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

Lecture 18: Online to batch conversion; Azuma's Inequality; online gradient descent

Lecturer: Chicheng Zhang *Scribe: Sheldon Deeny*

(Tuesday, March 23.)

1 Online optimization

Think of the interaction between learner and environment (i.e a 2-player game).

Our setting is:

- decision set Ω, or *action space* (often convex).

For $t = 1, 2, \ldots, T$:

- learner picks $w_t \in \Omega$
- environment picks loss function: $f_t : \Omega \mapsto \mathbb{R}$.
- learner suffers loss $f_t(w_t)$

The goal of the learner is to minimize the cumulative loss. A special case of online optimization is online *convex* optimization, where Ω is a convex set and the f_t are convex functions.

Key performance measure is regret. It is the difference between the cumulative loss of the learner, and the cumulative loss of some fixed action (or predictor) u:

$$R_T(u) := \sum_{t=1}^{T} f_t(w_t) - \sum_{t=1}^{T} f_t(u)$$
$$R_T(\Omega) := \max_{u \in \Omega} R_T(u)$$

Our goal for the learning algorithm is to achieve sublinear regret, e.g. we want $R_T(u)$ to be $o(T)$, or $R_T/T \to 0$. Achieving regret sublinear in time has important implications in statistical learning.

2 Online vs. statistical learning

Recall the basic set-up of statistical learning:

- Training samples chosen iid from distribution $\mathcal{D}$:

$$S = (z_1, \ldots, z_T) \overset{iid}{\sim} \mathcal{D}$$

- We have a learning algorithm which maps the samples to a predictor:

$$\hat{w} = \mathcal{A}(S)$$

- The set of predictors that the learning algorithm chooses from is Ω: some hypothesis class.
- Loss function: $\ell(w, z)$. Performance of predictor w on example z.

- Goal of statistical learning: generate an algorithm which outputs predictor with a low "population" loss, in the sense:
$$L_D(\hat{w}) = \mathbb{E}_{z\sim D}\ell(\hat{w}, z) = \text{some small amount}$$
- The *excess loss* is defined as $L_D(\hat{w}) - L_D(w^*)$, i.e. the difference in the loss of the output classifier and best classifier in the class,
$$w^* = \operatorname*{argmin}_{w\in\Omega} L_D(w).$$
We want this excess loss to be $o(1)$ i.e. $\to 0$ when $T \to \infty$.

2.1 Differences from statistical learning:

- Online learning doesn't necessarily make iid distribution assumption. The environment may potentially pick the loss function which can be adaptive to the previous choices of the learners action.
- Sometimes online learning algorithms are more computationally efficient because they use a "fast update rule."

Connection: online to batch conversion

- What this means is that a statistical learning task can be reduced to an online learning task.
- More specifically, given some online learning algorithm with good regret guarantees, it can be used to construct a statistical learning algorithm with a good excess loss guarantee.

How can this be achieved? How should we define the loss?

Algorithm:

- Inputs: $(z_1, \ldots, z_T) \overset{iid}{\sim} D$, online learning algorithm $\mathcal{A}$, action space Ω
- For $t = 1, \ldots, T$:
  $\mathcal{A}$ outputs w_t.
  $f_t(w) = \ell(w, z_t)$ is loss induced by example t.
- The learning algorithm generates $w_1, \ldots, w_T$, and outputs
$$\hat{w} = \frac{1}{T}\sum_{t=1}^{T} w_t$$

The regret guarantees of $\mathcal{A}$ on u:

$$\frac{1}{T}R_T(u) = \frac{1}{T}\sum_{t=1}^{T}\ell(w_t, z_t) - \frac{1}{T}\sum_{t=1}^{T}\ell(u, z_t) = o(1)$$

Key observation: (assume loss is unbounded) we have

$$\sum_{t=1}^{T}\ell(u, z_t) \to L_D(u) \text{ (the population loss)}$$

by Hoeffding's inequality. Also, each w_t only depends on the previous $t-1$ examples, which is independent from the new example z_t. This (heuristically) suggests that each the $\ell(w_t, z_t)$ is an unbiased estimator of the generalization loss $L_D(w_t)$. We can conclude that the average generalization losses of these iterates will be

competitive with the generalization loss of any other predictor (assume regret does not depend on u). This justifies the output $\hat{w}$ of the above algorithm.

