A machine-learning approach for accurate detection of copy-number variants from exome sequencing

Vijay Kumar[1], Gopal Jayakar[2], Matthew Jensen[1], Neil Kelkar[2], Santhosh Girirajan[1,3,4]

[1]Bioinformatics and Genomics Graduate Program of the Huck Institutes of the Life Sciences, [2]The Schreyer Honors College, [3]Department of Biochemistry and Molecular Biology, [4]Department of Anthropology, The Pennsylvania State University, University Park, PA, 16802.

Keywords: Copy-number variants, Exome sequencing, Machine learning.

Correspondence to:
Santhosh Girirajan, MBBS, PhD
205A Life Sciences Building
Pennsylvania State University
University Park, PA 16802
Phone: 814-865-0674
E-mail: sxg47@psu.edu

ABSTRACT

Copy-number variants (CNVs) are a major cause of several genetic disorders, making their detection an essential component of genetic analysis pipelines. Current methods for detecting CNVs from exome sequencing data are limited by high false positive rates and low concordance due to the inherent biases of individual algorithms. To overcome these issues, calls generated by two or more algorithms are often intersected using Venn-diagram approaches to identify "high-confidence" CNVs. However, this approach is inadequate, as it misses potentially true calls that do not have consensus from multiple callers. Here, we present CN-Learn, a machine-learning framework (https://github.com/girirajanlab/CN_Learn) that integrates calls from multiple CNV detection algorithms and learns to accurately identify true CNVs using caller-specific and genomic features from a small subset of validated CNVs. Using CNVs predicted by four exome-based CNV callers (CANOES, CODEX, XHMM and CLAMMS) from 503 samples, we demonstrate that CN-Learn identifies true CNVs at higher precision (~90%) and recall (~85%) rates while maintaining robust performance even when trained with minimal data (~30 samples). CN-Learn recovers twice as many CNVs compared to individual callers or Venn diagram-based approaches, with features such as exome capture probe count, caller concordance and GC content providing the most discriminatory power. In fact, about 58% of all true CNVs recovered by CN-Learn were either singletons or calls that lacked support from at least one caller. Our study underscores the limitations of current approaches for CNV identification and provides an effective method that yields high-quality CNVs for application in clinical diagnostics.

INTRODUCTION

Copy-number variants (CNVs) are a major source for genomic variation, evolution, and disease (Sebat et al. 2004; Redon et al. 2006; Perry et al. 2008; Girirajan et al. 2011). About 15% of affected individuals referred for clinical genetic testing carry a disease-associated CNV (Miller et al. 2010), making CNV detection an essential aspect of genetic analysis pipelines (Sathirapongsasuti et al. 2011; Krumm et al. 2012). Although the clinical utility of microarrays has not diminished (Coughlin et al. 2012), exome sequencing is becoming a prevalent technology for genetic testing (Yang et al. 2013; Stark et al. 2016; Tan et al. 2017; Retterer et al. 2016; de Ligt et al. 2013). Several algorithms are available to call CNVs from exome sequencing data in both clinical (Retterer et al. 2016) and disease-specific cohorts, including autism (Krumm et al. 2012), schizophrenia (Fromer et al. 2012), epilepsy (Epilepsy Phenome/Genome Project & Epi4K Consortium 2015), and cancer (Koboldt et al. 2012). A common strategy employed by these CNV callers is to apply various statistical distributions to model the aggregate read depth of the exons and use read depth fluctuations between adjacent exons to identify duplication or deletion events (Backenroth et al. 2014; Jiang et al. 2015; Packer et al. 2015; Krumm et al. 2012; Fromer et al. 2012).

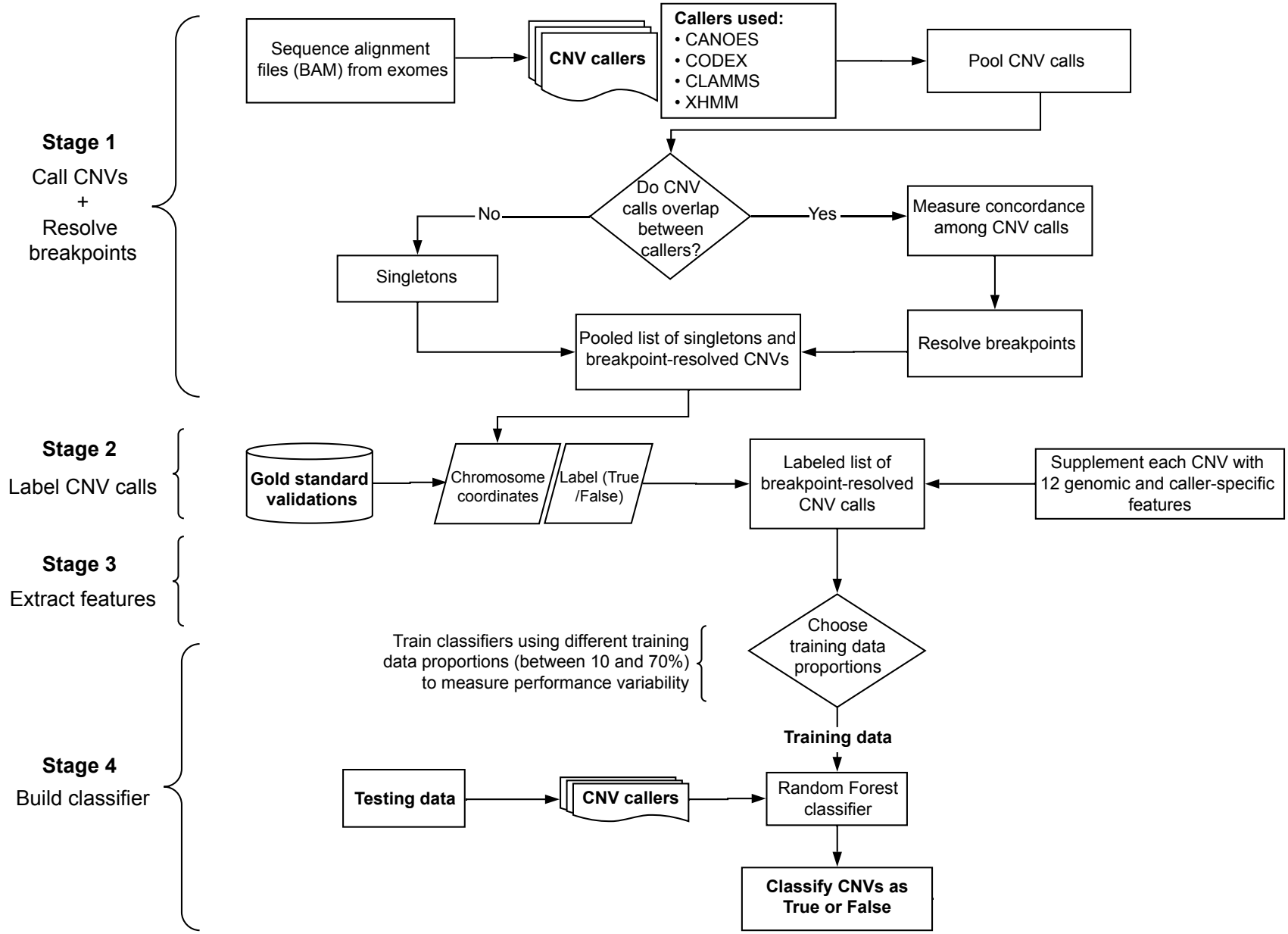
Several themes have emerged due to variations in the approaches employed by different CNV callers to model read-depth distributions. *First*, the distributions and algorithms chosen to model read depth depend on the expertise of the researchers and their subjective assumptions about the underlying data. For example, callers such as XHMM (Fromer et al. 2012) assume the read-depth distribution to be Gaussian, while CANOES (Backenroth et al. 2014) assumes a negative binomial distribution, CODEX (Jiang et al. 2015) assumes a Poisson distribution, and CoNIFER (Krumm et al. 2012) makes no assumptions about the read-depth distribution. *Second*, while every method normalizes data to eliminate noise and outliers resulting from GC and repeat content biases, the number of samples required for normalization and the definition of outliers are inconsistent among the callers. For example, a principal component analysis (PCA) based method such as XHMM requires at least 50 unrelated samples for effective normalization, while CANOES only requires as low as 15 samples (Backenroth et al. 2014). Similarly, the annotations for "extreme" GC content differ among XHMM (<0.1 or >0.9), CODEX (<0.2 or >0.8), and CLAMMS (<0.3 or >0.7) (Packer et al. 2015). Further, XHMM only considers exome-capture targets between the size range of 10 bp and 10 kbp and with average coverage >10X across all

