EN.601.774 Theory of Replicable ML Spring 2025

Lecture 20

Instructor: *Jess Sorrell* Scribe: *Jess Sorrell*

1 Replicable Heavy Hitters

Definition 1.1 (v-heavy-hitter)**.** Let D be a distribution over $\mathcal{X}$. We say that $x \in \mathcal{X}$ is a v-heavy-hitter of D if $\Pr_{x' \sim D}[x = x'] \geq v$.

Definition 1.2 (Approximate Heavy-Hitter Problem)**.** Let L_v be the set of $x \in \mathit{support}(D)$ that are v-heavy-hitters. For a target $v, \varepsilon > 0$, output L satisfying

$$L_{v-\varepsilon} \subseteq L \subseteq L_{v+\varepsilon}$$

Identifying heavy hitters of a distribution or stream is a common problem in ML. It has numerous applications:

- NLP - identifying common tokens
- security - anomaly detection
- ML - class imbalance handling
- recommender systems - data summarization and sketching

Let D be a distribution over $\mathcal{X}$. The following algorithm reproducibly returns a set of v'-heavy-hitters of D, where v' is a random value in $[v - \epsilon, v + \epsilon]$. Picking v' randomly allows the algorithm to, with high probability, avoid a situation where the cutoff for being a heavy-hitter (i.e. v') is close to the probability mass of any $x \in \text{support}(D)$.

Algorithm 1 rHeavyHitters(ρ, v, ϵ, S)
Input: Sample S from distribution D over $\mathcal{X}$
Target reproducibility ρ
target range $[v-\varepsilon, v+\varepsilon]$
Output: List of v'-heavy-hitters of D, where $v' \in [v-\varepsilon, v+\varepsilon]$

S_1 = first $m_1 := \frac{\ln(6/(\rho(v-\epsilon)))}{v-\epsilon}$ examples from S {Step 1: Find candidate heavy-hitters}
S_2 = remaining $m_2 := \frac{9\ln(12m_1/\rho)\cdot m_1^2}{2(\rho\epsilon)^2}$ fresh examples from S {Step 2: Estimate probabilities}
for all $x \in S_1$ **do**
  $\widehat{p_x} \leftarrow \Pr_{x' \sim S_2}[x' = x]$ {Estimate $p_x := \Pr_{x' \sim D}[x' = x]$}
end for
$v' \leftarrow_r [v-\epsilon, v+\epsilon]$ uniformly at random {Step 3: Remove non-v'-heavy-hitters}
Remove from S_1 all x for which $\widehat{p_x} < v'$.
return S_1

Algorithm rHeavyHitters returns exactly the list of v'-heavy-hitters so long as the following hold:

1. In Step 1 of Algorithm 1, all $(v-\epsilon)$-heavy-hitters of D are included in S_1.
2. In Step 2, the probabilities $\widehat{p_x}$ for all $x \in S_1$ are correctly estimated to within error $\rho\epsilon/(3m_1)$.
3. In Step 3, the randomly sampled v' does not fall within an interval of width $\rho\epsilon/(3m_1)$ centered on the true probability of a $(v-\epsilon)$-heavy-hitter of D.

We show that these 3 conditions will hold with probability at least $1-\rho/2$, and so will hold for two executions with probability at least $1-\rho$.

Lemma 1.3. *For all $\varepsilon \in (0, 1/2)$, $v \in (\varepsilon, 1-\varepsilon)$, with probability at least $1-\rho$,* rHeavyHitters *is reproducible, returns a list of v'-heavy-hitters for some $v' \in [v-\varepsilon, v+\varepsilon]$, and has sample complexity $\widetilde{O}\left(\frac{1}{\rho^2\epsilon^2(v-\epsilon)^2}\right)$.*

Proof. We say Step 1 of Algorithm 1 succeeds if all $(v-\epsilon)$-heavy-hitters of D are included in S_1 after Step 1. By definition, a $v-\varepsilon$-heavy-hitter has probability at least $v-\varepsilon$ under

