A comprehensive benchmarking of WGS-based structural variant callers

Varuni Sarwal[1,2], Sebastian Niehus[3,4], Ram Ayyala[1], Sei Chang[1], Angela Lu[1], Nicholas Darci-Maher[1], Russell Littman[1], Karishma Chhugani[5], Arda Soylev[6], Zoia Comarova[7], Emily Wesel[1], Jacqueline Castellanos[1], Rahul Chikka[1], Margaret G. Distler[8], Eleazar Eskin[1,9,10], Jonathan Flint[8†], Serghei Mangul[5,11*†]

[1] Department of Computer Science, University of California Los Angeles, 580 Portola Plaza, Los Angeles, CA 90095, USA

[2] Indian Institute of Technology Delhi, Hauz Khas, New Delhi, Delhi 110016, India

[3] Berlin Institute of Health (BIH), Anna-Louisa-Karsch-Str. 2, 10178 Berlin, Germany

[4] Charité-Universitätsmedizin Berlin, corporate member of Freie Universität Berlin, Humboldt-Universität zu Berlin, and Berlin Institute of Health, Charitéplatz 1, 10117 Berlin, Germany

[5] Department of Clinical Pharmacy, School of Pharmacy, University of Southern California 1985 Zonal Avenue Los Angeles, CA 90089-9121

[6] Konya Gıda Ve Tarım Üniversitesi, Melikşah, Beyşehir Cd. No:9, 42080 Meram/Konya, Turkey

[7] Paradigm Environmental, 3911 Old Lee Highway, Fairfax, VA 22030

[8] Center for Neurobehavioral Genetics, Semel Institute for Neuroscience and Human Behavior, University of California Los Angeles, 760 Westwood Plaza, Los Angeles, CA 90095, USA

[9] Department of Human Genetics, David Geffen School of Medicine at UCLA, 695 Charles E. Young Drive South, Box 708822, Los Angeles, CA, 90095, USA

[10] Department of Computational Medicine, David Geffen School of Medicine at UCLA, 73-235 CHS, Los Angeles, CA, 90095, USA

[11] Quantitative and Computational Biology, University of Southern California, Los Angeles, CA, 90089, USA

* Corresponding author: serghei.mangul@gmail.com.

† These authors contributed equally to the work.

Abstract

Advances in whole genome sequencing promise to enable the accurate and comprehensive structural variant (SV) discovery. Dissecting SVs from whole genome sequencing (WGS) data presents a substantial number of challenges and a plethora of SV-detection methods have been developed. Currently, there is a paucity of evidence which investigators can use to select appropriate SV-detection tools. In this paper, we evaluated the performance of SV-detection tools using a comprehensive PCR-confirmed gold standard set of SVs. In contrast to the previous benchmarking studies, our gold standard dataset included a complete set of SVs allowing us to report both precision and sensitivity rates of SV-detection methods. Our study investigates the ability of the methods to detect deletions, thus providing an optimistic estimate of SV detection performance, as the SV-detection methods that fail to detect deletions are likely to miss more complex SVs. We found that SV-detection tools varied widely in their performance,with several methods providing a good balance between sensitivity and precision. Additionally, we have determined the SV callers best suited for low and ultra-low pass sequencing data.

Introduction

Structural variants (SVs) are genomic regions that contain an altered DNA sequence due to deletion, duplication, insertion, or inversion[1]. SVs are present in approximately 1.5% of the human genome[1,2], but this small subset of genetic variation has been implicated in the pathogenesis of psoriasis[3], Crohn’s disease[4] and other autoimmune disorders[5], autism spectrum and other neurodevelopmental disorders[6–9], and schizophrenia[10–13]. Specialized computational methods—often referred to as SV callers—are capable of detecting structural variants directly from sequencing data. At present, the reliability, sensitivity, and precision of SV callers has not been systematically assessed. We benchmarked currently-available WGS-based SV callers in order to determine the efficacy of available tools and find methods with a good balance between sensitivity and precision.

Substantial differences exist in the number of identified variants in SV catalogs published during the past decade. The 1000 Genomes Project SV dataset identified over 68,000 SVs[14], a genome-wide survey of 769 Dutch individuals identified approximately 1.9 million structural variants[15], and a survey based on profiled whole genomes of 14,891 individuals across diverse global populations identified 498,257 SVs[16]. In addition, discrepancies in the number of SVs reported by these methods suggest that SV callers may fail to detect SVs and may report false positives (i.e., SVs that do not actually exist).

Lack of comprehensive benchmarking makes it impossible to adequately compare the performance of SV callers. In the absence of benchmarking, biomedical studies rely on the consensus of several SV callers[16,17]. In order to compare SV callers given the current lack of a comprehensive gold standard dataset, a recent study[18] used long read technologies to define a ground truth in order to evaluate a large number of currently available tools. However, a comprehensive gold standard dataset is still needed; current long read technologies are prone to producing high error rates, which confounds efforts to detect SVs at single-base pair resolution. In response to the pressing need for a comprehensive gold standard dataset, our paper presents a rigorous assessment of sensitivity and precision of SV-detection tools when applied to mouse data.

Results

Preparing the gold standard data and WGS data

Over the last decade, a plethora of SV-detection methods have been developed (Table 1 and Supplemental Table 1), but the relative performance of these tools is unknown[19–25]. In order to assess the precision and accuracy of currently available SV callers, we simplified the problem presented to the detectors by using a set of homozygous deletions present in inbred mouse chromosomes. Methods failing to detect deletions are likely unreliable for the more challenging task of identifying other SV categories (e.g., insertions, inversions, translocations). We manually curated the mouse deletions used in this benchmarking study, and we used targeted PCR amplification of the breakpoints and sequencing to resolve the ends of each deletion to the base pair[26]. The same read alignment file which was used for manual curation of the deletions was used as an input to the structural variant callers, making our gold standard complete, containing all possible true deletions (true positives). We only used deletions since we could not confidently determine that other forms of SVs could be comprehensively detected with today's SV callers.

The set of deletions we used among seven inbred strains, called with reference to C57BL/6J, is illustrated in Figure 1a and listed in Supplemental Table 2[26]. We filtered out deletions shorter than 50 bp, as such genomic events that are usually detected by indel callers rather than SV callers. In total, we obtained 3,710 deletions with lengths ranging from 50 to 239,572 base pairs (Supplemental Figure 1 and Supplemental Table 2[26]). Almost half of the deletions were in the range of 100-500 bp. Almost 30% of deletions were larger than 1000 bp (Supplemental Figure 1). High coverage sequence data was used as an input to the SV callers in the form of aligned reads. Reads were mapped to the mouse genome (GRCm38 Mouse Build) using BWA with -a option. In total, we obtained 5.2 billion 2x100 bp paired end reads across seven mouse strains. The average depth of coverage was 50.75x (Supplemental Table 3). Details regarding the gold standard and raw data preparation and analysis are presented in the Supplementary Materials.

Choice of SV callers

For this benchmarking study, we selected methods capable of detecting SVs from aligned WGS reads. SV detection algorithms typically use information about coverage profile in addition to the alignment patterns of abnormal reads. We excluded tools that were designed to detect SVs in tumor-normal samples (e.g., Patchwork[86], COPS[87], rSW-seq[88], bic-seq[63], seqCBS[89]) and tools designed to detect only small (less than 50 bp in length) SVs (e.g., GATK[91], Platypus[92], Varscan[93]). Some tools were not suitable for inclusion in our dataset as they were unable to process aligned WGS data (e.g., Magnolya[27]). Other tools were designed solely for long reads (e.g., Sniffles[28]). The complete list of tools excluded from our analysis are provided in Supplemental Table 4. In total, we identified 55 suitable SV methods capable of detecting deletions from WGS data (Table 1 and Supplemental Table 1).

Our benchmarking study produced an analysis of the results generated by 15 SV-detection tools (Table 1). We were able to internally install and run all tools except Biograph, which was run by the developers of the tool. The remaining 40 tools could not be installed and were not included in this study. Supplemental Table 4 presents detailed information about the issues that prevented us from installing these software tools. Commands to install the tools and details of the installation process are provided in the Supplementary Materials.

Comparing the performance of SV callers on mouse WGS data

We compared the performance of 15 SV callers in terms of inferring deletions. The number of deletions detected varied from 899 (indelMINER[29]) to 82,225 (GASV[38]). 53% of the methods reported fewer deletions than are known to be present in the sample (Figure 1b). We allowed deviation in the coordinates of the detected deletions and compared deviations to the coordinates of the true deletions. Even at relaxed stringency, the best method correctly detected the breakpoints of only 20% of known deletions in our curated dataset.

The majority of SV callers typically detect deletions whose coordinates differ from the correct positions by up to 100 bp . Figure 1c and 1d show the true positive (TP) and true negative (TN) rates for the SV callers at four different resolution values. It is notable that some tools with high TP rates also have decreased TN rates. For example, at the 100 bp threshold, the highest TP rate was achieved by CLEVER[30] followed by Manta[80], GRIDSS[39] and DELLY[31] (Figure 1c). However, for the same threshold, GRIDSS[39] and DELLY[31] underperform in the number of correctly detected non-deletions (TNs) compared to Manta[80] (Figure 1d-f). The total number of false negative (FN) and false positive (FP) calls decreased with increase in threshold (Supplemental Figure 2). The FP rate for pindel Popdel[93] was more susceptible to changes in the threshold as compared to Pindel[39], GASV[38]. In general, the length distribution of detected deletions varied across tools and was substantially different from the distribution of true deletions across multiple SV detection methods (Figure 2 and Supplemental Table 2). Deletions detected by BreakDancer[32] were the closest to the true median deletion length, while seven out of 15 SV callers overestimated deletion lengths (Figure 2).

Increasing the resolution threshold increases the number of deletions detected by the SV callers (Figure 1c). Several methods detected all deletions in the sample at 10,000 bp resolution but with precision close to zero (Figure 3b). We used the harmonic mean between precision and sensitivity (F-score) rates to determine the method with the best balance between sensitivity and precision. Several methods (e.g., Manta[80], LUMPY[94]) offered the highest F-score for deletion detection—consistently between 100-10,000 bp resolution across all the mouse strains (Supplemental Figure 3). For a resolution of 10 bp, the method with the best performance for all the samples was LUMPY[94] while at higher resolutions the best performing method was Manta[80](Supplemental Figure 3). The method with the best precision for a threshold of 100-1,000 bp was PopDel[93], but the sensitivity rate of PopDel[93] did not exceed 50% (Figure 3a, Supplemental Figure 4 and 5).

While specificity is often used to compare the deletions detected by tools, we use specificity to provide insights on the tools' ability to predict diploid regions of the genome. Methods that produced a higher F-score tend to also have significantly higher specificity rates with Spearman's correlations greater than 0.66 for thresholds greater than 100 bp (Figure 3d and Supplemental Figure 3 and 6) and are the most balanced in precision and sensitivity; few methods skewed towards just one of the metrics (Figure 3e and Supplemental Figure 7.)

Specificity rate was generally lower for the majority of the methods when compared with sensitivity rate. All methods except CREST[79] with a high precision tend to also have significantly higher specificity rates, with Spearman's correlation greater than 0.84 for thresholds greater than 10 bp (p-value<0.0005) (Supplemental Figure 8). Several tools, such as PopDel[93] and LUMPY[94], were able to balance precision and specificity, with rates exceeding 70% for each metric (Figure 3f). Manta[80], LUMPY[94] and CLEVER[30] were the only methods able to successfully balance precision and sensitivity, with rates above 50% for each metric (Figure 3e and Supplemental Figure 7). CLEVER[30] was able to achieve the highest sensitivity rate at the majority of thresholds (Figure 3a and Supplemental Figure 5). The most precise method we observed was PopDel[93], with rates exceeding 80% for thresholds 1000 bp onwards, but the sensitivity of this method was two times lower than the majority of other tools (Figure 3b).

We examined whether the SV callers included in this study maintained similar SV detection accuracy across the different mouse strains. We compared results from each tool to study how consistent the results were across the samples. Among the tools with a sensitivity rate above 10%, LUMPY[94] maintained the most consistent sensitivity rate across samples, with the highest rate of 60% when applied to both C3H/HeJ and CBA/J strains. The lowest sensitivity rate achieved by LUMPY[94] was 58% for A/J and DBA/2J strains. Several tools, such as CREST[79] and PopDel[93] maintained a consistent specificity across mouse strains (Figure 3c and Supplemental Figure 6). Sensitivity rates were the most stable across the 7 different strains (Supplemental Figure 5). Specificity showed the highest variability among strains compared to other measures

(Supplemental Figure 6)**.** Precision shows the second highest variability across the strains, with the most stable results provided by Pindel[39] and indelMINER[29] (Supplemental Figure 4).

We have also compared CPU time and the maximum amount of RAM used by each of the tools. Across all of the tools, GASV[38] required the highest amount of RAM while PopDel[93] required the lowest amount of RAM to run the analysis. CREST[79] required the longest amount of time to perform the analysis. Breakdancer[32] was the fastest tool. We have also compared the computational resources and speed of SV callers based on datasets with full coverage and those with ultra-low coverage (Supplemental Figure 9).

Performance of SV-detection tools on low and ultra-low coverage data

We assessed the performance of SV callers at different coverage depths generated by down-sampling the original WGS data. The simulated coverage ranged from 32x to 0.1x, and ten subsamples were generated for each coverage range. For each method, the number of correctly detected deletions generally decreased as the coverage depth decreased (Supplemental Figure 10). Some of the methods were able to call deletions from ultra low coverage (<=0.5x) data. While tools like Manta[80] reached a precision of 82%, the overall sensitivity, specificity, and F-score values were less than 8% for all tools. None of the methods were able to detect deletions from 0.1x coverage.

As suggested by other studies[89], most tools reached a maximum precision and specificity at an intermediate coverage (Figure 4b-c). Both the sensitivity rate and the F-score improved as the coverage increased (Figure 4a,d). Overall, DELLY[31] showed the highest F-score for coverage below 4x (Figure 4d). For coverage between 8 and 32x, Manta[80] showed the best performance. LUMPY[94] was the only tool to attain precision above 90% for coverages 1x to 4x. However, a decreased sensitivity in coverages below 4x led to a decreased F-score when compared to DELLY[31]. Precision in results from DELLY[31] for ultra-low coverage data was above 90% when the threshold was set at 1000 bp, but changing the threshold had no effect on LUMPY[94] (Supplemental Figure 11).

Length of deletions impacts the performance of the SV callers

We separately assessed the effect of deletion length on the accuracy of detection for four categories of deletions (Figure 5). The performance of the SV callers was significantly affected by deletion length. For example, for deletions shorter than 100 bp, precision, specificity, and F-score values were typically below 40% regardless of the tool (Figure 5b,c,d and Supplemental Figures 12, 14, 15), while sensitivity values were above 50% for several tools (Figure 5a) (Supplemental Figure 13). For deletions longer than 100 bp, the best performing tool in terms of sensitivity and precision significantly varied depending on the deletion length (Figure 5a,b). CLEVER[30] provided a sensitivity of above 60% for deletions less than 500 bp, however DELLY[31] provided the highest sensitivity for deletions longer than 500 bp (Figure 5a and Supplemental Figures 17, 21, 25) . LUMPY[94] delivered the best precision for deletion lengths from 50-500 bp, and CLEVER[30] performed well for longer deletion lengths (Figure 5b and Supplemental Figure 14, 18, 22, 26). indelMINER[29] provided the high precision rate of detection of deletions in the range of 100 bp-500 bp and when longer than 1000 bp, but the precision of detecting deletion in the 500 bp-1000 bp range was lower than that of other tools (Figure 5b). In general, Manta[80] and LUMPY[94] were the only methods able to deliver an F-score above 30% across all categories (Figure 5d and Supplemental Figure 15, 19, 23, 27). Manta[80] and CREST[79] were the only tools with high specificity for deletions shorter than 500bp. For all the other tools, specificity was low across all the categories, except for deletions with lengths higher than 1,000bp (Figure 5c and Supplemental Figure 12, 16, 20, 24).

Discussion

In this paper, we performed a systematic benchmarking of algorithms to identify structural variants (SVs) from whole-genome sequencing data. In contrast to methods which are used to identify single nucleotide polymorphisms and have coalesced around a small number of approaches, there is currently no consensus on the best way to detect SVs in mammalian genomes. Indeed, we were able to find 56 different methods, each claiming relatively high specificity and sensitivity rates in the original publication. Upon applying the tools to our curated datasets, many did not perform as reported in the original publication. This discrepancy may be because molecular data were not used in the analyses performed for the original publication. Instead, authors often solely derive conclusions from simulated data that may fail to capture the full complexity of real sequencing data[33].

In comparison to previous benchmarking efforts based on simulated data [19,21,25,34,35], we obtained and employed a set of molecularly defined deletions for which breakpoints are known at base pair resolution. Other benchmarking studies have employed long-read-based gold standard datasets with approximate coordinates of deletions[18]. Long read technologies are often unable to cover the entire genome at sufficient resolution for precise SV characterization. In addition, long-read technologies carry a high error rate that limits their ability to detect SVs at single-base pair resolution. Our benchmarking method, using a gold standard set of molecular-defined deletions, overcomes the limitations of simulated data and incomplete characterization. Thus, our benchmarking study represents a robust assessment of the performance of currently-available SV detection methods when applied to a representative data set.

When installing the majority of SV callers, we noticed significant difficulties due to inadequate software implementation and technical factors[36]. Deprecated dependencies and segmentation faults were the most common reasons preventing successful tool installation[37]. The majority of the tools have a consensus on the output format to be used (Table S6) , but the requirements for the format varied among tools. Lack of documentation about format requirements may further limit the use of SV callers.

We identified a series of factors that determined the performance of SV-caller methods. The most important factors were the size of deletions and the coverage of WGS data. For example, BreakDancer[32] only detected deletions larger than 100 bp. Some tools achieved excellent sensitivity, with the caveat that their precision was close to zero. For example, Pindel[32] achieved the highest sensitivity rate among all the tools, with a precision rate of less than 0.1%. Other tools (e.g., PopDel[93]) employ a more conservative SV detection approach, resulting in higher precision at the cost of decreased sensitivity for smaller deletion events. Few tools were able to maintain a good balance between precision and sensitivity. For example, Manta[80], CLEVER[30], LUMPY[94], BreakDancer[32], and BioGraph maintained both precision and sensitivity rates above 40%. In addition to differences in the accuracy of SV detection, we observed significant differences in run times and required computational resources (Supplemental Figure 9).

We envision that future SV-caller methods should enable detection of deletions with precise coordinates. The inability of current methods to precisely detect breakpoints was coupled with the issue of 61.5% tools underestimating the true size of SVs. A limitation of our benchmarking study is that our gold standard used inbred homozygous mouse genomes, which potentially poses as an easier target for assessment when compared to heterozygous human genomes. Additionally, the human genomes, for which most SV callers were designed, contain a higher number of repetitive regions than does the mouse, posing an additional challenge which is not reflected in mouse-based gold standard datasets.

Data availability

WGS mouse samples used for benchmarking of SV-callers are available under the A/J (ERP000038), AKR/J (ERP000037), BALB/cJ (ERP000039), C3H/HeJ (ERP000040), DBA/2J (ERP000044), LP/J (ERP000045) accession numbers in the European Nucleotide Archive. VCF file with true deletions from gold standard, and the output VCF's produced by the tools, the gold standard VCF's, the analysis scripts, and figures are available at https://github.com/Mangul-Lab-USC/benchmarking-sv-callers-paper/

Code availability

Source code to compare SV detection methods and to produce the figures contained within this text is open source, free to use under the MIT license, and available at https://github.com/Mangul-Lab-USC/benchmarking-sv-callers-paper/

Acknowledgments

We thank Dr. Lana Martin for the helpful discussions and comments on the manuscript. We thank our colleagues Niranjan Shekar, Lisa Herta, and Adam English at Spiral Genetics for developing and running their tool, BioGraph software (http://www.spiralgenetics.com/biograph-engine) on mouse WGS data. We thank the authors of the tools surveyed in this work—Cenk Sahinalp, Ryan Layer, Ira Hall, Tony Papenfuss, Gerton Lunter, Michael Schatz, Alexander Schoenhuth, Ken Chen, Aakrosh Ratan, and Tobias Rausch—for providing helpful feedback and verifying information related to their tool.

Author contributions

V.S. created scripts for running and evaluating the software tools. E.W., J.C., M.G.D., N.D.-M., R.A., R.C., R.L., S.C., and V.S. contributed to installing, running, and evaluating software tools.

S.N. applied GRIDSS[39], LUMPY[94], Manta[80], and PopDel[93] to the mouse data and discussed evaluation metrics. R.A. curated data and prepared mouse data for evaluation with the software tools. A.L., R.A., and V.S. generated the figures; R.A. and V.S. generating the tables. E.E., J.F., M.G.D., S.M., S.N., and V.S. wrote, reviewed, and edited the manuscript. J.F. and S.M. led the project.

Funding

S.N. is funded by the BMBF Grant #031L0180 from the German Federal Ministry for Education and Research.

Competing interests

The authors declare that they have no competing interests.

References Cited

1. Feuk, L. *et al.* Absence of a paternally inherited FOXP2 gene in developmental verbal dyspraxia. *Am. J. Hum. Genet.* **79**, 965–972 (2006).
2. Pang, A. W. *et al.* Towards a comprehensive structural variation map of an individual human genome. *Genome Biol.* **11**, R52 (2010).
3. Hollox, E. J., Barber, J. C. K., Brookes, A. J. & Armour, J. A. L. Defensins and the dynamic

genome: what we can learn from structural variation at human chromosome band 8p23.1. *Genome Res.* **18**, 1686–1697 (2008).

4. McCarroll, S. A. *et al.* Deletion polymorphism upstream of IRGM associated with altered IRGM expression and Crohn's disease. *Nat. Genet.* **40**, 1107–1112 (2008).
5. Fanciulli, M. *et al.* FCGR3B copy number variation is associated with susceptibility to systemic, but not organ-specific, autoimmunity. *Nat. Genet.* **39**, 721–723 (2007).
6. Girirajan, S. & Eichler, E. E. De novo CNVs in bipolar disorder: recurrent themes or new directions? *Neuron* vol. 72 885–887 (2011).
7. Pinto, D. *et al.* Functional impact of global rare copy number variation in autism spectrum disorders. *Nature* **466**, 368–372 (2010).
8. Sanders, S. J. *et al.* Multiple recurrent de novo CNVs, including duplications of the 7q11.23 Williams syndrome region, are strongly associated with autism. *Neuron* **70**, 863–885 (2011).
9. Elia, J. *et al.* Genome-wide copy number variation study associates metabotropic glutamate receptor gene networks with attention deficit hyperactivity disorder. *Nat. Genet.* **44**, 78–84 (2011).
10. Kirov, G. *et al.* De novo CNV analysis implicates specific abnormalities of postsynaptic signalling complexes in the pathogenesis of schizophrenia. *Mol. Psychiatry* **17**, 142–153 (2012).
11. Stefansson, H. *et al.* Large recurrent microdeletions associated with schizophrenia. *Nature* **455**, 232–236 (2008).
12. Walsh, T. *et al.* Rare structural variants disrupt multiple genes in neurodevelopmental

pathways in schizophrenia. *Science* **320**, 539–543 (2008).

13. Marshall, C. R. *et al.* Contribution of copy number variants to schizophrenia from a genome-wide study of 41,321 subjects. *Nat. Genet.* **49**, 27–35 (2017).
14. Sudmant, P. H. *et al.* An integrated map of structural variation in 2,504 human genomes. *Nature* **526**, 75–81 (2015).
15. Hehir-Kwa, J. Y. *et al.* A high-quality human reference panel reveals the complexity and distribution of genomic structural variants. *Nat. Commun.* **7**, 12989 (2016).
16. Collins, R. L. *et al.* An open resource of structural variation for medical and population genetics. *bioRxiv* 578674 (2019) doi:10.1101/578674.
17. Werling, D. M. *et al.* An analytical framework for whole-genome sequence association studies and its implications for autism spectrum disorder. *Nat. Genet.* **50**, 727–736 (2018).
18. Kosugi, S. *et al.* Comprehensive evaluation of structural variation detection algorithms for whole genome sequencing. *Genome Biol.* **20**, 117 (2019).
19. Alkodsi, A., Louhimo, R. & Hautaniemi, S. Comparative analysis of methods for identifying somatic copy number alterations from deep sequencing data. *Brief. Bioinform.* **16**, 242–254 (2015).
20. Pabinger, S., Rödiger, S., Kriegner, A., Vierlinger, K. & Weinhäusel, A. A survey of tools for the analysis of quantitative PCR (qPCR) data. *Biomolecular Detection and Quantification* **1**, 23–33 (2014).
21. Duan, J., Zhang, J.-G., Deng, H.-W. & Wang, Y.-P. Comparative studies of copy number variation detection methods for next-generation sequencing technologies. *PLoS One* **8**, e59128 (2013).

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

23. Legault, M.-A., Girard, S., Lemieux Perreault, L.-P., Rouleau, G. A. & Dubé, M.-P. Comparison of sequencing based CNV discovery methods using monozygotic twin quartets. *PLoS One* **10**, e0122287 (2015).

24. Hasan, M. S., Wu, X. & Zhang, L. Performance evaluation of indel calling tools using real short-read data. *Hum. Genomics* **9**, 20 (2015).

25. Neuman, J. A., Isakov, O. & Shomron, N. Analysis of insertion-deletion from deep-sequencing data: software evaluation for optimal detection. *Brief. Bioinform.* **14**, 46–55 (2013).

26. Keane, T. M. *et al.* Mouse genomic variation and its effect on phenotypes and gene regulation. *Nature* **477**, 289–294 (2011).

27. Nijkamp, J. F. *et al.* De novo detection of copy number variation by co-assembly. *Bioinformatics* **28**, 3195–3202 (2012).

28. Sedlazeck, F. J. *et al.* Accurate detection of complex structural variations using single-molecule sequencing. *Nat. Methods* **15**, 461–468 (2018).

29. Ratan, A., Olson, T. L., Loughran, T. P., Jr & Miller, W. Identification of indels in next-generation sequencing data. *BMC Bioinformatics* **16**, 42 (2015).

30. Marschall, T. *et al.* CLEVER: clique-enumerating variant finder. *Bioinformatics* **28**, 2875–2882 (2012).

31. Rausch, T. *et al.* DELLY: structural variant discovery by integrated paired-end and split-read analysis. *Bioinformatics* **28**, i333–i339 (2012).

32. Fan, X., Abbott, T. E., Larson, D. & Chen, K. BreakDancer: Identification of Genomic Structural Variation from Paired-End Read Mapping. in *Current Protocols in Bioinformatics* 15.6.1–15.6.11 (2014).

33. Mangul, S. *et al.* Systematic benchmarking of omics computational tools. *Nat. Commun.* **10**, 1393 (2019).

34. Guan, P. & Sung, W.-K. Structural variation detection using next-generation sequencing data: A comparative technical review. *Methods* **102**, 36–49 (2016).

35. Cameron, D. L., Di Stefano, L. & Papenfuss, A. T. Comprehensive evaluation and characterisation of short read general-purpose structural variant calling software. *Nat. Commun.* **10**, 3240 (2019).

36. Mangul, S., Martin, L. S., Eskin, E. & Blekhman, R. Improving the usability and archival stability of bioinformatics software. *Genome Biol.* **20**, 47 (2019).

37. Mangul, S. *et al.* Challenges and recommendations to improve the installability and archival stability of omics computational tools. *PLoS Biol.* **17**, e3000333 (2019).

38. Sindi, S., Helman, E., Bashir, A. & Raphael, B. J. A geometric approach for classification and comparison of structural variants. *Bioinformatics* **25**, i222–30 (2009).

39. Ye, K., Schulz, M. H., Long, Q., Apweiler, R. & Ning, Z. Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics* **25**, 2865–2871 (2009).

40. Lin, Y.-Y. *et al.* Computational identification of micro-structural variations and their proteogenomic consequences in cancer. *Bioinformatics* **34**, 1672–1681 (2018).

41. Qi, J. & Zhao, F. inGAP-sv: a novel scheme to identify and visualize structural variation

from paired end mapping data. - PubMed - NCBI. https://www.ncbi.nlm.nih.gov/pubmed/21715388.

42. Medvedev, P., Fiume, M., Dzamba, M., Smith, T. & Brudno, M. Detecting copy number variation with mated short reads. *Genome Res.* **20**, 1613–1622 (2010).

43. Jiang, Y., Wang, Y. & Brudno, M. PRISM: Pair-read informed split-read mapping for base-pair level detection of insertion, deletion and structural variants. *Bioinformatics* **28**, 2576–2583 (2012).

44. Karakoc, E. *et al.* Detection of structural variants and indels within exome data. *Nat. Methods* **9**, 176–178 (2011).

45. Hajirasouliha, I. *et al.* Detection and characterization of novel sequence insertions using paired-end next-generation sequencing. *Bioinformatics* **26**, 1277–1283 (2010).

46. Yang, L. *et al.* Diverse mechanisms of somatic structural variations in human cancer genomes. *Cell* **153**, 919 (2013).

47. Lee, S., Hormozdiari, F., Alkan, C. & Brudno, M. MoDIL: detecting small indels from clone-end sequencing with mixtures of distributions. *Nat. Methods* **6**, 473–474 (2009).

48. Korbel, J. O. *et al.* PEMer: a computational framework with simulation-based error models for inferring genomic structural variants from massive paired-end sequencing data. *Genome Biol.* **10**, R23 (2009).

49. Iakovishina, D., Janoueix-Lerosey, I., Barillot, E., Regnier, M. & Boeva, V. SV-Bay: structural variant detection in cancer genomes using a Bayesian approach with correction for GC-content and read mappability. *Bioinformatics* **32**, 984–992 (2016).

50. Mimori, T. *et al.* iSVP: an integrated structural variant calling pipeline from

high-throughput sequencing data. *BMC Syst. Biol.* **7**, 1–8 (2013).

51. Vikas Bansal, O. L. A probabilistic method for the detection and genotyping of small indels from population-scale sequence data. *Bioinformatics* **27**, 2047 (2011).
52. Albers, C. A. *et al.* Dindel: accurate indel calls from short-read data. *Genome Res.* **21**, 961–973 (2011).
53. Bartenhagen, C. & Dugas, M. Robust and exact structural variation detection with paired-end and soft-clipped alignments: SoftSV compared with eight algorithms. *Brief. Bioinform.* **17**, 51–62 (2016).
54. Dan, S. *et al.* Prenatal detection of aneuploidy and imbalanced chromosomal arrangements by massively parallel sequencing. *PLoS One* **7**, e27835 (2012).
55. Suzuki, S., Yasuda, T., Shiraishi, Y., Miyano, S. & Nagasaki, M. ClipCrop: a tool for detecting structural variations with single-base resolution using soft-clipping information. *BMC Bioinformatics* **12**, S7 (2011).
56. Emde, A.-K. *et al.* Detecting genomic indel variants with exact breakpoints in single- and paired-end sequencing data using SplazerS. *Bioinformatics* **28**, 619–627 (2012).
57. Chen, K. *et al.* TIGRA: A targeted iterative graph routing assembler for breakpoint assembly. *Genome Res.* **24**, 310 (2014).
58. Simpson, J. T., McIntyre, R. E., Adams, D. J. & Durbin, R. Copy number variant detection in inbred strains from short read sequence data. *Bioinformatics* **26**, 565 (2010).
59. Magi, A., Benelli, M., Yoon, S., Roviello, F. & Torricelli, F. Detecting common copy number variants in high-throughput sequencing data by using JointSLM algorithm. *Nucleic Acids Res.* **39**, e65 (2011).

60. Ivakhno S, E. al. CNAseg--a novel framework for identification of copy number changes in cancer from second-generation sequencing data. - PubMed - NCBI. https://www.ncbi.nlm.nih.gov/pubmed/20966003.

61. Sindi, S. S., Önal, S., Peng, L. C., Wu, H.-T. & Raphael, B. J. An integrative probabilistic model for identification of structural variation in sequencing data. *Genome Biol.* **13**, R22 (2012).

62. Wong, K., Keane, T. M., Stalker, J. & Adams, D. J. Enhanced structural variant and breakpoint detection using SVMerge by integration of multiple detection methods and local assembly. *Genome Biol.* **11**, R128 (2010).

63. Xi, R., Lee, S., Xia, Y., Kim, T.-M. & Park, P. J. Copy number analysis of whole-genome data using BIC-seq2 and its application to detection of cancer susceptibility variants. *Nucleic Acids Res.* **44**, 6274–6286 (2016).

64. Chiara, M., Pesole, G. & Horner, D. S. SVM2: an improved paired-end-based tool for the detection of small genomic structural variations using high-throughput single-genome resequencing data. *Nucleic Acids Res.* **40**, e145 (2012).

65. Bellerophon. http://cbc.case.edu/Bellerophon/.

66. Moncunill, V. *et al.* Comprehensive characterization of complex structural variations in cancer by directly comparing genome sequence reads. *Nat. Biotechnol.* **32**, 1106–1112 (2014).

67. Xiao, F. *et al.* modSaRa: a computationally efficient R package for CNV identification. *Bioinformatics* **33**, 2384–2385 (2017).

68. Miller, C. A., Hampton, O., Coarfa, C. & Milosavljevic, A. ReadDepth: A Parallel R

Package for Detecting Copy Number Alterations from Short Sequencing Reads. *PLoS One* **6**, e16327 (2011).

69. Budczies, J. *et al.* Ioncopy: an R Shiny app to call copy number alterations in targeted NGS data. *BMC Bioinformatics* **19**, 157 (2018).

70. Website. https://github.com/xyc0813/SVmine.

71. Noll, A. C. *et al.* Clinical detection of deletion structural variants in whole-genome sequences. *NPJ Genom Med* **1**, 16026 (2016).

72. Abyzov, A., Urban, A. E., Snyder, M. & Gerstein, M. CNVnator: an approach to discover, genotype, and characterize typical and atypical CNVs from family and population genome sequencing. *Genome Res.* **21**, 974–984 (2011).

73. Lindberg, M. R., Hall, I. M. & Quinlan, A. R. Population-based structural variation discovery with Hydra-Multi. *Bioinformatics* **31**, 1286–1289 (2015).

74. Klambauer, G. *et al.* cn.MOPS: mixture of Poissons for discovering copy number variations in next-generation sequencing data with a low false discovery rate. *Nucleic Acids Res.* **40**, e69 (2012).

75. Zhao, H. & Zhao, F. BreakSeek: a breakpoint-based algorithm for full spectral range INDEL detection. *Nucleic Acids Res.* **43**, 6701–6713 (2015).

76. Li, H. FermiKit: assembly-based variant calling for Illumina resequencing data. *Bioinformatics* **31**, 3694–3696 (2015).

77. Lam, H. Y. K. *et al.* Nucleotide-resolution analysis of structural variants using BreakSeq and a breakpoint library. *Nat. Biotechnol.* **28**, 47–55 (2010).

78. Toolkit for automated and rapid discovery of structural variants. *Methods* **129**, 3–7 (2017).

79. Wang, J. *et al.* CREST maps somatic structural variation in cancer genomes with base-pair resolution. *Nat. Methods* **8**, 652–654 (2011).

80. Chen, X. *et al.* Manta: rapid detection of structural variants and indels for germline and cancer sequencing applications. *Bioinformatics* **32**, 1220–1222 (2016).

81. Stancu, M. C. *et al.* Mapping and phasing of structural variation in patient genomes using nanopore sequencing. *Nat. Commun.* **8**, 1–13 (2017).

82. English, A. C., Salerno, W. J. & Reid, J. G. PBHoney: identifying genomic variants via long-read discordance and interrupted mapping. *BMC Bioinformatics* **15**, 1–7 (2014).

83. Hormozdiari, F. *et al.* Next-generation VariationHunter: combinatorial algorithms for transposon insertion discovery. *Bioinformatics* **26**, i350 (2010).

84. Kim, M., Farnoud, F. & Milenkovic, O. HyDRA: gene prioritization via hybrid distance-score rank aggregation. *Bioinformatics* **31**, 1034–1043 (2015).

85. Schröder, J. *et al.* Socrates: identification of genomic rearrangements in tumour genomes by re-aligning soft clipped reads. *Bioinformatics* **30**, 1064 (2014).

86. Mayrhofer, M., DiLorenzo, S. & Isaksson, A. Patchwork: allele-specific copy number analysis of whole-genome sequenced tumor tissue. *Genome Biol.* **14**, 1–10 (2013).

87. Krishnan, N. M., Gaur, P., Chaudhary, R., Rao, A. A. & Panda, B. COPS: A Sensitive and Accurate Tool for Detecting Somatic Copy Number Alterations Using Short-Read Sequence Data from Paired Samples. *PLoS One* **7**, (2012).

88. Kim, T.-M., Luquette, L. J., Xi, R. & Park, P. J. rSW-seq: Algorithm for detection of copy number alterations in deep sequencing data. *BMC Bioinformatics* **11**, 1–13 (2010).

89. Chen, H., Bell, J. M., Zavala, N. A., Ji, H. P. & Zhang, N. R. Allele-specific copy number

profiling by next-generation DNA sequencing. *Nucleic Acids Res.* **43**, e23 (2015).

90. McKenna, A. *et al.* The Genome Analysis Toolkit: A MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res.* **20**, 1297–1303 (2010).

91. Rimmer, A. *et al.* Integrating mapping-, assembly- and haplotype-based approaches for calling variants in clinical sequencing applications. *Nat. Genet.* **46**, 912–918 (2014).

92. Koboldt, D. C. *et al.* VarScan 2: Somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Res.* **22**, 568–576 (2012).

93. Roskosch, Sebastian, et al. "PopDel identifies medium-size deletions jointly in tens of thousands of genomes." *bioRxiv* (2019): 740225.

94. Layer, Ryan M., et al. "LUMPY: a probabilistic framework for structural variant discovery." *Genome biology* 15.6 (2014): R84.

95. Yoon, Seungtai, et al. "Sensitive and accurate detection of copy number variants using read depth of coverage." *Genome research* 19.9 (2009): 1586-1592.

96. Sedlazeck, Fritz J., et al. "Accurate detection of complex structural variations using single-molecule sequencing." Nature methods 15.6 (2018): 461-468.

97. Kosugi, Shunichi, et al. "Comprehensive evaluation of structural variation detection algorithms for whole genome sequencing." Genome biology 20.1 (2019): 117.

Tables

Software tool	Version	Underlying algorithm	Published year	Tool's webpage	Bioconda version	Format
GASV[38]	1.4	RP	2009	http://compbio.cs.brown.edu/projects/GASV/	No	Custom
Pindel[39]	0.2.5b9	RP+SR SR (2015,0.2.5b9)	2009	http://gmt.genome.wustl.edu/packages/pindel/	Yes	Custom
RDXplorer[95]	3.2	RD	2009	http://RDXplorer.sourceforge.net/	No	Custom
CLEVER[30]	2.4	RP	2012	https://bitbucket.org/tobiasmarschall/CLEVER-toolkit/wiki/Home	Yes	Custom
DELLY[3]	0.8.	RP+S	2012	https://github.com/DELLYtools/D	Yes	Custom

[1]	2	R		ELLY		
BreakDancer[32]	1.3.6	RP	2014	https://github.com/genome/BreakDancer	Yes	Custom
indelMINER[29]	N/A	RP+SR	2015	https://github.com/aakrosh/indelMINER	No	VCF
GRIDSS[39]	2.5.1	RP+SR	2017	https://github.com/PapenfussLab/GRIDSS	Yes	VCF
MiStrVar[40]	N/A	N/A	2017	https://bitbucket.org/compbio/MiStrVar	No	VCF
LUMPY[94]	0.2.4	RP, SR, RD	2018	https://github.com/brentp/smoove	Yes	VCF
PopDel[93]	1.1.3	RP	2019	https://github.com/kehrlab/PopDel	Yes	VCF
BioGraph*	5.0.1	N/A	2020	http://www.spiralgenetics.com/biograph-engine	No	VCF
CREST[79]	1.0	SR	2011	https://www.stjuderesearch.org/site/lab/zhang	No	Custom

Manta[80]	1.6.0	SR	2016	https://github.com/Illumina/manta	Yes	VCF
genomeSTRIP[71]	2.0	RP+SR+RD	2015	http://software.broadinstitute.org/software/genomestrip/	Yes	VCF

Table 1. Overview of SV-detection methods included in this study. Surveyed SV-detection methods sorted by their year of publication from 2009 to 2018 are listed along with their underlying algorithm: Read-depth (RC), Read-Pair Algorithms (RP), Split-Read Approaches (SR), Discordant Pairs (DP), or a combination of algorithms. We documented the version of the software tool used in the study ('Version'), the year the software tool was published ('Published year'), the webpage where each SV-detection method is hosted ('Tool's webpage'), and whether or not the Bioconda package of the software was available ('Bioconda version'). Asterisk (*) denotes that the method was proprietary.

Figures

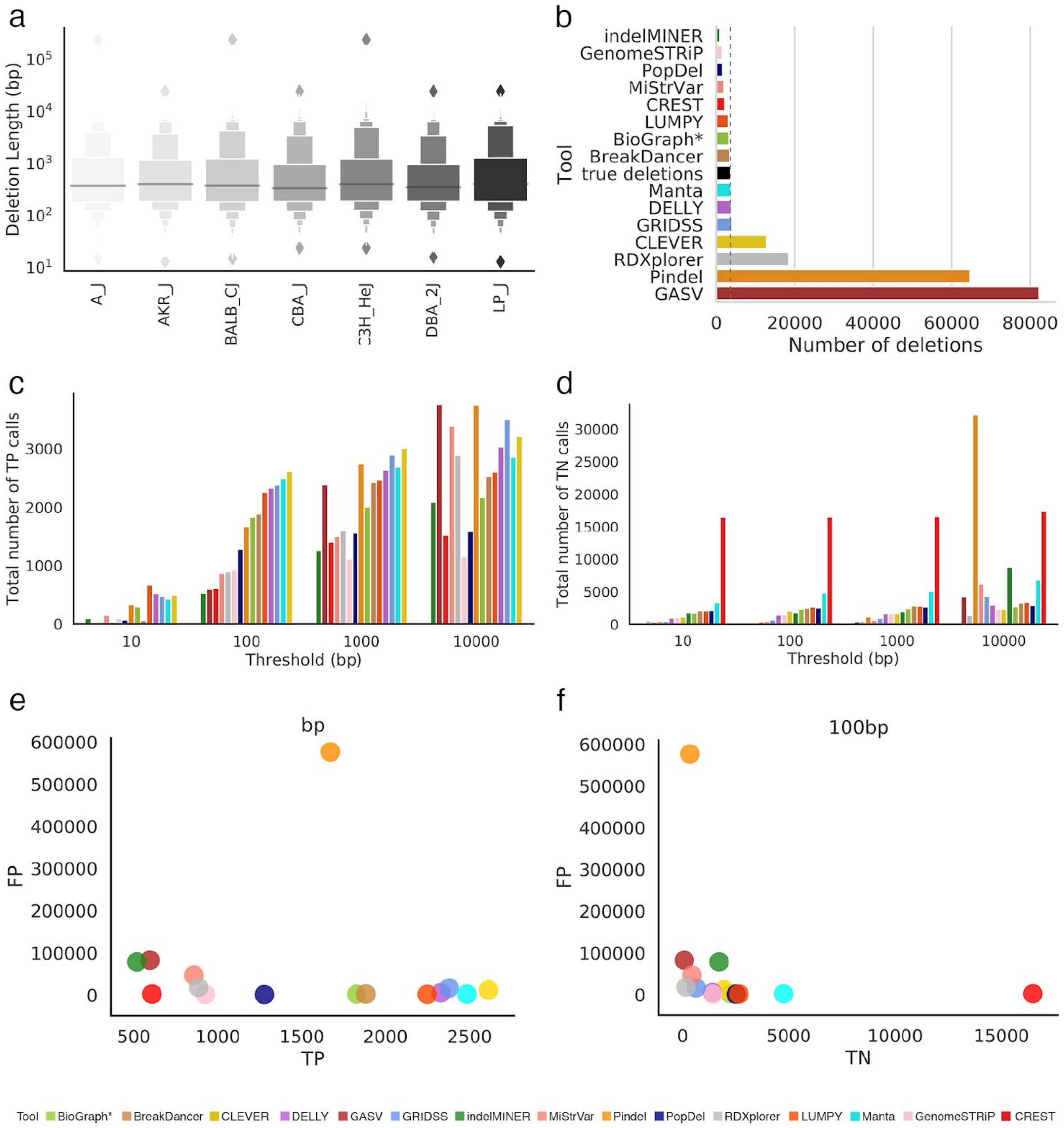


Figure 1. Comparison of inferred deletions across SV callers on mouse data. (a) Length distribution of molecularly-confirmed deletions from chromosome 19 across seven strains of mice. (b) Number of molecularly-confirmed deletions ('true deletions' black color) and number of deletions detected by SV callers. (c) Barplot depicting the total number of true positive calls

across all error thresholds for each SV caller. (d) Barplot depicting the total number of true negative calls across all error thresholds for each SV caller. (e) Scatter plot depicting number of correctly detected deletions (true positives - ‘TP’) by number of incorrectly detected deletions (false positives - ‘FP’) at the 100 bp threshold. Deletion is considered to be correctly predicted if the distance of right and left coordinates are within the given threshold from the coordinates of true deletion. (f) Scatter plot depicting number of correctly detected non-deletions (true negatives - ‘TN’) by number of incorrectly detected deletions (false positives - ‘FP’) at the 100 bp threshold. An SV caller was considered to detect a given non-deletion if no deletions were reported in a given region.

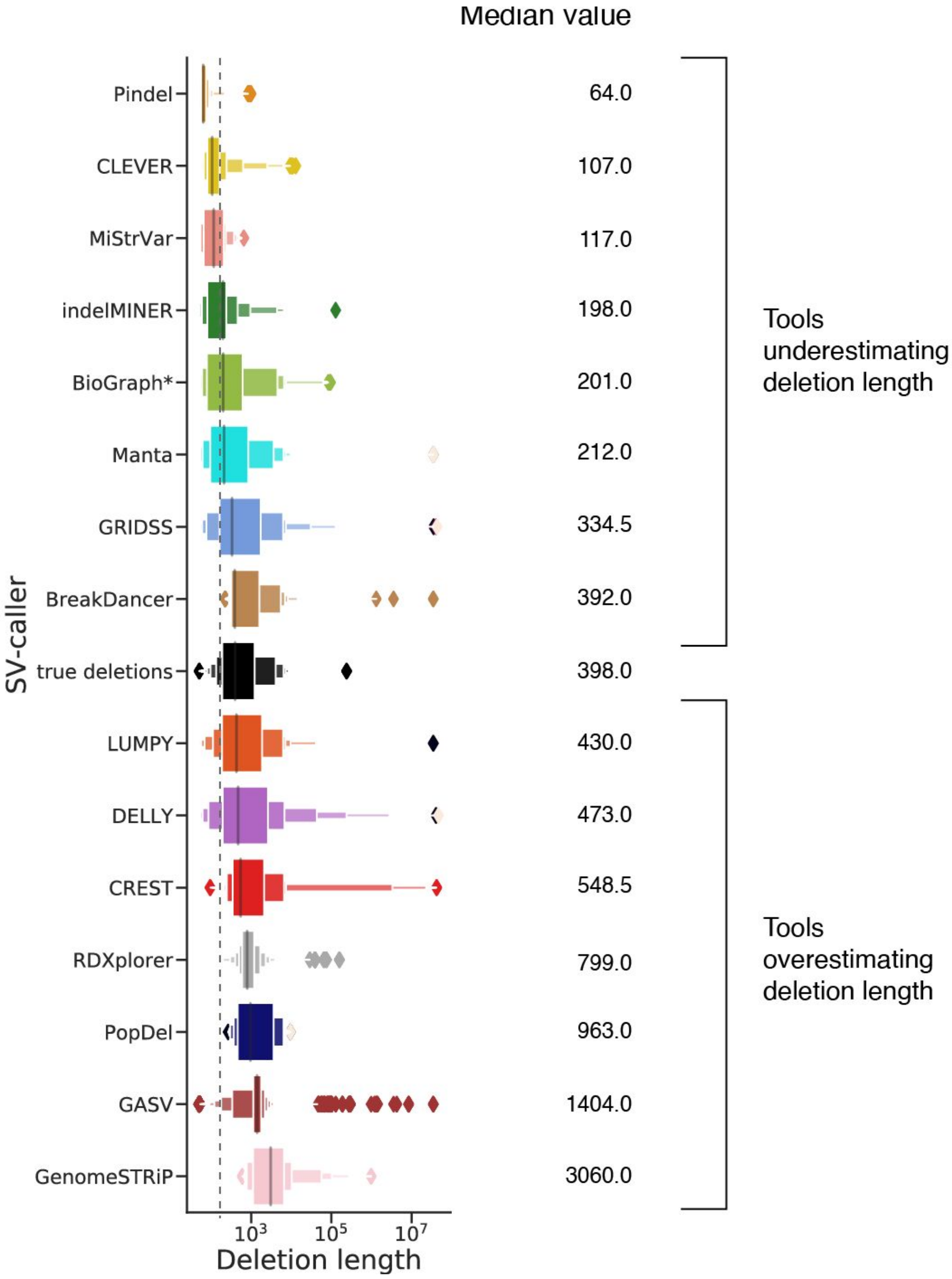

Median value
Pindel 64.0
CLEVER 107.0
MiStrVar 117.0
indelMINER 198.0
BioGraph* 201.0
Manta 212.0
GRIDSS 334.5
BreakDancer 392.0
true deletions 398.0
LUMPY 430.0
DELLY 473.0
CREST 548.5
RDXplorer 799.0
PopDel 963.0
GASV 1404.0
GenomeSTRiP 3060.0
Tools underestimating deletion length
Tools overestimating deletion length
SV-caller
10^3
10^5
10^7
Deletion length

Figure 2. Length distribution of deletions detected by each SV caller. True deletions indicated in black. Tools were sorted in increasing order based on their median deletion length. The vertical dashed line corresponds to the median value of true deletions

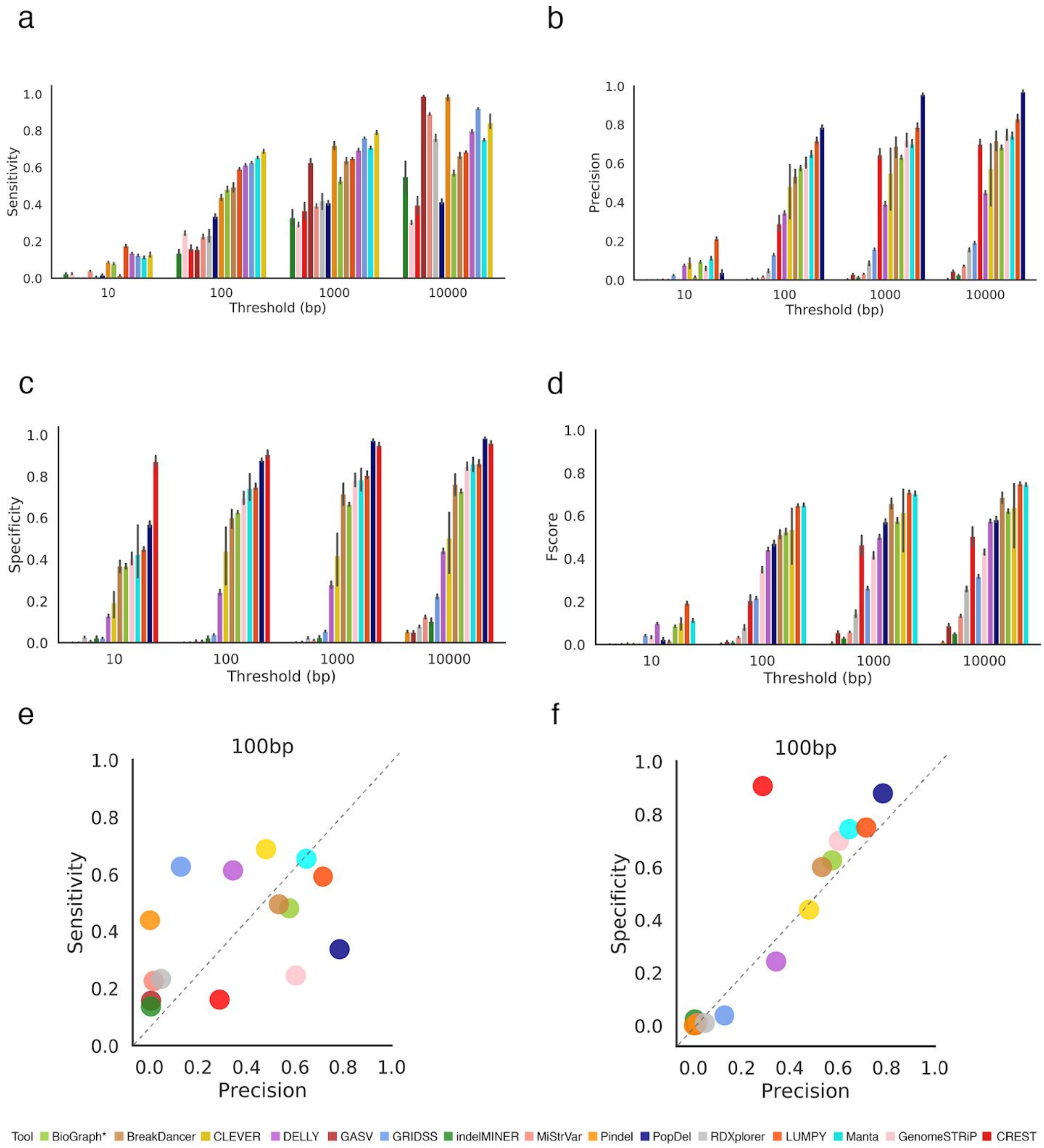


Figure 3. Comparing the performance of SV callers based on whole genome (WGS) data across seven inbred mouse strains. A deletion is considered to be correctly predicted if the distance of right and left coordinates are within the threshold τ from the coordinates of a true

deletion. (a) Sensitivity of SV callers at different thresholds. (b) Precision of SV callers at different thresholds. (c) Specificity of SV callers at different thresholds. (d) F-score of SV callers at different thresholds. (e) Scatter plot depicting the Precision (x-axis) and Sensitivity (y-axis) for 100 bp threshold. (f) Scatter plot depicting the Precision(x-axis) and Specificity (y-axis) 100 bp threshold. Figures (a-d) are sorted in increasing order based on their performance at 100bp threshold. Results for other thresholds are presented in Supplemental Figure 6.