75 samples, while CODEX uses targets that are >20 bp long and with median coverage >20X.
76 CODEX eliminates targets with mappability scores <0.9, while CLAMMS eliminates regions
77 with scores <0.75 along with a custom list of "blacklisted" regions (Packer et al. 2015). *Fourth*,
78 the validation methods and the subsets of CNV calls used to estimate sensitivity and specificity
79 measures vary widely among the callers. For example, large CNVs (>100 kbp) validated with
80 microarrays and analyzed with PennCNV (Wang et al. 2007) were used to estimate the
81 performance of CANOES (Backenroth et al. 2014), while only a subset of CLAMMS calls with
82 minor allele frequency <0.1 were validated using PCR. Also, not all callers report confidence
83 scores for the CNVs they identify. Even when they do, their confidence scales are not directly
84 comparable. These issues influence the number and type of CNVs detected by each caller in a
85 given sample, resulting in pronounced differences in accuracy, false positive rates, and
86 concordance among the callers (Hong et al. 2016; Yao et al. 2017). *Finally*, studies using exome-
87 sequencing data to detect CNVs either utilize predictions made by a single caller (Krumm et al.
88 2013; Poultney et al. 2013) or use a Venn-diagram approach to identify calls with concordance
89 among multiple callers as "high-confidence" CNVs (Kataoka et al. 2016; Priest et al. 2016;
90 Bademci et al. 2016; Krumm et al. 2015). While using data from multiple callers minimizes false
91 positive rates, this approach discards a large subset of non-concordant CNVs, thereby reducing
92 the overall CNV yield (Hong et al. 2016). In addition to a low CNV yield, the reported
93 breakpoints of the concordant calls do not necessarily agree between callers. These limitations
94 associated with individual CNV callers as well as the methods used to integrate predictions from
95 different CNV callers necessitate a better approach to identify and prioritize clinically relevant
96 CNVs.
97 Here, we propose a machine-learning method called CN-Learn that overcomes the
98 limitations of high false positive and low concordance rates among calls generated by different
99 CNV algorithms to identify high-confidence CNVs. Our method leverages several attributes
100 intrinsic to each CNV call, such as GC content, mappability of the genomic region, and CNV
101 size, in addition to concordance among the callers. CN-Learn learns the associations between
102 these attributes and the presence or absence of CNVs using a small subset of validated CNVs in
103 the cohort (known truth), and then segregates true CNVs from false positives in the test samples
104 with high precision. Using exome-sequencing data and validations from 503 samples, we
105 demonstrate CN-Learn's ability to recover more than twice as many potentially true variants

106 compared to a Venn-diagram approach. Our study reiterates the limitations of existing CNV
107 detection and integration methods, and offers a better alternative that yields a set of high-quality
108 CNVs for application in clinical diagnostics.
109
110 **RESULTS**
111 We developed CN-Learn as a binary Random Forest classifier that can be trained to differentiate
112 true CNV predictions from false positives using a small subset of validated CNVs (**Figure 1**).
113 We identified twelve features that represented the extent of support from individual algorithms
114 and the genomic context for each CNV (see Methods). We detected statistically significant
115 correlations for several pairs of the quantitative features (**Figure 2; Supplemental Table 1**).
116 Based on a collection of decision trees built using the twelve features extracted from CNVs in
117 the training samples, CN-Learn estimates the probability of each CNV in the test sample to be
118 true (**Figure 2**). In addition to a Random Forest classifier, CN-Learn can also be built as a
119 Logistic Regression (LR) or Support Vector Machine (SVM) based classifier (**Supplemental**
120 **file**).
121
122 **CN-Learn detects high-confidence CNVs with high precision and recall rates**
123 To build the CN-Learn classifier, we first identified 41,791 CNV predictions from 503 samples
124 using four exome-based CNV callers (CANOES, Codex, CLAMMS, and XHMM). Using a read-
125 depth based method (**Supplemental Fig. S1)** to resolve breakpoint conflicts of overlapping CNV
126 predictions obtained from different callers, we identified 29,101 unique CNV events among the
127 503 samples (**Supplemental file**). An alternate approach to resolve breakpoint conflicts
128 (**Supplemental Fig. S2)** also provided the same number of unique CNV events. We selected
129 2,506 of these CNVs from 291 samples with microarray validations that were between 50 kbp
130 and 5 Mbp and spanned regions covered by microarray probes (**Supplemental Fig. S3)**. After
131 determining the proportion of CNVs that overlapped with microarray validations at different
132 thresholds (**Supplemental Table 2**), we labeled each of the selected CNVs as either "true" or
133 "false" based on a 10% reciprocal overlap threshold. We next used CNVs from 70% of the 291
134 samples to train CN-Learn as a Random Forest classifier and the remaining 30% of samples to
135 test its performance. Given the uneven distribution of the labels between the two CNV classes
136 (11% true vs. 89% false), we chose precision and recall as the measures of classifier performance