D, so the probability it's not included in S_1 which is of size m_1 is

$$\begin{aligned}\Pr_{S_1 \sim D}[x \notin S_1] &\leq (1 - v + \varepsilon)^{m_1} \\ &\leq (e^{-v+\varepsilon})^{m_1} \\ &= e^{-\ln(6/(\rho(v-\varepsilon)))} \\ &= e^{\frac{\ln \rho(v-\varepsilon)}{6}} \\ &= \frac{\rho(v-\varepsilon)}{6}.\end{aligned}$$

There are at most $\frac{1}{v-\varepsilon}$ $(v-\epsilon)$-heavy-hitters in any distribution, so union bounding over all of them, we see that

$$\begin{aligned}\Pr_{S_1 \sim D}[\exists \text{ v-heavy-hitter } x, x \notin S_1] &\leq \frac{1}{v-\varepsilon}\frac{\rho(v-\varepsilon)}{6} \\ &= \rho/6.\end{aligned}$$

So Step 1 succeeds except with probability at most $\rho/6$.

Step 2 succeeds if the probabilities for all $x \in S_1$ are correctly estimated to within error $\rho\epsilon/(3m_1)$. We draw $m_2 = \frac{9\ln(6m_1/\rho)m_1^2}{(\rho\varepsilon)^2}$ examples from D to estimate the probability. Applying the Chernoff-Hoeffding inequality, it follows that

$$\begin{aligned}&\Pr_{S_2 \sim D}[|\tfrac{1}{|S_2|}\sum_{x' \in S_1} \mathbb{1}[x' = x] - \Pr_{x' \sim S_2}[x' = x]| \geq t] \leq 2e^{-2t^2 m_2} \\ &\Rightarrow \Pr_{S_2 \sim D}[|\tfrac{1}{|S_2|}\sum_{x' \in S_1} \mathbb{1}[x' = x] - \Pr_{x' \sim S_2}[x' = x]| \geq \frac{\rho\varepsilon}{3m_1}] \leq 2e^{\frac{-2\rho^2\varepsilon^2 m_2}{9m_1^2}} \\ &\leq 2e^{-\ln 12m_1/\rho} \\ &= \frac{\rho}{6m_1}.\end{aligned}$$

So each p_x is estimated to within error $\frac{\rho\epsilon}{3m_1}$ except with probability at most $\frac{\rho}{6m_1}$ in Step 2. Union bounding over all m_1 possible $x \in S_1$,

$$\Pr_{S_2 \sim D}[\exists p_x \text{ with estimation error } > \tfrac{\rho\varepsilon}{3m_1}] \leq \frac{\rho}{6}$$

so Step 2 succeeds except with probability $\rho/6$.

Step 3 succeeds if the returned S_1 is exactly the set of v'-heavy-hitters of D. Conditioned on the previous steps succeeding, Step 3 succeeds if the randomly chosen v' is not within $\frac{\rho\epsilon}{3m_1}$ of the true probability of any $x \in S$ under distribution D. A v' chosen randomly from the interval $[v-\epsilon, v+\epsilon]$ lands in any given subinterval of width $\rho\epsilon/(3m_1)$ with probability $\rho/(6m_1)$,

$$\Pr_{v' \sim Unif([v-\varepsilon, v+\varepsilon])}[v' \in [p_x, \hat{p}_x]] \leq \tfrac{\rho\epsilon}{3m_1} \cdot \tfrac{1}{2\varepsilon}$$
$$= \tfrac{\rho}{6m_1}.$$

Union bounding over the m_1 elements in S_1, Step 3 succeeds with except with probability at most $\rho/6$.

Therefore, Algorithm 1 outputs exactly the set of v'-heavy-hitters of D except with probability at most $\rho/2$. If we consider two executions of Algorithm 1, both using the same shared randomness for chooosing v', they will both output the set of v'-heavy-hitters of D except with probability at most ρ, and so rHeavyHitters is ρ-replicable.

