CMSC33581 UNIT 1 REVIEW

MAKSIM LEVENTAL

Exercise 1. For some finite set of words S and any **collision-resistant** hash function h let *Median Hash* $h_{\text{med}}(S)$ be the median value of the image of S under h. That is to say

$$h_{\text{med}}(S) \coloneqq \text{med}(\{h(s) \mid s \in S\})$$

Problem (1a)**.** Let $\sigma \in \Sigma$ be a uniform random permutation and S, S' be finite sets of words. Does

$$\mathbb{P}_{\Sigma}[h_{\text{med}}(\sigma(S)) = h_{\text{med}}(\sigma(S'))]$$

equal the Jaccard similarity $\text{J}(S, S')$ of S and S'?

Solution (1a)**. No**. We prove by contradiction; let

$$S \coloneqq \{\text{'a'}\} \quad S' \coloneqq \{\text{'a'}, \text{'b'}\} \tag{0.1}$$

Clearly $h_{\text{med}}(\sigma(S)) = h(\sigma(\text{'a'}))$ and

$$h_{\text{med}}(\sigma(S')) = \frac{h(\sigma(\text{'a'})) + h(\sigma(\text{'b'}))}{2}$$

for any permutation σ (assuming no collisions). Therefore, since h is collision-resistant

$$\mathbb{P}_{\sigma \in \Sigma}[h_{\text{med}}(\sigma(S)) = h_{\text{med}}(\sigma(S'))] = 0$$

while

$$\text{J}(S, S') = \frac{|\{\text{'a'}\} \cap \{\text{'a'}, \text{'b'}, \text{'b'}\}|}{|\{\text{'a'}\} \cup \{\text{'a'}, \text{'b'}, \text{'b'}\}|} = \frac{|\{\text{'a'}\}|}{|\{\text{'a'}, \text{'b'}\}|} = \frac{1}{2}$$

Problem (1b)**.** Under which conditions would the Median Hash be an exact estimator of Jaccard similarity?

Solution (1b)**.** The crux of the issue is that the median of a set of numbers isn't necessarily in the set:

$$\text{med}(\{1, 2\}) = 1.5$$

Note, further, that

$$\text{med}(\{1, 2\}) = \text{med}(\{0, 3\})$$

and therefore strings with an even number of words might spuriously appear to have high similarity according to median hash.

On the other hand, **if the cardinality of the set is odd** then the median is necessarily a member of the set. Thus, if the set of words (i.e. after making distinct) has odd cardinality then we claim $h_{\text{med}}(S)$ is an estimator for Jaccard similarity. We prove by reduction to Minimum Hash h_{min} (which is known to accurately estimate Jaccard similarity). First, for any finite set A with odd cardinality,

Date: March 3, 2021.

let h' be the map that maps the minimum of A to the median of A and vice-versa, but otherwise is the identity:

$$h'(a) \coloneqq \begin{cases} \operatorname{med}(\{a \mid a \in A\}) & \text{if } s = \min(\{a \mid a \in A\}) \\ \min(\{a \mid a \in A\}) & \text{if } s = \operatorname{med}(\{a \mid a \in A\}) \\ a & \text{otherwise} \end{cases}$$

Note that h' is well defined since $\operatorname{med}(\{a\}) \in A$. Then for any hash function h, any finite set of words S with odd cardinality, and any permutation σ

$$(h' \circ h)_{\min}(\sigma(S)) = h_{\operatorname{med}}(\sigma(S))$$

and thus

$$\begin{aligned} \mathbb{P}_{\sigma \in \Sigma}\left[h_{\operatorname{med}}(\sigma(S)) = h_{\operatorname{med}}(\sigma(S'))\right] &= \mathbb{P}_{\sigma \in \Sigma}\left[(h' \circ h)_{\min}(\sigma(S)) = (h' \circ h)_{\min}(\sigma(S'))\right] \\ &= \mathrm{J}(S, S') \end{aligned}$$

since $h' \circ h$ is a valid hash function.

Exercise 2. Let $A \coloneqq [\text{NOV}-11, \text{DEC}-12, \ldots, \text{NOV}-13, \text{MAR}-02]$.

Problem (2a)**.** Applying standard dictionary encoding 1 to the collection of dates will?