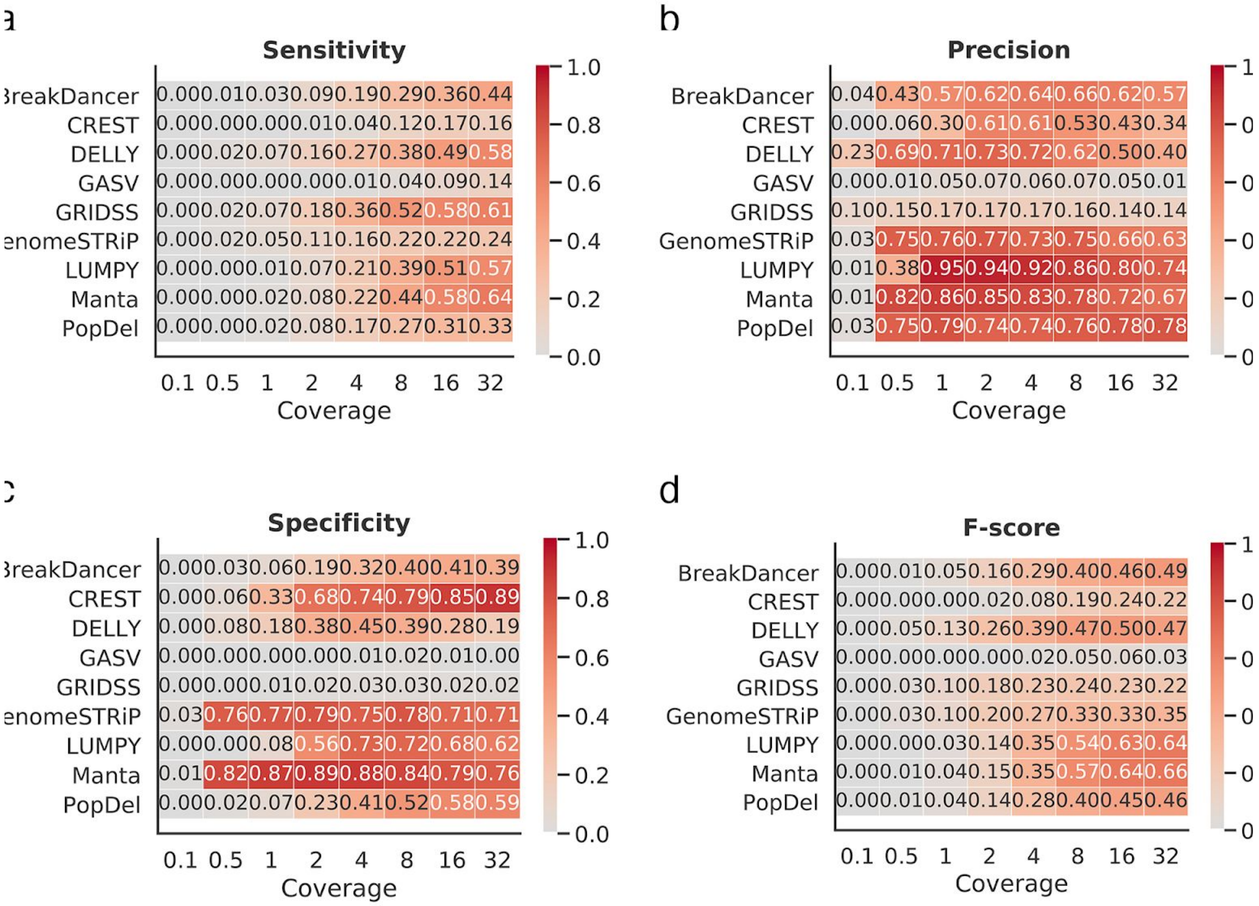


Figure 4. Performance of SV-detection tools on low and ultra-low coverage data. (a) Heatmap depicting the sensitivity based on 100 bp threshold across various levels of coverage. (b) Heatmap depicting the precision based on 100 bp threshold across various levels of coverage. (c) Heatmap depicting the specificity based on 100 bp threshold across various levels of coverage. (d) Heatmap depicting the F-score based on 100 bp threshold across various levels of coverage.

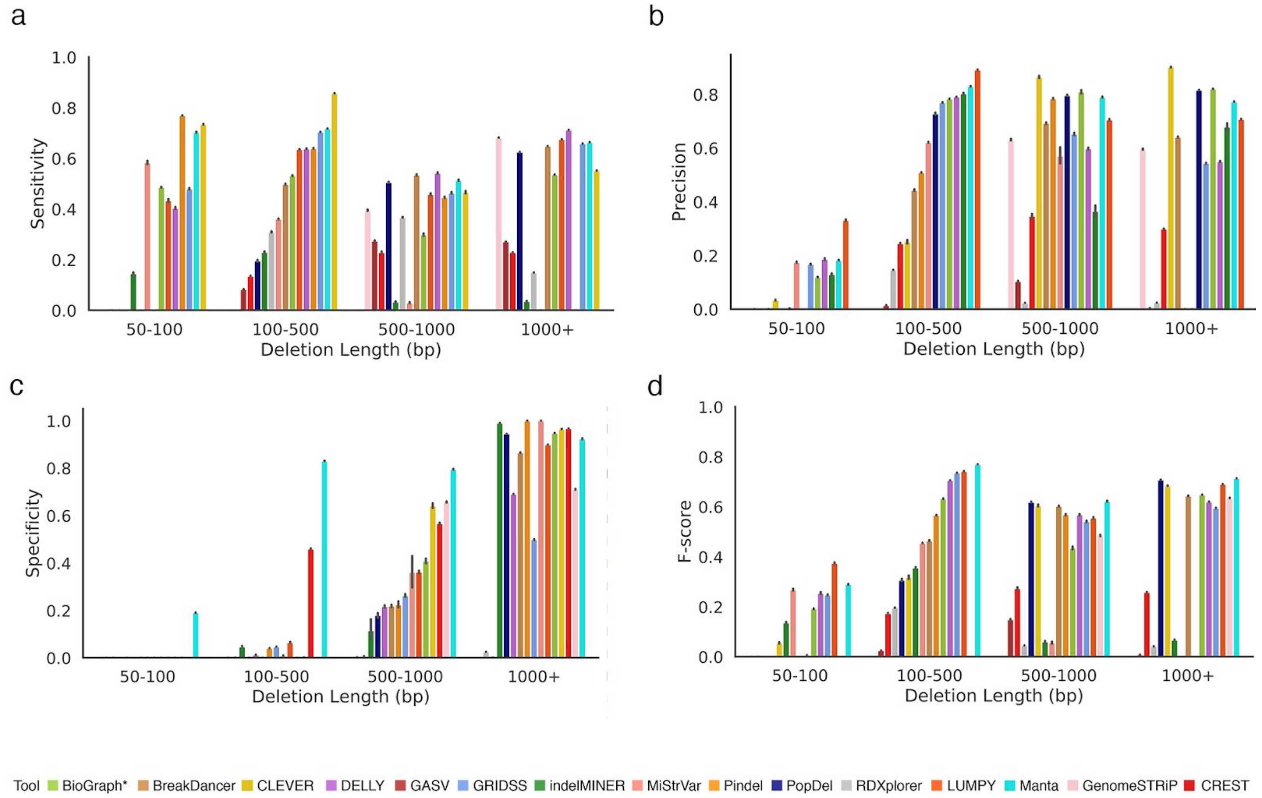


Figure 5. Comparing the performance of SV callers across various deletions lengths. (a) Sensitivity of SV callers at 100 bp thresholds across deletion length categories. (b) Precision of SV callers at 100 bp thresholds across deletion length categories. (c) Specificity of SV callers at 100 bp thresholds across deletion length categories. (d) F-score of SV callers at 100 bp thresholds across deletion length categories.

Methods

Running SV-detection tool

Commands required to run each of the tools and the installation details are available in Supplemental Table S5. The tools were run based on the recommended settings. LUMPY[94] was run using smoove, as was recommended by developers of smoove. The diploidSV vcf files were used for Manta, based on the recommendation of the developers. BioGraph was run by developers of the tool, based on the provided BAM files.

Convert the output of the SV-detection tool to a universal format

We have adopted the VCF format proposed by VCFv4.2, as the universal format used in this study. Custom format of SV-detection tools were converted to VCFv4.2. Description of custom formats is provided in Supplemental Table S6. Scripts to convert custom formats of SV-detection tools to VCFv4.2 are available at

https://github.com/Mangul-Lab-USC/benchmarking-sv-callers-paper

Compare deletion inferred from WGS data with the gold standard

We compared the deletions inferred from SV callers from WGS data (inferred deletions) with the molecular-based gold standard (true deletions). Start and end position of the deletion were considered when comparing true deletions and inferred deletions. Inferred deletion was considered correctly predicted if the distance of right and left coordinates are within the resolution threshold τ from the coordinates of true deletion. We consider the following values for resolution threshold τ: 0 bp, 10 bp, 100 bp, 1000 bp,10000 bp. Correctly predicted deletions are defined as true positives (TP). In case an inferred deletion matches several true deletions, we randomly choose one of them. Similarly, in case true deletion matches several inferred deletions we choose the first deletion that matches. randomly choose one of them. The deletion predicted by the SV caller but not present in the golden standard was defined as false positives (FP). Similarly, each deletion present in the gold standard was matched with only one deletion predicted by the software. The SV that was not predicted by the SV caller were defined as false negatives (FN). SV detection accuracy was assessed using various detection thresholds (τ). The accuracy at threshold τ is defined as the percentage of SVs with an absolute error of deletion coordinates smaller or equal to τ. To compute specificity, we have defined non-deletions calls as regions of the genome not containing deletions. We have generated the true set of non-deletions based on the gold standard. Start and end position of the non-deletion were considered when comparing true non-deletions and inferred non-deletions. Inferred non-deletion was considered correctly predicted if the distance of right and left coordinates are within the resolution threshold τ from the coordinates of true non-deletion. We consider the following values for resolution threshold τ: 0 bp, 10 bp, 100 bp, 1000 bp,10000 bp. Correctly predicted non-deletions are defined as true positives (TN).

We have used the following measured to compare the accuracy of SV- allers:

- Sensitivity=TP/(TP+FN)
- Precision=TP/(TP+FP)

- F-score=2*Sensitivity*Precision/(Sensitivity+Precision)
- Specificity= TN/(TN+FP)

The scripts to compare the deletions inferred by the SV caller versus the true deletions is available at https://github.com/Mangul-Lab-USC/benchmarking-sv-callers-paper.

Compare computational performance of SV callers

The CPU time and RAM of each tool was measured to determine it's computational performance. The statistics were measured for 1x coverage and full coverage bam files, with samples A/J and BALB/cJ for mouse data. The CPU time was computed using either the GNU time program that is inbuilt in make bash terminals or the Hoffman2 Cluster qsub command. For GNU time, we used this specific command /usr/bin/time -f "%e\t%U\t%S\t%M" which we either had to run manually on an interactive qsub session or through another method that wasn't a qsub. This GNU time command would output one line containing Wallclock time in seconds, user-time in seconds, kernel-space time in seconds, and peak memory consumption of the process in kilobytes. CPU-time was calculated by adding user-time and kernel-space time. RAM usage was equivalent to peak memory consumption in the case of this command. For qsubs on the Hoffman2 Cluster, we used the command qsub -m e which would email the user a full list of records when the tool finished running. This list included CPU-time and Max Vmem which was designated as RAM usage for each tool.

Downsampling the WGS samples

We have used custom script to downsample the full coverage BAM file to desired coverage. Existing tools (e.g., samtools) are not suitable for this purpose as they treat each read form a read pair independently, resulting in singletons reads in the downsample BAM file.

Supplementary Tables

ID	Tool name	Version	Year	Webpage of the tool	Github issue/Email	Log file	Bioconda

1.	1-2-3SV[N/A]	N/A	2012	https://github.com/Vityay/1-2-3-SV/	https://github.com/Vityay/1-2-3-SV/issues/1	Log	no
2.	inGAP-sv[41]	3.1.1	2011	http://ingap.sourceforge.net/	https://sourceforge.net/p/ingap/discussion/966200/thread/599197bd71/	Log	no
3.	CNVer[42]	0.8.1	2010	http://compbio.cs.toronto.edu/cnver	brudno@cs.toronto.edu	Log	no
4.	PRISM[43]	1.1.6	2012	http://compbio.cs.toront	yue.jiang.hit@gmail.com	Log	no

				o.edu/prism/			
5.	Splitread[44]	0.1	2011	http://splitread.sourceforge.net/	https://sourceforge.net/p/splitread/discussion/	Log	no
6.	NovelSeq[45]	1.0.2	2010	http://novelseq.sourceforge.net/Home#hn_Resources	cenk@cs.sfu.ca	Log	no
7.	Meerkat[46]	0.189	2013	http://compbio.m	ylixing@gmail.com	Log	no

				ed.harvard.edu/Meerkat/			
8.	MoDIL[47]	1.1	2009	http://compbio.cs.toronto.edu/modil/	brudno@cs.toronto.edu	Log	no
9.	PEMer[48]	N/A	2009	http://sv.gersteinlab.org/pemer/	korbel@embl.de	Log	no
10.	SV-Bay[49]	N/A	2016	https://github.com/Instit	N/A	Log	no

				utCurie/SV-Bay			
11.	iSVP[50]	N/A	2013	http://nagasakilab.csml.org/en/isvp	N/A	Log	no
12.	piCALL[51]	.01	2011	https://sites.google.com/site/vibansal/software/picall	N/A	Log	no
13.	Dindel[52]	1.01	2011	https://github.com	N/A	Log	no

				/genome/dindel-tgi			
14.	SoftSV[53]	1.4.2	2016	https://sourceforge.net/projects/softsv/	https://sourceforge.net/p/softsv/discussion/	Log	no
15.	SeqSeq[54]	N/A	2012	N/A	sd_bao@sina.com	Log	no
16.	ClipCrop[55]	N/A	2011	https://github.com/shinout/clipcrop	N/A	Log	no
17.	SplazerS	2.4.0	2012	https:/	emde@inf.fu-berli	Log	no

	[56]			/github.com/seqan/seqan/tree/master/apps/splazers	n.de		
18.	TIGRA[57]	0.4.0	2014	http://odin.mdacc.tmc.edu/~kchen3/tigra/tigra-0.4.0.tar.gz	kchen3@mdanderson.org	Log	no
19.	Read-Depth	N/A	2010	N/A	rd@sanger.ac.uk	Log	no

	(CND)[58]						
20.	JointSLM[59]	N/A	2011	https://omictools.com/jointslm-tool	albertomagi@gmail.com	Log	no
21.	CNAseg[60]	1.0	2010	Broken Link	Sergii.Ivakhno@cancer.org.uk	Log	no
22.	GASVPro[61]	0.9	2012	https://storage.googleapis.com/google-code-archive-downlo	Suzanne_Sindi@Brown.edu	Log	no

				ads/v2/code.google.com/GASV/GASVProVersion0.9.tar.gz			
23.	SVMerge[62]	1.2r37	2010	https://sourceforge.net/projects/svmerge/files/latest/download?sourc	https://sourceforge.net/p/svmerge/discussion/1199306/thread/1e781e3143/	Log	no

				e=files			
24.	BIC-seq 2[63]	0.2.4 0.7.2	2016	http://compbio.med.harvard.edu/BIC-seq/NBICseq-norm_v0.2.4.tar.gz/ http://compbio.med.harvard.edu/BIC-seq/	peter_park@harvard.edu	Log	no

				NBIC seq-seg_v0.7.2.tar.gz			
25.	SVM2[64]	1.2	2012	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3467043/	N/A	Log	no
26.	Bellerophon[65]	N/A	2013	http://cbc.case.edu/Bellerophon/	jingli@case.edu	Log	no

27.	SMUFIN[66]	0.9.4	2014	http://cg.bsc.es/smufin/	http://cg.bsc.es/smufin/	Log	no
28.	falcon[N/A]	2018.31.08-03.06-py2.7-ucs4-beta	2015	https://github.com/PacificBiosciences/FALCON	N/A	Log	yes
29.	modSaRa New version[67]	1.0	2017	https://publichealth.yale.edu/c2s2/software/modSaRa/	heping.zhang@yale.edu	Log	no

30.	readDepth[68]	0.9.8.4	2015	https://github.com/chrisamiller/readDepth	N/A	Log	
31.	freec[21]	11.5	2016	https://github.com/BoevaLab/FREEC	N/A	Log	
32.	ioncopy[69]	2.1.1	2018	https://cran.r-project.org/web/packages/ion	https://github.com/cran/ioncopy/issues/1	Log	no

bioRxiv preprint doi: https://doi.org/10.1101/2020.04.16.045120; this version posted August 4, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder. All rights reserved. No reuse allowed without permission.

				copy/index.html			
33.	SVmine[70]	2.0	2017	https://github.com/xyc0813/SVmine	https://github.com/xyc0813/SVmine/issues/2	Log	no
34.	CNVnator[72]	0.4.1	2014	https://github.com/abyzovlab/CNVnator	N/A	Log	No

35.	Hydra-Multi[73]	0.5.2	2015	https://github.com/arq5x/Hydra	N/A	Log	No
36.	cnmops[74]	1.26.0	2012	http://www.bioinf.jku.at/software/cnmops/	N/A	Log	Yes

37.	breakseek[75]	1.2	2015	https://sourceforge.net/projects/breakseek/	https://sourceforge.net/p/breakseek/discussion/general/thread/4cc99d0a/	Log	no
38.	fermikit[76]	0.13	2015	https://github.com/lh3/fermikit/	https://github.com/lh3/fermikit/issues/20	Log	yes

39.	breakseq[77]	1.3	2009	http://sv.gersteinlab.org/breakseq/	N/A	Log	No
40.	tardis[78]	1.0.19	2018	https://github.com/BilkentCompGen/tardis	N/A	Log	Yes

Table S1. Existing methods to detect structural variants which failed to be installed. If github was available, we have opened the issue. Otherwise, email was sent to the corresponding authors of the paper.

Strain name	**Total number**	**Average**	**Minimum**	**Maximum**

	of gold standard deletions	**deletion length**	**deletion length**	**deletion length**
A/J	533	1968	15	239572
AKR/J	504	1400	13	24301
BALB/cJ	545	1947	15	239572
C3H/HeJ	539	1993	15	239572
DBA/2J	609	1301	16	24301
LP/J	483	1534	13	24301
CBA/J	586	1330	24	24301

Table S2. Gold standard deletion calls from chromosome 19 from 7 inbred mouse strains.

Strain name	**Total number PE reads**	**Number of reads mapped**	**Number of reads mapped to**	**Number of discordant**	**Number of multi-map**	**Average coverage (chr19)**

		reads	**chr 19**	**reads (chr19)**	**ped reads (chr19)**	
A/J	691798369	15097330	14903814	97469	14765316	48.52
AKR/J	758992189	16205276	15814814	66938	16205276	51.49
BALB/cJ	828502183	17292403	17093309	106915	17292403	55.65
C3H/HeJ	828898555	17293104	17093572	77254	17293104	55.65
CBA/J	736613128	15517147	15343318	67076	15517147	49.95
DBA/2J	738506916	14860907	14690187	72075	14860907	47.83
LP/J	703108275	14331024	14176319	60233	14331024	46.15
Total	5286419615	110597191	109,115,333	547960	110265177	50.75

Table S3. Information about WGS data obtained for seven inbred mouse strains.

S.No	**Tool name**	**Version**	**Year**	**Webpage**	**Reason the tools was excluded**
1.	GATK[90]	4.1.6.0	2010	https://github.com/broadinstitute/gatk	designed to detect only small SVs (less than 50 bp in length)
2.	rSW-seq[88]	N/A	2010	https://bioinformaticshome.com/tools/cnv/descriptions/rSW-seq.html	designed to detect SVs in tumor-normal samples
3.	Variation Hunter[83]	0.04	2010	http://variationhunter.sourceforge.net/	replaced with TARDIS
4.	inGAP-SV[41]	3.1.1	2011	http://ingap.sourceforge.net/	utilizes a GUI, hence command

					line tasks with many datasets would not be viable for benchmarking
5.	COPS[87]	N/A	2012	ftp://115.119.160.213 username: cops password:cops	designed to detect SVs in tumor-normal samples
6.	Magnolya[27]	0.15	2012	https://sourceforge.net/projects/magnolya/files/	unable to process aligned WGS data
7.	Varscan[92]	2.3.9	2012	https://sourceforge.net/projects/varscan/files/	designed to detect only small SVs

					(less than 50 bp in length)
8.	Patchwork[86]	2.4	2013	http://patchwork.r-forge.r-project.org/	designed to detect SVs in tumor-normal samples
9.	PBHoney-NGM[82]	13.10	2014	https://sourceforge.net/projects/pb-jelly/	uses long-read sequencing data
10.	Platypus[91]	0.8.1	2014	https://github.com/andyrimmer/Platypus	designed to detect only small SVs (less than 50 bp in length)
11.	seqCBS[89]	1.2.1	2014	https://cran.r-project.org/web/packages/seqCBS/index.	designed to detect SVs in tumor-normal samples

				html	
12.	Socrates[85]	1.1	2014	http://bioinf.wehi.edu.au/socrates/ https://github.com/jibsch/socrates	replaced with GRIDSS, as the toolmaker who developed both suggested GRIDSS for SV detection
13.	Hydra[84]	N/A	2015	http://web.engr.illinois.edu/~mkim158/HyDRA.zip	replaced with LUMPY
14.	Bic-seq [63]	0.2.4	2016	http://compbio.med.harvard.edu/BIC-seq/	designed to detect SVs in tumor-normal samples

15.	Control-FREEC[21]	11.5	2016	https://github.com/BoevaLab/FREEC	It only accepts only GZ format and no longer supports bam files (starting with from version 8.0)
16.	NanoSV[81]	1.2.4	2017	https://github.com/mroosmalen/nanosv	uses long-read sequencing data
17.	Sniffles[96]	1.0.8	2018	https://github.com/fritzsedlazeck/Sniffles	Designed for long reads

Table S4. Tools excluded from analysis.

S.No	**Tool Name**	**Command**
1.	BreakDancer[32]	bam2cfg.pl $input | BreakDancer-max > ${outdir}/${filename}.vcf
2.	CLEVER[30]	./CLEVER --sorted --use_xa data.bam ref.fa output
3.	DELLY[31]	DELLY call -g example/reference.fa -t DEL example/alignments.bam -o DELLY.output//alignments.bam.bcf
4.	GASV[38]	java -Xms512m -Xmx2048m -jarGASV/bin/BAMToGASV.jar data.bam
5.	indelMINER[29]	$toolname $reference sample=$alignments > ${outdir}/${toolName}_${filename}.vcf
6.	MiStrVar[40]	python3 ${toolname} \ -p ${filename} \ -r /u/scratch/r/ramayyal/reference_genome/full_bams/mouseBAM/chr19.fa \

<table>
<tr><td></td><td></td><td>--files alignment=${input}
awk 'NR<12 {print$0}' $filename/${filename}.vcf| cat >> $filename/${toolName}.${filename}.modified.vcf

grep "DEL" $filename/${filename}.vcf | awk 'NR>11 {split($8,a,"="); split($8,b,";");print "19"" "$2" ""."" ""."" ""<DEL>"" ""."" "$7" "b[1]";""SVLEN="a[3]-$2";"b[2]}' | cat >> $filename/${toolName}.${filename}.modified.vcf</td></tr>
<tr><td>7.</td><td>Pindel[39]</td><td>samtools view ../example/A_J.chr19.2.5p_sorted.bam | sam2Pindel - A_J.chr19.2.5p_sorted.bam.Pindeloutput.txt 300 tomour 0 >> Pindel.output/report_Pindel.A_J.chr19.2.5p_sorted.bam.log 2>&1</td></tr>
<tr><td>8.</td><td>Smoove</td><td>smoove call -x --outdir $out --name $batchName --fasta $reference.fa --p $threads --genotype $sample1.bam [$sample2.bam ...]</td></tr>
<tr><td>9.</td><td>GRIDSS[39]</td><td>GRIDSS.sh -j GRIDSS.jar -a $tmp/assembly -w $tmp/workdir --jvmheap 8g -r $reference.fa -t 1 -o $out/example.vcf.gz $sample1.bam [$sample2-bam ...] ;</td></tr>
</table>

		Rscript --vanilla annotate.R $out/example.vcf.gz >(gzip -9 -c > out/example.annotated.vcf.gz)
10.	PopDel[93]	popdel profile -o $out/sampleX.profile -mrg -n 1000 -i $samplingRegions.bed sampleX.bam 19; popdel call $sample_profile_location -o $out/example.vcf -r 19
11.	RDXplorer[95]	python ${toolname} ${path2bam} ${reference} ${wrkgdir} ${chromOfInterest} ${gender} ${hg} ${winSize} ${baseCopy} ${filter} ${sumWithZero} ${debug} ${delete} awk '{ if ($3 =="1") print $8" "$9" ""."" ""."" ""<DEL>"" ""."" ""PASS"" ""SVTYPE=DEL;SVLEN="$10-$9";END="$10}' chr19.sum
12.	Biograph*	N/A
13.	CREST[79]	N/A
14.	Manta[80]	python configManta.py --bam $sample1.bam [...] --bam $sample7.bam --referenceFasta $reference.fa --runDir $tmp/workdir --region 19; python runWorkflow.py -m local -j 1

15.	GenomeStrip[71]	N/A

Table S5. Commands used to run the tools and produce an output vcf file with predicted deletions using a reference (FASTA format) and aligned reads (BAM format) file as input arguments.

S.No	Tool Name	Output type	Description of the format
1.	BreakDancer[32]	Custom	Start, end, type, length
2.	CLEVER[30]	Custom	VCFv4.1
3.	DELLY[31]	Custom	VCFv4.2
4.	GASV[38]	Custom	Start, end, type
5.	indelMINER[29]	VCF	VCFv4.1
6.	LUMPY[94]	VCF	VCFv4.2
7.	MiStrVar[40]	VCF	VCFv4.2
8.	Pindel[39]	Custom	Start,end
9.	BioGraph*	VCF	VCFv4.2
10.	GRIDSS[39]	VCF	VCFv4.2
11.	Popdel[93]	VCF	VCFv4.3

12.	RDXplorer[95]	Custom	Start, end, length
13.	CREST[79]	Custom	Start, end, type
14.	Manta[80]	VCF	VCFv4.1
15.	genomeSTRIP[71]	VCF	VCFv4.2

Table S6. Output format of different tools

Supplementary Figures

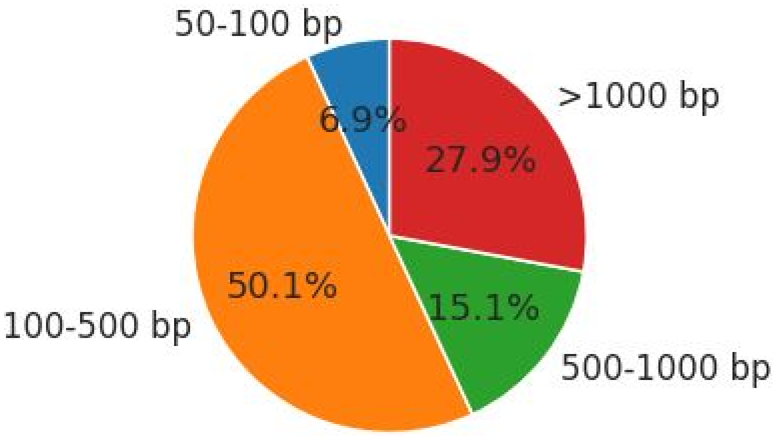
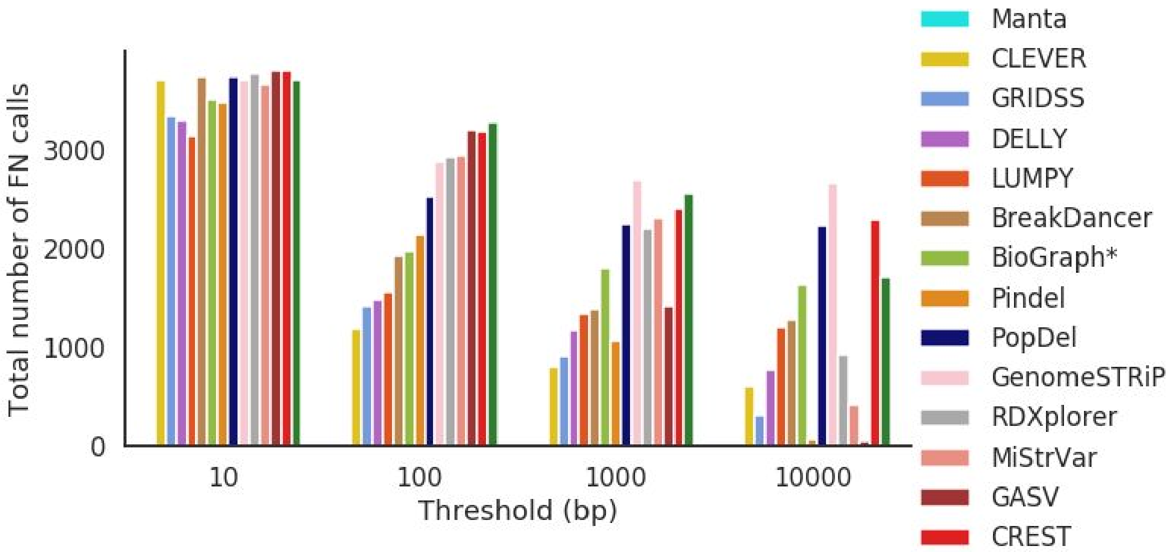


Figure S1 Total number of deletions across seven mouse strains split across categories defined based on deletion length

a)



b)

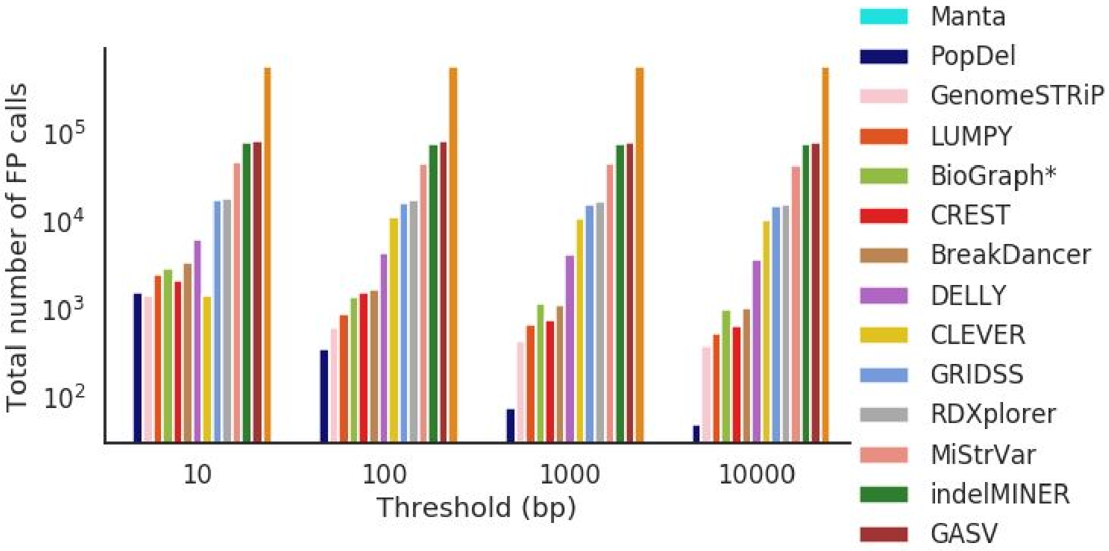
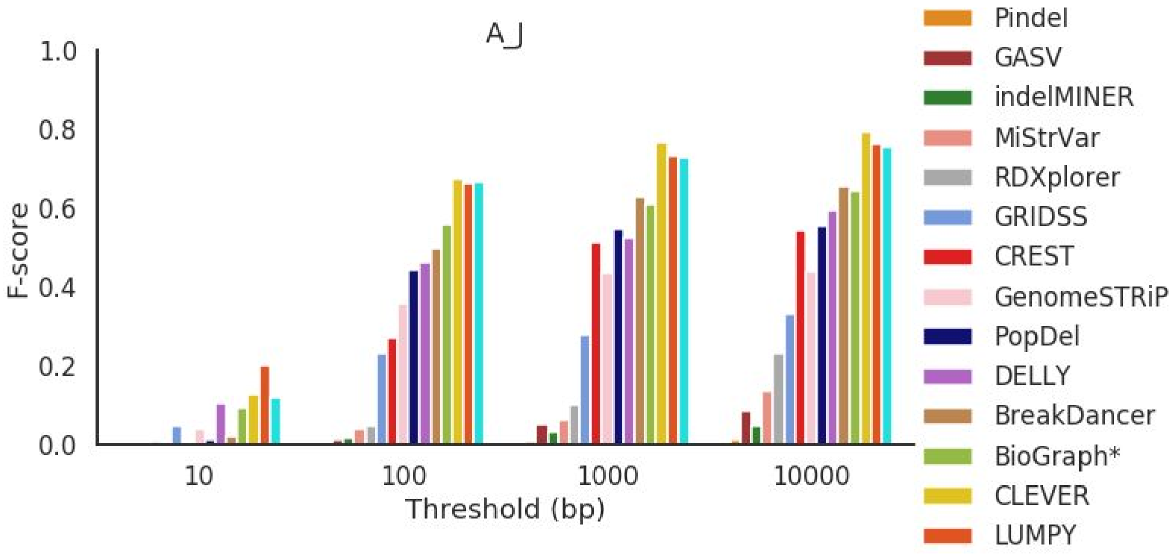
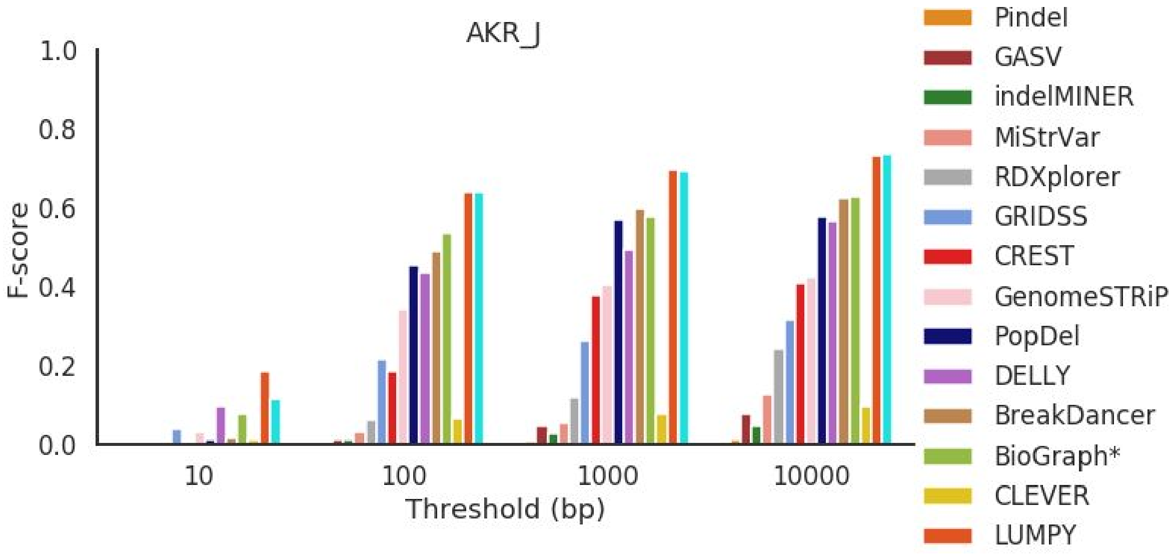
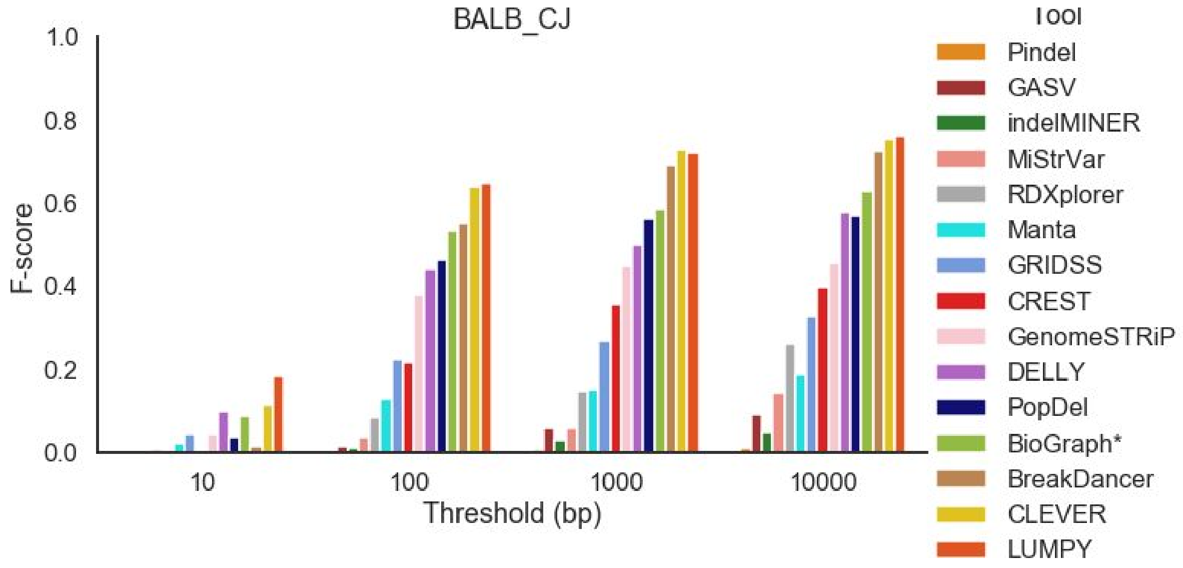
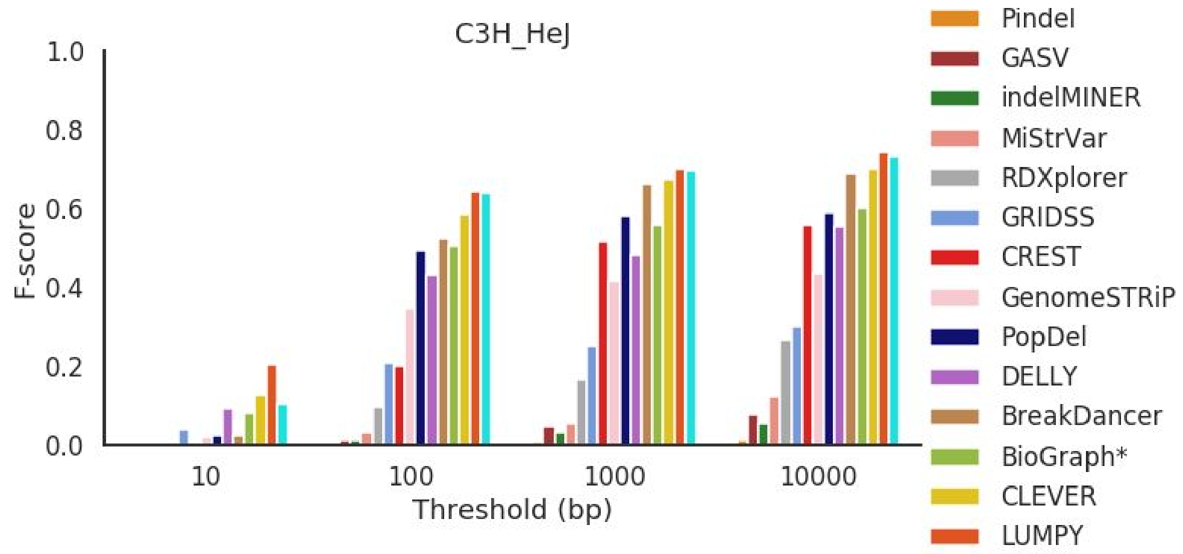


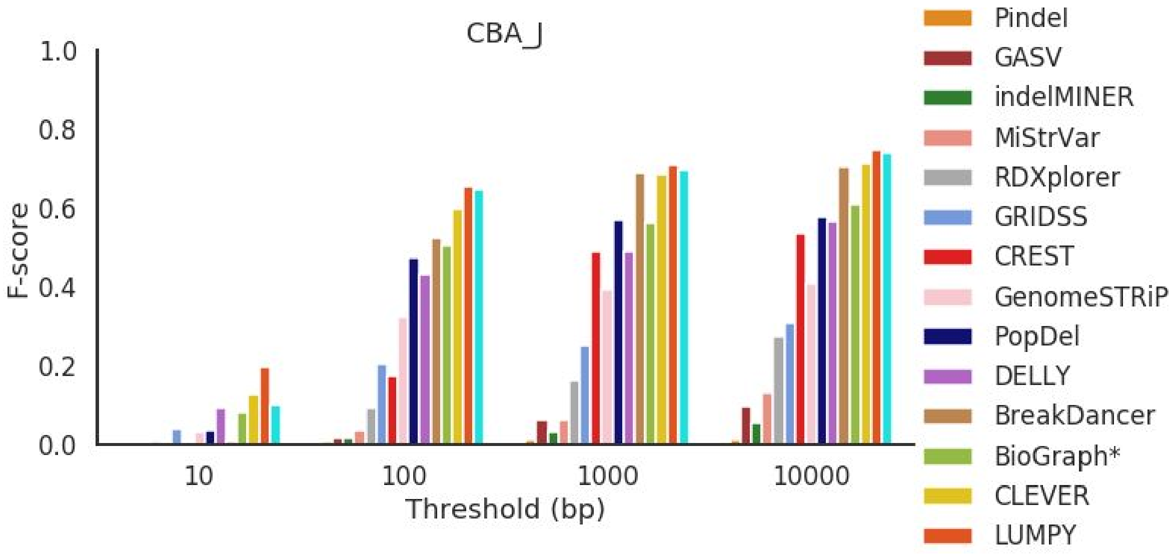
Figure S2. (a) Total number of FN calls for different error thresholds across 7 different mouse strains (b) Total number of FP calls for different error thresholds across 7 different mouse strains


A_J
1.0
0.8
0.6
0.4
0.2
0.0
F-score
10
100
1000
10000
Threshold (bp)
Pindel
GASV
indelMINER
MiStrVar
RDXplorer
GRIDSS
CREST
GenomeSTRiP
PopDel
DELLY
BreakDancer
BioGraph*
CLEVER
LUMPY


AKR_J
1.0
0.8
0.6
0.4
0.2
0.0
F-score
10
100
1000
10000
Threshold (bp)
Pindel
GASV
indelMINER
MiStrVar
RDXplorer
GRIDSS
CREST
GenomeSTRiP
PopDel
DELLY
BreakDancer
BioGraph*
CLEVER
LUMPY


BALB_CJ
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
Tool
Pindel
GASV
indelMINER
MiStrVar
RDXplorer
Manta
GRIDSS
CREST
GenomeSTRiP
DELLY
PopDel
BioGraph*
BreakDancer
CLEVER
LUMPY


C3H_HeJ
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
Pindel
GASV
indelMINER
MiStrVar
RDXplorer
GRIDSS
CREST
GenomeSTRiP
PopDel
DELLY
BreakDancer
BioGraph*
CLEVER
LUMPY


CBA_J
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
Pindel
GASV
indelMINER
MiStrVar
RDXplorer
GRIDSS
CREST
GenomeSTRiP
PopDel
DELLY
BreakDancer
BioGraph*
CLEVER
LUMPY

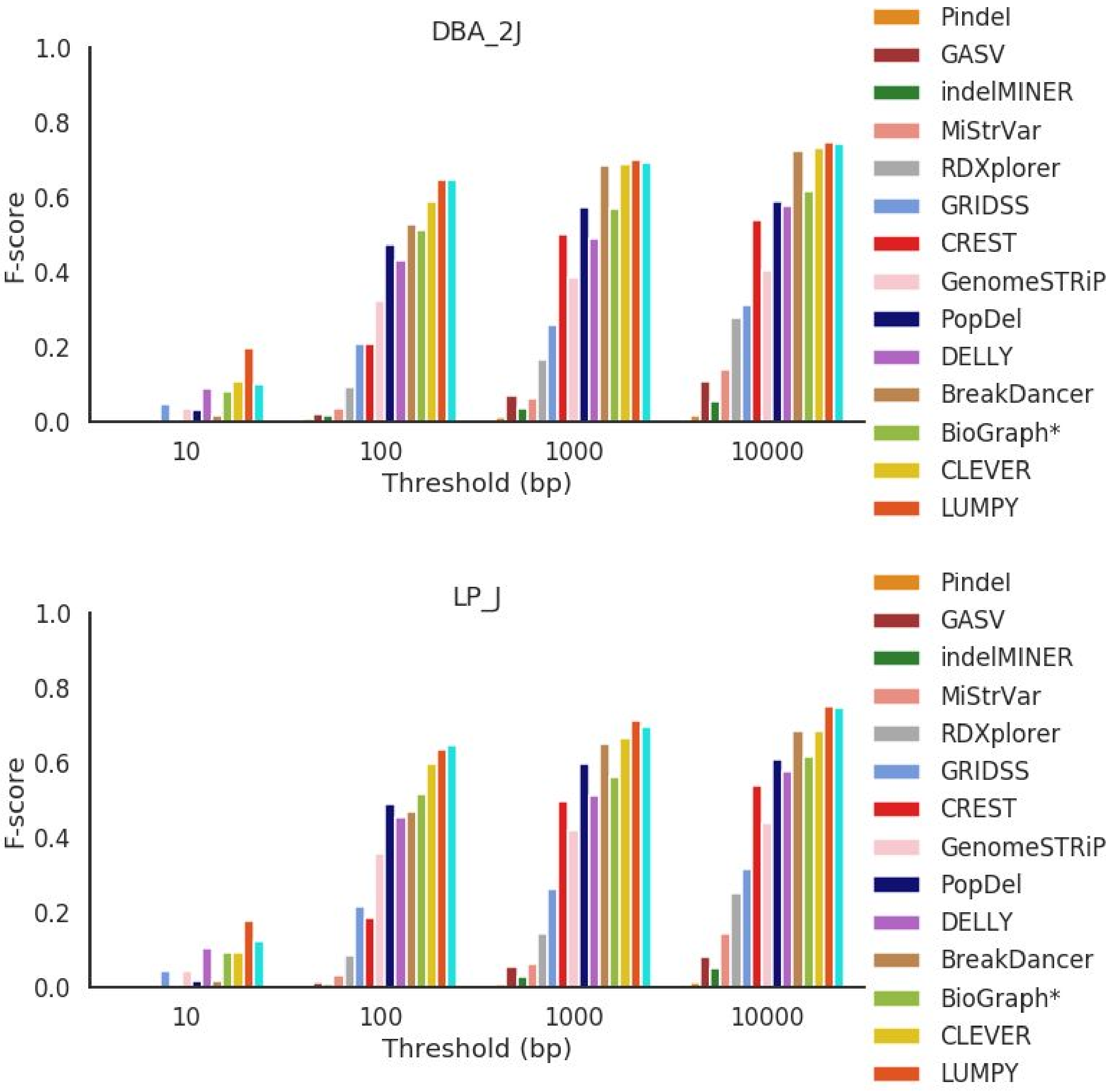
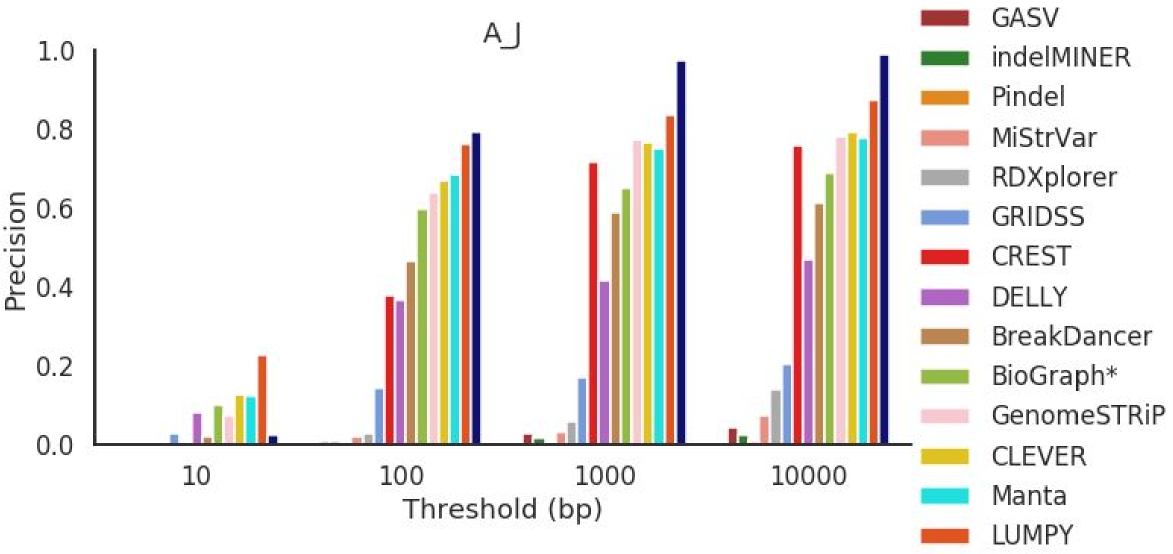
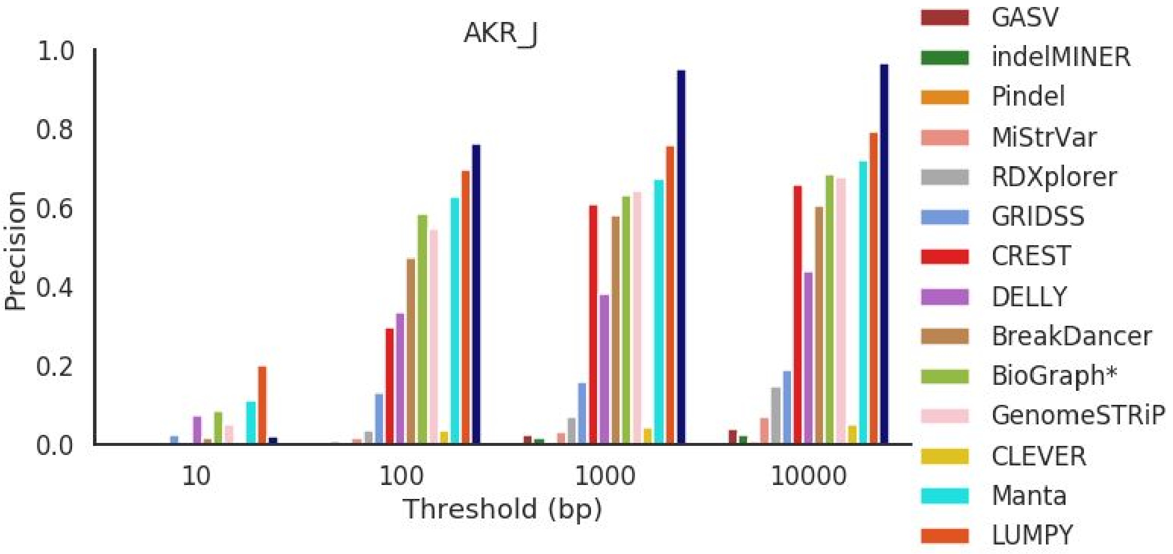
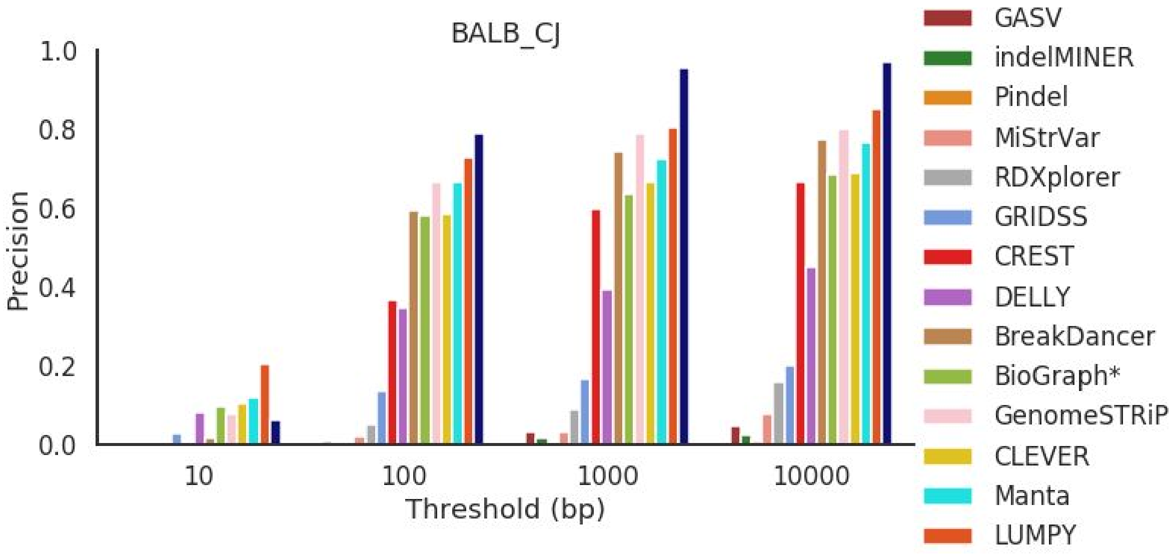
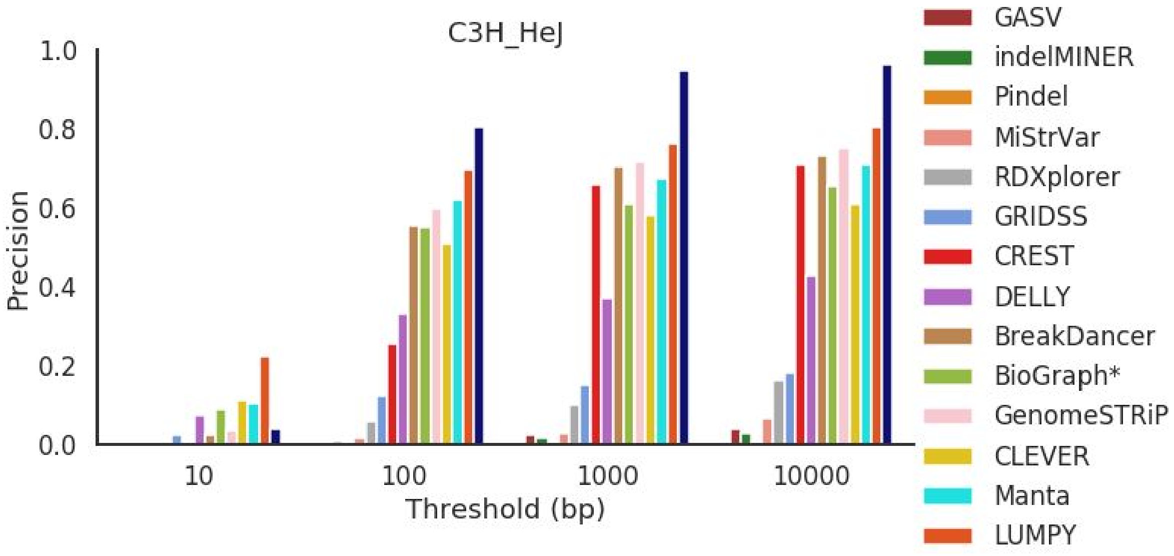

DBA_2J
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
Pindel
GASV
indelMINER
MiStrVar
RDXplorer
GRIDSS
CREST
GenomeSTRiP
PopDel
DELLY
BreakDancer
BioGraph*
CLEVER
LUMPY
LP_J
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
Pindel
GASV
indelMINER
MiStrVar
RDXplorer
GRIDSS
CREST
GenomeSTRiP
PopDel
DELLY
BreakDancer
BioGraph*
CLEVER
LUMPY

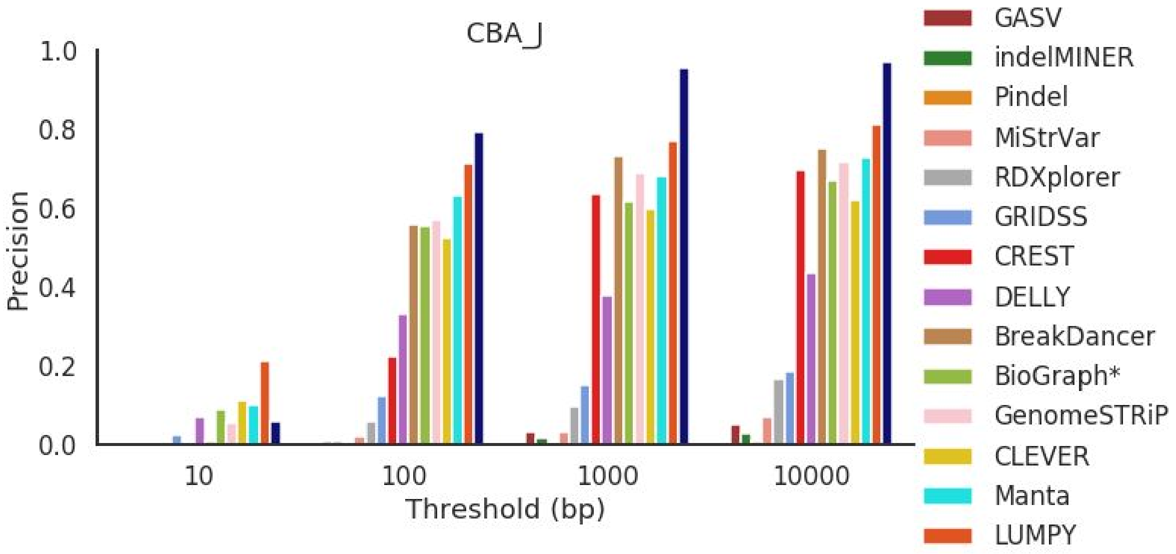
Figure S3. F-score presented for each mouse strain across all error thresholds.