The sample complexity is

$$m_1 + m_2 \in \tilde{\Omega}\left(\frac{1}{(\rho\epsilon(v-\epsilon))^2}\right).$$

□

Corollary 1.4. *If v and ε are constants, then* $\mathsf{rHeavyHitters}_{\rho,v,\varepsilon}$ *has sample complexity* $\widetilde{O}\left(1/\rho^2\right)$.

Learning Heavy-hitters using Statistical Queries. Next, we show that any statistical query algorithm for the v-heavy-hitters problem requires $\Omega(\log|\mathcal{X}|/\log(1/\tau))$ calls to the SQ oracle. Since Algorithm 1 has a sample complexity independent of the domain size, this implies a separation between reproducible problems and problems solvable using only SQ queries.

Consider the ensemble $\{D_x\}_{x\in\mathcal{X}}$ on $\mathcal{X}$, where distribution D_x is supported entirely on a single $x \in \mathcal{X}$.

Claim 1.5 (Learning Heavy-hitters using Statistical Queries)**.** *Any statistical query algorithm for the v-heavy-hitters problem on ensemble $\{D_x\}_{x\in\mathcal{X}}$ requires $\Omega(\log|\mathcal{X}|/\log(1/\tau))$ calls to the SQ oracle.*

Proof. An SQ algorithm for the v-heavy-hitters problem must, for each distribution D_x, output set $\{x\}$ with high probability. An SQ oracle is allowed tolerance τ in its response to statistical query ϕ. So, for any ϕ, there must be some distribution D_x for which the following holds: at least a τ-fraction of the distributions $D_{x'}$ in the ensemble satisfy $|\phi(x') - \phi(x)| \leq \tau$. Thus, in the worst case, any correct SQ algorithm can rule out at most a $(1-\tau)$-fraction of the distributions in the ensemble with one query. If $\mathcal{X}$ is finite, then an SQ algorithm needs at least $\log_{1/\tau}(|\mathcal{X}|)$ queries. □

2 Replicable learning of finite hypothesis classes

Now that we've seen natural settings in which our reduction is tight (and therefore exhibited a quadratic statistical separation between privacy and replicability), it is reasonable to ask whether there are any settings under which the reduction is loose, or even where privacy and replicability might have the same statistical cost. In this section, we'll show this is indeed the case for (certain regimes of) a closely related problem: realizable PAC-learning. In particular, in this section we exhibit a replicable algorithm for PAC-learning that gives a quadratically improved dependence on the accuracy and confidence parameters over applying our reduction from privacy (see thm:finitehypred).

Theorem 2.1 (Finite Classes are Replicably Learnable)**.** *Any class H is replicably Agnostic learnable with sample complexity:*

$$m(\rho, \alpha, \beta) \leq O\left(\frac{\log^2 |H| + \log \frac{1}{\rho\beta}}{\alpha^2 \rho^2} \log^3 \frac{1}{\rho}\right).$$

In the realizable setting, the α-dependence can be improved to linear:

$$m(\rho, \alpha, \beta) \leq O\left(\frac{\log^2 |H| + \log \frac{1}{\rho\beta}}{\alpha \rho^2} \log^3 \frac{1}{\rho}\right).$$

thm:rFinite gives a quadratic improvement over the sample complexity via reduction from private learning in both confidence and accuracy, and in particular has the same asymptotic dependence as in private PAC-learning (and hence avoids any statistical blowup in the setting where $\log |H|$ is thought of as small). In fact, it's worth noting the result is tight in these parameters, as even standard PAC-learning requires the same dependencies.

2.0.1 Algorithm

At its core, the algorithm achieving thm:rFinite relies on a simple random thresholding trick. In particular, the idea is roughly to estimate the risk of each concept in the class H by standard uniform convergence bounds, choose a random error threshold $v \in [OPT, OPT + \alpha]$, and finally output a random $f \in H$ with empirical error $S(f) = \frac{1}{|S|} \sum\limits_{(x,y) \in S} \mathbf{1}[f(x) \neq y]$ at most v. Implementing this strategy requires a bit more effort, and is achieved formally by the following algorithm.

