

**Prediction with Expert Advice:** predict binary outcomes  $y_t \in \{0, 1\}$ ,  $t \in [T]$  using experts  $E = \{e_1, \dots, e_N\}$  with predictions  $x_t(e_i)$ . **Loss:**  $\ell_t(\hat{y}_t) := \mathbb{1}[\hat{y}_t \neq y_t]$ , cumulative loss  $L_T = \sum_{t=1}^T \ell_t(\hat{y}_t)$ . **Regret:**  $R_T = \sum_{t=1}^T \ell_t(\hat{y}_t) - \min_{i \in [N]} \sum_{t=1}^T \ell_t(x_t(e_i))$ . **No-regret algorithms:**  $\lim_{T \rightarrow \infty} R_T/T = 0$ .

**Challenges:** best expert unknown a priori; algorithm must adapt sequentially. **Follow the Leader (FTL):** Simple idea, two steps. For each  $t$ , find the best expert  $i_{t-1}^*$  in hindsight till  $(t-1)$ :

$i_{t-1}^* = \arg \min_{i \in [N]} \sum_{\tau=1}^{t-1} \ell_\tau(x_\tau(e_i))$ , then predict  $x_t(e_{i_{t-1}^*})$ . FTL is a “natural” or greedy choice; can work okay in benign settings (few great experts), but switches too easily and may lead to unstable behavior. Adversary can exploit FTL; it is not no-regret.

**FTL Regret:** Assume existence of a perfect expert  $e^* \in E$  with  $\sum_{t=1}^T \ell_t(x_t(e^*)) = 0$ . Then, the regret of FTL is  $\text{Reg}(T) =$

$\sum_{t=1}^T \ell_t(x_t(i_{t-1}^*)) - \min_{i \in [N]} \sum_{t=1}^T \ell_t(x_t(e_i)) \leq N - 1$ . **Halving Algorithm (with Majority Vote):**

Maintain a set of experts  $E_t$ , with  $E_0 = E$ . Predict  $\hat{y}_t$  based on majority vote in  $E_t$ . If prediction is correct, continue; if incorrect, delete all experts in  $E_t$  who made a mistake. Each mistake eliminates at least half of the experts. Total number of mistakes  $\leq \log_2 N$ , and in the end we are left with the perfect expert  $e^*$ .

**Halving Algorithm Regret:** Assume existence of a perfect expert  $e^* \in E$  with  $\sum_{t=1}^T \ell_t(x_t(e^*)) = 0$ . Then, the regret of the Halving Algorithm is  $\text{Reg}(T) =$

$\sum_{t=1}^T \ell_t(\hat{y}_t) - \min_{i \in [N]} \sum_{t=1}^T \ell_t(x_t(e_i)) \leq \log_2 N$ . **Weighted Majority:** Remove the perfect-expert assumption. Let  $m = \min_{i \in [N]} \sum_{t=1}^T \ell_t(x_t(e_i))$  be the mistakes of the best expert. Maintain all experts  $E_t$ , predict  $\hat{y}_t$  by weighted majority vote. Experts that were wrong have their weights halved each round.

**Weighted Majority: Regret Bound:** General update:  $w_{t+1}(i) \leftarrow w_t(i)(1 - \epsilon)^{\mathbb{1}\{x_t(e_i) \neq y_t\}}$ . Mistake bound:  $\sum_{t=1}^T \ell_t(\hat{y}_t) \leq \frac{2 \ln N}{\epsilon} + 2(1 + \epsilon)m$ . Choosing

$\epsilon = \sqrt{\frac{\ln N}{m}}$  gives the regret bound

$\text{Reg}(T) = \sum_{t=1}^T \ell_t(\hat{y}_t) - \min_{i \in [N]} \sum_{t=1}^T \ell_t(x_t(e_i)) \leq 2\sqrt{m \ln N} + m$ . **Weighted Majority Mistake Bound Proof (Sketch):** Let  $w_t(i)$  be the weight of expert  $i$  at time  $t$ , initialized as  $w_1(i) = 1$ . Define total weight  $W_t = \sum_{i=1}^N w_t(i)$ .

1. \*\*Weight decrease on mistake:\*\* If  $\hat{y}_t \neq y_t$ , then at least half of the total weight was on wrong experts, which are penalized by factor  $(1 - \epsilon)$ . Hence  $W_{t+1} \leq W_t(1 - \frac{\epsilon}{2})$ .

2. \*\*Lower bound by perfect expert:\*\* Let  $e^*$  be the best expert making  $m$  mistakes. Its weight satisfies  $w_{T+1}(e^*) = (1 - \epsilon)^m \leq W_{T+1}$ .

3. \*\*Combine bounds:\*\* Iterating over  $M$  mistakes of the algorithm, we have

$(1 - \epsilon)^m \leq W_{T+1} \leq N(1 - \frac{\epsilon}{2})^M$ .

4. \*\*Take logarithms and rearrange:\*\*  $M \leq \frac{2 \ln N}{\epsilon} + (1 + \epsilon)m$ .

**Remark:** This shows the number of mistakes is bounded in terms of  $N$ ,  $\epsilon$ , and the mistakes of the best expert  $m$ .

**Randomized Weighted Majority (RWM):**

Maintain weights  $w_t(i) \geq 0$  for experts  $i \in E$ . Define probability distribution  $p_t(i) = \frac{w_t(i)}{\sum_{j=1}^N w_t(j)}$ . Sample

$I_t \sim p_t(\cdot)$  and predict  $\hat{y}_t = x_t(I_t)$ . Update weights:

$w_{t+1}(i) \leftarrow w_t(i)(1 - \epsilon)^{\mathbb{1}\{x_t(e_i) \neq y_t\}}$ .

**Expected Mistake Bound:** Let  $m = \min_{i \in [N]} \sum_{t=1}^T \ell_t(x_t(e_i))$  be mistakes of the best expert. Then  $\mathbb{E}[\sum_{t=1}^T \ell_t(\hat{y}_t)] \leq \frac{\ln N}{\epsilon} + (1 + \epsilon)m$ .

**Remark:** Randomization allows removing adversarial exploitation of deterministic predictions; expected mistake bound is similar to Weighted Majority but in expectation. **RWM Mistake Bound Proof (Sketch):** Maintain weights  $w_t(i)$ ,  $W_t = \sum_i w_t(i)$ ,  $w_1(i) = 1$ . Let  $m = \min_{i \in [N]} \sum_{t=1}^T \ell_t(x_t(e_i))$  be mistakes of the best expert.

1. **Expected weight decrease:** For any round  $t$ ,  $\mathbb{E}[W_{t+1} | W_t] = \sum_{i=1}^N w_t(i)(1 - \epsilon)^{\mathbb{1}\{x_t(i) \neq y_t\}} \leq W_t(1 - \epsilon \Pr[\hat{y}_t \neq y_t])$

2. **Lower bound by best expert:**  $w_{T+1}(e^*) = (1 - \epsilon)^m \leq W_{T+1}$

3. **Combine bounds:** Iterating over  $T$  rounds, take logarithms:  $\sum_{t=1}^T \Pr[\hat{y}_t \neq y_t] \leq \frac{\ln N}{\epsilon} + (1 + \epsilon)m$

**Remark:** The expected number of mistakes is bounded similarly to Weighted Majority; the only change is replacing deterministic weight drop with expectation over randomized predictions.

**Hedge: Weighted Majority for Bounded Losses:** Generalizes WM from 0-1 losses to arbitrary bounded losses  $\ell_t(i) = \ell(x_t(e_i), y_t) \in [0, 1]$ . On each round  $t$ , assign probability distribution over experts:

$p_t(i) = \frac{\exp(-\eta L_{t-1}(i))}{\sum_{j=1}^N \exp(-\eta L_{t-1}(j))}$ ,  $L_{t-1}(i) = \sum_{s=1}^{t-1} \ell_s(i)$

Predict  $\hat{y}_t$  by sampling from  $p_t(\cdot)$  or taking weighted combination. Update multiplicative weights:

$w_{t+1}(i) \leftarrow w_t(i) \exp(-\eta \ell_t(i))$

**Hedge: Regret Bound:** Let  $R_T = \sum_{t=1}^T \mathbb{E}[\ell_t(\hat{y}_t)] - \min_{i \in [N]} \sum_{t=1}^T \ell_t(i)$ . Then

$R_T \leq \frac{\ln N}{\eta} + \eta T$  Choosing  $\eta = \sqrt{\frac{\ln N}{T}}$  gives

$R_T \leq \sqrt{T \ln N}$ .

**Proof Sketch:** 1. Define total weight

$W_t = \sum_{i=1}^N w_t(i)$ . Update:  $W_{t+1} = \sum_i w_t(i) \exp(-\eta \ell_t(i))$ . 2. Use convexity of  $e^{-x}$ :  $W_{t+1} \leq W_t \exp(-\eta \sum_i p_t(i) \ell_t(i))$ . 3. Lower bound  $W_{T+1} \geq \exp(-\eta \sum_{t=1}^T \ell_t(i^*))$  for best expert  $i^*$ .

4. Combine upper and lower bounds:  $\sum_{t=1}^T \sum_i p_t(i) \ell_t(i) - \sum_{t=1}^T \ell_t(i^*) \leq \frac{\ln N}{\eta}$

5. Add  $\eta$  term from approximation of  $e^{-x}$  to get:  $R_T \leq \frac{\ln N}{\eta} + \eta T$

**Remark:** Hedge reduces to Weighted Majority when losses are 0-1; allows smooth, probabilistic predictions for arbitrary bounded losses. **Bayes Rule as Online Learning:** Predictors  $h \in H$  with prior  $P_0$ . Data until  $t-1$ :  $S_{t-1} = \{(x_1, y_1), \dots, (x_{t-1}, y_{t-1})\}$ . On round  $t$ , each  $h$  outputs  $p(y|x_t, h)$ . Mixture prediction:  $p(y|x_t, S_{t-1}) = \mathbb{E}_{h \sim P_{t-1}}[p(y|x_t, h)]$ . True label  $y_t$  revealed; per-expert loss  $\ell_t(h) = -\log p(y_t|x_t, h)$ .

**Bayesian Update:**  $P_t(h) = p(h|S_t) = \frac{p(y_t|x_t, h)P_{t-1}(h)}{Z_t}$ ,  $Z_t = \int p(y_t|x_t, h)P_{t-1}(h)dh$

**Equivalence to Hedge:**

$P_t(h) = \frac{\exp(-\ell_t(h))P_{t-1}(h)}{Z_t}$ ,  $\ell_t(h) = -\log p(y_t|x_t, h)$

**Regret of Bayes Rule:** Cumulative Bayesian log-loss:  $L_{\text{BLL}}(S_T) = -\sum_{t=1}^T \log p(y_t|x_t, S_{t-1})$ . Comparator for any distribution  $Q$  on  $H$ :  $L_Q(S_T) = \mathbb{E}_{h \sim Q} \sum_{t=1}^T \ell_t(h)$ . Then  $L_{\text{BLL}}(S_T) - L_Q(S_T) \leq \text{KL}(Q||P_0)$

**Remark:** Bayes rule in this online setting is equivalent to Hedge with log-loss; regret is controlled by KL divergence to the prior. **Summary of Online Learning Algorithms:**

**Follow the Leader (FTL)** [Hannan, 1957; Kalai & Vempala, 2005]: 0-1 loss, perfect expert assumption, mistake bound  $N - 1$ .

**Halving Algorithm** [Barzdinvs & Freivalds, 1972; Angluin, 1988; Littlestone, 1988]: 0-1 loss, perfect expert assumption, mistake bound  $\log_2 N$ .

**Weighted Majority (WM)** [Littlestone & Warmuth, 1994]: 0-1 loss, mistake bound  $\frac{2}{\epsilon} \ln N + 2(1 + \epsilon)m$ , where  $m$  is mistakes of best expert.

**Randomized Weighted Majority (RWM)** [Littlestone & Warmuth, 1994]: 0-1 loss, expected mistake bound  $\frac{1}{\epsilon} \ln N + \epsilon m$ .

**Hedge** [Freund & Schapire, 1997]: Bounded losses  $\ell_t(i) \in [0, 1]$ , regret bound  $\frac{\ln N}{\eta} + \eta T$ .

**Bayesian Online Learning** [Freund et al., 1997; Kakade & Ng, 2004; Banerjee, 2006]: Log-loss, cumulative regret w.r.t any distribution  $Q$  bounded by KL divergence:  $L_{\text{Bayes}} - L_Q \leq \text{KL}(Q||P_0)$ .

**Online Convex Optimization Online Convex Optimization (OCO)** Decision space  $X \subset \mathbb{R}^d$  convex, nonempty. Adversary selects convex loss  $f_t : X \rightarrow \mathbb{R}$  at each round  $t \in [T]$ . Assumptions: bounded losses

$\sup_{x \in X} f(x) - \inf_{x \in X} f(x) \leq B$ , bounded diameter  $D = \sup_{x, y \in X} \|x - y\| < \infty$ . Protocol: learner chooses  $x_t \in X$ , adversary reveals  $f_t$ , loss incurred  $f_t(x_t)$ .

**Regret of OCO** Static regret:

$R_T := \sup_{f_1, T: \mathcal{F}} \sum_{t=1}^T f_t(x_t) - \min_{x \in X} \sum_{t=1}^T f_t(x)$ .

**Applications:**

**Prediction with Expert Advice** Decision set:  $n$ -dimensional simplex

$\Delta_n = \{x \in \mathbb{R}^n : \sum_i x_i = 1, x_i \geq 0\}$ . Cost vector

$g_t \in \mathbb{R}^n$ , linear loss  $f_t(x) = g_t^\top x$ .

**Online Spam Detection** Emails  $a \in \mathbb{R}^d$ , decision  $x_t \in \mathbb{R}^d$ , prediction  $\hat{y}_t = \text{sign}(x_t^\top a)$ , loss

$f_t(x) = (\hat{y}_t - y_t)^2$  with label  $y \in \{-1, 1\}$ .

**Online Stock Market Movement** Classification  $Y = \{0, 1\}$ , predict  $\hat{y}_t$  using features  $x_t$ , loss can be binary cross-entropy.

**Portfolio Selection** Decision  $x_t \in \Delta_n$  (wealth distribution), market returns  $r_t \in \mathbb{R}_{>0}^n$ , gain  $r_t^\top x_t$ , regret:  $\sum_{t=1}^T \log(r_t^\top x^*) - \sum_{t=1}^T \log(r_t^\top x_t)$  for best fixed  $x^* \in \Delta_n$ .

**Online Shortest Path** Graph  $G = (V, E)$ , pick path  $p_t$  from  $u$  to  $v$ , adversary chooses edge weights  $w_t \in \mathbb{R}^{|E|}$ , loss  $f_t(p_t) = \sum_{e \in p_t} w_t(e)$ . Flow view:

$x \in K$ , expected cost  $f_t(x) = w_t^\top x$ , subject to unit flow, conservation, and capacities.

**Recommendation Systems**  $n$  users,  $m$  items, preference matrix  $X \in \{0, 1\}^{n \times m}$ . Decision  $X_t \in K$ , adversary reveals  $(i_t, j_t)$ , loss  $f_t(X) = (X_{i_t j_t} - y_t)^2$ . Comparator: best low-rank matrix (few latent factors).

**Convex sets:**  $X \subset \mathbb{R}^d$  is convex if

$\forall x, y \in X, \alpha \in [0, 1], (1 - \alpha)x + \alpha y \in X$ . **Convex functions:**  $f : X \rightarrow \mathbb{R}$  is convex if  $\forall x, y \in X, \alpha \in [0, 1], f((1 - \alpha)x + \alpha y) \leq (1 - \alpha)f(x) + \alpha f(y)$ .

• **Differentiable:**  $f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle$ . • **Twice differentiable:**  $\nabla^2 f(x) \succeq 0$ . **Sub-gradient:**  $g$  satisfies  $f(y) \geq f(x) + \langle g, y - x \rangle, \forall y$ . Set of all sub-gradients:  $\partial f(x)$ .

**Jensen's inequality:** For convex  $f$  and integrable random variable  $Z$ ,  $f(\mathbb{E}[Z]) \leq \mathbb{E}[f(Z)]$ . **Strong Convexity:**  $f$  is  $\alpha$ -strongly convex if

$\forall x, y \in X, f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\alpha}{2} \|y - x\|^2$ .

**Smoothness:**  $f$  is  $\beta$ -smooth if

$\forall x, y \in X, f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\beta}{2} \|y - x\|^2$ ,

equivalently  $\|\nabla f(y) - \nabla f(x)\| \leq \beta \|y - x\|$ .

**Constrained Convex Optimization: Optimality Condition:** Minimize a convex  $f : X \rightarrow \mathbb{R}$  over nonempty, closed, convex  $X \subset \mathbb{R}^d$ . A differentiable point  $x^* \in X$  is a global minimizer iff

$\langle \nabla f(x^*), y - x^* \rangle \geq 0$  for all  $y \in X$ . If  $X = \mathbb{R}^d$  (unconstrained), reduces to  $\nabla f(x^*) = 0$ .

**Projection to Convex Sets:** For closed, convex  $X \subset \mathbb{R}^d$ , the Euclidean projection of  $y \in \mathbb{R}^d$  is  $\Pi_X(y) := \arg \min_{x \in X} \|x - y\|$ . Pythagorean property: if  $x = \Pi_X(y)$ , then

$\forall z \in X, \|y - z\|^2 \geq \|x - z\|^2 + \|y - x\|^2$ , equivalently  $\langle y - x, z - x \rangle \leq 0$ .

**Online Gradient Descent (OGD):** At each round  $t$ :

Learner plays  $x_t \in X$ ; adversary reveals convex  $f_t : X \rightarrow \mathbb{R}$  and subgradient  $g_t \in \partial f_t(x_t)$ ; learner suffers loss  $f_t(x_t)$ ; update  $x_{t+1} \leftarrow \Pi_X(x_t - \eta_t g_t)$ .

**OGD Regret (Convex Functions):** Assume  $D := \sup_{x, y \in X} \|x - y\| < \infty$  and  $\|g_t\| \leq G$  for all  $t \in [T]$ . For any  $x \in X$  and stepsize  $\eta_t > 0$ ,

$\text{Reg}(T) \leq \sum_{t=1}^T \frac{\|x_t - x\|^2 - \|x_{t+1} - x\|^2}{2\eta_t}$ . With step-size

$\eta_t = \frac{D}{G\sqrt{t}}$ ,  $\text{Reg}(T) \leq \frac{3}{2} DG\sqrt{T}$ .

**OGD Regret Proof (Sketch):** By convexity,  $f_t(x_t) - f_t(x) \leq \langle g_t, x_t - x \rangle$ . By projection property,  $\|x_{t+1} - x\|^2 \leq \|x_t - x\|^2 - 2\eta_t \langle g_t, x_t - x \rangle + \eta_t^2 \|g_t\|^2$ . Rearrange and sum over  $t = 1 : T$ :

$\sum_{t=1}^T (f_t(x_t) - f_t(x)) \leq \sum_{t=1}^T \frac{\|x_t - x\|^2 - \|x_{t+1} - x\|^2}{2\eta_t} + \sum_{t=1}^T \frac{\eta_t}{2} \|g_t\|^2$ . Use telescoping and bounds  $\|x_t - x\| \leq D$ ,  $\|g_t\| \leq G$ , set

$\eta_t = \frac{D}{G\sqrt{t}}$ , and note  $\sum_{t=1}^T \frac{1}{\sqrt{t}} \leq 2\sqrt{T}$  to get

$\text{Reg}(T) \leq \frac{3}{2} DG\sqrt{T}$ .