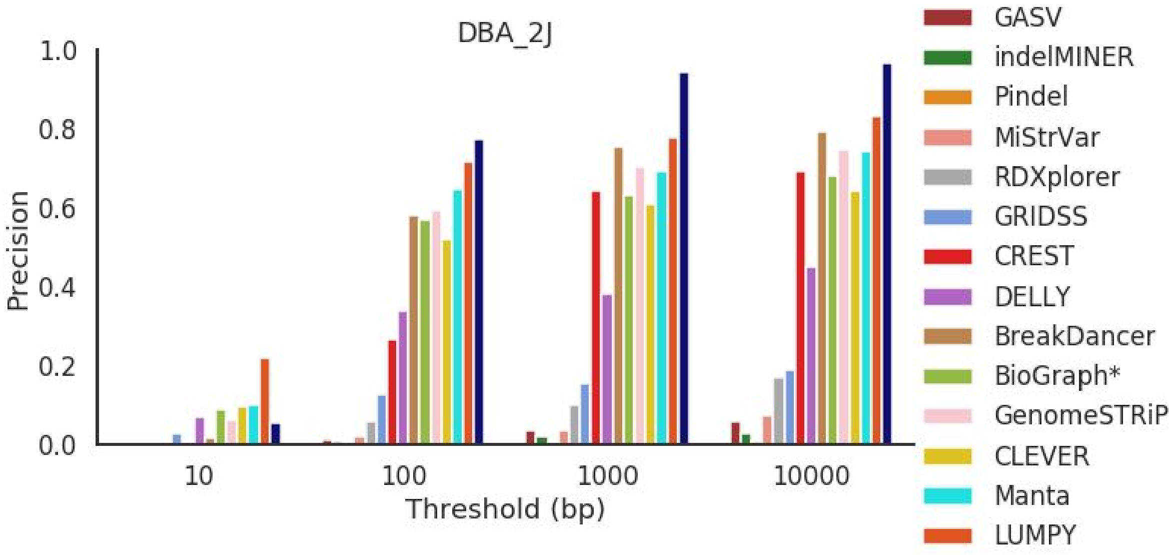

A_J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GASV
indelMINER
Pindel
MiStrVar
RDXplorer
GRIDSS
CREST
DELLY
BreakDancer
BioGraph*
GenomeSTRiP
CLEVER
Manta
LUMPY


AKR_J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GASV
indelMINER
Pindel
MiStrVar
RDXplorer
GRIDSS
CREST
DELLY
BreakDancer
BioGraph*
GenomeSTRiP
CLEVER
Manta
LUMPY


BALB_CJ
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GASV
indelMINER
Pindel
MiStrVar
RDXplorer
GRIDSS
CREST
DELLY
BreakDancer
BioGraph*
GenomeSTRiP
CLEVER
Manta
LUMPY


C3H_HeJ
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GASV
indelMINER
Pindel
MiStrVar
RDXplorer
GRIDSS
CREST
DELLY
BreakDancer
BioGraph*
GenomeSTRiP
CLEVER
Manta
LUMPY


CBA_J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GASV
indelMINER
Pindel
MiStrVar
RDXplorer
GRIDSS
CREST
DELLY
BreakDancer
BioGraph*
GenomeSTRiP
CLEVER
Manta
LUMPY


DBA_2J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GASV
indelMINER
Pindel
MiStrVar
RDXplorer
GRIDSS
CREST
DELLY
BreakDancer
BioGraph*
GenomeSTRiP
CLEVER
Manta
LUMPY

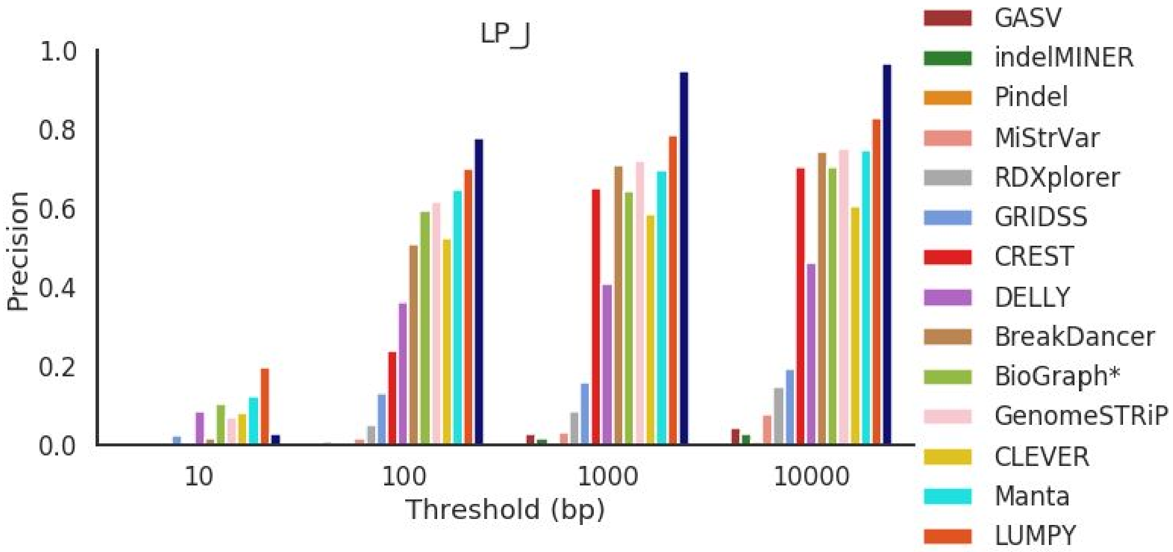
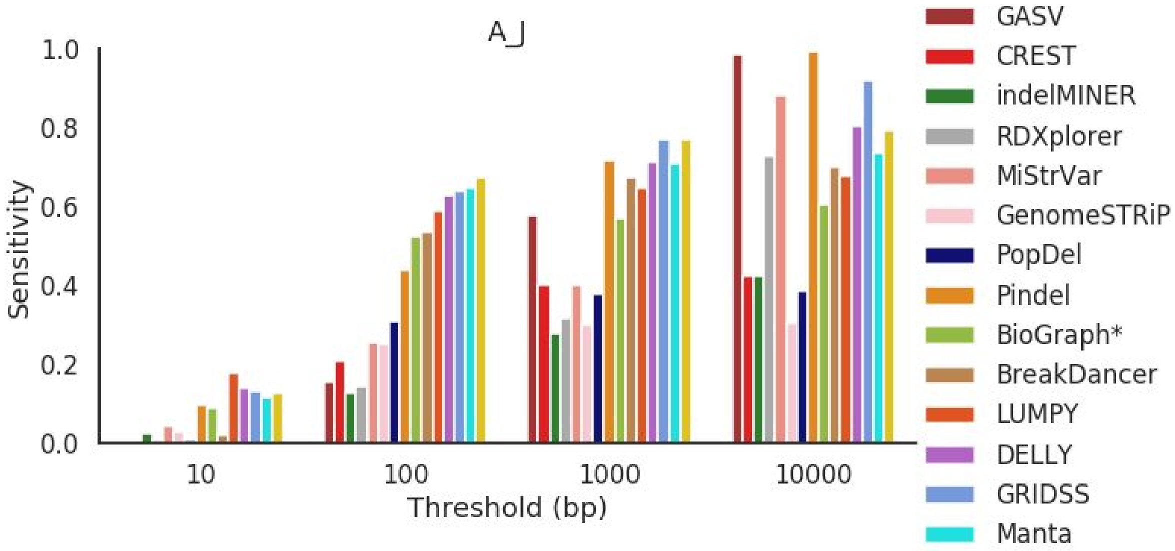
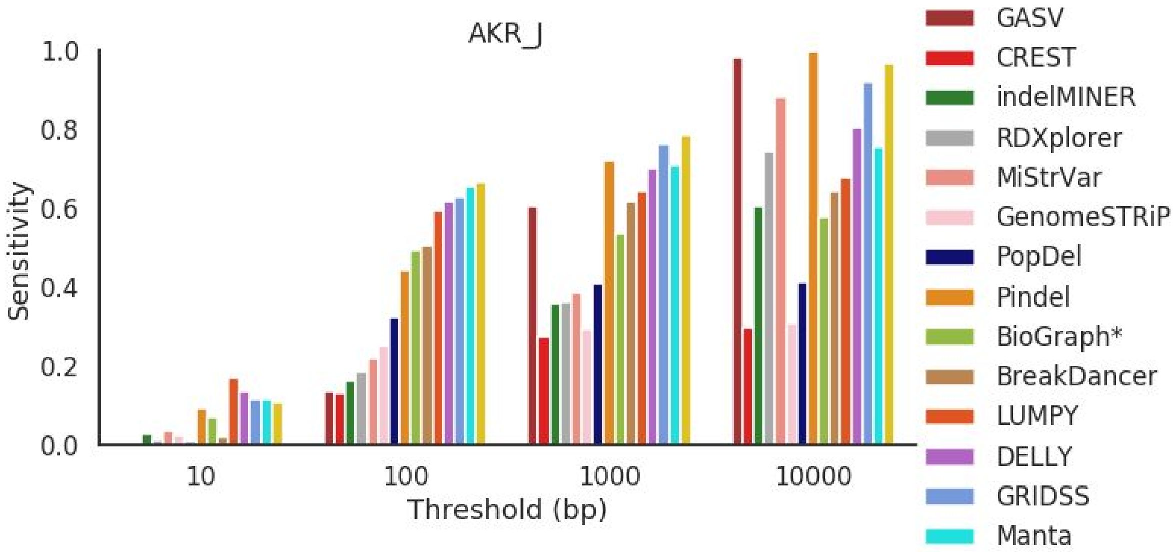
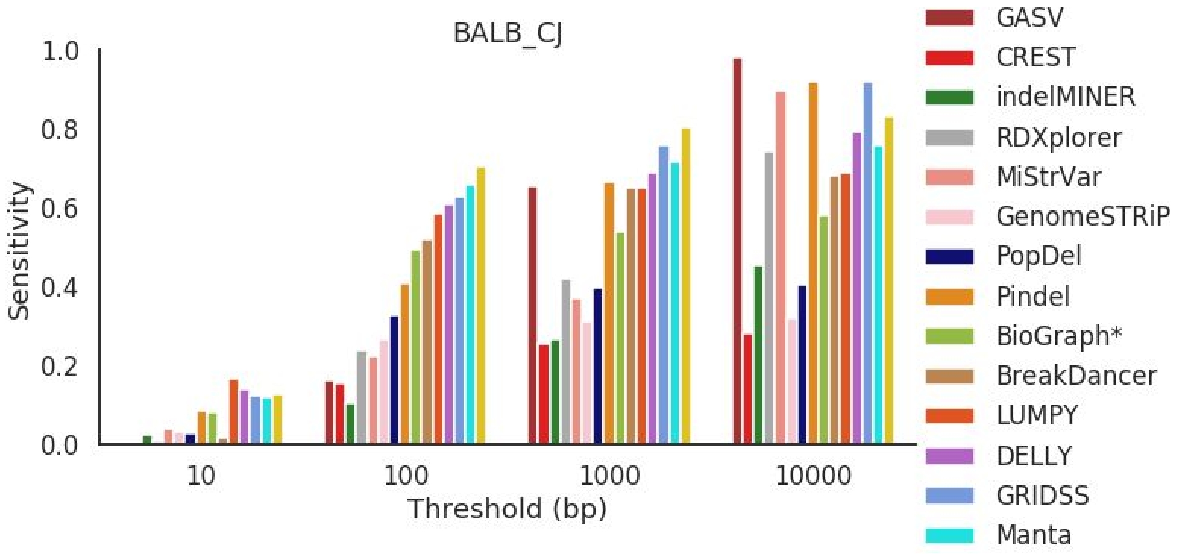
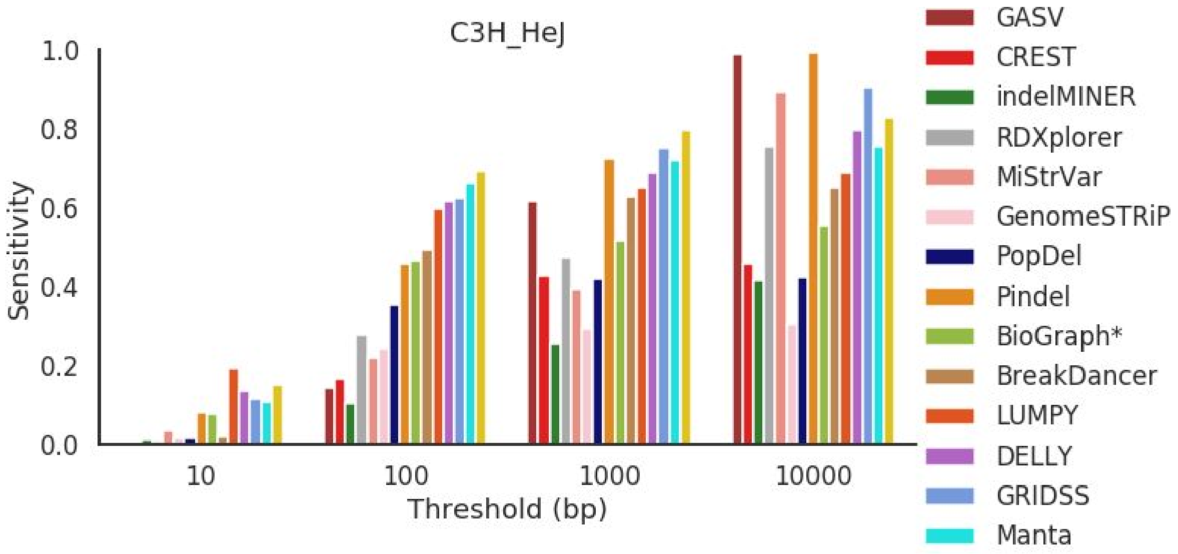


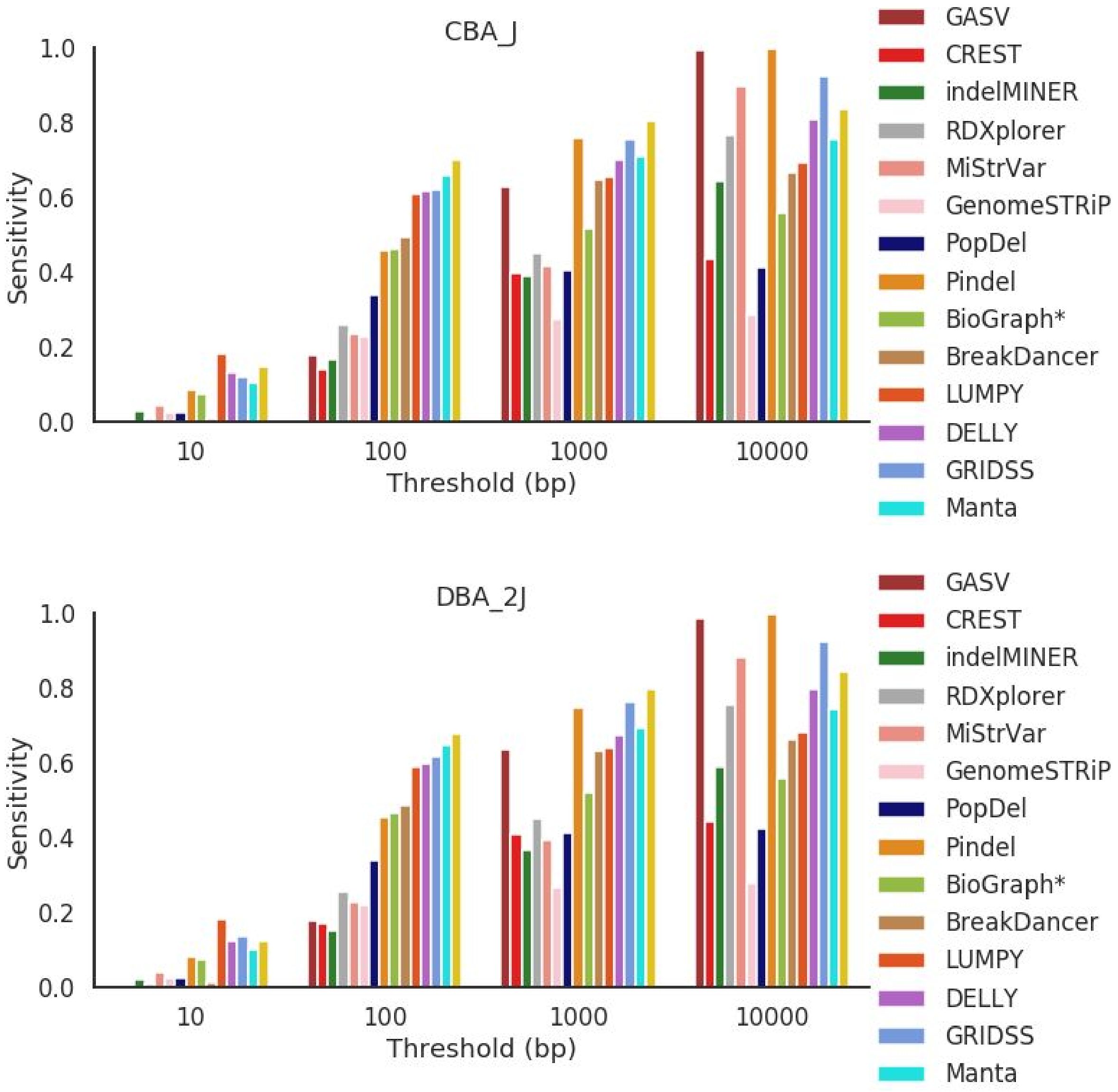
Figure S4. Precision presented for each mouse strain across all error thresholds.


A_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GASV
CREST
indelMINER
RDXplorer
MiStrVar
GenomeSTRiP
PopDel
Pindel
BioGraph*
BreakDancer
LUMPY
DELLY
GRIDSS
Manta


AKR_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GASV
CREST
indelMINER
RDXplorer
MiStrVar
GenomeSTRiP
PopDel
Pindel
BioGraph*
BreakDancer
LUMPY
DELLY
GRIDSS
Manta


BALB_CJ
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GASV
CREST
indelMINER
RDXplorer
MiStrVar
GenomeSTRiP
PopDel
Pindel
BioGraph*
BreakDancer
LUMPY
DELLY
GRIDSS
Manta


C3H_HeJ
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GASV
CREST
indelMINER
RDXplorer
MiStrVar
GenomeSTRiP
PopDel
Pindel
BioGraph*
BreakDancer
LUMPY
DELLY
GRIDSS
Manta


CBA_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GASV
CREST
indelMINER
RDXplorer
MiStrVar
GenomeSTRiP
PopDel
Pindel
BioGraph*
BreakDancer
LUMPY
DELLY
GRIDSS
Manta
DBA_2J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GASV
CREST
indelMINER
RDXplorer
MiStrVar
GenomeSTRiP
PopDel
Pindel
BioGraph*
BreakDancer
LUMPY
DELLY
GRIDSS
Manta

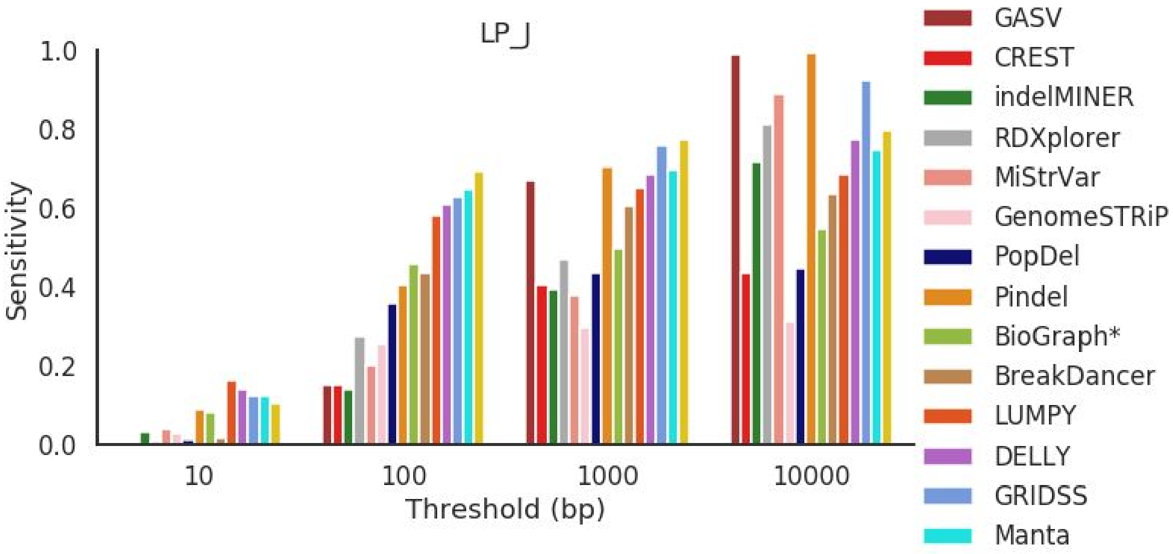
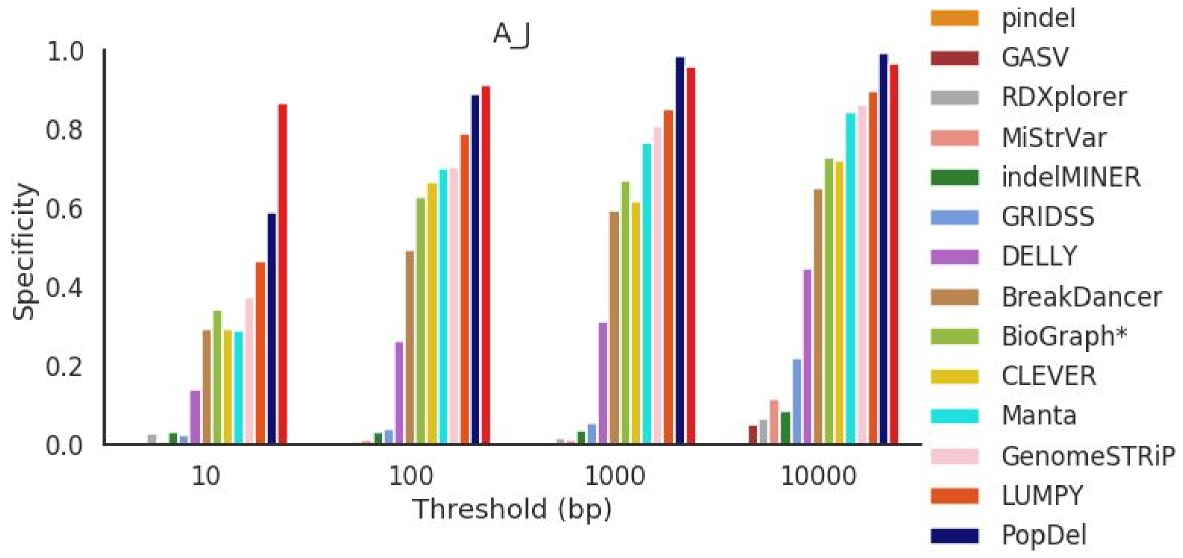
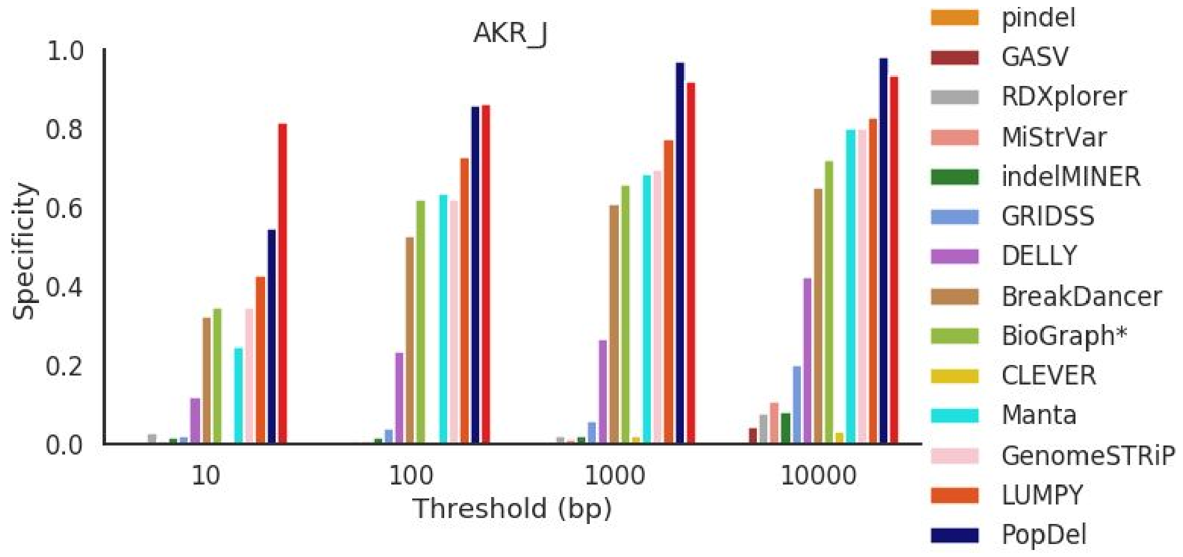
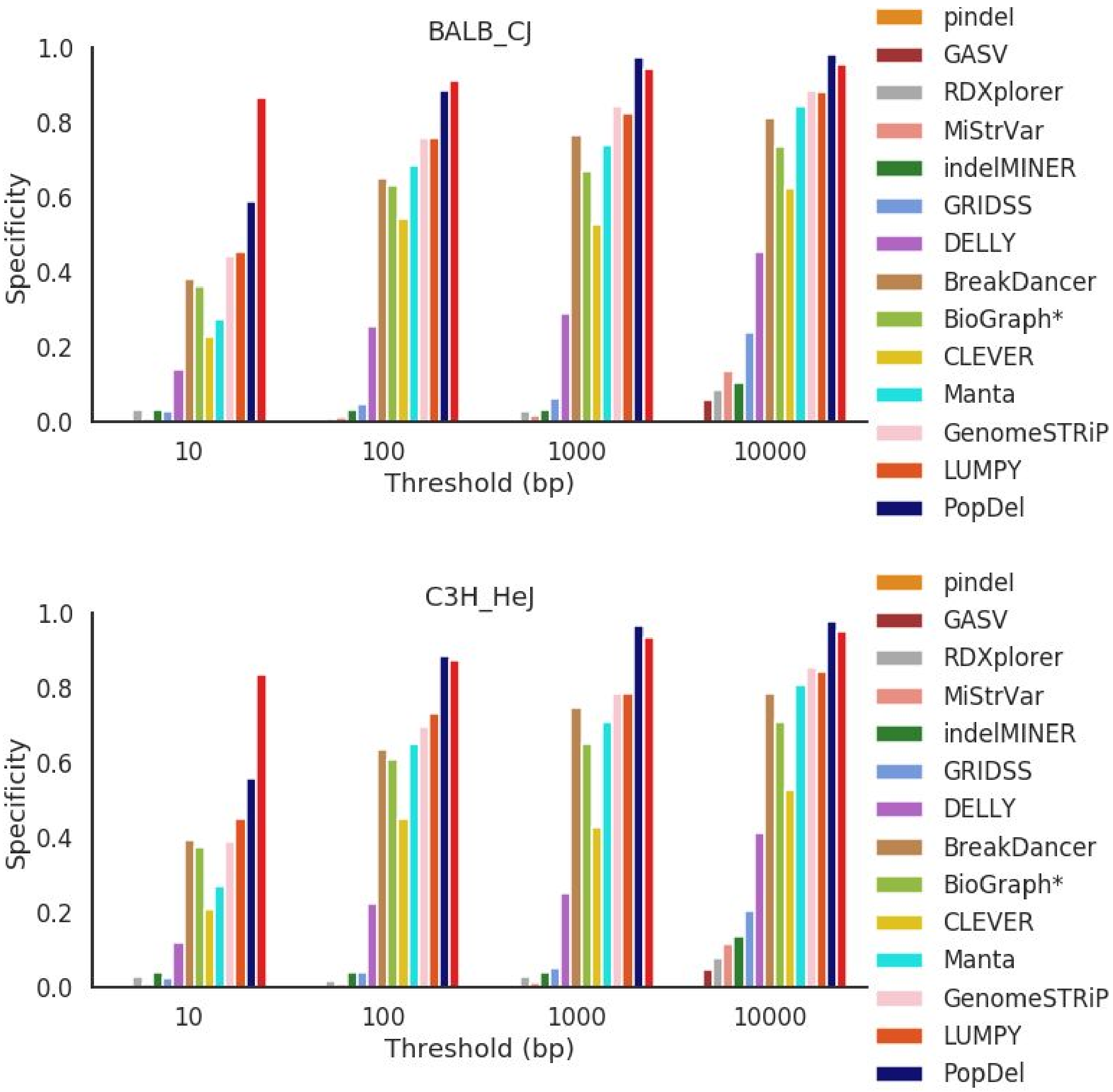
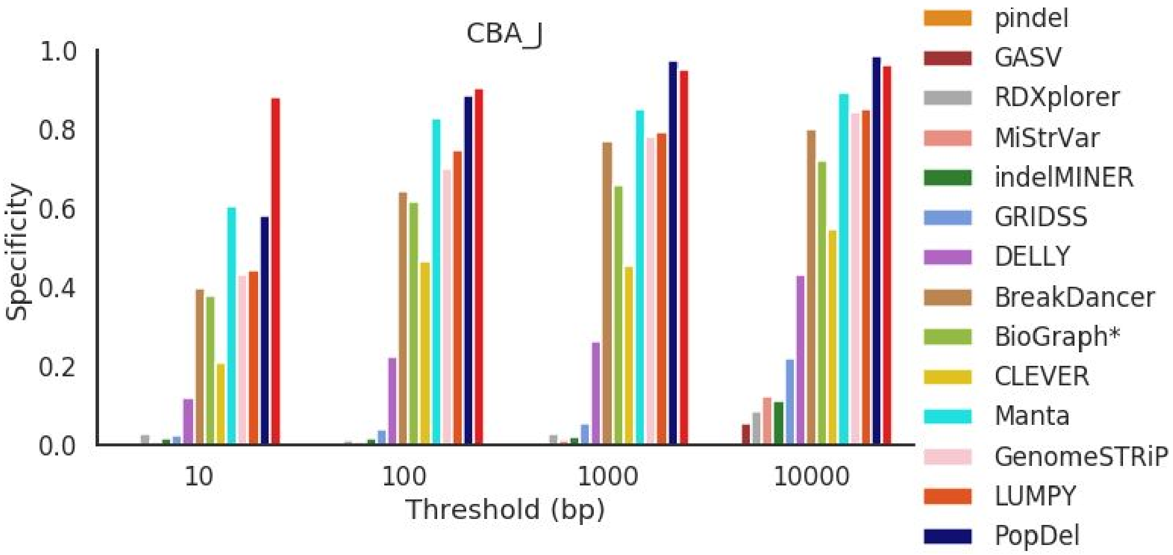


Figure S5. Sensitivity presented for each mouse strain across all error thresholds


A_J
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
pindel
GASV
RDXplorer
MiStrVar
indelMINER
GRIDSS
DELLY
BreakDancer
BioGraph*
CLEVER
Manta
GenomeSTRiP
LUMPY
PopDel


AKR_J
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
pindel
GASV
RDXplorer
MiStrVar
indelMINER
GRIDSS
DELLY
BreakDancer
BioGraph*
CLEVER
Manta
GenomeSTRiP
LUMPY
PopDel


BALB_CJ
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
pindel
GASV
RDXplorer
MiStrVar
indelMINER
GRIDSS
DELLY
BreakDancer
BioGraph*
CLEVER
Manta
GenomeSTRiP
LUMPY
PopDel
C3H_HeJ
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
pindel
GASV
RDXplorer
MiStrVar
indelMINER
GRIDSS
DELLY
BreakDancer
BioGraph*
CLEVER
Manta
GenomeSTRiP
LUMPY
PopDel


CBA_J
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
pindel
GASV
RDXplorer
MiStrVar
indelMINER
GRIDSS
DELLY
BreakDancer
BioGraph*
CLEVER
Manta
GenomeSTRiP
LUMPY
PopDel

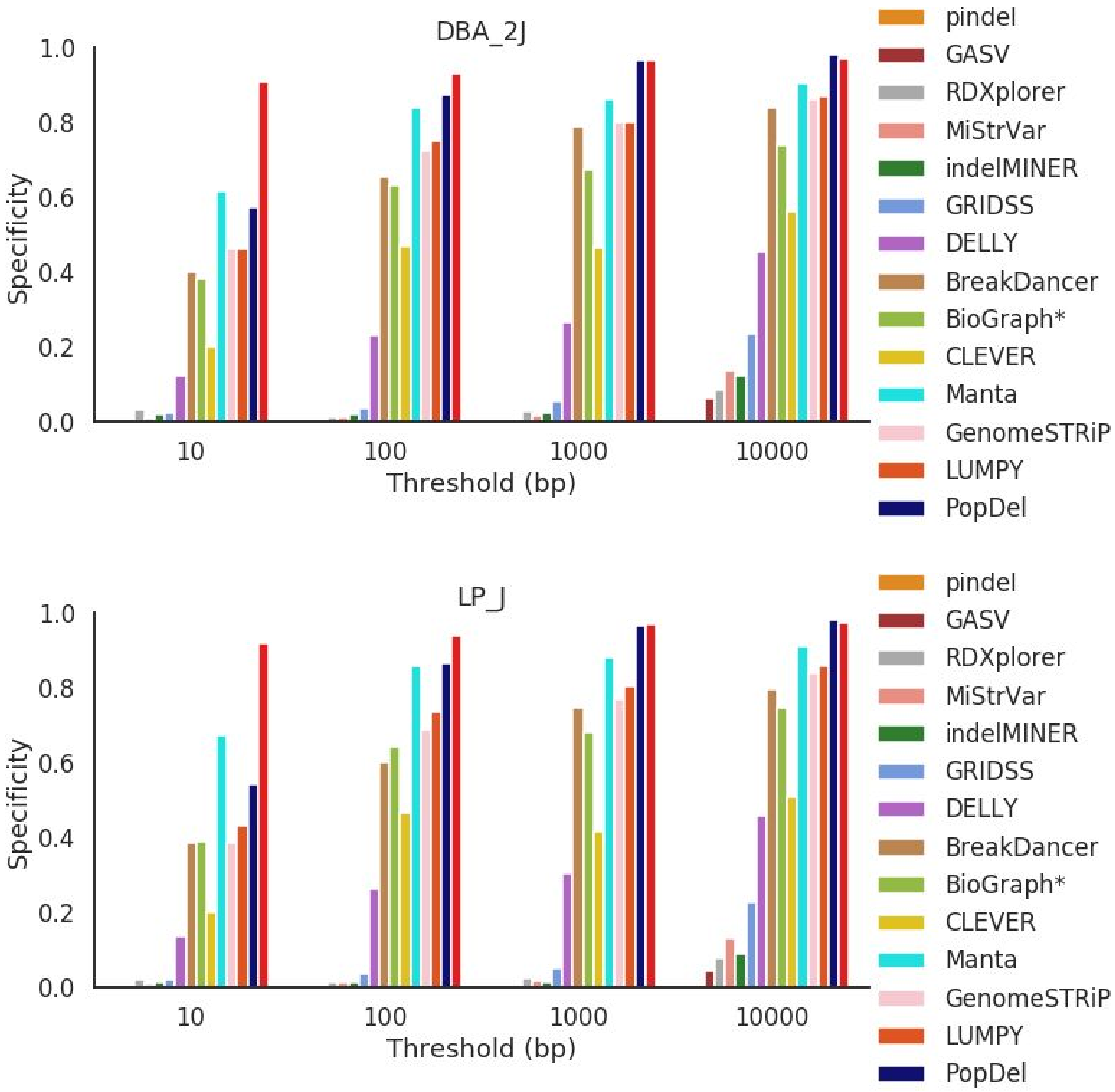
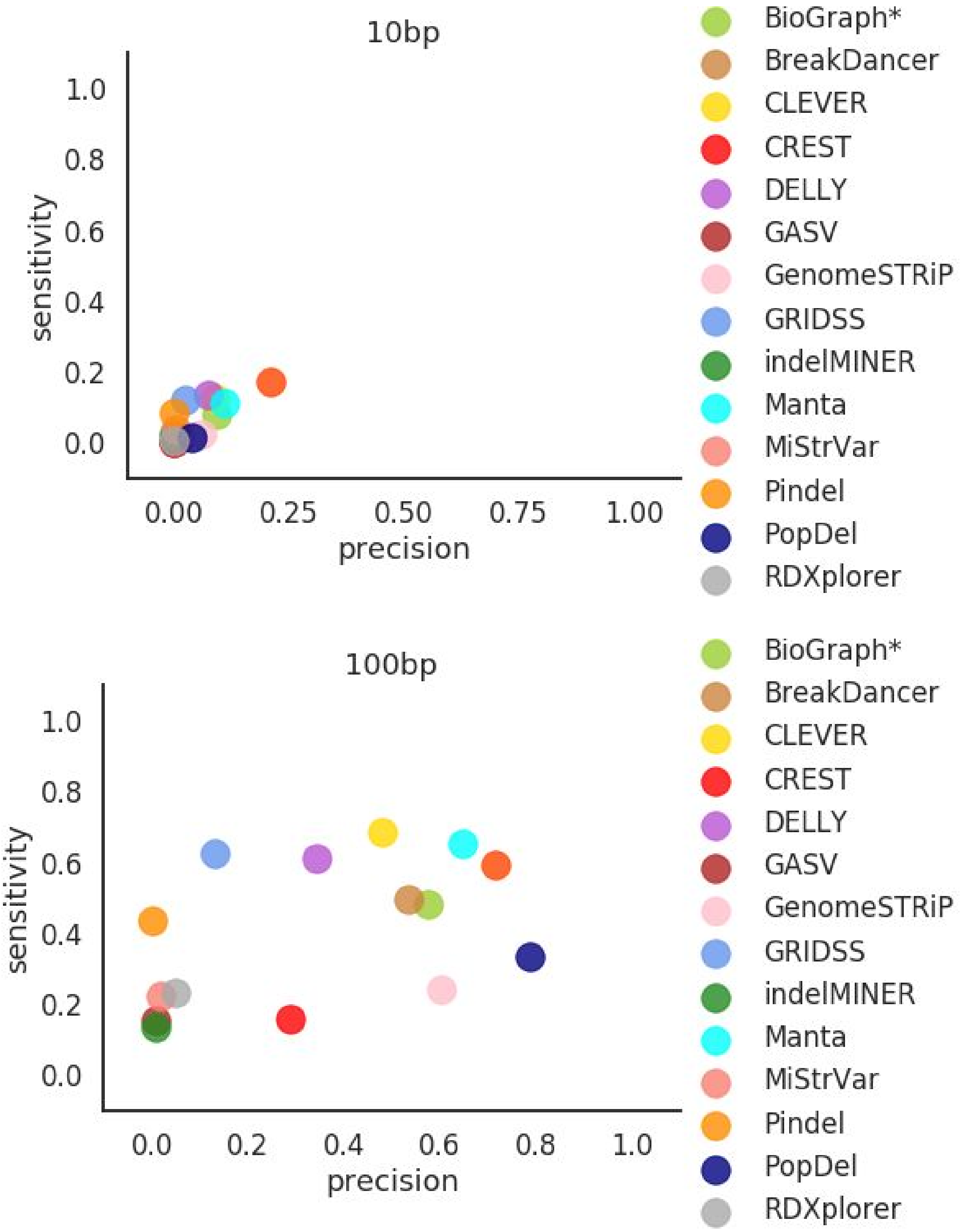


Figure S6. Specificity presented for each mouse strain across all error threshold


10bp
sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
0.00
0.25
0.50
0.75
1.00
precision
BioGraph*
BreakDancer
CLEVER
CREST
DELLY
GASV
GenomeSTRiP
GRIDSS
indelMINER
Manta
MiStrVar
Pindel
PopDel
RDXplorer
100bp
sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
precision
BioGraph*
BreakDancer
CLEVER
CREST
DELLY
GASV
GenomeSTRiP
GRIDSS
indelMINER
Manta
MiStrVar
Pindel
PopDel
RDXplorer

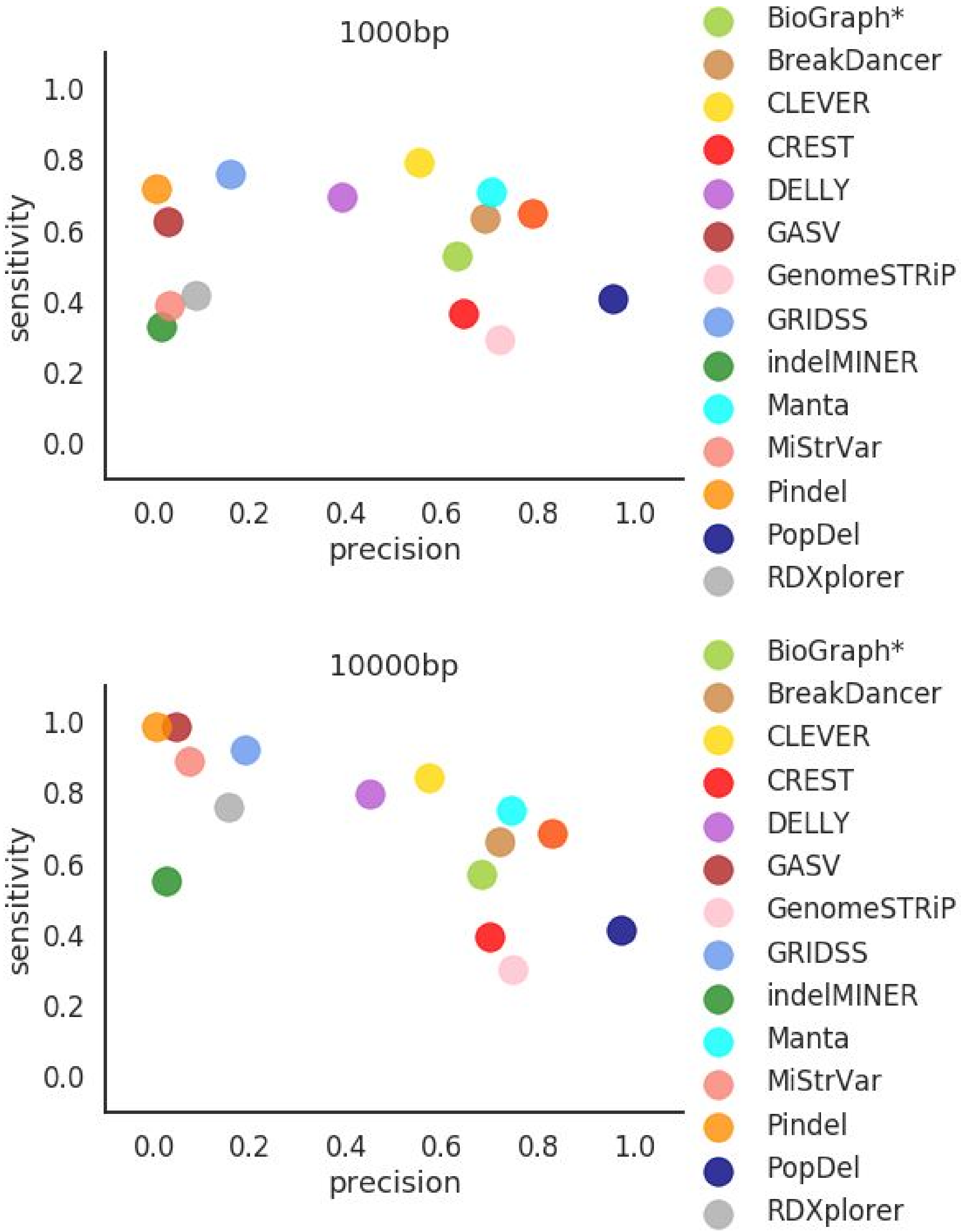
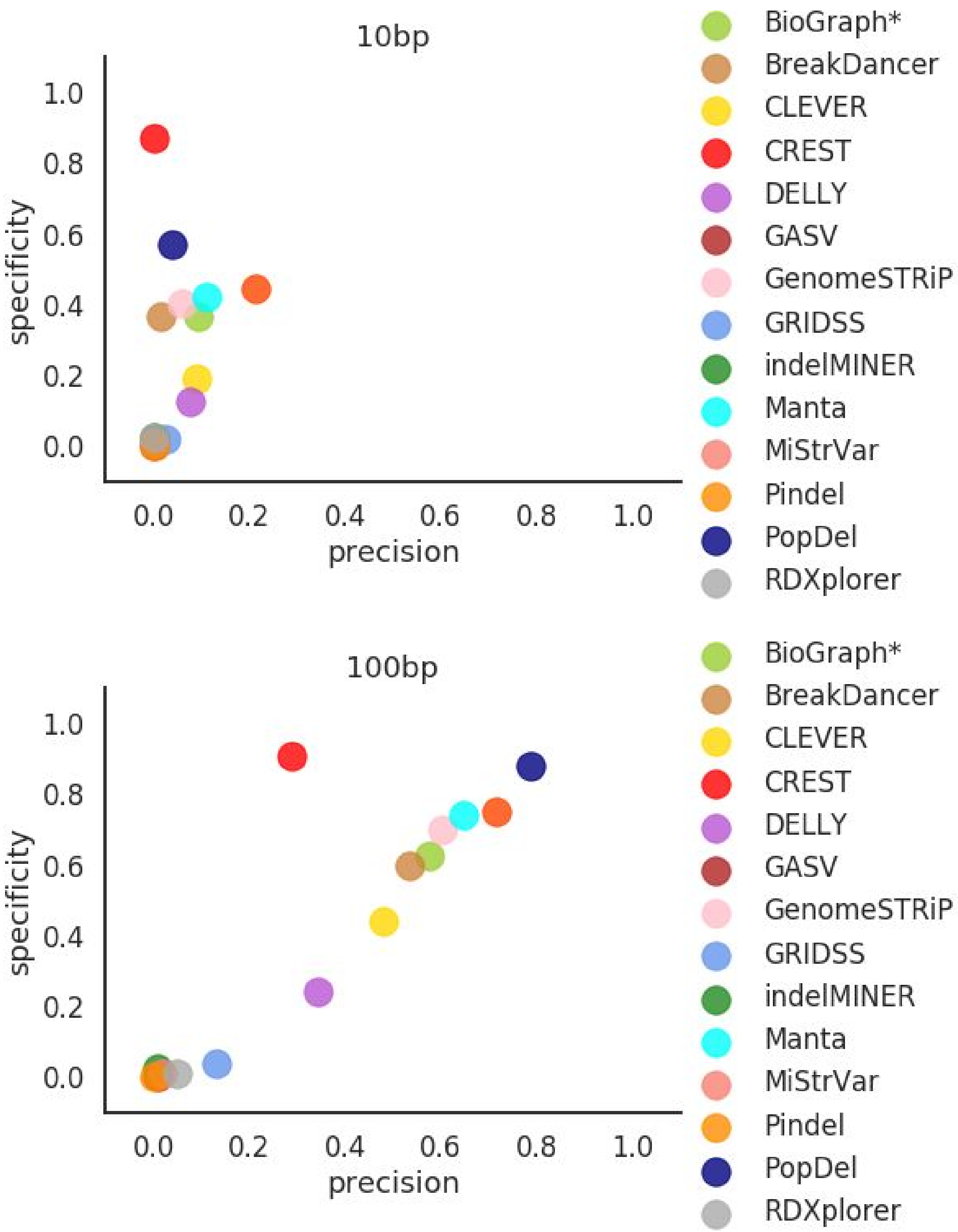


Figure S7. **S**catter plot depicting the Precision (x-axis) and Sensitivity (y-axis) for 10 bp, 100 bp, 1000 bp, and 10000 bp thresholds.