Algorithm 2 (Intermediate) Replicable Learner for Finite Classes

Replicably outputs hypothesis with error at most $OPT + \alpha$ **Input:** Finite Class H, Joint Distribution D over $X \times \{0,1\}$ (Sample Access)

Parameters:

- Replicability, Accuracy, Confidence $\rho, \alpha, \beta > 0$
- Sample Complexity $m = m(\rho, \alpha, \beta) \leq O\left(\frac{\log^2 |H| \log \frac{1}{\rho} + \rho^2 \log \frac{1}{\beta}}{\alpha^2 \rho^4}\right)$
- Replicability bucket size $\tau \leq O(\frac{\alpha\rho}{\ln |H|})$

Algorithm:

1. Draw a labeled sample $S \sim D^m$ and compute $S(f)$ for every $f \in H$.
2. Replicably output initialization $v_{\text{init}} \in [OPT, OPT + \alpha/2]$ (see alg:agnostic-subroutine)
3. Select random threshold $v \leftarrow_r \{v_{\text{init}} + \frac{3}{2}\tau, v_{\text{init}} + \frac{5}{2}\tau, \ldots, v_{\text{init}} + \alpha/4 - \tau/2\}$
4. Randomly order all $f \in H$

return Output the first hypothesis f in the order s.t. $S(f) \leq v$.

We note that Step 2, estimating OPT, follows essentially the same argument as the basic replicable statistical query algorithm of **?**. We give the argument in app:OPT for completeness.

We note that while Algorithm rFinite is a replicable agnostic PAC learner, it is not quite sufficient to prove thm:rFinite due to its poor dependence on ρ. We'll see in the next section how to obtain the stated parameters by separately amplifying rFinite starting from good constant replicability.

2.0.2 Analysis

We'll start by proving the following weaker bound for our intermediate learner.

Theorem 2.2 (Intermediate Learnability of Finite Classes)**.** *Let H be any finite concept class. Algorithm* rFinite *is a (proper) agnostic replicable learning algorithm for H with sample complexity:*

$$m(\rho, \alpha, \beta) \leq O\left(\frac{\log^2 |H| \log(\frac{1}{\rho}) + \rho^2 \log \frac{1}{\beta}}{\alpha^2 \rho^4}\right).$$

In the realizable setting, the (α, β)-dependence can be improved to:

$$m(\rho, \alpha, \beta) \leq O\left(\frac{\log^2 |H| \log(\frac{1}{\rho}) + \rho^4 \log \frac{1}{\beta}}{\alpha \rho^4}\right).$$

The main challenge in thm:intermediate-learner is proving replicability. (Accuracy and failure probability are essentially immediate from standard uniform convergence arguments.) To this end, note that the randomness r used by rFinite is largely broken into three parts: estimating OPT, choosing a random threshold, and ordering the concepts in H. We'll focus first on the latter two, where the choice of v restricts H to two subsets H_1 and H_2 (those with empirical error at most v), depending on input samples S_1 and S_2. We first appeal to the classical observation of Broder **?** to argue that as long as the symmetric difference of H_1 and H_2 are small, outputting the first concept from these sets (according to the random ordering) is a replicable procedure.

Let $O(H, r)$ be a random ordering of concept class H. Let $\emptyset \subset H_1, H_2 \subseteq H$, and let f_1 and f_2 be the first elements of H_1 and H_2 respectively according to $O(H, r)$. Then $\Pr_r[f_1 \neq f_2] = \frac{|H_1 \Delta H_2|}{|H_1 \cup H_2|}$, where Δ denotes the symmetric difference.

