CS272 - Theoretical Foundations of Learning, Decisions, and Games

# Lecture 26: Omni-prediction and Multi-objective Learning

December 4, 2025

*Lecturer: Brian Lee* *Readings: N/A*

*Scribe: Owen Burns*

## 1 Recap

Last lecture, omni-calibration was introduced and related to step calibration. Intuitively, omni-calibration computes a probability forecaster that can be used by any of a family of downstream losses to achieve good results. Formally, given features $K$, labels $y = \{0, 1\}$, finite hypothesis class $\mathcal{H} = \{h : K \to [0, 1]\}$, and loss class $\mathcal{L}$, then $p^* : K \to [0, 1]$ is an $(\mathcal{L}, \mathcal{H}, \varepsilon)$-$\mathrm{omnipredictor}$ if:

$$\max_{l \in \mathcal{L}} |\mathop{\mathbb{E}}_{D}[l(y, \text{BR}_l(p^*(x))] - min_{h \in \mathcal{H}} \mathbb{E}[l(y, h(x))]| \leq \varepsilon \tag{1}$$

Where $\text{BR}_l(p^*(x)) = \arg\min_{v \in [0,1]} \mathbb{E}_{\bar{y} \sim \text{Ber}(p^*(x))}[l(\bar{y}, v)]$. We therefore define omni prediction error to match the above form and see:

$$\text{OmniErr}(p^*, \mathcal{L}, \mathcal{H}) \leq \varepsilon$$

Let us also re-state the formal definition of step calibration in this contextual setting. You last saw this definition as an upper bound on U-calibration in a non-contextual setting, where your goal was to learn a predictor $p$ that took no-inputs. In the following, we see an adjusted definition that takes into account the fact that predictors in omni-prediction sense are contextual: $p^* : K \to [0, 1]$ is $(\mathcal{H}, \varepsilon) - \mathrm{stepcalibrated}$ if:

$$\max_{v, \omega \in [0,1], h \in \mathcal{H}} |\mathbb{E}[\mathbb{1}\{p^*(x) \leq v\} \cdot \mathbb{1}\{h(x) \leq \omega\} \cdot (y - p^*(x))]| \leq \varepsilon \tag{2}$$

Similarly,

$$\text{StepCalErr}(p^*, \mathcal{H}) \leq \varepsilon$$

## 2 Main Theorem

In the last lecture, we claimed that

**Theorem 2.1.** *Suppose that the loss class we care about has bounded variation, i.e. that $\mathcal{L} \subseteq \mathcal{L}_{BV}$. Then,*

$$\textit{OmniErr}(p^*, \mathcal{L}, \mathcal{H}) \leq 8 \cdot \textit{StepCalErr}(p^*, \mathcal{H}) + 2\varepsilon$$

Today, we prove this theorem. From last lecture, we already have the following lemma:

**Lemma 2.2.** *Let the discrete derivative of* $l$ *be* $\Delta l(v) = l(1, v) - l(0, v)$. *Then, define DecOIErr*$(p^*, \mathcal{L})$ *and HypOIErr*$(p^*, \mathcal{L}, \mathcal{H})$ *as follows:*

$$\textit{DecOIErr}(p^*, \mathcal{L}) := \max_{l \in \mathcal{L}} |\,\mathbb{E}[\Delta l(\textit{BR}_l(p^*(x))) \cdot (y - p^*(x))]|$$

$$\textit{HypOIErr}(p^*, \mathcal{L}, \mathcal{H}) := \max_{l \in \mathcal{L}, h \in \mathcal{H}} |\,\mathbb{E}[\Delta l(h(x)) \cdot (y - p^*(x))]|$$

*With these definitions, the following is true:*

$$\textit{OmniErr}(p^*, \mathcal{L}, \mathcal{H}) \leq \textit{DecOIErr}(p^*, \mathcal{L}) + \textit{HypOIErr}(p^*, \mathcal{L}, \mathcal{H})$$

This lemma helps us transition between the form of (1), which is the subtraction of two terms, to the form of (2) which is a single multiplied term. The key intuition here is that, even though we still have two terms, by applying the discrete derivative we end up with two terms that are each closer in form to (2).