10bp
specificity
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
precision
BioGraph*
BreakDancer
CLEVER
CREST
DELLY
GASV
GenomeSTRiP
GRIDSS
indelMINER
Manta
MiStrVar
Pindel
PopDel
RDXplorer
100bp
specificity
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
precision
BioGraph*
BreakDancer
CLEVER
CREST
DELLY
GASV
GenomeSTRiP
GRIDSS
indelMINER
Manta
MiStrVar
Pindel
PopDel
RDXplorer

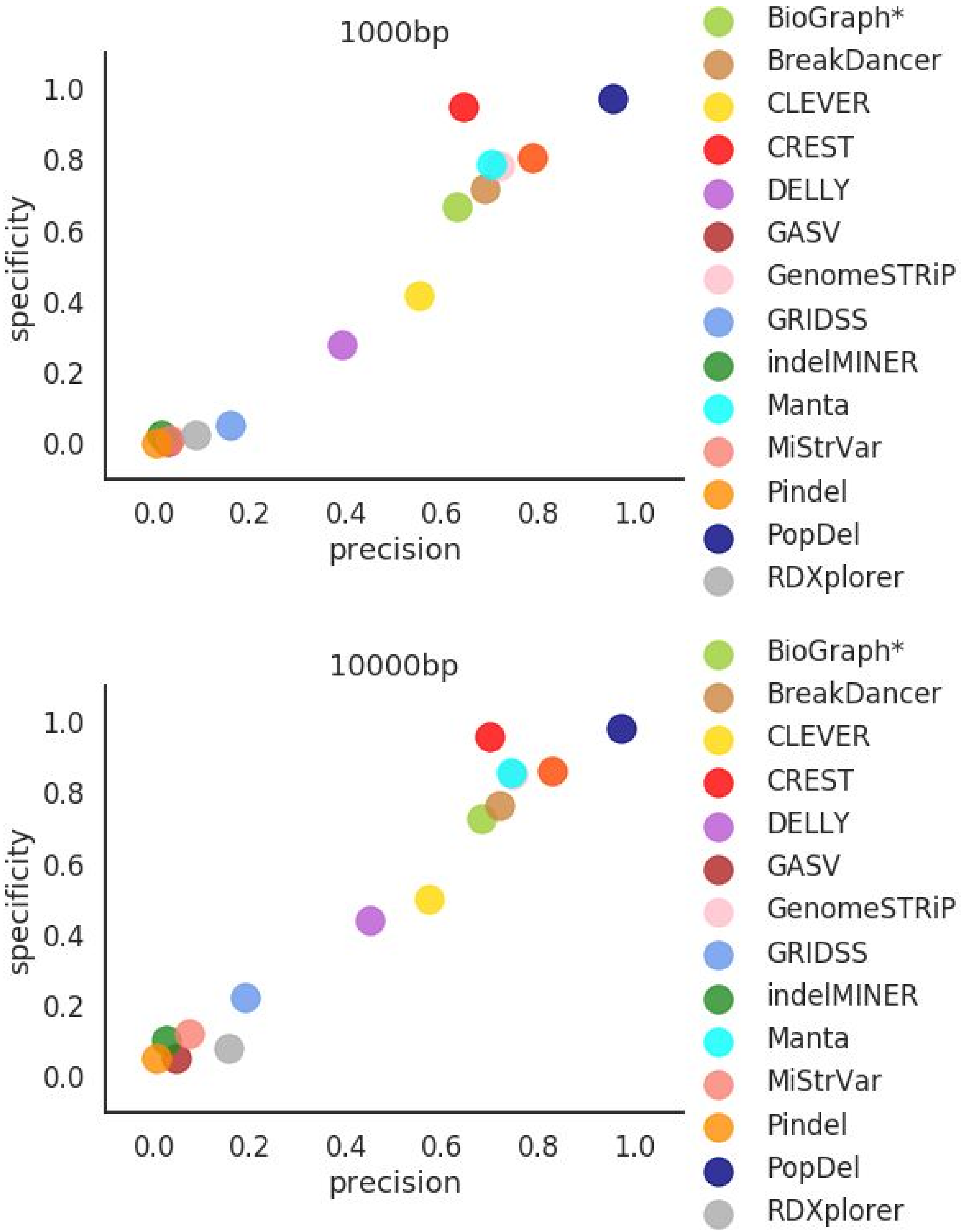


Figure S8. Scatter plot depicting the Precision (x-axis) vs Specificity (y-axis) for 10 bp, 100 bp, 1000 bp, and 10000 bp thresholds with Spearman's correlation and p-values (0.53, 0.027; 0.84, 2.37e-05, 0.92,1.18e-07; 0.93,5.64e-08) respectively

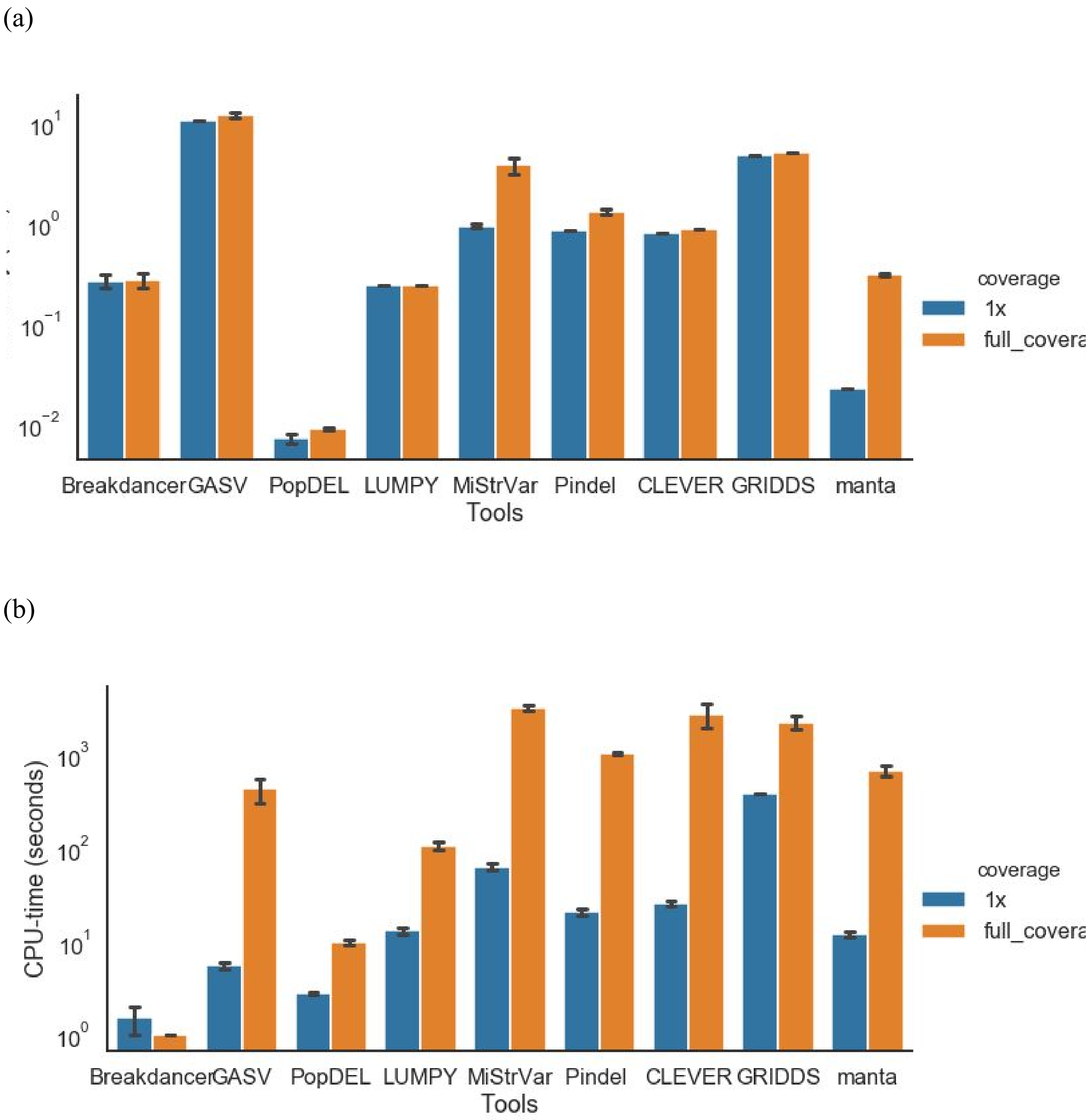
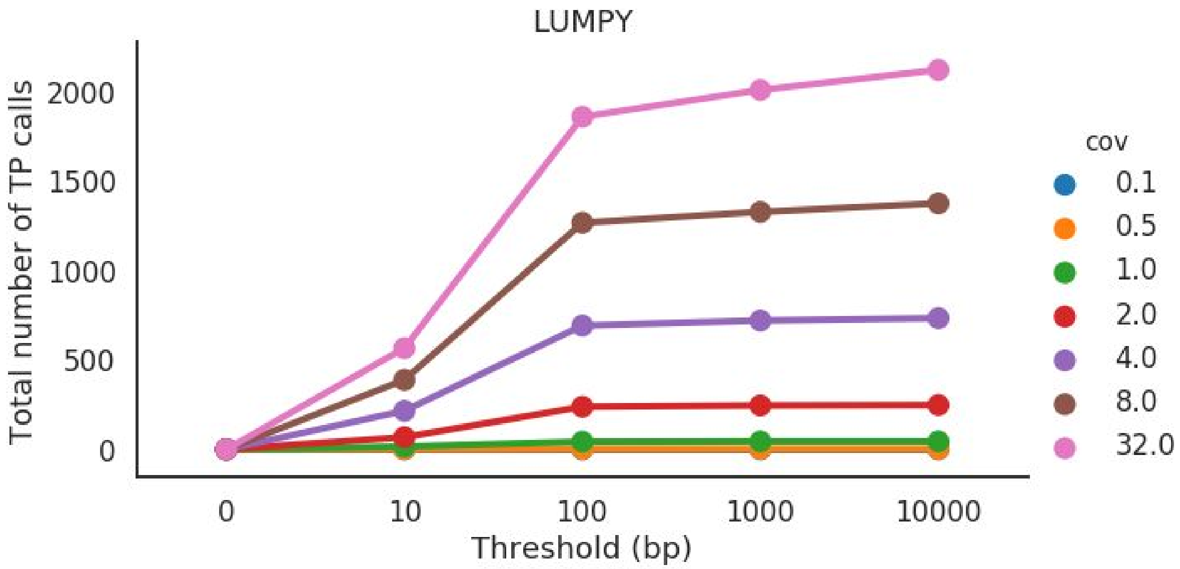
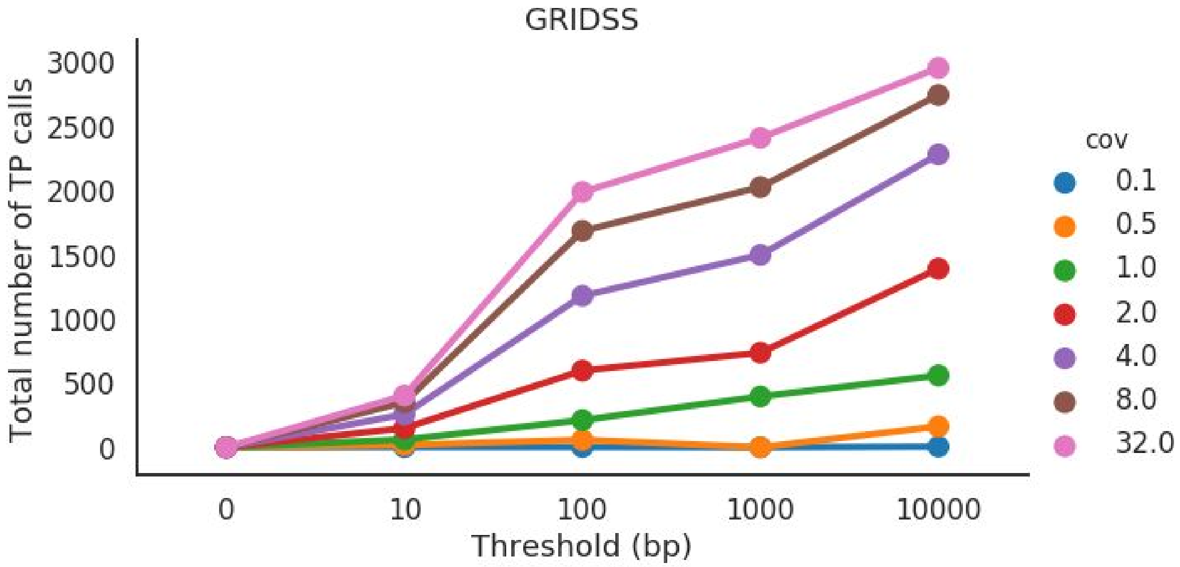
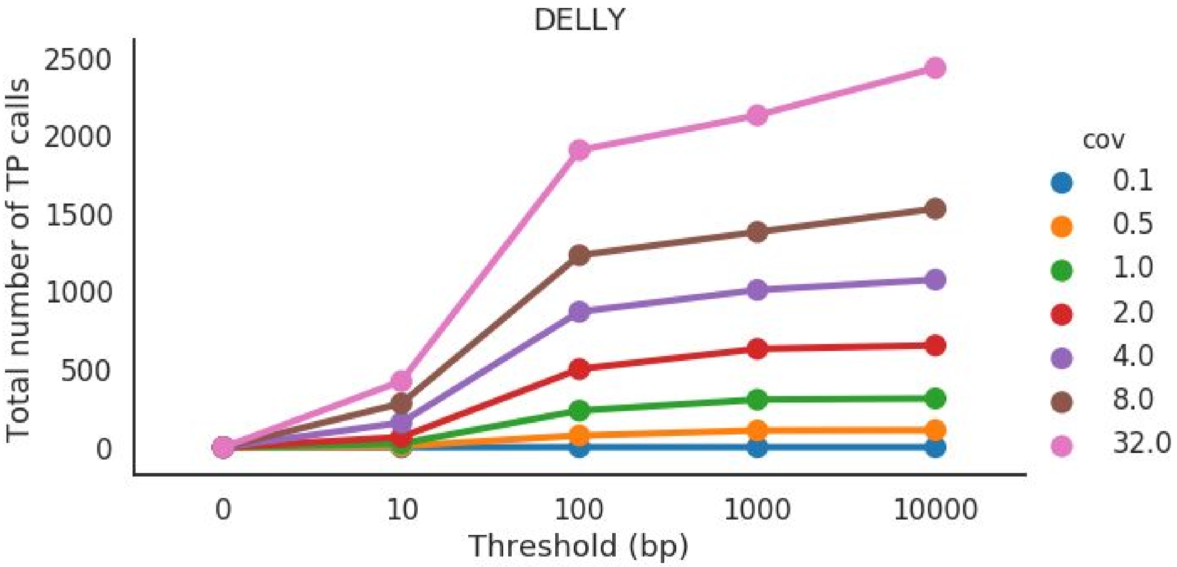
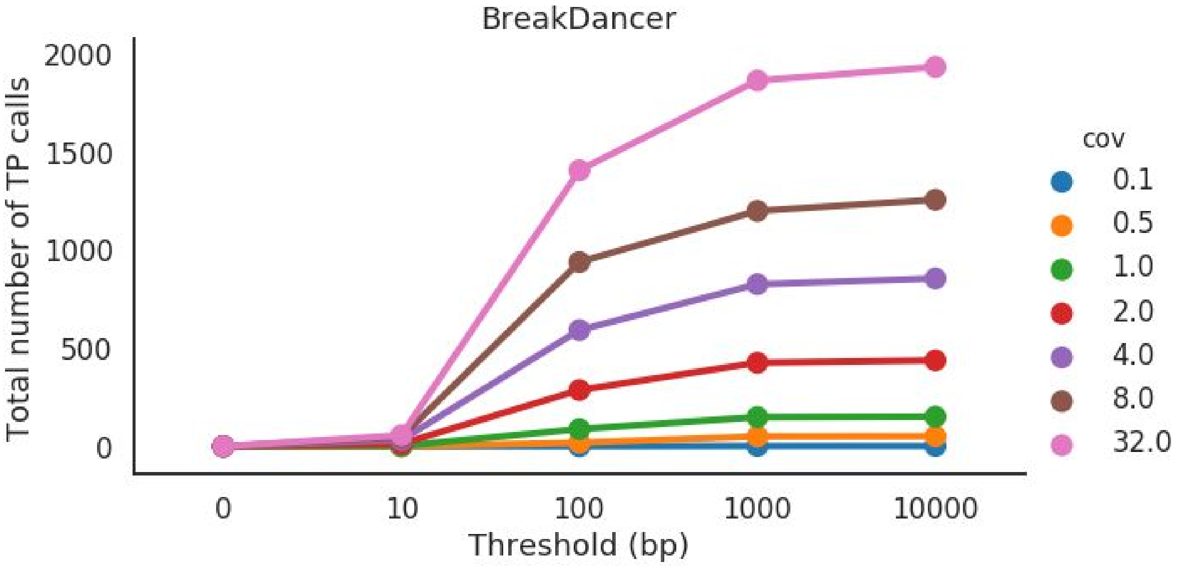

(a)
10^1
10^0
10^-1
10^-2
Breakdancer
GASV
PopDEL
LUMPY
MiStrVar
Pindel
CLEVER
GRIDDS
manta
Tools
coverage
1x
full_covera
(b)
CPU-time (seconds)
10^3
10^2
10^1
10^0
Breakdancer
GASV
PopDEL
LUMPY
MiStrVar
Pindel
CLEVER
GRIDDS
manta
Tools
coverage
1x
full_covera

Figure S9. Comparison of computational performance of SV callers. (a) A bar-plot depicting the RAM usage across all of the tools in Table 1. (b) A bar-plot depicting the CPU-time across all of the tools in Table 1.


LUMPY
Total number of TP calls
0
500
1000
1500
2000
0
10
100
1000
10000
Threshold (bp)
cov
● 0.1
● 0.5
● 1.0
● 2.0
● 4.0
● 8.0
● 32.0


GRIDSS
Total number of TP calls
0
500
1000
1500
2000
2500
3000
0
10
100
1000
10000
Threshold (bp)
cov
● 0.1
● 0.5
● 1.0
● 2.0
● 4.0
● 8.0
● 32.0


DELLY
Total number of TP calls
2500
2000
1500
1000
500
0
0
10
100
1000
10000
Threshold (bp)
cov
0.1
0.5
1.0
2.0
4.0
8.0
32.0


BreakDancer
Total number of TP calls
2000
1500
1000
500
0
0
10
100
1000
10000
Threshold (bp)
cov
0.1
0.5
1.0
2.0
4.0
8.0
32.0

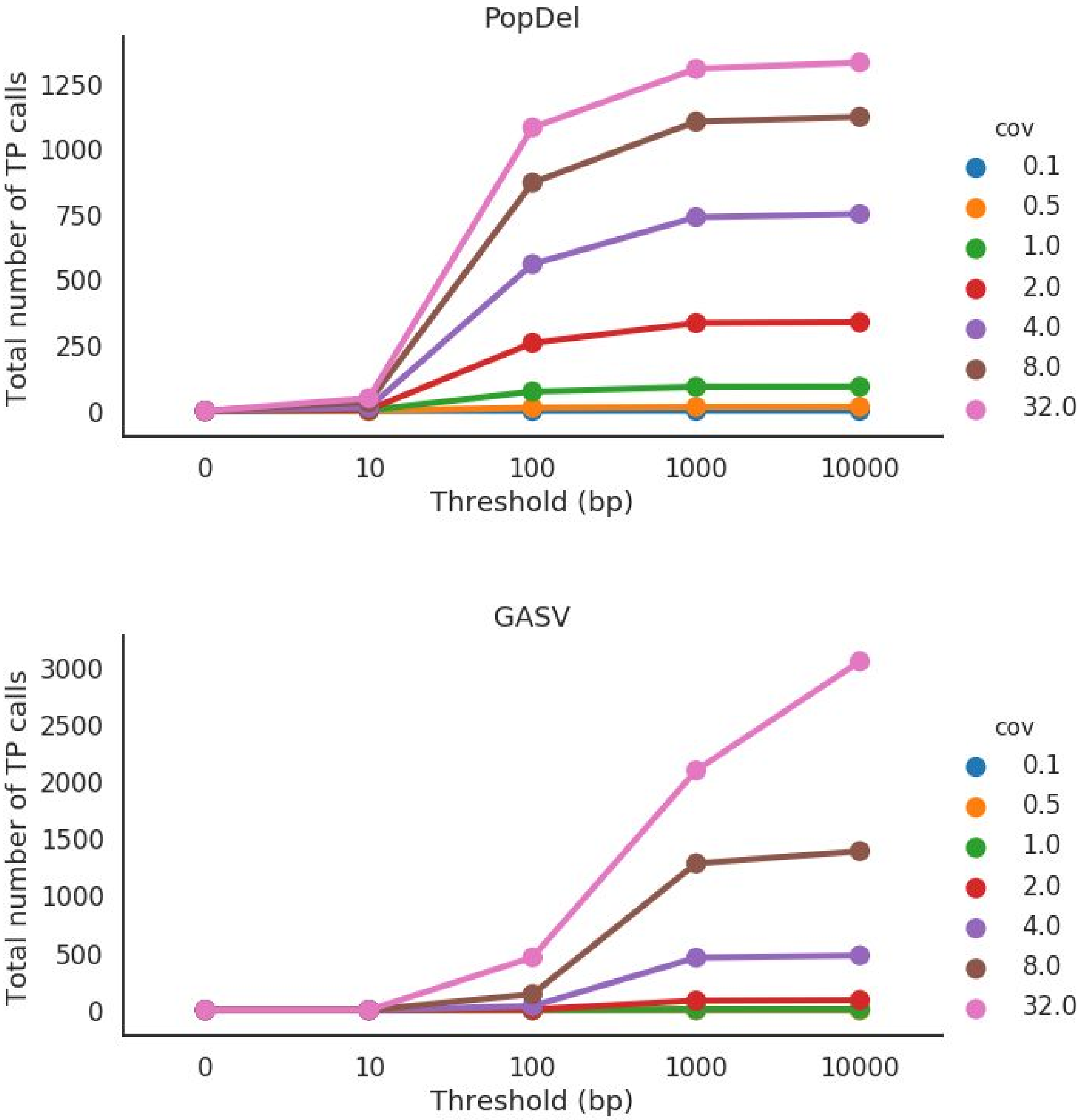
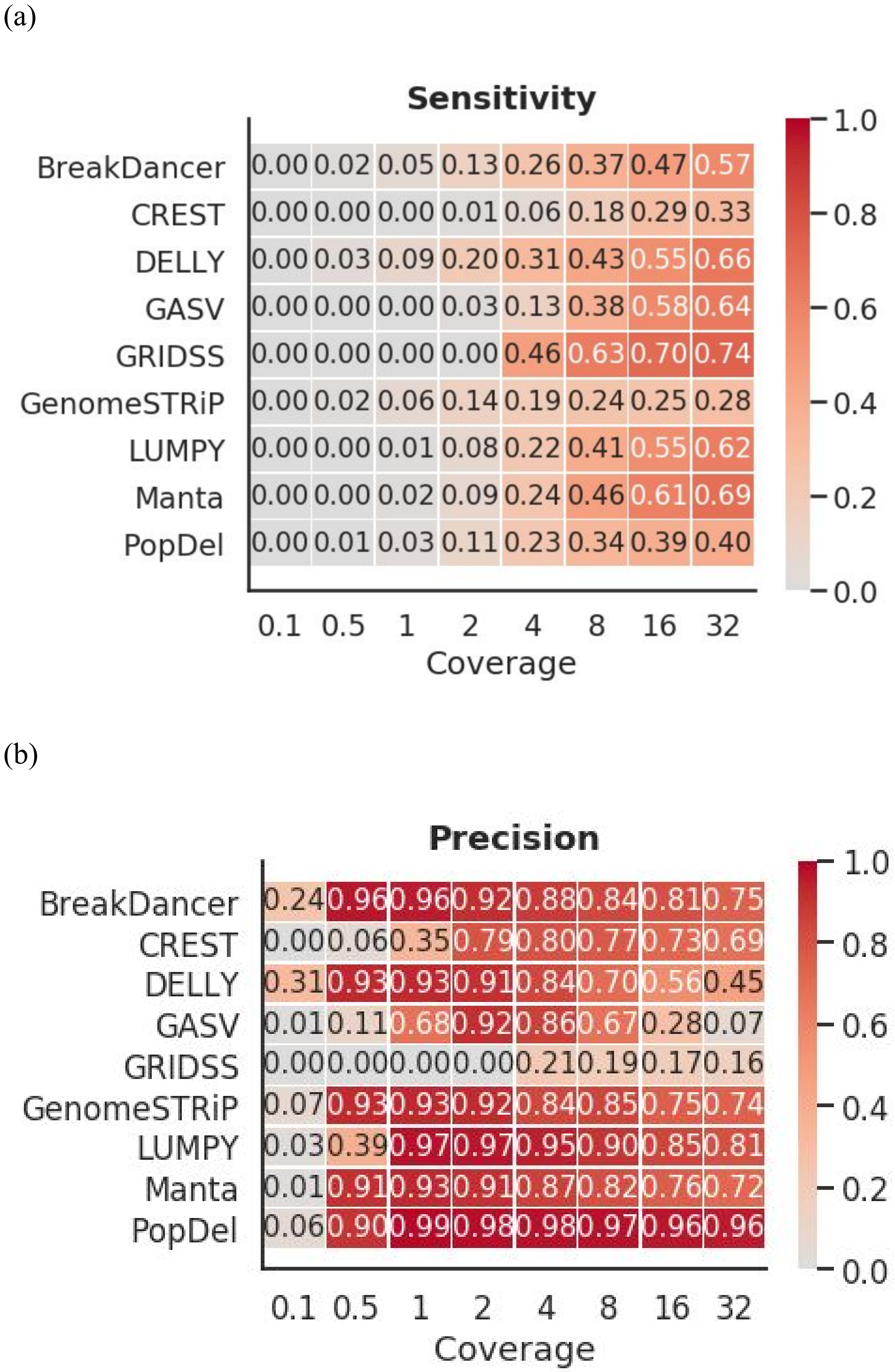
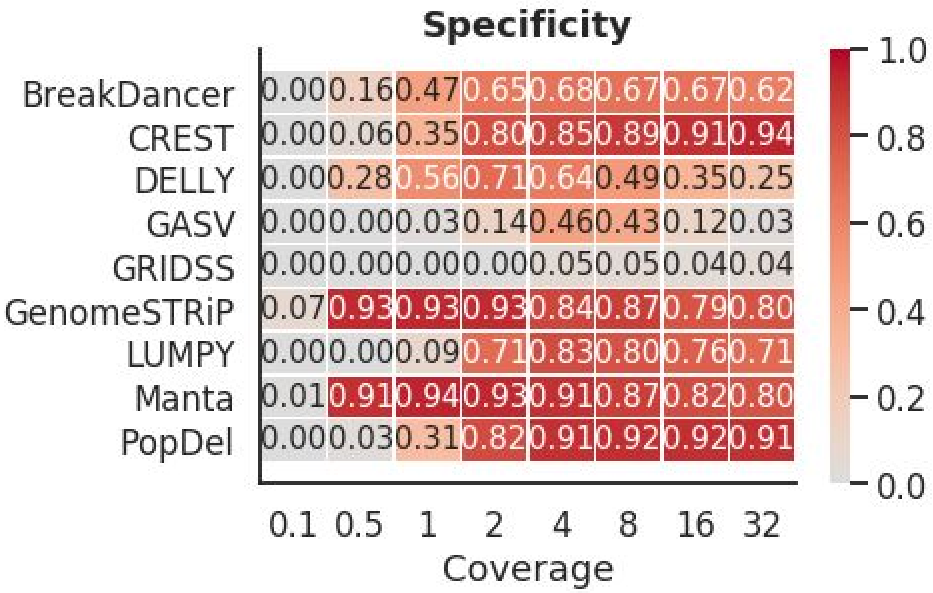


Figure S10**.** Number of correctly detected deletions (true positives - 'TP') by SV callers across various thresholds and genome coverages . Deletion is considered to be correctly predicted if the distance of right and left coordinates are within the given threshold from the coordinates of true deletion.


(a)
Sensitivity
BreakDancer 0.00 0.02 0.05 0.13 0.26 0.37 0.47 0.57
CREST 0.00 0.00 0.00 0.01 0.06 0.18 0.29 0.33
DELLY 0.00 0.03 0.09 0.20 0.31 0.43 0.55 0.66
GASV 0.00 0.00 0.00 0.03 0.13 0.38 0.58 0.64
GRIDSS 0.00 0.00 0.00 0.00 0.46 0.63 0.70 0.74
GenomeSTRiP 0.00 0.02 0.06 0.14 0.19 0.24 0.25 0.28
LUMPY 0.00 0.00 0.01 0.08 0.22 0.41 0.55 0.62
Manta 0.00 0.00 0.02 0.09 0.24 0.46 0.61 0.69
PopDel 0.00 0.01 0.03 0.11 0.23 0.34 0.39 0.40
0.1 0.5 1 2 4 8 16 32
Coverage
1.0 0.8 0.6 0.4 0.2 0.0
(b)
Precision
BreakDancer 0.24 0.96 0.96 0.92 0.88 0.84 0.81 0.75
CREST 0.00 0.06 0.35 0.79 0.80 0.77 0.73 0.69
DELLY 0.31 0.93 0.93 0.91 0.84 0.70 0.56 0.45
GASV 0.01 0.11 0.68 0.92 0.86 0.67 0.28 0.07
GRIDSS 0.00 0.00 0.00 0.00 0.21 0.19 0.17 0.16
GenomeSTRiP 0.07 0.93 0.93 0.92 0.84 0.85 0.75 0.74
LUMPY 0.03 0.39 0.97 0.97 0.95 0.90 0.85 0.81
Manta 0.01 0.91 0.93 0.91 0.87 0.82 0.76 0.72
PopDel 0.06 0.90 0.99 0.98 0.98 0.97 0.96 0.96
0.1 0.5 1 2 4 8 16 32
Coverage
1.0 0.8 0.6 0.4 0.2 0.0

(c)


Specificity
BreakDancer 0.00 0.16 0.47 0.65 0.68 0.67 0.67 0.62
CREST 0.00 0.06 0.35 0.80 0.85 0.89 0.91 0.94
DELLY 0.00 0.28 0.56 0.71 0.64 0.49 0.35 0.25
GASV 0.00 0.00 0.03 0.14 0.46 0.43 0.12 0.03
GRIDSS 0.00 0.00 0.00 0.00 0.05 0.05 0.04 0.04
GenomeSTRiP 0.07 0.93 0.93 0.93 0.84 0.87 0.79 0.80
LUMPY 0.00 0.00 0.09 0.71 0.83 0.80 0.76 0.71
Manta 0.01 0.91 0.94 0.93 0.91 0.87 0.82 0.80
PopDel 0.00 0.03 0.31 0.82 0.91 0.92 0.92 0.91
0.1 0.5 1 2 4 8 16 32
Coverage
1.0
0.8
0.6
0.4
0.2
0.0

(d)

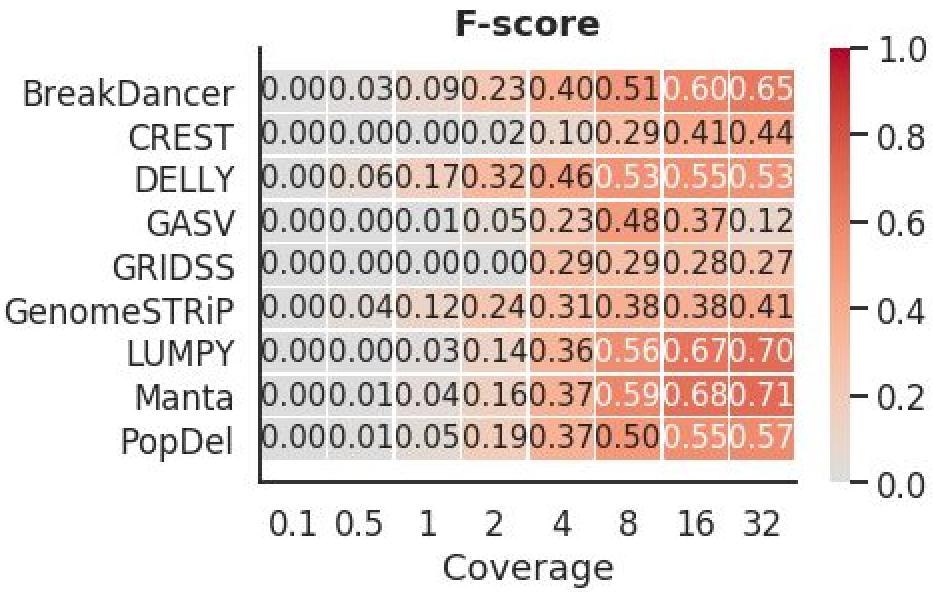
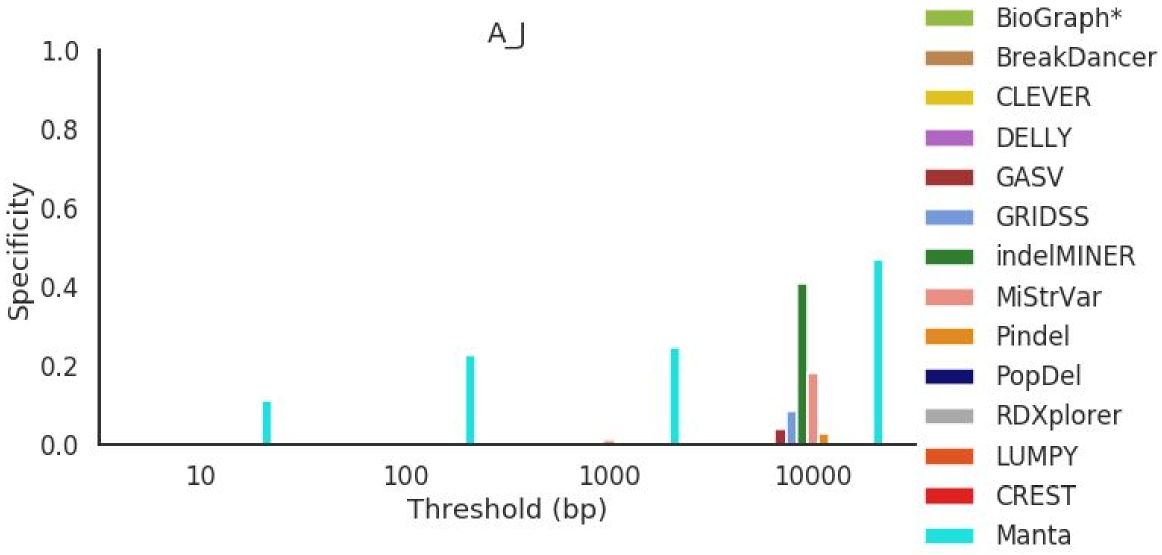
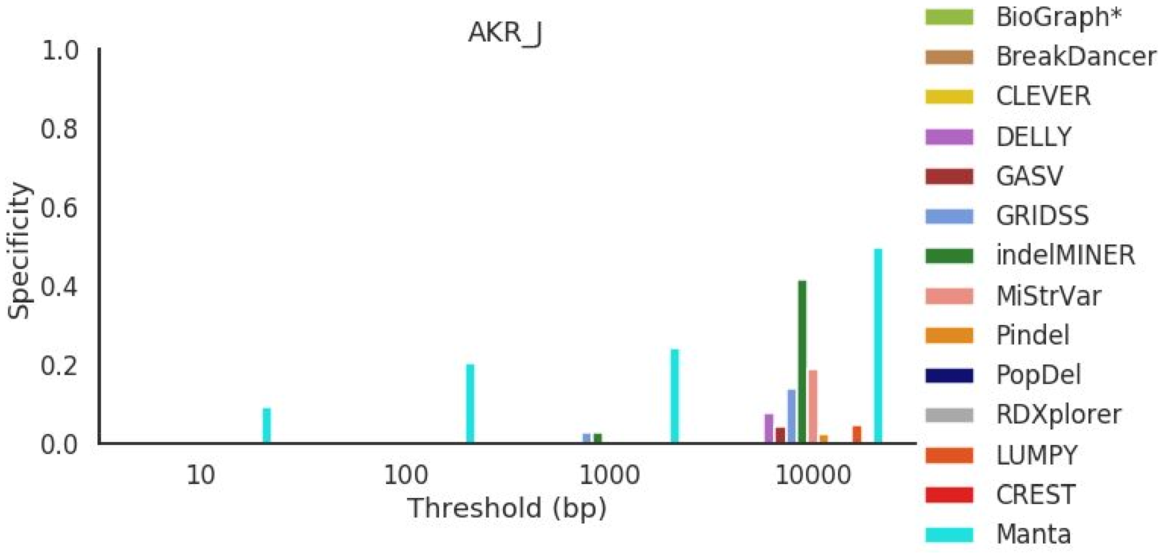
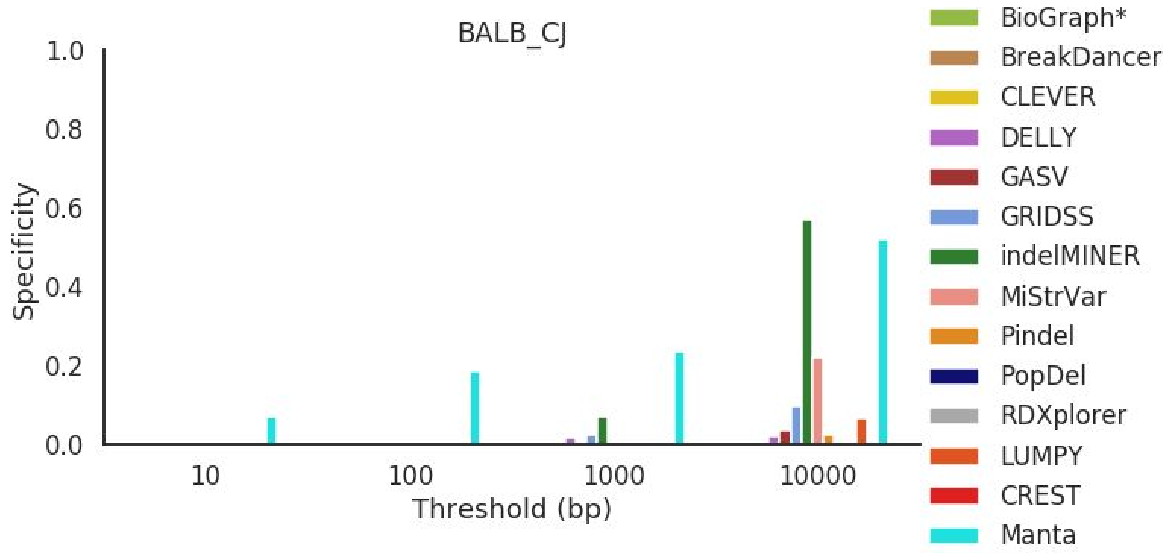
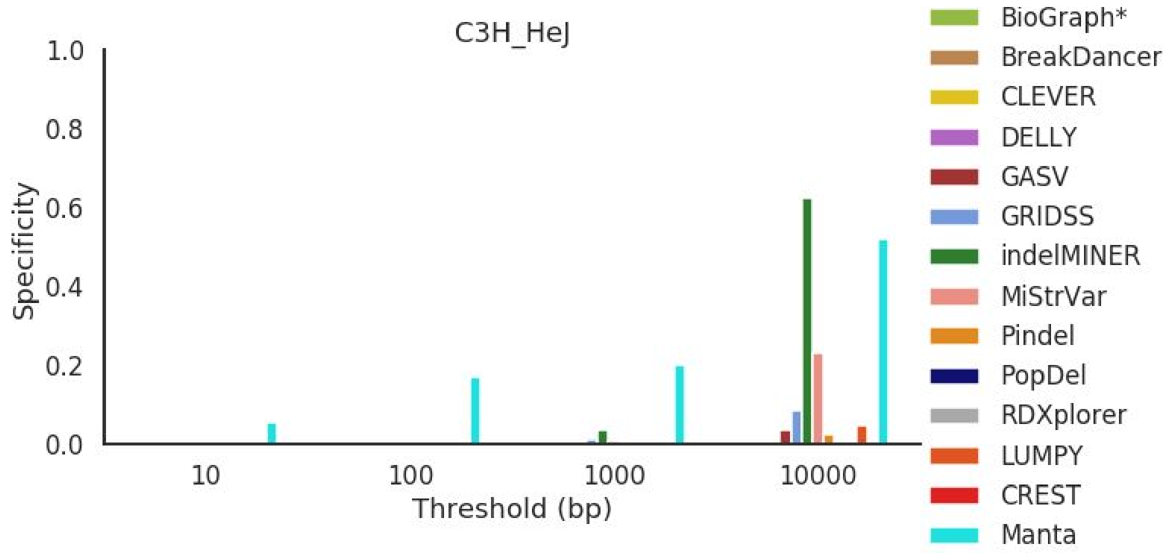

F-score
BreakDancer 0.00 0.03 0.09 0.23 0.40 0.51 0.60 0.65
CREST 0.00 0.00 0.00 0.02 0.10 0.29 0.41 0.44
DELLY 0.00 0.06 0.17 0.32 0.46 0.53 0.55 0.53
GASV 0.00 0.00 0.01 0.05 0.23 0.48 0.37 0.12
GRIDSS 0.00 0.00 0.00 0.00 0.29 0.29 0.28 0.27
GenomeSTRiP 0.00 0.04 0.12 0.24 0.31 0.38 0.38 0.41
LUMPY 0.00 0.00 0.03 0.14 0.36 0.56 0.67 0.70
Manta 0.00 0.01 0.04 0.16 0.37 0.59 0.68 0.71
PopDel 0.00 0.01 0.05 0.19 0.37 0.50 0.55 0.57
0.1 0.5 1 2 4 8 16 32
Coverage
1.0
0.8
0.6
0.4
0.2
0.0

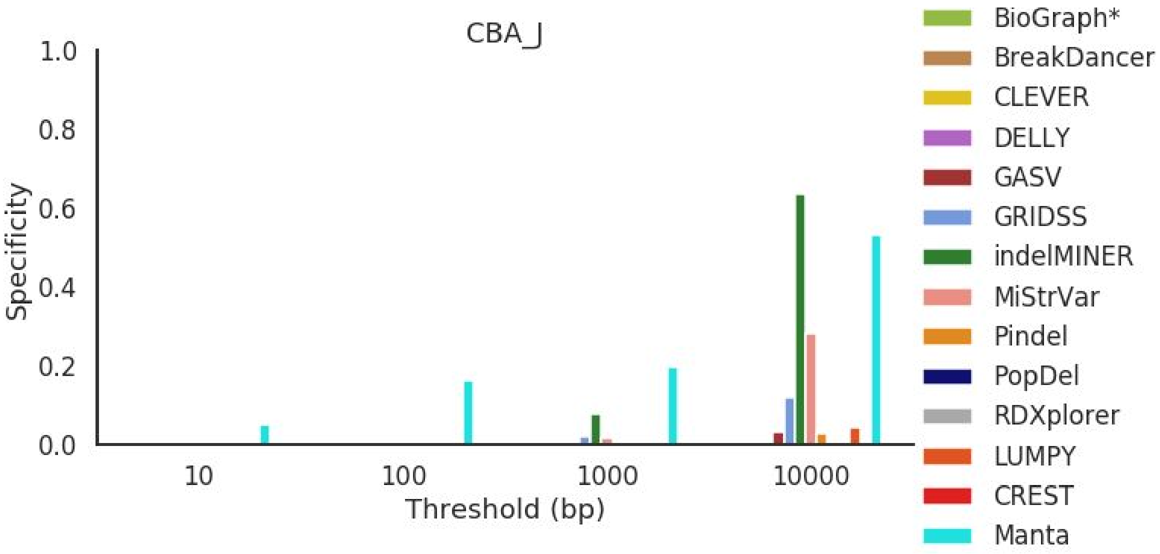
Figure S11. (a) Heatmap depicting the sensitivity based on 1000 bp threshold across various levels of coverage. (b) Heatmap depicting the precision based on 1000 bp threshold across various levels of coverage. (c) Heatmap depicting the specificity based on 1000 bp threshold across various levels of coverage. (d) Heatmap depicting the F-score based on 1000 bp threshold across various levels of coverage.






