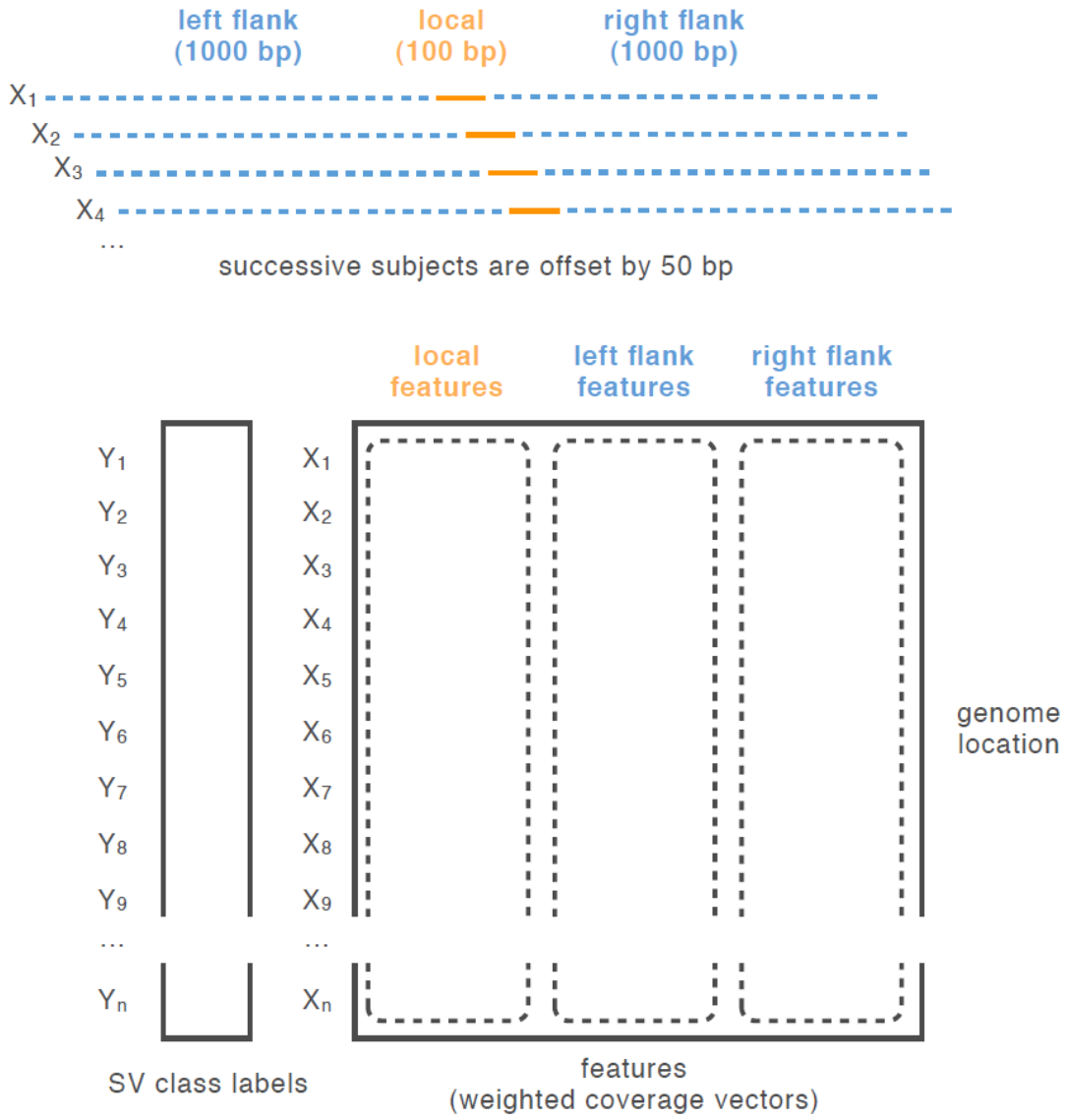
# forestSV: structural variant discovery through statistical learning
## (supplementary information)

