CSC 588: Machine learning theory **Spring 2021**

Lecture 23: Kernel Methods. Online Newton step for exp-concave functions

Lecturer: Chicheng Zhang *Scribe: Robert Vacareanu*

1 Kernel Methods

Goal Remember that the goal is to find $w \in \mathbb{R}^N$ that approximately minimizes:

$$\frac{1}{m}\sum_{i=1}^{m} \ell(w, (\phi(x_i), y_i)) + \frac{\lambda}{2}||w||_2^2 = \frac{1}{m}\sum_{i=1}^{m} f(y_i\langle w, \phi(x_i)\rangle) + \frac{\lambda}{2}||w||_2^2 \tag{1}$$

Key Idea Instead of keeping track of $w_t \in \mathbb{R}^N$ explicitly, keep track of the coefficient of w_t, $\alpha_t \in \mathbb{R}^m$. Maintain invariant that $w_t = \sum_{i=1}^{m} \alpha_t(i)\phi(x_i)$

Algorithm 1 Original Stochastic Gradient Descent

```
Input: data $(x_i, y_i)_{i=1}^m$, initialize $w_1 = \vec{0} \in \Omega = \mathbb{R}^d$.
for $t = 1, 2, \cdots, T$ do
    sample $i_t$ in $Uniform(\{1..n\})$
    $f_t(w) = l(w, (x_{i_t}, y_{i_t})) + \frac{\lambda}{2}||w||_2^2$
    calculate $g_t$ using $v_t \in \partial l_t(w_t)$: $g_t = v_t + \lambda w_t$
    update: $w_{t+1} \leftarrow w_t - \frac{1}{\lambda t}(\lambda w_t + v_t)$
end for
```

Question Can we modify the algorithm such that instead of keeping track of the w_t's, we keep track of α_t's. We are going to utilize this special structure of the loss function to calculate the subgradient v_t for each round. To do this, recal that $l(w, (x_{i_t}, y_{i_t})) + \frac{\lambda}{2}||w||_2^2$ is $f(y_{i_t}\langle w, \phi(x_{i_t})\rangle) + \frac{\lambda}{2}||w||_2^2$ (which is $= f_t(w) = l_t(w) + \frac{\lambda}{2}||w||_2^2$) To calculate the gradient of this we will use the following fact:

Fact 1. *$f : \mathbb{R} \to \mathbb{R}$ convex, $h(x) = f(\langle a, x\rangle + b)$. Suppose $z \in \partial f(\langle a, x\rangle + b)$, then $za \in \partial f(w)$.*

Applying Fact 1, we will have to find $z_t \in \partial f(y_{i_t}\langle w_t, \phi(x_{i_t})\rangle)$. We will use the kernel trick (discussed last lecture) to calculate this efficiently. Also, we have v_t as $v_t = z_t y_{i_t}\phi(x_{i_t}) \in \partial l_t(w_t)$. Rewriting the recurrence for w_t yields:

$$\begin{aligned} w_{t+1} &= (1 - \frac{1}{t})w_t - \frac{1}{\lambda t} z_t y_{i_t}\phi(x_{i_t}) \\ &= (1 - \frac{1}{t})\sum_{i=1}^{m} \alpha_t(i)\phi(x_i) - \frac{1}{\lambda t} z_t y_{i_t}\phi(x_{i_t}) \sum_{i=1}^{m} \alpha_{t+1}(i)\phi(x_i) \end{aligned}$$

How to update the α_{t+1} based on α_t? We will compare coefficients in two cases: $i = i_t$ and $i \neq i_t$:

$$\alpha_{t+1}(i) = \begin{cases} (1 - \frac{1}{t})\alpha_t(i) - \frac{1}{\lambda t} z_t y_{i_t}, & i = i_t \\ (1 - \frac{1}{t})\alpha_t(i), & i \neq i_t \end{cases}$$

We get the following recurrence:

$$\alpha_{t+1} = -\frac{1}{\lambda t}\sum_{s=1}^{t} z_s y_{i_s} e_{i_s}$$

Where e_i denotes the canonical basis in $\mathbb{R}^m$ with 1 on the i-th position and 0 everywhere else.

