Oded: April 23, 2020

Preface

This memo provides an overview of a new "testing by implicit sampling" approach proposed by Bshouty [1]. This approach refers to testing properties that are (symmetric) sub-classes of k-juntas; that is, $f : \{0,1\}^\ell \to \{0,1\}$ has the property if there exists a function $f' : \{0,1\}^k \to \{0,1\}$ that belongs to a predetermined class of functions (over k-bit strings) such that $f(x) = f'(x_J)$ for some k-subset J. The new approach builds upon the "testing by implicit sampling" approach of Diakonikolas *et al.* [2], while extending it from the case of uniform distribution to the case of arbitrary unknown distributions (a.k.a. the distribution-free model). This allows Bshouty [1] to present (optimal) *distribution-free* testers for classes of properties that are sub-classes of k-juntas, which correspond to classes of k-bit long Boolean functions. While Bshouty [1] follows Diakonikolas *et al.* [2] in considering learning algorithms for the underlying classes, we point out that the approach is also applicable to testing algorithms (see [3, Sec. 6.2]).[1]

Let us spell out the task considered by Bshouty [1]. For a class Π of ℓ-bit long Boolean functions and a proximity parameter ϵ, given samples from an unknown distribution $\mathcal{D}$ and oracle access to a function $f : \{0,1\}^\ell \to \{0,1\}$, we wish to distinguish the case that $f \in \Pi$ from the case that f is ϵ-far from Π (i.e., $\min_{g\in\Pi}\{\Pr_{x\sim\mathcal{D}}[f(x) \neq g(x)]\} > \epsilon$). Recall that Π is a (symmetric) class consisting of a symmetric subclass of k-juntas Π'; that is, $f \in \Pi$ if and only if there exists a k-subset $J \subset [\ell]$ and $f' \in \Pi'$ such that $f(x) = f'(x_J)$, where $x_{\{i_1,...,i_k\}} = (x_{i_1}, ..., x_{i_k})$. (We say that the class Π' is symmetric if for every permutation $\pi : [k] \to [k]$ it holds that $f' \in \Pi$ if and only if $f'_\pi \in \Pi'$, where $f'_\pi(x) = f(x_{\pi(1)}, ..., x_{\pi(k)})$.) Actually, we also assume that Π' is closed under setting function-variables to 0; that is, if $f' \in \Pi$, then, for every $i \in [k]$, the function $f^{(i)}$ defined by $f^{(i)}(x_1, ..., x_k) = f'(x_1, ..., x_{i-1}, 0, x_{i+1}, ..., x_k)$ is in Π'.

Notation: For $x \in \{0,1\}^\ell$ and $S \subseteq [\ell]$, we let x_S denote the projection of x on coordinates S (i.e., if $S = \{i_1, ..., i_m\}$ such that $i_1 < \cdots < i_m$, then $x_S = (x_{i_1}, ..., x_{i_n})$). This notation extends to distributions over $\{0,1\}^\ell$.

1 A Bird's Eye View

The basic strategy is to consider a random partition of $[\ell]$ to $m = O(k^2)$ parts, denoted $(S_1, ..., S_m)$, while relying on the fact that, whp, each S_i contains at most one influential variable (i.e., variable in the alleged k-junta). Assuming that $f \in \Pi$, first we determine a set I of at most k indices such that $\cup_{i\in[\ell]\setminus I} S_i$ contains no "significantly influential" variables of f. Suppose that $f' : \{0,1\}^k \to \{0,1\}$ is a function that corresponds to the tested function $f : \{0,1\}^\ell \to \{0,1\}$, and that $I \subset [\ell]$ is indeed the collection of all sets that contain influential variables. The crucial ingredient is devising a method that allows to generates samples of the form $(x', f'(x'))$, when given samples of the form $(x, f(x))$ (for $x \sim \mathcal{D}$). We stress that we cannot afford to find the influential variables, and so this method works without determining these locations. Using this method, we can test whether f'

[1] Actually, while Bshouty also mentions this fact in the middle of his paer (see [1, Thm. 30]). His testers for the k-junta class Π is optimal, since their complexity essentially equals the complexity of testing the underlying class Π' (of k-bit strings functions).

belongs to the underlying class Π'; hence, we test f by implicitly sampling the projection of $\mathcal{D}$ on the (unknown) influential variables.