*Jacob J Michaelson and Jonathan Sebat*

## Contents

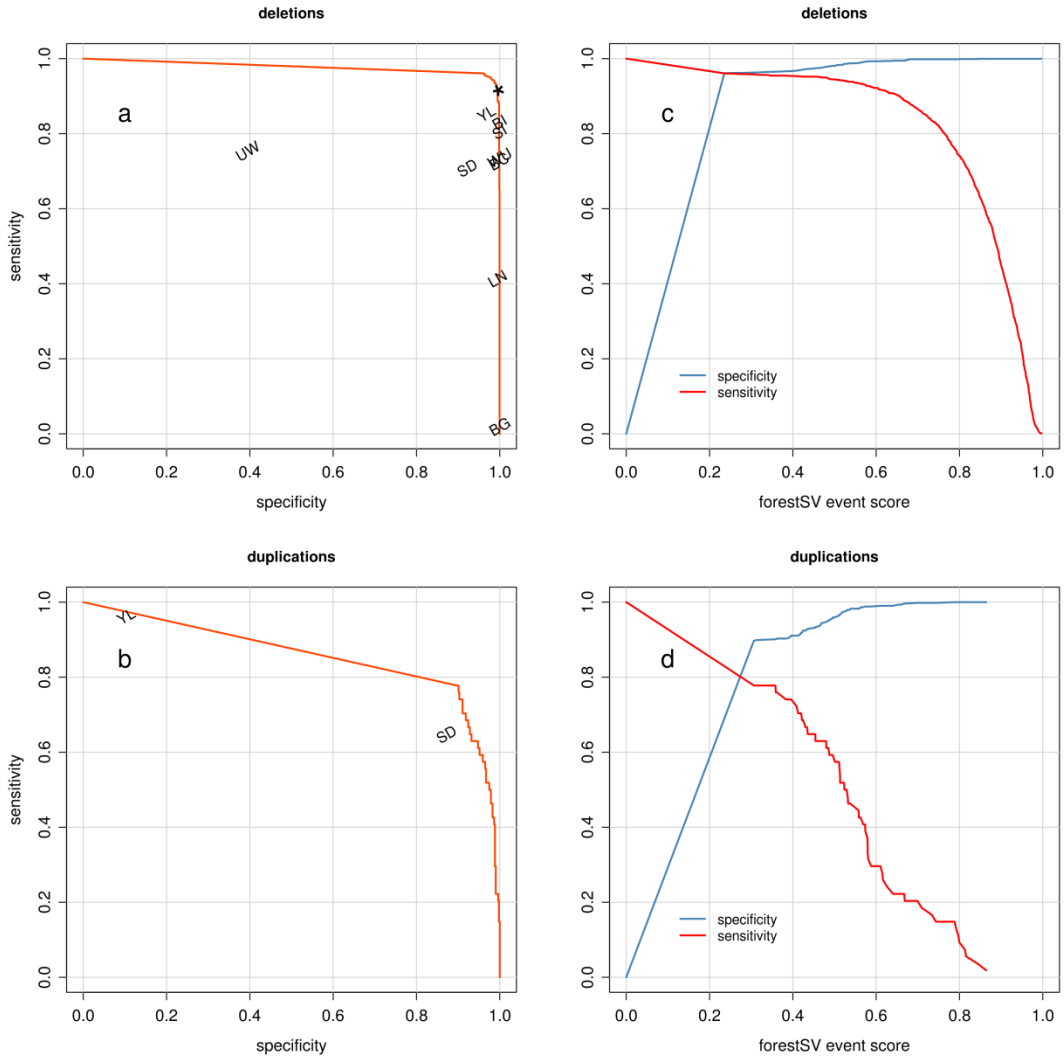
# Supplementary Figures

**Supplementary Figure 1: Structure of training data.**



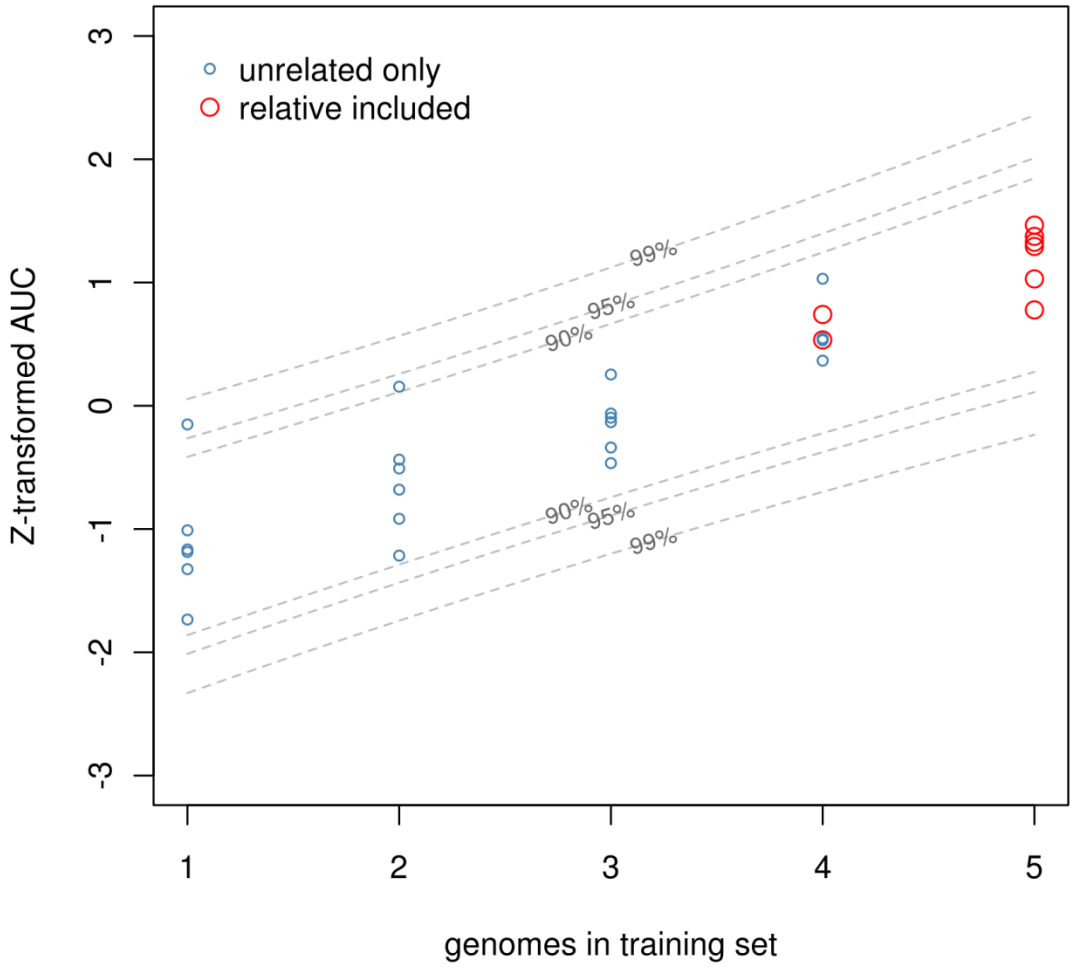

**Supplementary Figure 1: Structure of training data.** The columns of the feature matrix *X* are weighted coverage vectors, constructed using information available in the BAM file. These features convey information on such metrics as read depth, paired end signals, mapping quality, base content, etc., and are evaluated on several scales and relative locations. The local scale is a 100 bp window, while the left and right flanks are 1000 bp to the left and right (respectively) of this window. These "subjects" correspond to the rows of the feature matrix *X*, as well as the elements in the class label vector *Y*, and are offset from each other by 50 bp.

**Supplementary Figure 2: Performance relative to existing structural variant discovery methods.**



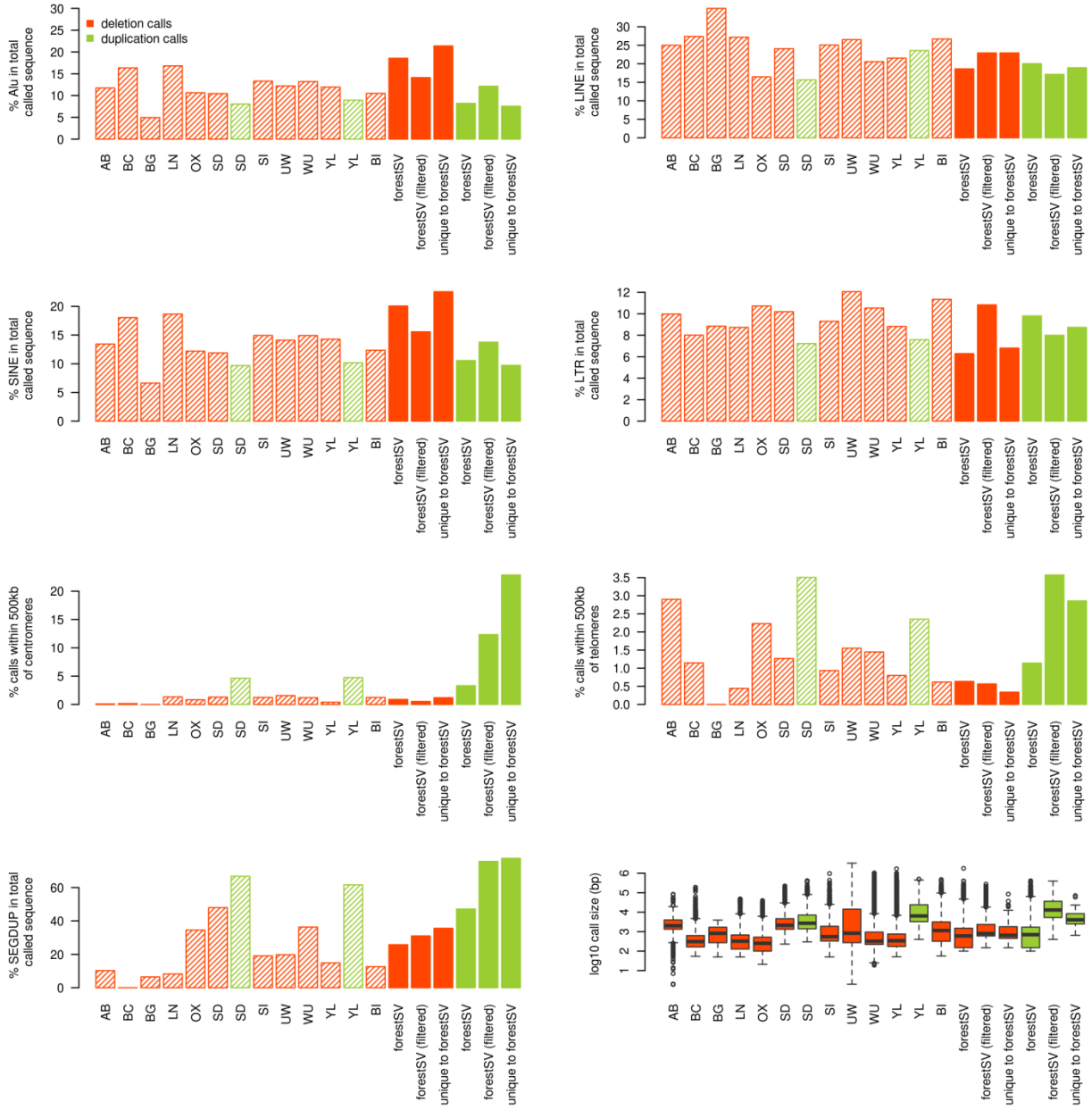

**Supplementary Figure 2: Performance relative to existing structural variant discovery methods.** We obtained structural variant predictions via a leave-one-out cross validation scheme, where each individual, in turn, was left out of the training stage. Predictions were then made for this individual, and the resulting call sets were compared against a set of gold standard positives and known false positive calls. Calls were required to have at least 50% reciprocal overlap with the gold standard set for consideration in this assessment. Comparisons were made with the call sets released in ref. 1 (the designations used there are carried over here). For deletion calls (a), forestSV (orange curve) had better sensitivity than any single donor method, while providing specificity comparable to the most specific methods. forestSV also provided a combination of sensitivity and specificity that matched that of the merged and genotyped call set (*), with even higher sensitivity at the cost of some specificity. In terms of duplications, which few methods attempt because of their difficult nature, forestSV performed better than both of the donor methods (b). Sensitivity and specificity are shown with their relationship to the forestSV event score for deletions (c) and duplications (d).

