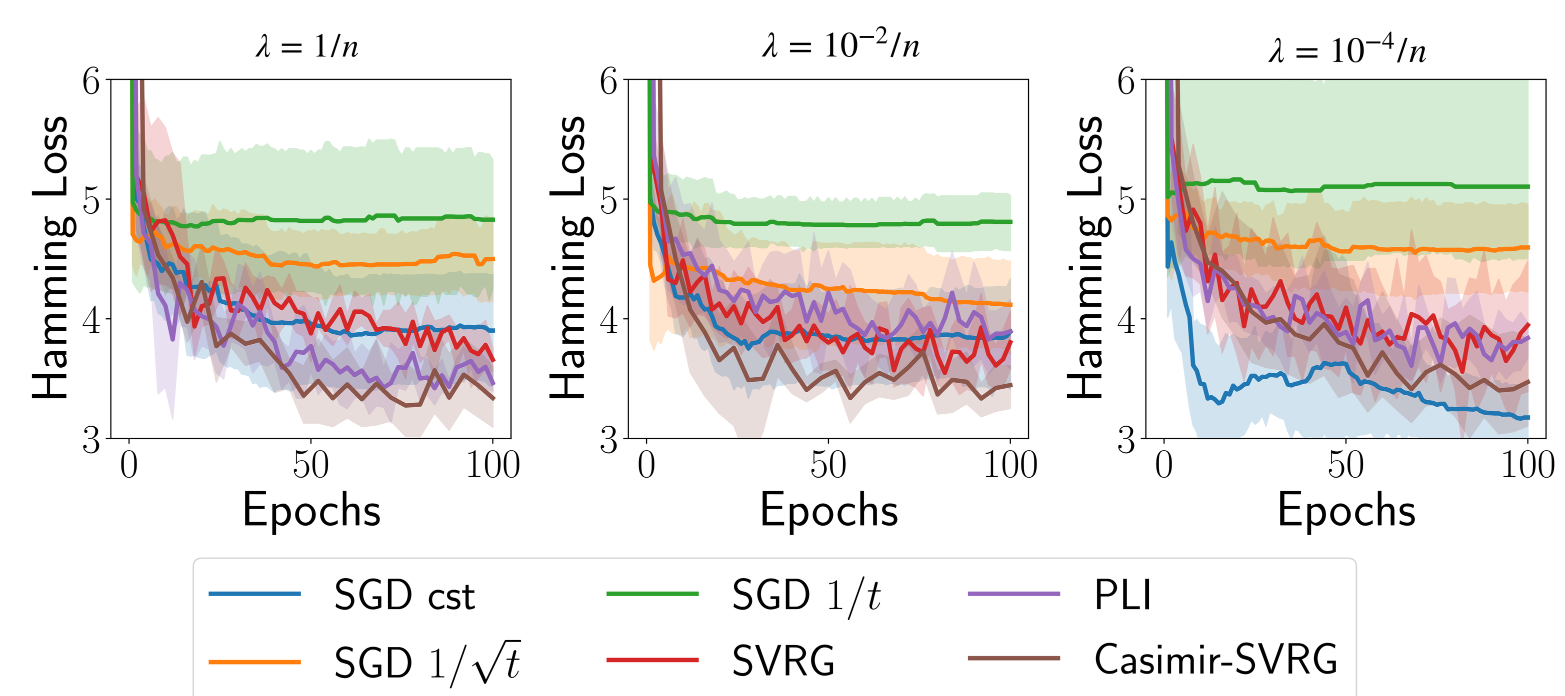
Input: Image of a Warcraft map

Output: Least cost path from start to finish

Loss: structural hinge loss

Model: convolutional net





krishnap25.github.io   KrishnaPillutla