Figure 1

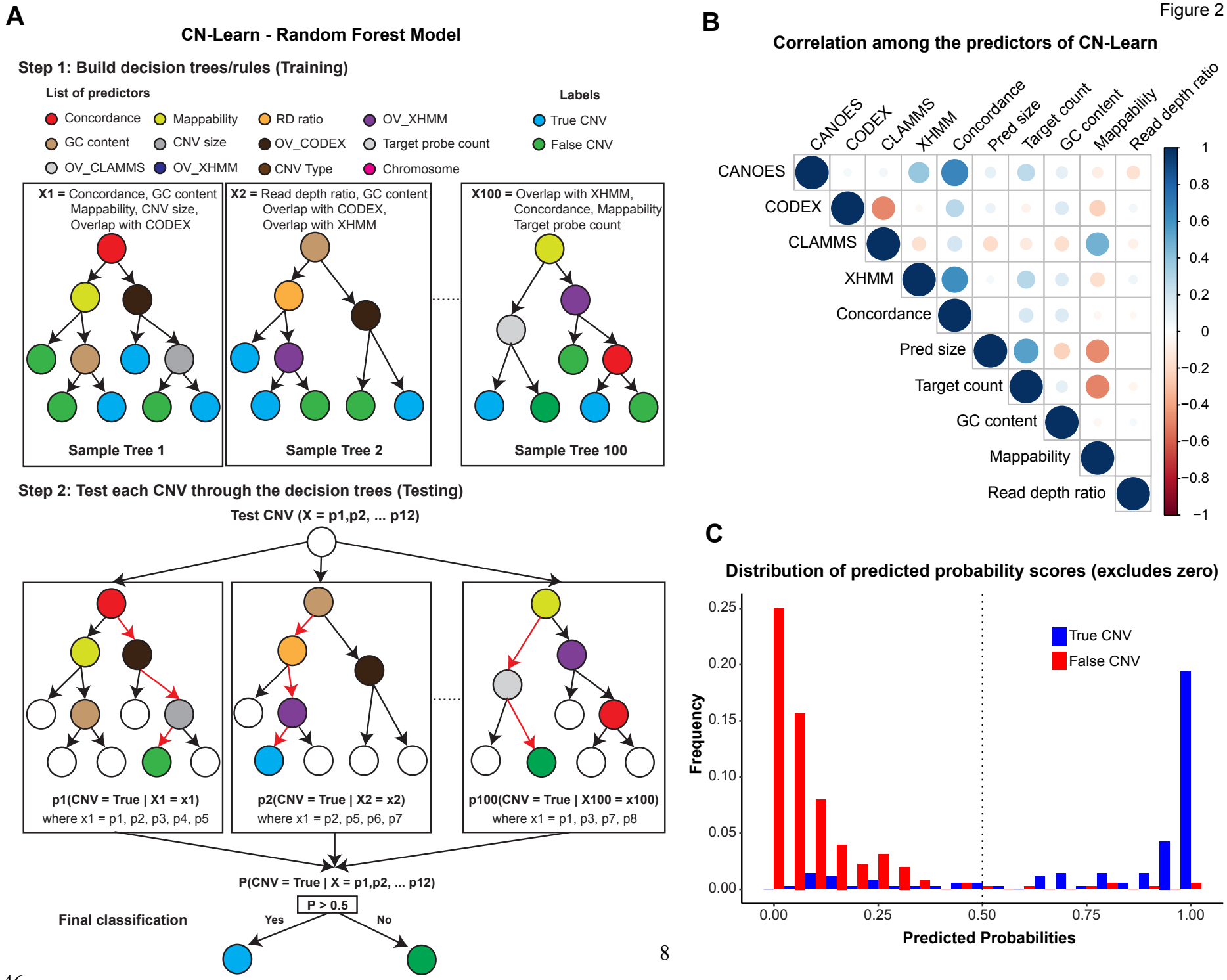
CN-Learn - Conceptual workflow



137

138 **Figure 1: Overview of the CN-Learn pipeline.** The CN-Learn pipeline consists of pre-processing steps (Step 1 and 2), followed by
139 building the classifier using training data and discriminatory features, and finally running the classifier on the test data. The complete
140 pipeline is outlined as follows. ***Step 1*****:** CNV predictions were made using four exome-based CNV callers. While CANOES, CODEX,
141 CLAMMS and XHMM were used in this study, a generic pipeline can be constructed with a different set or number of callers.
142 Breakpoints of overlapping calls from multiple callers were then resolved. ***Step 2*****:** Breakpoint-resolved CNVs were labeled as "true"
143 or "false" based on the overlap with "gold standard" calls and subsequently used to train CN-Learn. ***Step 3*****:** Caller-specific and
144 genomic features were extracted for the labeled CNVs in the training and testing set. ***Step 4*****:** CN-Learn was trained as a Random
145 Forest classifier using the extracted features of the CNVs in the training set to make predictions on the CNVs from the testing set.

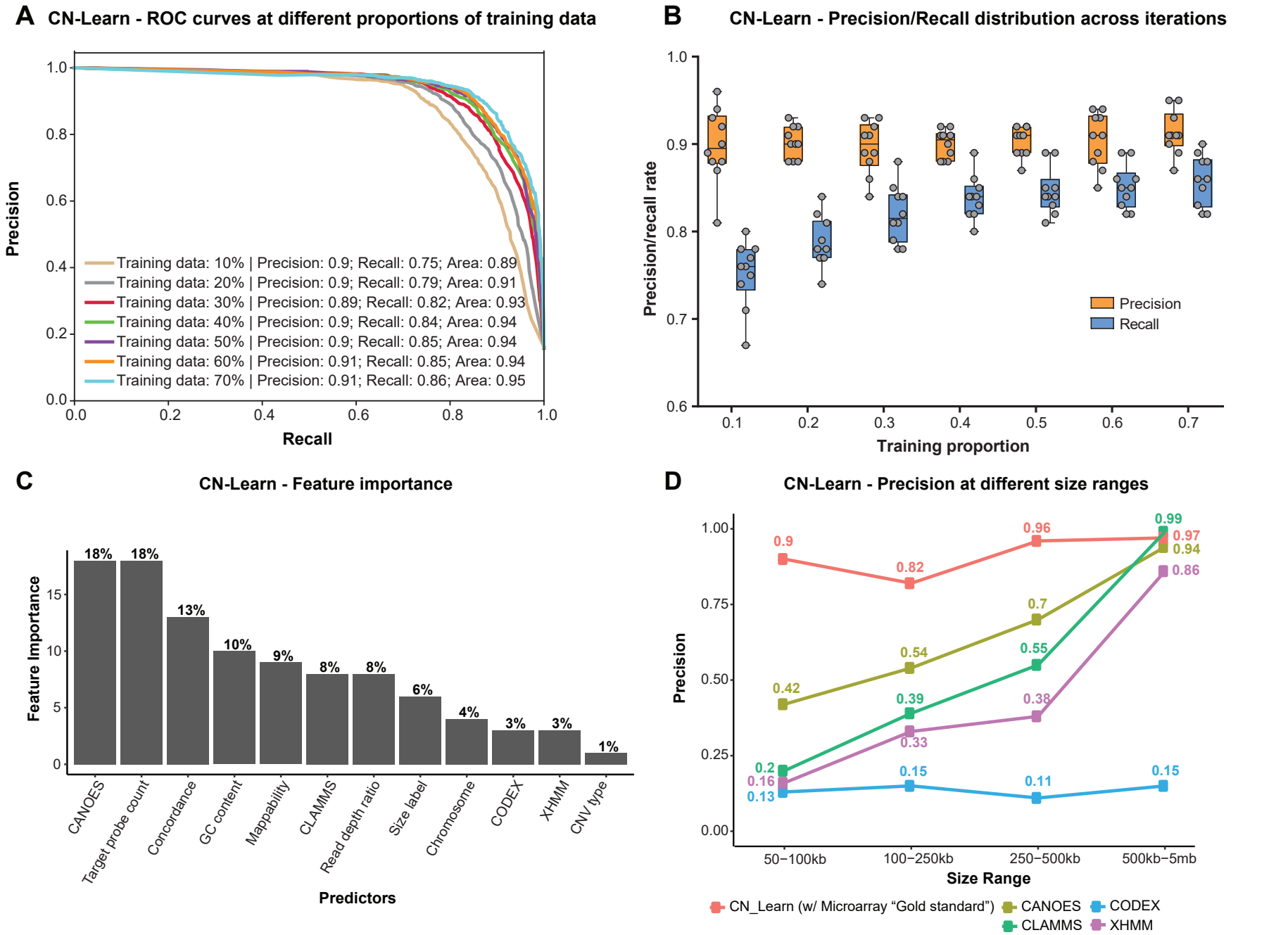
Figure 2


A
CN-Learn - Random Forest Model
Step 1: Build decision trees/rules (Training)
List of predictors
Concordance
Mappability
RD ratio
OV_XHMM
GC content
CNV size
OV_CODEX
Target probe count
OV_CLAMMS
OV_XHMM
CNV Type
Chromosome
Labels
True CNV
False CNV
X1 = Concordance, GC content Mappability, CNV size, Overlap with CODEX
X2 = Read depth ratio, GC content Overlap with CODEX, Overlap with XHMM
X100 = Overlap with XHMM, Concordance, Mappability Target probe count
Sample Tree 1
Sample Tree 2
Sample Tree 100
Step 2: Test each CNV through the decision trees (Testing)
Test CNV (X = p1,p2, ... p12)
p1(CNV = True | X1 = x1)
where x1 = p1, p2, p3, p4, p5
p2(CNV = True | X2 = x2)
where x1 = p2, p5, p6, p7
p100(CNV = True | X100 = x100)
where x1 = p1, p3, p7, p8
P(CNV = True | X = p1,p2, ... p12)
P > 0.5
Final classification
Yes
No
B
Correlation among the predictors of CN-Learn
CANOES
CODEX
CLAMMS
XHMM
Concordance
Pred size
Target count
GC content
Mappability
Read depth ratio
1
0.8
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−0.8
−1
C
Distribution of predicted probability scores (excludes zero)
Frequency
0.25
0.20
0.15
0.10
0.05
0.00
True CNV
False CNV
0.00
0.25
0.50
0.75
1.00
Predicted Probabilities

147 **Figure 2: Illustration of the Random Forest model used to build CN-Learn. (A)** The inner workings of the random forest model
148 used for training CN-Learn is shown. Twelve features were used to grow 100 trees with different subsamples of predictors and
149 training data to classify each CNV in the test set as either "True" or "False". If the predicted probability score was greater than 0.5, the
150 CNV call was classified as "True". Calls with predicted probability score <0.5 were labeled as "False". (**B)** A Spearman rank
151 correlation between pairs of quantitative predictors used by the CN-Learn classifier is shown. The color of the circles indicates the
152 direction of the correlation, while the size of the circles indicates the strength of the correlation. The correlation scores are provided in
153 the supplemental file (**Supplemental Table 1**). (**C)** The frequency of microarray validated and invalidated CNVs, distributed across
154 20 bins of increasing predicted probability scores, is shown. For the probability bins less than 0.5, the proportion of CNVs that were
155 validated was higher than the proportion of CNVs that was not validated. This indicated that classification score of 0.5 is an
156 appropriate threshold for distinguishing "true" and "false" CNVs.