**Supplementary Figure 3: Effect on predictive performance of adding related individuals to the training set.**



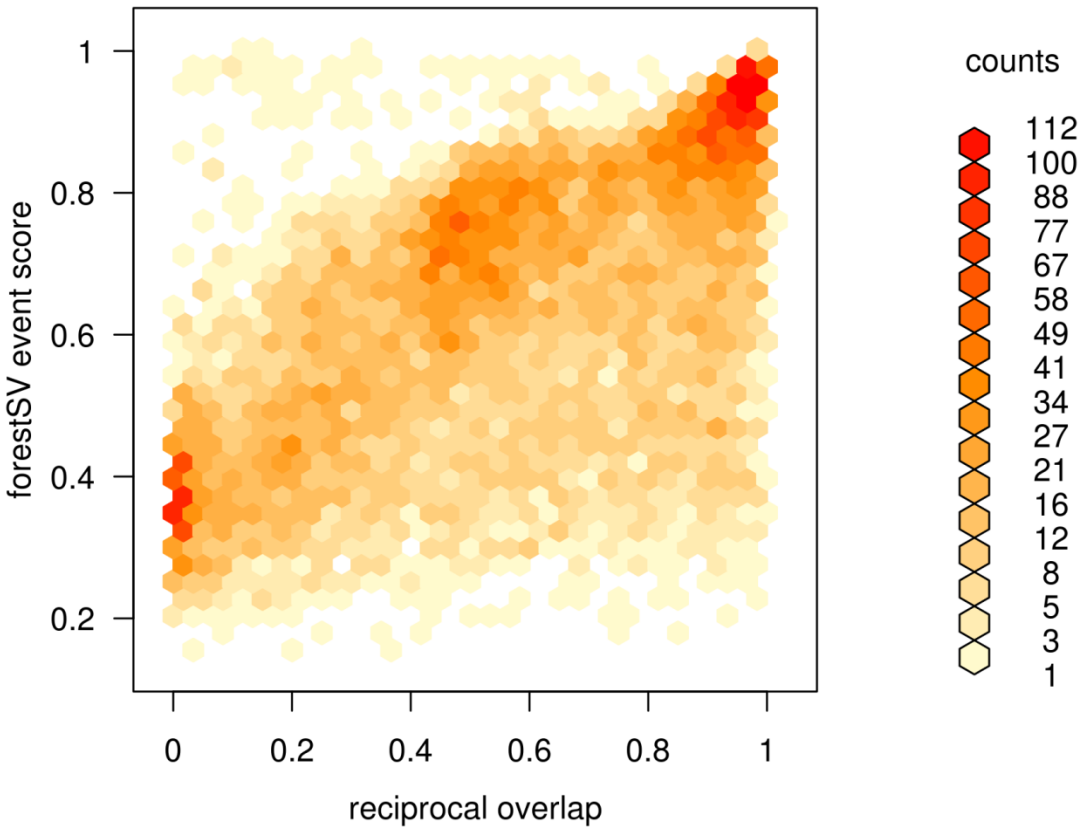

**Supplementary Figure 3: Effect on predictive performance of adding related individuals to the training set.** When including additional unrelated individuals in the training set, performance (here measured in area under the ROC curve, AUC) increases in a linear fashion. Training sets that include related individuals (red points) fall well within the 90% prediction interval of this "unrelated" model, suggesting that their contribution to increasing performance is no different than it would be when adding unrelated individuals.

**Supplementary Figure 4: Size distributions and characterization of the affinity of calls for various genomic features.**



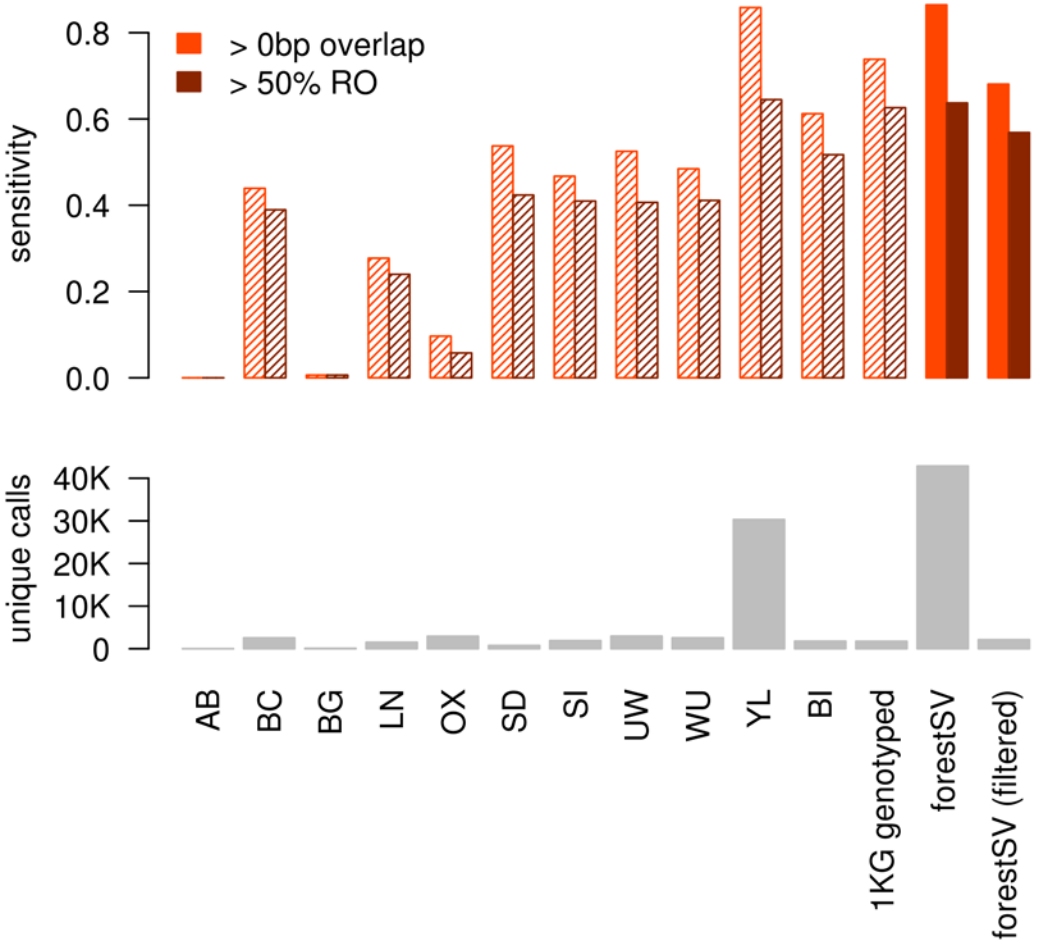

**Supplementary Figure 4: Size distributions and characterization of affinity of calls for various genomic features.** We compared forestSV deletion and duplication calls to call sets comprising the work in ref. 1 based on their retrotransposon content, segmental duplication content, proximity to centromeres and telomeres, and size. Calls made by forestSV were evaluated separately in three categories: all forestSV calls, forestSV calls filtered at a score of 0.65, and filtered forestSV calls that are unique to forestSV, i.e. calls that did not intersect any call from the call sets comprising ref. 1. Alu elements, and SINEs in general, are more predominant among deletion calls unique to forestSV (21% and 23% respectively, of called sequence). forestSV fares well in regions of segmental duplication, compared to some other methods (31% of sequence in called deletions, and 77% of sequence in called duplications). Duplications called by forestSV are more likely to fall within 500kb of a centromere (23% of unique calls) than other methods, and 3% of unique duplication calls fall within 500kb of a telomere.