Assuming $L_D(w)$ is convex (since $\ell(w, z)$ is convex for all z, and expectation is convex), can conclude

$$L_D(\hat{w}) \leq \frac{1}{T}\sum_{t=1}^{T} L_D(w_t) \leq L_D(u) + \frac{R_T(u)}{T} + \text{concentration factors (sublinear in } T).$$

Theorem 1. *Assume $\ell(w, z) \in [0, B]$ is convex in w, then with probability $1 - \delta$:*

$$L_D(\bar{w}) \leq L_D(w^*) + \frac{R_T(w^*)}{T} + 2B\sqrt{\frac{2\ln(4/\delta)}{T}}$$

for all $w^* \in \Omega$, where $\bar{w}$ is the average predictor.

Remark 2. *Chicheng notes after the lecture: there is another useful and simpler guarantee one can show on the expected loss of the average predictor $\bar{w}$:*

$$\mathbb{E}L_D(\bar{w}) \leq L_D(w^\star) + \frac{\mathbb{E}[R_T(w^\star)]}{T}. \tag{1}$$

On the left hand side, $\bar{w}$ is a random variable, as it depends on the random examples $z_1, \ldots, z_T$. To see this inequality, we recall that by the definition of regret,

$$\frac{1}{T}\sum_{t=1}^{T} \ell(w_t, z_t) - \frac{1}{T}\sum_{t=1}^{T} \ell(w^\star, z_t) = \frac{R_T(w^\star)}{T}$$

Now, taking expectations on both sides: by the law of iterated expectation, for every t, we have $\mathbb{E}\left[\ell(w_t, z_t)\right] = \mathbb{E}\left[\mathbb{E}\left[\ell(w_t, z_t) \mid w_t\right]\right] = \mathbb{E}L_D(w_t)$. Therefore, we have

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} L_D(w_t)\right] - L_D(w^\star) = \frac{\mathbb{E}\left[R_T(w^\star)\right]}{T}.$$

Equation (1) *now follows directly by Jensen's inequality.*

Proof. Use Hoeffding's inequality, which implies with probability $1 - \delta/2$:

$$\left|\frac{1}{T}\sum_{t=1}^{T} \ell(w^*, z_t) - L_D(w^*)\right| \leq B\sqrt{\frac{2\ln(4/\delta)}{T}} \tag{2}$$

On the other hand, we need to establish the concentration of the online losses to the "online population loss": $\frac{1}{T}\sum_{t=1}^{T} L_D(w_t)$. Need to look at

$$\frac{1}{T}\sum_{t=1}^{T} \ell(w_t, z_t) - \frac{1}{T}\sum_{t=1}^{T} L_D(w_t)$$

We don't yet have the tools to compare this quantity, because it is not necessarily the case that our random variables are iid: $\ell(w_t, z_t)$ may not be independent from $\ell(w_{t-1}, z_{t-1})$. But, as discussed above, w_t heuristically only depends on the previous examples, so the z_t still serves as a fresh example to w_t. Thus, in expectation, $\ell(w_t, z_t)$ will still concentrate around the generalization loss, $L_D(w_t)$. To bound these types of quantities, we use a new concentration inequality:

Theorem 3 (Azuma's inequality(a generalization of Hoeffding)). *Given random variables $X_1, \ldots, X_T \in [-B, B]$, where $B > 0$, and $\mathbb{E}[X_t] = 0$. Assume for all t,*

$$\mathbb{E}[X_t | X_1, \ldots, X_{t-1}] = 0$$

(i.e. $X_1, \ldots, X_T$ is a martingale difference sequence). Then with probability $1 - \delta$:

$$\left| \sum_{t-1}^{T} X_t \right| \leq B\sqrt{2T \ln 2/\delta}$$

(which is essentially the guarantee of Hoeffding's inequality).