**OCO Lower Bound:** For any (possibly randomized) online algorithm and any  $T \geq 1$  there exist a convex set  $X$  and convex losses  $\{f_t\}_{t=1}^T$  s.t.  $\text{Ref}(T) := \sum_{t=1}^T f_t(x_t) - \min_{x \in X} \sum_{t=1}^T f_t(x) = \Omega(DG\sqrt{T})$ , where  $D := \sup_{x, y \in X} \|x - y\|$  and  $G := \sup_t \|\nabla f_t(x_t)\|$ . The lower bound holds even if  $f_t$  are i.i.d.

**OGD (Strongly Convex) — Result:** Assume each  $f_t$  is  $\alpha$ -strongly convex on  $X$ ,

$D := \sup_{x, y \in X} \|x - y\| < \infty$  and  $\|g_t\| \leq G$ . With  $x_{t+1} = \Pi_X(x_t - \eta_t g_t)$  and  $\eta_t = 1/(\alpha t)$ ,

$\text{Ref}(T) \leq \frac{G^2}{2\alpha} (1 + \log T)$ .

**OGD (Strongly Convex) — Proof Sketch:** Strong convexity:  $f_t(x_t) - f_t(x) \leq \langle g_t, x_t - x \rangle - \frac{\alpha}{2} \|x_t - x\|^2$ . Projection gives

$\|x_{t+1} - x\|^2 \leq \|x_t - x\|^2 - 2\eta_t \langle g_t, x_t - x \rangle + \eta_t^2 \|g_t\|^2$ . Combine, rearrange and sum to obtain

$\sum_{t=1}^T (f_t(x_t) - f_t(x)) \leq \frac{1}{2} \sum_{t=1}^T \left( \frac{\|x_t - x\|^2 - \|x_{t+1} - x\|^2}{\eta_t} \right.$

$\left. + \alpha \|x_t - x\|^2 \right) + \frac{G^2}{2} \sum_{t=1}^T \eta_t$ . Set  $\eta_t = 1/(\alpha t)$ , use

$\sum_{t=1}^T 1/t \leq 1 + \log T$  and  $\|x_t - x\| \leq D$  to get the stated bound.

**OCO — Feedback Hierarchy:** Algorithms by feedback: (i) \*\*Zeroth-order (loss-only):\*\* observe only  $f_t(x_t)$ . e.g. Halving, Hedge; (ii) \*\*First-order:\*\* observe (sub)gradient at played point. e.g. OGD, OMD, FTRL; (iii) \*\*Second-order:\*\* use curvature/Hessian information. e.g. Online Newton methods. Choice trades off oracle power vs.

computational cost and achievable regret rates.

**Exp-Concave Functions:** A convex  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  is  $\alpha$ -exp-concave on  $K$  if  $g(x) = \exp(-\alpha f(x))$  is concave on  $K$ . For concave  $g$ ,  $\nabla^2 g(x) \preceq 0$ .

**Gradients and Hessians:**

$\nabla g(x) = -\alpha \nabla f(x) e^{-\alpha f(x)}$ ,  $\nabla^2 g(x) = \alpha e^{-\alpha f(x)} (\alpha \nabla f(x) \nabla f(x)^\top - \nabla^2 f(x))$ . Thus concavity of  $g$  implies the \*\*equivalent curvature condition\*\*:

$\nabla^2 f(x) \succeq \alpha \nabla f(x) \nabla f(x)^\top, \quad \forall x \in K$ .

**Comparison:** Strong convexity requires  $\nabla^2 f(x) \succeq 0$  (full-rank Hessian), while  $\alpha$ -exp-concavity only lower bounds curvature by a \*\*rank-1 matrix\*\*.

## Exp-Concave Functions — Quadratic Lower

**Bound:** If  $f : K \rightarrow \mathbb{R}$  is  $\alpha$ -exp-concave, with domain diameter  $D$  and gradient bound  $\|\nabla f(x)\| \leq G$ , then for any  $\gamma \leq \frac{1}{2} \min\{1, \frac{1}{GD\alpha}\}$  and all  $x, y \in K$ ,  $f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle + \frac{\gamma}{2} (x - y)^\top \nabla f(y) \nabla f(y)^\top (x - y)$ . This provides a \*\*quadratic lower bound\*\* analogous to strong convexity but scaled by gradient magnitude.

**Application: Universal Portfolios — CRP:** -

\*\*Constant Rebalanced Portfolio (CRP):\*\* comparator  $x^* \in \Delta_k$  fixed distribution over  $k$  assets. - Rebalance each step to restore  $x^*$  after price movements. - Goal: adaptive portfolio competitive with best CRP in hindsight.

**Example (2 assets):** alternating high returns

$$r_t = \begin{cases} (2, 1/2), & t \text{ odd} \\ (1/2, 2), & t \text{ even} \end{cases}, \quad x^* = (1/2, 1/2) \text{ Per-step}$$

growth factor:  $1/2 \cdot 2 + 1/2 \cdot 1/2 = 1.25$ .

**Wealth Dynamics:** At round  $t$ , pick  $x_t \in \Delta_n$ ,

adversary reveals  $r_t \in \mathbb{R}_{>0}^n$ . Wealth evolves as

$$W_{t+1} = W_t r_t^\top x_t, \quad W_T = W_1 \prod_{t=1}^T r_t^\top x_t. \text{ Maximizing}$$

$$W_T \equiv \text{maximizing } \sum_{t=1}^T h_t(x_t), \quad h_t(x) = \log r_t^\top x.$$

**Loss Perspective:** Define loss:  $f_t(x) = -\log r_t^\top x$ .

$$\text{Then } \nabla f_t(x) = -\frac{r_t}{r_t^\top x}, \quad \nabla^2 f_t(x) = \frac{r_t r_t^\top}{(r_t^\top x)^2} =$$

$\nabla f_t(x) \nabla f_t(x)^\top$ . Hence  $f_t$  is \*\*1-exp-concave\*\*.

**Goal:** design algorithm with \*\*sublinear regret\*\* w.r.t best CRP  $x^*$  in hindsight:

$$\text{Reg}(T) = \max_{x^* \in \Delta_n} \sum_{t=1}^T \log r_t^\top x^* - \sum_{t=1}^T \log r_t^\top x_t.$$

**Playing the Weighted Average — EWOO:** -

Maintain weights over all CRPs  $x \in K$ :

$$w_t(x) = \exp\left(-\alpha \sum_{r=1}^{t-1} f_r(x)\right) - \text{CRPs with low}$$

cumulative loss get higher weights. - Next portfolio:

$$\text{**weighted average**}: x_t = \frac{\int_K x w_t(x) dx}{\int_K w_t(x) dx} - \text{Regret}$$

guarantee for  $\alpha$ -exp-concave losses on  $K \subset \mathbb{R}^n$ :

$\text{Reg}_T(\text{EWOO}) \leq \frac{n}{\alpha} \log T + \frac{2}{\alpha}$ . - Sampling variant: draw  $x_t \sim w_t(\cdot)$ , achieves same bound in expectation.

**EWOO — Bayes Perspective:** - Normalize weights:

$\pi_t(x) \propto w_t(x)$ , acts as a \*\*prior\*\*.

- Posterior update:

$$\pi_{t+1}(x) \propto \pi_t(x) e^{-\alpha f_t(x)}, \quad x_t = \mathbb{E}_{x \sim \pi_{t+1}}[x].$$

- Posterior mean differs from MAP:

Posterior mean:  $\mathbb{E}_{x \sim \pi_t}(x|z)[X]$ , MAP:  $\arg \max_x \pi(x|z)$ .

- Finite experts ( $n$  experts,  $K = \Delta_n$ ), linear losses

$f_t(x) = \ell_t^\top x$ : EWOO reduces to \*\*exponential weights

over experts\*\*.

**Computational Considerations:** - Exact posterior

mean often intractable; use \*\*sampling\*\* to estimate

$\mathbb{E}_{x \sim \pi_{t+1}}[x]$ . - Classical: polynomial-time sampling;

modern: generative models. - Random sampling: regret

guarantee holds \*\*in expectation\*\*., not necessarily

high-probability.

**Online Newton Step (ONS) for Exp-Concave**

**Functions:** - Motivated by Newton updates  $A_t^{-1} \nabla_t$ ,

with  $A_t$  approximating Hessian and  $\nabla_t$  the gradient. -

Quasi-Newton approach: uses gradient outer products;

still first-order.

**Set-up:** for  $\alpha$ -exp-concave losses  $f_t$  with gradients

$\nabla f_t(x_t)$ :  $A_0 = \varepsilon I$ ,  $A_t = A_{t-1} + \nabla_t \nabla_t^\top$ ,  $\gamma > 0$

Updates (Newton step + generalized projection):

$$y_{t+1} = x_t - \frac{1}{\gamma} A_t^{-1} \nabla_t, \quad x_{t+1} = \Pi_K^{A_t}(y_{t+1}) :=$$

$$\arg \min_{x \in K} \|x - y_{t+1}\|_{A_t}^2.$$

**ONS for Portfolio Selection:** - Exp-concave loss:

$$f_t(x) = -\log(r_t^\top x), \text{ gradient } \nabla_t = -\frac{r_t}{r_t^\top x}. \text{ - Hessian}$$

$$\text{approximation: } A_t = A_{t-1} + \frac{r_t r_t^\top}{(r_t^\top x_t)^2}. \text{ - Updates: same}$$

Newton + generalized projection as above. - Projection: linear constrained convex optimization; e.g., onto  $\Delta_n$ .

**Regret of ONS:** Assumptions:  $\alpha$ -exp-concave losses on  $K \subset \mathbb{R}^n$ ,  $\|\nabla f_t(x_t)\| \leq G$ , diameter

$$D = \sup_{x, y \in K} \|x - y\|.$$

$$\text{Reg}_T(\text{ONS}) \leq 2\left(\frac{1}{\alpha} + GD\right)n \log T \quad \text{for suitable } \gamma, \varepsilon.$$

**Regret of ONS:** - Assumptions:  $\alpha$ -exp-concave losses

$f_t$  on  $K \subset \mathbb{R}^n$ ,  $\|\nabla f_t(x_t)\| \leq G$ , diameter

$$D = \sup_{x, y \in K} \|x - y\|. \text{ - ONS update:}$$

$$y_{t+1} = x_t - \frac{1}{\gamma} A_t^{-1} \nabla_t, \quad x_{t+1} = \Pi_K^{A_t}(y_{t+1}) \text{ with}$$

$$A_t = A_0 + \sum_{s=1}^t \nabla_s \nabla_s^\top, \quad A_0 = \varepsilon I.$$

- \*\*Proof sketch:\*\* 1. Exp-concavity implies a

\*\*quadratic lower bound\*\* on loss:

$$f_t(x_t) - f_t(x) \leq \nabla_t^\top (x_t - x) - \frac{\alpha}{2} (\nabla_t^\top (x_t - x))^2. \text{ 2. Use}$$

generalized projection property:

$$\|x_{t+1} - x\|_{A_t}^2 \leq \|x_t - x\|_{A_t}^2 - \frac{2}{\gamma} \nabla_t^\top (x_t - x) + \frac{1}{\gamma^2} \|\nabla_t\|_{A_t}^{-2}.$$

3. Combine 1 & 2, sum over  $t = 1 \dots T$ , and bound

$$\sum_t \|\nabla_t\|_{A_t}^{-2} \leq n \log T \text{ (standard ONS argument).}$$

**Computational Considerations:** - Rank-1 updates:

$A_t^{-1}$  via Sherman–Morrison in  $O(n^2)$  per

round. - Projection: solve quadratic program; efficient

for  $\Delta_n$ . - Scaling: memory  $O(n^2)$ ; for large  $n$ , use

diagonal or sketching approximations.

**OCO Regret: Summary:** - **First-order methods:** -

OGD (convex):  $\text{Reg}(T) = O(DG\sqrt{T})$  - OGD (strongly

$$\text{convex}): \text{Reg}(T) = O\left(\frac{G^2}{\alpha} \log T\right)$$

- **Exp-concave losses:** - EWOO:  $\text{Reg}_T \leq \frac{n}{\alpha} \log T + \frac{2}{\alpha}$

- ONS:  $\text{Reg}_T \leq 2\left(\frac{1}{\alpha} + GD\right)n \log T$  **Follow the**

**Leader (FTL):** - Strategy: pick action minimizing

cumulative past loss  $x_{t+1} = \arg \min_{x \in K} \sum_{\tau=1}^t f_\tau(x)$  -

Equivalent to ERM in i.i.d. supervised learning, or

fictitious play in economics. - Pitfall: in non-stationary

settings, FTL can oscillate and incur  $\Omega(T)$  regret.

**Regularization Preliminaries:** - Add strongly

convex, smooth regularizer  $R(x)$  to stabilize FTL. -

Diameter relative to  $R$ :  $D_R = \max_{x, y \in K} R(x) - R(y)$  -

Norms induced by Hessian:  $\|x\|_A = \sqrt{x^\top A x}$ , dual norm

$$\|y\|_A^* = \sqrt{y^\top A^{-1} y}.$$

**Bregman Divergence:** - For strongly convex  $R$ :

$$B_R(x, y) = R(x) - R(y) - \langle \nabla R(y), x - y \rangle - \text{Examples: -}$$

$$R(x) = \frac{1}{2} \|x\|_2^2 \implies B_R(x, y) = \frac{1}{2} \|x - y\|_2^2 -$$

$$R(x) = \sum_j x_j \log x_j, x \in \Delta_n \implies B_R(x, y) = \text{KL}(x, y)$$

**Regularized Follow the Leader (RFTL):** - Sequence

of convex functions  $f_t(x)$ . - Gradient:  $\nabla_t := \nabla f_t(x_t)$  -

Convexity:  $f_t(x_t) - f_t(x^*) \leq \nabla_t^\top (x_t - x^*)$  - RFTL

update:  $x_1 = \arg \min_{x \in K} R(x)$ ,  $x_{t+1} =$

$$\arg \min_{x \in K} \sum_{\tau=1}^t \eta \nabla_\tau^\top x + R(x) - \text{Stabilizes FTL and}$$

ensures sublinear regret.

**Bregman Divergences (Contd.):** - Bregman

divergence: error of first-order Taylor approximation

$$B_R(x, y) = R(x) - R(y) - \langle \nabla R(y), x - y \rangle - \text{Mean-value}$$

theorem:  $\exists z \in [x, y], H_z = \nabla^2 R(z)$

$$B_R(x, y) = (x - y)^\top H_z (x - y) - \text{Define local norm:}$$

$$\|v\|_{x, y}^2 := B_R(x, y), \|\cdot\|_t = \|\cdot\|_{x_t, y_t}, \text{ dual norm}$$

$$\|\cdot\|_{x, y}^* \text{ and } \|\cdot\|_t^* \text{ denotes the dual norm for } \|\cdot\|_t.$$

**Regret of RFTL: Proof Sketch:** - Let

$$x_{t+1} = \arg \min_{x \in K} \sum_{\tau=1}^t \nabla_\tau^\top x + R(x) \text{ and any } u \in K.$$

- By convexity and first-order optimality of RFTL:

$$\sum_{\tau=1}^t \nabla_\tau^\top (x_{\tau+1} - u) \leq R(u) - R(x_1) - \text{Use generalized}$$

Pythagorean inequality for Bregman divergence / local

$$\text{norm } \|\cdot\|_t:$$

$$\nabla_t^\top (x_t - u) \leq \eta \|\nabla_t\|_t^2 + \frac{1}{2\eta} (B_R(u, x_t) - B_R(u, x_{t+1}))$$

- Sum over  $t = 1 \dots T$ , telescoping cancels intermediate

Bregman terms:  $\text{Reg}_T(\text{RFTL}) = \sum_{t=1}^T f_t(x_t) - f_t(u) \leq$

$$\sum_{t=1}^T \eta \|\nabla_t\|_t^2 + R(u) - R(x_1) - \text{If } \|\nabla_t\|_t^* \leq G_R \text{ and}$$

$$D_R = R(u) - R(x_1), \text{ choose } \eta = \sqrt{D_R / (G_R^2 T)} \text{ to}$$

$$\text{balance terms: } \text{Reg}_T(\text{RFTL}) \leq 2D_R G_R \sqrt{2T}$$

**Bregman Divergence: Identities and Inequalities:**

- Definition: for a strictly convex, differentiable  $R(x)$

$$B_R(x, y) := R(x) - R(y) - \langle \nabla R(y), x - y \rangle - \text{Let } R(x) \text{ be}$$

strictly convex,  $C \subset \mathbb{R}^n$  closed and convex, and define the

$$\text{Bregman projection } \Pi_C^R(y) := \arg \min_{x \in C} B_R(x, y)$$

- Then for any  $x \in C$ :

$$B_R(x, y) \geq B_R(x, \Pi_C^R(y)) + B_R(\Pi_C^R(y), y)$$

- \*\*Non-negativity:\*\*

$$B_R(x, y) \geq 0, \quad \text{with equality iff } x = y$$

- \*\*Three-point identity:\*\* for any  $x, y, z$ :

$$\langle \nabla R(x) - \nabla R(y), z - x \rangle = B_R(z, y) - B_R(z, x) - B_R(x, y)$$

- \*\*Quadratic form (via mean value theorem):\*\*  $\exists z$  on

segment  $[x, y]$  s.t.  $B_R(x, y) = (x - y)^\top \nabla^2 R(z) (x - y)$

- \*\*Strong convexity bound:\*\* if  $R$  is  $\alpha$ -strongly convex

w.r.t norm  $\|\cdot\|$ :  $B_R(x, y) \geq \frac{\alpha}{2} \|x - y\|^2$

- \*\*Smoothness bound:\*\* if  $R$  has  $\beta$ -Lipschitz gradient:

$$B_R(x, y) \leq \frac{\beta}{2} \|x - y\|^2$$

**Gradient Descent vs Mirror Descent:** - First-order

Taylor approximation:  $f(x) \approx f(x_t) + \langle x - x_t, \nabla f(x_t) \rangle$

- Gradient descent: squared Euclidean regularization

$$x_{t+1} = \arg \min_{x \in K} f(x_t) + \langle x - x_t, \nabla f(x_t) \rangle + \frac{1}{2\eta} \|x - x_t\|_2^2$$

- Mirror descent: Bregman regularization  $x_{t+1} =$

$$\arg \min_{x \in K} f(x_t) + \langle x - x_t, \nabla f(x_t) \rangle + \frac{1}{2\eta} B_R(x, x_t) -$$

Intuition: Mirror descent generalizes GD to

non-Euclidean geometry via  $R(x)$ .