157 (**Supplemental Table 2**). We captured the aggregate performance of CN-Learn using ten
158 random draws of training data (10-fold cross validation), stratified by sample. CN-Learn used the
159 twelve predictors supplied with each CNV in the test set (see Methods) and classified the calls as
160 either "true" or "false" at 91% precision and 86% recall rates. The overall diagnostic ability of
161 the binary classifier, measured as the area under the Receiver Operating Characteristic (ROC)
162 curve, was 95% (**Figure 3A**). Despite sampling different sets of training data for each iteration
163 during cross-validation, the performance of CN-Learn was consistent across all ten iterations for
164 both precision (91±5%) and recall rates (86±5%) (**Figure 3B**).
165
166 We also assessed the relative importance of each feature towards making accurate CNV
167 predictions by calculating the Gini index (defined as the total decrease in improvements to the
168 node purity for all splits on each feature averaged over all trees in the forest) (Breiman 2001).
169 We found that features such as the number of exome capture probes spanning a given call, the
170 extent to which CANOES agreed with a CNV prediction, concordance among the callers, and
171 GC content provided the most discriminatory power to the classifier (**Figure 3C**). Post-
172 classification analysis of the concordance profile indicated that only 34% of all CNVs classified
173 as "true" had support from all the four callers, while the remaining 66% lacked support from at
174 least one caller (**Supplemental Fig. S4**). Overall, these results highlight the ability of CN-Learn
175 to look beyond the single measure of concordance typically used in a Venn-diagram based
176 approach, and to utilize the discriminatory power of additional variables to identify high-
177 confidence CNVs in a systematic manner.
178
179 **Performance of CN-Learn is robust across varying CNV sizes and training sets**
180 We independently trained CN-Learn using varying proportions of training data (between 10%
181 and 70% in increments of 10%) and observed steady performance gains with increase in the
182 number of training samples (**Figure 3A**). Interestingly, even when the classifier was built using
183 just 10% of the total samples (n=29 samples), we obtained 90% precision and a recall rate of
184 75%, indicating the robustness of the classifier when learning from minimal training data. We
185 further trained CN-Learn independently at four size ranges of CNVs and observed a modest
186 increase in precision with increase in CNV size (90% for 50-100 kbp CNV to 97% for 0.5-5 Mbp
187 CNV) (**Figure 3D**). In fact, the precision achieved by CN-Learn at each size interval was

Figure 3


A
CN-Learn - ROC curves at different proportions of training data
Precision
Recall
Training data: 10% | Precision: 0.9; Recall: 0.75; Area: 0.89
Training data: 20% | Precision: 0.9; Recall: 0.79; Area: 0.91
Training data: 30% | Precision: 0.89; Recall: 0.82; Area: 0.93
Training data: 40% | Precision: 0.9; Recall: 0.84; Area: 0.94
Training data: 50% | Precision: 0.9; Recall: 0.85; Area: 0.94
Training data: 60% | Precision: 0.91; Recall: 0.85; Area: 0.94
Training data: 70% | Precision: 0.91; Recall: 0.86; Area: 0.95
B
CN-Learn - Precision/Recall distribution across iterations
Precision/recall rate
Training proportion
Precision
Recall
C
CN-Learn - Feature importance
Feature Importance
Predictors
18%
18%
13%
10%
9%
8%
8%
6%
4%
3%
3%
1%
CANOES
Target probe count
Concordance
GC content
Mappability
CLAMMS
Read depth ratio
Size label
Chromosome
CODEX
XHMM
CNV type
D
CN-Learn - Precision at different size ranges
Precision
Size Range
50−100kb
100−250kb
250−500kb
500kb−5mb
CN_Learn (w/ Microarray "Gold standard")
CANOES
CODEX
CLAMMS
XHMM

188

189 **Figure 3: Characteristics of the CN-Learn binary Random Forest classifier. (A)** Receiver operating characteristic (ROC) curves
190 indicating the trade-off between the precision and recall rates when CN-Learn was trained as a Random Forest classifier are shown.
191 Each curve represents the performance achieved when using different proportions of samples to train CN-Learn, starting from 10% up
192 to 70% in increments of 10%. The results shown were from experiments aggregated across 10-fold cross validation. (**B)** Variability
193 observed in the precision and recall measures during the 10-fold cross validation at various proportions of training data is shown. Both
194 measures varied within $\pm$5% of their corresponding averages. **(C)** Precision rates for CNVs when CN-Learn was trained at four
195 different size ranges compared to the precision rates of CNVs from individual callers are shown. Precision rates for CN-Learn were
196 estimated as its classification accuracy (True positives/ [True positives + False positives]), while the precision rates for the individual
197 callers were calculated as the proportion of CNVs at each size range that were validated by the microarray calls. (**D)** The relative
198 importance of each genomic and caller-specific feature supplemented to CN-Learn is shown. Data shown here are the averages of the
199 values obtained across 10-fold cross-validation after using 70% of the samples for training.

200 substantially higher than the precision achieved by the individual CNV callers. We also observed
201 comparable precision and recall rates when CN-Learn was run on breakpoint-resolved CNVs
202 obtained by merging overlapping predictions (**Supplemental Fig. S5**). While the performance of
203 the Random Forest classifier was robust, Logistic Regression & SVM classifiers failed to match
204 its performance (**See Supplemental file**; **Supplemental Figs. S6 & S7**).
205 We next assessed the performance of CN-Learn by considering calls made by CLAMMS
206 as the truth set and classified CNVs obtained from predictions made by the other three callers
207 (CANOES, CODEX and XHMM). We categorized 25,019 breakpoint-resolved CNVs
208 (**Supplemental Fig. S3**) from all 503 samples as either "true" or "false" based on their
209 intersection (10% reciprocal overlap) with CNVs predicted by CLAMMS. Given the higher
210 resolution of CNVs detected from exome data relative to SNP microarrays, we used a total of
211 16,497 CNVs between 5 kbp and 5 Mbp in size to build CN-Learn (**Supplemental Fig. S3**). CN-
212 Learn achieved an aggregate precision rate of 94% with an overall recall rate of 85% during the
213 10-fold cross validation, and achieved comparable performance when independently trained with
214 CNVs at different training sample proportions (**Supplemental Fig. S8A**). Performance
215 variability observed both during cross-validation and across the size ranges were comparable to
216 the variability observed when microarray was used as the truth set (**Supplemental Figs. S8B &**
217 **S8C**). Among the features used by CN-Learn to classify the CNVs, the relative importance of the
218 mappability score was the highest, with GC content being the next important feature
219 (**Supplemental Fig. S8C**). While caller-specific features contributed to the discriminatory power
220 of CNV classification when microarray was used for validation, genomic features played a more
221 prominent role when a sequence-based method was used for validation (**Supplemental Fig. S9**).
222 These results show that the performance of CN-Learn is robust with minimal training data, at
223 different size ranges and even when orthogonal validations are not available.
224
225 **CN-Learn recovers true CNVs that lack complete concordance among callers**
226 To assess the ability of CN-Learn to correctly identify true CNVs that lack support from multiple
227 callers, we analyzed the concordance profile of all CNVs classified by CN-Learn as "true", based
228 on microarray validations, before and after CN-Learn classification. Using a single random draw
229 of 29 samples (10%) to train CN-Learn and 262 samples as the test set, we obtained predictions
230 for 2,245 CNVs with microarray validations (**Figure 4A)**. Among these predictions, only

231 122/2,245 CNVs (5.4%) were supported by all four CNV callers prior to classification by CN-
232 Learn. The strong concordance of the four methods for these CNV predictions was corroborated
233 by a high microarray validation rate (116/122, 95%) (**Figure 4A**). In contrast, CNVs that lacked
234 support from one or more callers were less likely to intersect with microarray data. For example,
235 only 41% (11/27) of the CNVs with support from XHMM, CODEX and CLAMMS intersected
236 with microarray data. After classification by CN-Learn, 84% (266/315) of all CNVs labeled as
237 "true" intersected with microarray calls, indicating the high classification accuracy achieved by
238 the classifier (**Figure 4A**). All CNVs classified as "true" from this analysis are provided in
239 **Supplemental Table 3**. Of the 266 CNVs validated by microarrays and correctly identified as
240 "true" by the classifier, only 42% (112/266) were supported by all four CNV callers. The
241 remaining 58% (154/266) were either singletons or calls that lacked support from at least one
242 caller, and would have been excluded if complete concordance was used as the only determinant
243 for selecting high confidence CNVs. For example, 13% of all true CNVs (35/266) recovered by
244 CN-Learn were missed by CLAMMS but were identified by one or more of the other callers,
245 reiterating the limitations of using a single exome-based CNV caller for variant predictions.
246 Further, CN-Learn managed to recover 97% (112/116) of the true CNVs validated by
247 microarrays that were supported by all of the four callers. Although these 112 CNVs could have
248 been identified by a simple caller intersection approach, CN-Learn was uniquely able to recover
249 CNVs that lacked support from at least one other caller. For example, CN-Learn classified 66
250 CNVs supported by CANOES, CODEX and XHMM as "true", of which 51% (34/66) were also
251 validated by microarrays. This result is notable because, without using CN-Learn for
252 classification, only 31% (46/148) of CNVs supported by CANOES, CODEX and XHMM would
253 intersect with the microarray calls, indicating the inherently high false positive rate associated
254 with simply intersecting calls from individual callers using a Venn diagram. Our results indicate
255 that in addition to correctly identifying almost every "true" CNV reported by the four callers,
256 CN-Learn overcame the limitations of the Venn-diagram based approach and recovered 154
257 additional high-confidence CNVs with sub-optimal concordance, thereby improving the CNV
258 yield by 2.37-fold (266/112). Further, the true positive calls recovered by CN-Learn spanned the
259 entire spectrum of CNV size range (**Figure 4B**). We also obtained comparable improvements in
260 CNV yield when CNVs predicted by CLAMMS were used as the truth set (**Supplemental file**;
261 **Supplemental Figs. S10 & S11, Supplemental Table 3**).

