Second Order Strikes Back

Globally Convergent Newton Methods for Ill-conditioned Generalized Self-concordant Losses

Francis Bach

INRIA - Ecole Normale Supérieure, Paris, France

Joint work with Ulysse Marteau-Ferey and Alessandro Rudi

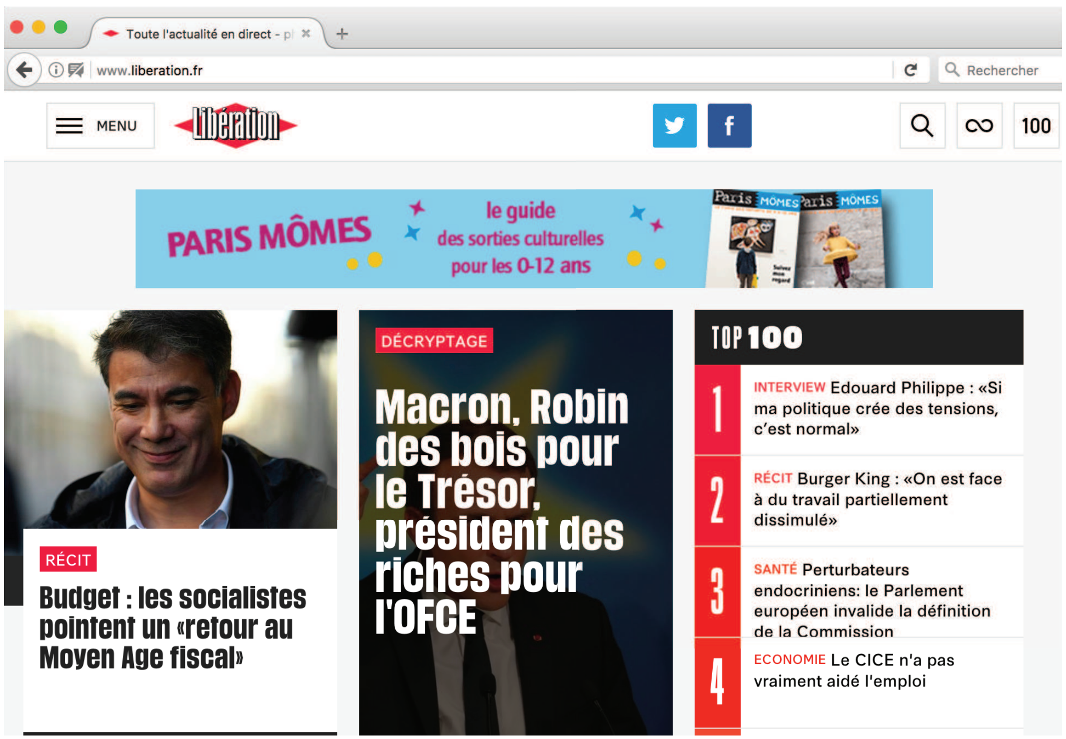
Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$

- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$

Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$
- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$