Solution (2a)**.** A dictionary encoding will **sometimes increase the size of the dataset and sometimes decrease the size of the dataset**; let D_I be the dictionary encoding that appends '!' i.e.

$$D_I : A[i] \leftrightarrow A[i] \cup !$$

and let $D_{\mathbb{N}}$ be the dictionary encoding that ordinalizes

$$A[i] \leftrightarrow i$$

Then clearly D_I increases the size of the dataset by approximately 100×1 bytes and $D_{\mathbb{N}}$ reduces the size of the dataset by

$$\begin{aligned} \Delta &= \left(\sum_{i=1}^{100} \lfloor \log_{10}(i) \rfloor + 1\right) - 100 \times 6 \\ &= 192 - 600 = -408 \end{aligned}$$

bytes since the length of a number i is $\lfloor \log_{10}(i) \rfloor + 1$.

Problem (2b)**.** Suppose, we calculate an optimal prefix-free code (like Huffman coding) to the collection of dates will?

Solution (2b)**.** Code words (and therefore code word lengths) will be **arbitrarily associated with each date** but they won't have arbitrary lengths because otherwise the code wouldn't be an optimal; we know from Huffman coding that the average code length is approximately $\log_2(100) \approx 6.644$ and therefore at least one code word must have length < 7 and at least one code word must have length ≥ 7 (i.e. not same code word lengths).

Problem (2c)**.** Consider the following encoding algorithm...

Solution (2c)**.** The algorithm produces a codex C with cardinality bounds

$$|\{\text{'MAR'}, \text{'NOV'}, \text{'DEC'}\}| \times |\{\text{'02'}, \text{'11'}, \text{'12'}, \text{'13'}\}| \leq |C| \leq |\{\text{'JAN'}, \ldots, \text{'DEC'}\}| \times |\{\text{'01'}, \ldots, \text{'31'}\}|$$

i.e. $12 \leq |C| \leq 372$ depending on how much repetition there is among the months and the days of the month. Thus, the algorithm will **sometimes increase the size of the dataset and sometimes decrease the size of the dataset**.

Exercise 3. Suppose, you are given a single hash function $\texttt{hash}(s)$ that takes in a string s as an argument and returns an integer from 1 to H and satisfies the simple uniform hashing assumption (SUHA).

Problem (3a)**.** Is it guaranteed that at least two different strings from the list have the same hash code?

Solution (3a)**. True**, by pigeonhole principle.

Problem (3b)**.** Define

```
def h2(s): return hash(str(hash(s)))
```

The maximum that `h2` can return is?

Solution (3b)**.** The answer depends on whether the question is asking the about max over arbitrary strings or the max over the same set of N strings given in the prompt. Regarding, the former (max over arbitrary strings) by the simple uniform hashing assumption the answer is **always equal to** H since

$$\mathbb{P}\left[\texttt{hash}\left(s\right) = H\right] = \frac{1}{H}$$

for any s.

Regarding the latter: let $S \coloneqq \{s_1, \ldots, s_N\}$ be the set of N strings with $H < N$, and $L \coloneqq \{\text{'1'}, \ldots, \text{'H'}\}$ be the string representations for integers $\{1, \ldots, H\}$, and

$$J \coloneqq \max\left(\{\texttt{hash}\left(s_i\right) \mid s_j \in S\}\right)$$

Note that $J \in S'$. Then $\texttt{hs2}\left(s_i\right) = l_{s_i}$ for some $l_{s_i} \in L$ and by the SUHA

$$\mathbb{P}\left[\texttt{hs2}\left(l_{s_i}\right) = J\right] = \frac{1}{H} \Rightarrow \mathbb{P}\left[\texttt{hs2}\left(l_{s_i}\right) \neq J\right] = 1 - \frac{1}{H}$$

for all $s_j \in S$. Therefore, by independence of $\texttt{hash}\left(l_{s_i}\right), \texttt{hash}\left(l_{s_j}\right)$ for $i \neq j$

$$\begin{aligned}
\mathbb{P}\left[\texttt{hs2}\left(l_{s_1}\right) \neq J \wedge \cdots \wedge \texttt{hs2}\left(l_{s_N}\right) \neq J\right] &= \prod_{i=1}^{N} \mathbb{P}\left[\texttt{hs2}\left(l_{s_i}\right) \neq J\right] \\
&= \prod_{i=1}^{N} \left(1 - \mathbb{P}\left[\texttt{hs2}\left(l_{s_i}\right) = J\right]\right) \\
&= \prod_{i=1}^{N} \left(1 - \frac{1}{H}\right) \\
&= \left(1 - \frac{1}{H}\right)^N
\end{aligned}$$

and hence

$$\mathbb{P}\left[\max\left(\{\texttt{hash}\left(s_i\right) \mid s_j \in S\}\ \right) \neq \max\left(\{\texttt{hs2}\left(s_i\right) \mid s_j \in S\}\ \right)\right] = \left(1 - \frac{1}{H}\right)^N$$

i.e. with small probability $\max\left(\{\texttt{hs2}\left(s_i\right) \mid s_j \in S\}\ \right)$ is **possibly less than** $\max\left(\{\texttt{hash}\left(s_i\right) \mid s_j \in S\}\ \right)$**.**

Problem (3c)**.** Define

```
def h3(s): return hash(s[1:])
```

What applies under the assumptions above?