The method employed by Diakonikolas *et al.* [2] only handles the uniform distribution (i.e., the case that $\mathcal{D}$ is uniform over $\{0,1\}^\ell$), and so it only yields testers for the standard testing model (rather than for the distribution-free testing model). Furthermore, their method as well as the identification of the set I rely heavily on the notion of influence of sets, where the influence of a set S of locations on the value of a function is defined as $\Pr_{x',x''\in\{0,1\}^\ell:x'_S=x''_S}[f(x')\neq f(x'')]$. However, this notion refers to the uniform distribution (over $\{0,1\}^\ell$) and does not seem adequate for the distribution-free context (e.g., for $f(x)=x_1+x_2$ we may get $\Pr_{x',x''\sim\mathcal{D}:x'_1=x''_1}[f(x')\neq f(x'')]=0$).

Indeed, Bshouty [1] uses a different way for identifying the set I and for generating samples for the underlying function f'. Loosely speaking, he identifies I as the set of indices i for which $f(e_{S_i})\neq f(0^\ell)$, where e_S is a string that is 1 on the locations in S and is 0 on other locations. (*Be warned that this description is an over-simplification*!) This means that for every $i\in I$ and $x\in\{0,1\}^\ell$, the value of x at the influential variable in the set S_i (a variable whose location is unknown to us!), equals $f(x')\oplus f(0^\ell)$ where $x'_j=x_j$ if $j\in S_i$ and $x'_j=0$ otherwise.[2] Note that the foregoing holds when $f\in\Pi$; in general, we can test whether $x\mapsto f(x')\oplus f(0^\ell)$ is close to a dictatorship (under the uniform distribution) and reject otherwise, whereas if the mapping is close to a dictatorship we can self-correct it.

To sample the distribution $\mathcal{D}_J$, where J is the influential variables in $S_I=\cup_{i\in I}S_i$, we sample $\mathcal{D}$ and determine the value of the influential variable in each set S_i, for $i\in I$. Queries to the function f' are answered by querying f such that the query $y=y_1\cdots y_k$ is mapped to the query $\mathtt{ext}(y)$ such that $\mathtt{ext}(y)_j=y_i$ if j belongs to the i^{th} set in the collection I (and $\mathtt{ext}(y)_j=0$ if $j\in[\ell]\setminus S_I$). Effectively, we query the function $f^{\mathtt{ext}}:\{0,1\}^\ell\to\{0,1\}$ defined as $f^{\mathtt{ext}}(x)=f(\mathtt{ext}(x_J))$, and this makes sense provided that $f^{\mathtt{ext}}$ is close to f (under the distribution $\mathcal{D}$). To test the latter hypothesis condition we sample $\mathcal{D}$ and for each sample point x we compare $f(x)$ to $f^{\mathtt{ext}}(x)$, where here we again use the ability to determine the value of the influential varaible in each set. Specifically, $\mathtt{ext}(x_J)$ is computed by determining the value of x_J (without knowing J), and using our knowledge of $(S_i)_{i\in I}$.

We warn that the foregoing description presumes that we have correctly identified the collection I of all sets containing an influential variable. This leaves us with two questions: The first question is how do we identify the set I. (Note that the influence of a variable may be as low as 2^{-k}, whereas we seek algorithms of poly(k)-complexity.) The solution (to be presented in Section 2.1) will be randomized, and will have one-sided error; specifically, we may fail to indentify some sets that contain influential variables, but will never include in our collection sets that have no influential variables. Consequently, $f(e_{S_i})\neq f(0^\ell)$ may not hold for some $i\in I$, and (over-simplifying again) we shall seek instead some $w^{(i)}\in\{0,1\}^\ell$ such that $f(w^{(i)})\neq f(v^{(i)})$, where $v^{(i)}_j=w^{(i)}_j$ if $j\in[\ell]\setminus S_i$ and $v^{(i)}_j=0$ otherwise. Second, as before, for every $i\in I$ and $x\in\{0,1\}^\ell$, we wish to determine the value in x of the influential variable in the set S_i (a variable whose location is unknown to us!). This is done by observing that if $f\in\Pi$ then this value equals $f(x')\oplus f(w^{(i)})\oplus 1$ where $x'_j=x_j$