BALB_CJ
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


C3H_HeJ
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


CBA_J
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta

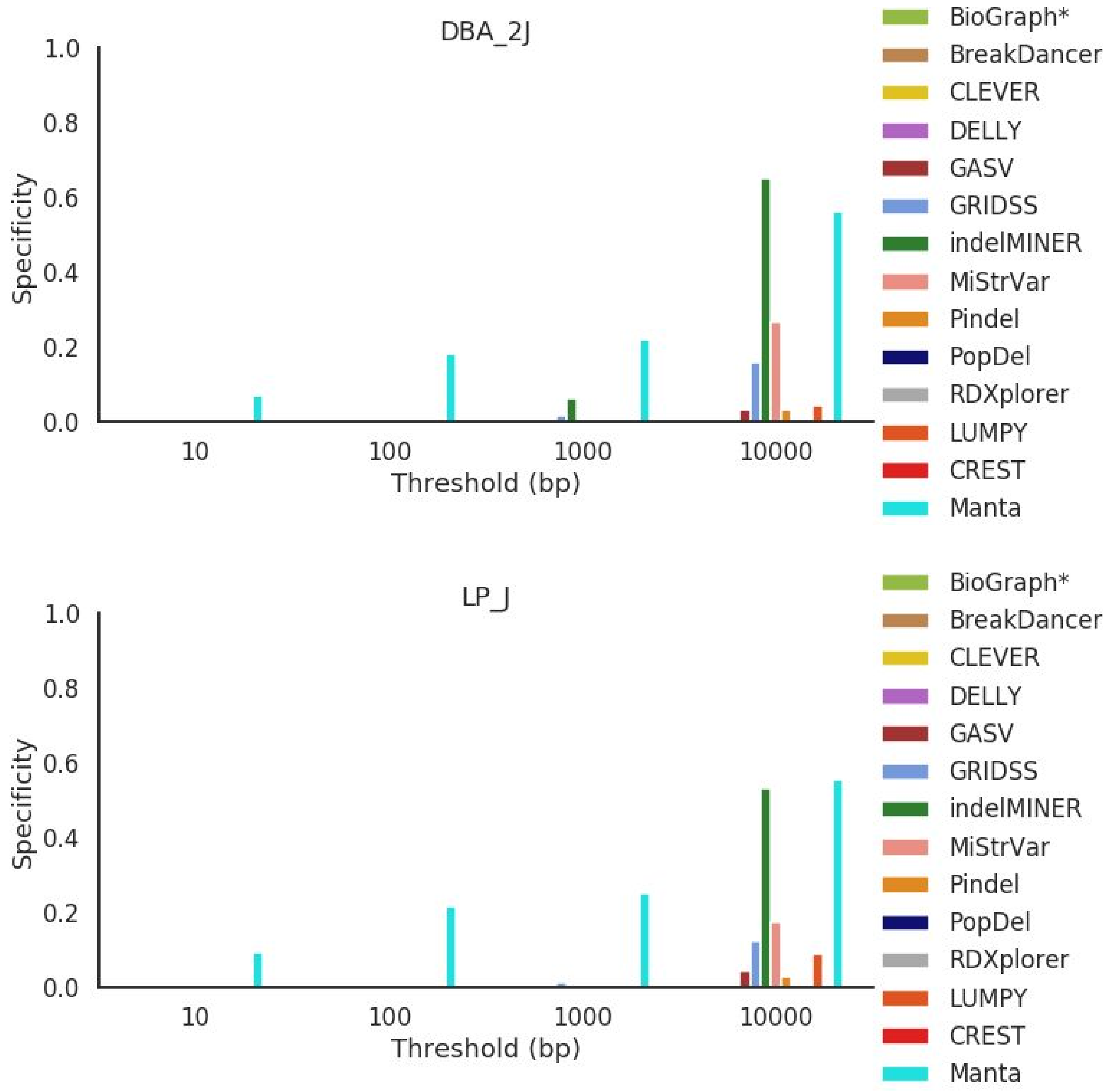
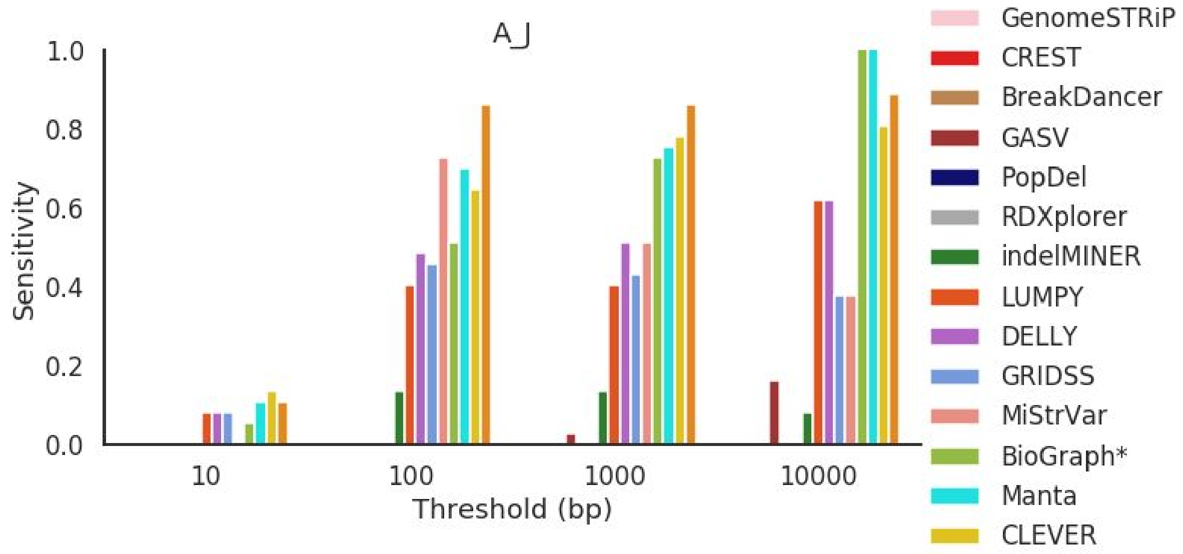
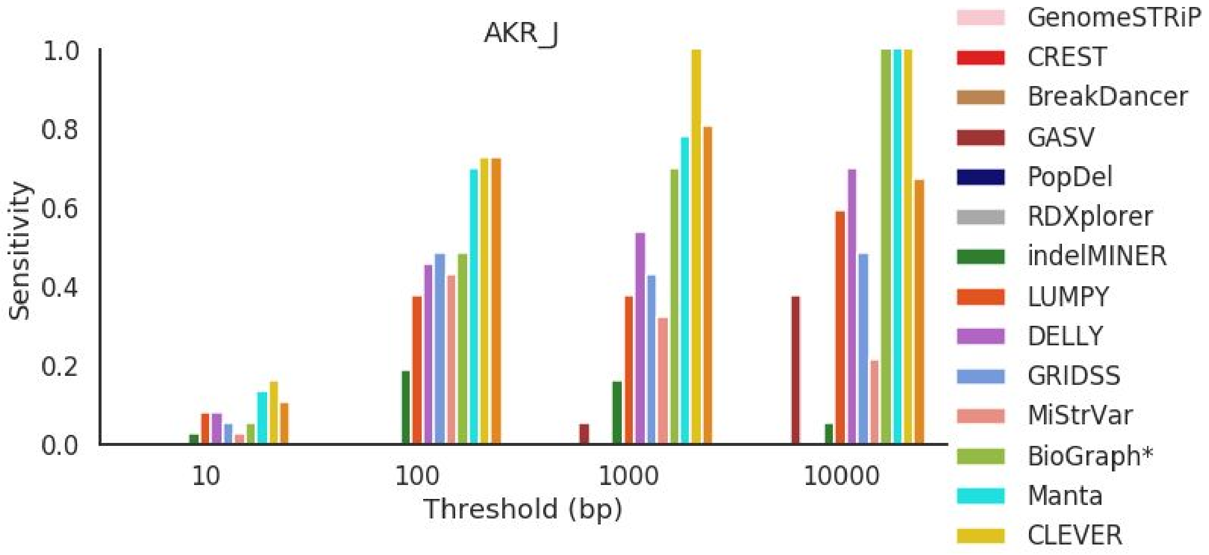
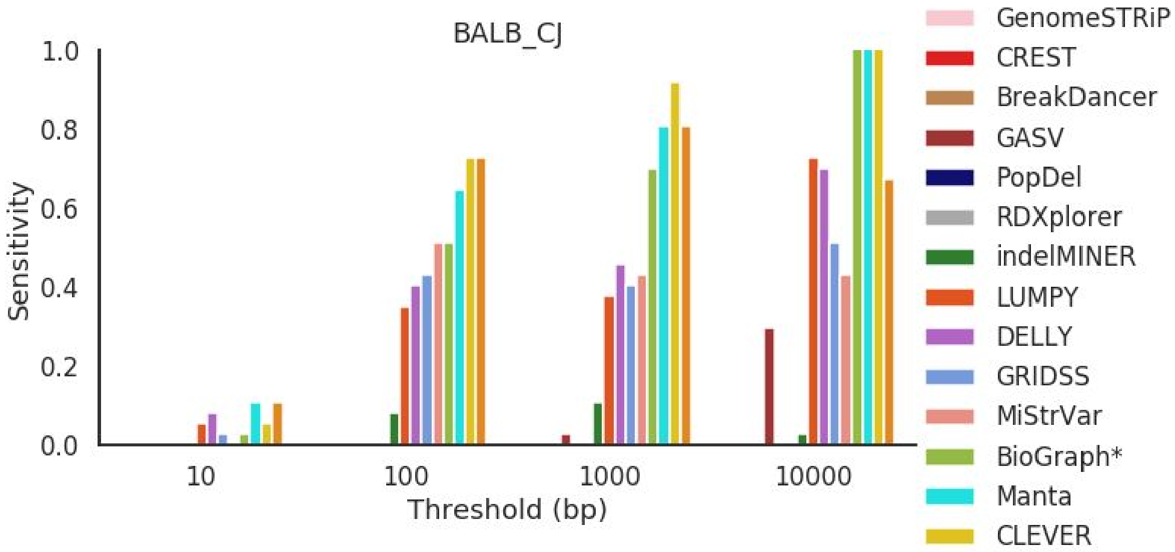
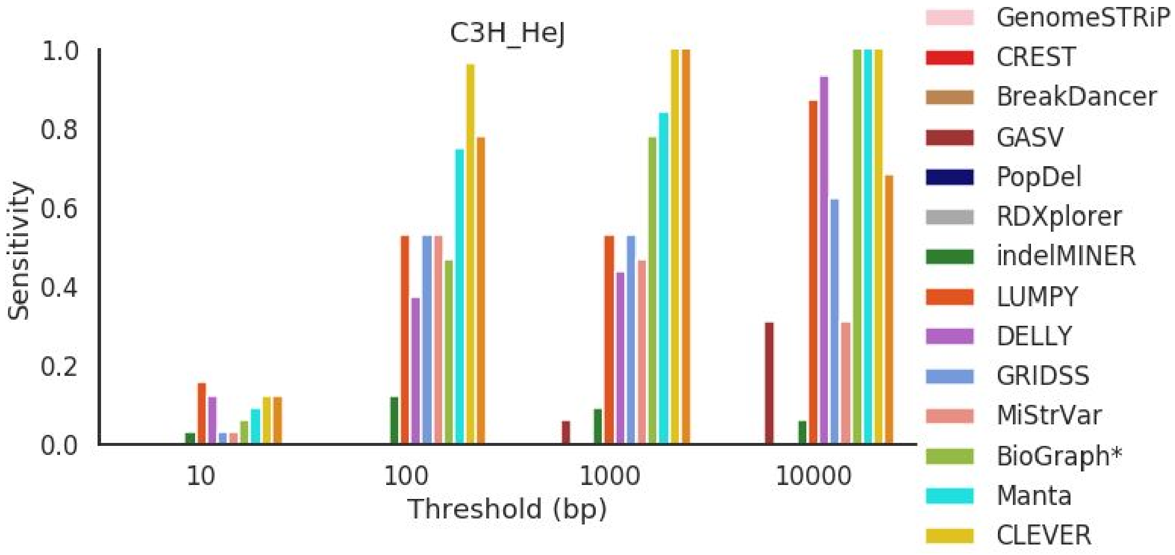


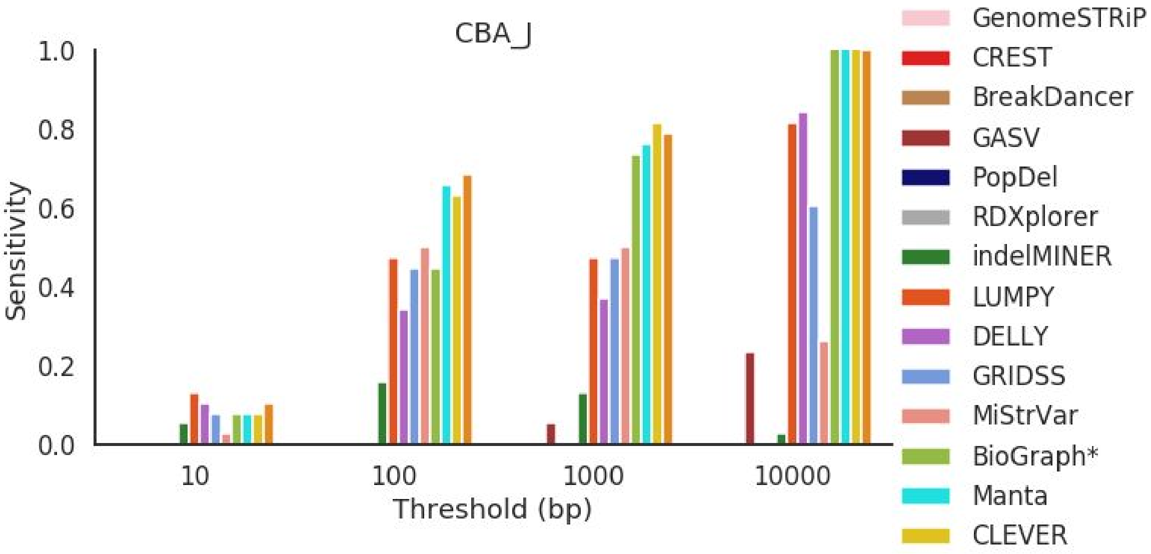
Figure S12. Specificity across all mouse strains for deletions between 50 bp and 100 bp in length.


A_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GenomeSTRiP
CREST
BreakDancer
GASV
PopDel
RDXplorer
indelMINER
LUMPY
DELLY
GRIDSS
MiStrVar
BioGraph*
Manta
CLEVER


AKR_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GenomeSTRiP
CREST
BreakDancer
GASV
PopDel
RDXplorer
indelMINER
LUMPY
DELLY
GRIDSS
MiStrVar
BioGraph*
Manta
CLEVER


BALB_CJ
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GenomeSTRiP
CREST
BreakDancer
GASV
PopDel
RDXplorer
indelMINER
LUMPY
DELLY
GRIDSS
MiStrVar
BioGraph*
Manta
CLEVER


C3H_HeJ
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GenomeSTRiP
CREST
BreakDancer
GASV
PopDel
RDXplorer
indelMINER
LUMPY
DELLY
GRIDSS
MiStrVar
BioGraph*
Manta
CLEVER


CBA_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GenomeSTRiP
CREST
BreakDancer
GASV
PopDel
RDXplorer
indelMINER
LUMPY
DELLY
GRIDSS
MiStrVar
BioGraph*
Manta
CLEVER

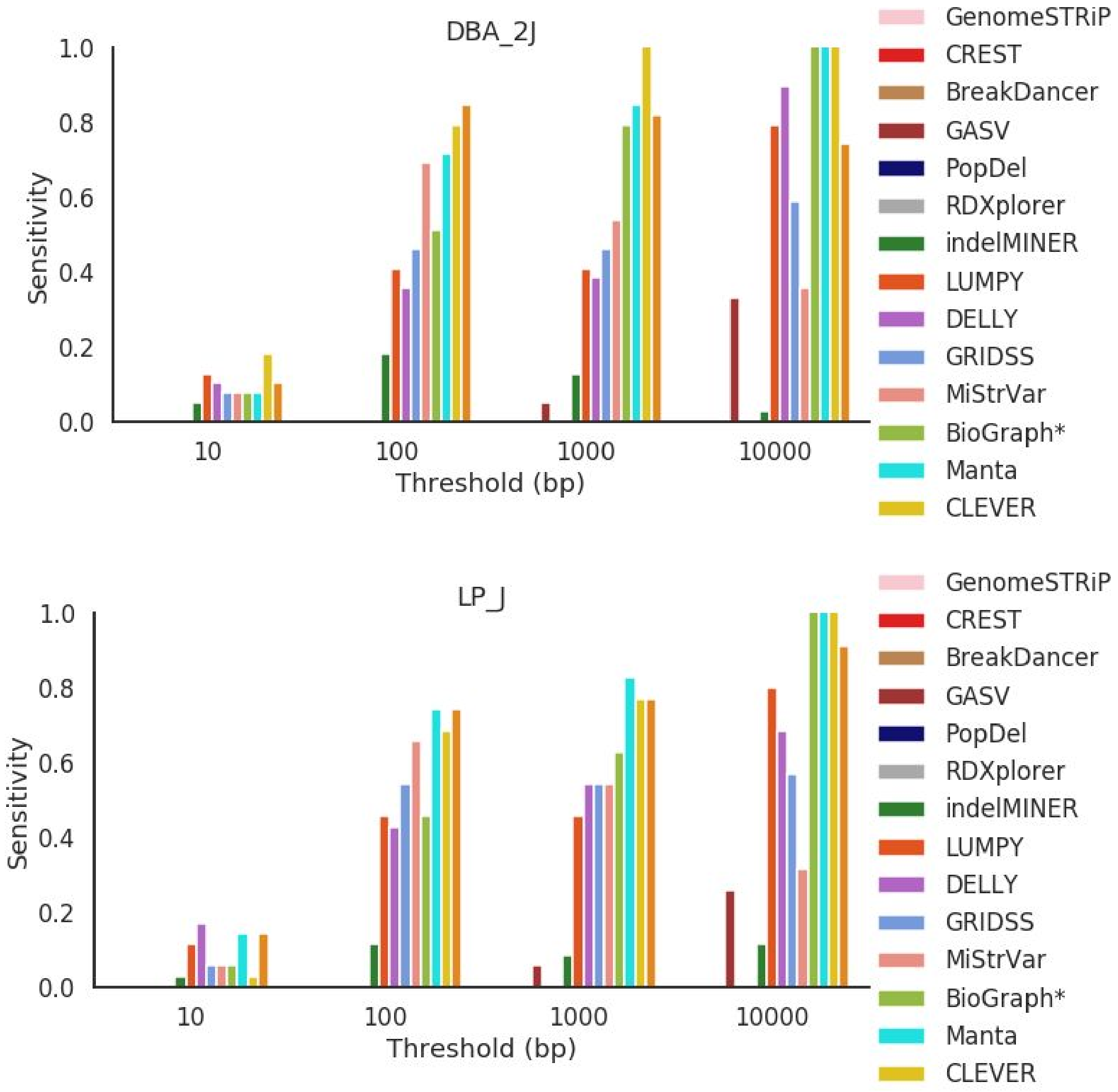
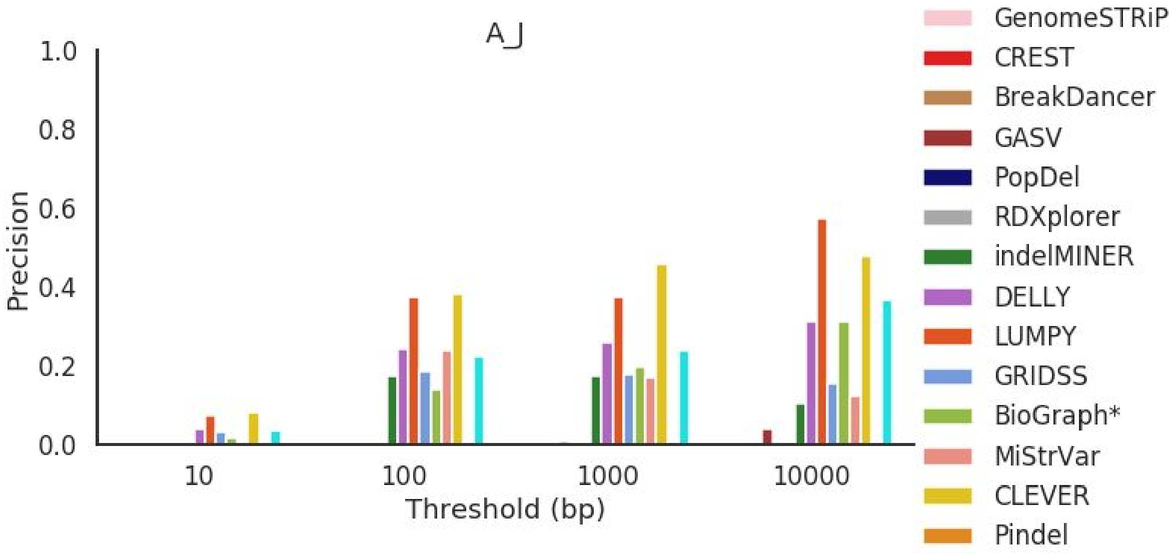
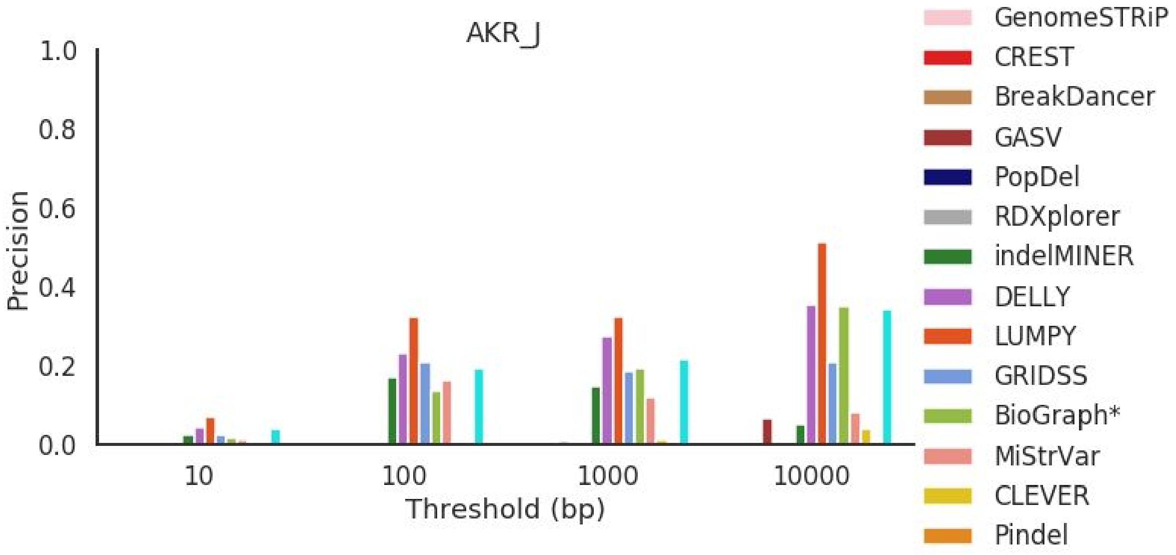
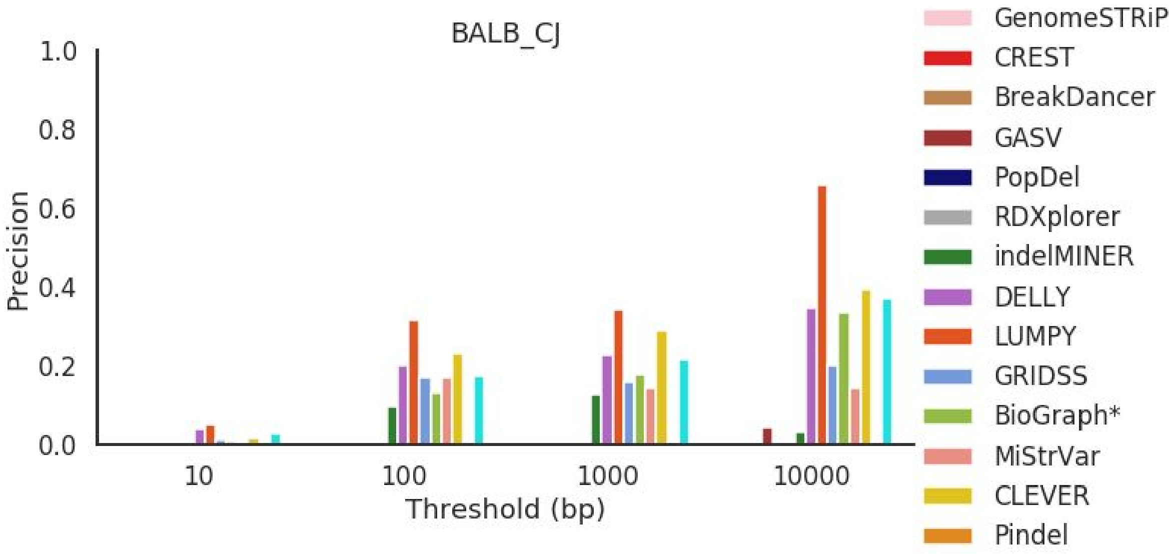
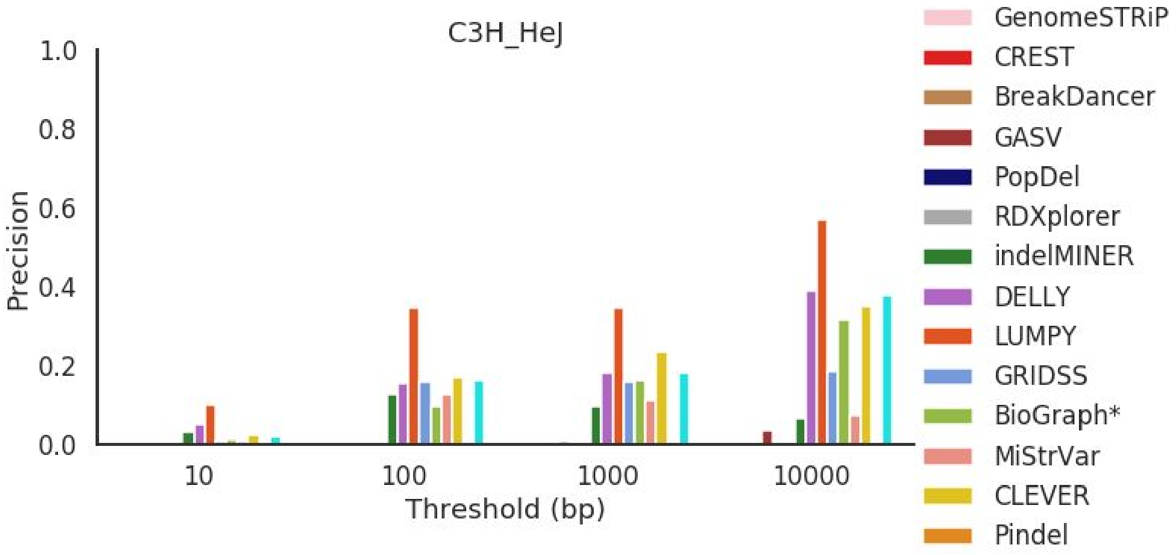


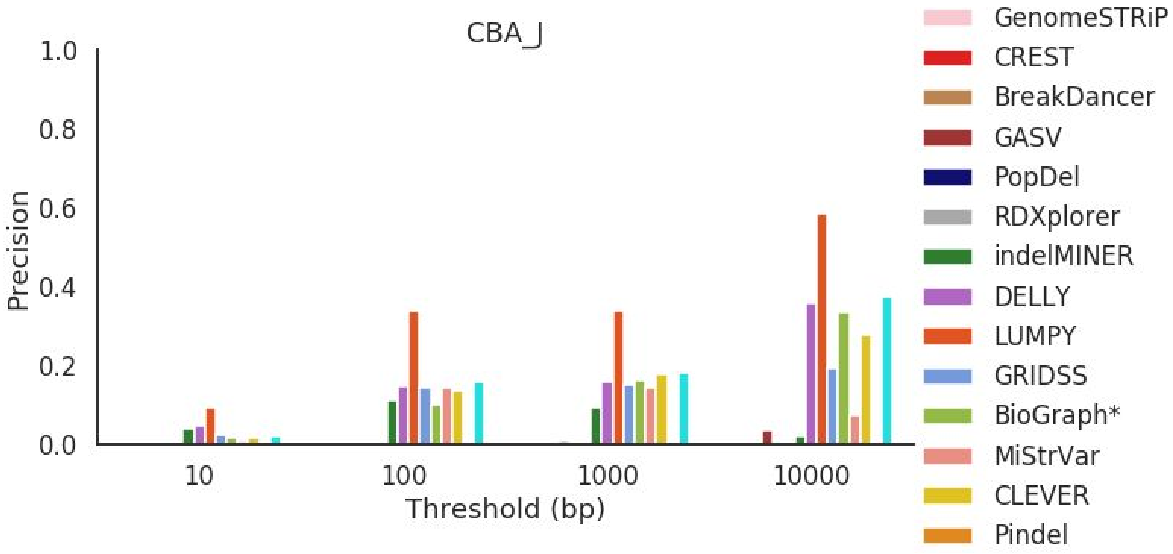
Figure S13. Sensitivity across all mouse strains for deletions between 50 bp and 100 bp in length.


A_J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GenomeSTRiP
CREST
BreakDancer
GASV
PopDel
RDXplorer
indelMINER
DELLY
LUMPY
GRIDSS
BioGraph*
MiStrVar
CLEVER
Pindel


AKR_J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GenomeSTRiP
CREST
BreakDancer
GASV
PopDel
RDXplorer
indelMINER
DELLY
LUMPY
GRIDSS
BioGraph*
MiStrVar
CLEVER
Pindel


BALB_CJ
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GenomeSTRiP
CREST
BreakDancer
GASV
PopDel
RDXplorer
indelMINER
DELLY
LUMPY
GRIDSS
BioGraph*
MiStrVar
CLEVER
Pindel


C3H_HeJ
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GenomeSTRiP
CREST
BreakDancer
GASV
PopDel
RDXplorer
indelMINER
DELLY
LUMPY
GRIDSS
BioGraph*
MiStrVar
CLEVER
Pindel


CBA_J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
GenomeSTRiP
CREST
BreakDancer
GASV
PopDel
RDXplorer
indelMINER
DELLY
LUMPY
GRIDSS
BioGraph*
MiStrVar
CLEVER
Pindel

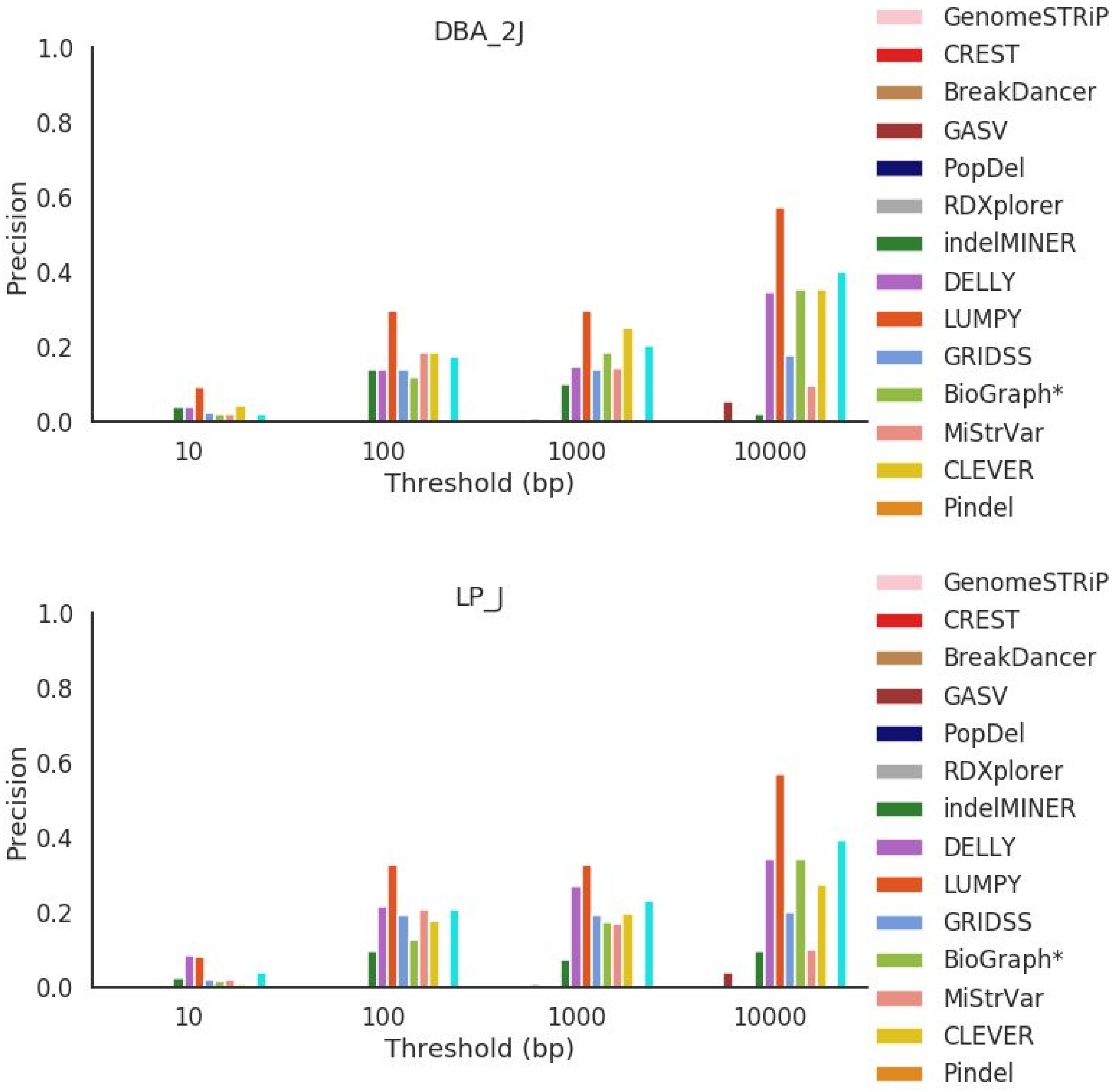
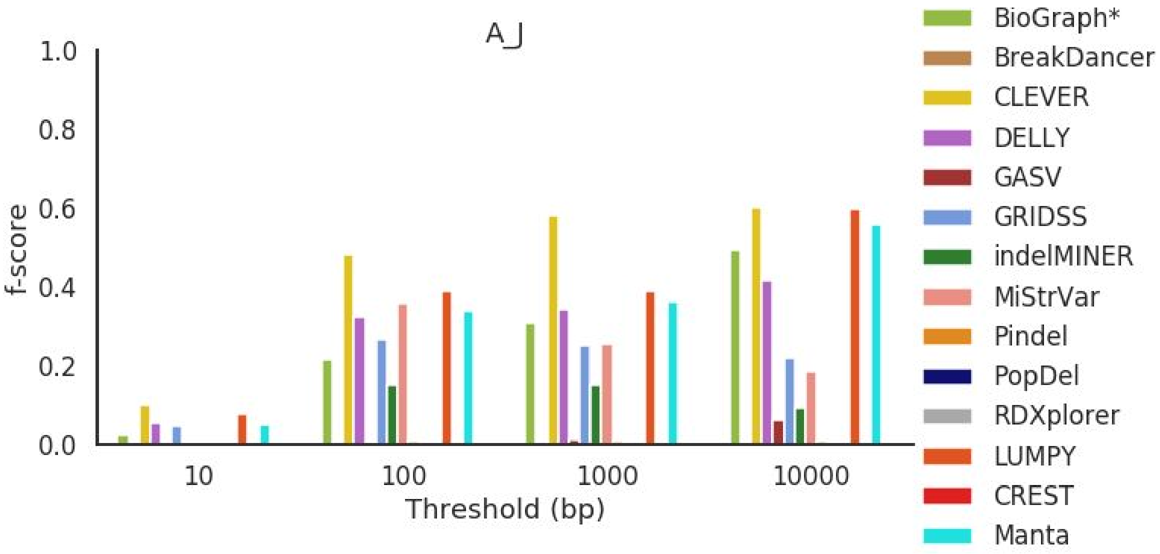
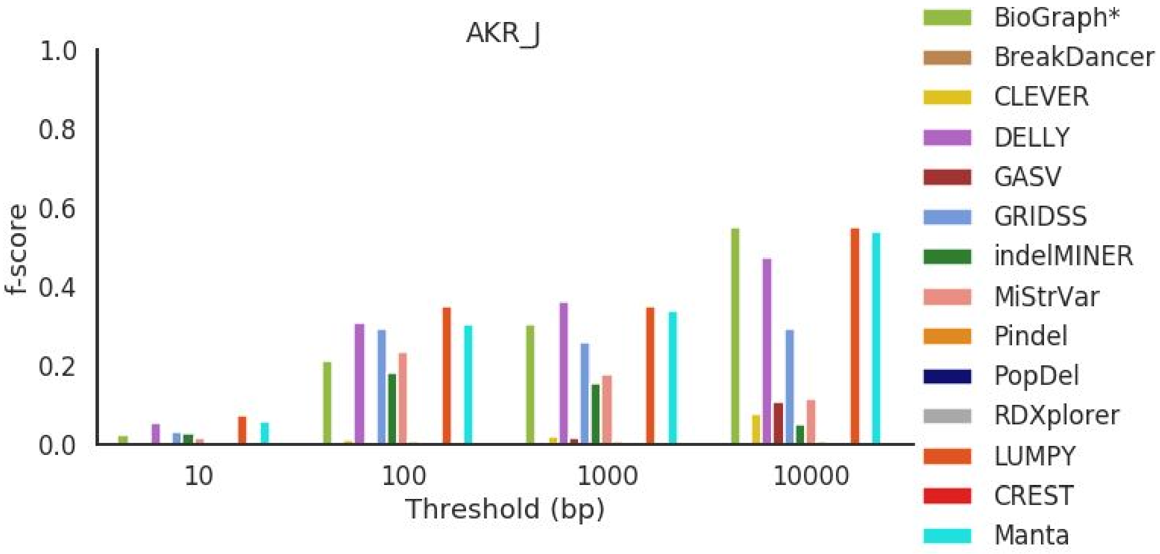
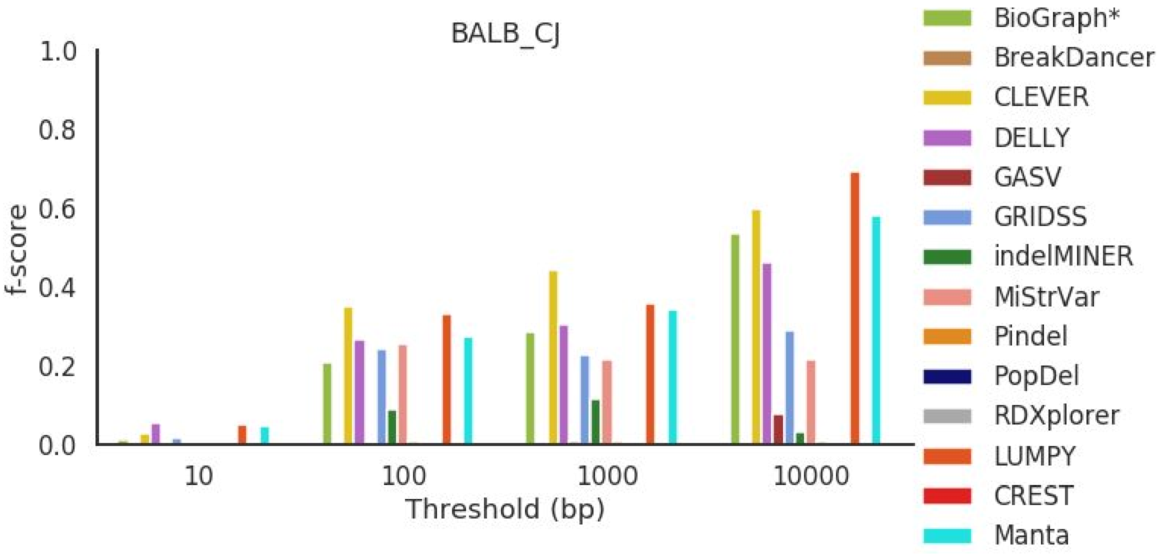
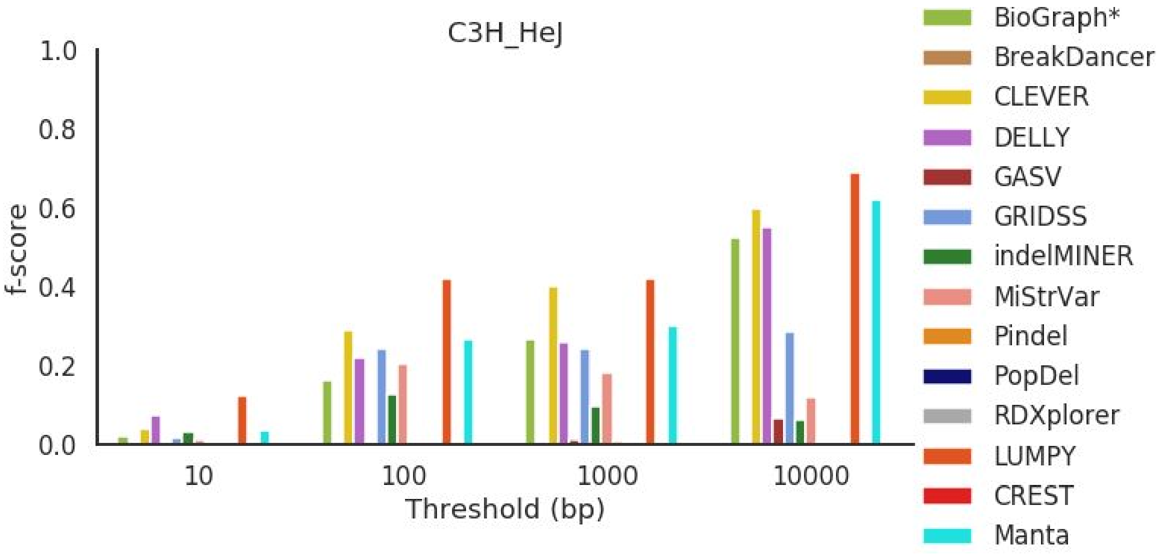


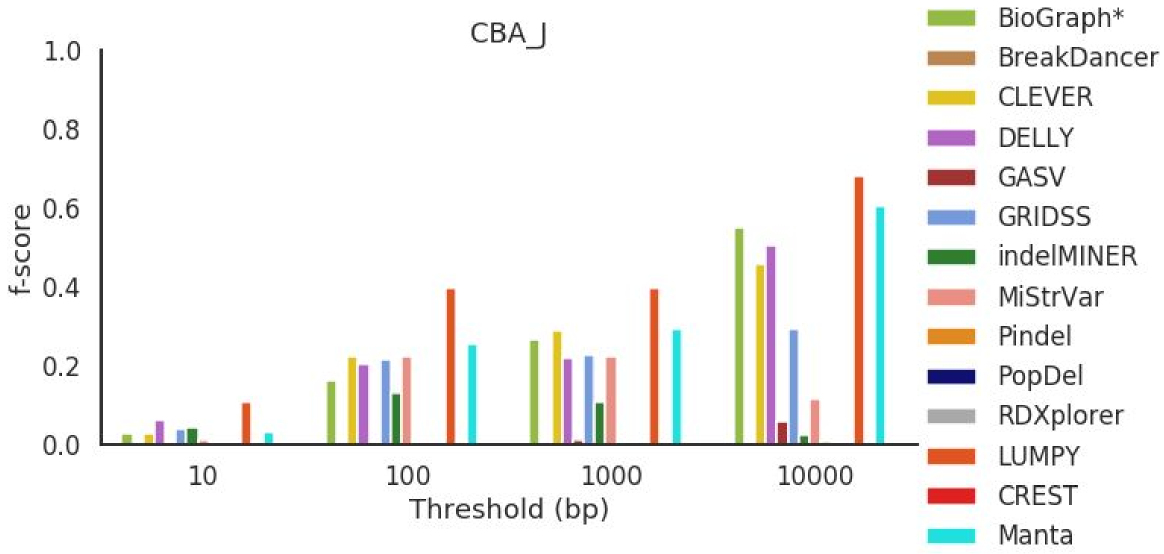
Figure S14. Precision across all mouse strains for deletions between 50 bp and 100 bp in length.


A_J
f-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


AKR_J
f-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


BALB_CJ
1.0
0.8
0.6
0.4
0.2
0.0
f-score
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


C3H_HeJ
1.0
0.8
0.6
0.4
0.2
0.0
f-score
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


CBA_J
f-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta

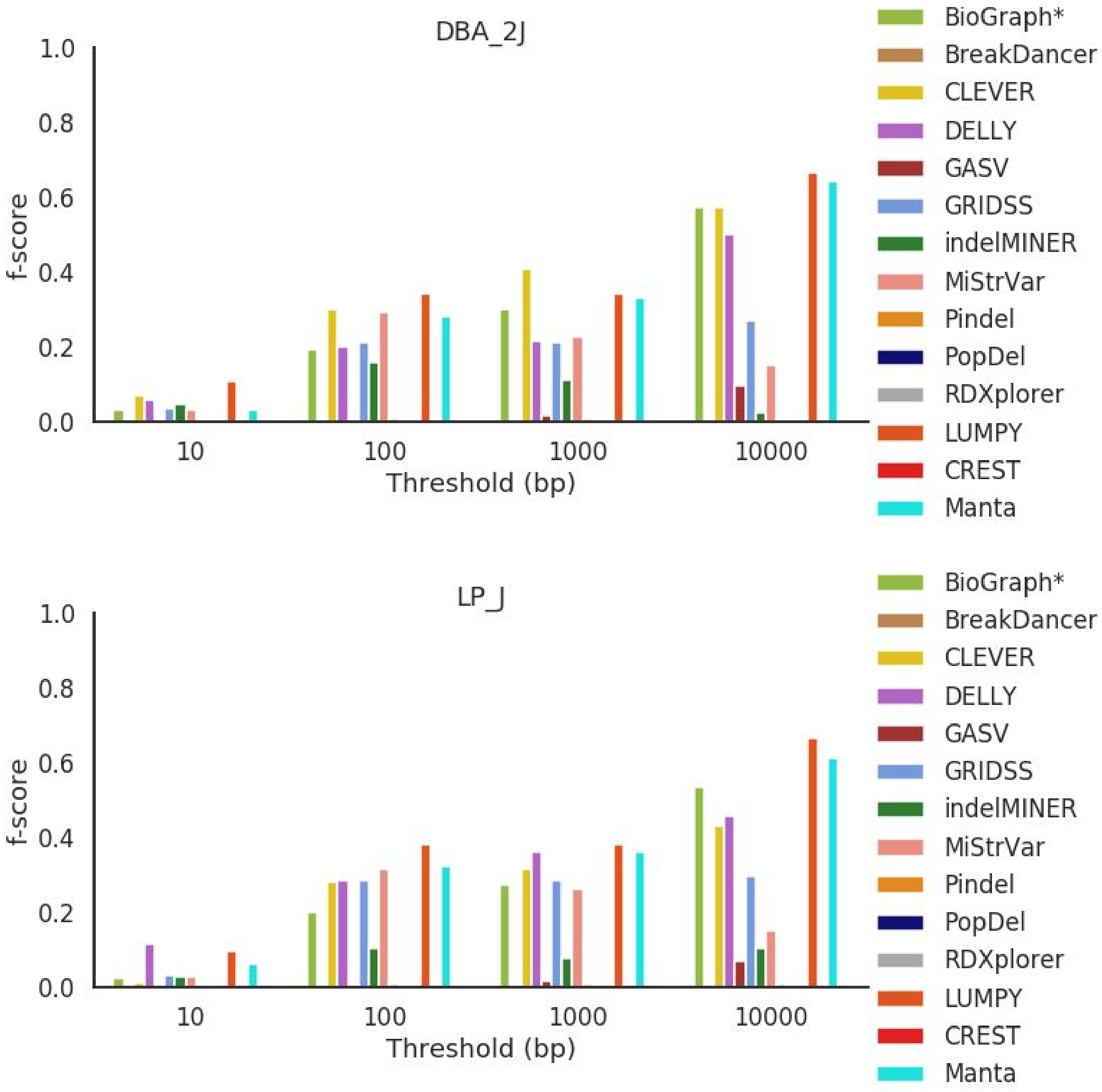
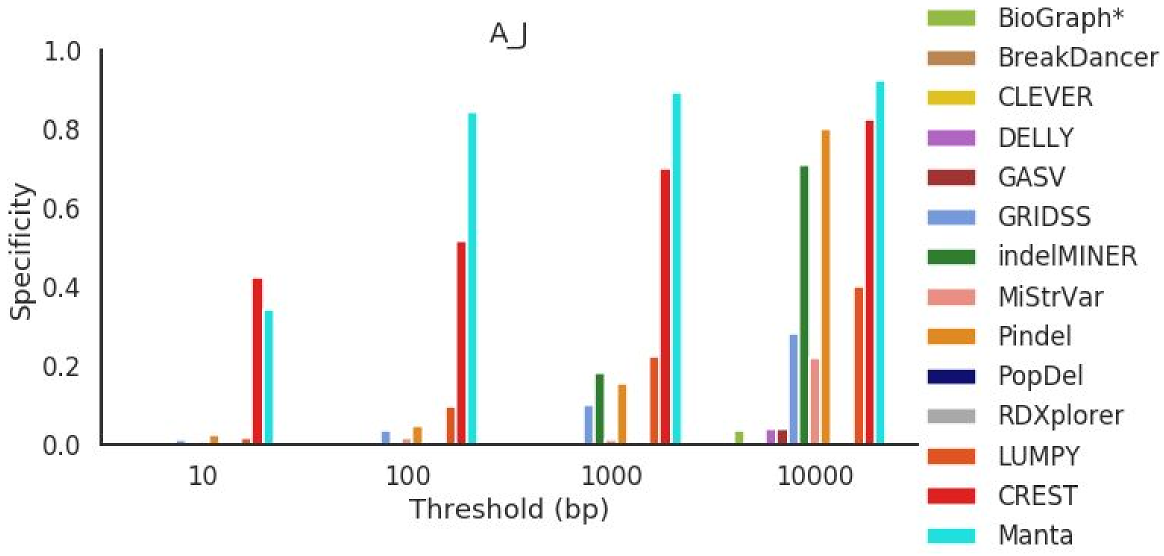
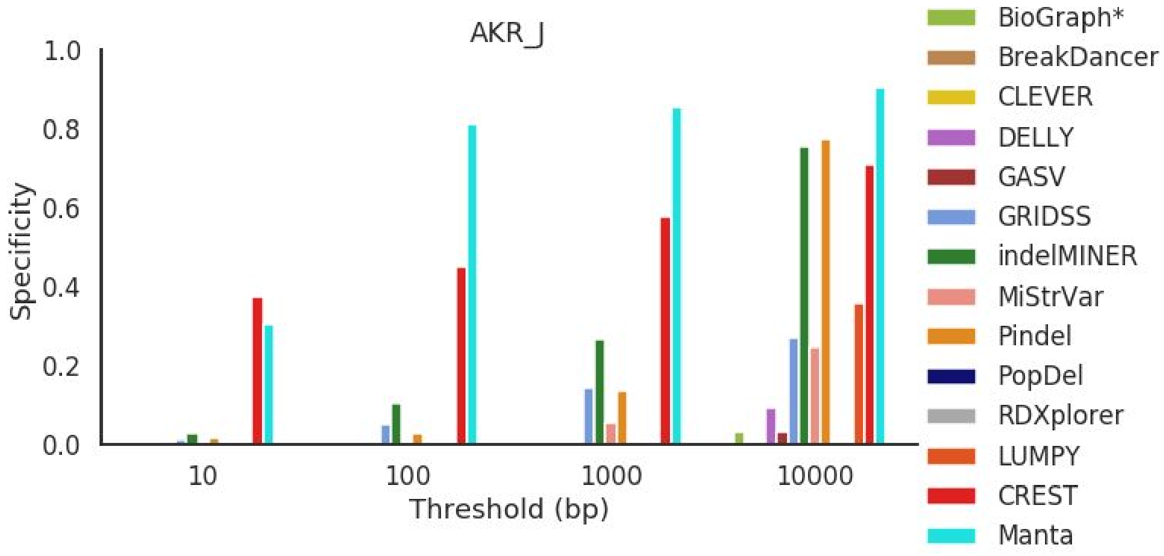
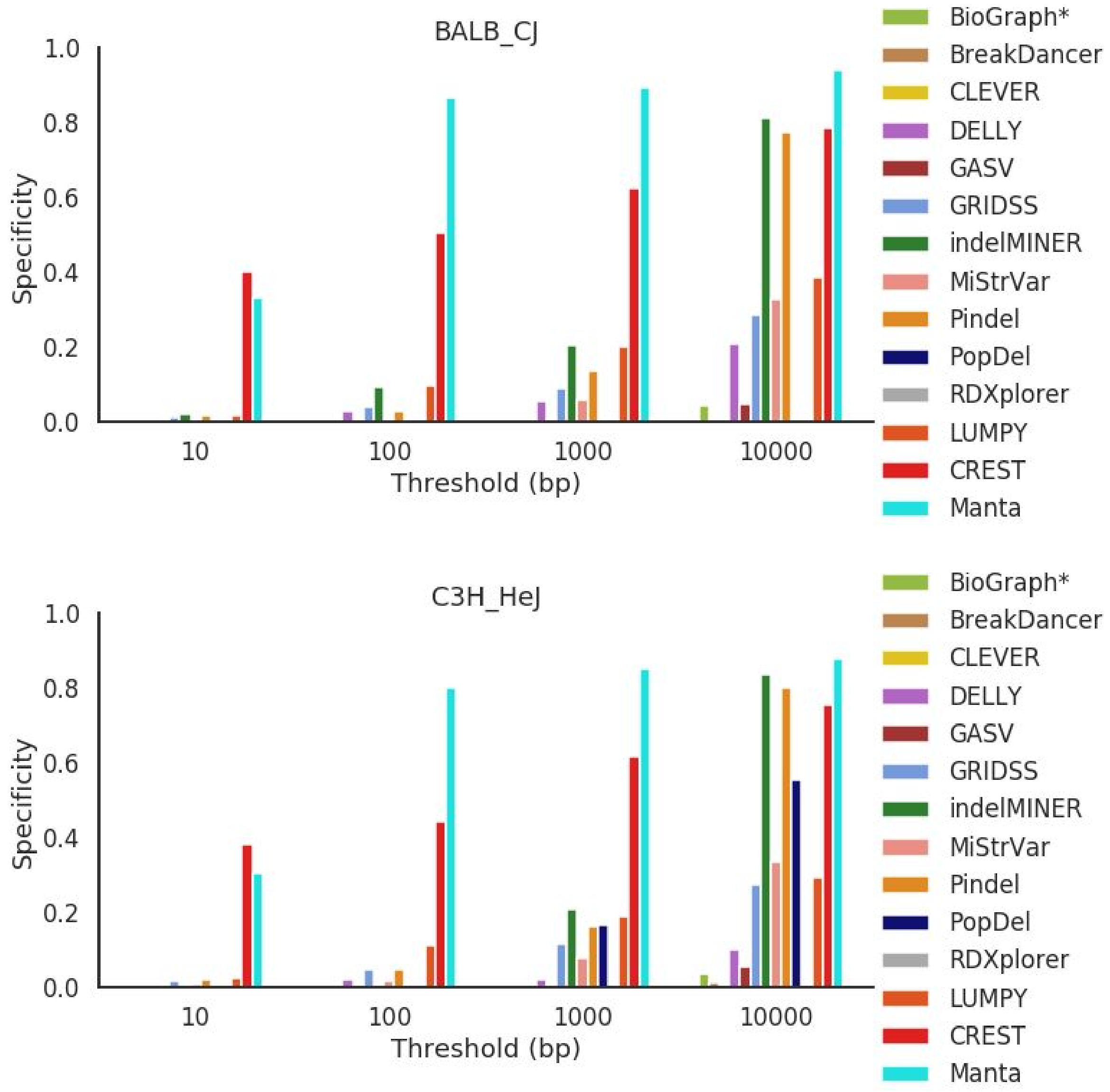
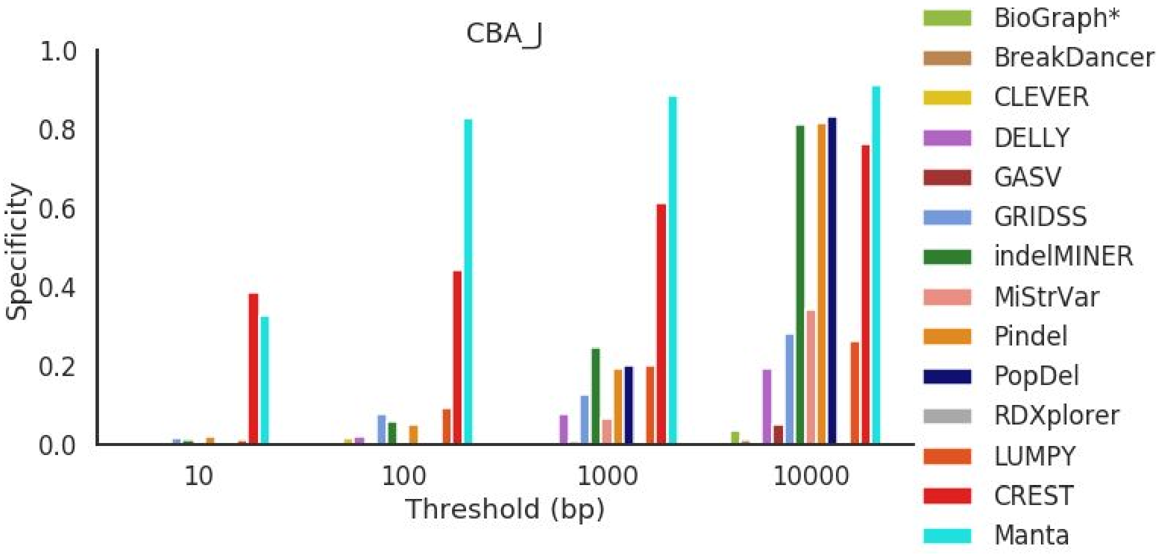


Figure S15. F-score across all mouse strains for deletions between 50 bp and 100 bp in length.