**Supplementary Figure 5: Relationship between reciprocal overlap and forestSV event score.**



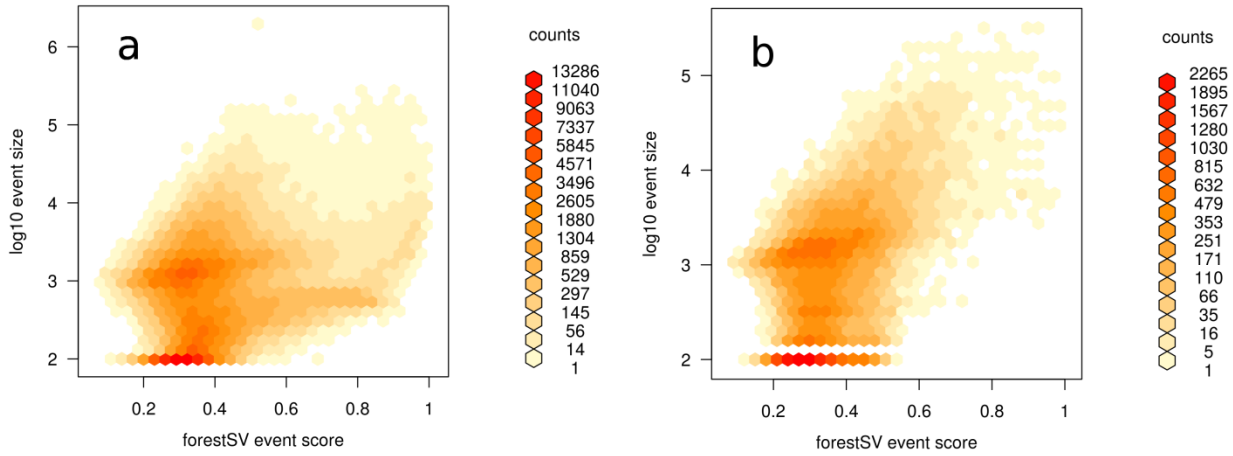

**Supplementary Figure 5: Relationship between reciprocal overlap and forestSV event score.** We took merged/genotyped deletion calls from 1KG trios (ref. 1) and intersected them with forestSV calls from the corresponding individuals. We found that on average, 89.5% of 1KG calls intersected a forestSV call by 1bp or more. Here we show that increasing forestSV event scores generally coincide with increased reciprocal overlap with 1KG deletion calls.

**Supplementary Figure 6: Sensitivity of structural variant discovery methods when evaluated in the NA12878 gold standard call set.**



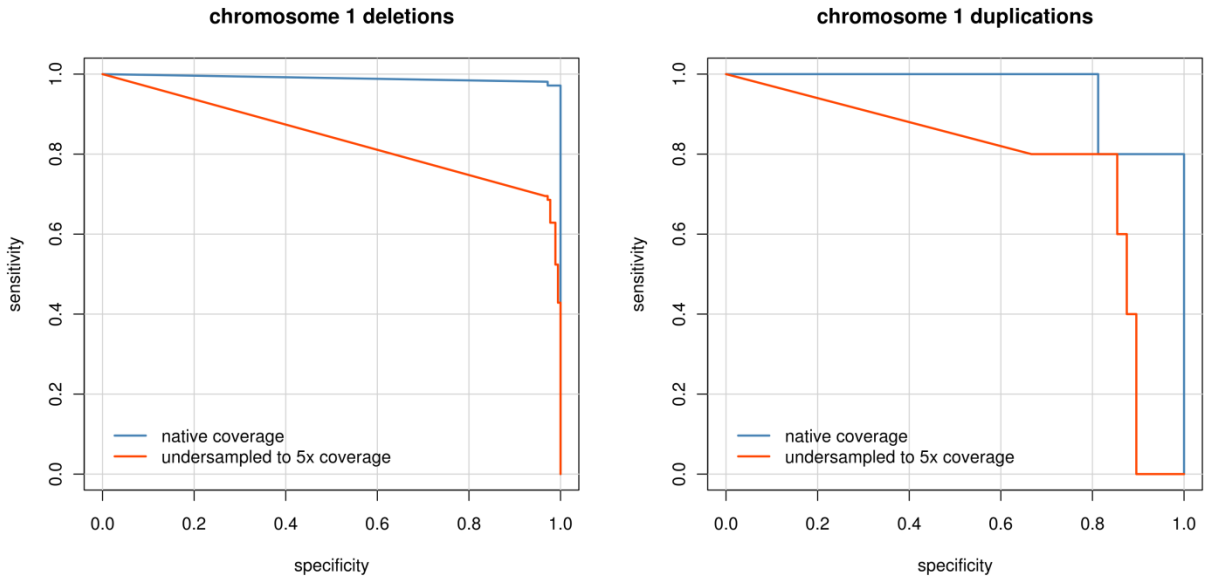

**Supplementary Figure 6: Sensitivity of structural variant discovery methods when evaluated on the NA12878 gold standard call set.** Here we show the sensitivity of forestSV vs. each of the individual call sets, as well as that of the merged/genotyped set from ref. 1. We also show the number of unique calls from each call set to give some surrogate for specificity when comparing call sets. At 68% and 57% sensitivity for > 0bp and > 50% reciprocal overlap, respectively, it was exceeded only by the YL call set (which made over 30,000 unique calls to achieve 86% and 64% sensitivity at > 0bp and > 50% overlap) and the merged/genotyped call set (74% and 63% at > 0bp and > 50% overlap).