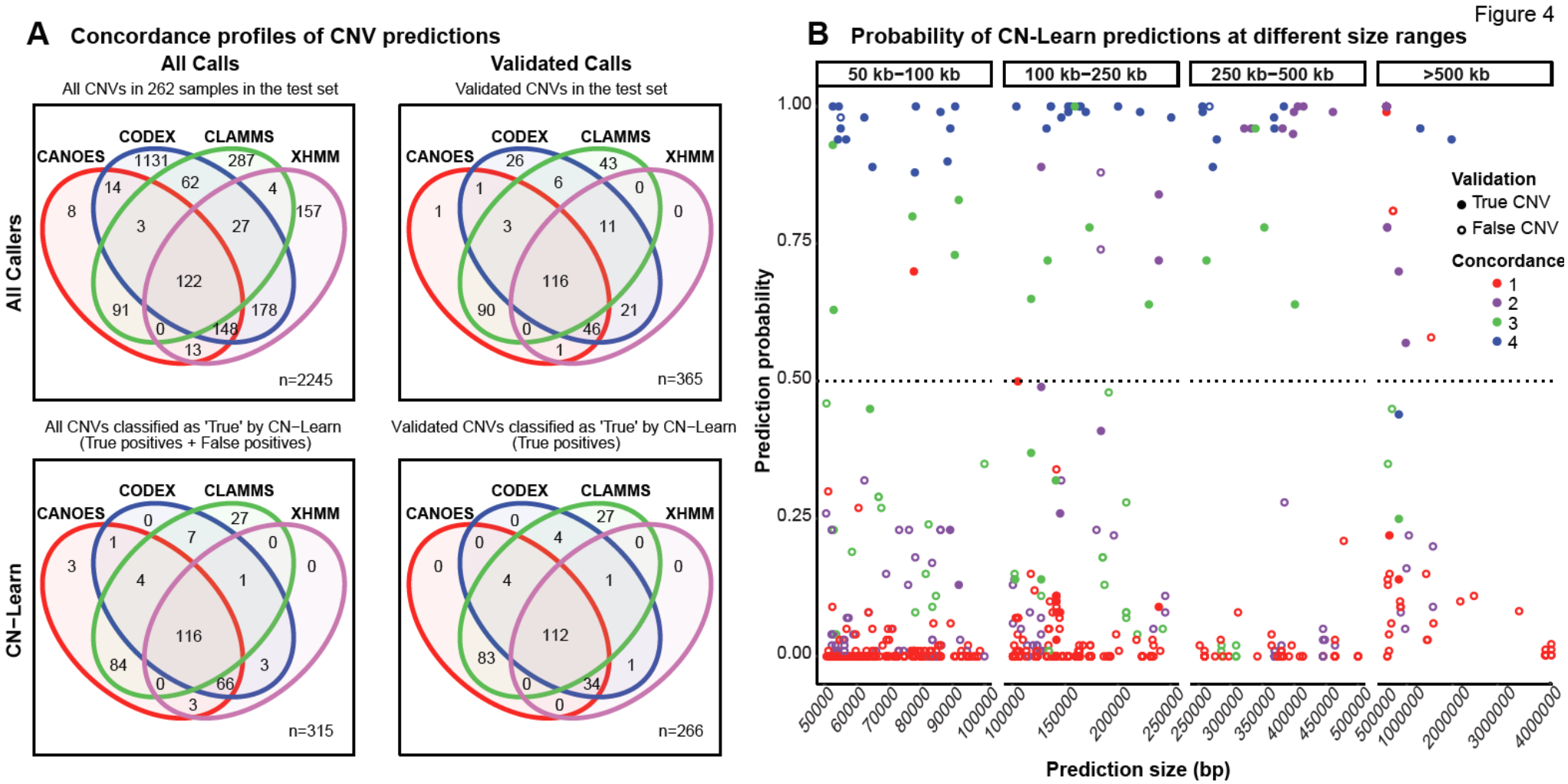


Figure 4: Concordance profile of CNVs before and after classification by CN-Learn using microarray calls as the "gold standard" validation set. (A) Venn diagrams are shown for CNVs (≥50 kbp) identified from a random draw of 262 samples out of the total 291 samples before (**Top panels**) and after (**Bottom panels**) classification by CN-Learn. The Venn diagrams show the overlap of calls among the four callers (**Top left**) versus those that were validated by microarrays (**Top right**). Venn diagrams depicting the overlap of all "true" calls among the four callers after classification by CN-Learn (**Bottom left**) and "true" calls that were also validated by microarrays are shown (**Bottom right**). (**B**) The distribution of all calls within the 262 samples based on the

269 probability scores (Y-axis) predicted by CN-Learn across four size ranges (X-axis) is shown. The number of overlaps of each CNV
270 with exome-based CNV callers is represented by different colors. CNVs that validated with microarrays are indicated by filled circles,
271 while CNVs that did not validate with microarrays are represented by hollow circles.

272 **DISCUSSION**

273 Exome sequencing is a cost-effective assay to accurately identify single base pair mutations and
274 small insertions or deletions within protein-coding genes, which are more likely to cause disease
275 than mutations in other regions of the genome (Bamshad et al. 2011). Even with the emergence of
276 whole genome sequencing (WGS) techniques, exome sequencing has continued to flourish,
277 resulting in an increase in the availability of datasets for analysis. This widespread data availability
278 has allowed for repurposing data generated for single nucleotide variant detection to identify CNVs
279 as well (Koboldt et al. 2013). Given the high false positive rates among calls reported by individual
280 exome-based callers (Tan et al. 2014), existing CNV identification pipelines that leverage more than
281 one calling algorithm have depended on naïve Venn diagram-based integration approaches to
282 identify high-confidence CNVs. One reason for the overwhelming trust in such approaches could be
283 the fact that the self-reported performance measures of each individual algorithm are typically high.
284 Therefore, there is little reason to doubt that performance variations among the callers could affect
285 the precision of Venn diagram-based integration approaches. While we found that the likelihood of
286 a CNV prediction to be true increased with an increase in the number of callers supporting it
287 (**Supplemental Fig. S12**), there are two key limitations of this approach. *First*, even among
288 completely concordant predictions, the observed false positive rates were not zero. Inflated false
289 positives pose a large hurdle for researchers interested in identifying clinically relevant CNVs, as it
290 is time consuming to validate a large number of false positive CNVs using orthogonal methods.
291 *Second*, the Venn diagram approach failed to identify a large subset of non-concordant or singleton
292 CNVs supported by microarray validations. In fact, approximately one true CNV (>50 kbp) per
293 individual in our cohort would have been discarded under a Venn-diagram based approach. This
294 evidence reiterates that Venn diagram-based approaches do not have the required precision for
295 usage in both clinical and research settings.