Summary In summary, we developed a regularized loss minimization algorithm on the feature space with time complexity independent of the feature dimension N.

2 Online convex optimization of exp-concave functions

Informally, exp-concave functions are a family of concave function with desirable properties (they have curvature, but not as strong as with strongly-convex functions; quadratic lower-bound property only happens for some directions)

Definition f is called α-exp-concave if $exp(-\alpha f(a)$ is concave

Lemma 2. *If f is twice differentiable and α-exp-concave, then $\forall x$:*

$$\nabla^2 f(x) \succeq \alpha \nabla f(x) \cdot \nabla f(x)^T$$

Where $A \succeq B$ means that $A - B$ is positive semi definite

Fact 3. *If f is twice differentiable, then:*

$$\begin{aligned} f \textit{ convex} &\iff \nabla^2 f(x) \succeq 0 \\ f \textit{ concave} &\iff \nabla^2 f(x) \preceq 0 \end{aligned}$$

Furthermore, if f is λ-sc with respect to $||\cdot||_2 \iff \forall x \quad \nabla^2 f(x) \succeq \lambda \cdot I$ (Chicheng notes: this does not necessarily hold for other norms.)

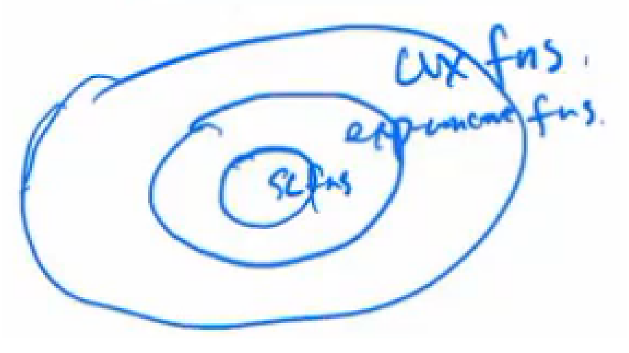

Figure 1: Venn diagram of the three types of convex functions that we encountered so far.

For two of the three families of convex functions we know upper bounds on the achievable regret ($\sqrt{T}$ for convex functions and $\frac{\ln(T)}{\lambda}$ for strongly convex functions). But what regret upper bounds are achievable for exp-concave funnctions?

The proof of the lemma uses the afore introduced fact on $g(w) = \exp(-\alpha f(w))$, which is concave. Then, we calculate the first-order and second order derivatives and use the fact that the Hessian is negative semi definite and use Lemma 2:

$$\begin{aligned}\nabla g(w) &= -\alpha \exp(-\alpha f(w)) \nabla f(w) \\ \nabla^2 g(w) &= \alpha^2 \exp(-\alpha f(w)) \nabla f(w) \cdot \nabla f(w)^T + (-\alpha) \exp(-\alpha f(w) \nabla^2 f(w)) \\ &= \exp(-\alpha f(w))(\alpha^2 \nabla f(w) \nabla f(w)^T - \alpha \nabla^2 f(w))\end{aligned}$$

2.0.1 Two examples

Portofolio selection

$$\begin{aligned}\Omega &= \triangle^{d-1} \\ r_t &= \frac{\text{unit price of asset at the end of the day } t}{\text{unit price of asset at the end of the day } t-1} \\ f_t(w) &= -\ln(\langle r_t, w\rangle) \\ R_T(w^*) &= \sum_{t=1}^{T} f_t(w_t) - \sum_{t=1}^{T} f_t(w^*)\end{aligned}$$

The first term of $R_T(w^*)$ is called *negative log wealth of the investor/learning algorithm* and the second term is called *constant rebalance portofolio* (CRP). The reason of why is CRP an interesting benchmark is summarized in the table below:

t	$r_t(1)$	$r_t(2)$
1	1	$\frac{1}{2}$
2	1	2
3	1	$\frac{1}{2}$
4	1	2
..	..	..

Table 1: Stock evolution for stock 1 ($r_t(1)$) and stock 2 ($r_t(2)$). Notice that the evolution of $r_t(2)$ is cyclic.