**Supplementary Figure 7: Relationship between forestSV event scores and event size.**



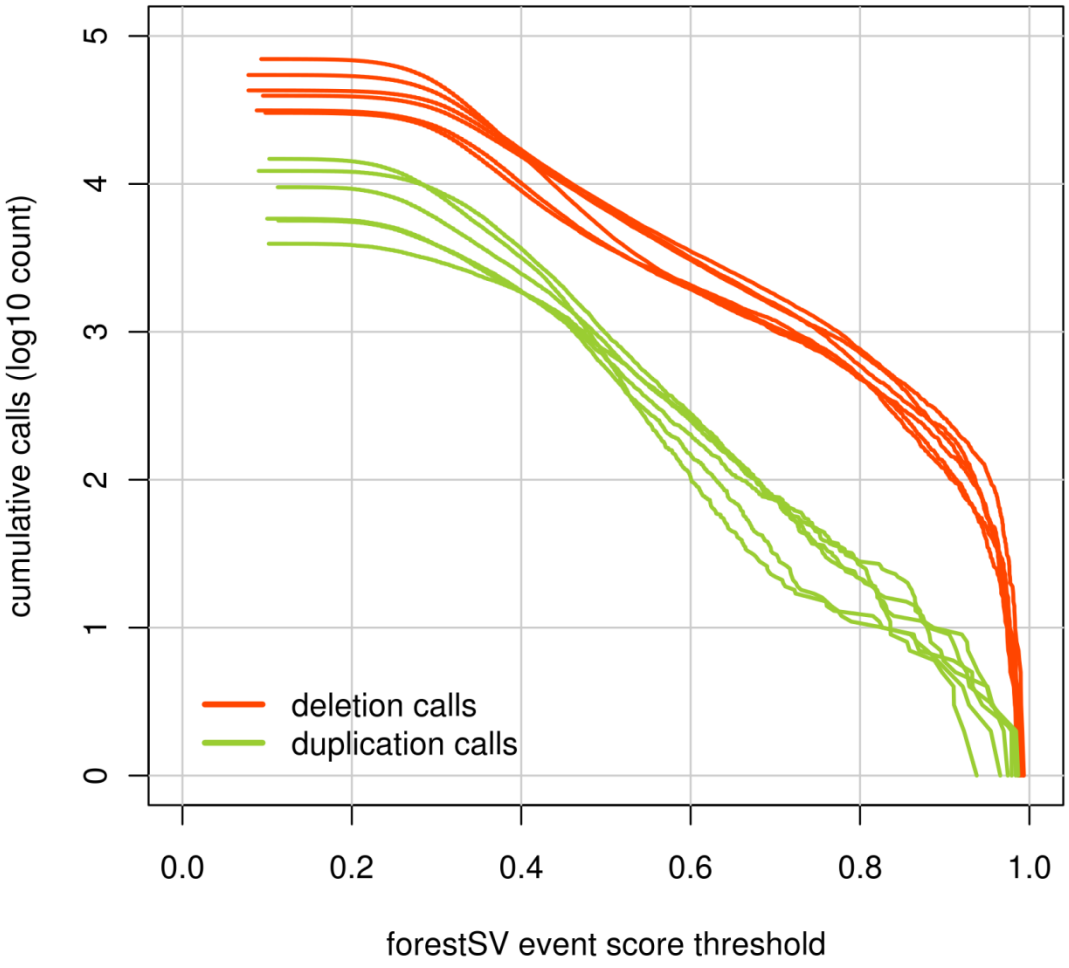

**Supplementary Figure 7: Relationship between forestSV event scores and event size.** We investigated the test set calls from the 1KG data to determine whether an event size bias existed in the forestSV score. Deletions (a) seem to be relatively unbiased in their event scores for events larger than several hundred base pairs. Conversely, duplications (b) show a clear absence of small events in high-scoring calls. This may be a side-effect of the small number of duplications used for training.

**Supplementary Figure 8: Performance of forestSV on low coverage (5x) sequencing data.**



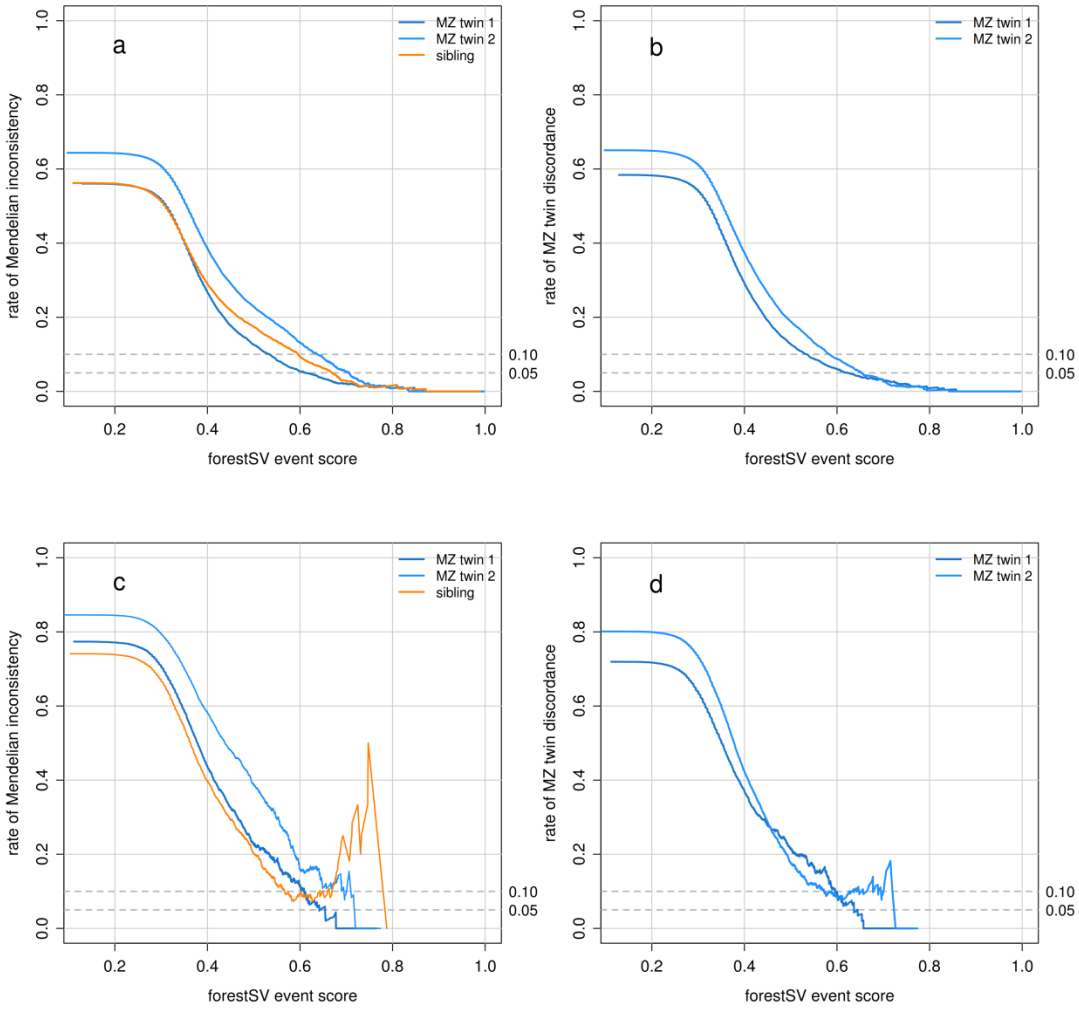

**Supplementary Figure 8: Performance of forestSV on low coverage (5x) sequencing data.** We investigated the performance of forestSV on low-coverage sequencing data by downsampling to 5x coverage the same (high coverage) BAM files we had previously used for training and cross-validation. We applied forestSV to these downsampled BAM files and constructed sensitivity-specificity curves based on > 50% reciprocal overlap with calls in the gold standard set (as was done in Supplementary Figure 2). When examining the AUC, performance fell by 15% (0.99 to 0.84) for deletions and 20% (0.96 to 0.77) for duplications. This suggests that forestSV has the potential to be of use to researchers with low-coverage data, but as anticipated, expectations of performance would need to be adjusted accordingly.

**Supplementary Figure 9: Accumulation of deletion and duplication calls in the 1KG data with decreasing event score threshold.**