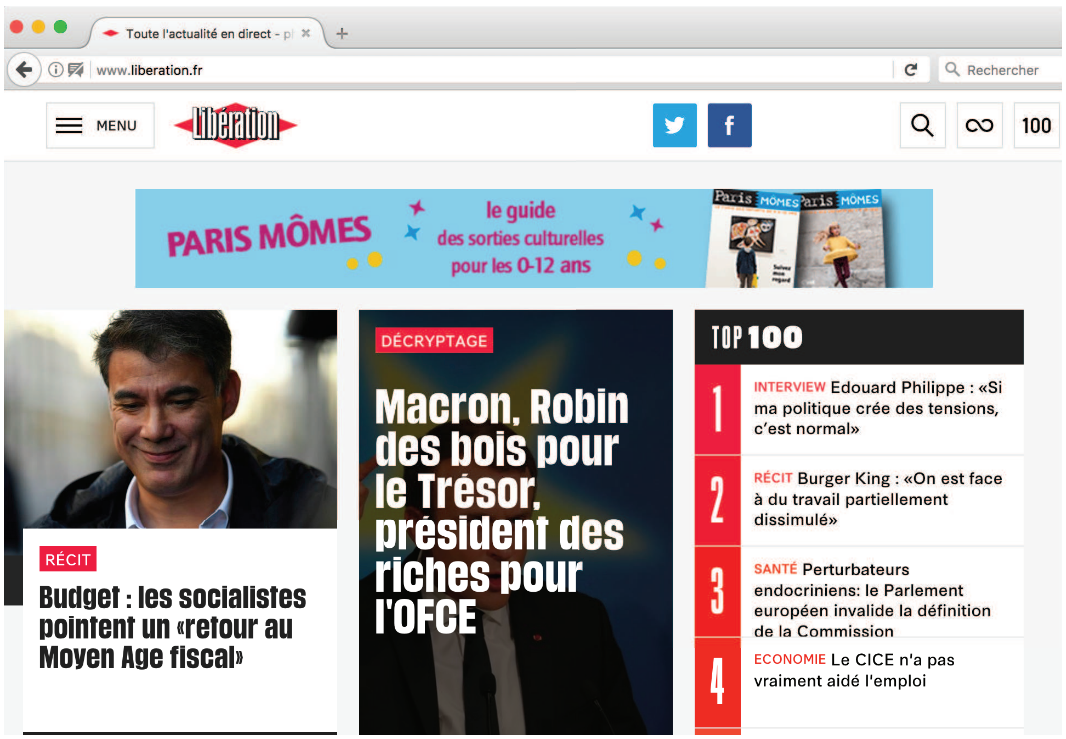


- **Advertising**: $n > 10^9$
  - $\Phi(x) \in \{0, 1\}^d$, $d > 10^9$
  - Navigation history + ad

- Linear predictions
  - $h(x, \theta) = \theta^\top \Phi(x)$

Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$
- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$



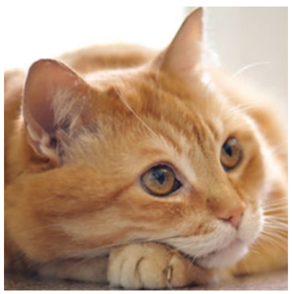
- **Advertising**: $n > 10^9$
  - $\Phi(x) \in \{0, 1\}^d$, $d > 10^9$
  - Navigation history + ad
- Linear predictions
  - $h(x, \theta) = \theta^\top \Phi(x)$
- Kernel methods
  - $k(x, x') = \Phi(x)^\top \Phi(x')$

Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$

- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$

x_1

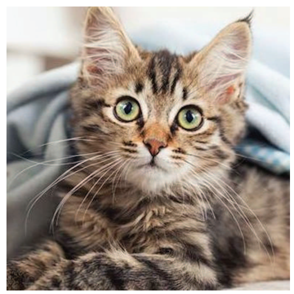


$y_1 = 1$

x_2

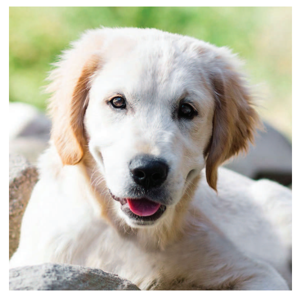
$y_2 = 1$

x_3



$y_3 = 1$

x_4

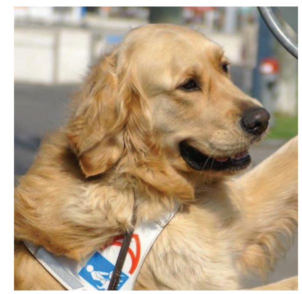


$y_4 = -1$

x_5

$y_5 = -1$

x_6



$y_6 = -1$

Parametric supervised machine learning

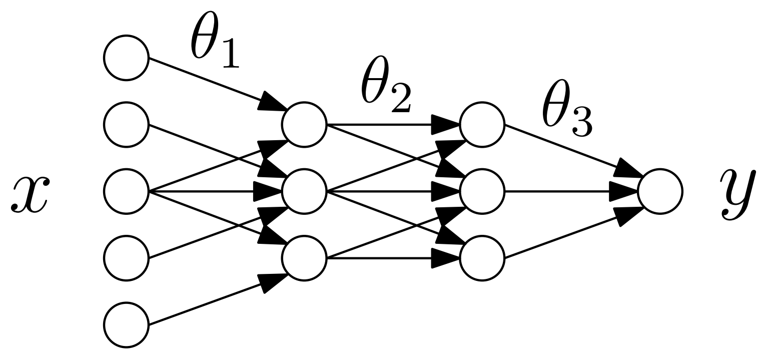
- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$

- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$

x_1 x_2 x_3 x_4 x_5 x_6

$y_1 = 1$ $y_2 = 1$ $y_3 = 1$ $y_4 = -1$ $y_5 = -1$ $y_6 = -1$

  – Neural networks $(n, d > 10^6)$: $h(x, \theta) = \theta_m^\top \sigma(\theta_{m-1}^\top \sigma(\cdots \theta_2^\top \sigma(\theta_1^\top x))$



Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$
- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$
- **(regularized) empirical risk minimization**:

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^{n} \quad \ell\big(y_i, h(x_i, \theta)\big) \quad + \quad \lambda \Omega(\theta)$$

data fitting term $+$ regularizer

Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$
- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$
- **(regularized) empirical risk minimization**:

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{2n} \sum_{i=1}^{n} \big(y_i - h(x_i, \theta)\big)^2 \quad + \quad \lambda \Omega(\theta)$$

(least-squares regression)

Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$

- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$

- **(regularized) empirical risk minimization**:

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^{n} \log\big(1 + \exp(-y_i h(x_i, \theta))\big) \quad + \quad \lambda \Omega(\theta)$$

(logistic regression)

Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$
- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$
- **(regularized) empirical risk minimization**:

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^{n} \quad \ell\big(y_i, h(x_i, \theta)\big) \quad + \quad \lambda \Omega(\theta)$$

data fitting term $+$ regularizer

Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$
- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$
- **(regularized) empirical risk minimization**:

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^{n} \ \ell\big(y_i, h(x_i, \theta)\big) \quad + \quad \lambda \Omega(\theta)$$

data fitting term + regularizer

- **Actual goal**: minimize test error $\mathbb{E}_{p(x,y)} \ell(y, h(x, \theta))$

Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$
- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$
- **(regularized) empirical risk minimization**:

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^{n} \quad \ell\big(y_i, h(x_i, \theta)\big) \quad + \quad \lambda \Omega(\theta) \qquad = \frac{1}{n} \sum_{i=1}^{n} f_i(\theta)$$

data fitting term + regularizer

- **Actual goal**: minimize test error $\mathbb{E}_{p(x,y)} \ell(y, h(x, \theta))$
- **Machine learning through large-scale optimization**

Parametric supervised machine learning

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \ldots, n$
- **Prediction function** $h(x, \theta) \in \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^d$
- **(regularized) empirical risk minimization**:

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^{n} \ \ell\big(y_i, h(x_i, \theta)\big) \quad + \quad \lambda \Omega(\theta) \qquad = \frac{1}{n} \sum_{i=1}^{n} f_i(\theta)$$

data fitting term $+$ regularizer

- **Actual goal**: minimize test error $\mathbb{E}_{p(x,y)} \ell(y, h(x, \theta))$
- **Machine learning through large-scale optimization**
  - Convex vs. non-convex optimization problems

Stochastic vs. deterministic methods

- Minimizing $g(\theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\theta)$ with $f_i(\theta) = \ell\big(y_i, h(x_i, \theta)\big) + \lambda\Omega(\theta)$

Stochastic vs. deterministic methods

- Minimizing $g(\theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\theta)$ with $f_i(\theta) = \ell\big(y_i, h(x_i, \theta)\big) + \lambda\Omega(\theta)$

- Condition number
  - $\kappa =$ ratio between largest and smallest eigenvalues of Hessians
  - Typically proportional to $1/\lambda$ when $\Omega = \|\cdot\|^2$.

Stochastic vs. deterministic methods

- Minimizing $g(\theta) = \dfrac{1}{n}\displaystyle\sum_{i=1}^{n} f_i(\theta)$ with $f_i(\theta) = \ell\big(y_i, h(x_i, \theta)\big) + \lambda\Omega(\theta)$

- Batch gradient descent: $\theta_t = \theta_{t-1} - \gamma \nabla g(\theta_{t-1}) = \theta_{t-1} - \dfrac{\gamma}{n}\displaystyle\sum_{i=1}^{n} \nabla f_i(\theta_{t-1})$

  – Exponential convergence rate in $O(e^{-t/\kappa})$ for convex problems
  – Can be accelerated to $O(e^{-t/\sqrt{\kappa}})$ (Nesterov, 1983)
  – Iteration complexity is linear in n, typically $O(nd)$

Stochastic vs. deterministic methods

- Minimizing $g(\theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\theta)$ with $f_i(\theta) = \ell\big(y_i, h(x_i, \theta)\big) + \lambda\Omega(\theta)$

- Batch gradient descent: $\theta_t = \theta_{t-1} - \gamma \nabla g(\theta_{t-1}) = \theta_{t-1} - \frac{\gamma}{n}\sum_{i=1}^{n} \nabla f_i(\theta_{t-1})$
  - Exponential convergence rate in $O(e^{-t/\kappa})$ for convex problems
  - Can be accelerated to $O(e^{-t/\sqrt{\kappa}})$ (Nesterov, 1983)
  - Iteration complexity is linear in n, typically $O(nd)$

- Stochastic gradient descent: $\theta_t = \theta_{t-1} - \gamma_t \nabla f_{i(t)}(\theta_{t-1})$
  - Sampling with replacement: $i(t)$ random element of $\{1, \dots, n\}$
  - Convergence rate in $O(\kappa/t)$
  - Iteration complexity is independent of n, typically $O(d)$