**Gradient Descent vs. Mirror Descent:**

- First-order Taylor expansion at current iterate  $x_t$ :

$$f(x) \approx f(x_t) + \langle x - x_t, \nabla f(x_t) \rangle$$

- \*\*Gradient Descent (Euclidean regularization):\*\*

$$x_{t+1} = \arg \min_{x \in K} \langle \nabla f(x_t), x - x_t \rangle + \frac{1}{2\eta} \|x - x_t\|_2^2$$

- \*\*Mirror Descent (Bregman regularization):\*\*

$$x_{t+1} = \arg \min_{x \in K} \langle \nabla f(x_t), x - x_t \rangle + \frac{1}{2\eta} B_R(x, x_t)$$

$$\implies \nabla R(x_{t+1}) = \nabla R(x_t) - \eta \nabla f(x_t), \quad x_{t+1} =$$

$$(\nabla R)^{-1}(\nabla R(x_t) - \eta \nabla f(x_t))$$

- Properties of  $R(\cdot)$  (Legendre type): -  $\nabla R(x)$

monotone, invertible - Conjugate:

$$R^*(y) = \sup_{x \in K} \langle x^\top y - R(x) \rangle, \quad \nabla R^* = (\nabla R)^{-1} -$$

Duality between primal space  $x$  and dual space  $\nabla R(x)$  -

Connections: exponential families, information

geometry

**Online Mirror Descent (OMD):**

- Maintain two iterates: -  $y_t$  (unconstrained),  $x_t \in K$

(constrained) - Regularization  $R(x)$ , step size  $\eta$  -

Initialize:  $\nabla R(y_1) = 0$ ,  $x_1 = \arg \min_{x \in K} B_R(x, y_1)$

- For  $t = 1, \dots, T$ : 1. Play  $x_t$ , observe loss  $f_t$ , gradient

$$\nabla_t = \nabla f_t(x_t) \text{ 2. Update } y_{t+1} - \text{Lazy:}$$

$$\nabla R(y_{t+1}) = \nabla R(y_t) - \eta \nabla_t - \text{Agile:}$$

$$\nabla R(y_{t+1}) = \nabla R(x_t) - \eta \nabla_t \text{ 3. Project onto } K:$$

$$x_{t+1} = \arg \min_{x \in K} B_R(x, y_{t+1})$$

- \*\*Connection to RFTL:\*\* - For linear losses

$f_t(x) = \ell_t^\top x$ , lazy OMD = RFTL - Updates satisfy

$$\nabla R(x_{t+1}) = -\eta \sum_{\tau=1}^t \nabla_\tau - \text{Regret bound of lazy OMD}$$

follows directly from RFTL analysis

**OMD (Lazy) and RFTL:** For linear losses

$$f_t(x) = \ell_t^\top x, \nabla f_t(x) = \ell_t. \text{ Lazy OMD update:}$$

$$x_{t+1} = \arg \min_{x \in K} B_R(x, y_t),$$

$$\nabla R(y_{t+1}) = \nabla R(y_t) - \eta \ell_t. \text{ RFTL update:}$$

$$x_{t+1} = \arg \min_{x \in K} \sum_{\tau=1}^t \eta \ell_\tau^\top x + R(x),$$

$$\nabla R(x_{t+1}) = -\eta \sum_{\tau=1}^t \ell_\tau. \text{ Conclusion: OMD (lazy) =}$$

RFTL for linear losses, regret bound follows from

RFTL analysis.

**OMD (Agile) Regret:** For any  $u \in K$ , OMD (agile)

$$\text{satisfies } \text{Reg}_T(\text{OMD}) \leq \frac{\eta}{4} \sum_{t=1}^T \|\nabla_t\|_t^2 + \frac{R(u) - R(x_1)}{2\eta}.$$

$$\text{Assume } \|\nabla_t\|_t^* \leq G_R. \text{ Choosing } \eta = \sqrt{\frac{R(u) - R(x_1)}{2G_R^2 T}}$$

gives  $\text{Reg}_T(\text{OMD}) \leq D_R G_R \sqrt{T}$ . If  $T$  unknown, use

doubling trick.

**Proof Sketch:** Update rule:

$$x_{t+1} = \arg \min_{x \in \mathcal{X}} \left\{ \nabla_t^\top x + \frac{1}{\eta} D_R(x, x_t) \right\}$$

Optimality condition:  $\nabla_t + \frac{1}{\eta} (\nabla R(x_{t+1}) - \nabla R(x_t)) = 0$

Bregman property and Fenchel-Young:

$$(\nabla R(x_t) - \nabla R(x_{t+1}))^\top (u - x_{t+1}) =$$

$$D_R(u, x_t) - D_R(u, x_{t+1}) - D_R(x_{t+1}, x_t),$$

$$(\nabla R(x_t) - \nabla R(x_{t+1}))^\top (x_t - x_{t+1}) \leq \frac{\eta^2}{2} \|\nabla_t\|_{t,*}^2$$

$$\text{Final inequality:}$$

$$\nabla_t^\top (x_t - u) \leq \frac{1}{2} (\|u - x_t\|_t^2 - \|u - x_{t+1}\|_t^2) + \frac{\eta^2}{2} \|\nabla_t\|_{t,*}^2$$

By convexity,  $f_t(x_t) - f_t(u) \leq \nabla_t^\top (x_t - u)$ . From OMD

update and Bregman divergence definition:

&lt;

expectation and stability of  $D$  give  $\mathbb{E}[f_t(x_t)] - f_t(x^*) \leq \eta G^* L + \sigma/\eta$ . 4. Sum over  $t = 1 : T$  and optimize  $\eta$  to get  $\text{Reg}_T \leq 2DG^* L \sqrt{\sigma T}$ . **FTPL (single-sample, linear losses)** Draw  $\xi_0 \sim D$  once and keep it. Initialize  $x_1 = \arg \min_{x \in K} \xi_0^\top x$ . For  $t = 1, \dots, T$  play  $x_t$ , incur linear loss  $f_t(x_t) = g_t^\top x_t$  (so  $\nabla_t = g_t$ ), and set  $\hat{x}_{t+1} = \arg \min_{x \in K} \eta \sum_{\tau=1}^t g_\tau^\top x + \xi_0^\top x$ . **In-expectation regret bound (linear losses)** Because expectation can be moved outside the loss, the single-sample FTPL satisfies

$$\mathbb{E}_{\xi_0 \sim D} \left[ \sum_{t=1}^T f_t(\hat{x}_t) - f_t(x^*) \right] \leq \eta D G^* L T + \frac{\sigma D}{\eta},$$

where  $D$  is the diameter of  $K$ ,  $G^*$  is the gradient bound in the dual norm, and  $D$  (resp.  $\sigma, L$ ) denote diameter (resp. noise std / sensitivity) constants as above.

Choosing  $\eta = \sqrt{\frac{\sigma}{G^* L T}}$  yields

$\text{Reg}_T(\text{FTPL}) \leq 2DG^* L \sqrt{\sigma T}$ , an  $O(\sqrt{T})$  in-expectation bound. Note: this is an average-case guarantee; obtaining high-probability bounds requires extra concentration arguments and stronger assumptions on  $D$ . **Stochastic GD with Momentum** Idea: smooth the effective gradient to accelerate and stabilize updates, useful when condition number is large. (Mini-batch) gradient at step  $t$ :  $g_t = \nabla L_B(\theta_t)$ . A common momentum form: velocity and parameter updates  $v_{t+1} = \beta v_t - g_t$ ,  $\theta_{t+1} = \theta_t + \eta v_{t+1}$ . An alternative (equivalent up to rescaling of  $v, \eta$ ) writes  $v_{t+1} = \beta v_t - \eta g_t$ ,  $\theta_{t+1} = \theta_t + v_{t+1}$ . Choice  $\beta \in [0, 1]$ .  $\beta = 0$  recovers (mini-batch) GD. Typical  $\beta \in [0.5, 0.99]$ . **Polyak Momentum (Heavy Ball)** Updates with momentum coefficient  $\mu \in [0, 1]$ :

$v_{t+1} = \mu v_t - \eta \nabla L_B(\theta_t)$ ,  $\theta_{t+1} = \theta_t + v_{t+1}$ . Combine:  $\theta_{t+1} - \theta_t = \mu(\theta_t - \theta_{t-1}) - \eta \nabla L_B(\theta_t)$ . Interpretation: keep moving in previous direction (inertia) plus a GD correction at current location. Can accelerate but may overshoot if  $\mu, \eta$  not tuned.

#### Nesterov Momentum

Set  $v_0 = 0, \mu \in [0, 1]$ . Lookahead gradient:  $v_{t+1} = \mu v_t - \eta \nabla L_B(\theta_t + \mu v_t)$ ,  $\theta_{t+1} = \theta_t + v_{t+1}$ . Equivalently  $\theta_{t+1} - \theta_t = \mu(\theta_t - \theta_{t-1}) - \eta \nabla L_B(\theta_t + \mu(\theta_t - \theta_{t-1}))$ . Interpretation: gradient evaluated at the “lookahead” position  $\theta_t + \mu v_t$  gives an anticipatory correction; often yields improved empirical and theoretical convergence.

#### Practical Notes

Momentum smooths noisy gradients (reduces variance), helps escape shallow minima and speeds convergence on ill-conditioned problems. Tune  $\eta$  and  $\beta/\mu$  jointly; large  $\beta$  demands smaller  $\eta$ . For convex quadratic problems, properly tuned momentum can provably accelerate convergence; in nonconvex deep learning practice, momentum + learning-rate schedules (and sometimes weight decay / batch-norm) give best results.

#### Adaptive Gradient Methods: Newton and Variants

For simplicity, denote loss as  $L(\theta)$ . Gradient:  $\nabla L(\theta)$ , Hessian:  $\nabla^2 L(\theta)$ . Newton’s method directly uses curvature (second-order information):  $\theta_{t+1} \leftarrow \theta_t - [\nabla^2 L(\theta_t)]^{-1} \nabla L(\theta_t)$ . This is difficult for high-dimensional problems (e.g., neural networks). For  $10^{12}$  parameters, the Hessian is a  $10^{12} \times 10^{12}$  matrix. Moreover, the Hessian may not be positive definite. Related methods include: limited-memory BFGS, quasi-Newton methods, and conjugate gradient methods. (4pt)

#### Adaptive Gradient Methods: Other Ideas

Adaptive curvature-based methods: AdaGrad, AdaDelta, RMSProp, Adam. Noisy gradient algorithms: Langevin dynamics, dropout, simulated

annealing. Natural gradient methods: view gradient descent from the function space perspective, while computation happens in the parameter space.

#### Gradient Descent: Regret Proof (Detailed)

The projected gradient descent update is given by  $\theta_{t+1} = \Pi_\Theta(\theta_t - \eta g_t)$ , where  $g_t \in \partial \ell_t(\theta_t)$ . Using the Pythagorean property of projection, we have  $\|\theta_{t+1} - \theta^*\|^2 \leq \|\theta_t - \eta g_t - \theta^*\|^2 = \|\theta_t - \theta^*\|^2 - 2\eta g_t^\top(\theta_t - \theta^*) + \eta^2 \|g_t\|^2$ . Rearranging gives, for any  $t$ ,

$$g_t^\top(\theta_t - \theta^*) \leq \frac{\|\theta_t - \theta^*\|^2 - \|\theta_{t+1} - \theta^*\|^2}{2\eta} + \frac{\eta}{2} \|g_t\|^2.$$

By convexity of each loss  $\ell_t$ ,  $\ell_t(\theta_t) - \ell_t(\theta^*) \leq g_t^\top(\theta_t - \theta^*)$ . Summing over  $t = 1, \dots, T$  and using telescoping terms, we obtain  $\sum_{t=1}^T (\ell_t(\theta_t) - \ell_t(\theta^*)) \leq \frac{\|\theta_1 - \theta^*\|^2 - \|\theta_{T+1} - \theta^*\|^2}{2\eta} + \frac{\eta}{2} \sum_{t=1}^T \|g_t\|^2$ . Dropping the nonnegative term yields the standard bound:

$$R_T \leq \frac{\|\theta_1 - \theta^*\|^2}{2\eta} + \frac{\eta}{2} \sum_{t=1}^T \|g_t\|^2.$$

Assume the gradients are bounded,  $\|g_t\| \leq G$ , and define  $D := \|\theta_1 - \theta^*\|$ . Then  $R_T \leq \frac{D^2}{2\eta} + \frac{\eta}{2} G^2 T$ .

Choosing the learning rate  $\eta = \frac{D}{G\sqrt{T}}$  (i.e.,

$$\eta = O(1/\sqrt{T})) \text{ gives}$$

$$R_T \leq \frac{D^2}{2} \frac{G\sqrt{T}}{D} + \frac{1}{2} \frac{D}{G\sqrt{T}} G^2 T = DG\sqrt{T}.$$

Thus, the average regret satisfies  $R_T/T = O(1/\sqrt{T}) \rightarrow 0$ , showing that projected gradient descent achieves sublinear (no-regret) performance.

#### Gradient Descent with Mahalanobis Distance: Proof Details

We analyze regret for the update  $\theta_{t+1} = \arg \min_{\theta \in \Theta} \|\theta - (\theta_t - \eta_t A^{-1} g_t)\|_A^2$ . **Step 1: Expansion using Mahalanobis norm.** By definition,

$$\|\theta - (\theta_t - \eta_t A^{-1} g_t)\|_A^2 = (\theta - \theta_t + \eta_t A^{-1} g_t)^\top A (\theta - \theta_t + \eta_t A^{-1} g_t) = \|\theta - \theta_t\|_A^2 + 2\eta_t g_t^\top(\theta - \theta_t) + \eta_t^2 \|g_t\|_{A^{-1}}^2.$$

**Step 2: Use optimality of projection.** By definition of  $\theta_{t+1}$  as the minimizer over  $\Theta$ , for any  $\theta^* \in \Theta$ :

$$(\theta^* - \theta_{t+1})^\top A (\theta_t - \eta_t A^{-1} g_t - \theta_{t+1}) \leq 0.$$

**Step 3: Rearrange to bound linearized loss.** This implies  $g_t^\top(\theta_t - \theta^*) \leq$

$$\frac{1}{2\eta_t} (\|\theta_t - \theta^*\|_A^2 - \|\theta_{t+1} - \theta^*\|_A^2) + \frac{\eta_t}{2} \|g_t\|_{A^{-1}}^2.$$

**Step 4: Use convexity of  $\ell_t$ .** Convexity gives

$$\ell_t(\theta_t) - \ell_t(\theta^*) \leq g_t^\top(\theta_t - \theta^*).$$

**Step 5: Sum over  $t = 1, \dots, T$ .**  $R_T =$

$$\sum_{t=1}^T [\ell_t(\theta_t) - \ell_t(\theta^*)] \leq \frac{\|\theta_1 - \theta^*\|_A^2}{2\eta} + \frac{\eta}{2} \sum_{t=1}^T \|g_t\|_{A^{-1}}^2.$$

**Step 6: Optimize  $A$ .** To minimize  $\sum_{t=1}^T \|g_t\|_{A^{-1}}^2$  subject to  $\text{tr}(A) \leq C$ , choose  $A = c \sum_{t=1}^T g_t g_t^\top$  for some constant  $c > 0$ . This aligns the Mahalanobis metric with the observed gradient covariance, giving the minimal regret bound.

**Conclusion:** This derivation shows how adaptive gradient methods (e.g., AdaGrad) naturally arise from choosing the Mahalanobis matrix  $A$  based on past gradients, leading to tighter regret bounds in high-dimensional, sparse, or anisotropic problems.

#### AdaGrad

At time  $t$ , estimate the optimal  $A_t$ . A natural choice is  $A_t = \sum_{\tau=1}^t g_\tau g_\tau^\top$ . For high-dimensional problems use the diagonal approximation  $(A_t)_{jj} = \sum_{\tau=1}^t g_{\tau,j}^2$  and  $(A_t)_{jj'} = 0$  for  $j \neq j'$ . The preconditioned update can be written as

$$\theta_{t+1} \leftarrow \arg \min_{\theta \in \Theta} \|\theta - (\theta_t - \eta \text{diag}(A_t)^{-1} g_t)\|_{\text{diag}(A_t)}^2.$$

#### AdaGrad Algorithm (coordinate form)

Use  $\Theta = \mathbb{R}^p$ . For each coordinate  $j \in [p]$  update  $\theta_{t+1,j} = \theta_{t,j} - \eta_{t,j} g_{t,j}$  with  $\eta_{t,j} = \frac{\eta}{\sqrt{\sum_{\tau=1}^t g_{\tau,j}^2}}$ . Each

parameter thus has its own adaptive learning rate: coordinates with small accumulated squared gradients have larger effective steps and vice versa. AdaGrad remembers all past gradients equally (no forgetting).

#### AdaGrad: Diagonal and Full-matrix forms

Fix  $\eta$  and  $\theta_1 \in \mathcal{K}$ . Let  $G_0 = 0$ . For  $t = 1, \dots, T$  play  $\theta_t$ , observe loss  $f_t(\theta_t)$  and gradient  $\nabla_t = \nabla f_t(\theta_t)$ , and update  $G_t = G_{t-1} + \nabla_t \nabla_t^\top$ . Define the preconditioner  $H_t$  by minimizing  $G_t \cdot H^{-1} + \text{Tr}(H)$  over PSD  $H$ . The diagonal choice gives  $H_t = \text{diag}(G_t)^{1/2}$  and the full-matrix choice gives  $H_t = G_t^{1/2}$ . The (preconditioned) gradient step is  $\hat{\theta}_{t+1} = \theta_t - \eta H_t^{-1} \nabla_t$  and then  $\theta_{t+1} = \arg \min_{\theta \in \mathcal{K}} \|\theta - \hat{\theta}_{t+1}\|_{H_t}$ .

#### Regret of AdaGrad (Diagonal) — corrected proof

Let  $H_1 = \{H \succeq 0 : H = \text{diag}(H), \text{Tr}(H) \leq 1\}$  and define  $D_\infty = \max_{x,y \in \mathcal{K}} \|x - y\|_\infty$ . The standard mirror-descent (local-norm) inequality for each  $t$  states  $\nabla_t^\top(\theta_t - \theta^*) \leq$

$$\frac{1}{2\eta} (\|\theta^* - \theta_t\|_{H_t}^2 - \|\theta^* - \theta_{t+1}\|_{H_t}^2) + \frac{\eta}{2} \|\nabla_t\|_{H_t,*}^2.$$

Summing over  $t = 1, \dots, T$  and using convexity yields  $\text{Reg}_T := \sum_{t=1}^T (f_t(\theta_t) - f_t(\theta^*)) \leq$

$$\frac{1}{2\eta} \|\theta^* - \theta_1\|_{H_1}^2 + \frac{\eta}{2} \sum_{t=1}^T \|\nabla_t\|_{H_t,*}^2.$$

For the diagonal choice  $H_t = \text{diag}(G_t)^{1/2}$  we have

$$\|\nabla_t\|_{H_t,*}^2 = \sum_{j=1}^d \frac{g_{t,j}^2}{\sqrt{G_{t,jj}}} \text{ where } G_{t,jj} = \sum_{\tau=1}^t g_{\tau,j}^2.$$

We use the scalar lemma: for any nonnegative sequence  $a_1, a_2, \dots, a_T$  and  $A_t = \sum_{s=1}^t a_s^2$  (with  $A_0 = 0$ ) it

holds that  $\sum_{t=1}^T \frac{a_t^2}{\sqrt{A_t}} \leq 2\sqrt{A_T}$ . Proof: note

$$\sqrt{A_t} - \sqrt{A_{t-1}} = \frac{a_t^2}{\sqrt{A_t} + \sqrt{A_{t-1}}} \geq \frac{a_t^2}{2\sqrt{A_t}},$$

$$\frac{a_t^2}{\sqrt{A_t}} \leq 2(\sqrt{A_t} - \sqrt{A_{t-1}}); \text{ summing yields the claim.}$$

Applying the lemma coordinate-wise gives

$$\sum_{t=1}^T \frac{g_{t,j}^2}{\sqrt{G_{t,jj}}} \leq 2\sqrt{G_{T,jj}}. \text{ Summing over } j \text{ we obtain}$$

$$S := \sum_{t=1}^T \|\nabla_t\|_{H_t,*}^2 \leq 2 \sum_{j=1}^d \sqrt{G_{T,jj}} =$$

$$2 \text{Tr}(\text{diag}(G_T)^{1/2}).$$

Also  $\|\theta^* - \theta_1\|_{H_1}^2 \leq D_\infty^2 \text{Tr}(H_1) \leq D_\infty^2$ . Therefore

$$\text{Reg}_T \leq \frac{D_\infty^2}{2\eta} + \frac{\eta}{2} S \leq \frac{D_\infty^2}{2\eta} + \eta \text{Tr}(\text{diag}(G_T)^{1/2}).$$

Choosing the concrete (simple) stepsize  $\eta = \sqrt{2} D_\infty$

$$\text{yields } \text{Reg}_T \leq \sqrt{2} D_\infty \text{Tr}(\text{diag}(G_T)^{1/2}) + \frac{D_\infty}{2\sqrt{2}}.$$

Dropping the lower-order additive term gives the commonly cited leading-order form

$$\text{Reg}_T(\text{AdaGrad-Diag}) \lesssim \sqrt{2} D_\infty \text{Tr}(\text{diag}(G_T)^{1/2}).$$

Remark: a tighter (and more common) coordinate-wise bound is  $\text{Reg}_T \leq \sum_{j=1}^d D_j \sqrt{\sum_{t=1}^T g_{t,j}^2}$  where

$$D_j = \sup_{x,y \in \mathcal{K}} |x_j - y_j|; \text{ setting } D_j \leq D_\infty \text{ recovers a}$$

bound with leading term  $D_\infty \text{Tr}(\text{diag}(G_T)^{1/2})$  (without the  $\sqrt{2}$ ), but the  $\sqrt{2}$  form above follows from the simple global- $\eta$  choice shown.