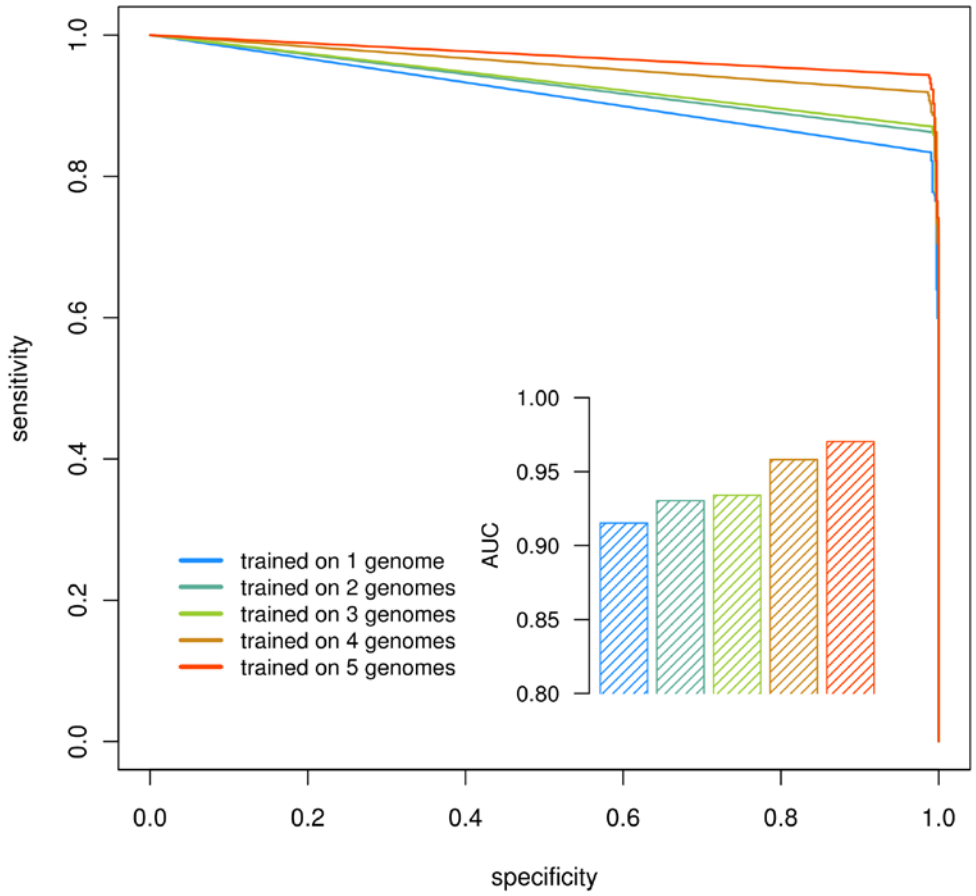

**Supplementary Figure 9: Accumulation of deletion and duplication calls in the 1KG data with decreasing event score threshold.** As the event score threshold is relaxed, deletion and duplication calls are accumulated in characteristically different ways, suggesting that when applying the same threshold to deletions and duplications, the sensitivity to duplications will be considerably less than for deletions. At a threshold of 0.65, 1386-2506 deletions are called per individual and 44-144 duplications are called.

**Supplementary Figure 10: Estimation of error rates based on family relationships.**



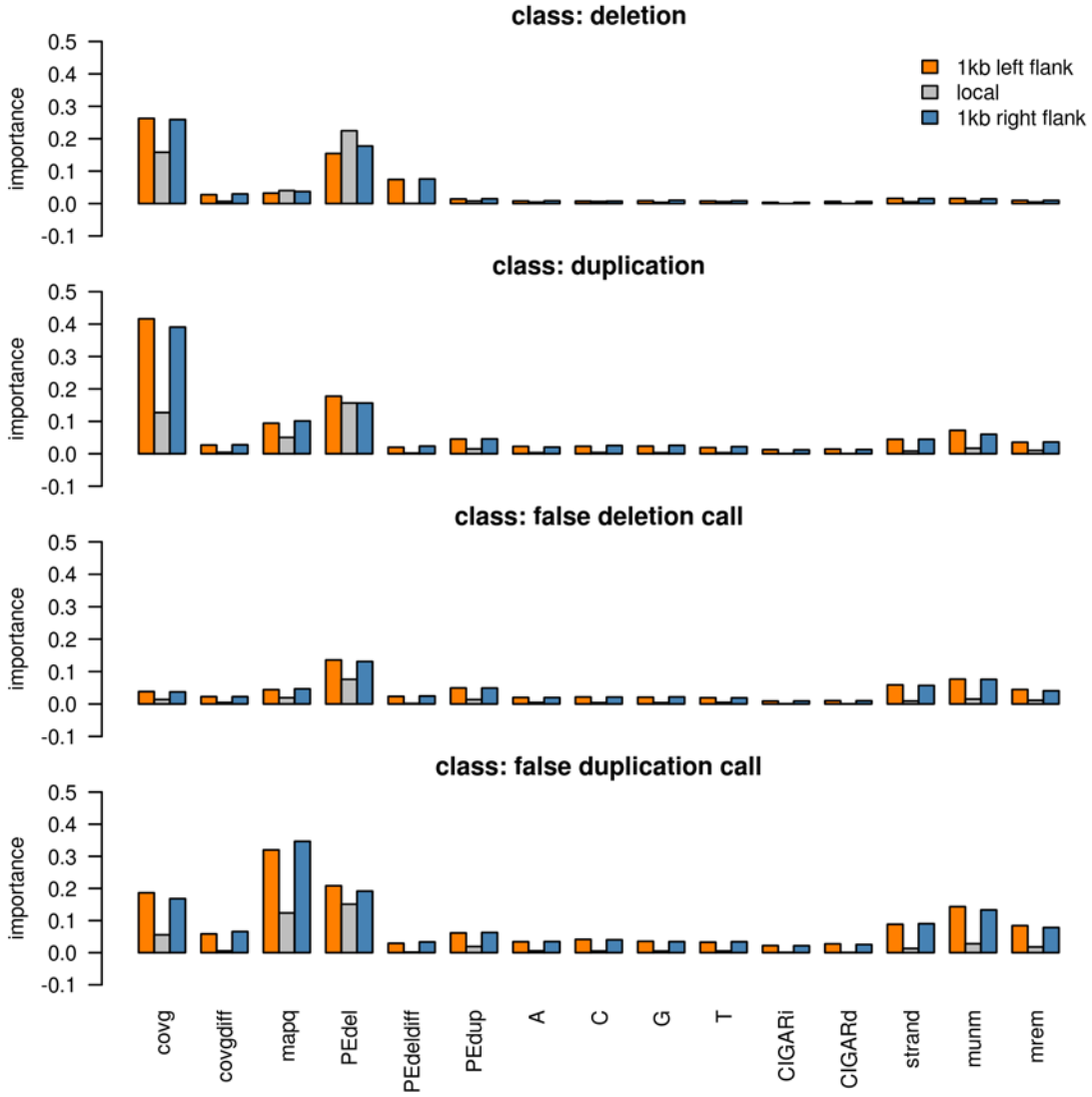

**Supplementary Figure 10: Estimation of error rates based on family relationships.** We validated the forestSV framework by applying it to unpublished whole genome sequencing data from a family in an autism cohort. After calling structural variants genome-wide, we examined how the rate of Mendelian inconsistency varied with the event score for deletions (a). Similarly, we examined how the rate of monozygotic (MZ) twin discordance varied with the event score (b). Analogous plots are shown for duplication events in (c) and (d); note the higher error rates for duplications vs. deletions at the same event score threshold. These two rates are an indication of the overall error rate, and serve to inform the user about a sensible threshold for predicted structural variants. Based on the observation of these error rates, calls from forestSV should be thresholded between 0.65-0.70 to maintain an error rate of 5% or less in deletions and ~10% in duplications. Using a threshold of 0.65 in these individuals led to call sets containing 2,300-2,593 unique deletion events, and 53-72 duplication events.