The key to proving replicability is then to observe that most choices of v induce small symmetric difference between the corresponding H_1 and H_2. Namely, the idea is to observe that for any fixed joint distribution D, intervals

$$I_0 = [OPT, OPT + \tau], \ldots, I_{\alpha/(2\tau)} = [OPT + \alpha/2 - \tau, OPT + \alpha/2],$$

and corresponding threshold positions $v_i = OPT + \frac{(2i+1)}{2}\tau$, the sets

$$H_1^{(i)} = \{h \in H : S_1(h) \leq v_i\}, \quad H_2^{(i)} = \{h \in H : S_2(h) \leq v_i\}$$

are close for most choices of v_i, S_1, and S_2. To adjust for the fact that we don't know the value of OPT, we will in fact prove something slightly more general that allows our starting point to range anywhere from OPT to $OPT + \alpha/2$.

Lemma 2.3. *Let $v_{init} \in [OPT, OPT + \alpha/2]$ and $\tau \leq O\left(\frac{\alpha\rho^2}{\log|H|}\right)$ a parameter that divides $\alpha/4$. Define the intervals*

$$I_0 = [v_{init}, v_{init} + \tau),\ I_1 = [v_{init} + \tau, v_{init} + 2\tau),\ \ldots\ ,\ I_{\frac{\alpha}{4\tau}} = \left[v_{init} + \frac{1}{4}\alpha - \tau, v_{init} + \frac{1}{4}\alpha\right]$$

and corresponding thresholds $v_i = v_{init} + \frac{(2i+1)}{2}\tau$, and let

$$H_1^{(i)} = \{h \in H : S_1(h) \leq v_i\}, \quad H_2^{(i)} = \{h \in H : S_2(h) \leq v_i\}$$

denote the hypotheses with empirical error at most v_i across two independent samples S_1 and S_2 of size $O(\frac{\log \rho^{-1}}{\tau^2})$. Then with probability at least $1 - \rho/4$, a uniformly random choice of $i \in [\frac{\alpha}{4\tau}]$ satisfies:

$$\frac{|H_1^{(i)} \Delta H_2^{(i)}|}{|H_1^{(i)} \cup H_2^{(i)}|} \leq \rho/4.$$

Proof. For convenience of notation, let $|I_i|$ denote the number of hypotheses whose true risk lies in interval I_i, and $|I_{[i]}|$ the number of hypotheses in intervals up through I_i. We call a threshold v_i "bad" if any of the following conditions hold.

1. The ith interval has too many elements:

$$|I_i| > \frac{\rho}{30}|I_{[i-1]}|.$$

2. The number of elements beyond I_i increases too quickly:

$$\exists j \geq 1 : |I_{i+j}| \geq e^j |I_{[i-1]}|.$$

and "good" otherwise. We will argue the following two claims.

1. If v_i is a good threshold, then $H_1^{(i)}$ and $H_2^{(i)}$ are probably close

$$\Pr_{S_1,S_2}\left[\frac{|H_1^{(i)} \Delta H_2^{(i)}|}{|H_1^{(i)} \cup H_2^{(i)}|} \leq \frac{\rho}{4}\right] \geq 1 - \frac{\rho}{8}.$$

2. At most a $\frac{\rho}{8}$ fraction of thresholds are bad.

Since we pick a threshold uniformly at random, it is good with probability at least $1 - \rho/8$ and a union bound gives the desired result.

It remains to prove the claims. For the first, observe that for any fixed hypothesis h with true risk $D(h) \in I_{i+j}$, the probability that the empirical risk of h is less than v_i is at most

$$\Pr[S(h) \leq v_i] \leq e^{-\Omega(j^2\tau^2|S|)} \tag{1}$$

by a Chernoff bound. Let x_i denote the variable which counts the number of hypotheses with true risk beyond I_i that cross the threshold v_i empirically. If v_i is "good," we can bound $\mathbb{E}[x_i]$ by

$$\mathbb{E}[x_i] \leq |I_{[i-1]}| \sum_{j>0} e^{-\Omega(j^2\tau^2|S|-j)} \leq \frac{\rho^2}{2000}|I_{[i-1]}|$$