296 The utility of alternate methods for CNV detection hence rests on the ability to both
297 eliminate false positives among completely concordant predictions and recover true CNVs that lack
298 adequate support from multiple callers. Therefore, instead of addressing the shortcomings of
299 existing methods by developing yet another CNV detection tool, our study focused on offering a
300 reliable integrative approach. In this study, we demonstrate a machine-learning approach that
301 leverages caller-specific and genomic contexts from a subset of validated calls to identify high-
302 confidence CNVs more thoroughly than individual callers on their own or Venn diagram-based

303 approaches. CN-Learn achieved precision as high as 94% while doubling the CNV yield, showing
304 its ability to capture singletons that would have been missed by other approaches. Importantly, the
305 precision of CN-Learn is robust to variation in CNV size, ratio of training to testing samples, and
306 validation method/type, indicating the utility of CN-Learn in a variety of clinical or research
307 contexts. The use of multiple variables that capture the genomic context of each CNV in addition to
308 caller concordance is a major reason for the high recovery and precision rates achieved by CN-
309 Learn. In fact, GC content and mappability scores were some of the most useful features in
310 predicting true CNVs.

311 One of the limitations of CN-Learn is its dependency on a small set of biologically validated
312 CNVs. Our study leverages microarray validations to evaluate the predictions of each caller and to
313 label the breakpoint-resolved CNV predictions as either "true" or "false". While microarrays may
314 not be considered as a "gold standard" for CNV detection, we were able to use microarray calls as
315 an orthogonal validation to demonstrate the utility of CN-Learn. Our study also serves as a proof-of-
316 principle for future studies that could utilize CN-Learn with "gold-standard" CNVs curated from
317 multiple genomic technologies, such as PacBio SMRT sequencing (John et al. 2009), Illumina long-
318 read sequencing (Voskoboynik et al. 2013), 10X linked-read sequencing (Zheng et al. 2016),
319 BioNano Genomics genome mapping (Lam et al. 2012; Mak et al. 2016) or PCR. Another limitation
320 of our study is the time and computational capacity required to run four different CNV callers. Each
321 CNV calling algorithm extracts read-depth information for each sample using tools such as
322 Samtools (Li et al. 2009), Bedtools (Quinlan and Hall 2010) or Genome Analysis Toolkit (GATK)
323 (McKenna et al. 2010), which are often the rate-limiting steps for each pipeline. Future studies
324 could simplify this data extraction layer by using a single read-depth tool without adversely
325 impacting the results of the individual callers. Finally, we also acknowledge the complexity
326 associated with resolving breakpoint conflicts from multiple callers that arise during data
327 integration. While we presented two strategies to resolve breakpoints of concordant CNVs (see
328 Methods), future studies could explore more effective strategies before using CN-Learn. As
329 population-scale projects continue to generate large exome-sequencing datasets, the need and
330 importance of robust CNV integration approaches such as CN-Learn is apparent.

331 Overall, CN-Learn integrates predictions from multiple CNV callers and overcomes the
332 limitations of existing integration approaches, even when the availability of samples with biological
333 validations is limited. Although we chose a set of four CNV calling algorithms with microarray

validations, CN-Learn framework can be extended to use different sets of CNV callers or validation types to identify high-confidence CNVs, making our framework easy to adopt and customize. This can allow clinicians and researchers to use their preferred callers and validation methods to detect CNVs from exomes. Our results suggest that a small set of high-quality validated CNVs and an objective machine learning method can help alleviate several shortcomings of existing integration approaches to generate an informed set of clinically relevant CNVs.

341 **METHODS**

342 **Samples**

343 We obtained exome sequencing data from 503 individuals in the Simons Variation in Individual
344 Project (SVIP), which were generated using the Agilent Sure-Select Human All Exon v2.0 capture
345 kit containing 182,430 autosomal probes (Spiro and Chung 2012). Overall, 187 individuals within
346 this cohort carry the 16p11.2 deletion, and 143 individuals carry the reciprocal duplication. Single
347 nucleotide polymorphism (SNP) based microarray data were available for 291 samples, and CNVs
348 >100 kbp were confirmed experimentally using array CGH (Duyzend et al. 2016).

349

350 **Exome CNV callers**

351 We chose four exome CNV callers to obtain the initial sets of CNV calls: CANOES, CLAMMS
352 (v1.1), CODEX (v0.2.2), and XHMM (Backenroth et al. 2014; Packer et al. 2015; Jiang et al. 2015;
353 Fromer et al. 2012). As the SVIP samples were sequenced in two sets (312 and 191 samples), each
354 set of samples was treated as individual batches for running the CNV-calling pipelines. CANOES
355 and CODEX were run using the default parameters. CLAMMS models were built with the
356 assumption that the samples were independent without accounting for batch effects, and XHMM
357 was run with a "PVE_mean_factor" parameter value of 0.7. Running the four CNV callers yielded
358 41,791 calls of varying sizes (**Supplemental Fig. S13**). These original CNV calls (deletions and
359 duplications) were then characterized based on their level of concordance among the four callers
360 (**Supplemental file**).

361

362 **Resolving breakpoint conflicts for overlapping predictions**

363 Multiple CNV callers can make predictions that overlap with each other in a given genomic region.
364 Treating such overlapping predictions with different breakpoints as separate CNVs would result in
365 double-counting the calls for the same CNV event. Therefore, it is important to merge concordant
366 predictions and represent them as a single event for downstream analyses. We developed a five-step
367 procedure that uses fluctuations in local read depth to resolve breakpoint conflicts among
368 overlapping predictions by identifying the most likely start and end coordinates of the underlying
369 event**.** A detailed explanation of this strategy is presented in the **Supplemental file (Supplemental**
370 **Figs. S1 & S14)**. As an alternate strategy, we resolved breakpoint conflicts by simply selecting the
371 smallest and largest coordinates among the overlapping predictions as the start and end coordinates

372 of the underlying CNV event. This strategy is also described in the **Supplemental file**
373 **(Supplemental Fig. S2)**. Applying both methods to the 41,791 CNVs detected in the SVIP samples
374 yielded 8,382 unique breakpoint-resolved CNV events, in addition to the 20,719 singleton calls
375 (**Supplemental Fig. S3**).
376
377 **Microarray validations**
378 SNP microarray data were generated by the SVIP consortium (Duyzend et al. 2016) using Illumina
379 OmniExpress-12 microarrays for 272 samples and OmniExpress-14 for 19 samples. PennCNV
380 (v.1.0.3) was used to identify CNVs from microarray data for all 291 individuals using standard
381 parameters (Wang et al. 2007). Individual and family-based (trios and quads) CNV calls were
382 combined for autosomal chromosomes, while CNVs on chromosome X were called only at the
383 individual level. CNV calls with ≥1 bp overlap or gaps <20% of the total CNV length and <50 kbp
384 were merged. CNVs ≥50 kbp in length and containing ≥5 SNP target probes were subsequently
385 considered for further analyses.
386
387 **Feature selection for CN-Learn classifier**
388 We identified twelve features that represent the extent of support provided by the four individual
389 callers and genomic context as predictors of true CNVs to the CN-Learn Random Forest classifier.
390 Since we used four callers in our study, the extent of overlap calculated during the breakpoint-
391 resolution process (**Supplemental file** & **Supplemental Fig. S1**) served as individual predictors.
392 Concordance count and read-depth ratio (RD_{ratio}) for both breakpoint-resolved concordant calls and
393 singletons were also supplied as features. As individual algorithms use different GC and repeat
394 content (mappability) thresholds to classify CNV predictions as outliers, CNVs with extreme GC
395 content or low mappability could be predicted by one caller but discarded as an outlier by the other
396 callers. To take this into account, we extracted GC content data using the "nuc" option in bedtools
397 (Quinlan and Hall 2010) and mappability scores (Derrien et al. 2012) using the
398 "bigWigAverageOverBed" option in kentUtils (Kent et al. 2010) for use as predictors. Similarly, the
399 efficacy of CNV detection can vary across chromosomes, size ranges, and CNV type
400 (duplication/deletion). To take these variations into account, we used chromosome number, CNV
401 size, CNV type and the number of exome capture probes as the final set of CN-Learn features.
402