**Supplementary Figure 11: Classifier improvement with additional training data.**



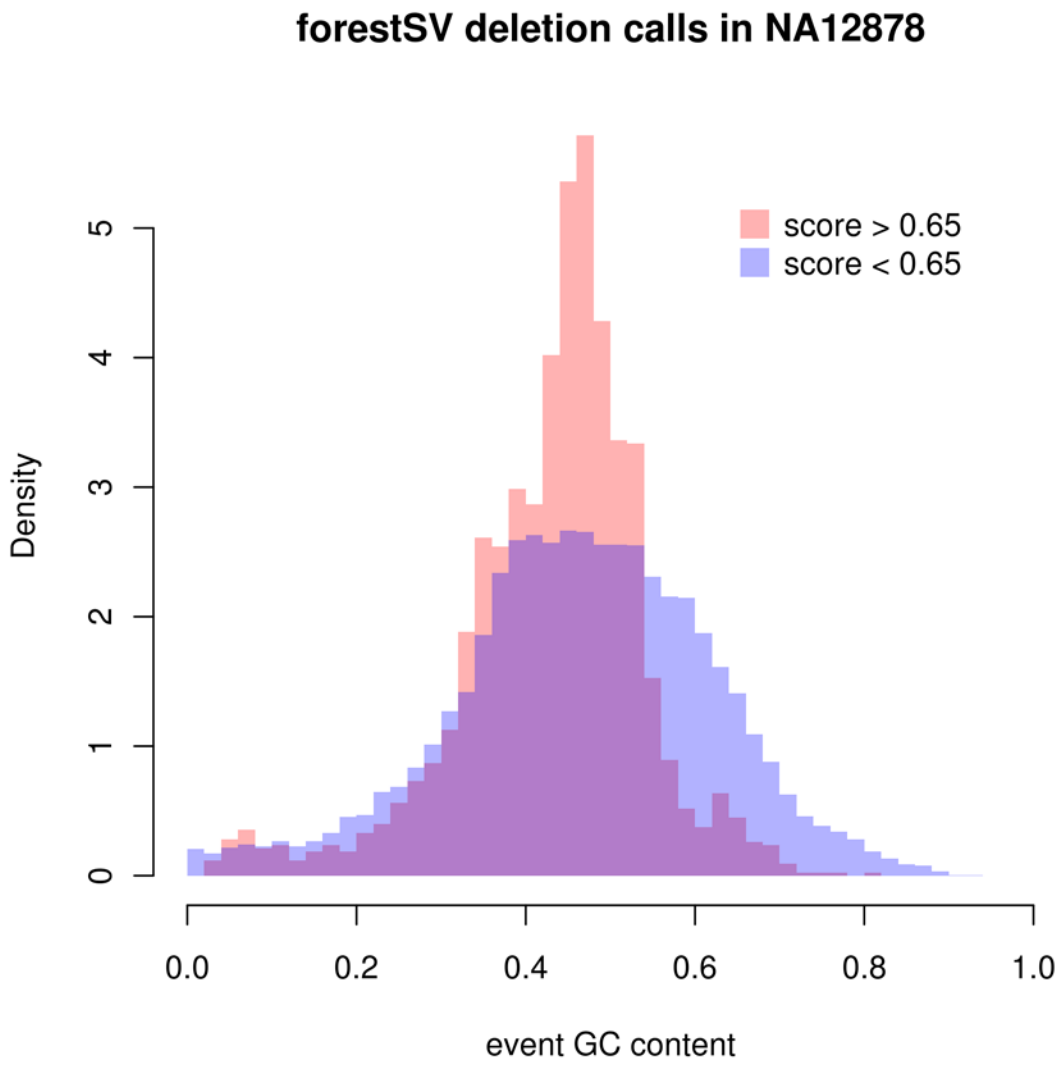

**Supplementary Figure 11: Classifier improvement with additional training data.** forestSV provides a framework that is flexible and is able to learn through exposure to additional and better data, resulting in improved structural variant discovery. This is demonstrated here, where we made successive rounds of deletion calls in NA12878 by using first 1, then 2, 3, 4, and 5 other individuals to train the Random Forest classifier used to make the calls. There is a steady improvement in both the sensitivity-specificity curves and the area under the ROC curve (AUC) when additional data is used in training. Performance was evaluated using the same gold standard set as in Supplementary Figure 2.

**Supplementary Figure 12: Random Forest permutation importance measures by class.**



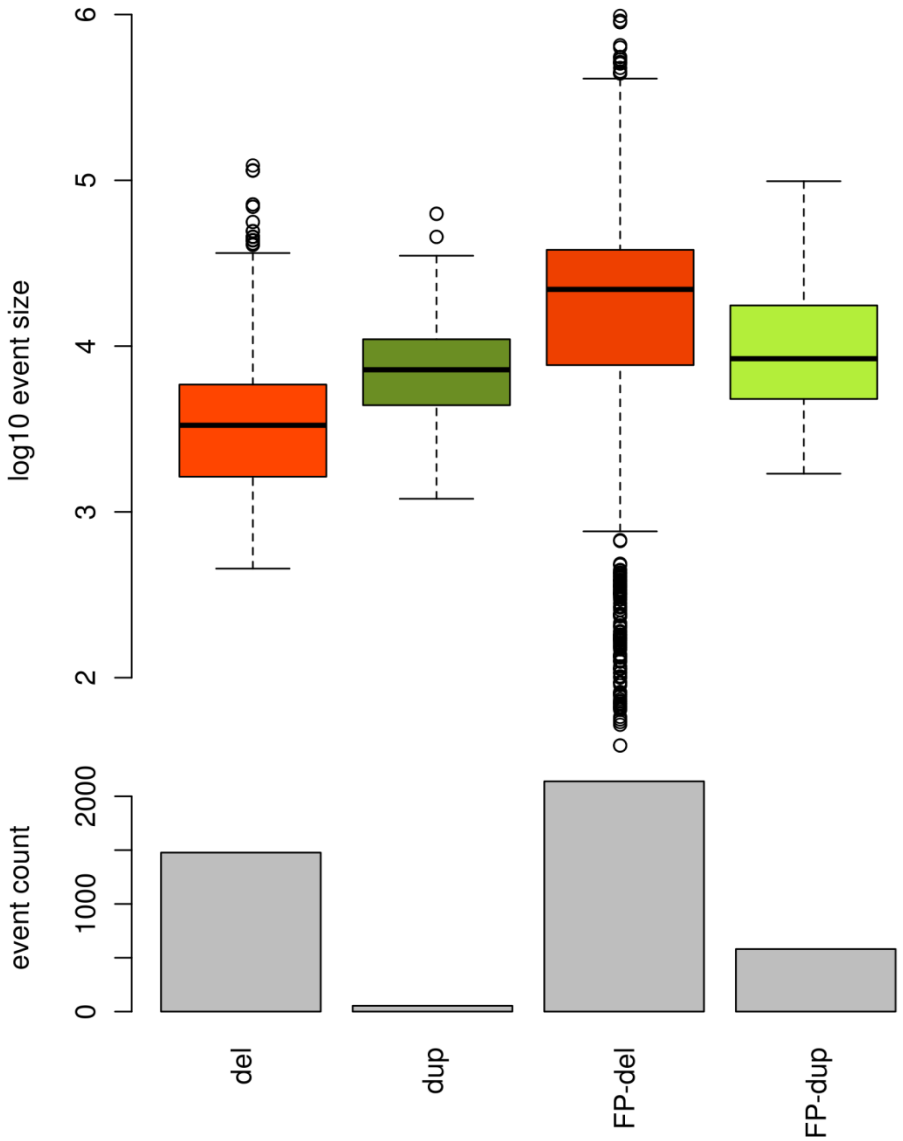

**Supplementary Figure 12: Random Forest permutation importance measures by class.** The importance measures indicate a feature's contribution to the ability of the classifier to correctly discriminate a class. Read depth (covg) and paired end signals (PEdel) are consistently the most important features for discrimination of structural variants, which is somewhat unsurprising given that they are the two most widely used features in SV discovery methods. However, we found that other features not previously used in structural variant discovery (mapq, strand, munm, and mrem) are also useful, especially in avoiding would-be false positive calls.

**Supplementary Figure 13: Distribution of GC content among forestSV deletion calls in NA12878.**




**Supplementary Figure 13: Distribution of GC content among forestSV deletion calls in NA12878.** Higher confidence calls have noticeably lower GC content, suggesting that the inclusion of base content as predictors in the classifier has the effect of an implicit GC correction. The distributions shown differ significantly ($P < 10^{-10}$ by the Kolmogorov-Smirnov test).

**Supplementary Figure 14: Size distribution and event counts for the gold standard set of structural variants.**




**Supplementary Figure 14: Size distribution and event counts for the gold standard set of structural variants.** Finding duplication calls that were of sufficient confidence to include in the gold standard set (and consequently the training set) was difficult, owing to the lack of prior data and diminished focus on duplications by other structural variant discovery methods. This obstacle is the most plausible explanation for forestSV's performance disparity between calling deletions and duplications.