for our choice of $|S|$. Markov's inequality then promises

$$\Pr\left[x_i \geq \frac{\rho}{30}|I_{[i-1]}|\right] \leq \frac{\rho}{64}.$$

On the other hand, the probability any hypothesis in $I_{[i-1]}$ crosses v_i is at most $e^{-\Omega(\tau^2|S|)}$, so similarly the probability that more than a $\frac{\rho}{30}$ fraction of such hypotheses cross v_i is at most $\frac{\rho}{64}$. Finally, since v_i is 'good,' I_i itself contributes at most $\frac{\rho}{30}|I_{[i-1]}|$ hypotheses that cross the threshold in the worst case, so in total we have that with probability at least $1 - \frac{\rho}{32}$, at most $\frac{\rho}{10}|I_{[i-1]}|$ hypotheses cross the threshold in either direction. Considered over two runs of the algorithm, this implies that with probability at least $1 - \frac{\rho}{16}$, $|H_1^{(i)} \Delta H_2^{(i)}|$ cannot be too big

$$|H_1^{(i)} \Delta H_2^{(i)}| \leq \frac{\rho}{5}|I_{[i-1]}|.$$

Furthermore, since the probability that more than a $\frac{\rho}{30}$ fraction of hypotheses in $I_{[i-1]}$ cross v_i is at most $\frac{\rho}{64}$, we also have that $|H_1^{(i)} \cup H_2^{(i)}|$ cannot be too small:

$$|H_1^{(i)} \cup H_2^{(i)}| \geq \left(1 - \frac{\rho}{15}\right) |I_{[i-1]}|$$

with probability at least $1 - \frac{\rho}{64}$. Thus altogether a union bound gives

$$\Pr_{S_1,S_2}\left[\frac{|H_1^{(i)} \Delta H_2^{(i)}|}{|H_1^{(i)} \cup H_2^{(i)}|} \leq \frac{\rho}{4}\right] \geq 1 - \frac{\rho}{8}$$

as desired.

Finally, we need to show that almost all thresholds are good. To see this, first observe that since $v_{\text{init}} \geq OPT$, $|I_{[i]}| > 0$ for all $i \geq 0$. To count the number of bad thresholds, let $i_1 \geq 1$ be the position of the first bad threshold, and t_1 denote the largest index such that $i_1 + t_1$ fails a condition. Define i_j and t_j recursively as the first bad threshold beyond $i_{j-1} + t_{j-1}$ and its corresponding latest failure. Observe that by construction, any interval that does not lie in any $[i_j.i_j + t_j]$ is good, so there are at most $\sum t_j$ bad thresholds.

Let ℓ denote the final index of the above greedy process. By definition of a bad interval, each t_j multiplicatively increases the number of total hypotheses from $I_{[i_j]}$ by at least $\left(1 + \frac{\rho}{30}\right)^{t_j}$. Since $|I_0| \geq 1$ and the total number of hypotheses is $|H|$ by definition, we may therefore write:

$$|H| \geq |I_{[i_\ell + t_\ell]}| \geq \left(1 + \frac{\rho}{30}\right)^{\sum_{j=1}^{\ell} t_j}$$

and thus that the total number of bad intervals is at most

$$\sum_{j=1}^{\ell} t_j \leq O\left(\frac{\log(|H|)}{\rho}\right).$$

Since we have chosen τ such that the total number of intervals altogether is at least $\Omega\left(\frac{\log(|H|)}{\rho^2}\right)$, the appropriate choice of constant gives that at most a $\rho/8$ fraction are bad as desired. □

To complete the argument, it is enough to show we can find a good starting point v_{init}.

Lemma 2.4. *There exists a ρ-replicable algorithm over $O\left(\frac{\log(\frac{|H|}{\rho\beta})}{\rho^2\alpha^2}\right)$ samples that outputs a good estimate of OPT with high probability:*

$$\Pr_{r,S}\left[\mathcal{A}(S) \in [OPT, OPT + \alpha/2]\right] \geq 1 - \beta$$

References