Solution (3c)**.**

(1) `h3` **is independent from** `hash`. Proof: let $z \coloneqq x \cdot z'$ the concatenation of a symbol x and a string z and let $s$be a string such that $s \neq z'$. Then by SUHA

$$\begin{aligned}\mathbb{P}\left[\texttt{hs3}\left(z\right) = i \wedge \texttt{hash}\left(s\right) = j\right] &= \mathbb{P}\left[\texttt{hash}\left(z'\right) = i \wedge \texttt{hash}\left(s\right) = j\right] \\ &= \mathbb{P}\left[\texttt{hash}\left(z'\right) = i\right]\mathbb{P}\left[\texttt{hash}\left(s\right) = j\right] \\ &= \frac{1}{H}\frac{1}{H} \\ &= \mathbb{P}\left[\texttt{hs3}\left(z\right) = i\right]\mathbb{P}\left[\texttt{hash}\left(s\right) = j\right]\end{aligned}$$

(2) `h3` **and** `hash` **can be combined to make a code space of** H^2. Proof: since both $\texttt{hs3} : S \to \{1, \ldots, H\}$ and $\texttt{hash} : S \to \{1, \ldots, H\}$, taking $g\left(s\right) \coloneqq \left(\texttt{hs3}\left(s\right), \texttt{hash}\left(s\right)\right)$ then $g\left(s\right) \in H^2$.

(3) `h3` **is independent from** `hash` **but will no longer satisfy SUHA**: let $z' \coloneqq x \cdot z$ and $z'' \coloneqq y \cdot z'$ with $x \neq y$. Then

$$\begin{aligned}\mathbb{P}\left[\texttt{hs3}\left(z'\right) = i \wedge \texttt{hs3}\left(z''\right) = j\right] &= \mathbb{P}\left[\texttt{hash}\left(z\right) = i \wedge \texttt{hash}\left(z\right) = j\right] \\ &= \mathbb{P}\left[\texttt{hash}\left(z\right) = i \wedge \texttt{hash}\left(z\right) = j \wedge i \neq j\right] \\ &\quad + \mathbb{P}\left[\texttt{hash}\left(z\right) = i \wedge \texttt{hash}\left(z\right) = j \wedge i = j\right] \\ &= 0 + \mathbb{P}\left[\texttt{hash}\left(z\right) = i\right] \\ &= \frac{1}{H}\end{aligned}$$

Exercise 4. Let P, Q be discrete distributions over $\mathcal{X}$ with $P\left(x\right) > 0, Q\left(x\right) > 0$ for all $x \in \mathcal{X}$. Prove

$$H\left(P\right) \leq \sum_{x \in \mathcal{X}} \frac{P\left(x\right)}{Q\left(x\right)}$$

Solution (4)**.** Let $H(P)$ be the Shannon entropy of P. Then

$$\begin{aligned}
H(P) &= -\sum_{x\in\mathcal{X}} P(x)\log(P(x)) \\
&\leq -\sum_{x\in\mathcal{X}} P(x)\log(Q(x)) \quad \text{by Gibb's inequality} \\
&= \sum_{x\in\mathcal{X}} P(x)\log\left(\frac{1}{Q(x)}\right) \\
&\leq \sum_{x\in\mathcal{X}} P(x)\left(\frac{1}{Q(x)}-1\right) \quad \text{by } \log(x)\leq x-1 \\
&= \sum_{x\in\mathcal{X}} \frac{P(x)}{Q(x)} - \sum_{x\in\mathcal{X}} P(x) \\
&= \sum_{x\in\mathcal{X}} \frac{P(x)}{Q(x)} - 1 \leq \sum_{x\in\mathcal{X}} \frac{P(x)}{Q(x)}
\end{aligned}$$

Exercise 5. Derive the optimal number of buckets M for a histogram that buckets a discrete random variable X.