Recent progress in single machine optimization

- **Variance reduction**
  - Exponential convergence with $O(d)$ iteration cost
  - SAG (Le Roux, Schmidt, and Bach, 2012)
  - SVRG (Johnson and Zhang, 2013; Zhang et al., 2013)
  - SAGA (Defazio, Bach, and Lacoste-Julien, 2014), etc...

$$\theta_t = \theta_{t-1} - \gamma \Big[\nabla f_{i(t)}(\theta_{t-1}) \qquad\qquad\qquad\qquad \Big]$$

Recent progress in single machine optimization

- **Variance reduction**
  - Exponential convergence with $O(d)$ iteration cost
  - SAG (Le Roux, Schmidt, and Bach, 2012)
  - SVRG (Johnson and Zhang, 2013; Zhang et al., 2013)
  - SAGA (Defazio, Bach, and Lacoste-Julien, 2014), etc...

$$\theta_t = \theta_{t-1} - \gamma \Big[\nabla f_{i(t)}(\theta_{t-1}) + \frac{1}{n} \sum_{i=1}^{n} z_i^{t-1} - z_{i(t)}^{t-1} \Big]$$

(with z_i^t stored value at time t of gradient of the i-th function)

Recent progress in single machine optimization

- **Variance reduction**
  - Exponential convergence with $O(d)$ iteration cost
  - SAG (Le Roux, Schmidt, and Bach, 2012)
  - SVRG (Johnson and Zhang, 2013; Zhang et al., 2013)
  - SAGA (Defazio, Bach, and Lacoste-Julien, 2014), etc...

- **Running-time to reach precision** ε (with κ = condition number)

Stochastic gradient descent	$d \times \quad \kappa \quad \times \frac{1}{\varepsilon}$
Gradient descent	$d \times \quad n\kappa \quad \times \log \frac{1}{\varepsilon}$
Variance reduction	$d \times \quad (n+\kappa) \quad \times \log \frac{1}{\varepsilon}$

Recent progress in single machine optimization

- **Variance reduction**
  - Exponential convergence with $O(d)$ iteration cost
  - SAG (Le Roux, Schmidt, and Bach, 2012)
  - SVRG (Johnson and Zhang, 2013; Zhang et al., 2013)
  - SAGA (Defazio, Bach, and Lacoste-Julien, 2014), etc...

- **Running-time to reach precision ε** (with $\kappa =$ condition number)

Stochastic gradient descent	$d\times$	κ	$\times$	$\frac{1}{\varepsilon}$
Gradient descent	$d\times$	$n\kappa$	$\times \log$	$\frac{1}{\varepsilon}$
Variance reduction	$d\times$	$(n+\kappa)$	$\times \log$	$\frac{1}{\varepsilon}$

  - Can be accelerated (e.g., Lan, 2015): $n+\kappa \Rightarrow n+\sqrt{n\kappa}$
  - Matching upper and lower bounds of complexity

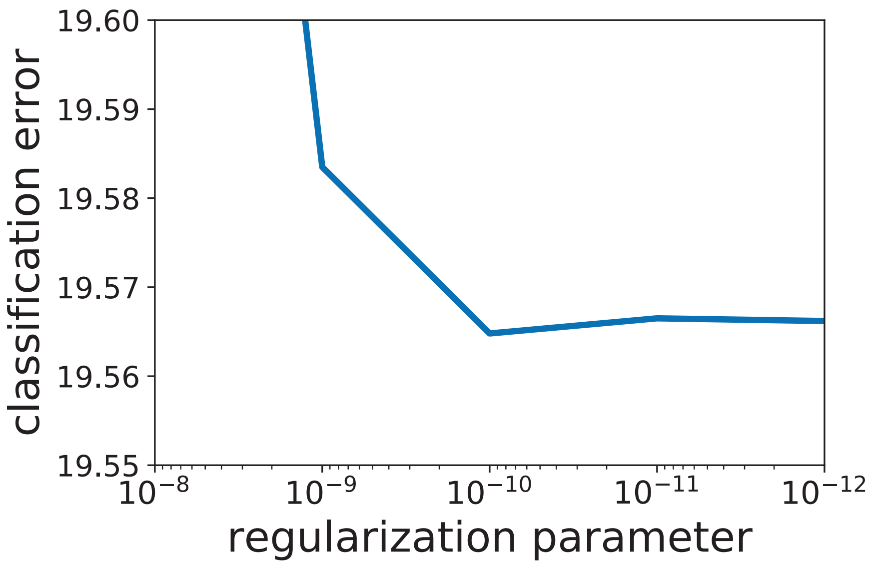
First-order methods are great!