Now, to prove Thm 2.1, we simply need to prove that DecOIErr and HypOI err added together are upper bounded by $8 \cdot \text{StepCalErr}(p^*, \mathcal{H}) + 2\varepsilon$. We will do this by performing a basis decomposition of the discrete derivatives of the losses we're considering, establishing a relationship between them and the step functions StepCalErr is concerned with.

First, we formally define the aforementioned $\mathcal{L}_{\text{BV}}$ as follows:

$$\mathcal{L}_{\text{BV}} = \{l : v(\Delta l) \leq 2\} \tag{3}$$

Where variation is defined as $v(\Delta l) = \sup_{n \in \mathbb{N}} \sup_{p_0 < \cdots < p_{n+1}} \sum_{i=1}^{n+1} |\Delta l(p_i) - \Delta l(p_{i-1})|$. In intuitive terms, the variation of the derivative of the loss varies a finite amount between $0$ and $n+1$ looking at all possible $n$.

Then, we define an approximate basis as:

**Approximate Basis** A set of functions $B \subseteq \{B : [0, 1] \to [-1, 1]\}$ is an $\varepsilon -$ approximate basis for $\Delta\mathcal{L}_{\text{BV}}$ with sparsity $s \in \mathbb{N}$, coefficient norm $C > 0$ if:

$$\exists B_1, ..., B_s \in B, \exists c_1, ..., c_s \in \mathbb{R}$$

such that $\forall \Delta l \in \Delta\mathcal{L}_{\text{BV}}$,

1. $\sup_{p \in (0,1]} |\Delta l(p) - \sum_{i=1}^{s} c_i B_i(p)| \leq \varepsilon$
2. $\sum_{i=1}^{s} |c_i| \leq c$

Intuitively, an approximate basis is where you can approximate any function in the class it is an approximate basis of up to an error $\varepsilon$, and that you can do this without taking unbounded norms.

We assert that the following proposition.

**Proposition 2.3.** *The class of all step functions, which we'll write as Step*$\{\mathbb{1}\{p \leq v\} : v \in [0,1]\}$*, is a* $\varepsilon -$ *approx basis for* $\Delta\mathcal{L}_{BV}$ *with* $s = O(\frac{1}{\varepsilon}),\ c \leq 4$

To prove Prop 2.3, first fix any $\Delta l \in \mathcal{L}_{\text{BV}},\ \varepsilon \in [0,1]$. Then, construct the following sequence $p_0 < \cdots < p_{n+1}$ inductively.

$$p'_0 = 1,\ p'_{i+1} = \sup\{p' < p'_i : |\Delta l(p'_i) - \Delta l(p')|\},\ p'_{n+1} = 1$$

Effectively, we are taking the unit interval and working backwards from 1, defining intervals at each point where the variation in $\Delta l$ since the last point is at least $\varepsilon$. Then we re-index the intervals to remove the $'$, such that $p_0 = 0$ and $p_{n+1} = 1$.

Clearly, $n\varepsilon \leq v(\Delta l) \leq 2$, which gives us $n \leq \frac{2}{\varepsilon}$ by the definition of $\mathcal{L}_{\text{BV}}$ and the definition of our $n$ intervals.