A_J
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


AKR_J
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


BALB_CJ
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta
C3H_HeJ
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


CBA_J
1.0
0.8
0.6
0.4
0.2
0.0
Specificity
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta

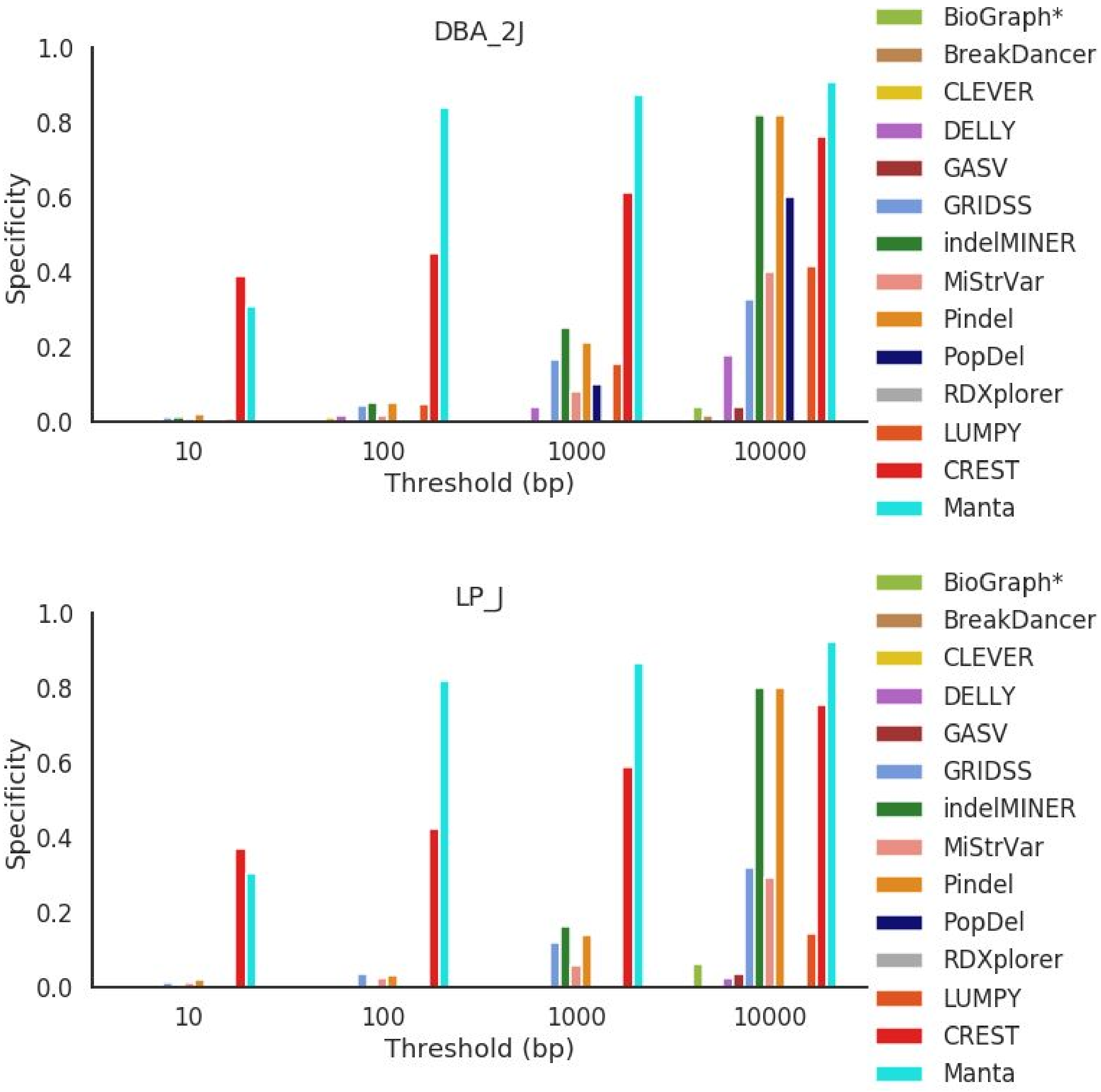
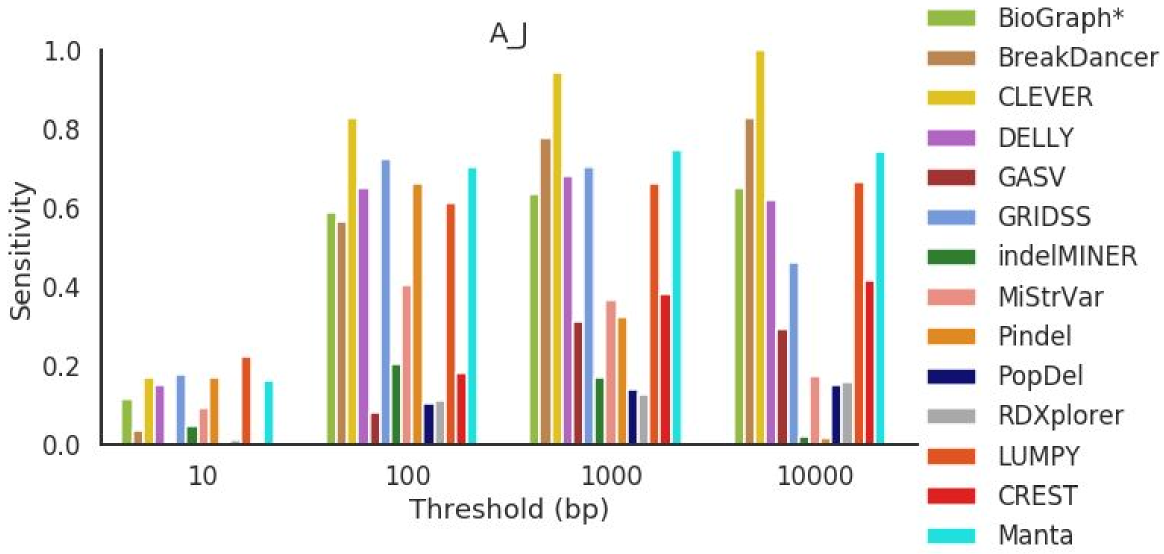
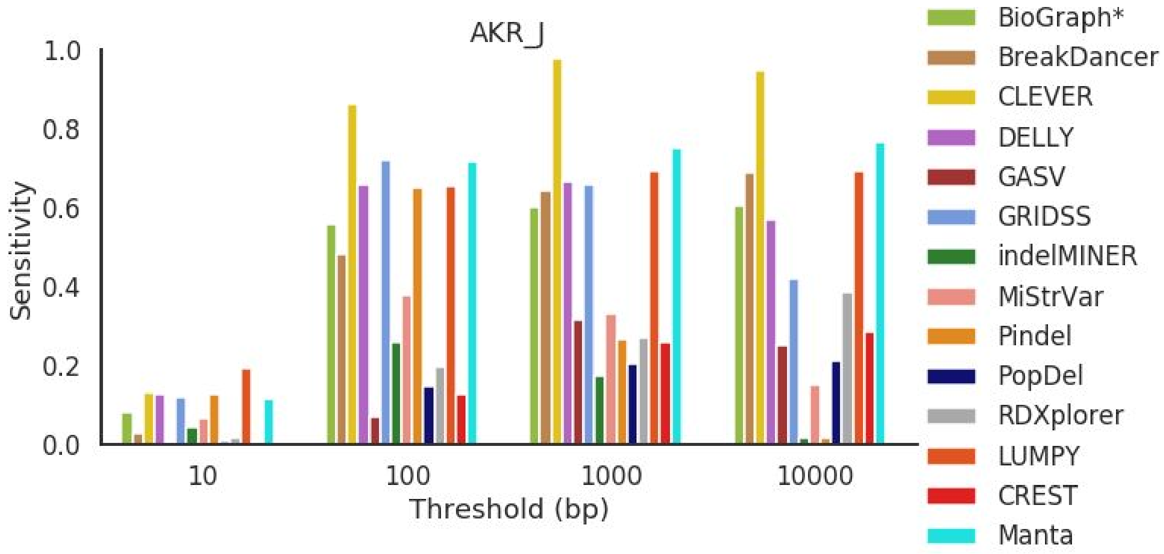
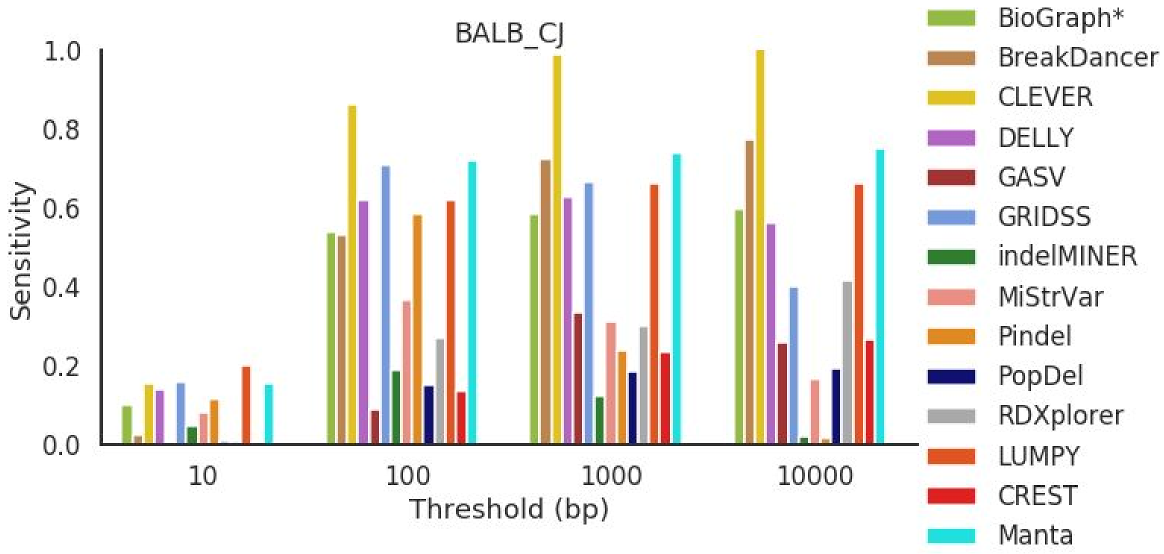
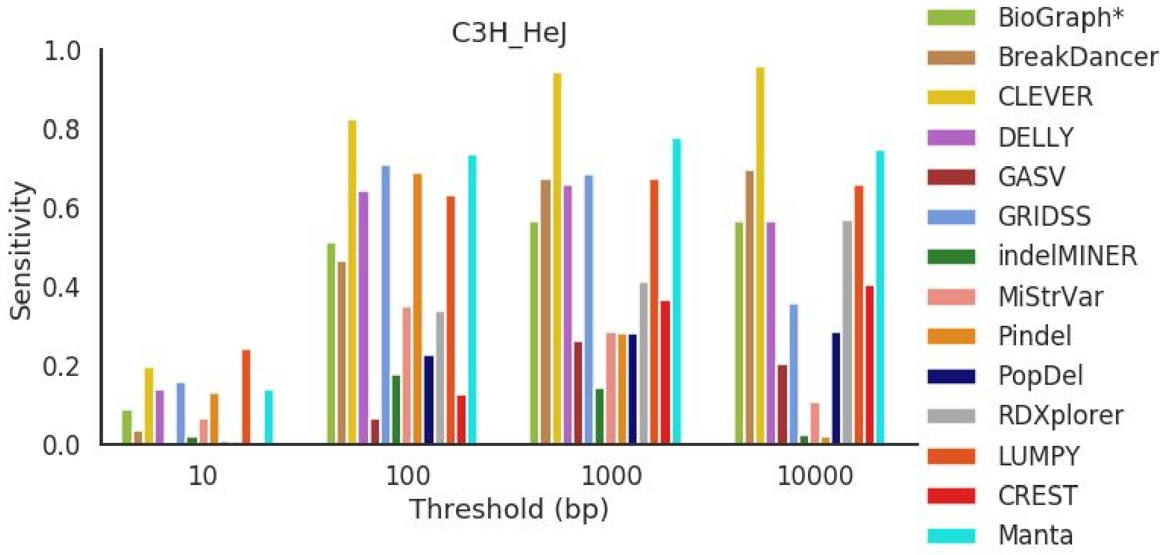


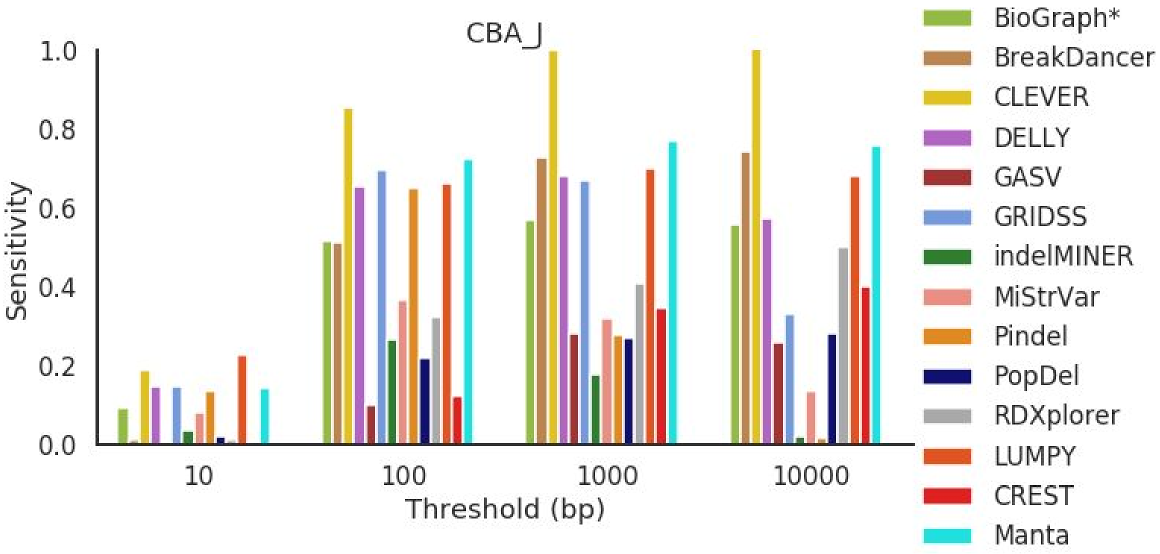
Figure S16. Specificity across all mouse strains for deletions between 100 bp and 500 bp in length.


A_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


AKR_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


BALB_CJ
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


C3H_HeJ
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


CBA_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta

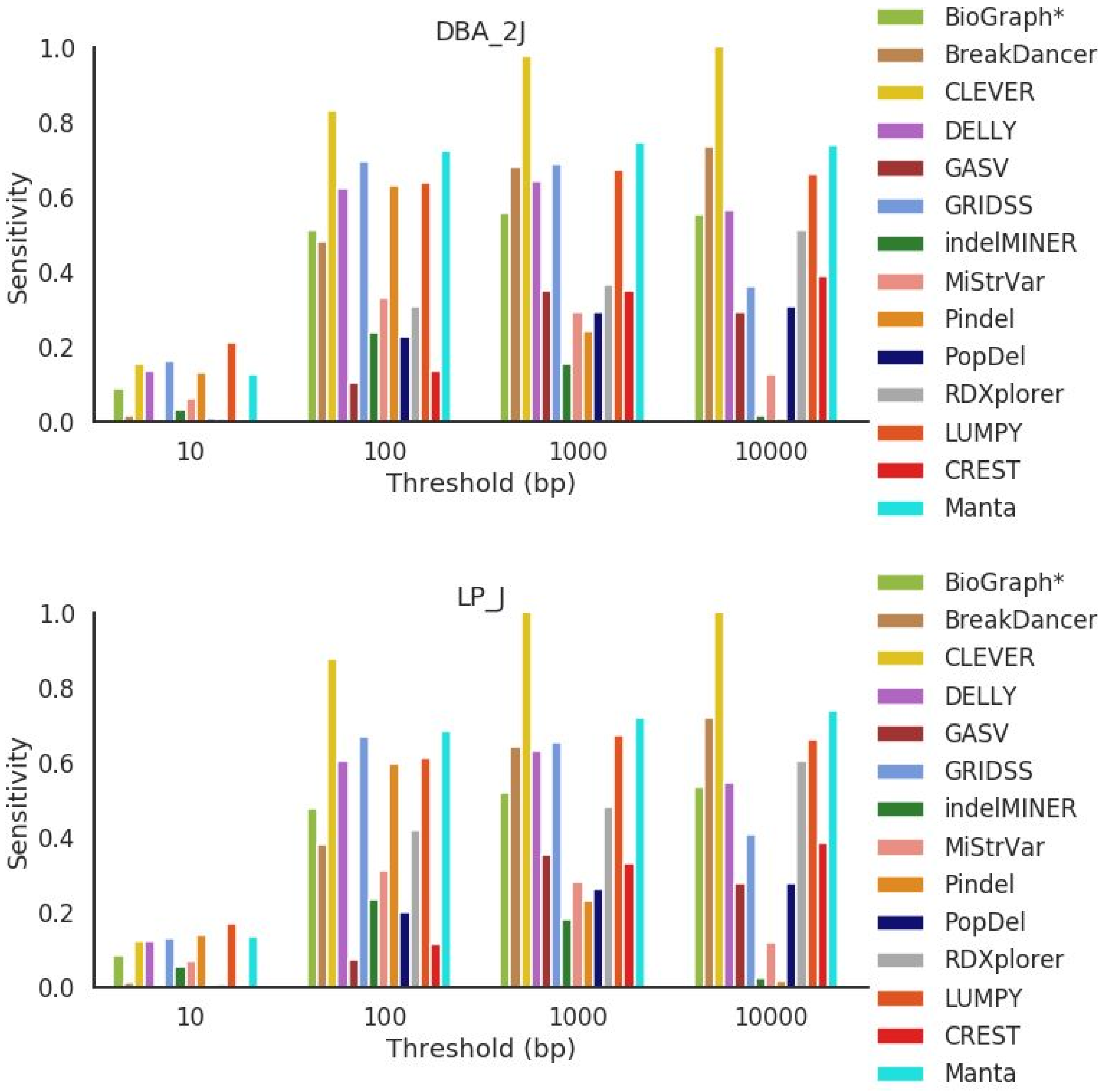
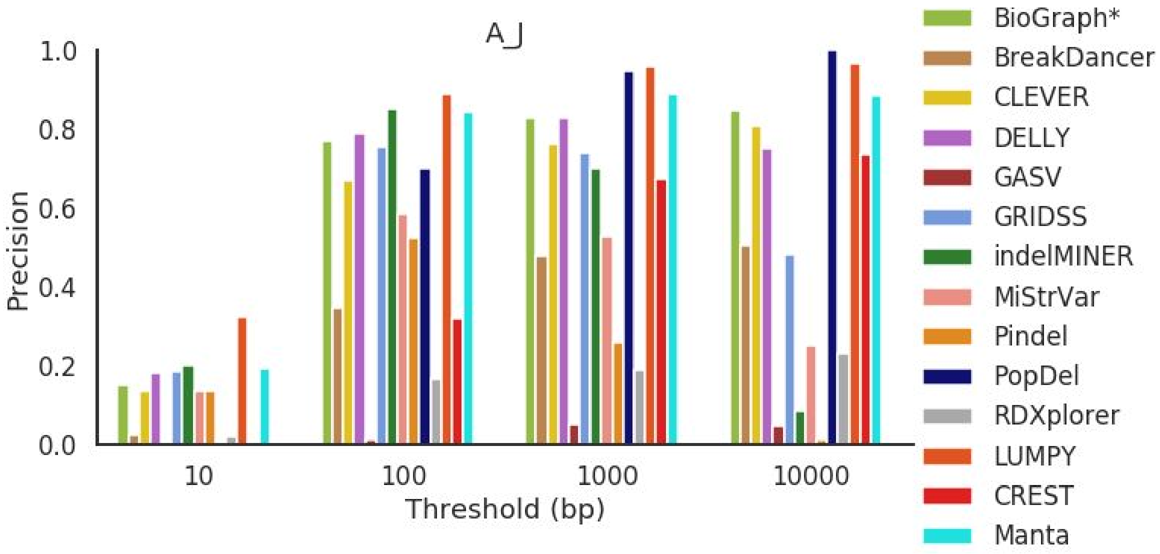
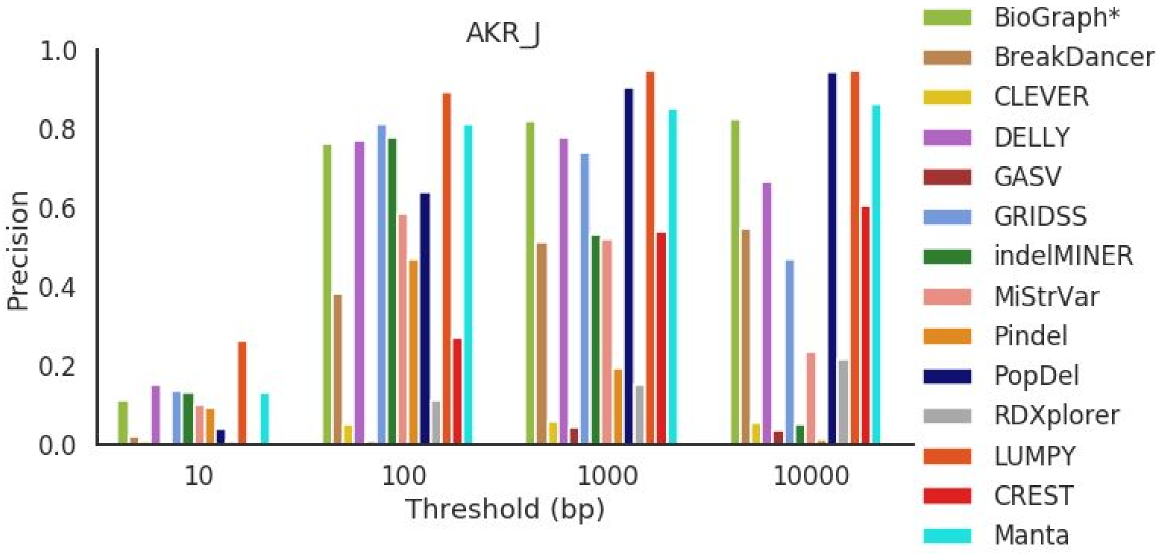
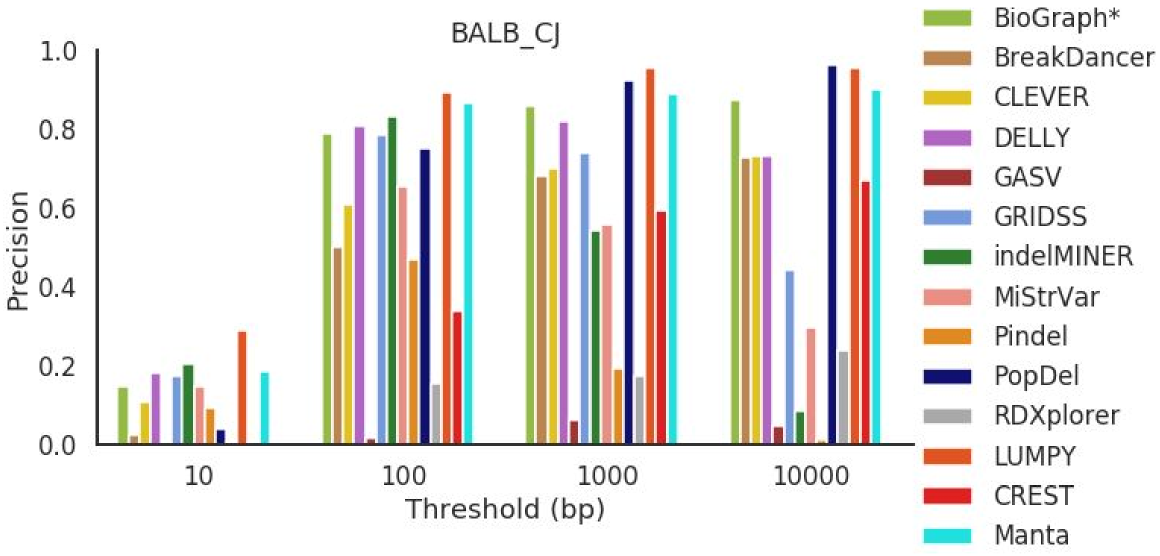
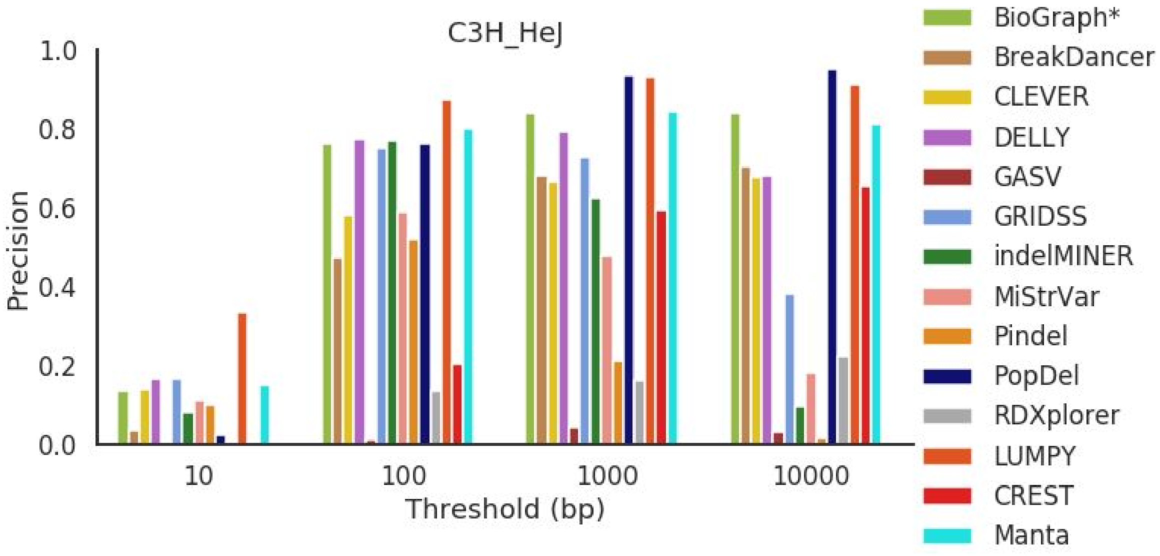


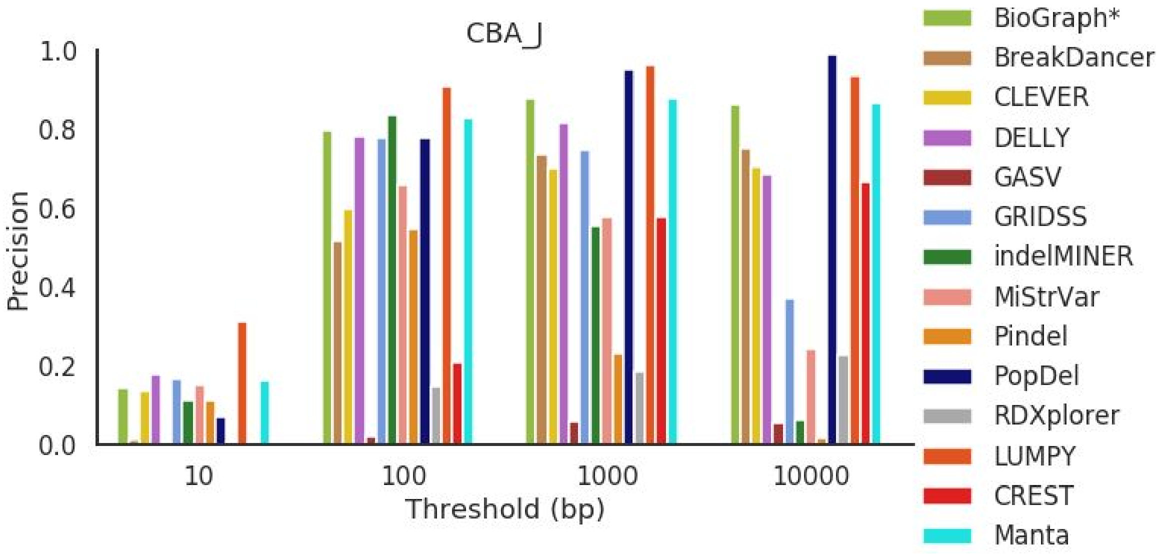
Figure S17. Sensitivity across all mouse strains for deletions between 50 bp and 100 bp in length.


A_J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


AKR_J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


BALB_CJ
1.0
0.8
0.6
0.4
0.2
0.0
Precision
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


C3H_HeJ
1.0
0.8
0.6
0.4
0.2
0.0
Precision
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


CBA_J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta

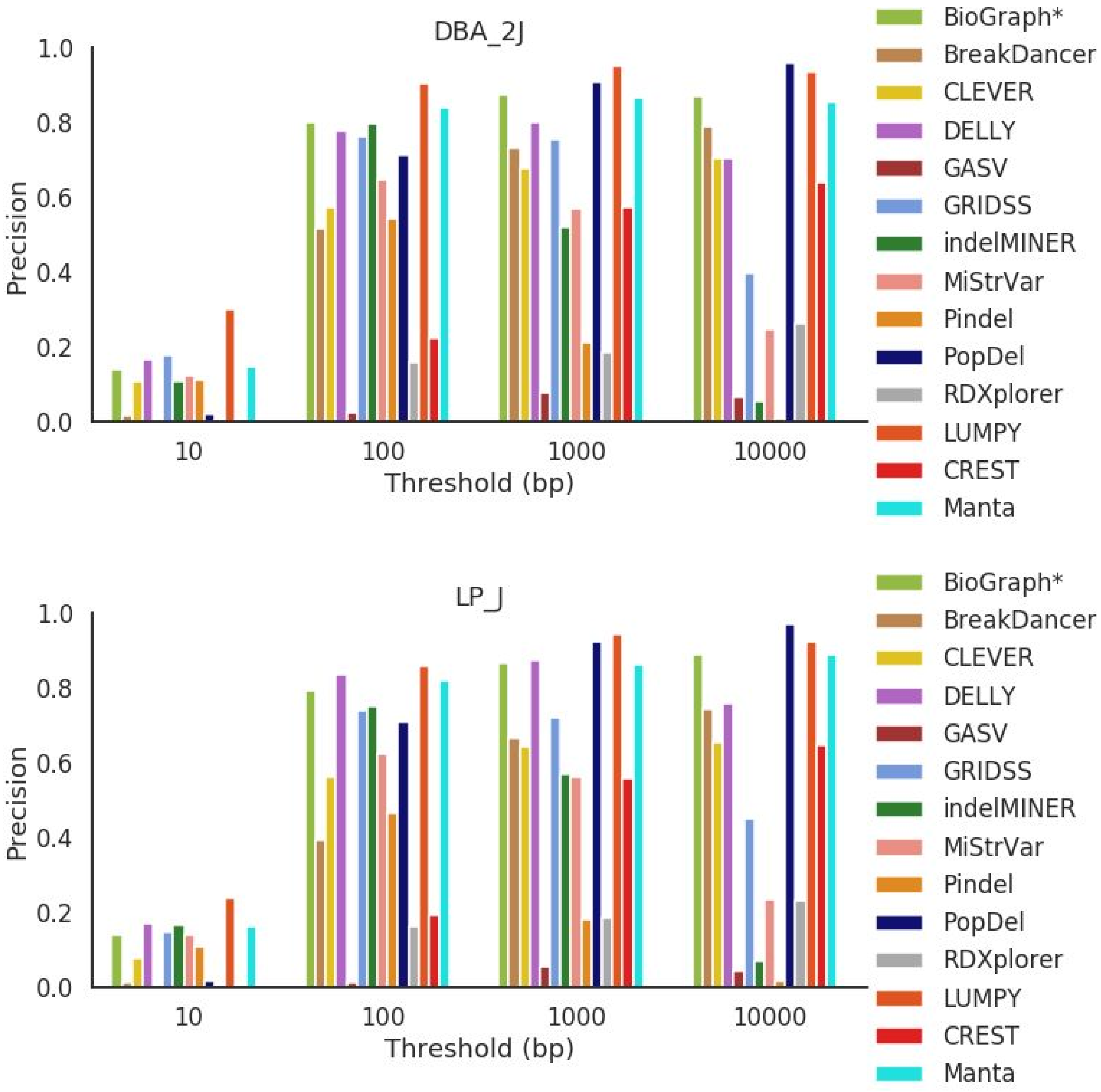
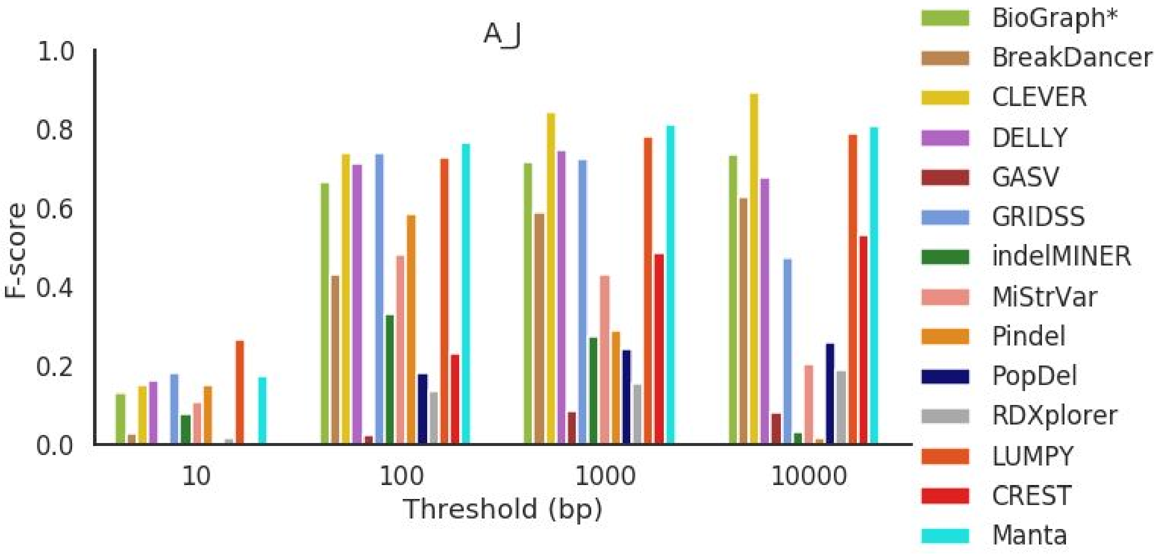
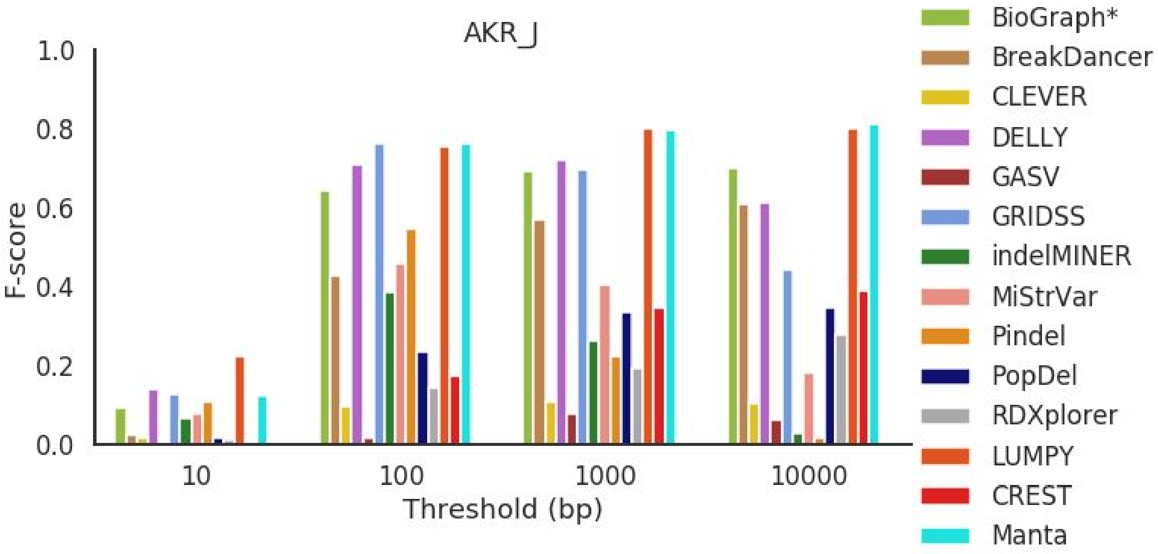
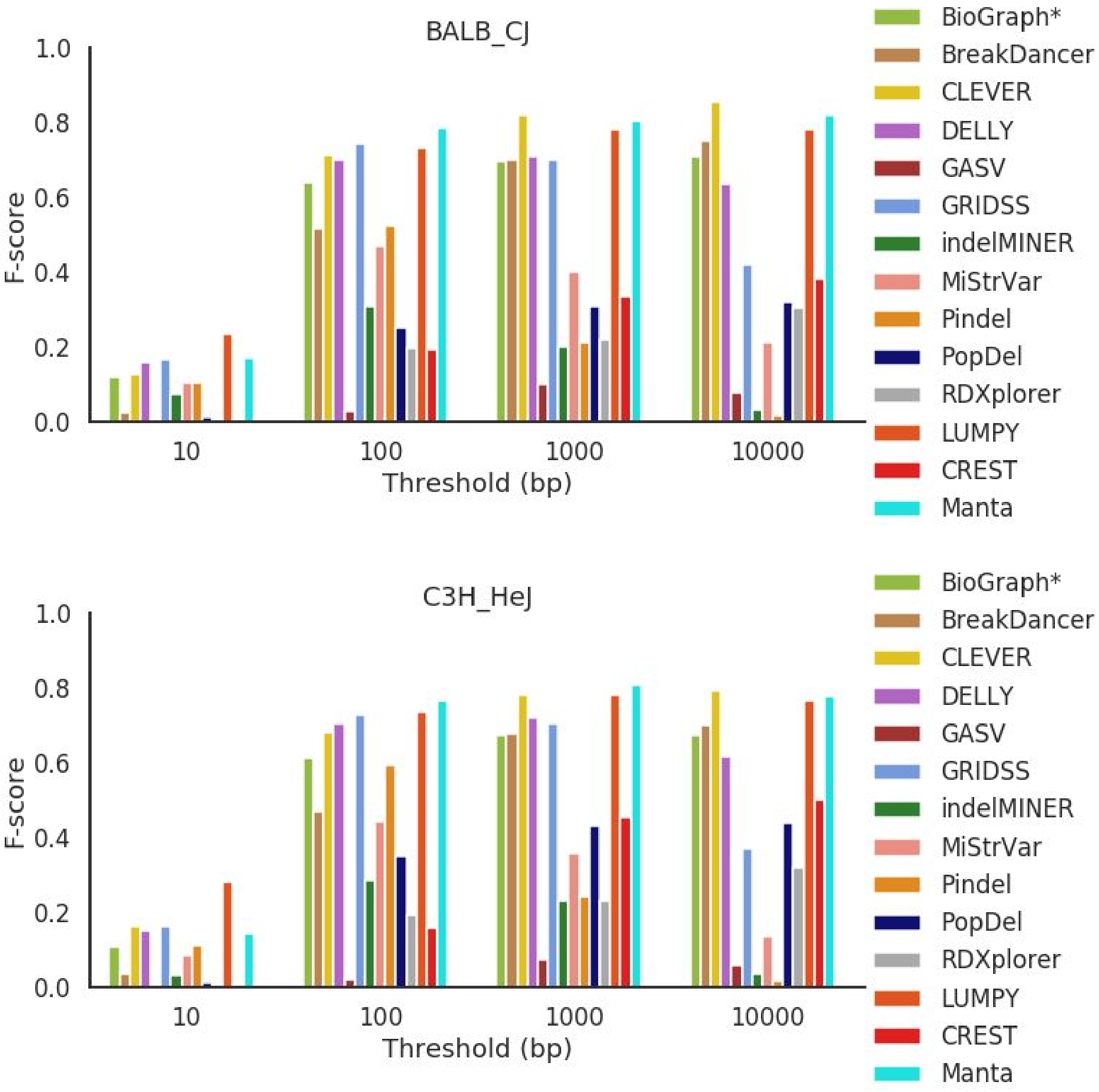
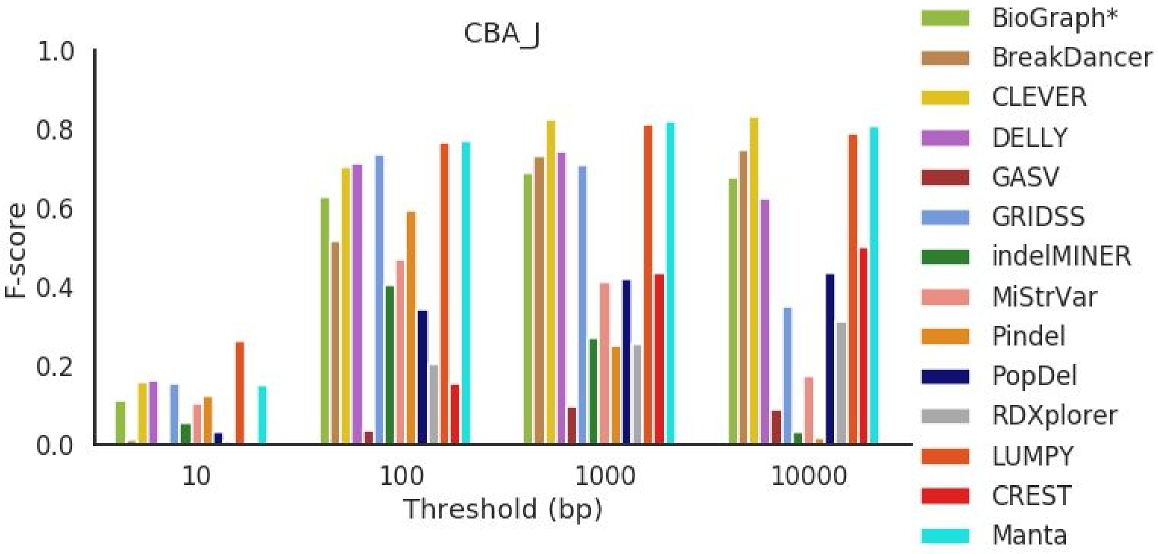


Figure S18. Precision across all mouse strains for deletions between 100 bp and 500 bp in length.


A_J
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


AKR_J
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


BALB_CJ
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta
C3H_HeJ
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


CBA_J
1.0
0.8
0.6
0.4
0.2
0.0
F-score
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta

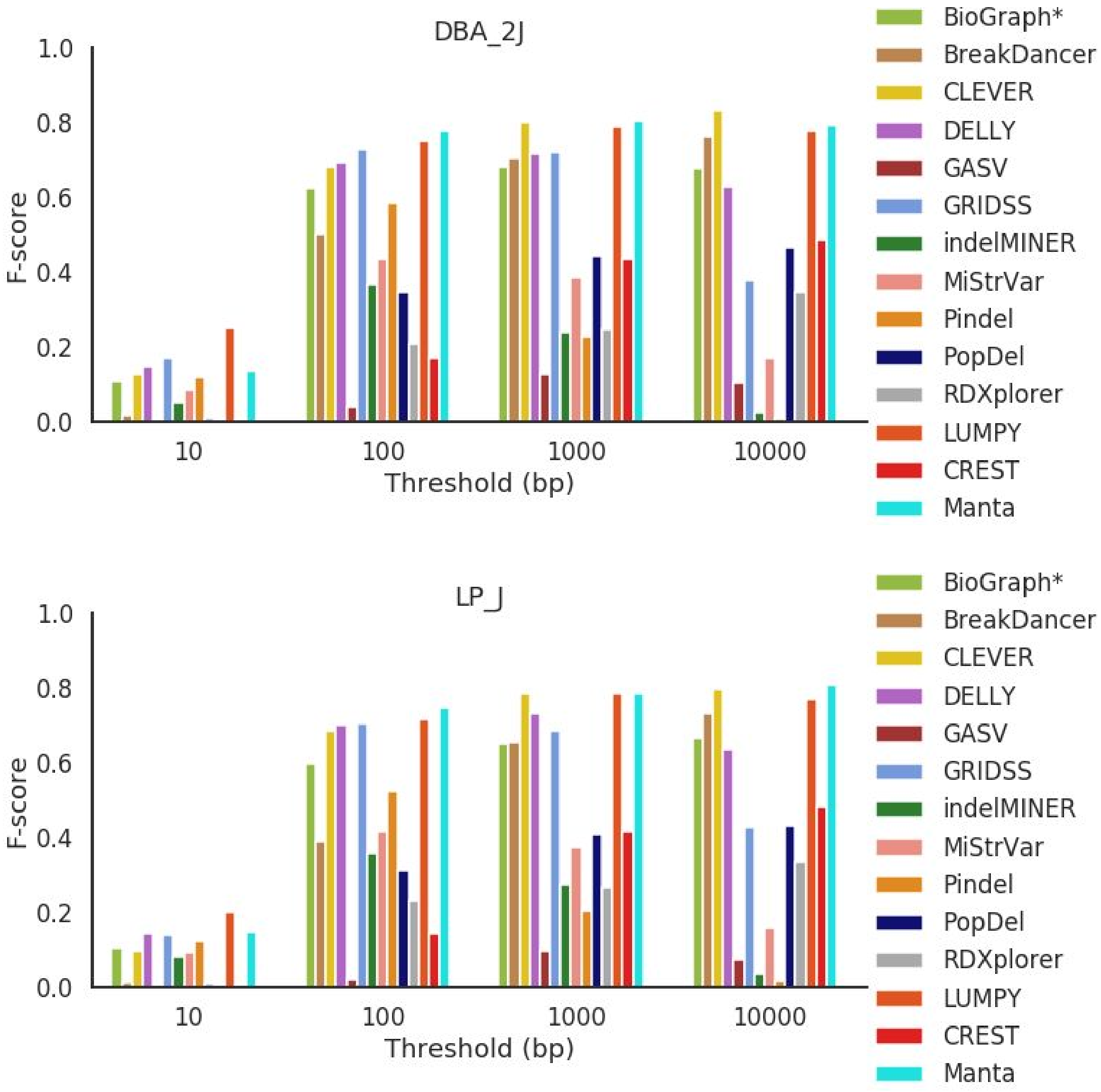
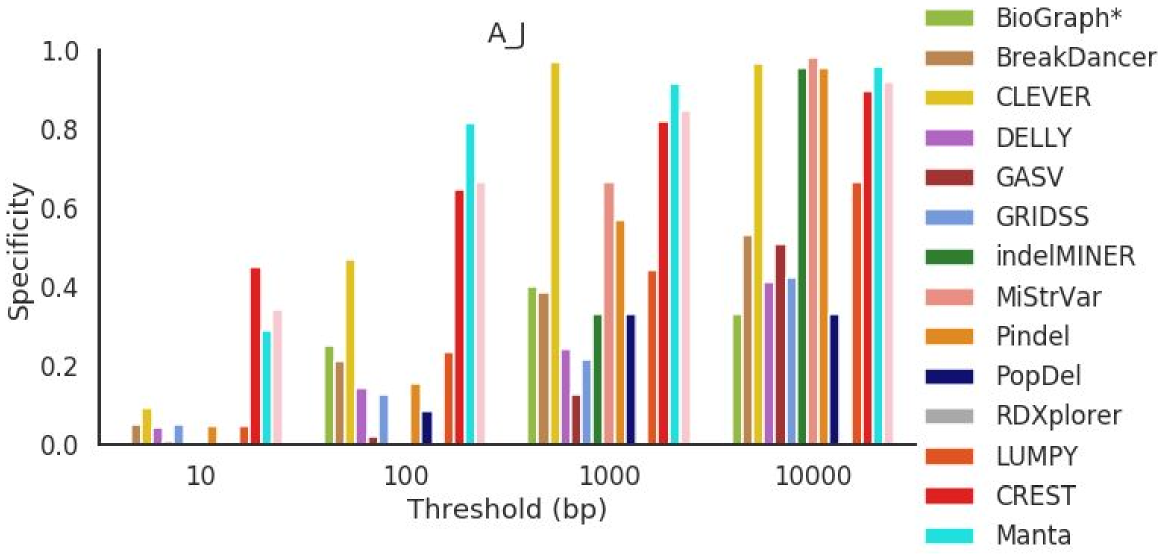
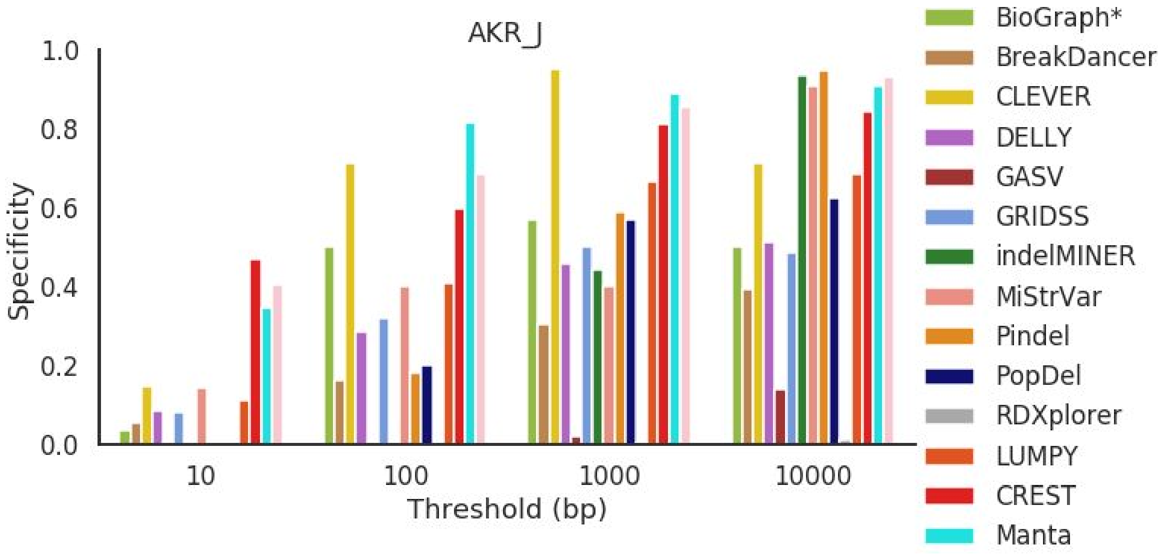
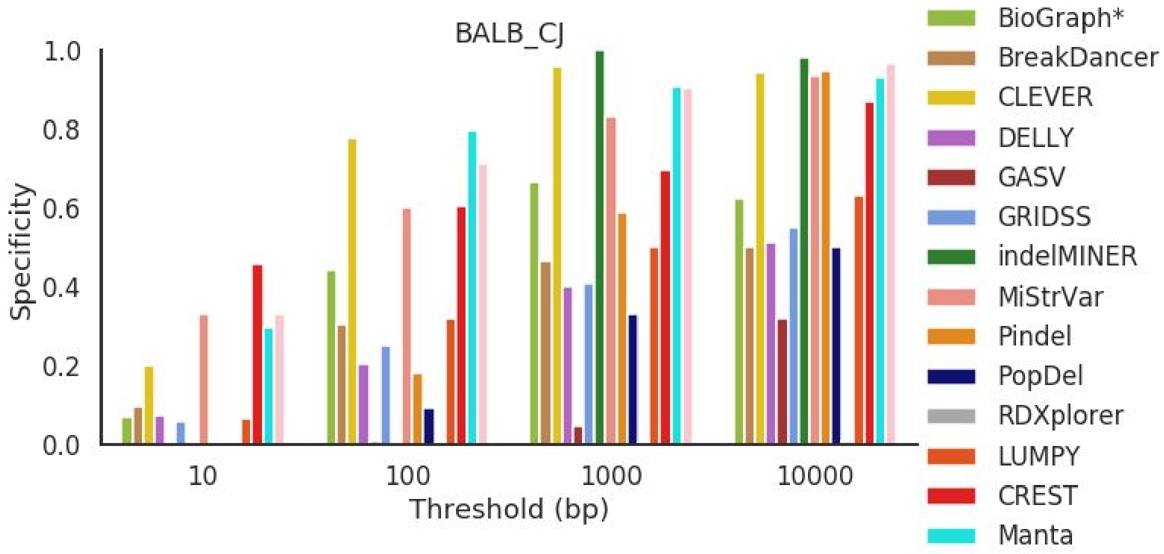
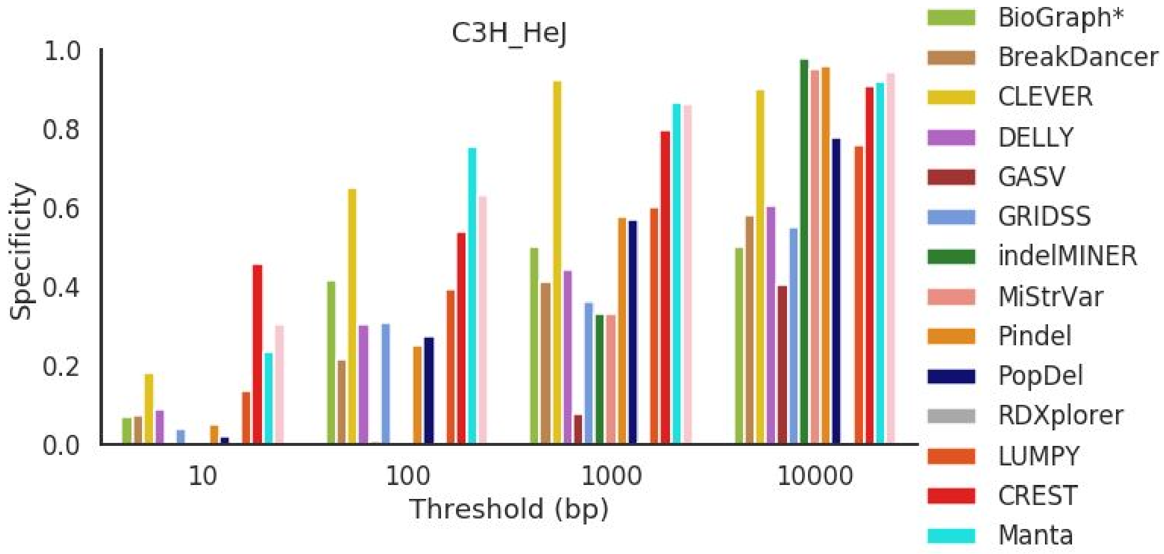


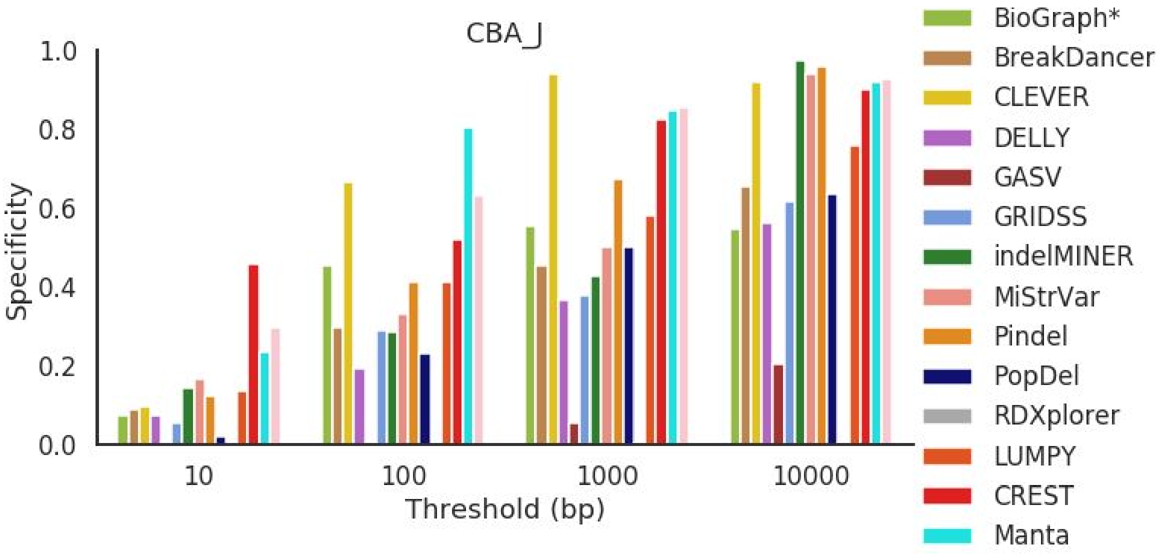
Figure S19. F-score across all mouse strains for deletions between 100 bp and 500 bp in length.