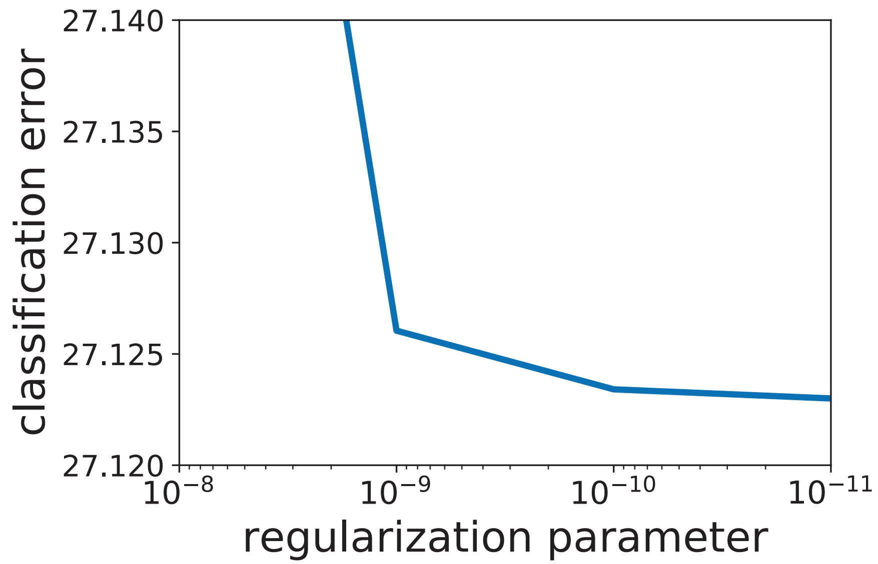
- **But...**
  - What if the condition number is huge?

First-order methods are great!

- **But...**
  - What if the condition number is huge?

- **Test errors: Logistic regression with Gaussian kernels**
  - Left: *Susy* dataset ($n = 5 \times 10^6$, $d = 18$)
  - Right: *Higgs* dataset ($n = 1.1 \times 10^7$, $d = 28$)





What about second-order methods?

- **Using the Hessian of** g
  - Newton method: $\theta_t = \theta_{t-1} - \nabla^2 g(\theta_{t-1})^{-1} \nabla g(\theta_{t-1})$
  - Local quadratic convergence: need $O(\log \log \frac{1}{\varepsilon})$ iterations

What about second-order methods?

- **Three classical reasons for discarding them in machine learning**
  1. Only useful for high precision, but ML only requires low precision
  2. Computing the Newton step is too expensive
  3. No global convergence for many ML problems

What about second-order methods?

- **Three classical reasons for discarding them in machine learning**
  1. Only useful for high precision, but ML only requires low precision
  2. Computing the Newton step is too expensive
  3. No global convergence for many ML problems

- **Three solutions**
  1. Even a low-precision solution requires second-order schemes
  2. Approximate linear system solvers
  3. Novel globally convergent second-order method

What about second-order methods?

- **Three classical reasons for discarding them in machine learning**
  1. Only useful for high precision, but ML only requires low precision
  2. Computing the Newton step is too expensive
  3. No global convergence for many ML problems

- **Three solutions**
  1. Even a low-precision solution requires second-order schemes
  2. Approximate linear system solvers
  3. Novel globally convergent second-order method

- **Globally Convergent Newton Methods for Ill-conditioned Generalized Self-concordant Losses**
  - Marteau-Ferey, Bach, and Rudi (2019a)

Generalized self-concordance

$$\min_{\theta \in \mathbb{R}^d} \quad g_\lambda(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \frac{\lambda}{2} \|\theta\|^2$$

- **Regular self-concordance** (Nemirovskii and Nesterov, 1994)
  - One dimension: for all t, $|\varphi^{(3)}(t)| \leqslant 2(\varphi''(t))^{3/2}$
  - Affine invariance
  - Few instances in machine learning
  - See Pilanci and Wainwright (2017)

Generalized self-concordance

$$\min_{\theta \in \mathbb{R}^d} \quad g_\lambda(\theta) = \frac{1}{n} \sum_{i=1}^n \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \frac{\lambda}{2} \|\theta\|^2$$

- **Regular self-concordance** (Nemirovskii and Nesterov, 1994)
  - One dimension: for all t, $|\varphi^{(3)}(t)| \leqslant 2(\varphi''(t))^{3/2}$
  - Affine invariance
  - Few instances in machine learning
  - See Pilanci and Wainwright (2017)