[2] Indeed, if $\tau(i)\in S_i$ is the index of the (unique) influential variable that resides in the set S_i, then

$$f(x')=x_{\tau(i)}\cdot f(e_{S_i})\oplus(x_{\tau(i)}\oplus 1)\cdot f(0^\ell)=x_{\tau(i)}\oplus f(0^\ell)$$

since $f(e_{S_i})\oplus f(0^\ell)=1$.

if $j \in S_i$ and $x'_j = w_j^{(i)}$ otherwise.[3] Again, we need to test whether $x \mapsto f(x') \oplus f(w^{(i)}) \oplus 1$ is a dictatorship, and use self-correction.

2 The Actual Tester

As warned, the above description is an over-simplication, and the actual way in which the set I is identified and used is more complex.

We fix a random partition of $[n]$ to $m = O(k^2)$ parts, denoted $(S_1, ..., S_m)$. If $f \in \Pi$, then, with high probability, each S_i contains at most one influential variable, denoted $\tau(i)$. We assume that this is the case when providing intuition throughot this section.

2.1 Stage 1: Finding I and corresponding $w^{(i)}$

Our goal is to find a collection I of at most k sets such that the function h_I that is $\epsilon/3$-close to f (w.r.t distribution $\mathcal{D}$), where h_I is defined as $h_I(x) = f(x')$ such that x' equals x on $S_I \stackrel{\text{def}}{=} \cup_{i \in I} S_i$ and equals 0^ℓ on $\overline{S_I} = [\ell] \setminus S_I$; that is, $x'_j = x_j$ if $j \in S_I$ and $x'_j = 0$ otherwise (e.g., $h_I(1^\ell) = f(e_{S_I})$). In addition, for each $i \in I$, we seek a witness $w^{(i)}$ for the fact that f depends on some varials in S_i; that is, $f(w^{i)}) \neq f(v^{(i)})$ for some $v^{(i)}$ that differ from $w^{(i)}$ only on S_i.

The procedure. We proceed in iterations, starting with $I = \emptyset$.

1. We sample $\mathcal{D}$ for $O(1/\epsilon)$ times, trying to find $u \sim \mathcal{D}$ such that $f(u) \neq h_I(u)$.

   (Note that if $I = \emptyset$, then $h_I(u) = f(0^n)$. In general, we seek u such that $f(u) \neq f(u')$, where $u'_{S_I} = u_{S_I}$ and $u'_{\overline{S_I}} = 0^{|\overline{S_I}|}$.)

   If no such u is found, then we set $h = h_I$ and proceed to Stage 2. In this case, we may assume that h_I is $\epsilon/3$-close to f (w.r.t $\mathcal{D}$).

2. Otherwise (i.e., $f(u) \neq h_I(u)$), we find an $i \in [m] \setminus I$ and $w^{(i)}$ such that $h_I(w^{(i)}) \neq h_{I \cup \{i\}}(w^{(i)})$, which means that S_i contains an influential variable and $w^{(i)}$ is the witness for the sensitivity that we seek. We set $I \leftarrow I \cup \{i\}$ and proceed to next iteration.

   (We find this i by binary search that seeks i and S such that $h_{I \cup S \cup \{i\}}(u) \neq h_{I \cup S}(u)$, which means that $w^{(i)}$ equals u in locations outside S and is zero on S.)[4]

Once the iterations are suspended (due to not finding u), we reject if $|I| > k$, and continue to the Stage 2 otherwise. Recall that in the latter case $h = h_I$ is $\epsilon/3$-close to f (w.r.t $\mathcal{D}$).

Note that if $f \in \Pi$, then I contain only sets that contain variables of the k-junta, and so we never reject in this stage. In general, if $i \in I$, then $h_{I \setminus \{i\}}(w^{(i)}) \neq h_I(w^{(i)})$, which implies that $f(x') \neq f(x'')$, where x' and x'' differ only on S_i (e.g., $x''_{S_I} = w^{(i)}_{S_I}$ and $x''_j = 0$ if $j \notin S_I$).

