Online to offline, constrained subgradient descent

Lecturer: Ofer Dekel
Scribe: Jinna Lei

1 Review

1.1 Doob martingale

\[ \forall i = 0, \ldots, m W_i = \mathbb{E}[F(u_1, \ldots, u_m)|u_1, \ldots, u_i] \]
\[ W_0 = \mathbb{E}[f(u_1, \ldots, u_m)] \]
\[ W_m = f(u_1, \ldots, u_m) \]
\[ |W_i - W_{i-1}| < c/m \]

1.2 Online learning algorithm

In the following, \( U \) is an update function.

**Algorithm 1** Online learning

Pick default \( h_0 \in H \)

for \( i = 1, \ldots, m \) do

\[ \text{receive } x_i \in X \]
\[ \text{predict } h_{t-1}(x_i) \]
\[ \text{receive } y_i \in Y \]
\[ \text{suffer loss } l(h_{t-1}(x_i, y_i)) \]
\[ \text{update } h_i \leftarrow U(h_{t-1}, (x_i, y_i)) \] (or, alternatively, \( h_i \leftarrow U(h_0, \{x_j, y_j\}_{j=0}^i) \)).

end for

Note that there are no explicit limitations on the initial function \( h_0 \), but the update function \( U \) encodes an implicit restriction on the subsequent \( h_i \). In addition, “memorizing answers” is not a valid strategy, since this algorithm incurs loss based on the new sample in the next iteration.

1.3 Guarantee on cumulative loss

Let \( Q \) be a uniform distribution on \( h_0, \ldots, h_m \) and \( \ell \) a loss function with range in \([0, c]\). With probability at least \( 1 - \delta \) over \( S \sim \mathcal{D}^m \) for any update strategy \( U \),

\[ \ell(Q; \mathcal{D}) \leq \frac{1}{m} \sum_{i=1}^m \ell(h_{i-1}; (x_i, y_i)) + c \sqrt{\frac{\log(1/\delta)}{2m}} \]

We want two things of our learning algorithm: for the cumulative loss \( \sum_{i=1}^m \ell(h_{i-1}; (x_i, y_i)) \) to grow as \( O(\sqrt{m}) \), and the excess risk \( \ell(h; \mathcal{D}) - \ell(h^*; \mathcal{D}) \) to go to 0. Note that the latter condition is not a constraint on \( \ell(h; \mathcal{D}) \) itself; it only bounds the difference between our hypothesis and the best hypothesis in hindsight.
2 Online learning to offline learning: constrained subgradient descent

2.1 Subgradients

Definition 1 (subgradient). Let \( f \) be a convex function with domain \( \mathbb{R}^n \). Let \( w \in \mathbb{R}^n \). The subgradient of \( f \) at \( w \) is a vector \( v \) such that \( \forall w' \in \mathbb{R}^n 

\begin{align*}
    f(w') - f(w) &\geq \langle v, w' - w \rangle, \\
    \text{or equivalently, } f(w') &\geq f(w) + \langle v, w' - w \rangle.
\end{align*}

We will denote the subgradient of \( f \) at \( w \) by \( \nabla f(w) \).

If \( f \) is differentiable at \( w \), then the gradient is the only subgradient.

2.1.1 Example: Hinge loss

Notation: \([z]_+ = \max(z, 0)\).

Claim 2.

\[
    \nabla_w [1 - y\langle w, x \rangle]_+ = \begin{cases} 
    0 & y\langle w, x \rangle \geq 1 \\
    -yx & y\langle w, x \rangle < 1 
\end{cases}
\]

Proof. Trivial if \( y\langle w, x \rangle \geq 1 \), so assume \( y\langle w, x \rangle < 1 \).

\[
    [1 - y\langle w', x \rangle]_+ - [1 - y\langle w, x \rangle]_+ \\
    \geq (1 - y\langle w', x \rangle) - (1 - y\langle w, x \rangle) \\
    = (y - y\langle w', x \rangle) - (x - y\langle w, x \rangle) \\
    \geq \langle -yx, w' - w \rangle
\]

2.1.2 Example: Log loss

\[
    \nabla \log(1 + e^{-y\langle w, x \rangle}) = \frac{1}{1 + e^{-y\langle w, x \rangle}} (-yx)
\]

2.2 Subgradient descent algorithm

This is our general online algorithm, with the update strategy \( U \) explicitly specified as the subgradient and projection steps.

Definition 3 (Online regret). The online regret of an online algorithm \( \mathcal{A} \) is

\[
    \sum_{i=1}^m \ell(h_{i-1}; (x_i, y_i)) - \min_{h \in H} \sum_{i=1}^m \ell(h; (x_i, y_i)),
\]

or, intuitively, the cumulative loss of \( \mathcal{A} \) compared to the cumulative loss of the best fixed hypothesis in hindsight.
Algorithm 2 Subgradient descent (GD)

Init \( w_1 = 0 \)
for \( i = 1, \ldots, m \) do
receive \( x \in \mathbb{R}^n \)
predict \( \langle w_{i-1}, x_i \rangle \)
receive \( y \in \mathbb{R}^n \)
suffer loss \( \ell(y; w, x) \)
\( w'_i \leftarrow w_{i-1} - \eta \nabla \ell(w_{i-1}) \) (subgradient step)
\( w_i \leftarrow \min(1, \frac{B}{||w'_{i-1}||}) w'_i \) (projection step)
end for

Regret is the online equivalent of excess risk.