Then, define the following approximation:

$$\Delta\hat{l}(p) = \sum_{i=0}^{n} \Delta l(p_{i+1}) \cdot \mathbb{1}\{p_i < p \leq p_{i+1}\}$$

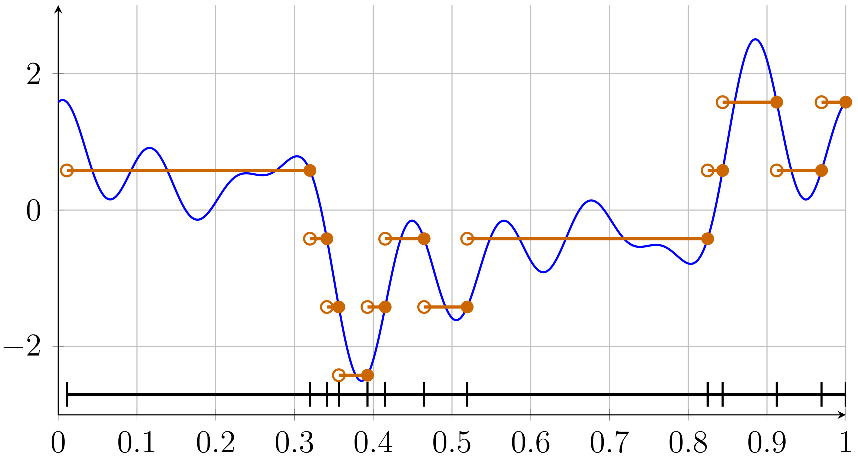

Figure 1: Right-endpoint step approximation and interval partition.

This is a piecewise linear approximation of $\Delta l$ such that within each interval the value of the approximation is the value of $\Delta l$ on the right hand side of the interval. Figure 1 gives a visualization of what this might look like for some loss function in $\mathcal{L}_{\text{BV}}$. This guarantees that the error within each interval will be at most $\varepsilon$. Formally, $||\Delta l - \Delta\hat{l}||_\infty \leq \varepsilon$. We can rewrite our approximation as

$$\Delta\hat{l}(p) = \Delta l(p_1)\mathbb{1}\{p \leq p_0\} + \sum_{i=1}^{n}(\Delta l(p_{i+1}) - \Delta l(p_i))\mathbb{1}\{p \leq p_i\} + \Delta l(p_{i+1})\mathbb{1}\{p \leq p_i\}$$

Crucially, by upper bounding the indicators by 1 and the $\Delta l$'s by the absolute value of the function, we get

$$\Delta\hat{l}(p) \leq 2 + v(\Delta l) \leq 4$$

In summary, because the number of intervals we have is n, we know that the number of step functions needed to approximate any $\Delta l$ is $\frac{2}{\varepsilon}$, which establishes the sparsity of Prop 2.3, and we know that the coefficient norm is at most $4$. Thus we have proven Prop 2.3.

We now assert another proposition:

**Proposition 2.4.**

$$\textit{DecOIErr}(p^*, \mathcal{L}) \leq 4 \cdot \textit{StepCalErr}(p^*, \mathcal{H}) + \varepsilon$$

To begin, note that the term $\Delta l(\text{BR}_l(p^*(x))$ in the definition of DecOIErr in Lemma 2.2 is a proper scoring rule, which means that it falls into $\mathcal{L}_{\text{BV}}$

This lets us upper bound DecOIErr:

$$\leq \max_{\tilde{l} \in \mathcal{L}_{\text{BV}}} |\, \mathbb{E}[\Delta\tilde{l}(p^*(x)) \cdot (y - p^*(x))]|$$

We use our approximate basis to further upper bound the error

$$\leq |\, \mathbb{E}[\sum_{i=1}^{s} \tilde{c}_i \mathbb{1}\{p^*(x) \leq \tilde{v}_i\} \cdot (y - p^*(x))]| + \varepsilon$$

We now apply Cauchy-Schwarz to get

$$\leq (\sum_{i=1}^{s} |\tilde{c}_i|) \cdot (\max_{i \sim [s]} |\, \mathbb{E}[\mathbb{1}\{p^*(x) \leq \tilde{v}_i\} \cdot (y - p^*(x))]|) + \varepsilon$$

Since the s in brackets is conditioned on the max over $\mathcal{L}_{\text{BV}}$ a couple steps ago, we can only further upper bound by taking the max over all the steps in the unit interval

$$\leq 4 \max_{v,w \in [0,1],\ h \in \mathcal{H}} |\, \mathbb{E}[\mathbb{1}\{p^*(x) \leq v, h(x) \leq w\}(y - p^*(x))]| + \varepsilon$$

Which is step calibration

$$\leq 4\text{StepCalErr}(p^*, \mathcal{H}) + \varepsilon$$

Thus proving Prop 2.4.