- **Generalized self-concordance** (Bach, 2010, 2014)
  - One dimension: for all t, $|\varphi^{(3)}(t)| \leqslant C\varphi''(t)$
  - No affine invariance
  - Applies to logistic regression and beyond

Generalized self-concordance

- **Examples**
  - Logistic regression: $\log(1+\exp(-y_i\Phi(x_i)^\top\theta))$
  - Softmax regression: $\log\big(\sum_{j=1}^k \exp(\theta_j^\top\Phi(x_i))\big) - \theta_{y_i}^\top\Phi(x_i)$
  - Generalized linear models with bounded features, including conditional random fields (Sutton and McCallum, 2012)
  - Robust regression: $\varphi(y_i - \Phi(x_i)^\top\theta)$ with $\varphi(u) = \log(e^u + e^{-u})$

Generalized self-concordance

- **Examples**
  - Logistic regression: $\log(1 + \exp(-y_i \Phi(x_i)^\top \theta))$
  - Softmax regression: $\log\big(\sum_{j=1}^k \exp(\theta_j^\top \Phi(x_i))\big) - \theta_{y_i}^\top \Phi(x_i)$
  - Generalized linear models with bounded features, including conditional random fields (Sutton and McCallum, 2012)
  - Robust regression: $\varphi(y_i - \Phi(x_i)^\top \theta)$ with $\varphi(u) = \log(e^u + e^{-u})$

- **Statistical analysis**
  - Non-asymptotic locally quadratic analysis
  - Finite dimension: Ostrovskii and Bach (2018)
  - Kernels: Marteau-Ferey, Ostrovskii, Bach, and Rudi (2019b)

Newton method for self-concordant functions

$$\min_{\theta \in \mathbb{R}^d} \quad g_\lambda(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \frac{\lambda}{2} \|\theta\|^2$$

- **Newton step:** $\theta^{\mathrm{Newton}} = \theta - \nabla^2 g_\lambda(\theta)^{-1} \nabla g_\lambda(\theta)$

Newton method for self-concordant functions

$$\min_{\theta \in \mathbb{R}^d} \quad g_\lambda(\theta) = \frac{1}{n} \sum_{i=1}^n \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \frac{\lambda}{2} \|\theta\|^2$$

- **Newton step:** $\theta^{\text{Newton}} = \theta - \nabla^2 g_\lambda(\theta)^{-1} \nabla g_\lambda(\theta)$
  - Approximation by $\hat{\theta}$ with appropriate norm:

$$\big(\theta^{\text{Newton}} - \hat{\theta}\big)^\top \nabla^2 g_\lambda(\theta) \big(\theta^{\text{Newton}} - \hat{\theta}\big) \leqslant \rho^2 \nabla g_\lambda(\theta)^\top \nabla^2 g_\lambda(\theta)^{-1} \nabla g_\lambda(\theta)$$

Newton method for self-concordant functions

$$\min_{\theta\in\mathbb{R}^d} \quad g_\lambda(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell\big(y_i, \theta^\top\Phi(x_i)\big) + \frac{\lambda}{2}\|\theta\|^2$$

- **Newton step:** $\theta^{\rm Newton} = \theta - \nabla^2 g_\lambda(\theta)^{-1}\nabla g_\lambda(\theta)$
  - Approximation by $\hat\theta$ with appropriate norm:

$$\big(\theta^{\rm Newton} - \hat\theta\big)^\top \nabla^2 g_\lambda(\theta)\big(\theta^{\rm Newton} - \hat\theta\big) \leqslant \rho^2 \nabla g_\lambda(\theta)^\top \nabla^2 g_\lambda(\theta)^{-1}\nabla g_\lambda(\theta)$$

  - Local convergence: if $\rho \leqslant \frac{1}{7}$ and $\nabla g_\lambda(\theta_0)^\top \nabla^2 g_\lambda(\theta_0)^{-1}\nabla g_\lambda(\theta_0) \leqslant \frac{\lambda}{R^2}$

$$g_\lambda(\theta_t) - \inf_{\theta\in\mathbb{R}^d} g_\lambda(\theta) \leqslant 2^{-t}$$

  - Linear convergence with no dependence on condition number

Globalization scheme

$$\min_{\theta \in \mathbb{R}^d} \quad g_\lambda(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \frac{\lambda}{2}\|\theta\|^2$$

- **Start with large** $\lambda = \lambda_0$
  - Reduce it geometrically until desired λ
  - Minimize g_λ approximately with approximate Newton steps

Globalization scheme

$$\min_{\theta \in \mathbb{R}^d} \quad g_\lambda(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \frac{\lambda}{2} \|\theta\|^2$$