#### Regret of AdaGrad (Full matrix) — sketch

Let  $H_2 = \{H \succeq 0 : \text{Tr}(H) \leq 1\}$  and

$$D_2 = \max_{x,y \in \mathcal{K}} \|x - y\|_2. \text{ The same mirror-descent}$$

$$\text{telescoping gives } \text{Reg}_T \leq \frac{D_2^2}{2\eta} + \frac{\eta}{2} \sum_{t=1}^T \nabla_t^\top H_t^{-1} \nabla_t.$$

For the full-matrix choice  $H_t = G_t^{1/2}$  one can show the matrix analogue  $\sum_{t=1}^T \nabla_t^\top G_t^{-1/2} \nabla_t \leq 2 \text{Tr}(G_T^{1/2})$ .

Combining and taking  $\eta = \sqrt{2} D_2$  gives the leading-order bound

$$\text{Reg}_T(\text{AdaGrad-Full}) \lesssim \sqrt{2} D_2 \text{Tr}(G_T^{1/2}).$$

#### Takeaway

Diagonal AdaGrad is cheap and excels when gradients are coordinate-sparse (small  $\text{Tr}(\text{diag}(G_T)^{1/2})$ ); full-matrix AdaGrad attains stronger adaptivity at higher cost. The proofs above provide the corrected derivation that yields the  $\sqrt{2}$  leading constant under the simple global- $\eta$  choice; alternate choices or per-coordinate step sizes recover the  $D_\infty$ -linear-in-trace form without the  $\sqrt{2}$  constant.

#### Adaptive Gradient Descent with RMSProp

AdaGrad has the issue that step sizes keep decreasing because it maintains a full history with equal weight. RMSProp fixes this by “forgetting” the past gradually. For coordinate  $j \in [p]$ , maintain a decaying second moment estimate  $v_{t,j} = \beta v_{t-1,j} + (1 - \beta) g_{t,j}^2$ . Then update parameters using  $\theta_{t+1,j} = \theta_{t,j} - \eta_{t,j} g_{t,j}$  with step size  $\eta_{t,j} = \frac{\eta}{\sqrt{v_{t,j} + \epsilon}}$ . This gives more weight to recent gradients. Momentum can be combined with this idea.

#### Adam: RMSProp with Momentum

Compute current gradient  $g_t$ . Update biased first moment estimate:  $m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$ . Update biased second moment estimate:

$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$ . Typical choices:  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ . Compute bias-corrected moments:

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}, \quad \hat{v}_t = \frac{v_t}{1 - \beta_2^t}. \text{ Parameter update:}$$

$$\theta_{t+1} = \theta_t - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}}.$$

#### Regret of Adam

Assume gradient bounds  $\|\nabla f(\theta_t)\|_2 \leq G_2, \|\nabla f(\theta_t)\|_\infty \leq G_\infty$ , diameter bounds  $\|\theta_t - \theta_{t'}\|_2 \leq D_2, \|\theta_t - \theta_{t'}\|_\infty \leq D_\infty$ . Weights:  $\beta_1, \beta_2 \in [0, 1], \beta_1^{1/2} \leq 1, \beta_1, \lambda \in (0, 1)$ . Stepsize:  $\alpha_t = \alpha/\sqrt{t}$ . Then (from the Adam paper)  $\text{Reg}_T(\text{Adam}) \leq \frac{D_2}{2\alpha(1-\beta_1)} \sum_{i=1}^d \sqrt{T \hat{v}_{T,i}} + \frac{\alpha(1+\beta_1)G_\infty}{(1-\beta_1)\sqrt{1-\beta_2}(1-\gamma)^2} \sum_{i=1}^d \|\nabla f_{1:T,i}\|_2 + \sum_{i=1}^d \frac{D_\infty^2 G_\infty \sqrt{1-\beta_2}}{2\alpha(1-\beta_1)(1-\lambda)^2}$ , regret is  $O(\sqrt{T})$ .

#### Understanding Regret of Adam

If gradients are sparse (small w.r.t. features), i.e.,  $g_{t,i}$  are small, then  $\sum_i \|\nabla_{1:T,i}\|_2 \ll dG_\infty \sqrt{T}$  and  $\sum_i \sqrt{T \hat{v}_{T,i}} \ll dG_\infty \sqrt{T}$ , so Adam adapts better to sparse settings.

#### Issues with Adam Analysis

Generic adaptive algorithm:  $\nabla_t = \nabla f_t(\theta_t)$ ,  $m_t = \phi_t(g_1, \dots, g_t)$ ,  $V_t = \psi_t(g_1, \dots, g_t)$ ,  $\theta_{t+1} = \Pi_{\mathcal{K}} \sqrt{V_t}(\theta_t - \eta_t m_t / \sqrt{V_t})$ . Regret analysis relies

on  $\Gamma_{t+1} = \frac{\sqrt{V_{t+1}}}{\alpha_{t+1}} - \frac{\sqrt{V_t}}{\alpha_t}$ . For SGD or AdaGrad,  $\Gamma_t \geq 0$  and analysis is straightforward. For Adam,  $\Gamma_t$  can be indefinite, which can break the OCO convergence guarantee: there exists a stochastic convex problem and  $\beta_1, \beta_2 \in [0, 1]$  with  $\beta_1 < \sqrt{\beta_2}$  where  $\text{Reg}_T(\text{Adam})/T \not\rightarrow 0$  as  $T \rightarrow \infty$ .

#### AMSGrad Algorithm

Setup:  $x_1 \in \mathcal{K}$ , step sizes  $\{\alpha_t\}_{t=1}^T$ , momentum weights  $\{\beta_{1t}\}_{t=1}^T$ . Initialize  $m_0 = 0, v_0 = 0, \hat{v}_0 = 0$ . For  $t = 1, \dots, T$ :  $g_t = \nabla f_t(x_t)$   $m_t = \beta_{1t} m_{t-1} + (1 - \beta_{1t}) g_t$   $v_t = \beta_{2t} v_{t-1} + (1 - \beta_{2t}) g_t^2$

$\hat{v}_t = \max(\hat{v}_{t-1}, v_t)$ ,  $\hat{V}_t = \text{diag}(\hat{v}_t)$   
 $\theta_{t+1} = \Pi_{\mathcal{K}, \sqrt{\hat{V}_t}}(\theta_t - \alpha_t m_t / \sqrt{\hat{V}_t})$   
Key change from Adam:  $\hat{v}_t = \max(\hat{v}_{t-1}, v_t)$ .  
Empirically, Adam often works better, but AMSGrad ensures convergence theoretically.

**Regret of AMSGrad**  
Gradient bounds:  $\|\nabla f(\theta_t)\|_\infty \leq G_\infty$ . Diameter bound:  $\|\theta_t - \theta_{t'}\|_\infty \leq D_\infty$ . Weights:  $\beta_1 = \beta_{11}, \beta_{1t} \leq \beta_1, \gamma = \beta_1 / \sqrt{\beta_2} < 1$ . Step size  $\alpha_t = \alpha / \sqrt{t}$ .

Regret bound:  $\text{Reg}_T(\text{AMSGrad}) \leq \frac{D_\infty^2 \sqrt{T}}{\alpha(1-\beta_1)} \sum_{i=1}^m \sqrt{\hat{v}_{T,i}} + \frac{D_\infty^2}{(1-\beta_1)^2} \sum_{t=1}^T \sum_{i=1}^d \frac{\beta_{1t} \sqrt{\hat{v}_{T,i}}}{\alpha_t} + \frac{\alpha \sqrt{1+\log T}}{(1-\beta_1)^2(1-\gamma)\sqrt{1-\beta_2}} \sum_{i=1}^m \|\nabla f_{1:T,i}\|_2$

With  $\beta_{1t} = \beta_1 \lambda^{t-1}, \lambda \in (0, 1)$ , AMSGrad regret

satisfies  $\text{Reg}_T(\text{AMSGrad}) \leq \frac{D_\infty^2 \sqrt{T}}{\alpha(1-\beta_1)} \sum_{i=1}^m \sqrt{\hat{v}_{T,i}} + \frac{\beta_1 D_\infty^2 G_\infty}{(1-\beta_1)^2(1-\lambda)^2} + \frac{\alpha \sqrt{1+\log T}}{(1-\beta_1)^2(1-\gamma)\sqrt{1-\beta_2}} \sum_{i=1}^m \|\nabla f_{1:T,i}\|_2$

This bound can be considerably better than  $O(\sqrt{dT})$

when  $\sum_{i=1}^d \sqrt{\hat{v}_{T,i}} \ll \sqrt{d}$  and  $\sum_{i=1}^d \|\nabla f_{1:T,i}\|_2 \ll \sqrt{dT}$ .

**Revisiting Issues with Adam Analysis**  
Reddi et al. proved that Adam (and RMSProp) does not converge for certain hyperparameters, while Shi et al. showed that RMSProp converges for large enough  $\beta_2$ . Key points: 

- There exists convex problems where RMSProp fails if  $\beta_2$  is not sufficiently large.
- Choosing  $\beta_2$  sufficiently large ensures convergence, but the minimal  $\beta_2$  guaranteeing convergence is problem-dependent.
- Two key messages on  $\beta_2$ :
  - $\beta_2$  must be large enough for convergence.
  - Minimal-convergence  $\beta_2$  depends on the problem instance; there is no universal hyperparameter.

### Constrained Optimization

Consider equality and inequality constrained optimization: minimize  $f(x)$ , subject to  $h_i(x) = 0, i = 1, \dots, m$ ,  $g_j(x) \leq 0, j = 1, \dots, n$ . Domain:  $D = \text{dom}(f) \cap \bigcap_{i=1}^m \text{dom}(h_i) \cap \bigcap_{j=1}^n \text{dom}(g_j)$ .

Lagrangian:  $L(x, \lambda, \nu) = f(x) + \sum_{i=1}^m \lambda_i h_i(x) + \sum_{j=1}^n \nu_j g_j(x)$  with domain  $\text{dom}(L) = D \times \mathbb{R}^m \times \mathbb{R}^n$ , where  $\{\lambda_i\}$  and  $\{\nu_j\}$  are the Lagrange multipliers.

### Lagrange Dual

The Lagrange dual function:

$L^*(\lambda, \nu) = \inf_{x \in D} L(x, \lambda, \nu) =$

$\inf_{x \in D} \left[ f(x) + \sum_{i=1}^m \lambda_i h_i(x) + \sum_{j=1}^n \nu_j g_j(x) \right]$  Let  $p^*$

be the constrained optimum of  $f(x)$ . Then  $L^*$  is concave (even if the original problem is not convex) and provides a lower bound: for  $\nu \geq 0, L^*(\lambda, \nu) \leq p^*$ . The key question: how close is  $\max_{\lambda, \nu} L^*(\lambda, \nu)$  to  $p^*$ ?

### Example

Minimize  $x^\top x$  subject to  $Ax = b$ . Lagrangian:

$L(x, \lambda) = x^\top x + \lambda^\top (Ax - b)$  Dual function:

$\nabla_x L(x, \lambda) = 0 \implies x = -\frac{1}{2} A^\top \lambda$ ,

$L^*(\lambda) = L(-\frac{1}{2} A^\top \lambda, \lambda) = -\frac{1}{4} \lambda^\top A A^\top \lambda - \lambda^\top b$  Hence,  $L^*(\lambda)$  is a concave lower bound on the primal optimum.

### Lagrange Duality and the Conjugate

Consider the constrained problem:

minimize  $f(x)$ , subject to  $Ax = b, Cx \leq d$  The

Lagrangian dual function:

$L(\lambda, \nu) = \inf_x \left[ f(x) + \lambda^\top (Ax - b) + \nu^\top (Cx - d) \right] =$

$\inf_x \left[ f(x) + x^\top (A^\top \lambda + C^\top \nu) - \lambda^\top b - \nu^\top d \right] =$

$-f^*(-A^\top \lambda - C^\top \nu) - \lambda^\top b - \nu^\top d$  Recall the conjugate:

$f^*(z) = \sup_x (x^\top z - f(x)) \implies -f^*(-z) =$

$\inf_x (f(x) + x^\top z)$ .

**Example of Conjugate** If  $f(x) = \sum_{i=1}^n x_i \log x_i$ , then  $f^*(z) = \sum_{i=1}^n \exp(z_i - 1)$ .

### The Lagrange Dual Problem

maximize  $L^*(\lambda, \nu)$ , subject to  $\nu \geq 0$  This gives the best lower bound  $d^*$  to  $p^*$ , the primal optimum. It is a concave optimization problem. Example (linear programming):

minimize  $c^\top x$ , subject to  $Ax = b, x \geq 0 \implies$

dual: maximize  $-b^\top \lambda$ , subject to  $A^\top \lambda + c \geq 0$

### Weak and Strong Duality

- Weak duality:  $d^* \leq p^*$ , always holds. Useful for lower bounds and approximation algorithms.
- Strong duality:  $d^* = p^*$ , does not always hold. If it holds, solving the dual suffices. Constraint qualification: normally true for convex problems. For strict feasibility ( $\exists x \in \text{relint}(D)$  s.t.  $Ax = b, g_j(x) < 0$ ), Slater's condition guarantees strong duality.

### Example: Quadratic Programs

minimize  $x^\top x$ , subject to  $Ax \leq b$  Lagrange dual:  $L^*(\nu) = \inf_x [x^\top x + \nu^\top (Ax - b)] = -\frac{1}{4} \nu^\top A A^\top \nu - b^\top \nu$  Dual problem: maximize  $L^*(\nu)$ , subject to  $\nu \geq 0$ . From Slater's condition,  $p^* = d^*$ .

### Complementary Slackness

If strong duality holds, let  $x^*$  be primal optimum,

$(\lambda^*, \nu^*)$  dual optimum:

$x^*$  minimizes  $L(x, \lambda^*, \nu^*)$ ,  $\nu_j^* g_j(x^*) = 0 \forall j$  That is,

$\nu_j^* > 0 \implies g_j(x^*) = 0$  and  $g_j(x^*) < 0 \implies \nu_j^* = 0$ .

### Karush-Kuhn-Tucker (KKT) Conditions

Necessary (and sufficient for convex problems)

conditions for optimal primal-dual pair  $(\tilde{x}, \tilde{\lambda}, \tilde{\nu})$ :

- Primal feasibility:  $h_i(\tilde{x}) = 0, g_j(\tilde{x}) \leq 0$
- Dual feasibility:  $\tilde{\nu}_j \geq 0$
- Complementary slackness:  $\tilde{\nu}_j g_j(\tilde{x}) = 0$
- Gradient condition:  $\nabla f(\tilde{x}) + \sum_i \tilde{\lambda}_i \nabla h_i(\tilde{x}) + \sum_j \tilde{\nu}_j \nabla g_j(\tilde{x}) = 0$

### Projected Gradient Descent for Equality Constraints

Consider the problem:

minimize  $f(x)$ , subject to  $Ax = b$  Define the

constrained set  $K = \{x : Ax = b\}$ . Projected gradient

descent:  $x_{t+1} = \Pi_K(x_t - \eta \nabla f(x_t))$  The projection

$\Pi_K(y)$  can be computed by solving:

$\min_x \|x - y\|_2^2$  s.t.  $Ax = b$

### Dual Ascent

Lagrangian:  $L(x, \lambda) = f(x) + \lambda^\top (Ax - b)$  Lagrange

dual:  $L^*(\lambda) = \inf_x L(x, \lambda) = -f^*(-A^\top \lambda) - b^\top \lambda$

Dual ascent algorithm:  $\lambda \mapsto \lambda + \eta_t (Ax^+ - b)$ , where

$x^+ = \arg \min_x L(x, \lambda)$ .  $\nabla L^*(\lambda) = Ax^+ - b$ .

Non-differentiable cases use sub-gradient ascent.

### Augmented Lagrangian

$L_\rho(x, \lambda) = f(x) + \lambda^\top (Ax - b) + \frac{\rho}{2} \|Ax - b\|_2^2$  Equivalent to the original constrained problem.

**ADMM (Alternating Direction Method of Multipliers)**

Problem: min  $f(x) + g(z)$ , s.t.  $Ax + Bz = c$

Augmented Lagrangian:  $L_\rho(x, z, \lambda) =$

$f(x) + g(z) + \lambda^\top (Ax + Bz - c) + \frac{\rho}{2} \|Ax + Bz - c\|_2^2$

$x_{t+1} = \arg \min_x L_\rho(x, z_t, \lambda_t)$

ADMM updates:  $z_{t+1} = \arg \min_z L_\rho(x_{t+1}, z, \lambda_t)$

$\lambda_{t+1} = \lambda_t + \rho(Ax_{t+1} + Bz_{t+1} - c)$

### Online Constrained Convex Optimization (OCCO)

Adversary chooses  $f_t(x)$ . Best-in-hindsight loss:

$\min_{x \in \mathcal{X}, z \in \mathcal{Z}, Ax + Bz = c} \sum_{t=1}^T f_t(x) + g(z)$  Online

algorithm picks  $(x_t, z_t)$  before seeing  $f_t$ . Key questions:

Regret bounds	$\eta > 0$		$\eta = 0$	
	$R_1$	$R^c$	$R_2$	$R^c$
general convex	$O(\sqrt{T})$	$O(\sqrt{T})$	$O(\sqrt{T})$	$O(\sqrt{T})$
strongly convex	$O(\log T)$	$O(\log T)$	$O(\log T)$	$O(\log T)$

regret analysis, online ADMM, handling constraints over iterations.

### Online ADMM (OADM)

Augmented Lagrangian at step  $t$ :  $L_\rho^t(x, z, \lambda) =$

$f_t(x) + g(z) + \lambda^\top (Ax + Bz - c) + \frac{\rho}{2} \|Ax + Bz - c\|_2^2$

$x_{t+1} = \arg \min_x L_\rho^t(x, z_t, \lambda_t) + \eta D_R(x, x_t)$

Updates:  $z_{t+1} = \arg \min_z L_\rho^t(x_{t+1}, z, \lambda_t)$

$\lambda_{t+1} = \lambda_t + \rho(Ax_{t+1} + Bz_{t+1} - c)$

Loss revealed as  $f_{t+1}(x_{t+1}) + g(z_{t+1})$ , constraint

violation measured as  $\|Ax_{t+1} + Bz_{t+1} - c\|_2$ .