Assume $w^* = (1/2, 1/2)$ and initial weight 1. Because we have $\langle r_t, w\rangle$ equal to 3/4 when odd and 3/2 when even, we have:

$$\text{wealth} = (\frac{9}{8})^{\frac{T}{2}}$$

Regression example

$$\begin{aligned}\Omega &= \{w \in \mathbb{R}^d : ||w||_2 \leq B\} \\ f_t(w) &= \frac{1}{2}(\langle w, x_t\rangle - y_t)^2 \\ ||x_t||_2 &\leq R \\ |y_t| &\leq Y\end{aligned}$$

By using the Lemma 2 we can show that $f_t(w)$ is $\frac{1}{(RB+Y)^2}$-exp-concave (exercise).

2.1 Algorithms for online exp-concave optimization

We will reuse OMD framework, but we will not use one single distance generating function, but a family of distance generating function (adaptively generated on the fly).

Algorithm 2 Online Newton Step

```
  Assume: $\Omega$ with $\max_{u,v\in\Omega} ||u-v||_2 \leq B$, $\{f_t\}_{t=1}^T$ exp-concave and $L$ -Lipschitz
Require: $\lambda, \tilde{\alpha}$ (parameters of the algorithm)
  Initialize: $w_1 \leftarrow$ an arbitrary point in $\Omega$
  for $t = 1, 2, \cdots, T$ do
    show $w_t$
    receive $f_t$
    update $w_t$ as $w_{t+1} = \text{argmin} \langle w, g_t \rangle + D_{\psi_t}(w, w_t)$, where $g_t \in \partial f_t(w_t)$ and $\psi_t(w) = \frac{1}{2}||w||_{A_t}^2$ and
    $A_t = \lambda I + \tilde{\alpha} \sum_{s=1}^t g_s g_s^T$
  end for
```

Alternatively, the algorithm in 2 can also be viewed as a two step update (like in the beginner of this lecture). You can first do an unconstrained minimization, which is basically a quadratic minimization.

$$\begin{aligned}
w'_{t+1} &= w_t - A_t^{-1} g_t \\
w_{t+1} &= \underset{w\in\Omega}{\text{argmin}} ||w - w'_{t+1}||_{A_t}
\end{aligned}$$

2.1.1 Analysis of Online Newton Step

Theorem 4. *With the setting of $\lambda = \frac{1}{\tilde{\alpha} B^2}$, $\tilde{\alpha} = min(\frac{1}{8BL}, \frac{\alpha}{2})$, Online Newton Step gives:*

$$\begin{aligned}
R_T(\Omega) &\leq O(\frac{1}{\tilde{\alpha}} d \ln(T)) \\
&= O((\frac{1}{\alpha} + LB) d \ln(T))
\end{aligned}$$

Basically, Online Newton Step achieves a regret guarantee that is logarithmic in T, similar to strong convexity case, but we are paying an additional price: the dimensionality of the decision space. To prove the theorem we will need some properties of the exp-concave function. Especially, the following key lemma.

Lemma 5. *If f is α-exp-concave and L-Lipschitz, then $\forall u, v \in \Omega$:*

$$f(u) \geq f(v) + \langle \nabla f(v), u - v \rangle + \frac{\tilde{\alpha}}{2} \cdot (u-v)^T \nabla f(v) \nabla f(v)^T (u - v)$$

Remark 6. *Chicheng notes after the lecture: if f is nondifferentiable at v, then the above inequality is still true by replacing the above $\nabla f(v)$ with any $g_v \in \partial f(v)$. See Piazza for a link to the proof of this lemma.*

The proof of Theorem 4 also requires a "linearization" step similar to the first step in the regret analysis for OGD / OMD. To prove the theorem, we first upper-bound the instantaneous regret by taking advantage of Lemma 5:

$$f_t(w_t) - f_t(u) \leq \langle g_t, w_t - u \rangle - \frac{\tilde{\alpha}}{2} (w_t - u)^T g_t g_t^T (w_t - u)$$

The proof is delayed to next lecture.