- **Start with large $\lambda = \lambda_0$**
  - Reduce it geometrically until desired λ
  - Minimize g_λ approximately with approximate Newton steps
- **Rate of convergence**
  - reach precision ε after $\Omega(\log \frac{\lambda_0}{\lambda} + \log \frac{1}{\varepsilon})$ Newton steps

Approximate Newton steps

$$\min_{\theta \in \mathbb{R}^d} \quad g_\lambda(\theta) = \frac{1}{n}\sum_{i=1}^n \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \frac{\lambda}{2}\|\theta\|^2$$

- **Hessian**: $\nabla^2 g_\lambda(\theta) = \dfrac{1}{n}\displaystyle\sum_{i=1}^n \ell''\big(y_i, \theta^\top \Phi(x_i)\big)\Phi(x_i)\Phi(x_i)^\top + \lambda I$

Approximate Newton steps

$$\min_{\theta \in \mathbb{R}^d} \; g_\lambda(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \frac{\lambda}{2}\|\theta\|^2$$

- **Hessian**: $\nabla^2 g_\lambda(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell''\big(y_i, \theta^\top \Phi(x_i)\big)\Phi(x_i)\Phi(x_i)^\top + \lambda I$

- **Efficient Newton linear system** (Pilanci and Wainwright, 2017; Agarwal et al., 2017; Bollapragada et al., 2018; Roosta-Khorasani and Mahoney, 2019)
  - Hadamard transform (Boutsidis and Gittens, 2013)
  - Randomized sketching (Drineas et al., 2012)
  - Falkon: preconditioned Nyström method for kernel methods (Rudi, Carratino, and Rosasco, 2017)

Optimal predictions for kernel methods

$$\min_{\theta \in \mathbb{R}^d} \quad g_\lambda(\theta) = \frac{1}{n}\sum_{i=1}^n \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \frac{\lambda}{2}\|\theta\|^2$$

- **Nyström / Falkon method + globalization scheme**
  - Worst-case optimal regularization parameter $\lambda = 1/\sqrt{n}$
  - Optimal excess error $O(1/\sqrt{n})$.
  - $O(n)$ space and $O(n\sqrt{n})$ time

Optimal predictions for kernel methods

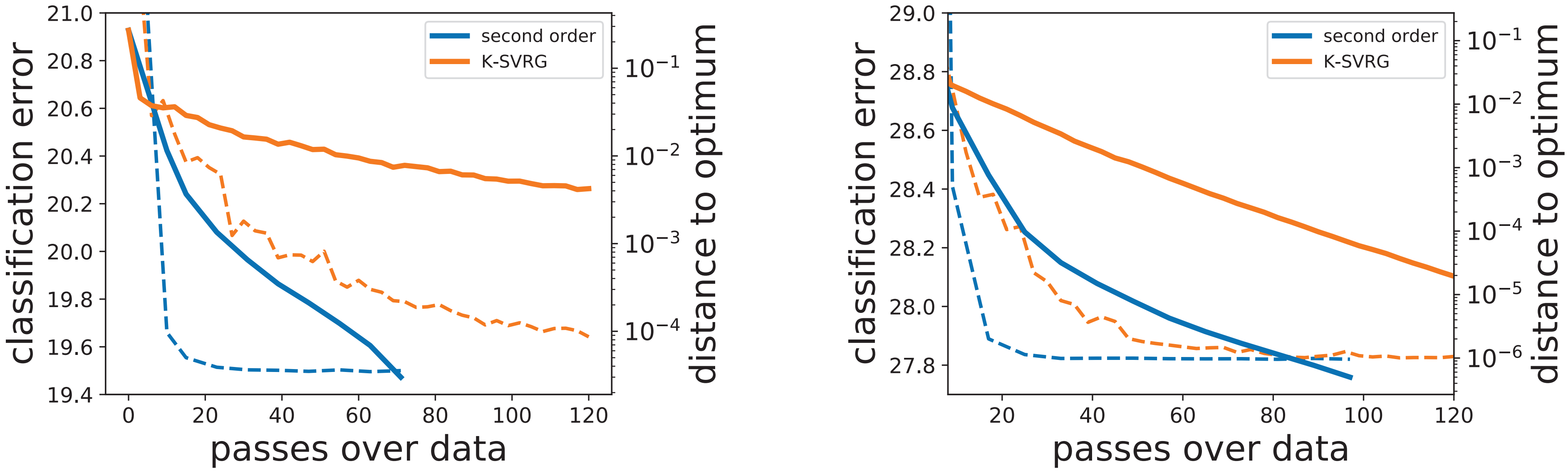
$$\min_{\theta \in \mathbb{R}^d} \quad g_\lambda(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \frac{\lambda}{2} \|\theta\|^2$$