### Primal and Dual Regret of OADM

Primal regret:  $R(T) =$

$\sum_{t=1}^T (f_t(x_t) + g(z_t)) - \min_{Ax + Bz = c} \sum_{t=1}^T (f_t(x) + g(z))$

Constraint violation / dual regret:

$R_c(T) = \|Ax_{t+1} + Bz_{t+1} - c\|_2^2 + \|Bz_{t+1} - Bz_t\|_2^2$

### Online Mirror Descent in Primal Updates (Bregman ADMM)

Primal updates with Bregman distances:  $x_{t+1} =$

$\arg \min_x f(x) + \lambda_t^\top (Ax + Bz_t - c) + \rho D_R(c - Ax, Bz_t)$ ,

$z_{t+1} =$

$\arg \min_z g(z) + \lambda_t^\top (Ax_{t+1} + Bz - c) + \rho D_R(Bz, c - Ax_{t+1})$ ,

$\lambda_{t+1} = \lambda_t + \rho(Ax_{t+1} + Bz_{t+1} - c)$

### General Bregman ADMM

Add proximal mirror terms for primal updates:

$x_{t+1} = \arg \min_x f(x) + \lambda_t^\top Ax + \rho D_R(c - Ax, Bz_t) +$

$\rho_x D_{R_x}(x, x_t)$ ,  $z_{t+1} = \arg \min_z g(z) + \lambda_t^\top Bz +$

$\rho D_R(Bz, c - Ax_{t+1}) + \rho_z D_{R_z}(z, z_t)$ ,

$\lambda_{t+1} = \lambda_t + \rho(Ax_{t+1} + Bz_{t+1} - c)$

### Linearizing the Functions

Linearize the loss function  $f(x)$  as in OGD/OMD:

$x_{t+1} = \arg \min_x \langle \nabla f(x_t), x - x_t \rangle + \lambda_t^\top Ax + \rho D_R(c -$

$Ax, Bz_t) + \rho_x D_{R_x}(x, x_t)$

Linearize the augmentation  $h(x) = \frac{1}{2} \|Ax + Bz_t - c\|_2^2$ :

$x_{t+1} =$

$\arg \min_x f(x) + \langle \lambda_t + \rho(Ax_t + Bz_t - c), Ax \rangle + \rho_x D_{R_x}(x, x_t)$

Linearize both: mirror descent style:  $x_{t+1} =$

$\arg \min_x \langle \nabla h_t(x), x - x_t \rangle + \rho_x D_{R_x}(x, x_t)$ ,  $h_t(x) =$

$f(x) + \lambda_t^\top Ax + \frac{\rho}{2} \|Ax + Bz_t - c\|_2^2$

### Convergence Bound for Bregman ADMM

Denote  $w = (x, z, \lambda)$ , distances (diameters) for primal

and dual:  $D_1(w^*, w_0)$ ,  $D_2(w^*, w_0)$ . Let averages:

$\bar{x}_T = \frac{1}{T} \sum_{t=1}^T x_t$ ,  $\bar{z}_T = \frac{1}{T} \sum_{t=1}^T z_t$

Under suitable step-size conditions, the convergence

bounds are:  $f(\bar{x}_T) + g(\bar{z}_T) - (f(x^*) + g(z^*)) \leq$

$\frac{D_1(w^*, w_0)}{\sqrt{T}}$ ,  $\|\bar{A}\bar{x}_T + B\bar{z}_T - c\|_2^2 \leq \frac{D_2(w^*, w_0)}{\sqrt{T}}$

### Proof Sketch / Regret Analysis

1. Use convexity of  $f$  and  $g$ , and the Bregman

divergence property:

$f(x_{t+1}) - f(x^*) \leq \langle \nabla f(x_t), x_{t+1} - x^* \rangle$  2. Mirror

descent inequality for Bregman distance  $D_{R_x}$ :

$\langle \nabla h_t(x_t), x_{t+1} - x^* \rangle \leq$

$\frac{1}{\eta} (D_{R_x}(x^*, x_t) - D_{R_x}(x^*, x_{t+1}) - D_{R_x}(x_{t+1}, x_t))$  3.

Sum over  $t = 1$  to  $T$ , use telescoping sums of  $D_{R_x}$  and

$D_{R_z}$ . 4. Bound dual updates via  $\lambda_{t+1} - \lambda^*$  using

squared norm:

$\|\lambda_{t+1} - \lambda^*\|_2^2 \leq \|\lambda_t - \lambda^*\|_2^2 - \rho^2 \|Ax_{t+1} + Bz_{t+1} - c\|_2^2$

5. Combine primal and dual inequalities, divide by  $T$ ,

and use convexity of  $f$  and  $g$  to get the average-iterate

bounds.

These steps yield the stated convergence rates

$O(1/\sqrt{T})$  for both objective suboptimality and

constraint violation. **Stochastic Bandits**

Set of available actions:  $\mathcal{A}$ . Pulling different arms  $a_i$  leads to stochastic rewards  $r_i$ .

At each round  $t = 1, 2, \dots, T$ : 

- Learner selects action

$a_t \in \mathcal{A}$  using history  $H_t = \{(a_\tau, r_\tau)\}_{\tau=1}^{t-1}$ . 

-

Environment generates reward  $r_t \equiv r(a_t) \sim P_{a_t}$ .

Learning objective: maximize expected cumulative

reward:  $\mathbb{E} \left[ \sum_{t=1}^T r_t \right]$  A stochastic bandit is a collection

of distributions  $\nu = (P_a : a \in \mathcal{A})$ , and the expectation is

with respect to  $\nu$ .

### Unstructured vs. Structured Bandits

*Unstructured Bandits*: Playing action  $a$  gives no

information about other actions  $b \neq a$ .

$\mathcal{E} = \times_{a \in \mathcal{A}} \mathcal{M}_a$ ,  $\mathcal{M}_a$  is set of distributions for action  $a$

Example: portfolio selection problem with independent

arms.

*Structured Bandits*: Reward from one arm provides

information about (some) other arms. Examples: 1.

Bernoulli two-armed bandit:  $A = \{1, 2\}$ ,

$\mathcal{E} = \{B(p), B(1-p)\}$ . Pulling one arm reveals info

about the other. 2. Stochastic linear bandits:  $A \subset \mathbb{R}^d$ ,

unknown parameter  $\theta \in \mathbb{R}^d$ . Reward distributions

$\nu_\theta = (N(a^\top \theta, 1) : a \in A)$ . Pulling any arm gives

information about  $\theta$ .

### Regret

Define  $\mu(a) = \mathbb{E}[r(a)]$  and let  $a^* = \arg \max_{a \in \mathcal{A}} \mu(a)$ ,

$\mu^* = \mu(a^*)$ . Regret of policy  $\pi$  over horizon  $T$ :

$\text{Reg}_T(\pi) = \mathbb{E}_\nu \left[ \sum_{t=1}^T (\mu^* - r(a_t)) \right] =$

$T\mu^* - \mathbb{E}_\nu \left[ \sum_{t=1}^T r(a_t) \right]$  Regret depends on  $\nu$  and  $\pi$ ,

often written as  $\text{Reg}(T)$  for brevity.

### “Frequentist” vs. “Bayesian” Regret

*Frequentist regret*: Analyze  $\text{Reg}_T(\pi, \nu)$  for each  $\nu \in \mathcal{E}$ .

- Asymptotic target:  $\forall \nu \in \mathcal{E}, \lim_{n \rightarrow \infty} \frac{\text{Reg}_n(\pi, \nu)}{n} = 0$ .

- Finite-time bounds:
  - o For some  $C > 0, p < 1$ ,

$\text{Reg}_n(\pi, \nu) \leq Cn^p$ . o For some

$C : \mathcal{E} \rightarrow [0, \infty), f : \mathbb{N} \rightarrow [0, \infty)$ ,  $\text{Reg}_T(\pi, \nu) \leq C(\nu)f(n)$ .

*Bayesian regret*: Fix a prior  $Q$  on  $\mathcal{E}$ , and take

$R_T \leq m \sum_{i=1}^k \Delta_i + (T - mk) \sum_{i=1}^k \Delta_i \exp(-m\Delta_i^2/4)$ , where the exponential term comes from Hoeffding's inequality bounding the probability that a suboptimal arm is mistakenly chosen as the best. Assume  $k = 2$  and the sub-optimality gap  $\Delta \leq 1$ .

Choosing  $m = \max\{1, \lfloor \frac{4}{\Delta^2} \log(\frac{n\Delta^2}{4}) \rfloor\}$  gives

$\text{Reg}(T) \lesssim \min\{T\Delta, \frac{\log T}{\Delta^2}\}$  (gap-dependent) and

$\text{Reg}(T) \lesssim \Delta + \sqrt{T}$  (gap-independent). The first term comes from the exploration phase, and the second term comes from the probability of selecting a suboptimal arm in the commit phase, bounded using Hoeffding's inequality.

### Proof of ETC General Regret Bound

Let  $i^* = \arg \max_i \mu_i$  and  $\Delta_i = \mu^* - \mu_i$ . ETC explores each arm  $m$  times, then commits to the arm  $\hat{i} = \arg \max_i \hat{\mu}_i$  with empirical mean  $\hat{\mu}_i = \frac{1}{m} \sum_{\tau: A_\tau = i} X_\tau$ .

**Exploration phase:** each arm pulled  $m$  times, regret  $R_{\text{Explore}} = \sum_{i=1}^k m\Delta_i = m \sum_{i=1}^k \Delta_i$ . **Commit phase:** total regret  $R_{\text{commit}} = \sum_{t=mk+1}^T \mathbb{E}[\mu^* - \hat{\mu}_i] =$

$(T - mk) \sum_{i \neq i^*} \Delta_i \Pr(\hat{i} = i)$ . By Hoeffding's inequality, the probability that a suboptimal arm  $i$  is selected satisfies  $\Pr(\hat{i} = i) = \Pr(\hat{\mu}_i \geq \hat{\mu}_{i^*}) \leq \Pr(\hat{\mu}_i \geq \mu_i + \Delta_i/2) + \Pr(\hat{\mu}_{i^*} \leq \mu^* - \Delta_i/2) \leq 2 \exp(-2m(\Delta_i/2)^2) \lesssim \exp(-m\Delta_i^2/4)$ . Thus, commit-phase regret is bounded by  $R_{\text{commit}} \leq (T - mk) \sum_{i \neq i^*} \Delta_i \exp(-m\Delta_i^2/4) \leq$

$(T - mk) \sum_{i=1}^k \Delta_i \exp(-m\Delta_i^2/4)$ . Combining exploration and commit phases gives the general bound  $R_T \leq m \sum_{i=1}^k \Delta_i + (T - mk) \sum_{i=1}^k \Delta_i \exp(-m\Delta_i^2/4)$ .

**Hoeffding's Inequality** Let  $X_1, \dots, X_n$  be independent, bounded random variables with  $X_i \in [a_i, b_i]$  and  $\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$ . Then for any  $\epsilon > 0$ ,  $\Pr(\bar{X} - \mathbb{E}[\bar{X}] \geq \epsilon) \leq \exp\left(-\frac{2n\epsilon^2}{\sum_{i=1}^n (b_i - a_i)^2}\right)$ ,  $\Pr(|\bar{X} - \mathbb{E}[\bar{X}]| \geq \epsilon) \leq 2 \exp\left(-\frac{2n\epsilon^2}{\sum_{i=1}^n (b_i - a_i)^2}\right)$ . Special case:

$X_i \in [0, 1]$  gives  $\Pr(|\bar{X} - \mathbb{E}[\bar{X}]| \geq \epsilon) \leq 2 \exp(-2n\epsilon^2)$ .

### Randomized ETC: $\epsilon$ -Greedy

This algorithm modifies ETC to explore with probability  $\epsilon_t$  at time  $t$ . Typically,  $\epsilon_t$  decreases over time. In ETC,  $\epsilon_t = 1$  until  $t \leq mK$ , then  $\epsilon_t = 0$ . At round  $t$ : with probability  $\epsilon_t$  explore uniformly by picking an arm at random; with probability  $1 - \epsilon_t$  exploit by choosing  $A_t \in \arg \max_{a \in A} \hat{\mu}_{t-1}(a)$ .

### Regret of $\epsilon$ -Greedy

- If  $\epsilon_t = \epsilon > 0$ , we have  $\lim_{T \rightarrow \infty} \frac{R_T}{T} = \frac{\epsilon}{k} \sum_{i=1}^k \Delta_i$ .  
Let  $\Delta_{\min} = \min\{\Delta_i : \Delta_i > 0\}$ ,  $\epsilon_t = \min\left\{1, \frac{C_k}{t\Delta_{\min}^2}\right\}$ , large  $C > 0$

$$R_T \leq C' \sum_{i=1}^k \left( \Delta_i + \frac{\Delta_i}{\Delta_{\min}^2} \log \max\left\{e, \frac{T\Delta_{\min}^2}{k}\right\} \right)$$

### Regret of $\epsilon$ -Greedy: Proof

Let  $i^* = \arg \max_i \mu_i$  and  $\Delta_i = \mu^* - \mu_i$ . At each step,  $\epsilon_t$ -Greedy chooses a random arm with probability  $\epsilon_t$  and the empirical best arm with probability  $1 - \epsilon_t$ . (0.2em) **Constant  $\epsilon$ :** if  $\epsilon_t = \epsilon > 0$ , the fraction of times a suboptimal arm  $i$  is chosen is  $\epsilon/k + (1 - \epsilon) \Pr(\hat{i}_t = i)$ . As  $t \rightarrow \infty$ , empirical best converges to  $i^*$ , so  $\Pr(\hat{i}_t = i) \rightarrow 0$  and

$\lim_{T \rightarrow \infty} \frac{R_T}{T} = \sum_{i=1}^k \Delta_i \cdot \frac{\epsilon}{k} = \frac{\epsilon}{k} \sum_{i=1}^k \Delta_i$ . **Decaying  $\epsilon_t$ :** let  $\Delta_{\min} = \min\{\Delta_i : \Delta_i > 0\}$  and  $\epsilon_t = \min\left\{1, \frac{C_k}{t\Delta_{\min}^2}\right\}$ . Exploration regret up to time  $T$  is

$$\sum_{i=1}^T \epsilon_t \sum_{i=1}^k \frac{\Delta_i}{k} \leq \sum_{i=1}^k \frac{\Delta_i}{k} \left( C_k / \Delta_{\min}^2 \sum_{t=1}^T \frac{1}{t} \right) \lesssim$$

$\sum_{i=1}^k \frac{\Delta_i}{\Delta_{\min}^2} \log \frac{T\Delta_{\min}^2}{k}$ . Exploitation regret comes from

choosing a suboptimal empirical best, bounded by  $\sum_i \Delta_i$ . Combining gives

$$R_T \leq C' \sum_{i=1}^k \left( \Delta_i + \frac{\Delta_i}{\Delta_{\min}^2} \log \max\left\{e, \frac{T\Delta_{\min}^2}{k}\right\} \right).$$

### Hoeffding Bound

Let  $X_1, \dots, X_s \in [0, 1]$  be i.i.d. with mean  $\mu$  and empirical mean  $\hat{\mu}_s$ . Then for any  $\delta \in (0, 1)$ ,  $\Pr(|\hat{\mu}_s - \mu| \geq \epsilon) \leq 2 \exp(-2s\epsilon^2)$ , or equivalently,  $\epsilon = \sqrt{\frac{2 \log(1/\delta)}{s}}$ . Confidence radius:  $c(s, \delta) = \sqrt{2 \log(1/\delta)/s}$ . Plan: set uncertainty bonus to  $c(s, \delta)$ .

### Upper Confidence Bound (UCB) Algorithm

Maintain counts  $N(a, t - 1)$  and empirical means  $\hat{\mu}(a, t - 1)$  for each action  $a \in \mathcal{A}$ . - Index (confidence level  $\delta$ ):  $\text{UCB}(a, t - 1, \delta) =$

$$\begin{cases} +\infty, & N(a, t - 1) = 0 \\ \hat{\mu}(a, t - 1) + \sqrt{\frac{2 \log(1/\delta)}{N(a, t - 1)}}, & \text{otherwise.} \end{cases} \quad \text{- UCB}$$

Algorithm: At stage  $t$  - Choose

$a_t \in \arg \max_a \text{UCB}(a, t - 1, \delta)$  - Observe reward  $r(a_t)$  - Update  $N(a_t, t), \hat{\mu}(a_t, t)$

**Understanding UCB** - Optimism: With high probability,  $\mu(a) \leq \hat{\mu}(a, t - 1) + \sqrt{2 \log(1/\delta)/N(a, t - 1)}$  - Automatic exploration: Small  $N(a, t - 1) \Rightarrow$  large bonus - More sampling to get better estimate - Self-tuning: Bonus shrinks as  $N(a, t - 1)$  grows - Suboptimal actions are discarded automatically

**Regret of UCB** - Let  $\mu(a)$  be the mean reward -  $\mu^* = \max_a \mu(a)$ , and gap  $\Delta(a) = \mu^* - \mu(a)$  - Gap-dependent (with  $\delta = 1/T^2$ ):

$$\text{Reg}(T) \leq 3 \sum_{a \in A} \Delta(a) + \sum_{a: \Delta(a) > 0} \frac{16 \log T}{\Delta(a)} \quad \text{- Gap-independent:}$$

$$\text{Reg}(f) \leq 8\sqrt{KT} \log T + 3 \sum_{a \in A} \Delta(a)$$

### Regret of UCB: Proof

Let  $\mu(a)$  be the mean reward,  $\mu^* = \max_a \mu(a)$ , and  $\Delta(a) = \mu^* - \mu(a)$ . UCB chooses

$$a_t = \arg \max_a \hat{\mu}(a, t - 1) + \sqrt{2 \log(1/\delta)/N(a, t - 1)}.$$

Decompose regret over arms:

$R_T = \sum_{a: \Delta(a) > 0} \Delta(a) \mathbb{E}[N(a, T)]$ , where  $N(a, T)$  is the number of times suboptimal arm  $a$  is pulled.

By Hoeffding bound, with probability  $1 - \delta$ ,

$$|\hat{\mu}(a, t - 1) - \mu(a)| \leq \sqrt{2 \log(1/\delta)/N(a, t - 1)}.$$

A suboptimal arm is chosen only if either: (i) empirical mean of  $i^*$  underestimates, or (ii) empirical mean of  $a$  overestimates. Both have probability  $\leq \delta$ .

Set  $\delta = 1/T^2$  and sum over  $T$  rounds: expected number of times arm  $a$  is chosen  $\mathbb{E}[N(a, T)] \leq \frac{8 \log T}{\Delta(a)^2} + 3$ . Hence

gap-dependent regret:

$$R_T \leq \sum_{a: \Delta(a) > 0} \Delta(a) \mathbb{E}[N(a, T)] \leq$$

$$\sum_{a: \Delta(a) > 0} \frac{16 \log T}{\Delta(a)} + 3 \sum_{a \in A} \Delta(a).$$

gap-independent bound, replace  $\Delta(a)$  by  $\sqrt{K \log T/T}$

to get  $R_T \leq 8\sqrt{KT} \log T + 3 \sum_{a \in A} \Delta(a)$ . **Adversarial Bandits**

Model: at each stage  $t$ , an adversary chooses reward vector  $r_t \in [0, 1]^A$ . Learner plays  $a_t$  and observes only  $r_t(a_t)$ . Learner samples  $a_t \sim P_t \in \Delta(A)$ .

Randomization is necessary: deterministic policies are easily exploited. Worst-case regret (minimax scale) is  $\tilde{O}(\sqrt{KT})$ .

### Adversarial Bandits with Finite Arms

Actions  $\mathcal{A}$  with  $|\mathcal{A}| = K$ . At round  $t$ , choose a distribution  $P_t$  over  $\mathcal{A}$ , draw  $a_t \sim P_t$ , observe only  $r_t(a_t)$ . A policy  $\pi$  maps histories to  $P_t$ .

### Regret

Against the best fixed action in hindsight:  $\text{Reg}_T(\pi, r) = \max_{a \in A} \sum_{t=1}^T r_t(a) - \mathbb{E} \sum_{t=1}^T r_t(a_t)$ .

Worst-case regret:  $\text{Reg}_T^*(\pi) = \sup_{r_1, \dots, r_T \in [0, 1]^A} \text{Reg}_T(\pi, r)$ , minimax rate  $\Theta(\sqrt{KT})$

up to logs.