Solution. Let $\{0,\ldots,D-1\}$ be the support of X. Then the bins partition the support in $K \coloneqq D/M$-width bins

$$B_1 \coloneqq [0,1K], B_2 \coloneqq (1K,2K],\ldots, \\ B_\ell \coloneqq ((\ell-1)K,\ell K],\ldots,B_M \coloneqq ((M-1)K,MK]$$

and, for a point $x\in B_\ell$, the point mass[1] estimator $\hat{p}_n$ (for $X_1,\ldots,X_n$ i.i.d samples) is defined

$$\hat{p}_n(x) \coloneqq \frac{1}{K}\frac{|\{X_i\in B_\ell\}|}{n} = \frac{1}{K}\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}_{B_\ell}(X_i)$$

where $\mathbb{1}_A$ is the indicator[2] over event A. Let

$$x^* \coloneqq (\ell-1)K+1$$

[1] As opposed to density.

[2] If $a\in A$ then $\mathbb{1}_A(a)=1$ else 0.

Then the expectiation of $\hat{p}_n(x)$ for $x \in B_\ell$ is

$$\begin{aligned}
\mathbb{E}[\hat{p}_n] &= \frac{1}{K}\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[\mathbb{1}_{B_\ell}(X_i)] \\
&= \frac{1}{K}\frac{1}{n}\sum_{i=1}^{n}\mathbb{P}[X_i \in B_\ell] \quad \text{by } \mathbb{E}[\mathbb{1}_A] = \mathbb{P}[A] \\
&= \frac{1}{K}\mathbb{P}[X_1 \in B_\ell] \quad \text{by } X_i \text{ i.i.d} \\
&= \frac{1}{K}\sum_{j=(\ell-1)K+1}^{\ell K} p(j) \\
&\leq \frac{1}{K}\left(\sum_{j=0}^{K-1} p(x^*) + j\Delta\right) \quad \text{by } p(j+1) - p(j) \leq \Delta \\
&= \frac{1}{K}\left(Kp(x^*) + \frac{(K-1)K}{2}\Delta\right) \\
&= p(x^*) + \frac{(K-1)}{2}\Delta
\end{aligned}$$

Therefore, the bias of $\hat{p}_n(x)$ for $x \in B_\ell$ is

$$\begin{aligned}
\text{Bias}[\hat{p}_n] &= \mathbb{E}[\hat{p}_n] - p(x) \\
&= p(x^*) + \frac{(K-1)}{2}\Delta - p(x) \\
&\leq \frac{(K-1)}{2}\Delta + (K-1)\Delta \quad \text{by } |((\ell-1)K+1) - x| \leq K-1 \\
&= \frac{3}{2}(K-1)\Delta
\end{aligned}$$

Then the variance of $\hat{p}_n(x)$ for $x \in B_\ell$ is

$$\begin{aligned}
\text{Var}[\hat{p}_n] &= \frac{1}{K^2}\text{Var}\left[\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}_{B_\ell}(X_i)\right] \\
&= \frac{1}{K^2}\frac{\mathbb{P}[X_i \in B_\ell](1 - \mathbb{P}[X_i \in B_\ell])}{n} \\
&\leq \frac{1}{nK^2}\mathbb{P}[X_i \in B_\ell] \\
&\leq \frac{1}{nK^2}\left(Kp(x^*) + \frac{(K-1)K}{2}\Delta\right) \\
&= \frac{p(x^*)}{nK} + \frac{(K-1)}{2nK}\Delta
\end{aligned}$$

Thus

$$\begin{aligned}
\text{MSE}[\hat{p}_n] &= (\text{Bias}[\hat{p}_n])^2 + \text{Var}[\hat{p}_n] \\
&\leq \left(\frac{3}{2}(K-1)\Delta\right)^2 + \frac{p(x^*)}{nK} + \frac{(K-1)}{2nK}\Delta \\
&\leq (2K\Delta)^2 + \frac{p(x^*)}{nK} + \frac{\Delta}{n}
\end{aligned}$$

Then solving for the optimal M^{opt}

$$\frac{\partial}{\partial K}\left((2K\Delta)^2+\frac{p\left(x^*\right)}{nK}+\frac{\Delta}{n}\right)=0\Rightarrow p\left(x^*\right)=8\Delta^2K^3n$$

and therefore

$$K^{\text{opt}}=\frac{1}{2}\left(\frac{p\left(x^*\right)}{\Delta^2n}\right)^{1/3}\Rightarrow M^{\text{opt}}=2\left(\frac{\Delta^2D^3n}{p\left(x^*\right)}\right)^{1/3}$$