We run an identical argument for HypOIErr$(p^*, \mathcal{L}, \mathcal{H})$ to find that it is similarly upper bounded by

$$\leq 4\text{StepCalErr}(p^*, \mathcal{H}) + \varepsilon$$

We are able to do this because if $B$ is a basis for $\Delta\mathcal{L}$, then $B \circ \mathcal{H}$ is a basis for $\Delta\mathcal{L} \circ \mathcal{H}$.

By combining these facts, we get our original statement of Theorem 2.1.

# 3 Omni-prediction through Multi-objective Learning

By the proof of Theorem 2.1, we see that to create an algorithm for omni calibration, we only need to create an algorithm for step calibration. In this section, we briefly state the form of this algorithm.

To begin, rewrite step calibration error as

$$\max_{\sigma\in\{\pm1\},v,w,h} \mathbb{E}[\sigma \cdot \mathbb{1}\{p^*(x) \leq v, h(x) \leq w\} \cdot (y - p^*(x))]$$

And state that each step calibration objective can be parameterized by

$$\theta = (\sigma, v, w, h) \in \{\pm 1\} \times I_\varepsilon \times I_\varepsilon \times \mathcal{H} = \Theta$$

Where $I_\varepsilon$ is a discretization of the unit interval. This means that $|\Theta| = O(\frac{|\mathcal{H}|}{\varepsilon^2})$. Thus, we are attempting to approximate a zero-sum games with losses

$$l_{\sigma,v,w,h}(p^*, (x, y)) = \frac{1}{2} + \frac{1}{2} \cdot \sigma \cdot \mathbb{1}\{p^*(x) \leq v, h(x) \leq w\} \cdot (y - p^*(x))$$

This leaves us with the game that we're trying to solve, which is

$$\max_{\theta\in\Theta} \mathbb{E}[l_\theta(p^*, (x, y))] \leq \min_{p\in\{p:\lambda\to[0,1]\}} \max_{\theta\in\Theta} \mathbb{E}[l_\theta(p, (x, y))] + \varepsilon$$

Clearly, this structure is a 2-player game where we have a maximizing adversary and a minimizing player. Unlike in multi-distribution learning, however, the maximizing adversary is now picking over parameters of an objective rather than over distributions.

With this in mind, here is a naive omni-calibration algorithm based on the game we have constructed.

**Algorithm 1** Omni-Calibration

```
$p^{(1)} \leftarrow (\frac{1}{2})^X$
$q^{(1)} \leftarrow$ Unif$(\Theta)$
$t \leftarrow 0$
for $t = 1, \ldots, T$ do
    $o^{(t)} \sim q^{(t)}$
    $(x^{(t)}, y^{(t)}) \sim D$
    for $x \in X$ do
        $l_x^{(t)} \leftarrow \frac{1}{2} + \frac{1}{2}\sigma^{(t)} \cdot \mathbb{1}\{p^{(t)}(x) \leq v^{(t)}, h^{(t)}(x) \leq w^{(t)}\} \cdot \hat{y}$
        $p^{(t+1)} \leftarrow$ NoRegretAlg$(l_x^{(1)}, \ldots, l_x^{(t)})$
    end for
    $q^{(t+1)} \leftarrow$ NoRegretAlg$(l_{\theta^{(1)}}, \ldots, l_{\theta^{(t)}})$
end for
$p^* \leftarrow$ Unif$(p^{(1)}, \ldots, p^{(T)})$
return $p^*$
```

We choose T to be something like $T = O(\frac{\log(\frac{|\mathcal{H}|}{\varepsilon})}{\varepsilon^2})$. Running the algorithm for that many steps, we get our step calibrated predictor.