Probability estimation and classification using CN-Learn

CN-Learn leverages the features extracted for the CNVs in the training samples to build a Random Forest (Breiman 2001) classifier with hundreds of decision trees and estimates the probability of each CNV in the test samples to be true (**Figure 2A**). Decision trees have been shown to perform well when the distribution of observations is unbalanced between the classes of interest (Cieslak and Chawla 2008). Given the high false positive rates of CNV callers, an uneven split between the number of true and false predictions is likely to occur in clinical samples. In the samples we analyzed, 10% of the CNVs overlapped with the truth sets derived from microarray calls at a 10% reciprocal overlap threshold (**Supplemental Table 2**). To address the imbalance between the number of true and false CNVs, we stratified the training data by sample to accurately reflect the clinical setting as follows: CNVs in p% of the samples were used for training, and the remaining (1-p)% were used as the testing set. For a random forest built with "N" trees, "M" predictors and "C" classes, the probability of an observation "o" belonging to the class "c" ("true" or "false") can be expressed as $p_{o,c} = \Pr(Y = c \mid X = x_i)$, where x_i is a vector that captures the values for each of the 12 predictors and Y is the outcome variable. The probability $p_{o,true}$ of the CNV "o" being "true" in the test set was then measured as the proportion of trees in the forest that assigned it to the "true" class. Specifically, the probability of a CNV prediction can be represented as:

$$P_{o,true} = \Pr(Y = \text{True} \mid X = x_i) = \frac{\text{Number of trees that identify the CNV as True}}{\text{Number of trees in the forest}}$$

where X represents the values of the following features for the observation "o":

x_1 = Overlap proportion with CANOES

x_2 = Overlap proportion with CODEX

x_3 = Overlap proportion with CLAMMS

x_4 = Overlap proportion with XHMM

x_5 = Concordance among callers

x_6 = Read depth ratio

x_7 = Chromosome number

x_8 = CNV type (Duplication/Deletion)

x_9 = CNV size

434 x_{10} = Target probe count

435 x_{11} = GC Ccontent

436 x_{12} = Mappability

437

438 CNV calls with predicted probability score >0.5 were then classified as "true". We selected this
439 cutoff based on the distribution of validated ("true") and invalidated ("false") CNVs across the
440 predicted probability scores (ranging between 0 and 1). For both duplications and deletions, the
441 proportion of true calls compared to false calls was higher for CNVs with probability scores >0.5
442 (**Figure 2C**). This indicated that at a 0.5 threshold, the classifier recovers as many true CNVs
443 (recall) as possible without compromising on the false positive rate (precision).

444

445 **Statistical analysis**

446 All statistical analyses, including the calculation of precision-recall rates, feature importance, and
447 ROC areas, were performed using the Python library scikit-learn (v 0.18.1) (Pedregosa et al. 2012).
448 Plots were generated using the R package ggplot2 and the Python library matplotlib.

449

450 **Code availability**

451 CN-Learn is available as an open-source software at *https://github.com/girirajanlab/CN_Learn*.
452 Sample input files to test the CN-Learn pipeline are provided with the software package. Both the
453 exome and microarray-based CNV caller pipelines and the parameters used to obtain the original
454 CNV calls are also available at
455 *https://github.com/girirajanlab/CN_Learn/tree/master/scripts/callers/*.

456

DECLARATIONS

Ethics approval and consent to participate

As these data were de-identified, all of our samples were exempt from IRB review and conformed to the Helsinki Declaration. No other approvals were needed for the study.

Consent for publication

All authors agree and consent for publication of the manuscript.

Availability of data and materials

BAM files for all samples, aligned to the hg19 reference genome, were obtained from the Simons Foundation Autism Research Initiative (SFARI) via SFARI Base following appropriate approvals (*https://www.sfari.org/resource/sfari-base/*). The scripts used to analyze data and generate the plots are available in the "research" directory of CN-Learn repository of CN-Learn (*https://github.com/girirajanlab/CN_Learn/tree/master/scripts/research/)*. Any intermediate datasets generated during this study are available from the corresponding author upon request.

Competing interests

The authors declare that no competing interests exist in relation to this work.

Funding

This work was supported by R01-MH107431, R01-GM121907, SFARI Pilot Grant 399894, and resources from the Huck Institutes of the Life Sciences (to SG). This work was funded partly by the Big Data to Knowledge (BD2K) pre-doctoral training program (T32LM012415) from the National Institutes of Health (to VK). The funding bodies had no role in data collection, analysis, and interpretation.

Authors' contributions

VK and SG conceived the project. VK performed the analyses, generated the plots/images, and wrote and revised the manuscript; MJ organized the images for publication and assisted with revision to the manuscript; GJ and NK collected data and performed parts of the analyses; SG

supervised the research and wrote and revised the manuscript. All authors read and approved the final draft of the manuscript.

Acknowledgements

We thank Naomi Altman (Penn State), Dajiang Liu (Penn State), Arjun Krishnan (Michigan State), Cooduvalli Shashikant (Penn State), Lucilla Pizzo (Penn State), and Shaun Mahony (Penn State) for constructive comments on the manuscript.

495 **REFERENCES**

496

497 Backenroth D, Homsy J, Murillo LR, Glessner J, Lin E, Brueckner M, Lifton R, Goldmuntz E,
498 Chung WK, Shen Y. 2014. CANOES: Detecting rare copy number variants from whole exome
499 sequencing data. *Nucleic Acids Res* **42**: 1–9.

500 Bademci G, Mahdieh N, Bonyadi M, Duman D, Cengiz FB, Menendez I, Horta OD, Shirkavand A,
501 Zeinali S, Subasioglu A, et al. 2016. Comprehensive Analysis via Exome Sequencing
502 Uncovers Genetic Etiology in Autosomal Recessive Non-Syndromic Deafness in a Large
503 Multiethnic Cohort. *Genet Med* **18**: 364–371.

504 Bamshad MJ, Ng SB, Bigham AW, Tabor HK, Emond MJ, Nickerson DA, Shendure J. 2011.
505 Exome sequencing as a tool for Mendelian disease gene discovery. *Nat Rev Genet* **12**: 745–
506 755.

507 Breiman L. 2001. Random forests. *Mach Learn* **45**: 5–32.

508 Cieslak DA, Chawla N V. 2008. Learning Decision Trees for Unbalanced Data. *Mach Learn Knowl*
509 *Discov Databases* **5211**: 241–256.

510 Coughlin CR 2nd, Scharer GH, Shaikh TH. 2012. Clinical impact of copy number variation analysis
511 using high-resolution microarray technologies: advantages, limitations and concerns. *Genome*
512 *Med* **4**: 80.

513 de Ligt J, Boone PM, Pfundt R, Vissers LE, Richmond T, Geoghegan J, O'Moore K, de Leeuw N,
514 Shaw C, Brunner HG, et al. 2013. Detection of clinically relevant copy number variants with
515 whole-exome sequencing. *Hum Mutat* **34**: 1439–1448.

516 Derrien T, Estellé J, Sola SM, Knowles DG, Raineri E, Guigó R, Ribeca P. 2012. Fast computation
517 and applications of genome mappability. *PLoS One* **7**: e30337.

518 Duyzend MH, Nuttle X, Coe BP, Baker C, Nickerson DA, Bernier R, Eichler EE. 2016. Maternal
519 Modifiers and Parent-of-Origin Bias of the Autism-Associated 16p11.2 CNV. *Am J Hum Genet*
520 **98**: 45–57.

521 Epilepsy Phenome/Genome Project & Epi4K Consortium. 2015. Copy number variant analysis from
522 exome data in 349 patients with epileptic encephalopathy. *Ann Neurol* **78**: 323–328.

523 Fromer M, Moran JL, Chambert K, Banks E, Bergen SE, Ruderfer DM, Handsaker RE, McCarroll
524 SA, O'Donovan MC, Owen MJ, et al. 2012. Discovery and statistical genotyping of copy-
525 number variation from whole-exome sequencing depth. *Am J Hum Genet* **91**: 597–607.