### Proof Sketch for Minimax Regret $\tilde{O}(\sqrt{KT})$

Use the Exp3 algorithm: maintain weights  $w_t(a)$  for each arm, update via  $w_{t+1}(a) = w_t(a) \exp(\eta \hat{r}_t(a))$  with unbiased estimator  $\hat{r}_t(a) = r_t(a)/P_t(a) \cdot \mathbf{1}_{\{a_t = a\}}$ . Expected reward:  $\mathbb{E}[r_t(a_t)] = \sum_a P_t(a) r_t(a)$ , unbiasedness of  $\hat{r}_t(a)$  ensures  $\mathbb{E}[\hat{r}_t(a)] = r_t(a)$ . Use standard Hedge analysis: regret against any fixed arm  $a$  is bounded by  $\sum_{t=1}^T r_t(a) - \sum_{t=1}^T \mathbb{E}[r_t(a_t)] \leq \frac{\log K}{\eta} + \eta \sum_{t=1}^T \sum_a P_t(a) \hat{r}_t(a)^2 \leq \frac{\log K}{\eta} + \eta KT$ .

Optimize  $\eta = \sqrt{\frac{\log K}{KT}}$  to get

$$\text{Reg}_T^*(\text{Exp3}) \leq \sqrt{KT \log K} = \tilde{O}(\sqrt{KT}).$$

### Experts and Importance Weighted (IW) Estimators

With  $P_t(i) > 0$ , define unbiased reward estimator for any  $i \in A$ :  $\hat{r}_t(i) = \frac{\mathbf{1}_{\{a_t=i\}} r_t(a_t)}{P_t(i)}$ ,  $\mathbb{E}[\hat{r}_t(i)] = r_t(i)$ .

Loss-based variant ( $\ell_t(i) = 1 - r_t(i)$ ):

$$\hat{\ell}_t(i) = \frac{\mathbf{1}_{\{a_t=i\}}(1 - r_t(a_t))}{P_t(i)}, \quad \bar{r}_t(i) = 1 - \hat{\ell}_t(i).$$

Importance weighting gives an unbiased estimate of rewards of each expert, even if that arm was not played, allowing exponential weights updates as in the full-information setting. The difference from full-information is that only the chosen arm is observed; IW corrects for this partial feedback.

### Exp3 Algorithm

Inputs: horizon  $T$ ,  $K$  arms, learning rate  $\eta > 0$ .

Initialize  $\tilde{S}_{0,i} = 0$  for all  $i$ . For  $t = 1, \dots, T$ :

- $P_t(i) = \frac{\exp(\eta \tilde{S}_{t-1,i})}{\sum_{j \in A} \exp(\eta \tilde{S}_{t-1,j})}$
- Sample  $a_t \sim P_t$ , observe  $r_t(a_t) \in [0, 1]$
- Update loss-based IW estimates:

$$\tilde{S}_{t,i} = \tilde{S}_{t-1,i} + 1 - \frac{\mathbf{1}_{\{a_t=i\}}(1 - r_t(a_t))}{P_t(i)}$$

### Understanding Exp3

Unconditional weighting: larger estimated reward  $\Rightarrow$  larger  $P_t(i)$ . Exploration controlled via  $\eta$ : small  $\eta \Rightarrow$  near-uniform  $P_t$  (more exploration), large  $\eta \Rightarrow$  aggressive exploitation. Variance: importance weights inflate variance for small  $P_t(i)$ .

### Regret of Exp3: Proof Sketch

Let  $\hat{r}_t(i)$  be the IW estimator. Exp3 is Hedge/Exponential Weights on  $\hat{r}_t(i)$ . Standard analysis of Hedge gives, for any fixed arm  $i$ :  $\sum_{t=1}^T \hat{r}_t(i) - \sum_{t=1}^T \sum_j P_t(j) \hat{r}_t(j) \leq \frac{\log K}{\eta} + \eta \sum_{t=1}^T \sum_j P_t(j) \hat{r}_t(j)^2 \leq \frac{\log K}{\eta} + \eta KT$ . Since  $\mathbb{E}[\hat{r}_t(i)] = r_t(i)$ , taking expectations gives expected

regret  $\text{Reg}_T(\pi, r) \leq \frac{\log K}{\eta} + \eta KT$ . Optimize  $\eta = \sqrt{\frac{\log K}{KT}}$  to get  $\text{Reg}_T(\pi, r) \leq 2\sqrt{KT \log K} = \tilde{O}(\sqrt{KT})$ , matching the minimax order up to constants/logs, valid for oblivious or adaptive adversaries.

### Exp3-IX Algorithm

Motivation: Exp3's IW estimator

$\hat{r}_t(i) = \mathbf{1}_{\{a_t = i\}} r_t(a_t) / P_t(i)$  can have huge variance when  $P_t(i)$  is small. Use loss view:

$\ell_t(i) = 1 - r_t(i) \in [0, 1]$ , observe only  $\ell_t(a_t)$ .

Implicit-exploration (IX) estimator:

$$\hat{\ell}_t(i) = \frac{\mathbf{1}_{\{a_t=i\}} \ell_t(a_t)}{P_t(i) + \gamma}, \quad \gamma > 0. \text{ Update \& sampling:}$$

$$L_{t,i} = L_{t-1,i} + \hat{\ell}_t(i), \quad L_{0,i} = 0, \quad P_t(i) = \frac{\exp(-\eta L_{t-1,i})}{\sum_{j \in A} \exp(-\eta L_{t-1,j})}.$$

### Regret of Exp3-IX - Choices for step-sizes

$\eta_1 = \sqrt{\frac{2 \log(K+1)}{KT}}, \quad \eta_2 = \sqrt{\frac{\log(K) + \log(\frac{K+1}{\delta})}{TK}}$  - For any  $\delta \in (0, 1)$ , setting  $\eta = \eta_1, \gamma = \eta/2$ , with probability at least  $1 - \delta$

$$\text{Reg}(T) \lesssim \sqrt{KT \log(K+1)} + \sqrt{\frac{KT}{\log(K+1)}} \log \frac{1}{\delta} + \log \frac{K+1}{\delta}$$

- For any  $\delta \in (0, 1)$ , setting  $\eta = \eta_2, \gamma = \eta/2$ , with probability at least  $1 - \delta$

$$\text{Reg}(T) \lesssim \sqrt{KT \log(1/\delta)} + \log(K+1) + \log \frac{K+1}{\delta}$$

### Regret of Exp3-IX: Rigorous Proof

Define the implicit-exploration (IX) estimator:

$$\hat{\ell}_t(i) = \frac{\mathbf{1}_{\{a_t=i\}} \ell_t(a_t)}{P_t(i) + \gamma}, \quad L_{t,i} = L_{t-1,i} + \hat{\ell}_t(i), \quad P_t(i) = \frac{\exp(-\eta L_{t-1,i})}{\sum_j \exp(-\eta L_{t-1,j})}.$$

**Step 1: Unbiasedness and variance bound**  $\ell_t(i) \in [0, 1]$ , then for any  $i$ :  $\mathbb{E}[\hat{\ell}_t(i) | \mathcal{F}_{t-1}] = \frac{P_t(i) \ell_t(i)}{P_t(i) + \gamma} \leq \ell_t(i)$ , and  $\hat{\ell}_t(i) \leq 1/\gamma$ .

Thus  $\hat{\ell}_t(i)$  is a valid upper bound and has bounded range.

**Step 2: Exponential weights analysis** Using standard Hedge analysis (for losses  $\hat{\ell}_t(i)$ ):

$$\sum_{t=1}^T \sum_j P_t(j) \hat{\ell}_t(j) - \hat{\ell}_t(i) \leq \frac{\log(K+1)}{\eta} + \eta \sum_{t=1}^T \sum_j P_t(j) \hat{\ell}_t(j)^2. \text{ Take expectation conditional on } \mathcal{F}_{t-1}:$$

$$\mathbb{E}[\sum_j P_t(j) \hat{\ell}_t(j) - \hat{\ell}_t(i) | \mathcal{F}_{t-1}] \leq \frac{\log(K+1)}{\eta} + \eta KT.$$

**Step 3: High-probability bound via Freedman inequality** Let  $X_t = \sum_j P_t(j) \hat{\ell}_t(j) - \hat{\ell}_t(i) - \mathbb{E}[\sum_j P_t(j) \hat{\ell}_t(j) - \hat{\ell}_t(i) | \mathcal{F}_{t-1}]$ .

Then  $|X_t| \leq 1/\gamma$ , and conditional variance  $\sigma_t^2 \leq \sum_j P_t(j) \hat{\ell}_t(j)^2 \leq 1/\gamma$ . Freedman inequality: for any  $\delta \in (0, 1)$ ,

$$\sum_{t=1}^T X_t \leq \sqrt{2 \sum_{t=1}^T \sigma_t^2 \log(1/\delta)} + \frac{\log(1/\delta)}{3\gamma} \leq \sqrt{\frac{2T \log(1/\delta)}{\gamma}} + \frac{\log(1/\delta)}{3\gamma}.$$

**Step 4: Combine steps** With probability at least  $1 - \delta$ :  $\sum_{t=1}^T \sum_j P_t(j) \hat{\ell}_t(j) - \hat{\ell}_t(i) \leq$

$$\frac{\log(K+1)}{\eta} + \eta KT + \sqrt{\frac{2T \log(1/\delta)}{\gamma}} + \frac{\log(1/\delta)}{3\gamma}.$$

**Step 5: Translate to regret** Since  $\hat{\ell}_t(i) \geq \ell_t(i)$  in expectation, we get for losses  $\ell_t(i)$ :

$$\text{Reg}(T) = \sum_{t=1}^T \sum_j P_t(j) \ell_t(j) - \ell_t(i) \lesssim$$

$$\frac{\log(K+1)}{\eta} + \eta KT + \sqrt{\frac{T \log(1/\delta)}{\gamma}} + \frac{\log(1/\delta)}{\gamma}.$$

**Step 6: Choose parameters** - Choice 1:

$$\eta = \eta_1 = \sqrt{\frac{2 \log(K+1)}{KT}}, \gamma = \eta/2 \rightarrow \text{Reg}(T) \lesssim \sqrt{KT \log(K+1)} + \sqrt{\frac{KT}{\log(K+1)}} \log(1/\delta) + \log \frac{K+1}{\delta}.$$

Choice 2:  $\eta = \eta_2 = \sqrt{\frac{\log K + \log((K+1)/\delta)}{TK}}, \gamma = \eta/2 \rightarrow$

$$\text{Reg}(T) \lesssim \sqrt{KT \log(1/\delta)} + \log(K+1) + \log \frac{K+1}{\delta}.$$

**Conclusion:** Implicit-exploration estimator controls variance of importance-weighted losses; exponential weights analysis plus Freedman inequality gives high-probability regret bound for Exp3-IX.

### Contextual Bandits (CBs) Actions $\mathcal{A}$ with $k = |\mathcal{A}|$ .

At round  $t$ , context  $c_t \in \mathcal{C}$  is observed and reward vector  $r_t \in [0, 1]^k$  is fixed but unknown. The protocol: observe  $c_t$ , choose a distribution  $P_t \in \Delta(\mathcal{A})$ , sample  $A_t \sim P_t$  and observe reward  $r_t(A_t)$ . The benchmark is the best context-dependent policy in hindsight:  $\text{Reg}(T) = \mathbb{E}[\max_{\pi: \mathcal{C} \rightarrow \mathcal{A}} \sum_{t=1}^T r_t(\pi(c_t)) - r_t(A_t)]$ , which is richer than the best single-arm benchmark.

### One Multi-Armed Bandit (OMAB) per Context

For finite  $\mathcal{C}$ , run an independent Exp3 instance for each  $c \in \mathcal{C}$ . When  $c_t = c$ , choose  $P_t^c$  from Exp3 for context  $c$ , play  $A_t \sim P_t^c$  and update  $P_t^c$ . Do not update  $P_t^{c'}$  for  $c' \neq c$ . The regret decomposes as  $\text{Reg}(T) = \sum_{c \in \mathcal{C}} \text{Reg}_c(T)$  where  $\text{Reg}_c(T)$  is the regret of Exp3 on rounds with context  $c$ .

**Regret of OMAB** Let  $N_c = \sum_{t=1}^T \mathbf{1}_{\{c_t = c\}}$  be the number of rounds with context  $c$ . Using anytime-Exp3 for each context gives  $\text{Reg}_c(T) \leq 2\sqrt{k N_c} \log k$ .

Summing over contexts and applying Cauchy–Schwarz yields  $\text{Reg}(T) \leq 2\sqrt{k \log k} \sum_{c=1}^C \sqrt{N_c} \leq 2\sqrt{TkC \log k}$ .

**Comparison with Exp3** Single context:

$\text{Reg}(T) \leq 2\sqrt{Tk \log k}$  versus the best single action. All contexts observed equally often:

$\text{Reg}(T) \leq 2\sqrt{Tk|\mathcal{C}| \log k}$  versus best action per context. Context-dependent benchmark is harder to learn, incurs extra  $|\mathcal{C}|$  factor.

**Bandits with Expert Advice** At step  $t$ , reward vector  $r_t \in [0, 1]^k$  and  $M$  experts provide recommendations  $E_t \in [0, 1]^{M \times k}$ . Each row  $E_t^{(m)}$  is a distribution over  $\Delta(\mathcal{A})$  corresponding to expert  $m$ 's advice. The learner chooses  $P_t \in \Delta(\mathcal{A})$ , samples  $A_t \sim P_t$  and observes  $r_t(A_t)$ . The regret relative to the best expert is

$\text{Reg}(T) = \mathbb{E}[\max_{m \in [M]} \sum_{t=1}^T \langle E_t^{(m)}, r_t \rangle - \sum_{t=1}^T r_t(A_t)]$ .

**Exp4 Algorithm** Exp4 maintains a probability distribution  $Q_t$  over  $M$  experts. At each round  $t$ , an expert  $M_t$  is sampled from  $Q_t$ , and the learner follows the expert's advice  $E_t^{(M_t)}$  to select  $A_t \sim P_t = Q_t E_t$ . The learner observes the reward  $r_t(A_t)$  and estimates the rewards for all actions using

$\hat{r}_{t,i} = \frac{1\{A_t=i\}}{r_{t,i} + \gamma} (1 - r_t(A_t))$ . These estimates are

propagated to the experts:  $\tilde{r}_t^{(m)} = \langle E_t^{(m)}, \hat{r}_t \rangle$ . Expert weights are updated with exponential weighting:

$$Q_{t+1,i} = \frac{Q_{t,i} \exp(\eta \tilde{r}_t^{(i)})}{\sum_{j=1}^M Q_{t,j} \exp(\eta \tilde{r}_t^{(j)})}.$$

**Exp4 Implementation Notes** Inputs:

$T, k, M, \eta > 0, \gamma \geq 0$ , initialize  $Q_1 = \text{Unif}([M])$ . Each round receives expert advice  $E_t$ , computes  $P_t = Q_t E_t$ , samples  $A_t \sim P_t$ , observes  $r_t(A_t)$ , estimates  $\hat{r}_t$ , propagates to experts, and updates  $Q_{t+1}$  via exponential weighting.  $\gamma > 0$  controls variance when some  $P_{t,i}$  are small. Memory is  $O(M)$ , per-round computation  $O(M+k)$  via two-stage sampling. Linear dependence on the number of experts or policies.

**Regret of Exp4** - Base bound ( $\gamma = 0$ ) : with step-size  $\eta = \sqrt{2 \log M / (Tk)}$ , expected regret

$\text{Reg}(T) \leq \sqrt{2Tk \log M}$  - Disagreement-adaptive (anytime): With  $\eta_t = \sqrt{\log M / E_t^*}$  where  $E_t^* = \sum_{s=1}^t \sum_{i=1}^k \max_{m \in [M]} E_{m,i}^{(s)} \text{Reg}(T) \leq C \sqrt{E_T^* \log M}$

**Regret of Exp4: Proof** Let  $Q_t$  be the distribution over experts at round  $t$  and  $P_t = Q_t E_t$  the induced distribution over actions. Define the importance-weighted reward estimates

$\hat{r}_{t,i} = r_t(i) / P_{t,i} \cdot 1\{A_t = i\}$ . Then  $\mathbb{E}[\hat{r}_{t,i} \mid Q_t] = r_t(i)$ . Let  $\tilde{r}_t^{(m)} = E_t^{(m)} r_t$  be the estimated reward of expert  $m$ .

Using the standard Hedge analysis with exponential weights, the regret relative to any expert  $m$  satisfies

$$\sum_{t=1}^T \tilde{r}_t^{(m)} - \sum_{t=1}^T \mathbb{E}_{i \sim P_t} [\tilde{r}_{t,i}] \leq \frac{\log M}{\eta} + \frac{\eta}{2} \sum_{t=1}^T \sum_{i=1}^k P_{t,i} (\hat{r}_{t,i})^2.$$

Taking expectation over the learner's randomness and using  $\hat{r}_{t,i} \leq 1/P_{t,i}$  gives  $\mathbb{E}[\text{Reg}(T)] \leq \frac{\log M}{\eta} + \frac{\eta}{2} Tk$ . Optimizing

$\eta = \sqrt{2 \log M / (Tk)}$  yields  $\mathbb{E}[\text{Reg}(T)] \leq \sqrt{2Tk \log M}$ . For the disagreement-adaptive anytime variant, let

$E_t^* = \sum_{s=1}^t \sum_{i=1}^k \max_{m \in [M]} E_{m,i}^{(s)}$ . Using a step-size  $\eta_t = \sqrt{\log M / E_t^*}$  in the same Hedge analysis gives

$\text{Reg}(T) \leq C \sqrt{E_T^* \log M}$ , where the bound adapts to the effective number of "active" actions chosen by the experts, reducing regret when many probabilities are small.

**Stochastic Contextual Bandits** At round  $t$ , observe context  $c_t \in \mathcal{C}$ , choose  $A_t \in [k]$ , and receive reward  $r_t(A_t) = r(c_t, A_t) + \eta_t$  where  $\eta_t$  is conditionally

1-subgaussian. The stochastic CB benchmark regret is  $\text{Reg}(T) = \mathbb{E}[\sum_{t=1}^T \max_{a \in [k]} r(c_t, a) - r_t(A_t)]$ . This benchmark is meaningful when actions do not strongly affect future contexts.

**Realizability** Assume a feature map  $\psi : \mathcal{C} \times [k] \rightarrow \mathbb{R}^d$  and unknown  $\theta^* \in \mathbb{R}^d$  such that  $r(c, a) = \langle \theta^*, \psi(c, a) \rangle$  for all  $(c, a)$ . Smoothness follows from linearity:  $|r(c, a) - r(c', a')| \leq \|\theta^*\| \cdot \|\psi(c, a) - \psi(c', a')\|$ .

**Stochastic Linear Bandits** At round  $t$ , decision set  $\mathcal{A}_t \subset \mathbb{R}^d$ , pick  $A_t \in \mathcal{A}_t$ , observe  $r_t(A_t) = \langle \theta^*, A_t \rangle + \eta_t$ ,  $\eta_t$  conditionally 1-subgaussian. The random or pseudo-regret is

$\text{Reg}_T = \sum_{t=1}^T \max_{a \in \mathcal{A}_t} \langle \theta^*, a \rangle - \langle \theta^*, A_t \rangle$ , and expected regret  $\text{Reg}(T) = \mathbb{E}[\text{Reg}_T]$ . Special cases include finite-armed bandits ( $\mathcal{A}_t = \{e_1, \dots, e_d\}$ ), contextual linear bandits ( $\mathcal{A}_t = \{\psi(c_t, i) : i \in [k]\}$ ), and combinatorial action sets.

**Subgaussian Random Variable** A random variable  $X$  is  $\sigma$ -subgaussian if for all  $\lambda \in \mathbb{R}$ ,  $\mathbb{E}[\exp(\lambda(X - \mathbb{E}[X]))] \leq \exp(\lambda^2 \sigma^2 / 2)$ . Equivalently,  $X$  has tails bounded like a Gaussian:  $\mathbb{P}(|X - \mathbb{E}[X]| \geq t) \leq 2 \exp(-t^2 / (2\sigma^2))$  for all  $t > 0$ .