A_J
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


AKR_J
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


BALB_CJ
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


C3H_HeJ
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


CBA_J
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta

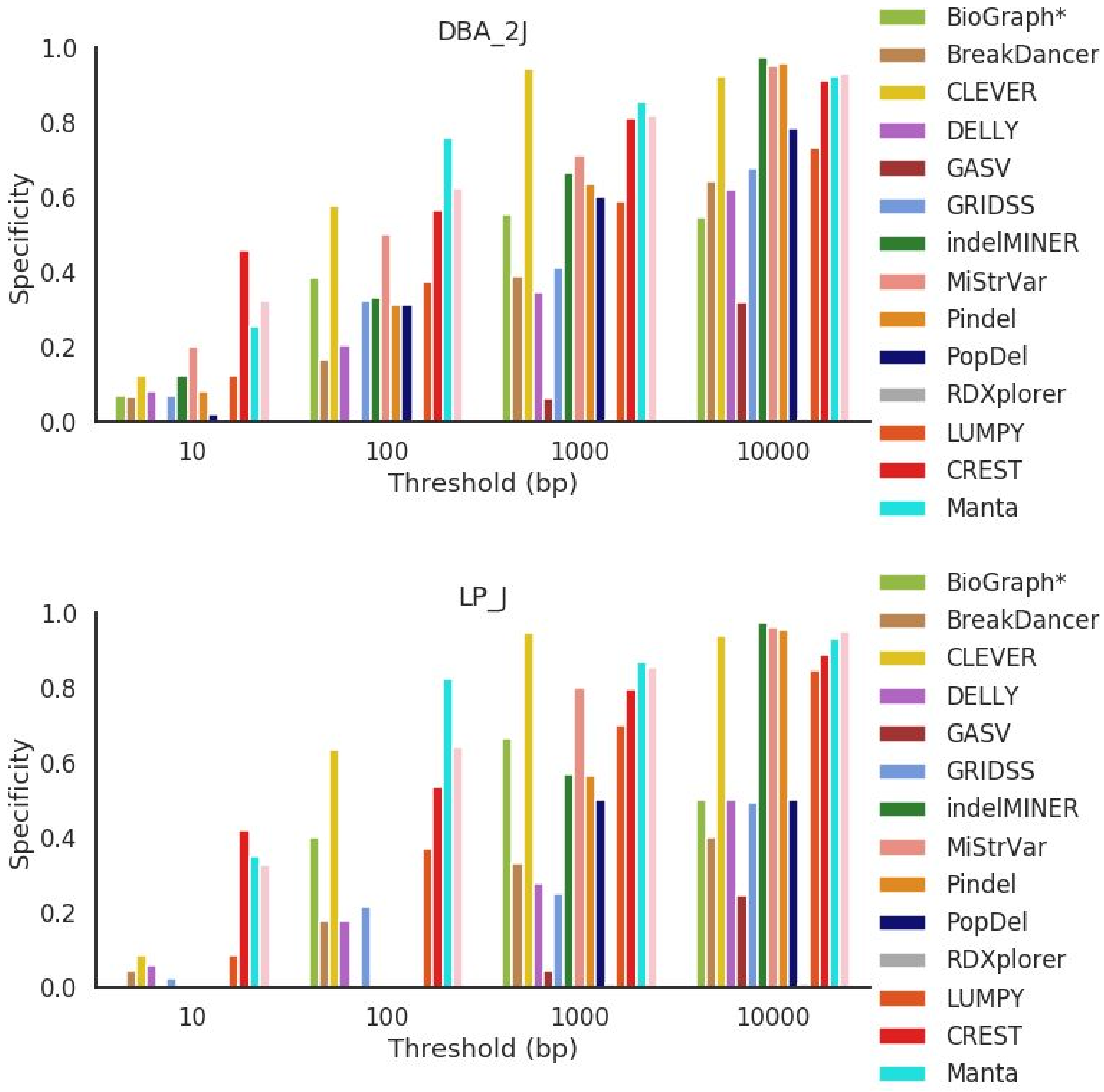
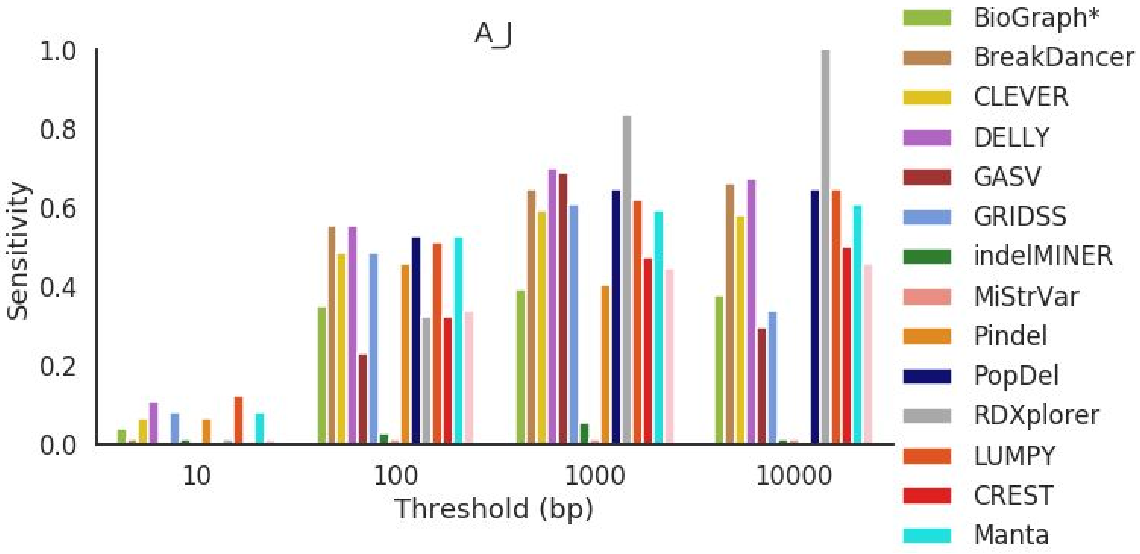
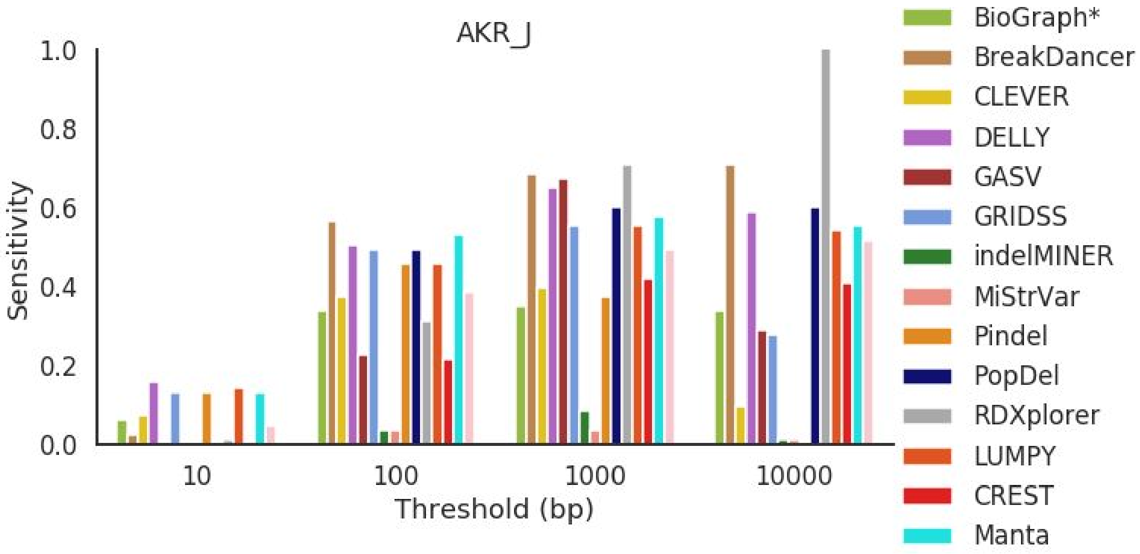
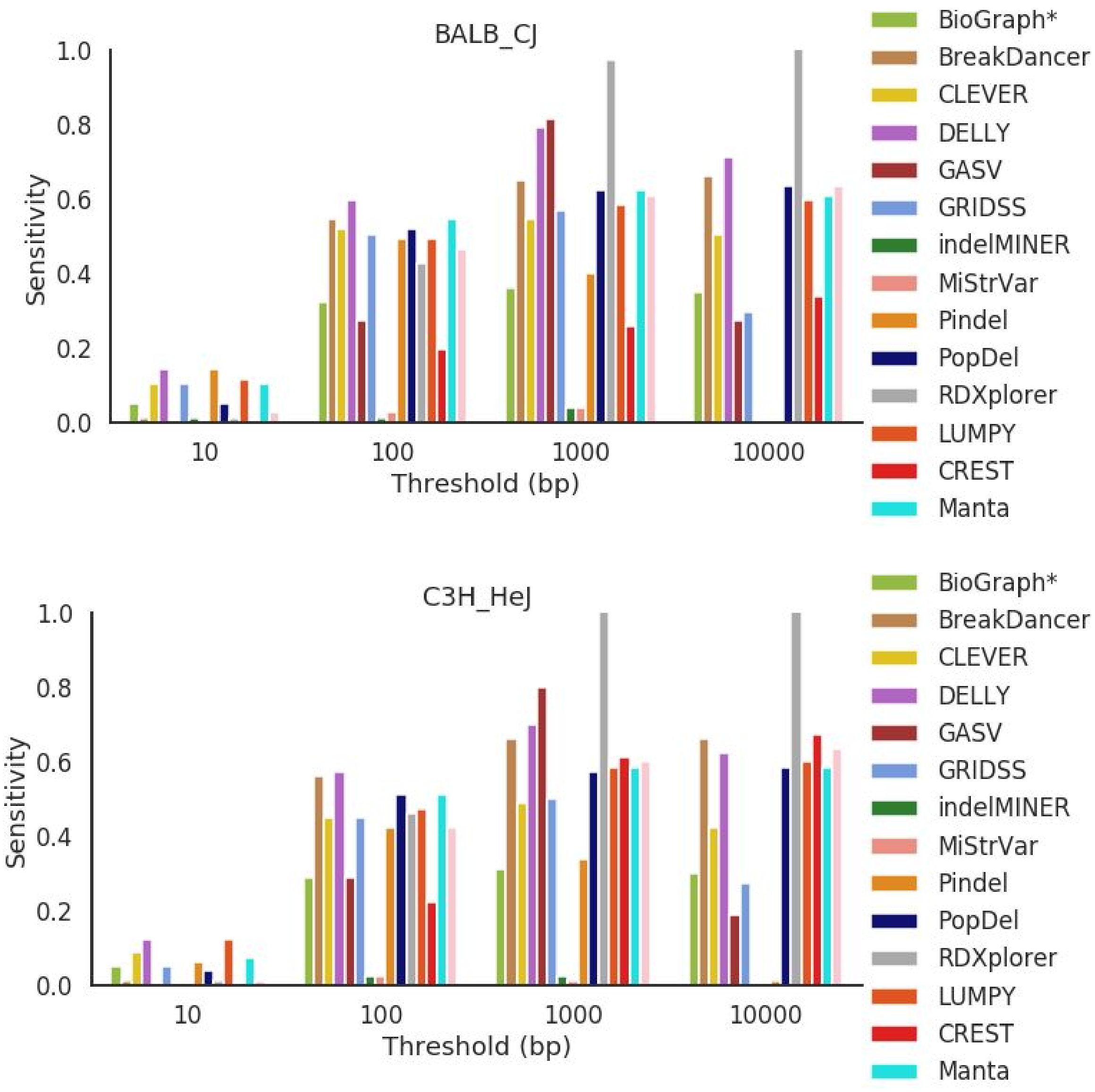
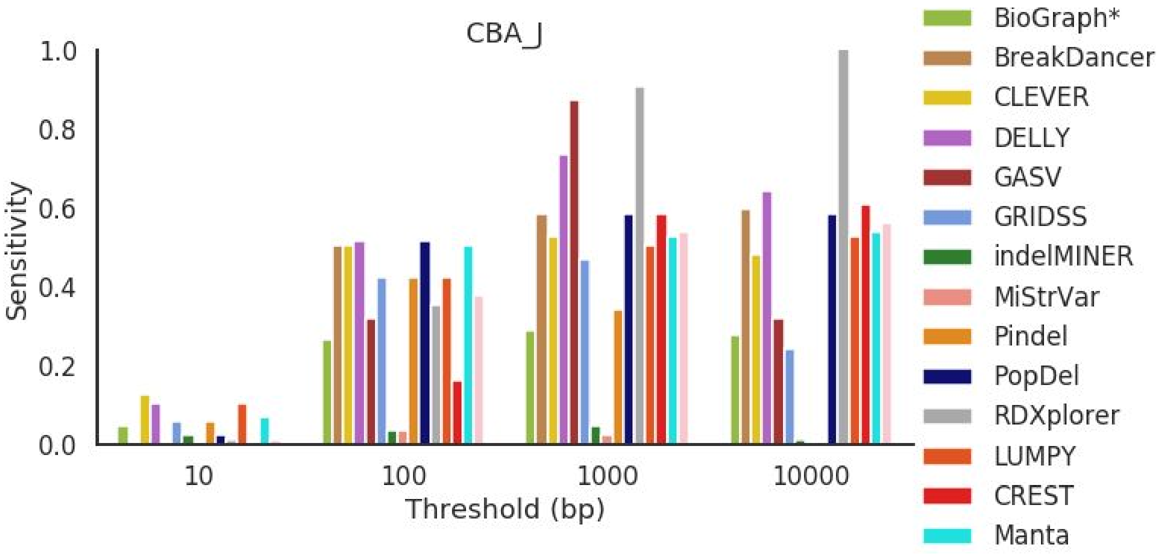


Figure S20. Specificity across all mouse strains for deletions between 500 bp and 1000 bp in length.


A_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


AKR_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


BALB_CJ
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta
C3H_HeJ
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


CBA_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta

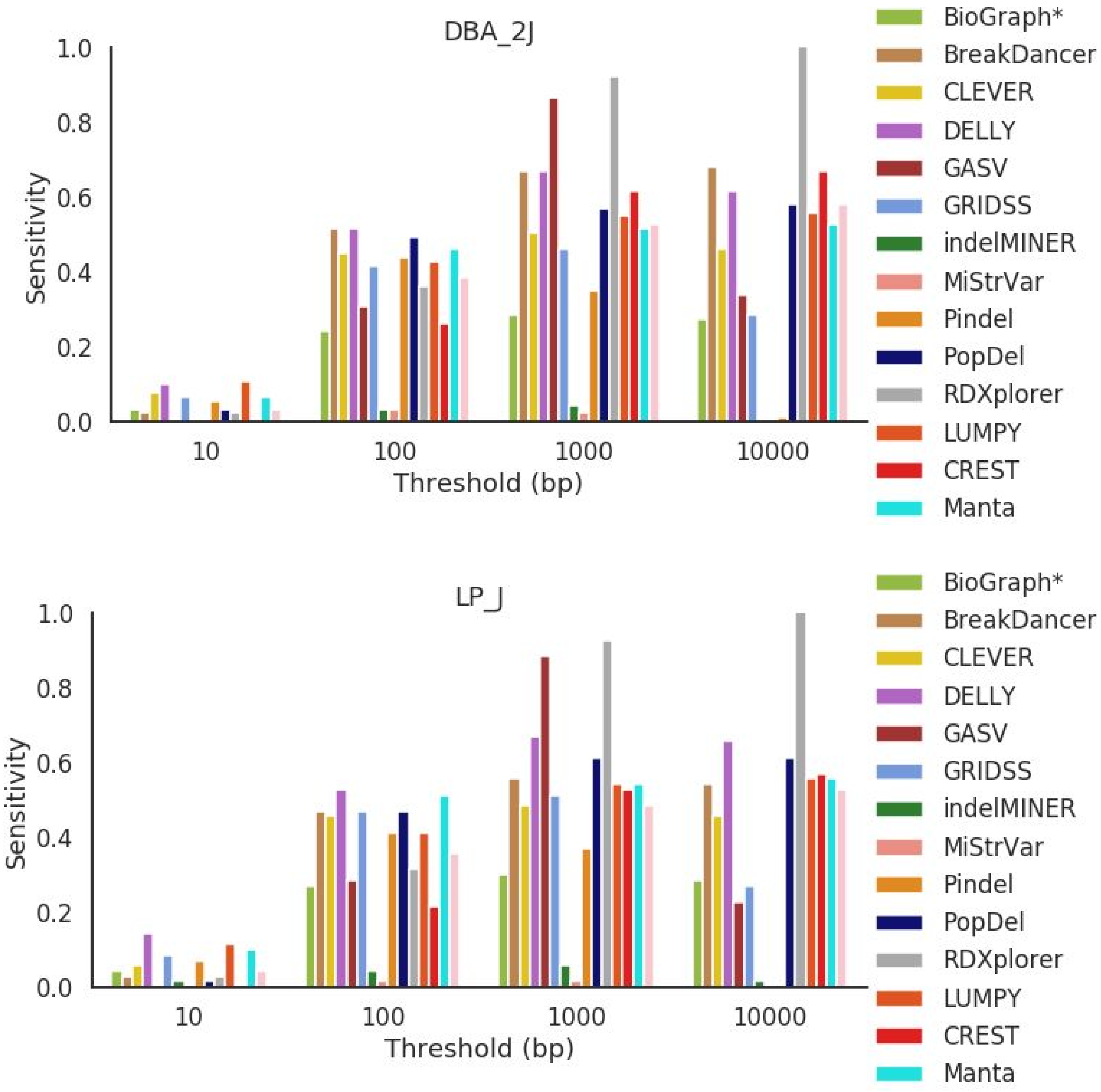
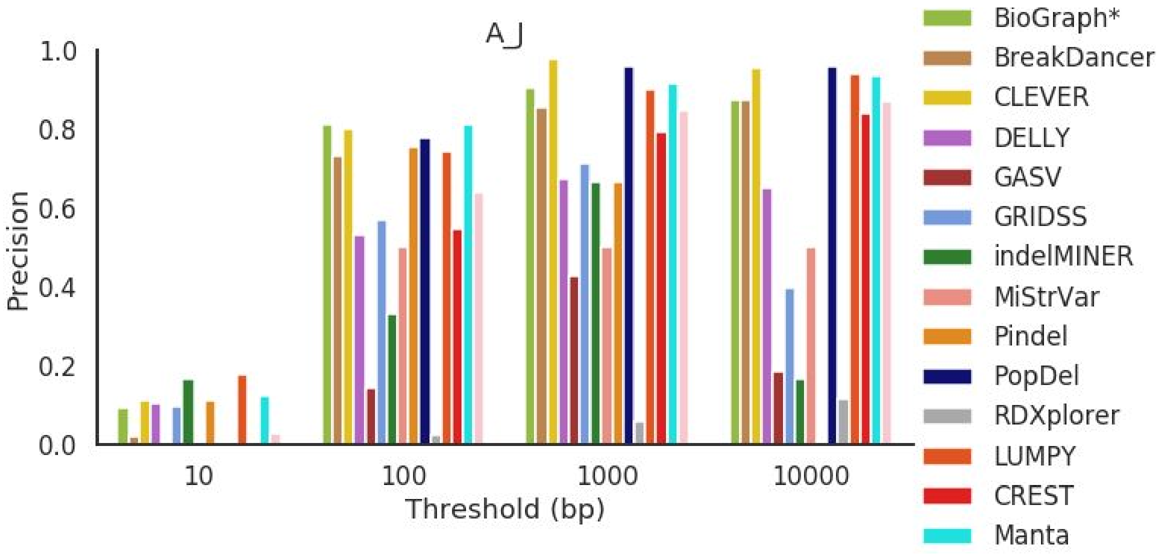
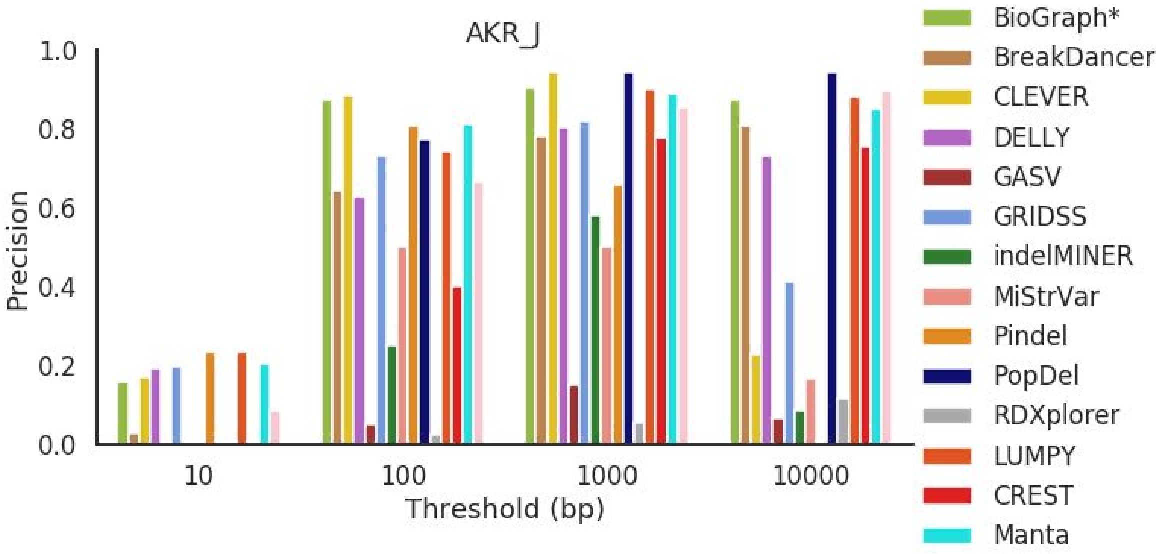
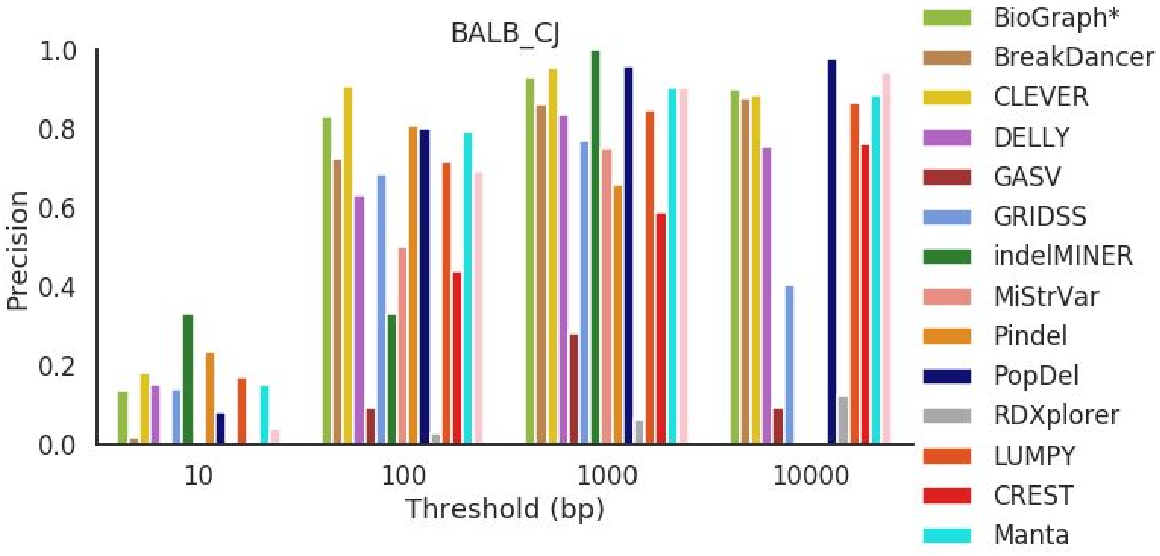
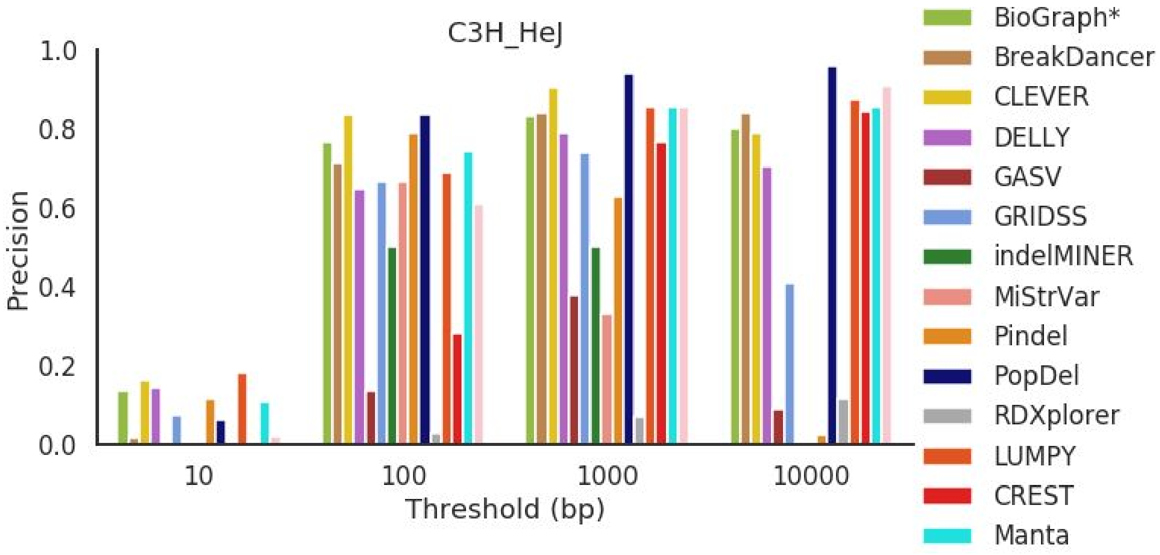


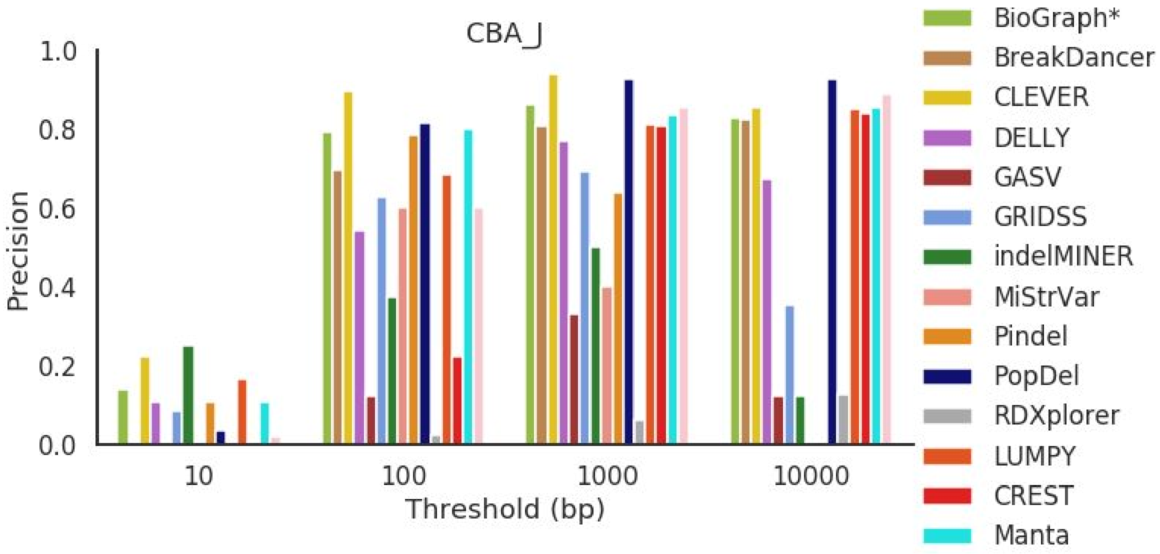
Figure S21. Sensitivity across all mouse strains for deletions between 500 bp and 1000 bp in length.


A_J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


AKR_J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


BALB_CJ
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


C3H_HeJ
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


CBA_J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta

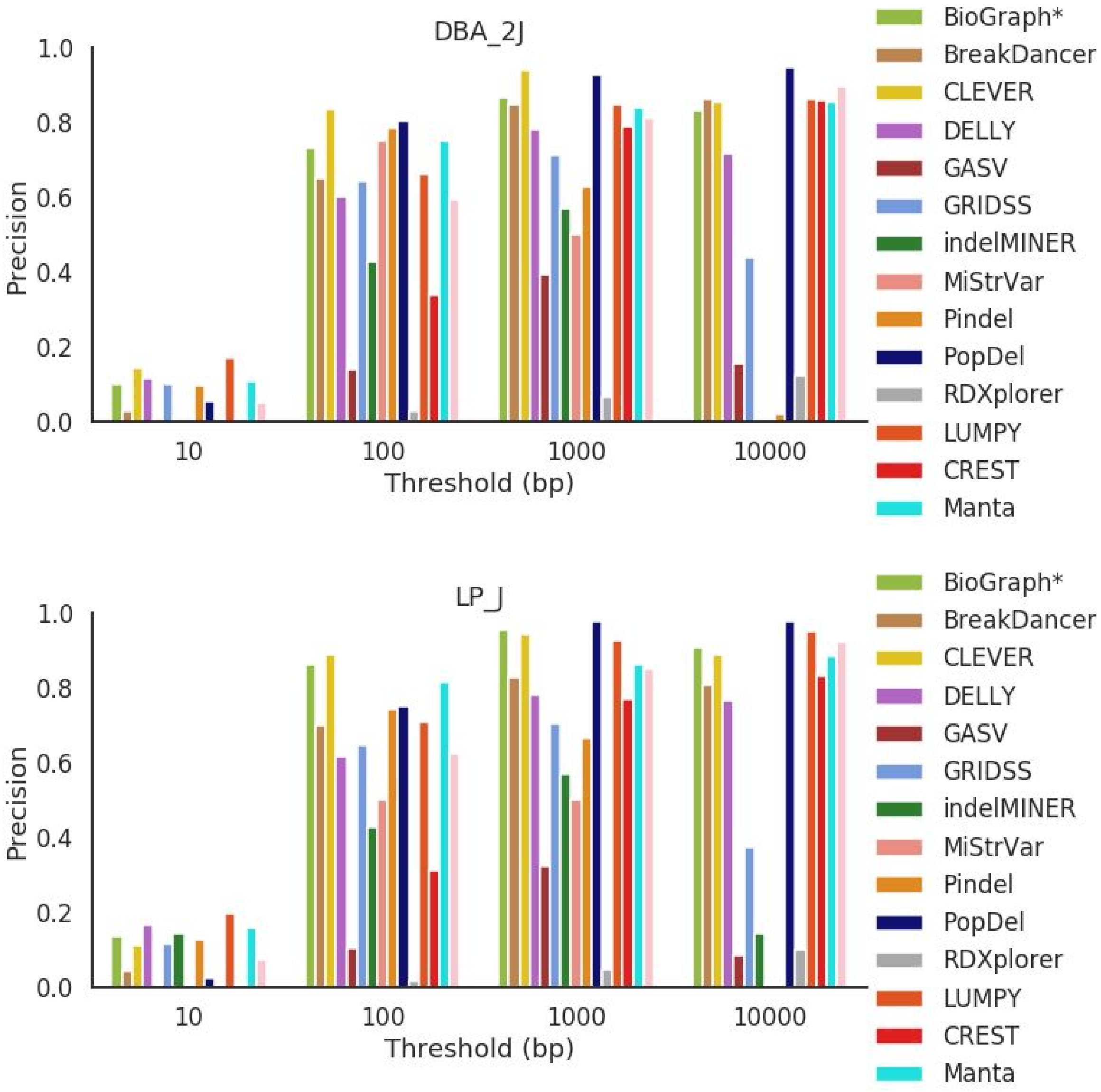
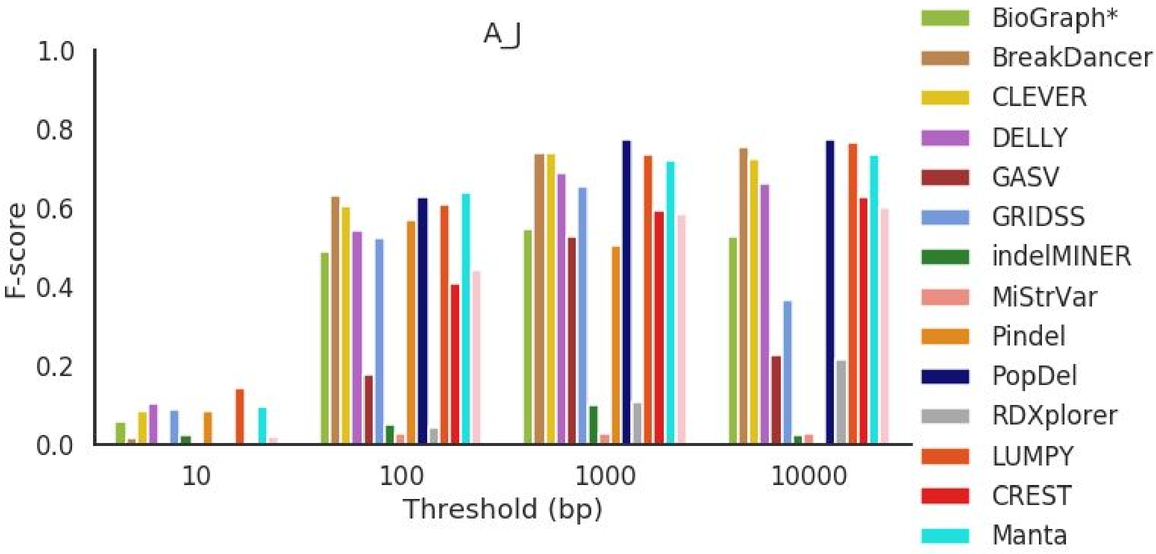
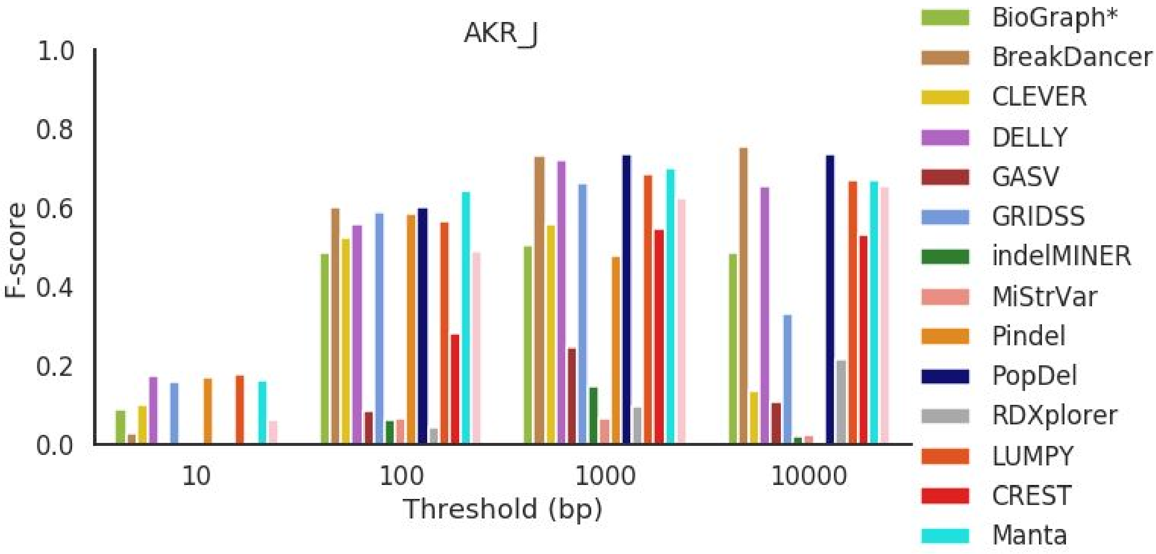
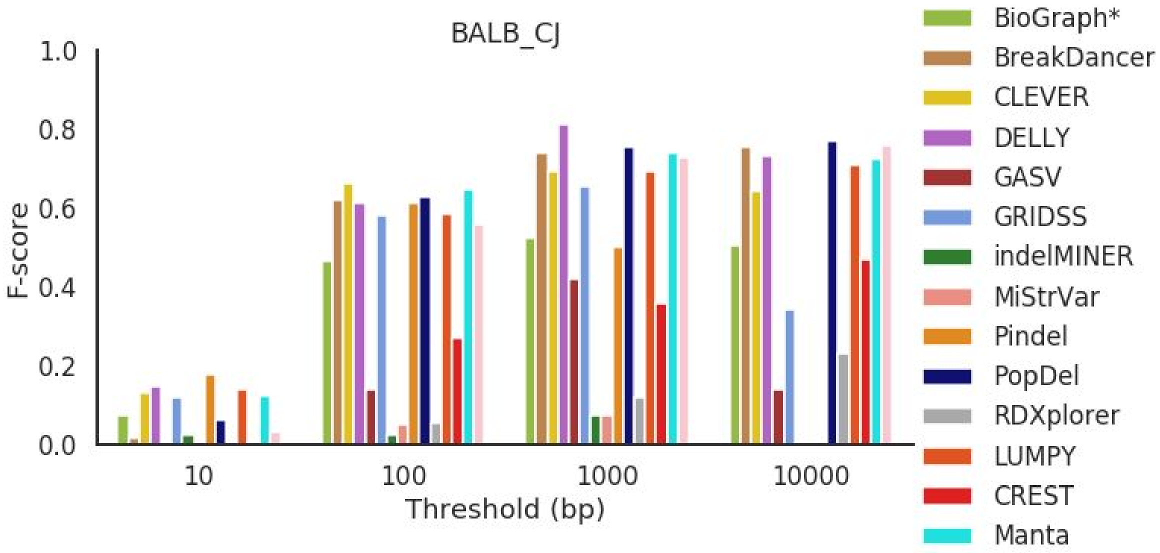
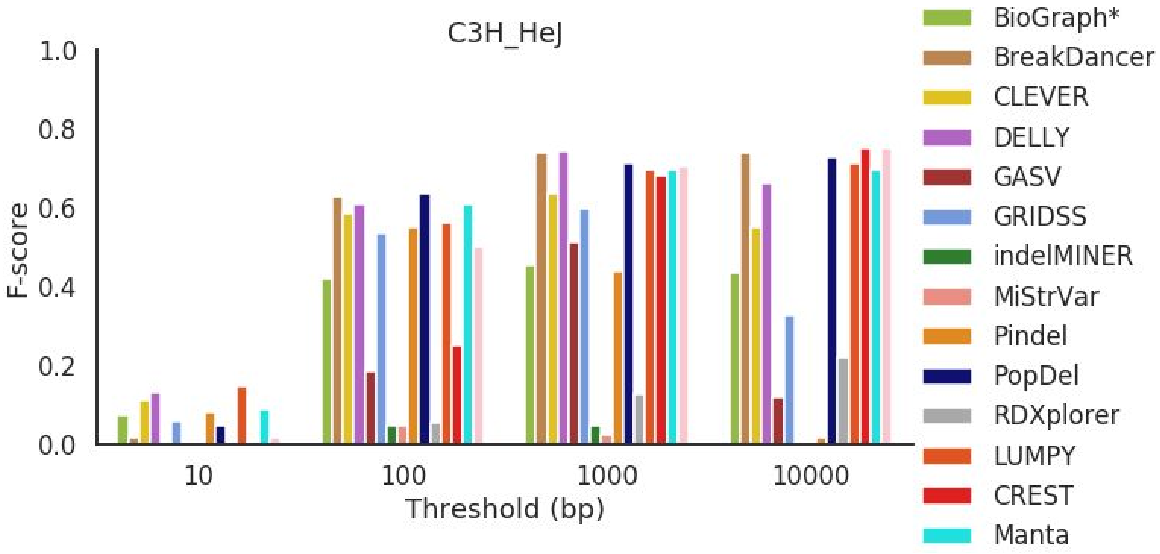


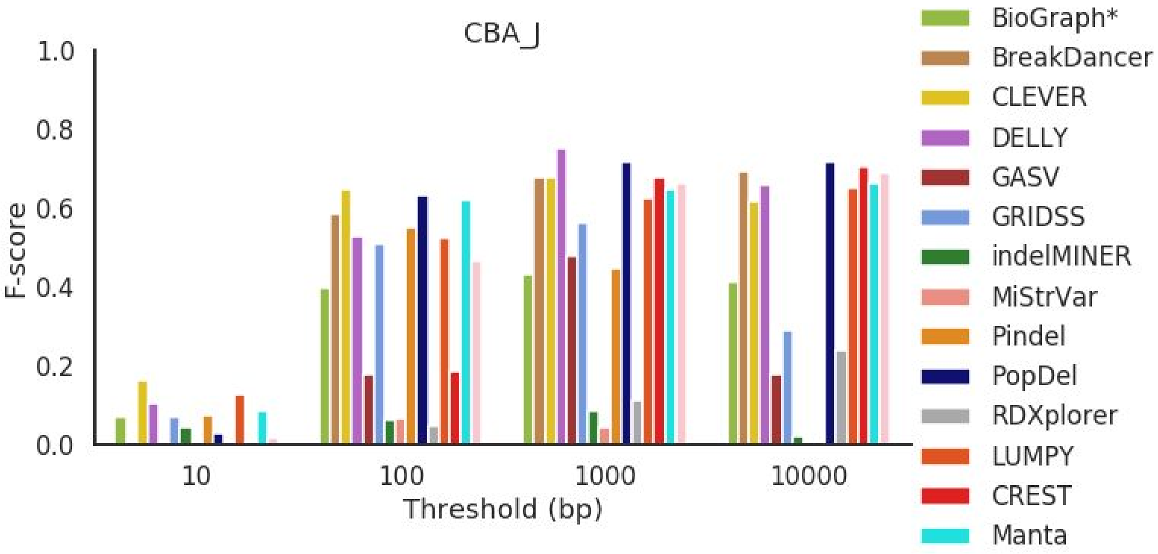
Figure S22. Precision across all mouse strains for deletions between 500 bp and 1000 bp in length.


A_J
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


AKR_J
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


BALB_CJ
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


C3H_HeJ
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


CBA_J
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta

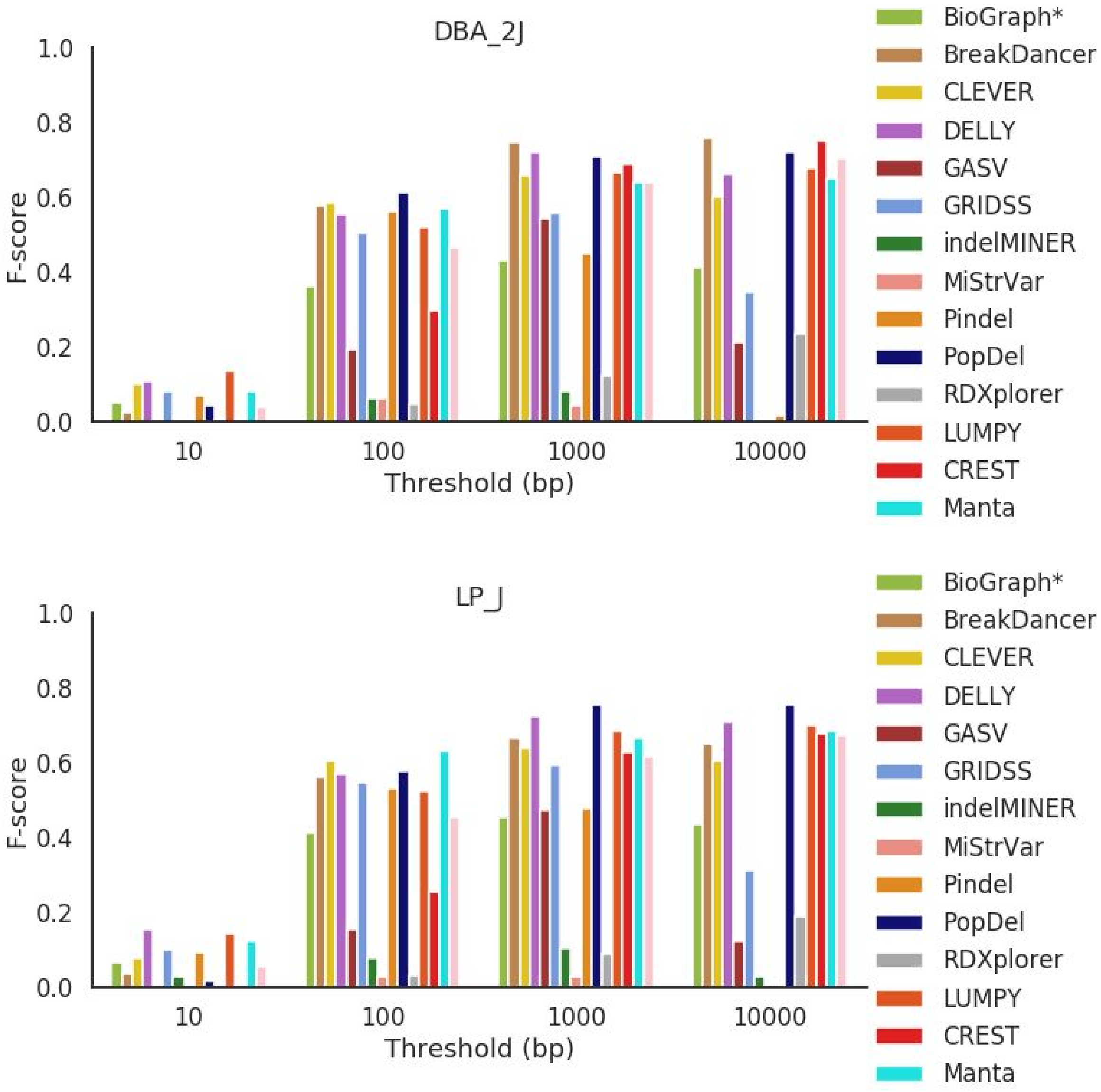
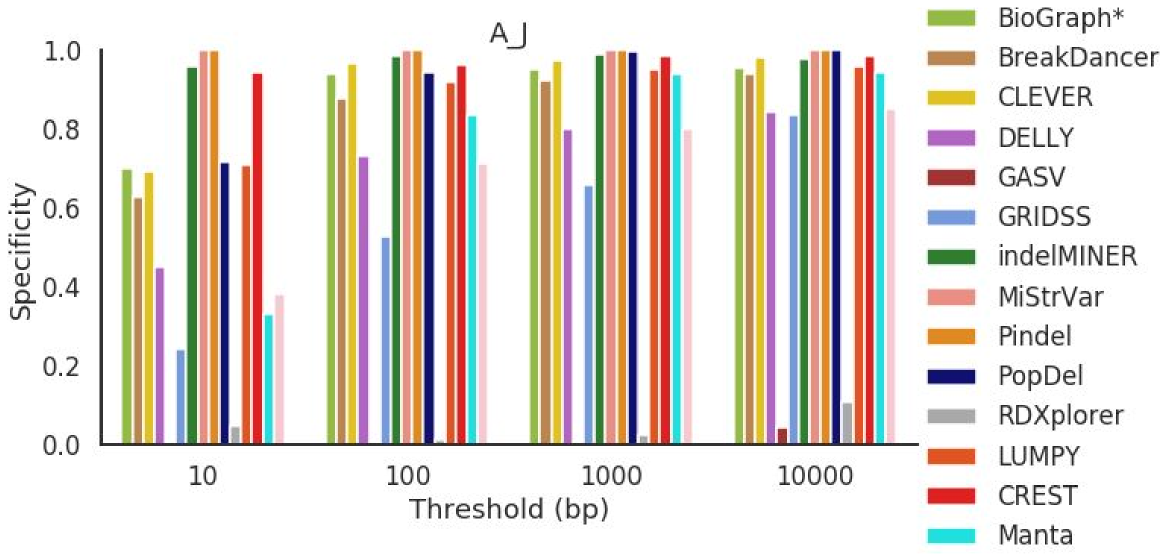
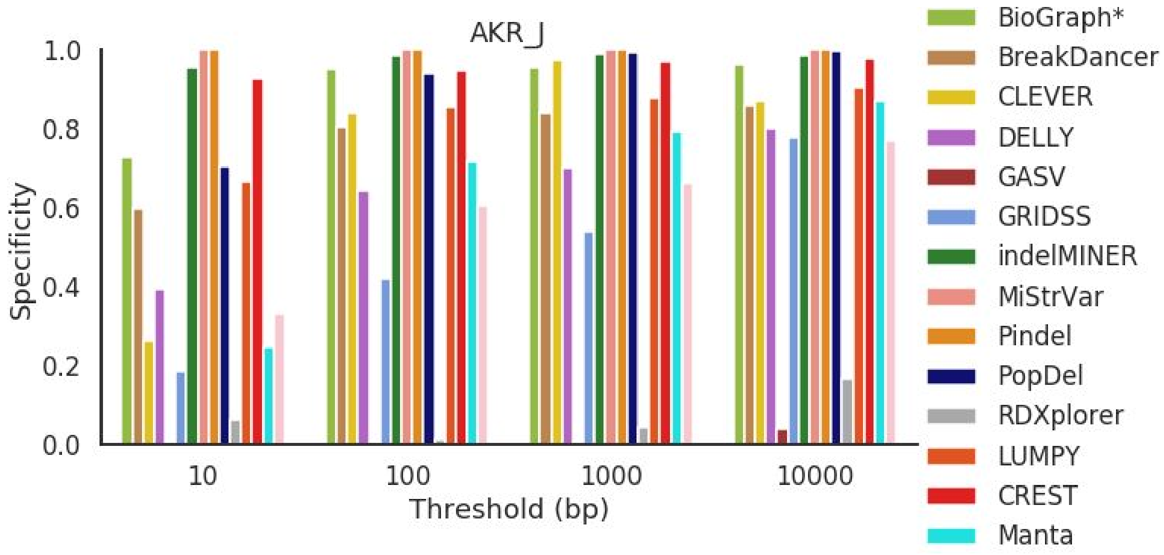
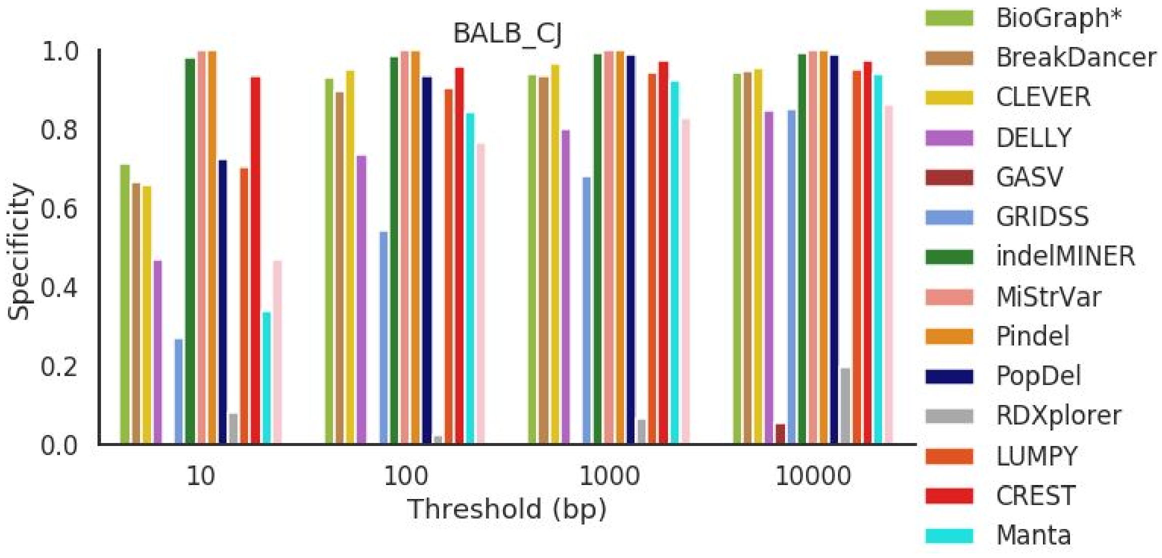
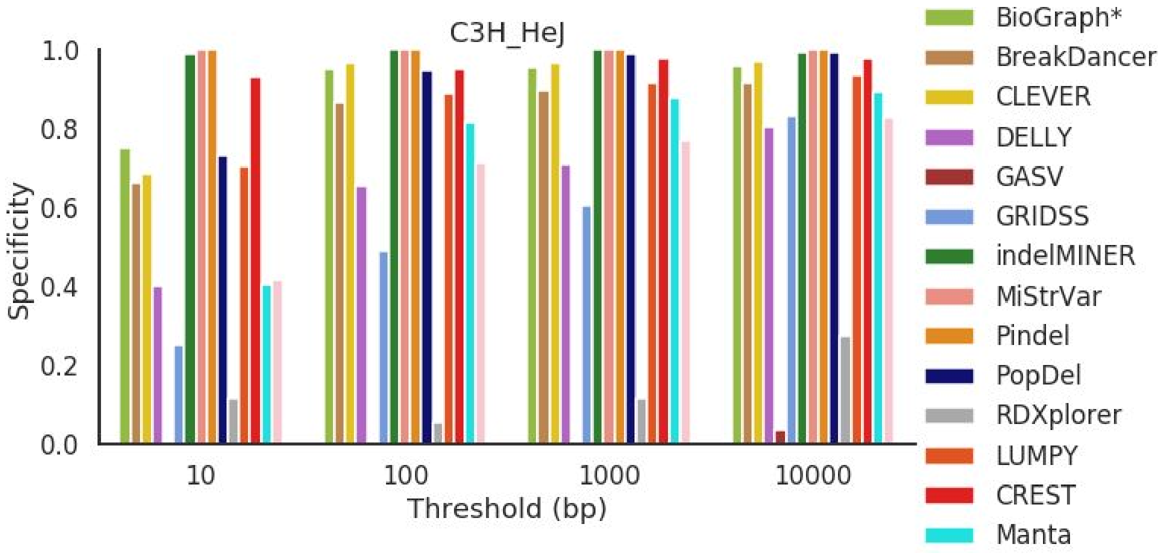


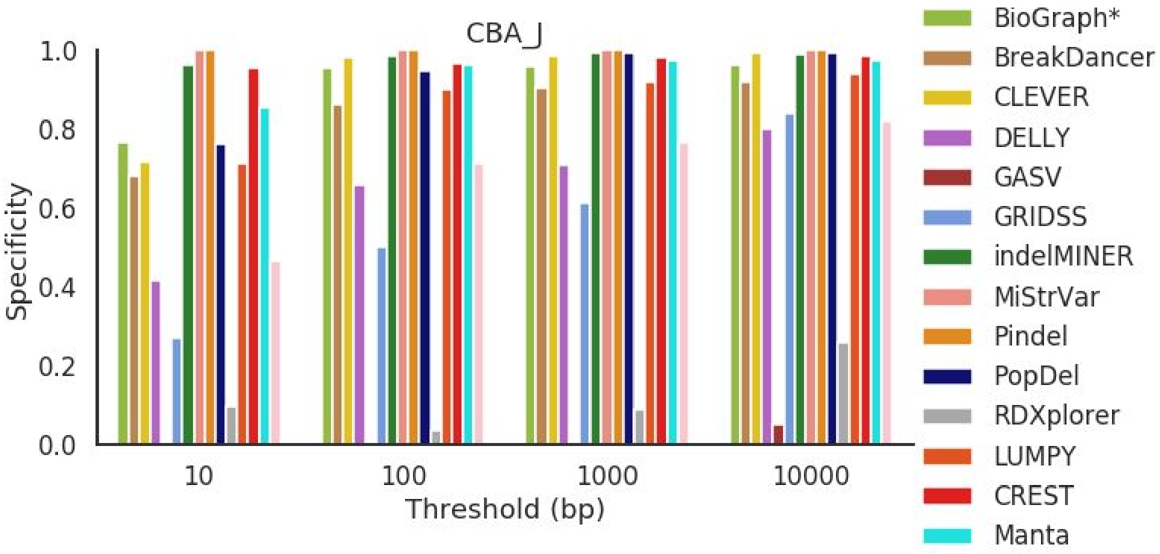
Figure S23. F-score across all mouse strains for deletions between 500 bp and 1000 bp in length.