526 Girirajan S, Campbell CD, Eichler EE. 2011. Human Copy Number Variation and Complex Genetic
527 Disease. *Annu Rev Genet* **45**: 203–226.
528 Hong CS, Singh LN, Mullikin JC, Biesecker LG. 2016. Assessing the reproducibility of exome copy
529 number variations predictions. *Genome Med* **8**: 1–11.
530 Jiang Y, Oldridge DA, Diskin SJ, Zhang NR. 2015. CODEX: A normalization and copy number
531 variation detection method for whole exome sequencing. *Nucleic Acids Res* **43**: e39.
532 John E, Adrian F, Jeremy G, Khai L, John L, Geoff O, Paul P, Rank D, Baybayan P, Bettman B, et
533 al. 2009. Real-Time DNA Sequencing from Single Polymerase Molecules. *Science* **323**: 133–
534 138.
535 Kataoka M, Matoba N, Sawada T, Kazuno AA, Ishiwata M, Fujii K, Matsuo K, Takata A, Kato T.
536 2016. Exome sequencing for bipolar disorder points to roles of de novo loss-of-function and
537 protein-altering mutations. *Mol Psychiatry* **21**: 885–893.
538 Kent WJ, Zweig AS, Barber G, Hinrichs AS, Karolchik D. 2010. BigWig and BigBed: Enabling
539 browsing of large distributed datasets. *Bioinformatics* **26**: 2204–2207.
540 Koboldt DC, Steinberg KM, Larson DE, Wilson RK, Mardis ER. 2013. The next-generation
541 sequencing revolution and its impact on genomics. *Cell* **155**: 27–38.
542 Koboldt DC, Zhang Q, Larson DE, Shen D, McLellan MD, Lin L, Miller CA, Mardis ER, Ding L,
543 Wilson RK. 2012. VarScan 2: Somatic mutation and copy number alteration discovery in
544 cancer by exome sequencing. *Genome Res* **22**: 568–576.
545 Krumm N, O'Roak BJ, Karakoc E, Mohajeri K, Nelson B, Vives L, Jacquemont S, Munson J,
546 Bernier R, Eichler EE. 2013. Transmission disequilibrium of small CNVs in simplex autism.
547 *Am J Hum Genet* **93**: 595–606.
548 Krumm N, Sudmant P, Ko A. 2012. Copy number variation detection and genotyping from exome
549 sequence data. *Genome Res* **22**: 1525–1532.
550 Krumm N, Turner TN, Baker C, Vives L, Mohajeri K, Witherspoon K, Raja A, Coe BP, Stessman
551 HA, He ZX, et al. 2015. Excess of rare, inherited truncating mutations in autism. *Nat Genet* **47**:
552 582–588.
553 Lam ET, Hastie A, Lin C, Ehrlich D, Das SK, Austin MD, Deshpande P, Cao H, Nagarajan N, Xiao
554 M, et al. 2012. Genome mapping on nanochannel arrays for structural variation analysis and
555 sequence assembly. *Nat Biotechnol* **30**: 771–776.
556 Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R.

557 2009. The Sequence Alignment/Map format and SAMtools. *Bioinformatics* **25**: 2078–2079.
558 Mak ACY, Lai YYY, Lam ET, Kwok TP, Leung AKY, Poon A, Mostovoy Y, Hastie AR, Stedman
559 W, Anantharaman T, et al. 2016. Genome-wide structural variation detection by genome
560 mapping on nanochannel arrays. *Genetics* **202**: 351–362.
561 McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, Garimella K, Altshuler
562 D, Gabriel S, Daly M, et al. 2010. The Genome Analysis Toolkit: A MapReduce framework
563 for analyzing next-generation DNA sequencing data. *Genome Res* **20**: 254–260.
564 Miller DT, Adam MP, Aradhya S, Biesecker LG, Brothman AR, Carter NP, Church DM, Crolla JA,
565 Eichler EE, Epstein CJ, et al. 2010. Consensus Statement: Chromosomal Microarray Is a First-
566 Tier Clinical Diagnostic Test for Individuals with Developmental Disabilities or Congenital
567 Anomalies. *Am J Hum Genet* **86**: 749–764.
568 Packer JS, Maxwell EK, O'Dushlaine C, Lopez AE, Dewey FE, Chernomorsky R, Baras A, Overton
569 JD, Habegger L, Reid JG. 2015. CLAMMS: A scalable algorithm for calling common and rare
570 copy number variants from exome sequencing data. *Bioinformatics* **32**: 133–135.
571 Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P,
572 Weiss R, Dubourg V, et al. 2012. Scikit-learn: Machine Learning in Python. *J Mach Learn Res*
573 **12**: 2825–2830.
574 Perry G, Yang F, Marques-Bonet T, Murphy C. 2008. Copy number variation and evolution in
575 humans and chimpanzees. *Genome Res* **18**: 1698–1710.
576 Poultney CS, Goldberg AP, Drapeau E, Kou Y, Harony-Nicolas H, Kajiwara Y, De Rubeis S,
577 Durand S, Stevens C, Rehnström K, et al. 2013. Identification of small exonic CNV from
578 whole-exome sequence data and application to autism spectrum disorder. *Am J Hum Genet* **93**:
579 607–619.
580 Priest JR, Osoegawa K, Mohammed N, Nanda V, Kundu R, Schultz K, Lammer EJ, Girirajan S,
581 Scheetz T, Waggott D, et al. 2016. De Novo and Rare Variants at Multiple Loci Support the
582 Oligogenic Origins of Atrioventricular Septal Heart Defects. *PLoS Genet* **12**: 1–25.
583 Quinlan AR, Hall IM. 2010. BEDTools: A flexible suite of utilities for comparing genomic features.
584 *Bioinformatics* **26**: 841–842.
585 Redon R, Ishikawa S, Fitch KR, Feuk L, Perry GH, Andrews TD, Fiegler H, Shapero MH, Carson
586 AR, Chen W, et al. 2006. Global variation in copy number in the human genome. *Nature* **444**:
587 444–454.

588 Retterer K, Juusola J, Cho MT, Vitazka P, Millan F, Gibellini F, Vertino-Bell A, Smaoui N, Neidich
589 J, Monaghan KG, et al. 2016. Clinical application of whole-exome sequencing across clinical
590 indications. *Genet Med* **18**: 696–704.
591 Sathirapongsasuti JF, Lee H, Horst BAJ, Brunner G, Cochran AJ, Binder S, Quackenbush J, Nelson
592 SF. 2011. Exome sequencing-based copy-number variation and loss of heterozygosity
593 detection: ExomeCNV. *Bioinformatics* **27**: 2648–2654.
594 Sebat J, Lakshmi B, Troge J, Alexander J, Young J, Lundin P, Månér S, Massa H, Walker M, Chi
595 M, et al. 2004. Large-scale copy number polymorphism in the human genome. *Science* **305**:
596 525–8.
597 Spiro JE, Chung WK. 2012. Simons Variation in Individuals Project (Simons VIP): A Genetics-
598 First Approach to Studying Autism Spectrum and Related Neurodevelopmental Disorders.
599 *Neuron* **73**: 1063–1067.
600 Stark Z, Tan TY, Chong B, Brett GR, Yap P, Walsh M, Yeung A, Peters H, Mordaunt D, Cowie S,
601 et al. 2016. A prospective evaluation of whole-exome sequencing as a first-tier molecular test
602 in infants with suspected monogenic disorders. *Genet Med* **18**: 1090–1096.
603 Tan R, Wang Y, Kleinstein SE, Liu Y, Zhu X, Guo H, Jiang Q, Allen AS, Zhu M. 2014. An
604 Evaluation of Copy Number Variation Detection Tools from Whole-Exome Sequencing Data.
605 *Hum Mutat* **35**: 899–907.
606 Tan TY, Dillon OJ, Stark Z, Schofield D, Alam K, Shrestha R, Chong B, Phelan D, Brett GR, Creed
607 E, et al. 2017. Diagnostic impact and cost-effectiveness of whole-exome sequencing for
608 ambulant children with suspected monogenic conditions. *JAMA Pediatr* **171**: 855–862.
609 Voskoboynik A, Neff NF, Sahoo D, Newman AM, Pushkarev D, Koh W, Passarelli B, Fan HC,
610 Mantalas GL, Palmeri KJ, et al. 2013. The genome sequence of the colonial chordate, Botryllus
611 schlosseri. *Elife* **2013**: 1–24.
612 Wang K, Li M, Hadley D, Liu R, Glessner J, Grant SFA, Hakonarson H, Bucan M. 2007.
613 PennCNV : An integrated hidden Markov model designed for high-resolution copy number
614 variation detection in whole-genome SNP genotyping data. *Genome Res* **17**: 1665–1674.
615 Yang Y, Muzny DM, Reid JG, Bainbridge MN, Willis A, Ward PA, Braxton A, Beuten J, Xia F,
616 Niu Z, et al. 2013. Clinical Whole-Exome Sequencing for the Diagnosis of Mendelian
617 Disorders. *N Engl J Med* **369**: 1502–1511.
618 Yao R, Zhang C, Yu T, Li N, Hu X, Wang X, Wang J, Shen Y. 2017. Evaluation of three read-depth

619 based CNV detection tools using whole-exome sequencing data. *Mol Cytogenet* **10**: 1–7.
620 Zheng GXY, Lau BT, Schnall-Levin M, Jarosz M, Bell JM, Hindson CM, Kyriazopoulou-
621 Panagiotopoulou S, Masquelier DA, Merrill L, Terry JM, et al. 2016. Haplotyping germline
622 and cancer genomes with high-throughput linked-read sequencing. *Nat Biotechnol* **34**: 303–
623 311.
624