Subgaussianity implies that the variance proxy  $\sigma^2$  controls concentration even if  $X$  is not truly Gaussian.

**Parameter Estimation and Confidence Set**

Regularized least squares (ridge):

$$\hat{\theta}_t = \arg \min_{\theta \in \mathbb{R}^d} \sum_{s=1}^t (r_s(A_s) - \langle \theta, A_s \rangle)^2 + \lambda \|\theta\|^2.$$

Closed form with  $V_t = \lambda I + \sum_{s=1}^t A_s A_s^\top$  is  $\hat{\theta}_t = V_t^{-1} \sum_{s=1}^t A_s r_s(A_s)$ . Ellipsoidal confidence set for suitable  $\beta_t$ :  $C_t = \{\theta \in \mathbb{R}^d : \|\theta - \hat{\theta}_{t-1}\|_{V_{t-1}} \leq \beta_t\}$  satisfies  $\theta^* \in C_t$  for all  $t$  with probability  $\geq 1 - \delta$ .

**LinUCB Algorithm** Inputs:  $\lambda > 0$ , schedule  $\beta_t$ , horizon  $T$ . Initialize  $V_0 = \lambda I$ ,  $b_0 = 0$ ,  $\hat{\theta}_0 = 0$ . For  $t = 1, \dots, T$ : compute UCB score

$$UCB_t(a) = \max_{\theta \in C_t} \langle \theta, a \rangle = \langle \hat{\theta}_{t-1}, a \rangle + \sqrt{\beta_t} \|a\|_{V_{t-1}^{-1}};$$

play  $A_t \in \arg \max_{a \in \mathcal{A}_t} UCB_t(a)$ ; observe  $r_t(A_t)$ ; update  $V_t = V_{t-1} + A_t A_t^\top$ ,  $b_t = b_{t-1} + A_t r_t(A_t)$ ,  $\hat{\theta}_t = V_t^{-1} b_t$ .

**Understanding LinUCB** Optimism: if  $\theta^* \in C_t$ , then  $\max_a \langle \theta^*, a \rangle \leq \max_a UCB_t(a)$ . Automatic exploration arises because larger  $\|a\|_{V_{t-1}^{-1}}$  implies sampling in

uncertain directions. Self-tuning: as  $V_t$  grows,  $\|a\|_{V_{t-1}^{-1}}$

shrinks, reducing bonus and increasing exploitation. Known as LinUCB, OFUL, LinRel.

**Ridge Regression Revisited** For fixed  $a$ , Hoeffding implies  $\Pr[|\langle \hat{\theta}_t - \theta^*, a \rangle| \geq \sqrt{2\|a\|_{V_t^{-1}} \log(1/\delta)}] \leq \delta$ .

Action selection introduces dependence; vector-valued martingale analysis gives  $\|\hat{\theta}_t - \theta^*\|_{V_t} \leq$

$$\sqrt{\lambda} \|\theta^*\|_2 + \sqrt{2 \log(1/\delta) + \log(\det V_t(\lambda) / \lambda^d)}.$$

These justify ellipsoids  $C_t$  and UCB bonus  $\sqrt{\beta_t} \|a\|_{V_{t-1}^{-1}}$ .

**Ridge Regression and Confidence Ellipsoid Proof** Consider the ridge estimator  $\hat{\theta}_t = V_t^{-1} \sum_{s=1}^t A_s r_s(A_s)$  with  $V_t = \lambda I + \sum_{s=1}^t A_s A_s^\top$  and

$r_s(A_s) = \langle \theta^*, A_s \rangle + \eta_s$ , where  $\eta_s$  is conditionally 1-subgaussian. Then

$\hat{\theta}_t - \theta^* = V_t^{-1} \sum_{s=1}^t A_s \eta_s - \lambda V_t^{-1} \theta^*$ . For fixed  $a \in \mathbb{R}^d$ , the error along  $a$  is

$\langle \hat{\theta}_t - \theta^*, a \rangle = \langle V_t^{-1} \sum_{s=1}^t A_s \eta_s, a \rangle - \lambda \langle V_t^{-1} \theta^*, a \rangle$ . By the property of vector-valued martingales and the self-normalized concentration inequality

(Abbasi-Yadkori et al. 2011), with probability at least  $1 - \delta$ ,  $|\langle \hat{\theta}_t - \theta^*, a \rangle| \leq$

$$\|a\|_{V_t^{-1}} (\sqrt{\lambda} \|\theta^*\| + \sqrt{2 \log(\det(V_t)^{1/2} \det(\lambda I)^{-1/2} / \delta)}).$$

Equivalently, the confidence ellipsoid

$$C_t = \{\theta : \|\theta - \hat{\theta}_{t-1}\|_{V_{t-1}} \leq \beta_t\} \text{ with}$$

$\beta_t = \sqrt{\lambda} \|\theta^*\| + \sqrt{2 \log(\det(V_t)^{1/2} \det(\lambda I)^{-1/2} / \delta)}$  contains  $\theta^*$  with probability at least  $1 - \delta$ . The UCB bonus  $\beta_t \|a\|_{V_{t-1}^{-1}}$  then directly follows from this bound, justifying optimism in LinUCB:

$\langle \theta^*, a \rangle \leq \langle \hat{\theta}_{t-1}, a \rangle + \beta_t \|a\|_{V_{t-1}^{-1}}$  for all  $a$ .

Regret of LinUCB - Assumptions: -  $(\beta_t)$  nondecreasing,  $\|a\|_2 \leq L$  for all  $a \in \bigcup_{t=1}^T \mathcal{A}_t$ . - Bounded gaps:  $\sup_{a, b \in \mathcal{A}_t} \langle \theta_*, a - b \rangle \leq 1$ . - With prob.  $\geq 1 - \delta$ ,  $\theta_* \in C_t$

for all  $t$ . - Theorem (high-probability regret):  $\text{Reg}_T \leq$

$$\sqrt{8dT\beta_T \log\left(\frac{\det V_T}{\det V_0}\right)} \leq \sqrt{8dT\beta_T \log\left(\frac{d\lambda + TL^2}{d\lambda}\right)}$$

- A valid choice for  $\beta_T$  :

$$\sqrt{\beta_T} = \sqrt{\lambda} \|\theta^*\| + \sqrt{2 \log \frac{1}{\delta}} + \sqrt{d \log\left(\frac{d\lambda + TL^2}{d\lambda}\right)}$$

**Regret of LinUCB Proof** Assume  $\theta^* \in C_t$  w.p.  $\geq 1 - \delta$ . Let  $A_t$  be the action selected by LinUCB:

$A_t = \arg \max_{a \in \mathcal{A}_t} \langle \hat{\theta}_{t-1}, a \rangle + \beta_t \|a\|_{V_{t-1}^{-1}}$ . Then for any

$a \in \mathcal{A}_t$ , by optimism  $\langle \theta^*, a \rangle \leq \langle \hat{\theta}_{t-1}, a \rangle + \beta_t \|a\|_{V_{t-1}^{-1}}$ .

In particular, for the optimal action

$A_t^* = \arg \max_{a \in \mathcal{A}_t} \langle \theta^*, a \rangle$ , we have

$\langle \theta^*, A_t^* \rangle - \langle \theta^*, A_t \rangle \leq$

$$\langle \hat{\theta}_{t-1}, A_t \rangle + \beta_t \|A_t\|_{V_{t-1}^{-1}} - \langle \theta^*, A_t \rangle \leq 2\beta_t \|A_t\|_{V_{t-1}^{-1}}.$$

Summing over  $t = 1, \dots, T$  gives

$$\text{Reg}_T \leq 2 \sum_{t=1}^T \beta_t \|A_t\|_{V_{t-1}^{-1}}.$$

Using the elliptical potential lemma,

$$\sum_{t=1}^T \|A_t\|_{V_{t-1}^{-1}}^2 \leq 2 \log(\det V_T / \det V_0).$$

By Cauchy-Schwarz,  $\sum_{t=1}^T \|A_t\|_{V_{t-1}^{-1}} \leq$

$$\sqrt{T \sum_{t=1}^T \|A_t\|_{V_{t-1}^{-1}}^2} \leq \sqrt{2T \log(\det V_T / \det V_0)}.$$

Combining,  $\text{Reg}_T \leq \sqrt{8T\beta_T \log(\det V_T / \det V_0)}$ . Using  $\det V_T \leq (d\lambda + TL^2)^d$  and  $\det V_0 = \lambda^d$ , we get

$\text{Reg}_T \leq \sqrt{8dT\beta_T \log((d\lambda + TL^2)/(d\lambda))}$ . Finally,  $\sqrt{\beta_T} =$

$$\sqrt{\lambda} \|\theta^*\| + \sqrt{2 \log(1/\delta)} + \sqrt{d \log((d\lambda + TL^2)/(d\lambda))}$$

gives a valid high-probability choice.

**Motivation for Optimal Design** Consider least squares regression  $r_t = \langle \theta^*, a_t \rangle + \epsilon_t$  with 1-subgaussian noise and estimator  $\hat{\theta}_t = V_t^{-1} \sum_{\tau=1}^t a_\tau r_\tau$ ,

$V_t = \sum_{\tau=1}^t a_\tau a_\tau^\top$ . Confidence bound:

$$\Pr[|\langle \hat{\theta}_t - \theta^*, a \rangle| \leq \sqrt{2\|a\|_{V_t^{-1}} \log(1/\delta)}] \geq 1 - \delta.$$

Goal: find shortest sequence of actions  $a_t$  such that  $\|a\|_{V_t^{-1}}$

is below desired  $\epsilon$ . Exact solution is integer program, but good approximations exist.

**Optimal Design** Fixed action set  $A \subset \mathbb{R}^d$ . For a design  $\pi \in \Delta(A)$ , define  $V(\pi) = \sum_{a \in A} \pi(a) a a^\top$ ,

$g(\pi) = \max_{a \in A} \|a\|_{V(\pi)^{-1}}^2$ . G-optimal design:

$\pi^* = \arg \min_\pi g(\pi)$ . D-optimal design:

$\pi^* = \arg \max_\pi \log \det V(\pi)$ . Sampling plan for target

$\epsilon, \delta$ :  $n_a = \lceil 2\pi(a)g(\pi)\epsilon^{-2} \log(1/\delta) \rceil$ ,

$V = \sum_a n_a a a^\top \geq 2g(\pi)\epsilon^{-2} \log(1/\delta) V(\pi)$ . Guarantees:

$|\langle \hat{\theta} - \theta^*, a \rangle| \leq \epsilon$  w.p.  $\geq 1 - \delta$  for all  $a \in A$ . Total samples  $n = \sum_a n_a \leq |\text{supp}(\pi)| + 2g(\pi)\epsilon^{-2} \log(1/\delta)$ .

**Understanding Optimal Design** Kiefer–Wolfowitz equivalence (compact  $A$ ,  $\text{span}(A) = \mathbb{R}^d$ ):

$\pi^* = \arg \min_\pi g(\pi) = \arg \max_\pi \log \det V(\pi)$ , and

$g(\pi^*) = d$ . Core set:  $|\text{supp}(\pi^*)| \leq d(d+1)/2$ . Gradient of concave  $f(\pi) = \log \det V(\pi)$  satisfies

$(\nabla f(\pi))_a = \|a\|_{V(\pi)^{-1}}^2$  and

$$\sum_{a \in A} \pi(a) \|a\|_{V(\pi)^{-1}}^2 = \text{tr}(V(\pi) V(\pi)^{-1}) = d.$$

Geometry: D-optimal design corresponds to

minimum-volume centered ellipsoid

$$E = \{x \in \mathbb{R}^d : \|x\|_{V(\pi)^{-1}}^2 \leq d\} \text{ containing } A.$$

**Stochastic Linear Bandits with Finite Arms** Fixed

action set  $A \subset \mathbb{R}^d$ ,  $|A| = k$ . Rewards are linear:

$r_t(A_t) = \langle \theta^*, A_t \rangle + \eta_t$ , with  $\eta_t$  1-subGaussian:

$\mathbb{E}[\exp(\lambda \eta_t) \mid \mathcal{F}_{t-1}] \leq \exp(\lambda^2 / 2)$ . Suboptimality gaps satisfy  $\Delta_a = \max_{b \in A} \langle \theta^*, b - a \rangle \leq 1$  for all  $a \in A$ .

**Proof Sketch** Let  $V_t = \sum_{s=1}^t A_s A_s^\top + \lambda I$  and define the regularized least-squares estimate

$\hat{\theta}_t = V_t^{-1} \sum_{s=1}^t A_s r_s$ . By standard self-normalized martingale concentration, with probability at least  $1 - \delta$  it holds that

$$\|\hat{\theta}_t - \theta^*\|_{V_t} \leq \sqrt{2 \log \frac{\det(V_t)^{1/2} \det(\lambda I)^{-1/2}}{\delta}}.$$

Then for any  $a \in A$ ,  $|\langle \hat{\theta}_t - \theta^*, a \rangle| \leq \|a\|_{V_t^{-1}} \|\hat{\theta}_t - \theta^*\|_{V_t}$ . By

controlling  $\|a\|_{V_{t-1}}$  via optimal experimental design, we

can guarantee accurate estimation of  $\langle \theta^*, a \rangle$  for all  $a \in A$ .

**Regret with Phased Elimination** Algorithm

(sketch) for phases  $\ell = 1, 2, \dots$  with  $\epsilon_\ell = 2^{-\ell}$ : compute G-optimal design  $\pi_\ell$  on surviving set  $A_\ell$  (support  $\leq d(d+1)/2$ ), sample each  $a \in A_\ell$  exactly

$T_\ell(a) = \lceil 2d\pi_\ell(a)\epsilon_\ell^{-2} \log(k\ell(\ell+1)/\delta) \rceil$  times, compute

$\hat{\theta}_\ell$  using  $V_\ell = \sum_a T_\ell(a) a a^\top$ , eliminate  $a$  if

$\max_{b \in A_\ell} \langle \hat{\theta}_\ell, b - a \rangle > 2\epsilon_\ell$ , set  $A_{\ell+1}$  to surviving actions.

High-probability regret:  $\text{Reg}(T) \leq C \sqrt{Td \log(k \log T)}$  for universal constant  $C > 0$ .

**Proof Sketch** By construction,  $V_\ell = \sum_a T_\ell(a) a a^\top$

guarantees  $\|a\|_{V_\ell^{-1}}^2 \leq \epsilon_\ell^2 / (2 \log(k\ell(\ell+1)/\delta))$  for all

$a \in A_\ell$ . Then with high probability  $|\langle \hat{\theta}_\ell - \theta^*, a \rangle| \leq \epsilon_\ell$  for all surviving actions. Therefore, any action  $a$  with true suboptimality  $\Delta_a > 4\epsilon_\ell$  is eliminated. Summing

over phases and using  $\sum_\ell |A_\ell| T_\ell(a) \leq O(d \log(k \log T) / \epsilon_\ell^2)$  gives

high-probability regret  $\text{Reg}(T) \leq C \sqrt{Td \log(k \log T)}$  for universal constant  $C > 0$ .

**Bayesian Bandit Environment**

$k$ -armed stochastic bandit: For environment  $\nu$  Arm  $i$  has reward distribution  $P_{\nu,i}$  on  $[0, 1]$  Arm  $i$  mean reward is  $\mu_i(\nu)$

A Bayesian bandit environment is  $(\mathcal{E}, \mathcal{G}, Q, P)$   $(\mathcal{E}, \mathcal{G})$ : measurable space of environments  $\nu$

$P = (P_{\nu,i} : \nu \in \mathcal{E}, i \in [k])$ ,  $P_{\nu,i}$  is the reward distribution of arm  $i$  in environment  $\nu$   $Q$ : prior over environments  $\nu$  Posterior  $Q(\cdot | H_{t-1})$  after history  $H_{t-1}$

In round  $t$ : Learner observes history  $H_{t-1}$  Chooses  $A_t \in [k]$ , based on policy  $\pi$  Receives reward

$r_t = r_t(A_t) \in [0, 1]$  Reward model: environment  $\nu \sim Q$ , reward  $r(A) \sim P_{\nu,A}$

**Bayesian Regret**

Consider the  $k$ -armed stochastic bandit setting Expected arm reward probabilities  $\mu_i \in [0, 1]$ ,  $i \in [n]$  True distribution of each arm is  $P_{\nu,i}$ , with expectation

$\mu_i$  Regret is defined as  $\text{Reg}_T(\pi, \nu) = T\mu^* - \mathbb{E} \sum_{t=1}^T X_t$  Given a Bayesian bandit environment, the Bayesian regret  $B\text{Reg}_T(\pi, Q) = \int \text{Reg}_T(\pi, \nu) dQ(\nu)$

Beta-Bernoulli example: Arm rewards are drawn from Bernoulli distributions  $\mu_i \in [0, 1]$  Beta distribution prior over the Bernoulli parameters

For Bernoulli  $k$ -armed bandits:  $\sup B\text{Reg}_T^*(Q) = \Theta(\sqrt{kT})$  where

$B\text{Reg}_T^*(Q) = \inf_\pi B\text{Reg}_T(\pi, Q)$

$$\text{Asymptotics for fixed } Q: \limsup_{T \rightarrow \infty} \frac{B\text{Reg}_T^*(Q)}{T^{1/2}} = 0$$

$$\text{but } \exists Q \forall \epsilon > 0 : \liminf_{T \rightarrow \infty} \frac{B\text{Reg}_T^*(Q)}{T^{1/2-\epsilon}} = \infty$$

### Thompson Sampling for Finite-Armed Bandits

Prior  $Q$  over environments  $\nu \in E$ , distribution  $P_{\nu,i}$ ,  $\nu \in E$ ,  $i \in [k]$ . For environment  $\nu$ ,  $\mu_i(\nu) = \int x dP_{\nu,i}(x)$ . At stage  $t$ , sample an environment  $\nu_t \sim Q(\cdot | H_{t-1})$  and play  $A_t = \arg \max_i \mu_i(\nu_t)$ . Equivalent perspective: sample a probability vector over actions from the posterior predictions and choose the action with the largest sampled mean. Pseudocode: for each round  $t$ , sample  $\nu_t \sim Q(\cdot | H_{t-1})$ , play  $A_t = \arg \max_i \mu_i(\nu_t)$ , observe  $r_t$ , update posterior  $Q(\cdot | H_t)$  using Bayes rule. This determines the Thompson Sampling (TS) policy  $\pi$ . **Bayesian Regret of Thompson Sampling** Assume a  $k$ -armed Bayesian bandit environment  $(E, B(E), Q, P)$  with centered rewards that are 1-subGaussian and means in  $[0, 1]$ . Then the Thompson Sampling policy  $\pi$  satisfies  $B\text{Reg}_T(\pi, Q) \leq C\sqrt{kT \log T}$  for a universal constant  $C > 0$ .

### Frequentist Perspective: Follow the Perturbed Leader

View TS as follow-the-perturbed-leader (FTPL). Input: cumulative distribution functions  $F_i(1), i \in [k]$ . In round  $t$ , for each  $i \in [k]$ , sample a score  $\theta_i(t) \sim F_i(t)$ , play  $A_t = \arg \max_i \theta_i(t)$ , observe reward  $r_t(A_t)$ , and update only the played arm's distribution:  $F_i(t+1) = F_i(t)$  for  $i \neq A_t$ ,  $F_{A_t}(t+1) = \text{Update}(F_{A_t}(t), A_t, r_t(A_t))$ . Choosing  $F_i(t)$  as the posterior over mean rewards independently for each arm recovers Thompson Sampling.