- **Nyström / Falkon method + globalization scheme**
  - Worst-case optimal regularization parameter $\lambda = 1/\sqrt{n}$
  - Optimal excess error $O(1/\sqrt{n})$
  - $O(n)$ space and $O(n\sqrt{n})$ time

- **Extensions to more refined convergence bounds**
  - Source and capacity conditions
  - See Marteau-Ferey et al. (2019b,a)

Experiments

- Left: *Susy* dataset ($n = 5 \times 10^6$, $d = 18$)
- Right: *Higgs* dataset ($n = 1.1 \times 10^7$, $d = 28$)



Conclusions

- **Second order strikes back**

  1. Even a low-precision solution requires second-order schemes
  2. Approximate linear system solvers
  3. Novel globally convergent second-order method

Conclusions

- **Second order strikes back**
  1. Even a low-precision solution requires second-order schemes
  2. Approximate linear system solvers
  3. Novel globally convergent second-order method
- **Extensions**
  - Beyond Euclidean regularization
  - Beyond convex problems

References

Naman Agarwal, Brian Bullins, and Elad Hazan. Second-order stochastic optimization for machine learning in linear time. *J. Mach. Learn. Res.*, 18(1):4148–4187, January 2017.

F. Bach. Self-concordant analysis for logistic regression. *Electronic Journal of Statistics*, 4:384–414, 2010. ISSN 1935-7524.

Francis Bach. Adaptivity of averaged stochastic gradient descent to local strong convexity for logistic regression. *Journal of Machine Learning Research*, 15(1):595–627, 2014.

Raghu Bollapragada, Richard H. Byrd, and Jorge Nocedal. Exact and inexact subsampled newton methods for optimization. *IMA Journal of Numerical Analysis*, 39(2):545–578, 2018.

Christos Boutsidis and Alex Gittens. Improved matrix algorithms via the subsampled randomized hadamard transform. *SIAM Journal on Matrix Analysis and Applications*, 34(3):1301–1340, 2013.

Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems*, 2014.

Petros Drineas, Malik Magdon-Ismail, Michael W Mahoney, and David P Woodruff. Fast approximation of matrix coherence and statistical leverage. *Journal of Machine Learning Research*, 13(Dec):3475–3506, 2012.

Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, 2013.

G. Lan. An optimal randomized incremental gradient method. Technical Report 1507.02000, arXiv, 2015.

N. Le Roux, M. Schmidt, and F. Bach. A stochastic gradient method with an exponential convergence rate for strongly-convex optimization with finite training sets. In *Advances in Neural Information Processing Systems (NIPS)*, 2012.

Ulysse Marteau-Ferey, Francis Bach, and Alessandro Rudi. Globally convergent newton methods for ill-conditioned generalized self-concordant losses. In *Advances in Neural Information Processing Systems*, pages 7636–7646, 2019a.

Ulysse Marteau-Ferey, Dmitrii Ostrovskii, Francis Bach, and Alessandro Rudi. Beyond least-squares: Fast rates for regularized empirical risk minimization through self-concordance. In *Proceedings of the Conference on Computational Learning Theory*, 2019b.

Arkadii Nemirovskii and Yurii Nesterov. Interior-point polynomial algorithms in convex programming. *Society for Industrial and Applied Mathematics*, 1994.

Y. Nesterov. A method for solving a convex programming problem with rate of convergence $O(1/k^2)$. *Soviet Math. Doklady*, 269(3):543–547, 1983.

Dmitrii Ostrovskii and Francis Bach. Finite-sample analysis of m-estimators using self-concordance. *arXiv preprint arXiv:1810.06838*, 2018.

Mert Pilanci and Martin J. Wainwright. Newton sketch: A near linear-time optimization algorithm with linear-quadratic convergence. *SIAM Journal on Optimization*, 27(1):205–245, 2017.

Farbod Roosta-Khorasani and Michael W. Mahoney. Sub-sampled Newton methods. *Math. Program.*, 174(1-2):293–326, 2019.

Alessandro Rudi, Luigi Carratino, and Lorenzo Rosasco. Falkon: An optimal large scale kernel method. In *Advances in Neural Information Processing Systems*, pages 3888–3898, 2017.

Charles Sutton and Andrew McCallum. An introduction to conditional random fields. *Foundations and Trends® in Machine Learning*, 4(4):267–373, 2012.

L. Zhang, M. Mahdavi, and R. Jin. Linear convergence with condition number independent access of full gradients. In *Advances in Neural Information Processing Systems*, 2013.