A_J
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


AKR_J
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


BALB_CJ
Specificity
0.0
0.2
0.4
0.6
0.8
1.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


C3H_HeJ
Specificity
0.0
0.2
0.4
0.6
0.8
1.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


CBA_J
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta

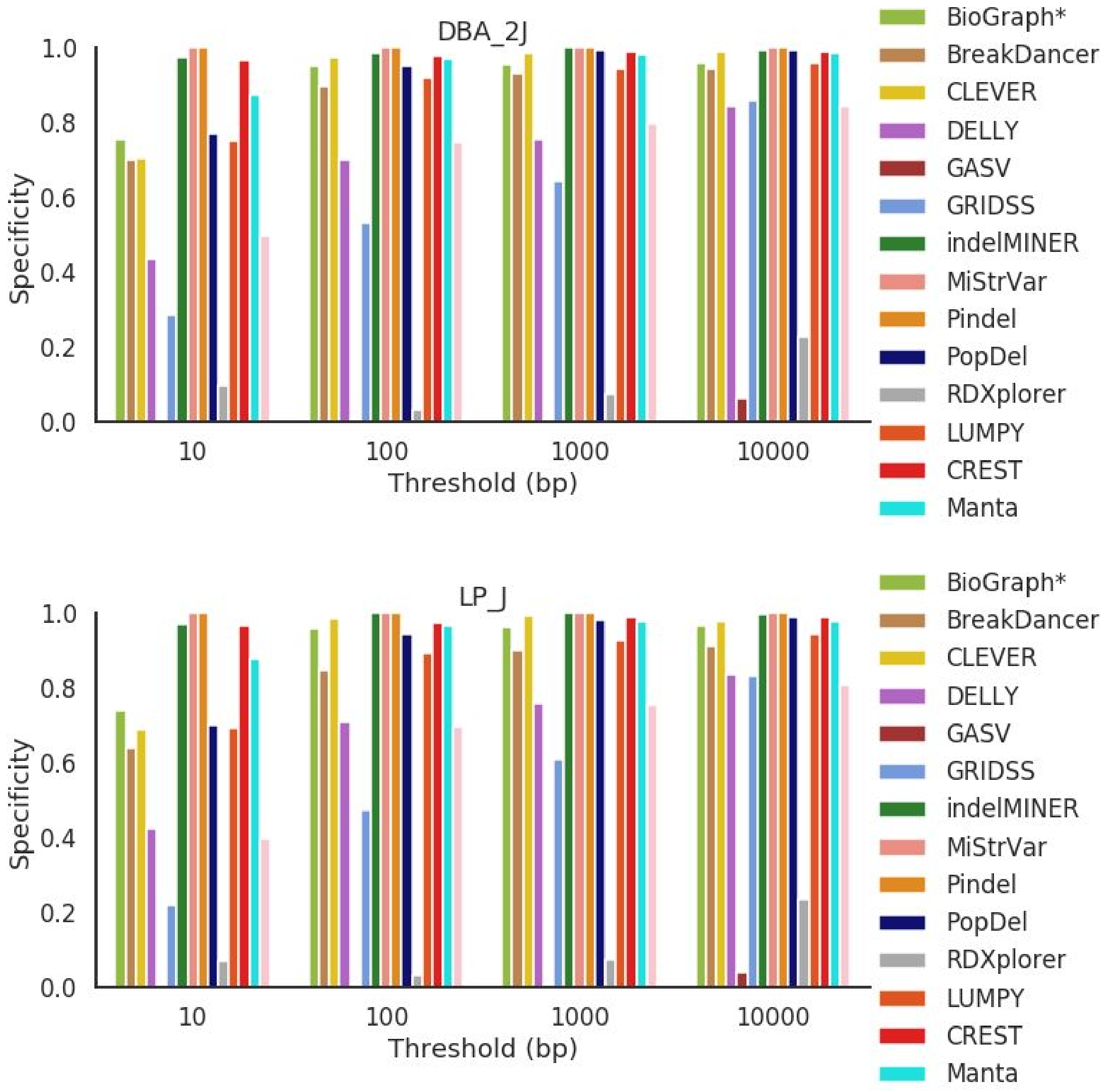
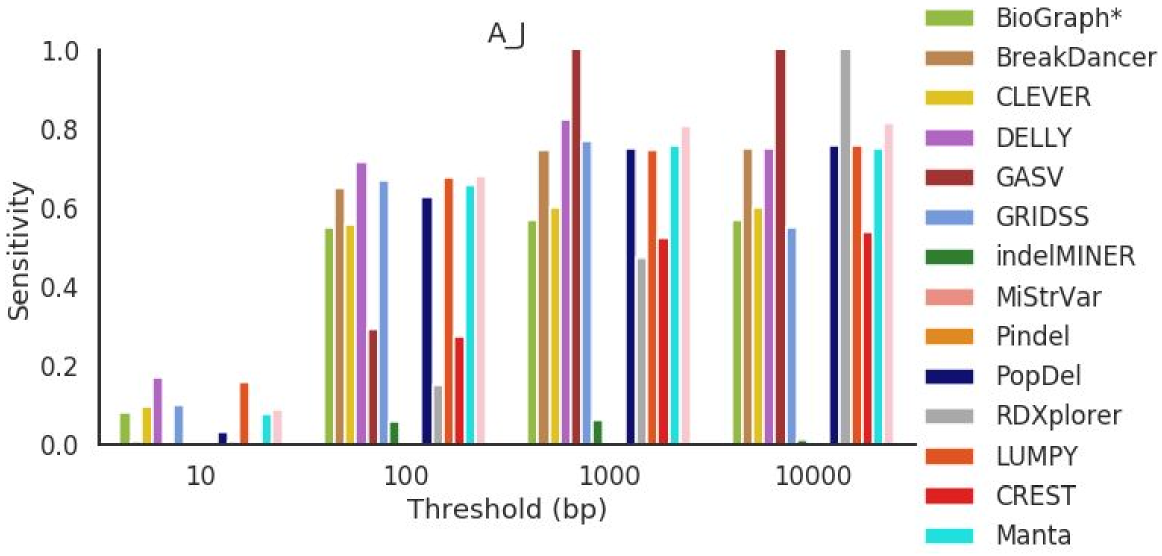
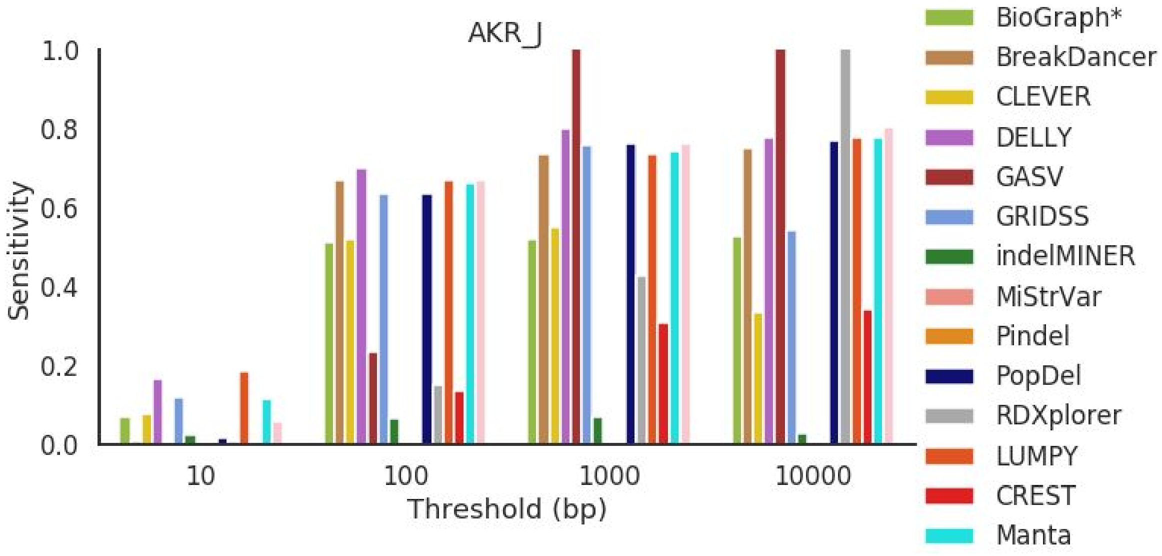
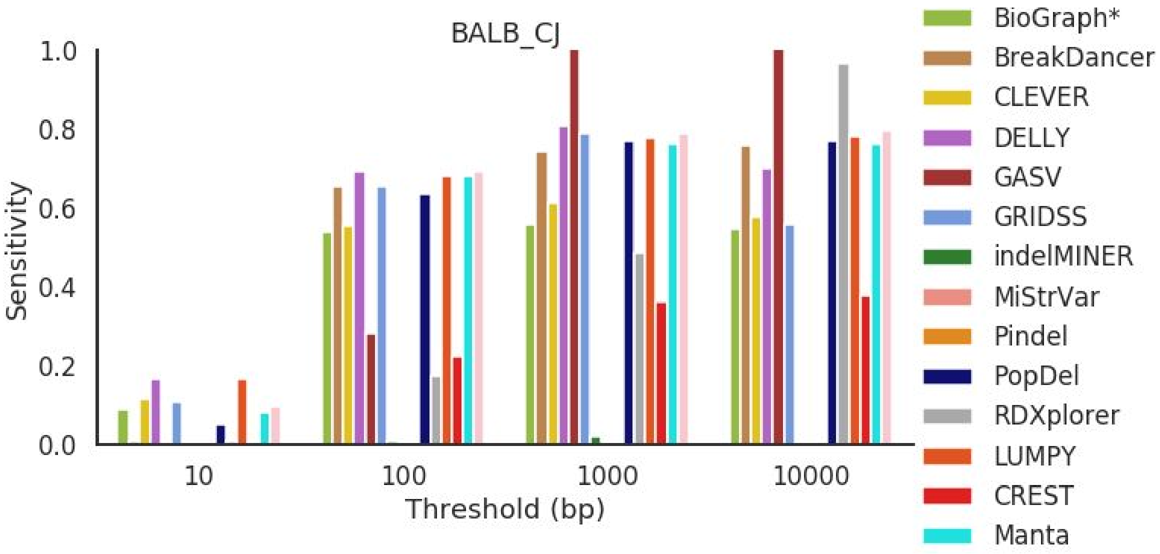
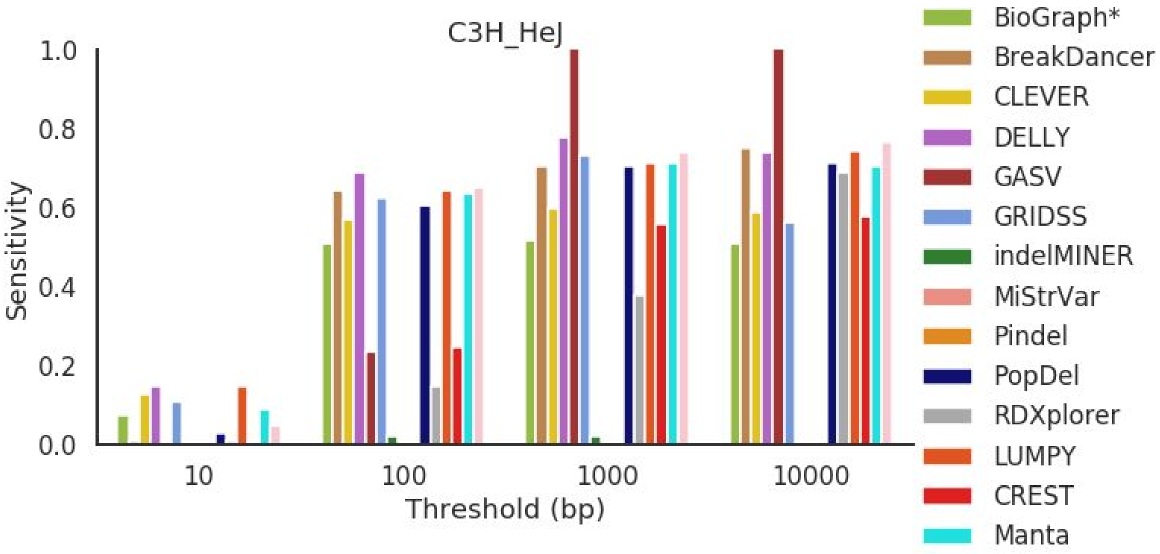


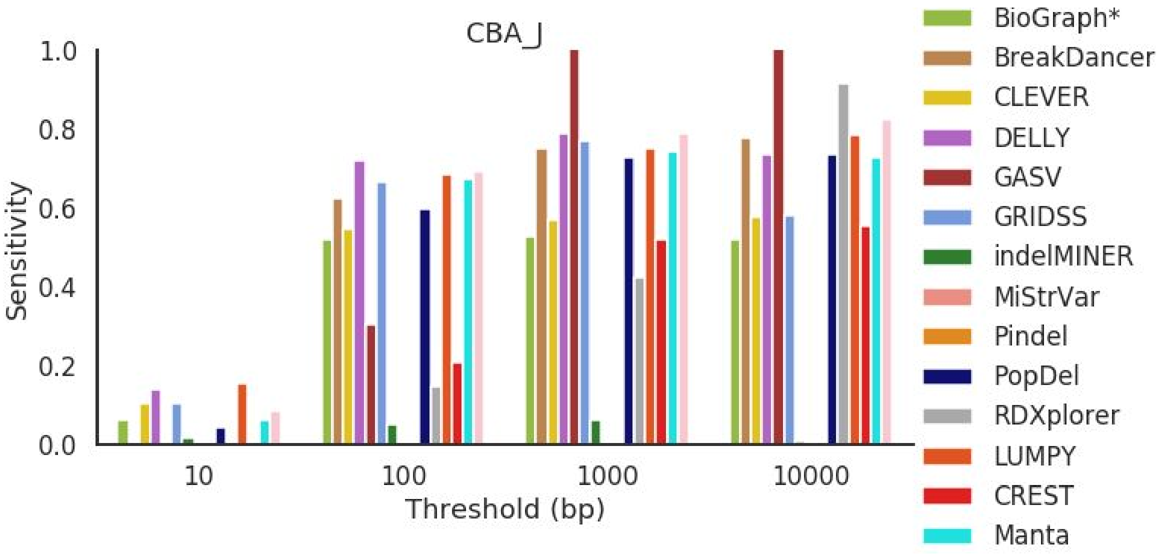
Figure S24 Specificity across all mouse strains for deletions 1000 bp and greater in length.


A_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


AKR_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


BALB_CJ
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


C3H_HeJ
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


CBA_J
Sensitivity
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta

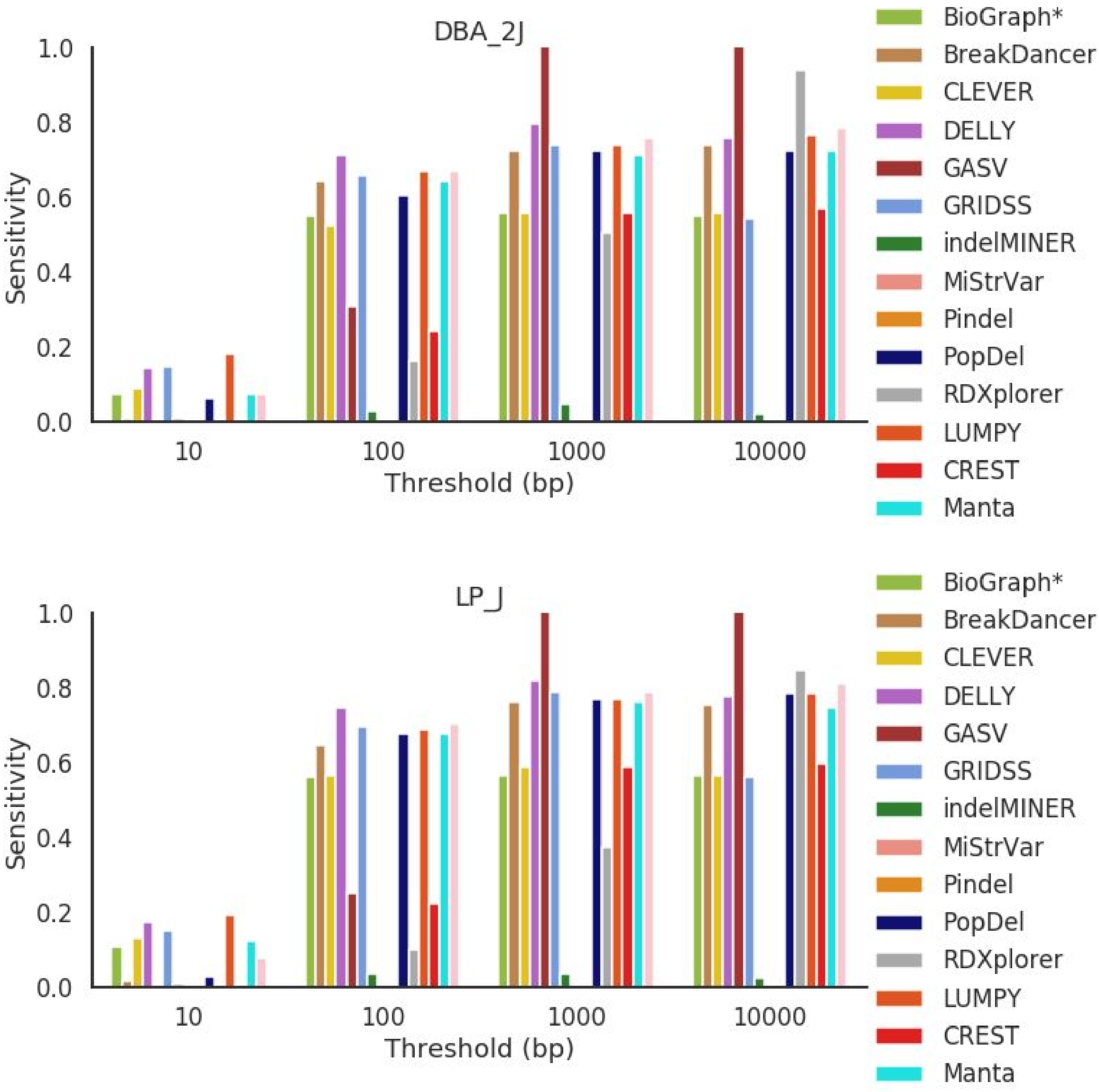
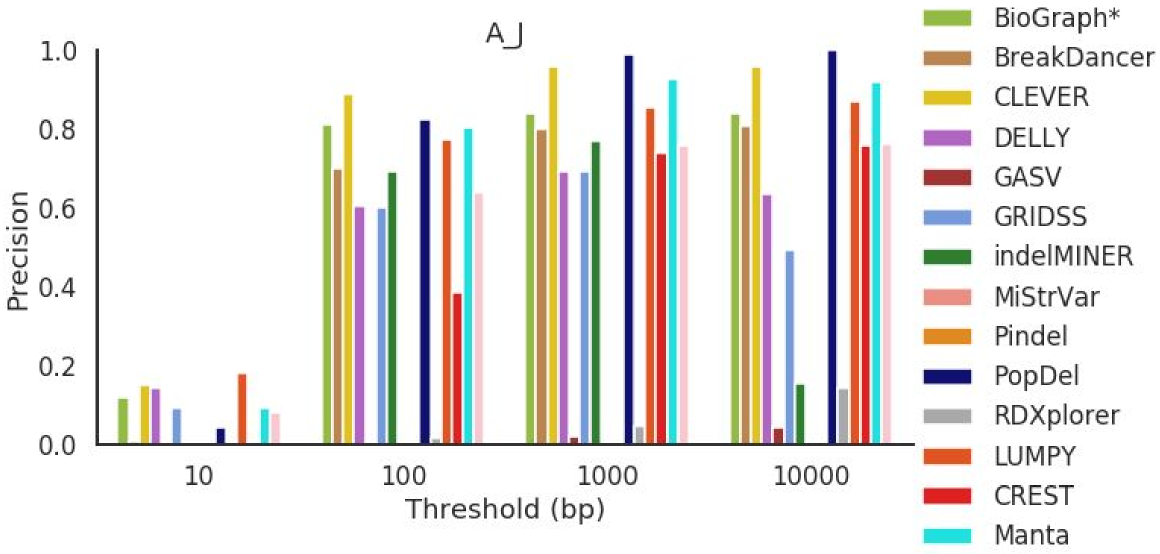
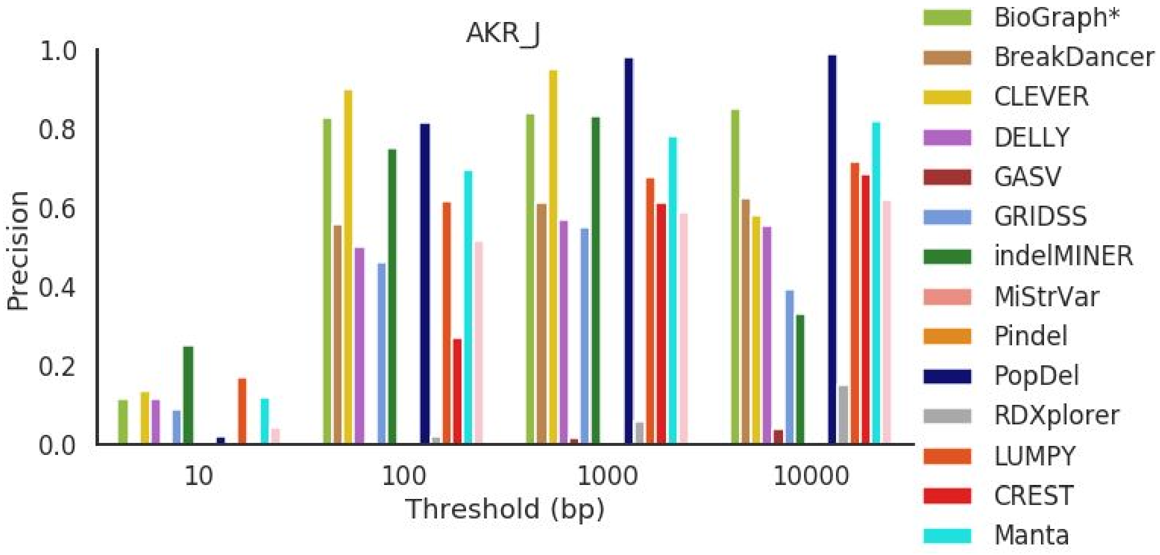
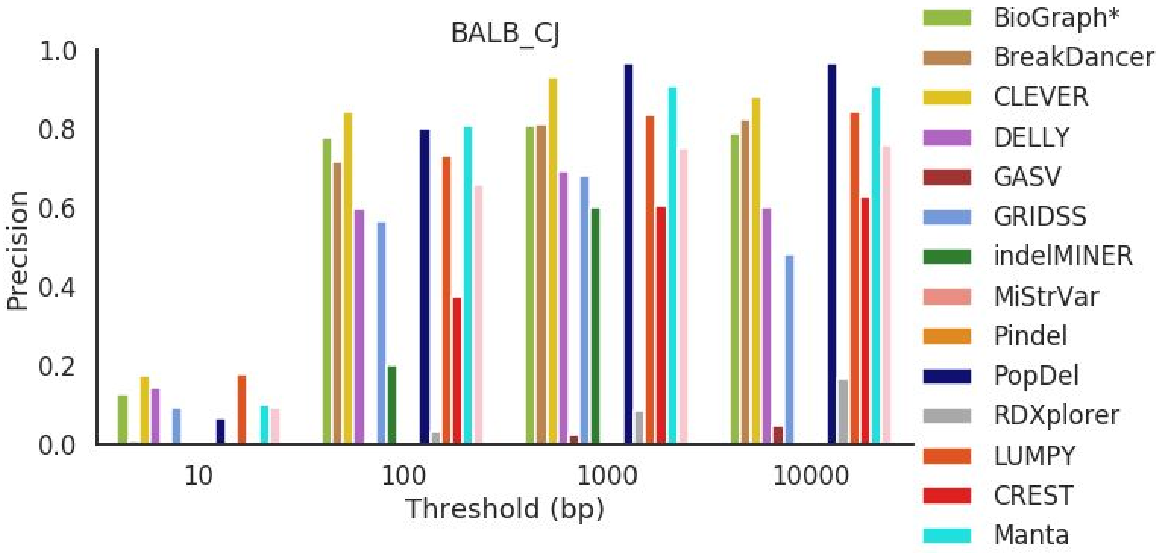
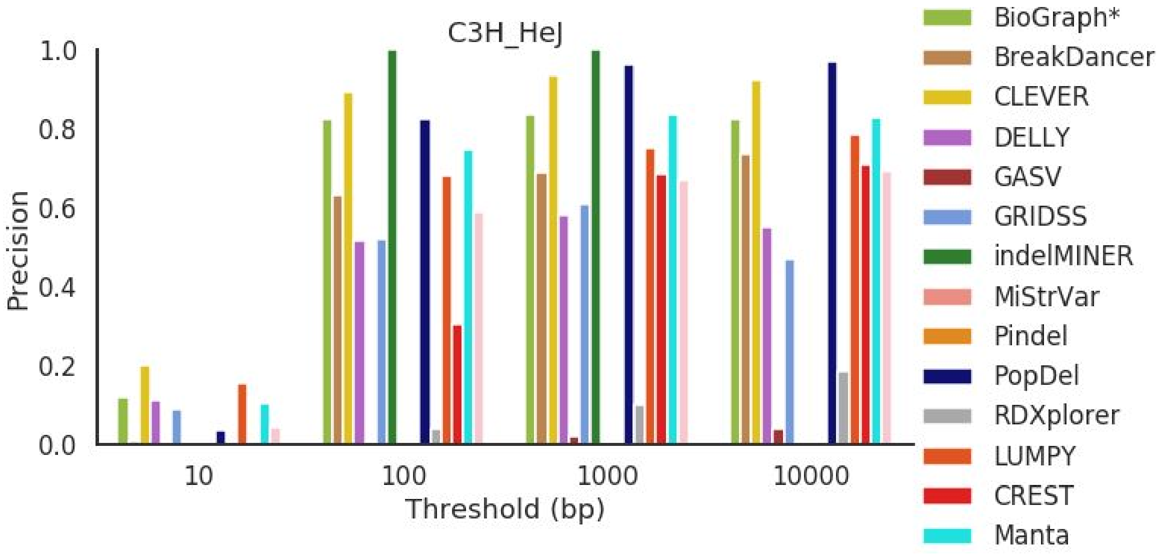


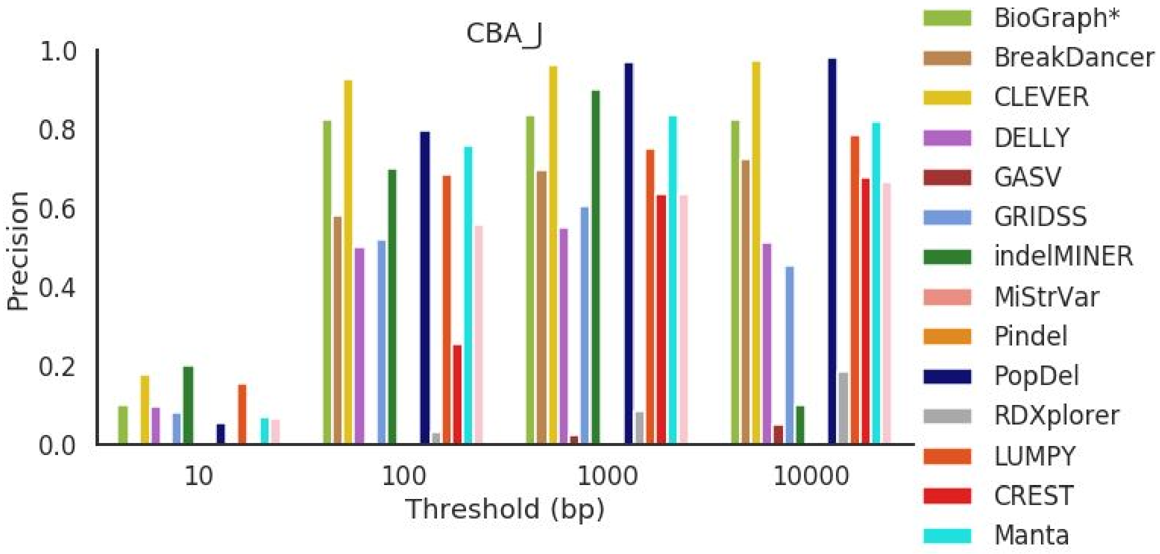
Figure S25. Sensitivity across all mouse strains for deletions 1000 bp and greater in length.


A_J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


AKR_J
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


BALB_CJ
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


C3H_HeJ
Precision
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


CBA_J
1.0
0.8
0.6
0.4
0.2
0.0
Precision
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta

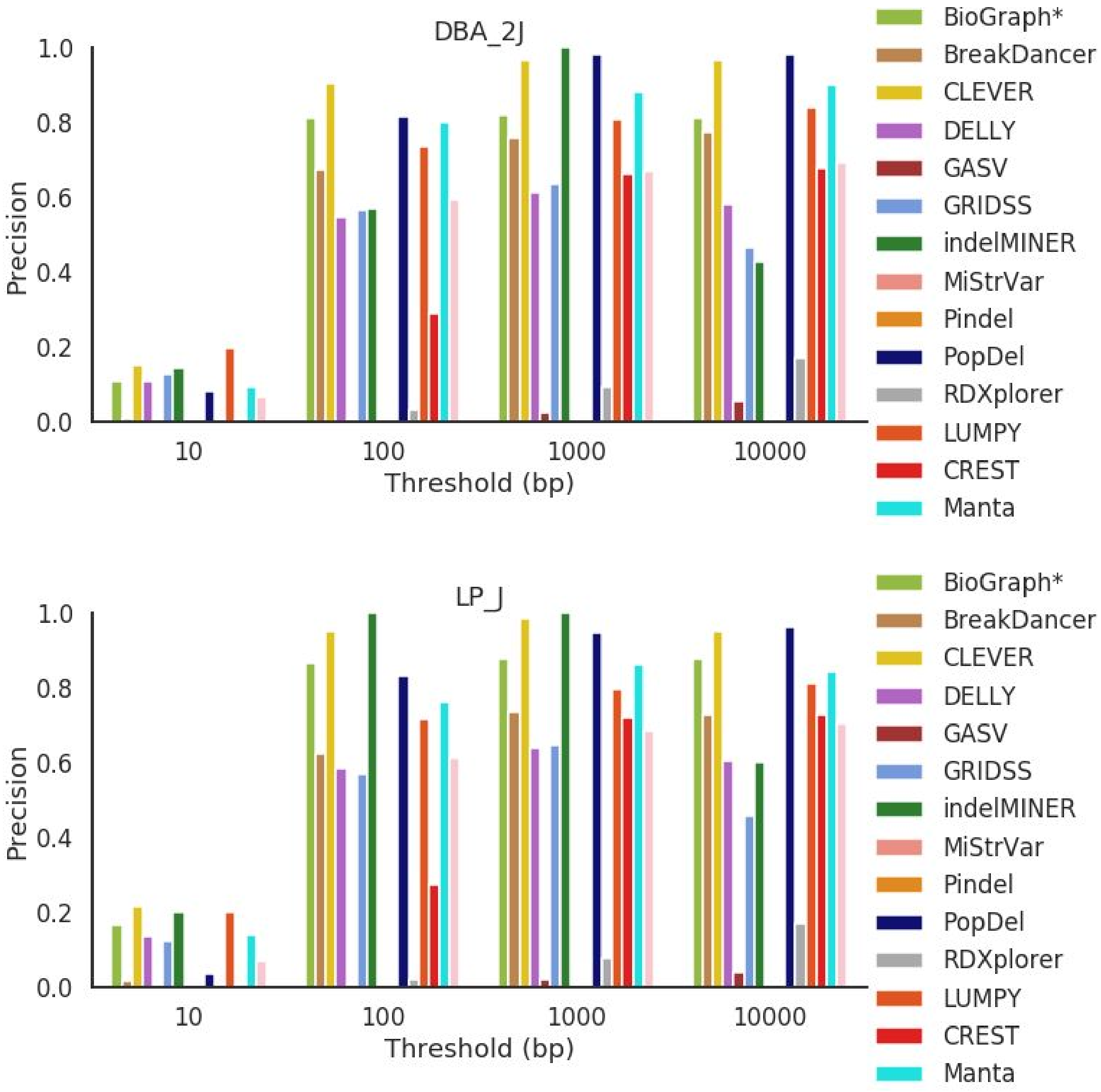
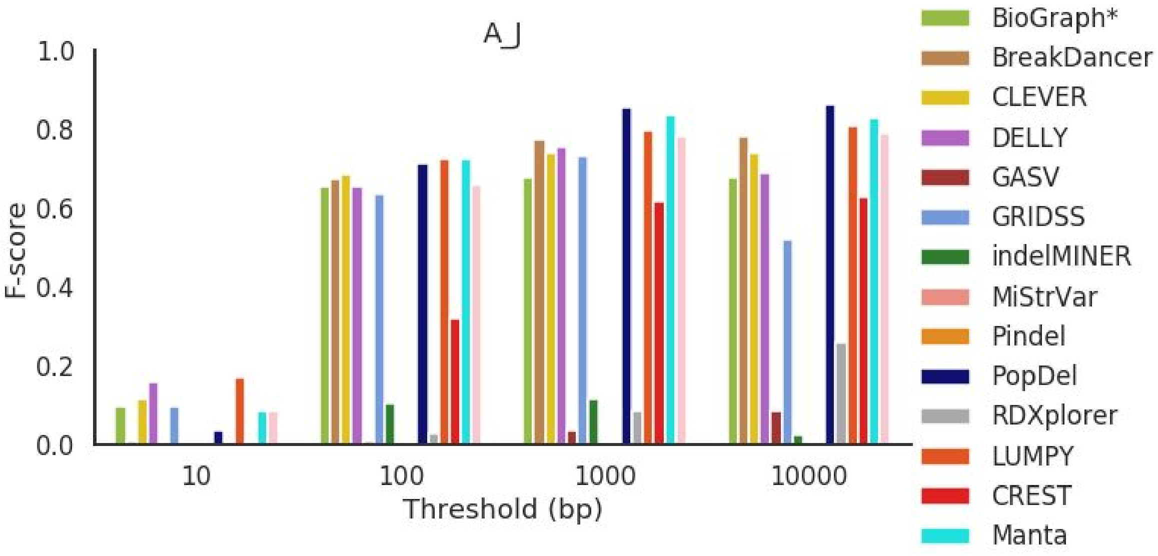
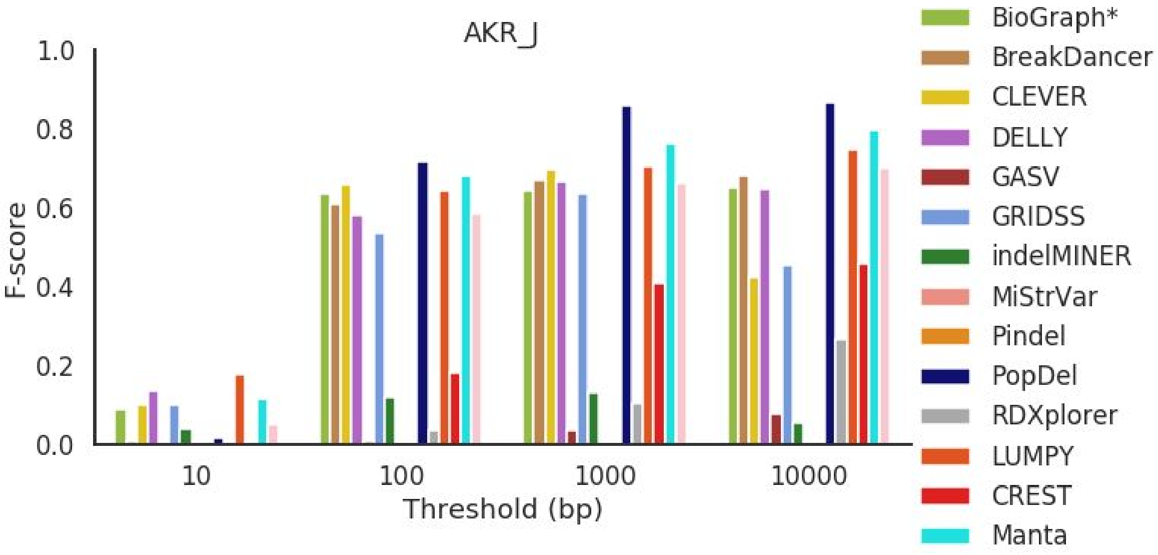
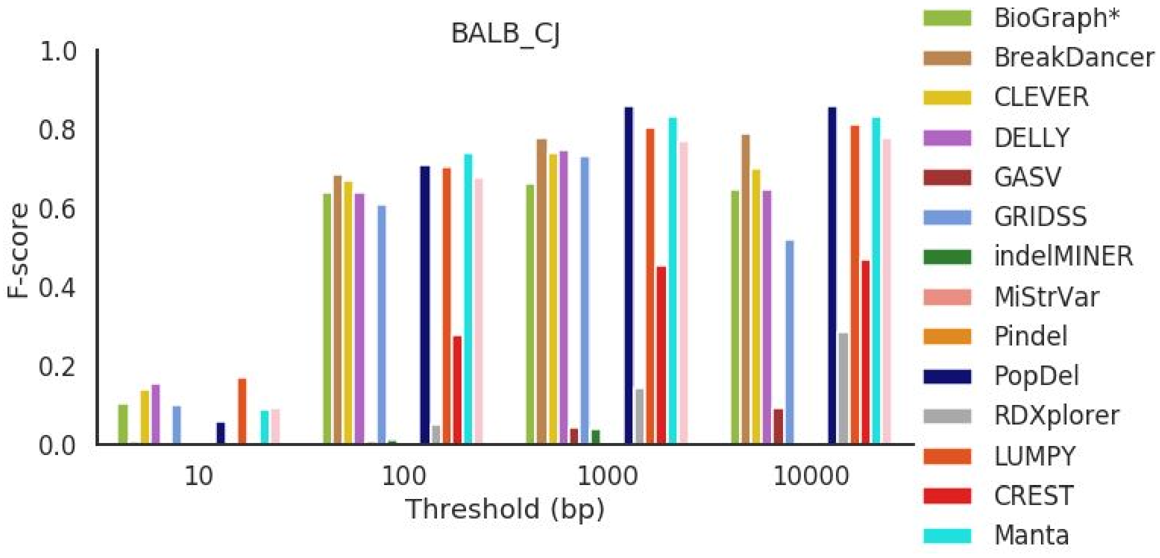
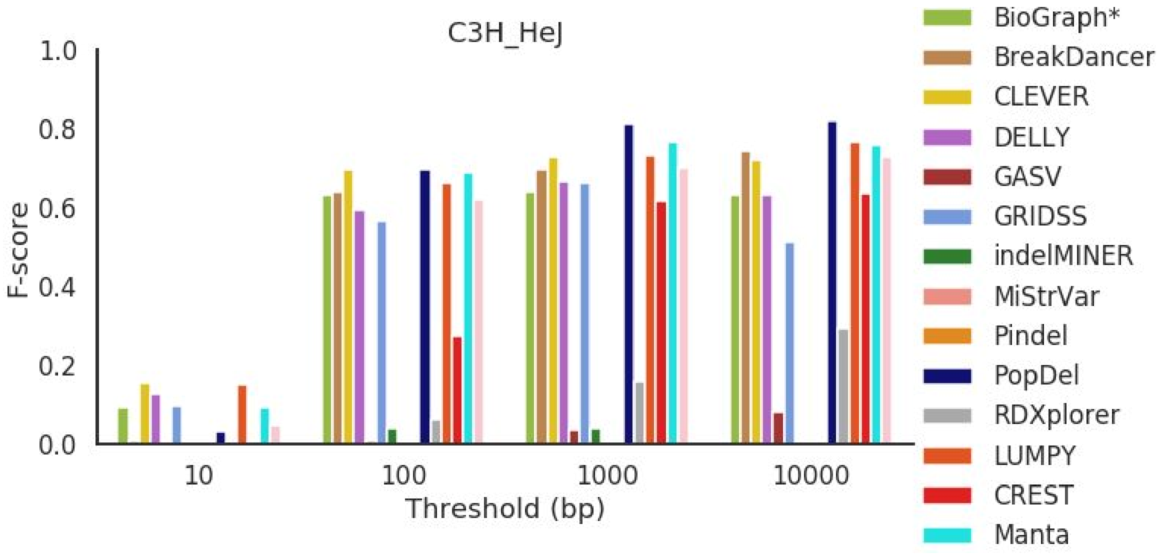


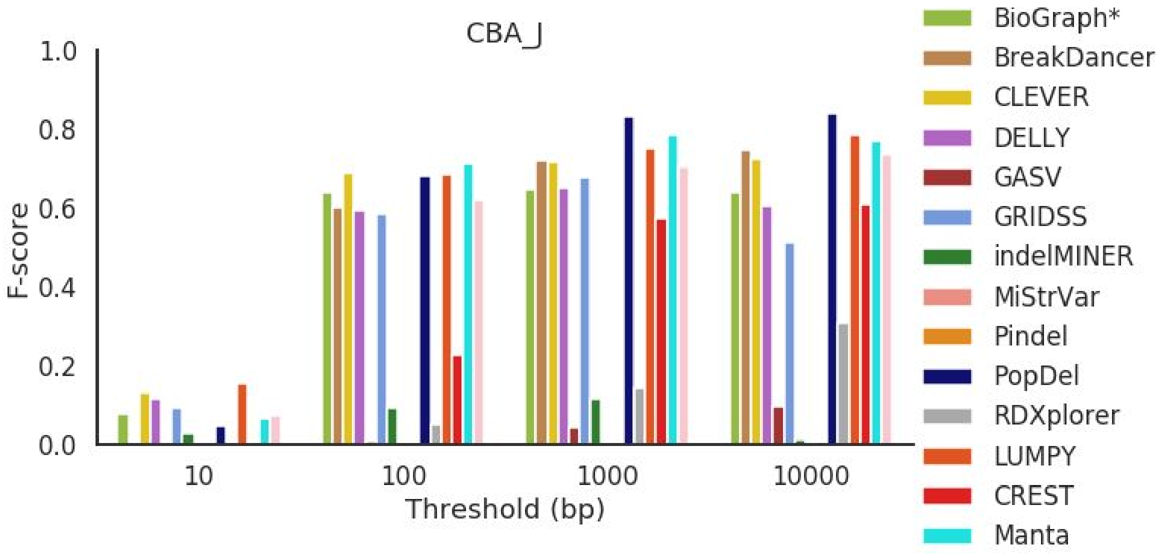
Figure S26. Precision across all mouse strains for deletions 1000 bp and greater in length.


A_J
1.0
0.8
0.6
0.4
0.2
0.0
F-score
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


AKR_J
1.0
0.8
0.6
0.4
0.2
0.0
F-score
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


BALB_CJ
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


C3H_HeJ
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta


CBA_J
F-score
1.0
0.8
0.6
0.4
0.2
0.0
10
100
1000
10000
Threshold (bp)
BioGraph*
BreakDancer
CLEVER
DELLY
GASV
GRIDSS
indelMINER
MiStrVar
Pindel
PopDel
RDXplorer
LUMPY
CREST
Manta

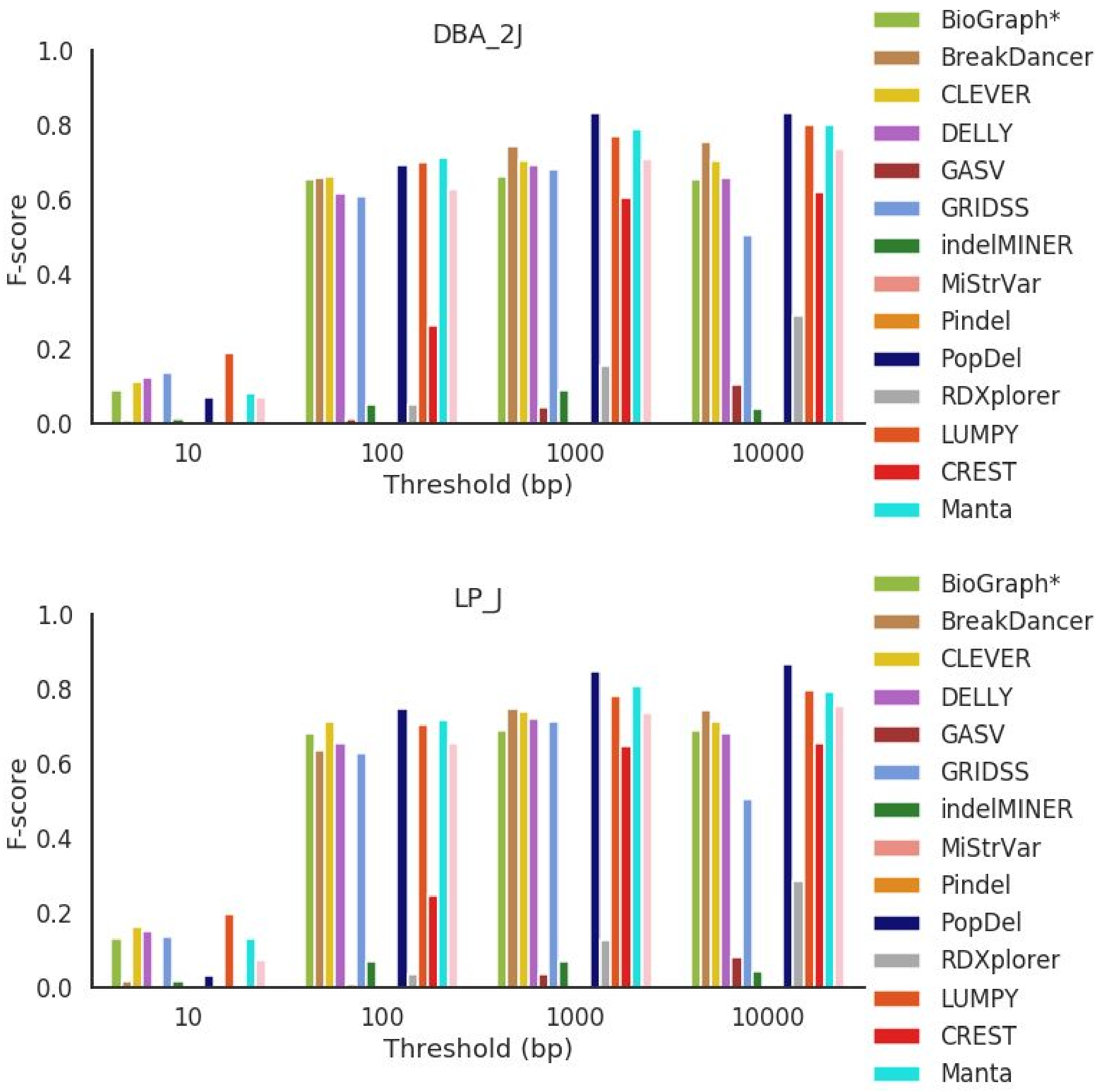


Figure S27. F-score across all mouse strains for deletions 1000 bp and greater in length.