### Frequentist Regret of Thompson Sampling

Let arm 1 be optimal with mean  $\mu_1$  and gaps  $\Delta_i = \mu_1 - \mu_i$ . For suitable choices of  $F_i(1)$  and Update, TS enjoys finite-time and asymptotic guarantees. Gaussian rewards: with appropriate  $F_i(t)$  and Gaussian updates,  $\lim_{T \rightarrow \infty} \text{Reg}(T)/\log T = \sum_{i: \Delta_i > 0} 2/\Delta_i$ . Further, for a universal constant  $C > 0$ ,  $\text{Reg}(T) \leq C\sqrt{kT \log T}$ .

### Thompson Sampling for Linear Bandits

Environments  $\theta \in E \subset \mathbb{R}^d$ , actions  $a \in A \subset \mathbb{R}^d$ . For  $\theta \in E$ ,  $a \in A$ ,  $P_{\theta,a}$  is 1-subGaussian with mean  $\langle \theta, a \rangle$ . TS for linear bandits: round  $t$ , sample  $\theta_t$  from the posterior over  $\theta$ , play  $A_t = \arg \max_{a \in A} \langle a, \theta_t \rangle$ , observe  $r_t(A_t)$ . Computation for TS can be considerably less than LinUCB, which requires forming the confidence set  $C_t$  and solving  $\max_{a \in A} \max_{\tilde{\theta} \in C_t} \langle a, \tilde{\theta} \rangle$ .

### Bayesian Regret of TS for Linear Bandits

Assume  $\|\theta\|_2 \leq S$  w.p. one under the prior,  $\sup_{a \in A} \|a\|_2 \leq L$ , and  $|\langle a, \theta \rangle| \leq 1$ . The Bayesian regret of TS satisfies

$$B\text{Reg}_T \leq 2 + 2\sqrt{2dT\beta^2 \log(1 + \frac{TS^2L^2}{d})} =$$

$O(d\sqrt{T \log T \log(1 + TS^2L^2/d)})$ , where

$$\beta = 1 + \sqrt{2 \log T + d \log(1 + TS^2L^2/d)}$$

### Bayesian Regret of TS for Linear Bandits: Proof

Assume  $\|\theta\|_2 \leq S$  w.p. 1,  $\sup_{a \in A} \|a\|_2 \leq L$ , and  $|\langle a, \theta \rangle| \leq 1$ . Define  $V_t = \lambda I + \sum_{s=1}^{t-1} A_s A_s^\top$  and  $\hat{\theta}_t = V_t^{-1} \sum_{s=1}^{t-1} r_s A_s$ . Construct the confidence set  $C_t = \{\theta : \|\theta - \hat{\theta}_t\|_{V_t} \leq \beta_t\}$  with  $\beta_t = 1 + \sqrt{2 \log(1/\delta) + d \log(1 + (t-1)L^2/\lambda)}$ . Thompson Sampling selects  $A_t = \arg \max_{a \in A} \langle a, \theta_t \rangle$  with  $\theta_t$  sampled from the posterior. Standard elliptical potential lemma gives  $\sum_{t=1}^T \min\{\|A_t\|_{V_t}^{-1}, 1\} \leq$

$$2 \log \frac{\det V_T + 1}{\det V_1} \leq 2d \log(1 + TL^2/(d\lambda)).$$

Using  $\langle A_t, \theta - \hat{\theta}_t \rangle \leq \|A_t\|_{V_t^{-1}} \|\theta - \hat{\theta}_t\|_{V_t} \leq \beta_t \|A_t\|_{V_t^{-1}}$  and Cauchy-Schwarz, we obtain

$$B\text{Reg}_T \leq 2 + 2\sqrt{2dT\beta^2 \log(1 + TL^2/d)} =$$

$$O(d\sqrt{T \log T \log(1 + TL^2/d)}).$$

### Contextual Bandits over a Large Policy Class II

Rounds  $t = 1, \dots, T$ : observe context  $x_t \in \mathcal{X}$ , choose action  $a_t \in [K]$ , observe reward  $r_t(a_t) \in [0, 1]$ . Context-reward pairs  $(x, r)$  drawn i.i.d. from distribution  $D$ . Compete with best policy  $\pi^* \in \Pi$ , with regret  $\text{Reg}_T = \mathbb{E}[\sum_{t=1}^T r_t(\pi^*(x_t)) - r_t(a_t)]$ . Goal: minimize regret and maintain computational efficiency when  $|\Pi|$  is huge, ideally sublinear in  $|\Pi|$ .

### Problem Setting and Regret Stochastic i.i.d.

contexts and rewards:  $(x_t, r_t) \sim D$ . Algorithm sees  $x_t$ , chooses  $a_t$ , observes  $r_t(a_t)$ . Actions  $a \in A$ ,  $|A| = K$ . Regret w.r.t. best  $\pi^* = \arg \max_{\pi \in \Pi} R(\pi)$ , with  $R(\pi) = \mathbb{E}_{(x,r) \sim D}[r(\pi(x))]$ . Instantaneous regret of any policy  $\pi \in \Pi$  is  $\text{Reg}(\pi) = R(\pi^*) - R(\pi)$ . Empirical cumulative regret after  $T$  rounds:  $\text{Reg}_T = \sum_{t=1}^T r_t(\pi^*(x_t)) - r_t(a_t)$ . Target regret:  $\hat{O}(\sqrt{KT \log |\Pi|})$ .

**Explore-Then-Commit: Choice of Exploration Length**

In ETC, total regret splits into exploration and commit phases:  $R(T) \sim T_{\text{explore}} + (T - T_{\text{explore}})\sqrt{K/T_{\text{explore}}}$ . Optimizing  $T_{\text{explore}}$  to minimize this upper bound gives  $T_{\text{explore}} \sim T^{2/3}K^{1/3}$ . Hence, the  $T^{2/3}$  exponent balances exploration and exploitation, minimizing total regret. The constant and  $K$  factors are often omitted in notes, leaving  $O(T^{2/3})$  as the typical guideline.

### Warm-Up: From Exp4 to EXP4.P

Goal: Develop high-probability guarantee for Exp4. Maintain weights  $w_t(\pi)$ ,  $\pi \in \Pi$  over policies. At each round, sample  $\pi_t \sim w_t$ , play  $a_t \sim \pi_t$ , receive reward  $r_t(a_t)$ , and update  $w_{t+1}(\pi)$  using inverse propensity scoring (IPS). High-probability regret is  $\hat{O}(\sqrt{KT \ln |\Pi|})$  for finite  $\Pi$ . Limitation: per-round computation  $\Omega(|\Pi|)$  due to maintaining/mixing all policy weights; statistically optimal but computationally demanding. Ideas for improvement: use Explore-Then-Commit (ETC) for computational efficiency, move to oracle-based models, and solve a suitable optimization problem over  $\pi \in \Pi$ .

### Epoch-Greedy: Efficient but Sub-Optimal Regret

Explore-Then-Commit approach: explore for  $O(T^{2/3})$  rounds, then commit to empirical best policy using a supervised oracle. In the i.i.d. setting, regret is  $O(T^{2/3}(K \ln |\Pi|)^{1/3})$ , worse than  $\sqrt{T}$  but computationally efficient (one oracle call per round). **Cost-Sensitive Classification (CSC) Oracle and Reduction** ArgMax Oracle (AMO) searches  $\Pi$  efficiently: given logged data  $(x_{t'}, r_{t'})$  for  $t' \leq t$ , solves  $\pi_t^* = \arg \max_{\pi \in \Pi} \sum_{t'=1}^t r_{t'}(\pi(x_{t'}))$ . Each policy  $\pi : \mathcal{X} \rightarrow [K]$  is like a classification model; misclassification cost is  $-r_{t'}(\pi(x_{t'}))$ . AMO is equivalent to a CSC oracle; allows search over  $\Pi$  without explicit enumeration.

**Inverse Propensity Score (IPS)** Logged bandit data only contains  $r_t(a_t)$ , not full reward vector  $r_t \in \mathbb{R}^K$ . Construct unbiased estimate via IPS: let  $p_{t'}(a_{t'})$  be probability of choosing  $a_{t'}$  at time  $t'$ . IPS estimator for  $\pi$  is  $\hat{\eta}_t(\pi) = \frac{1}{t} \sum_{\tau=1}^t \frac{r_\tau(a_\tau) \mathbf{1}\{\pi(x_\tau)=a_\tau\}}{p_\tau(a_\tau)}$ . Unbiased but high variance if  $p_\tau(a_\tau)$  is small; exploration must control variance via explicit constraints.

### Dudík et al. (2011) "Monster" Paper:

**RandomizedUCB (RUCB)** Algorithm maintains distribution  $P_t \in \Delta \Pi$  over policies, inspired by UCB. Computes  $P_t$  with small estimated regret and satisfies variance (exploration) constraints. Constrained optimization solved via ellipsoid-based separation

oracle. Sample actions from smoothed mixture induced by  $P_t$ . Conditional action distribution given context  $x$ :  $W_P(a|x) = \sum_{\pi \in \Pi: \pi(x)=a} P(\pi)$ , smoothed as  $W'_P(a|x) = (1 - K\mu)W_P(a|x) + \mu$ . History  $h_t = \{(x_\tau, a_\tau, r_\tau, p_\tau)\}_{\tau=1}^t$ , empirical regret  $\eta_t(W) = \frac{1}{t} \sum_{(x,a,r,p) \in h_t} \frac{r \cdot W(a|x)}{p}$ ,  $\Delta_t(W) = \eta_t(\pi_t) - \eta_t(W)$ .

### Randomized UCB

### RandomizedUCB Algorithm

**Input:** policy class  $\Pi$ , confidence  $\delta$ , number of arms  $K$  **Initialize:**  $h_0 \equiv \emptyset$

**For** each timestep  $t = 1, \dots, T$ : 1. Observe context  $x_t$  2. Define  $C_t \leq 2 \log\left(\frac{N_t}{\delta}\right)$ ,  $\mu_t \doteq \min\left\{\frac{1}{2K}, \sqrt{\frac{C_t}{2Kt}}\right\}$  3.

Let  $P_t$  be a distribution over  $\Pi$  that approximately solves:  $\min_P \sum_{\pi \in \Pi} P(\pi) \Delta_{t-1}(\pi)$  subject to, for all distributions  $Q$  over  $\Pi$ :

$$\mathbb{E}_{\pi \sim Q} \left[ \frac{1}{t-1} \sum_{i=1}^{t-1} \frac{1}{(1-K\mu_t)W_P(x_i, \pi(x_i)) + \mu_t} \right] \leq$$

$$\max \left\{ 4K, \frac{(t-1)\Delta_{t-1}(W_Q)^2}{180C_{t-1}} \right\} \text{ ensuring objective is}$$

within  $\epsilon_{\text{opt},t} = O(\sqrt{KC_t/t})$  of optimum, each constraint satisfied with slack  $\leq K$ .

4. Define distribution over actions:

$$W'_t(a) \doteq (1 - K\mu_t)W_{P_t}(x_t, a) + \mu_t, \quad \forall a \in A$$

5. Sample action  $a_t \sim W'_t$

6. Observe reward  $r_t$

7. Update history:  $h_t \doteq h_{t-1} \cup (x_t, a_t, r_t, W'_t(a_t))$

### End For

### Exploration Constraints: What $W'_P$ are Good?

- Optimizing over  $P(\pi)$ , which determines smoothed policy  $W'(a|x)$  - For all  $Q \in \Delta \Pi$

$$\mathbb{E}_{\pi \sim Q, x_\tau \sim h_{t-1}} \left[ \frac{1}{W'_P(\pi(x_\tau)|x_\tau)} \right] \leq$$

$\max(4K, \beta_t \Delta_{t-1}^2(W_Q))$  - Terms inside expectation are of granularity  $(a_\tau, x_\tau)$  - We have,  $a_\tau = \pi(\tau)$ ,  $\pi \sim Q$ ,  $\tau \in [t-1]$  - Thus,  $Q$  implies a weight of  $W_Q(\pi(x_\tau)|x_\tau)$  on  $x_\tau$  term -  $\Delta_{t-1}$  is of the same granularity, already in terms of  $W_Q$  - Unpacking the expectation notation: for any  $W_Q(\pi(x_\tau)|\tau)$

$$\mathbb{E}_{x_\tau \sim h_{t-1}} \left[ \sum_a \frac{W_Q(a|x)}{W'_P(a|x)} \right] \leq \max(4K, \beta_t \Delta_{t-1}^2(W_Q))$$

Upper bound is tied to regret of  $W_Q$  - Tighter bound for promising (low-regret) policies  $\pi \sim W_Q$  -  $W'_P$  needs to be like such good policies  $W_Q$  - Ratio  $\frac{W_Q}{W'_P}$  cannot be

large for good policies  $W_Q$

**RUCB: Solving the Optimization Problem** -  $\Delta \Pi$ : Convex hull of all policy vectors  $\pi$  - Avoid confusion with  $\Delta_t$ , which (unfortunately) is the empirical regret - Distributions over policies are points in  $\Delta \Pi$ , e.g.,  $P \in \Delta \Pi$  - Define  $W_P, W'_P$  as before, for the target  $P$  - Such  $P(\pi)$  implies  $W'_P(\pi(x)|x)$  at  $(\pi(x), x)$  granularity - Consider the following convex optimization problem:

$$\begin{aligned} & \min \\ \text{s.t.} \quad & \Delta_{t-1}(W) \leq s \\ & W \in \Delta \Pi \\ & \forall Z \in \Delta \Pi, \mathbb{E}_{x_\tau \sim h_{t-1}} \left[ \sum_a \frac{Z(a|x)}{W'_P(a|x)} \right] \\ & \leq \max(4K, \beta_t \Delta_{t-1}^2(Z)) \end{aligned}$$

- Same as the RUCB optimization problem, with  $Z = W_Q$

### RUCB: Guarantees and Complexity

- High-probability regret:

$$\text{Reg}_T = O(\sqrt{TK \ln(T|\Pi|/\delta)} + K \ln(|\Pi|K/\delta))$$

- Oracle complexity:  $\tilde{O}(T^5)$ , hence called "monster" - Efficient algorithm with optimal regret demonstrated

### Why RUCB is Computationally Heavy

- Multiple constraints must hold uniformly over  $\Pi$  - Feasibility checked via separation oracles invoking subroutines - Ellipsoid iterations scale polynomially in  $t$  with large exponents

### Agarwal et al. (2014): Taming the Monster

- Same i.i.d. setting as RUCB - Retain near-optimal regret while drastically cutting oracle calls - Key ideas: sparse distributions over  $\Pi$ , epoching with warm starts - Algorithm: ILOVETOCONBANDITS

### ILOVETOCONBANDITS Algorithm

**Input:** Epoch schedule  $0 = \tau_0 < \tau_1 < \dots$ , failure probability  $\delta \in (0, 1)$  **Initialize:**  $Q_0 := \mathbf{0} \in \Delta \Pi$ , epoch  $m := 1$   $\mu_m := \min\left\{1/(2K), \sqrt{\ln(16\tau_m^2 |\Pi|/\delta)/(K\tau_m)}\right\}$  **For**  $t = 1, 2, \dots$ : 1. Observe  $x_t \in X$  2.

$(a_t, p_t(a_t)) := \text{Sample}(x_t, Q_{m-1}, \pi_{\tau_{m-1}}, \mu_{m-1})$  3.

Select  $a_t$ , observe reward  $r_t(a_t) \in [0, 1]$  4. **If**  $t = \tau_m$  then Solve (OP) with history  $H_t$  and  $\mu_m$ , set  $Q_m m := m + 1$  **End For**

### Optimization Problem (OP) in Taming

- Given history  $H_t$ , minimum probability  $\mu_m$  - Define  $b_\pi := \text{Reg}_t(\pi)/100\mu_m$  - Feasibility: Find  $Q \in \Delta \Pi$  s.t.  $\sum_\pi Q(\pi) b_\pi \leq 2K$ ,  $\mathbb{E}_{x \sim H_t} \frac{1}{Q_\mu(\pi(x)|x)} \leq 2K + b_\pi, \forall \pi$  - Solved via coordinate descent  $\rightarrow$  sparse  $Q$

### Interpreting the Constraints

- First constraint: average estimated regret under  $Q \leq$  exploration budget  $2K$  - Second constraint: empirical variance control for IPS per policy; tighter for low  $b_\pi$  - Together: adaptive exploration focusing accuracy where it matters

### Algorithmic Structure (ILOVETOCONBANDITS)

- Update  $Q$  only at epoch boundaries  $\tau_m$  (e.g., doubling schedule) - Between epochs, sample from  $Q_{\mu_m}$  - Reduced smoothing  $\mu_m := \min\{1/(2K), \sqrt{\ln(16\tau_m^2 |\Pi|/\delta)/(K\tau_m)}\}$  - Sample  $(x_t, Q_{m-1}, \pi_{\tau_{m-1}}, \mu_{m-1})$  with parameter schedule

### Solving (OP) via Coordinate Descent

- Each epoch: call AMO once, add weight to single  $\pi$ , decrease potential function - Produces sparse  $Q$ , support size  $\tilde{O}(\sqrt{Kt}/\ln(|\Pi|/\delta))$  by round  $t$

### Sampling and Smoothing

- For context  $x$ , play action with  $Q_\mu(a|x) = (1 - K\mu) \cdot \text{Pr}_{\pi \sim Q}[\pi(x) = a] + \mu$  ensuring  $Q_\mu(a|x) \geq \mu$  for all  $a$  - Maintain accurate propensity logs  $p_t = Q_\mu(a_t|x_t)$  for IPS - Sparse  $Q$  helps with computation

### Main Theorems (Agarwal et al., 2014)

- With probability  $\geq 1 - \delta$ , regret:  $\text{Reg}(T) = O(\sqrt{KT \ln(T|\Pi|/\delta)} + K \ln(T|\Pi|/\delta))$  - Total oracle calls:  $\tilde{O}(\sqrt{KT}/\ln(|\Pi|/\delta))$  - Net running time:  $\tilde{O}(T^{1.5}\sqrt{K} \log |\Pi|)$  - Achieves effectively optimal regret with efficient computation

### Why the Tamed Approach is Fast

- Sparse  $Q \rightarrow$  cheap sampling, fewer constraints to check - Epoching + warm starts  $\rightarrow$  one oracle call per epoch - Total oracle calls sublinear in  $T$  **Regret vs. Compute: A Precise Comparison** - EXP4.P:  $\tilde{O}(\sqrt{KT \ln |\Pi|})$  regret;  $\Omega(|\Pi|)$  per-round cost - Epoch-Greedy:  $O\left(T^{2/3}(K \ln |\Pi|)^{1/3}\right)$  regret;  $\tilde{O}(1)$  oracle call/round - RUCB (2011): optimal regret (up to

logs);  $\tilde{O}(T^5)$  oracle calls total -  
 ILOVETOCNDBANDITS (2014): optimal regret (up to  
 logs);  $\tilde{O}(\sqrt{KT}/\ln(|\Pi|/\delta))$  oracle calls  
**Two Approaches, Towards Regression Oracles** -  
 Two groups of approaches: agnostic and realizability  
 based - Agnostic algorithms: Effective for any policy  
 class  $\Pi$  - Effective way to search  $\Pi$ , e.g., using  
 CSC/AMO oracles - "Policy"-based methods -  
 Realizability based algorithms: Assumption on reward  
 generation model - LinUCB assumes  $\mathbb{E}[r(a)] = \theta^* a$ ,  
 similarly other forms - "Value"-based methods -  
 Stochastic CB with realizability: For some function  
 class  $\mathcal{F}$ , there is a predictor  $f^* \in \mathcal{F}$ , s.t.

$$\mathbb{E}[r(a) \mid x, a] = f^*(x, a), \quad \forall x \in \mathcal{X}, a \in \mathcal{A}$$

- For history  $H$ , assume weighted least squares  
 regression oracle

$$\operatorname{argmin}_{f \in \mathcal{F}} \sum_{(w, x, a, y) \in H} w(f(x, a) - y)^2$$