[3] Indeed, if $\tau(i) \in S_i$ is the index of the (unique) influential variable that resides in the set S_i, then

$$f(x') = x_{\tau(i)} \cdot f(w^{(i)}) \oplus (x_{\tau(i)} \oplus 1) \cdot f(v^{(i)}) = x_{\tau(i)} \oplus f(w^{(i)} \oplus 1$$

since $f(w^{(i)} \oplus f(v^{(i)} = 1$.

[4] By Step 1, we have $h_{S' \cup I}(u) \neq h_{S'' \cup I}(u)$, for $S' = [\ell] \setminus I$ and $S'' = \emptyset$, and in each iteration we cut $S' \setminus S''$ by half while maintaining $h_{S' \cup I}(u) \neq h_{S'' \cup I}(u)$.

2.2 Stage 2: Extracting the value of an influential variable

Given a collection I as found in Stage 1 (and a sensitivity witness $w^{(i)}$ for each $i \in I$), let $h = h_I$ and recall that h is close to f w.r.t $\mathcal{D}$. For each $i \in I$, given $x \in \{0,1\}^\ell$, we wish to determine the value of x at the influential variable that resides in S_i.

For each $i \in I$, we define $\nu_i : \{0,1\}^{|S_i|} \to \{0,1\}$ such that $\nu_i(z) = h_I(y)$, where $y_{S_i} = z$ and $y_{\overline{S_i}} = w^{(i)}_{\overline{S_i}}$. Suppose that $f \in \Pi$, and recall that $\tau(i) \in S_i$ denotes the location of the influential variable in S_i. Let $\sigma(i)$ denote the index of $\tau(i)$ in S_i (i.e., the $\sigma(i)^{\text{th}}$ element of S_i is $\tau(i)$). Then, in this case, ν_i is either a dictatorship or an anti-dictatorship. In particular, if ν_i is a dictatorship, then $\nu_i(z) = z_{\sigma(i)}$ (and otherwise $\nu_i(z) = z_{\sigma(i)} \oplus 1$).

For each $i \in I$, we test whether ν_i is a dictatorship or anti-dictatorship, where testing is w.r.t the uniform distribution over $\{0,1\}^{|S_i|}$. Note that we also check whether ν_i is a dictatorship or anti-dictatorship. If the tester (run with proximity parameter 0.1) fails, we reject. Otherwise (i.e., if we did not reject), we can compute ν_i via self-correction on h_I; that is, to compute ν_i at z, we select $r \in \{0,1\}^{|S_i|}$ at random, and return $\nu_i(z+r) - \nu_i(r)$, which (w.h.p.) equals $(z+r)_{\sigma(i)} \oplus r_{\sigma(i)} = z_{\sigma(i)}$.

Hence, we always continue to Stage 3 if $f \in \Pi$, and whenever we continue to Stage 3 we can compute all ν_i (for $i \in I$) via self-correction.

2.3 Stage 3: Emulating a tester of Π'

Recall that when reaching this stage, we may assume that $h = h_I$ is $\epsilon/3$-close to f (w.r.t $\mathcal{D}$). Also recall that $h_I(x)$ depends only on x_{S_I}, where $S_I = \cup_{i \in I} S_i$, and that by Stage 2 we may assume that $\nu_i(z) = z_{\sigma(i)}$ (for every $i \in I$ and almost all z). In light of the forgoing, we define $g : \{0,1\}^\ell \to \{0,1\}$ such that $g(x) = h(x')$ where $x'_{S_i} = (x_{\sigma(i)}, ..., x_{\sigma(i)})$ (i.e., $x'_j = (x_{S_i})_{\sigma(i)} = x_{\tau(i)}$ if $j \in S_i$)[5] and $x'_j = 0$ otherwise. (Indeed, if $f \in \Pi$, then $g(x) = h(x)$, since $h(y)$ depends only on $(y_{\tau(i)})_{i \in I}$. Using hypothesis that Π' (and so Π) is closed under setting function-variables to 0, it follows that $g \in \Pi$.)