# Supplementary Tables

**Supplementary Table 1: Description of class labels.**

| Class label | Description |
|---|---|
| deletion | subjects falling within an event called as a deletion by at least one of the methods from ref. 1, and having at least 90% reciprocal overall with a deletion call from either ref. 2 or ref. 3, with the corresponding breakpoints differing by less than 1 kb. These events were considered "gold standards". Additional (non-gold standard) deletion events were added, provided that they were called by at least two methods from ref. 1 and were experimentally validated at least once. |
| duplication | subjects falling within an event called as a duplication by at least one of the methods from ref. 1, and having at least 90% reciprocal overall with a duplication call from either ref. 2 or ref. 3, with the corresponding breakpoints differing by less than 1 kb. These events were considered "gold standards". Additional (non-gold standard) duplication events were added, provided that they were called by at least two methods from ref. 1 and were experimentally validated at least once. |
| deletion-flanking | subjects falling within 1 kb up- or downstream of events labeled as *deletion* (and not intersecting events of other class types). |
| duplication-flanking | subjects falling within 1 kb up- or downstream of events labeled as *duplication* (and not intersecting events of other class types). |
| false positive deletion | subjects falling within singleton *deletion* calls that were experimentally invalidated in ref. 1. |
| false positive duplication | subjects falling within singleton *duplication* calls that were experimentally invalidated in ref. 1. |
| invariant | subjects sampled from regions where no calls were made (in ref. 1) by any method in any of the high coverage samples. In addition, windows that intersected with gaps in the reference assembly were dropped. |

**Supplementary Table 1: Description of class labels.** Each subject in the training data was labeled as one of seven classes. Consequently, predictions on new genomic data represent the confidence of belonging to one of these classes.

**Supplementary Table 2: Description of the weighted coverage vectors.**

| Weighted coverage vector | Abbreviation | Description |
|---|---|---|
| read depth | covg | a log2 ratio of the mean within-window coverage to the chromosomal median coverage |
| read depth difference | covgdiff | the difference between the maximum and minimum log2 read depth within a window |
| coverage weighted by mapping quality | mapq | coverage weighted by mapping quality as found in the BAM file |
| coverage weighted by positive outlier read pairs | PEdel | The distribution of the insert sizes of read pairs, and insert sizes outside of the median ± 4 × MAD are considered outliers. If the read in question belongs to an outlier pair, a weight of -1 is given, and 0 otherwise. |
| difference in outlier read pair signal | PEdeldiff | the difference between the within-window maximum and minimum for the weighted coverage vector used in PEdel |
| coverage weighted by negative outlier read pairs | PEdup | analogous to PEdel, but outliers are have a smaller (rather than larger) than expected insert size |
| coverage weighted by A content | A | contributing reads are weighted by the percent A content |
| coverage weighted by C content | C | contributing reads are weighted by the percent C content |
| coverage weighted by G content | G | contributing reads are weighted by the percent G content |
| coverage weighted by T content | T | contributing reads are weighted by the percent T content |
| coverage weighted by aligner insertions | CIGARi | each read is weighted by the number of insertions the aligner performed on it (as provided by the CIGAR string) |
| coverage weighted by aligner deletions | CIGARd | each read is weighted by the number of deletions the aligner performed on it (as provided by the CIGAR string) |
| coverage weighted by strand concordance | strand | if the read and its mate are mapped to the same strand, a weight of 1 is given, 0 otherwise |
| coverage weighted by mate unmapped | munm | if a read's mate is unmapped, the read is given a weight of 1, 0 otherwise |
| coverage weighted by mate mapping remotely | mrem | if a read's mate maps to farther than 10 Mb away, or to a different chromosome, the read is given a weight of 1, 0 otherwise |

**Supplementary Table 2: Description of the weighted coverage vectors.** Fifteen weighted coverage vectors are constructed by based on information available within the BAM file. These weighted coverage vectors are summarized at three relative locations and scales (1kb left, 100 bp local, and 1kb right) by the featMat() function, producing a feature matrix with 45 columns.

# Supplementary Results

## Benchmarking on 1000 Genomes Project data

We compared test set deletion calls made by forestSV to deletion calls from the 1KG merged/genotyped call set from ref. 1 and found that an average of 90% of the 1KG calls intersected with forestSV calls by 1bp or more (NA12878: 91%, NA12891: 82%, NA12892: 89%, NA19240: 96%, NA19238: 86%, NA19239: 95%). An average of 51% of these calls had reciprocal overlap of > 50% with calls generated by forestSV (NA12878: 51%, NA12891: 50%, NA12892: 58%, NA19240: 51%, NA19238: 48%, NA19239: 47%). We observed a roughly linear relationship between reciprocal overlap (between 1KG and forestSV calls) and the forestSV event score, with higher-scoring events having greater reciprocal overlap (see Supplementary Figure 5). In addition, we plotted the forestSV event score against event size and found that large duplications tend to receive larger event scores, while deletions tend to be less size-biased in their event scores (Supplementary Figure 7). This inability to reliably call smaller duplications may reflect the scarcity of good training examples for this class of SV (note the small number of duplication examples in Supplementary Figure 14), or it could reflect a smaller number of features that contain information useful for classifying duplications, or both. Calls made by forestSV were compared with the other call sets based on retrotransposon content, segmental duplication content, proximity to centromeres and telomeres, and size distributions (Supplementary Figure 4). In addition, sensitivity was evaluated on the NA12878 gold standard used in ref. 1 (Supplementary Figure 6). We also demonstrate that forestSV does work on low coverage data (5x), and compare performance metrics relative to high coverage data (Supplementary Figure 8). Finally, we show the accumulation of calls as the event score is relaxed in Supplementary Figure 9.

The curves depicted in Supplementary Figure 2 are useful for gauging the performance of methods relative to each other, but cannot be taken as absolute measures of performance and error rates. This particular benchmark used only the most obvious examples of SV events and false positive calls as the reference set. Performance when calling more ambiguous events is unclear and difficult to estimate. Even with extensive experimental validation efforts, some SV calls are difficult to conclusively validate or invalidate, making the calculation of an absolute error rate problematic. In the following section, we augment the performance analysis described here on the 1KG data with error rate estimates based on family relationships in an independent data set.

## Improvement of method performance with increasing number of genomes in the training set

One characteristic of forestSV that sets it apart from other SV discovery techniques is that it dynamically learns what an SV is, using supervised learning techniques. At the time of this writing, forestSV already performs very well in comparison to its peers. However, since its definition of an SV is embodied by the training data from which it learns, rather than a stereotype hard-coded into its algorithms, we anticipate a steady improvement in discovery accuracy as we accumulate more and better training data. Improvement will come through continually educating the classifier with confirmed SVs and newly defined error modes, rather than a fundamental rewrite of the concept.

To demonstrate this phenomenon, we successively trained the classifier on 1, 2, 3, 4, and finally 5 genomes, while at each iteration making calls in a separate left-out genome, NA12878 (Supplementary Figure 11). We found that as we added additional genomes to the training set, the performance of SV discovery in NA12878 improved incrementally (as measured by sensitivity-specificity curves as well as the area under the ROC curve (AUC)). Encouraged by these findings, we will periodically update the classifier on our website as additional training data becomes available.

# Supplementary Data

## File 1

Genomic regions used for training

## File 2

Structural variant calls produced by forestSV in NA12878, NA12891, NA12892, NA19240, NA19238, and NA19239

# References


1 Mills, R. E. *et al.* Mapping copy number variation by population-scale genome sequencing. *Nature* **470**, 59-65 (2011).

2 Conrad, D. F. *et al.* Origins and functional impact of copy number variation in the human genome. *Nature* **464**, 704-712 (2010).

3 McCarroll, S. A. *et al.* Integrated detection and population-genetic analysis of SNPs and copy number variation. *Nat. Genet.* **40**, 1166-1174 (2008).