**Theorem 4.** The regret of GD \( \leq \eta = \frac{B}{\sqrt{\lambda X}} \), where \( ||w|| \leq B \), \( \ell \) is \( \lambda \)-Lipschitz, and \( ||x|| \leq X \).

**Proof.** Let \( H \) be the ball of radius \( B \). Choose \( w^* \in H \) arbitrarily. Define: \( \alpha_i := \beta_i + \gamma_i \), where
\[
\beta_i := \frac{1}{2} ||w_{i-1} - w^*||^2 - \frac{1}{2} ||w'_{i-1} - w^*||^2,
\]
\[
\gamma_i := \frac{1}{2} ||w'_{i-1} - w^*||^2 - \frac{1}{2} ||w_i - w^*||^2.
\]

**Lemma 5.** \( \gamma_i \geq 0 \)

**Proof.** (Intuitively, projection onto a convex set brings you closer to any point in the convex set.)
Case 1: \( ||w'_{i-1}|| \leq B \Rightarrow w_i = w'_{i-1} \Rightarrow \gamma_i = 0 \).
Case 2: \( ||w'_{i-1}|| > B \Rightarrow \gamma_i = \frac{B^2}{||w'_{i-1}||} = \frac{B^2}{||w_{i-1}||} \Rightarrow \)
\[
\gamma_i = \frac{1}{2} ||w'_{i-1}||^2 + \frac{1}{2} ||w^*||^2 - \langle w'_{i-1}, w^* \rangle - \frac{1}{2} ||w_i||^2 - \frac{1}{2} ||w^*||^2 + \langle w_i, w^* \rangle
\]
\[
= \frac{1}{2} ||w'_{i-1}||^2 - \frac{1}{2} B^2 - (1 - \frac{B}{||w'_{i-1}||}) \langle w'_{i-1}, w^* \rangle
\]
\[
\geq \frac{1}{2} ||w'_{i-1}||^2 - \frac{1}{2} B^2 - (1 - \frac{B}{||w'_{i-1}||}) ||w'_{i-1}|| ||w^*||
\]
\[
\geq \frac{1}{2} ||w'_{i-1}||^2 - \frac{1}{2} B^2 - (1 - \frac{B}{||w'_{i-1}||}) ||w'_{i-1}|| B
\]
\[
= \frac{1}{2} ||w'_{i-1}||^2 + \frac{1}{2} B^2 - ||w_{i-1}|| B
\]
\[
= \frac{1}{2} (||w'_{i-1}|| - B)^2
\]
\[
\geq 0
\]

**Lemma 6.**
\[
\beta_i \geq -\frac{\eta^2 \lambda^2 X^2}{2} + \eta (\ell(w_{i-1}; (x_i, y_i)) - \ell(w^*; (x_i, y_i))).
\]

3
Proof. By the definition of \( w'_{i-1} \),
\[
\frac{1}{2} ||w'_{i-1} - w^\ast|| = \frac{1}{2} ||w_{i-1} - w^\ast - \eta \nabla \ell(w_{i-1})||^2.
\]

Thus,
\[
\beta_i = \frac{1}{2} ||w_{i-1} - w^\ast||^2 - \frac{1}{2} ||w'_{i-1} - w^\ast||^2
\]
\[
= \frac{1}{2} ||w_{i-1} - w^\ast||^2 - \frac{1}{2} ||w_{i-1} - w^\ast||^2 - \frac{\eta^2}{2} ||\nabla \ell(w_{i-1})||^2 + \eta \langle w_{i-1} - w^\ast, \nabla \ell(w_{i-1}) \rangle
\]
\[
\geq - \frac{\eta^2}{2} \lambda^2 X^2 + \eta (\ell(w_{i-1}; (x_i, y_i)) - \ell(w^\ast; (x_i, y_i))),
\]
where the last inequality is by the \( \lambda \)-Lipschitz condition and the definition of subgradient.

Putting it all together:
\[
\sum_{i=1}^{m} \alpha_i = \sum_{i=1}^{m} \beta_i + \gamma_i
\]
\[
\leq \sum_{i=1}^{m} \beta_i
\]
\[
\leq \frac{1}{2} m \eta^2 \lambda^2 X^2 + \eta \sum_{i=1}^{m} \ell(w_{i-1}; (x_i, y_i)) - \ell(w^\ast; (x_i, y_i))).
\]

The first equality is from Lemma 5 and the second from Lemma 6. Now we use \( \eta = \frac{B}{\sqrt{m} \lambda X} \) to get
\[
- \frac{1}{2} m \eta^2 \lambda^2 X^2 + \eta \sum_{i=1}^{m} \ell(w_{i-1}; (x_i, y_i)) - \ell(w^\ast; (x_i, y_i)) \leq \frac{1}{2} B^2
\]
\[
\Rightarrow \text{regret} \leq \frac{B^2}{2 \eta} + \frac{1}{2} m \eta \lambda^2 X^2.
\]

To be continued...