Using Azuma, have $X_t = \ell(w_t, z_t) - L_D(w_t) \in [-B, B]$. This satisfies the martingale difference sequence property, since

$$\mathbb{E}[X_t \mid \text{all observations up to } t-1 \text{ and } w_t] = 0$$

Thus, with probability $1 - \delta/2$:

$$\left| \frac{1}{T} \sum_{t=1}^{T} \ell(w_t, z_t) - \frac{1}{T} \sum_{t=1}^{T} L_D(w_t) \right| \leq B\sqrt{\frac{2\ln(4/\delta)}{T}}$$

If the above and (1) happen simultaneously, use union bound and algebra to show upper bound of average predictor. □

Example 1. *(Gambling) Let $c_1, \ldots, d_T \overset{iid}{\sim} U(\pm 1)$ be Rademacher random variables. At each time t, can place a bet that $X_t \in [-B, B]$ (at each time step, have a fixed budget) depends on previous observations $c_1, \ldots, c_{t-1}$. Think of $\operatorname{sign}(X_t)$ as the side of the coin c_t being bet on, and $|X_t|$ is the amount of money being bet on c_t. The profit at round t is $c_t X_t = z_t$. Applying Azuma's inequality in this gambling setting, can guarantee with high probability*

$$\sum_{t=1}^{T} z_t \in [\pm B\sqrt{T}]$$

i.e. the cumulative profit of the gambler is in the interval with high probability, i.e. the gambler doesn't lose/gain too quickly.

Now we prove Azuma's inquality by applying the law of iterative expectation:

Proof. (Azuma's Inequality) Verify that the sum of $X_1, \ldots, X_T$ has zero mean:

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{T} X_t\right] &= \mathop{\mathbb{E}}_{X_1,\ldots,X_{T-1}}\left[\mathop{\mathbb{E}}_{X_T}\left[\sum_{t=1}^{T} X_t\right]\right] \\
&= \mathbb{E}\sum_{t=1}^{T-1} X_t \\
&\vdots \\
&= 0.
\end{aligned}$$

If we condition on the first $T-1$ random variable's, only the last term in the sum $\sum_{t=1}^{T} X_t$ is random, and that random variable has mean zero, so the inner expectation simplifies to $\sum_{t=1}^{T-1} X_t$ by the Martingale difference sequence property. Apply repeatedly to arrive at $\mathbb{E}X_1 = 0$.

Next, verify that $\sum_{t=1}^{T} X_t$ is σ^2-sub-Gaussian. Moment generating function: for every λ,

$$\mathbb{E}\left[e^{\lambda\sum_{t=1}^{T}X_t}\right] = \mathop{\mathbb{E}}_{X_1,\ldots,X_{T-1}}\left[\mathop{\mathbb{E}}_{X_T}\left[e^{\lambda\sum_{t=1}^{T-1}X_t}e^{\lambda X_T}\right]\right]$$

Because we condition ("integrate out") the first $T-1$ random variables, the $e^{\lambda\sum_{t=1}^{T-1}}$ term can be ignored when taking the expectation over X_T. Apply the inequality: if Z is supported on $[a,b]$, then

$$\mathbb{E}e^{\lambda Z} \leq e^{\frac{\lambda^2}{2}\frac{(b-a)^2}{4}}$$

(Chicheng notes: conditioned on any realizations of $X_1,\ldots,X_{T-1}$, the distributional law of X_t has mean zero and has range $[-B,B]$.) Taking $b=B$ and $a=-B$, continuing from above, we can repeat the process and arrive at:

$$\begin{aligned}\ldots &\leq \mathbb{E}\left[e^{\lambda\sum_{t=1}^{T-1}X_t}e^{\frac{\lambda^2}{2}B^2}\right]\\ &\leq \ldots\\ &\leq e^{\frac{\lambda^2}{2}TB^2}\end{aligned}$$

where T is the number of iterations. Since the sub-Gaussian random variables X_t have the exponential (i.e. "light-tailed") property, we can conclude the result of Azuma's inequality. □

In conclusion, we have shown that applying the above concentration inequalities, by letting the online learning algorithm receive the loss function induced by the iid samples and output the average predictor, this yields a statistical learning algorithm that has excess loss guarantee that depends on the regret guarantee of the corresponding online learning algorithm.