We observe that if g is $\epsilon/3$-close (w.r.t $\mathcal{D}$) to both h and Π, then f must be ϵ-close to Π (since f is $\epsilon/3$-close to h). Hence, we test both these conditions. Specifically, using our ability to sample $\mathcal{D}$, query f, and determine the value of the influential variables in S_I, we proceed as follows:

1. Test whether $g = h$, where testing is w.r.t the distribution $\mathcal{D}$ and proximity parameter $\epsilon/3$.

   This is done by taking $O(1/\epsilon)$ samples of $\mathcal{D}$, and comparing the values of g and h on these sample points. Recall that $h(u) = h_I(u) = f(u')$, where $u'_{S_I} = u_{S_I}$ and $u'_{\overline{S_I}} = 0^{|\overline{S_I}|}$. The value of g on u is determined as follows.

   (a) For every $i \in I$, if ν_i is a dictatorship, then set v_i to equal the self-corrected value of $\nu_i(u_{S_i})$, where ν_i is as defined in Stage 2. Otherwise (i.e., when ν_i is an anti-dictatorship), we set v_i to equal the self-corrected value of $\nu_i(u_{S_i}) \oplus 1$.

   (b) Return the value $h(u')$, where $u'_j = v_i$ if $j \in S_i$ and $u'_j = 0$ otherwise.

   Indeed, $g = h$ always passes this test, whereas g that is $\epsilon/3$-far from h (w.r.t $\mathcal{D}$) is rejected w.h.p.

[5] In general, $\tau(i)$ denotes the location in $[\ell]$ of the $\sigma(i)^{\text{th}}$ element of S_i.

2. Test whether g is in Π, where testing is w.r.t the distribution $\mathcal{D}$ and proximity parameter $\epsilon/3$. This is done by testing whether g' is in Π, where $g'(z) = g(x)$ such that $x_j = z_i$ if j is in the i^{th} set in the collection I, and $x_j = 0$ otherwise. Here we use a distribution-free tester, and analyze it w.r.t the distribution $\mathcal{D}_I$. Toward this end we need to samples $\mathcal{D}_I$ as well as answer queries to g', where both tasks can be performed as in the prior step.

   Recall that if $f \in \Pi$, then $g \in \Pi$, and this test will accept (w.h.p.), whereas if g is $\epsilon/3$-far from Π the test will reject (w.h.p.).

We conclude that if we reached Stage 3 and $f \in \Pi$ (resp., f is ϵ-far from Π), then we accept (resp., reject) w.h.p.

3 Digest: The new approach of [1] vs the original one [2]

The new approach of Bshouty [1] differs from the original approach of Diakonikolas *et al.* [2] in two main aspects:

1. In [2], sets that contain influential variables are identified according to their influence, which is defined with respect to the uniform distribution. This definition seems inadequate when dealing with arbitrary distributions. Instead Bshouty [1] identifies such a set by searching for two assignments that differ only on this set and yield different function values. The actual process is iterative and places additional constraints on these assignments (as detailed in Section 2.1).

2. In [2], given an assignment to the function, the value of the unique influential variable that resides in a given set S is determined by approximating the influence of two subsets of S (i.e., the subsets of locations assigned the value 0 and 1, respectively). In contrast, Bshouty [1] determines this value by defining an auxiliary function that depends only on the unknown influential variable, and evaluating this function (via self-correction w.r.t the uniform distribution (!); see Section 2.2).

Acknowledgements

I am grateful to Nader Bshouty for numerous clarifications regarding his work [1].

References

[1] N. Bshouty. Almost Optimal Testers for Concise Representations. *ECCC*, TR19-156.

[2] I. Diakonikolas, H.K. Lee, K. Matulef, K. Onak, R. Rubinfeld, R.A. Servedio, and A. Wan. Testing for Concise Representations. In *48th IEEE Symposium on Foundations of Computer Science*, pages 549–557, 2007.

[3] O. Goldreich. *Introduction to Property Testing.* Cambridge University Press, 2017.