3 Algorithms for online learning/optimization

Given the online learning setup at the beginning of these notes, how can we design an algorithm to choose the actions w_t? Recall that up to time t, we have collected the previous $t-1$ predictors w and loss functions f. One idea is to take the greedy approach and just pick the best, i.e. $w_t = \text{argmin}_{w\in\Omega}\sum_{s=1}^{t-1}f_s(w)$ this is called "follow the leader".

How could we be efficient, in the sense that we build w_t from the previous $w_1,\ldots,w_{t-1}$?

3.1 Online gradient descent

Motivation: consider a loss function in terms of logistic regression, i.e.

$$\ell(w,(x,y)) = \ln\left(1+e^{-y\langle w,x\rangle}\right).$$

Given iid $(x_i,y_i)\overset{iid}{\sim}D$, using excess loss guarantee derived above, can we use online learning and online to batch conversion to develop algorithms that can output a predictor $\hat{w}$ with a small expected loss, $L_D(\hat{w})$? Can we make the procedure update as fast/efficient as possible? To discuss the idea of gradient descent, we review some basic convexity definitions and results...

3.2 Convex function basics

Recall: a function f is convex in domain Ω (which is a convex set), if, for all $x,y\in\Omega$, $\alpha\in(0,1)$,

$$f(\alpha x+(1-\alpha)y) \leq \alpha f(x)+(1-\alpha)f(y)$$

Also, for all $x\in\Omega$, we have $f(x)\in\mathbb{R}$.

Examples of convex functions:

- Affine functions: $f(w) = \langle a, w \rangle + b$.
- Norms: $\|w\|_2$
- Strongly convex (c.f. stability lecture): $f(w) = \frac{1}{2}\|w\|_2^2$ is 1-strongly convex, which implies convex

Basic properties of convex functions:

- f, g convex, $\alpha, \beta > 0$, $\implies$ $\alpha f + \beta g$ is also convex (check: straightforward application of definitions)
- f is convex implies $g(x) = f(Ax + b)$ is convex, where A is a matrix and b a vector.
- $f_1, \ldots, f_n$ convex $\implies$ $g(x) \max_i f_i(x)$ is convex, i.e. pointwise max of convex functions is convex.

By the 2nd-order Taylor expansion and convexity of f, we have $f(u) \geq f(w) + \langle \nabla f(w), u - w \rangle$.

Definition 4. *(Subgradient) If f is convex on Ω, then for "almost all" points $x \in \Omega$, we define the subgradient of f at x:*

$$\partial f(x) = \{g \ : \ \forall y \in \Omega, \ f(y) \geq f(x) + \langle g, y - x \rangle\} \neq \varnothing$$

When f is differentiable at x, there is just one choice of g:

$$\partial f(x) = \{\nabla f(x)\}$$

(because any other vector/chord which passes through a convex function f at x will violate the Taylor expansion inequality.)

Example 2. *Let $f(x) = |x|$. Then*

$$\partial f(x) = \begin{cases} \{1\}, & x > 0 \\ \{-1\}, & x < 0 \\ [-1, 1], & x = 0 \end{cases}$$

Fact 5. $x^* = \operatorname{argmin}_{x \in \Omega} f(x) \iff 0 \in \partial f(x^*)$

3.3 Online (sub)gradient descent algorithm

Initialize $w_1 \in \Omega$.
For $t = 1, 2, \ldots, T$:

- choose w_t
- Receive loss function f_t, suffer loss $f_t(w_t)$
- Set $g_t \in \partial f_t(w_t)$
- Update:
  $w'_{t+1} \leftarrow w_t - \eta g_t$
  $w_{t+1} \leftarrow \Pi_\Omega(w'_{t+1}) = \operatorname{argmin}_{w \in \Omega} \|w - w'_{t+1}\|_2$

Heuristic: imagine all f_t are fixed (for all t, $f_t \equiv f$), OGD walks in direction where f_t decreases fastest, where ∂f is introduced to account for non-differentiability.

Next time: we will analyze the regret performance of this online gradient descent algorithm.