Long-Short Term Memory Model for chromosomal aberration detection in Non-Invasive Prenatal Testing

Noor van Ruyven
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology, Delft, The Netherlands

Abstract

In 1997 it was discovered that fragments of DNA circulate freely in the blood plasma and, in the case of pregnancy, this DNA consists of DNA belonging to both the mother and the fetus. This circulating free DNA has made it possible to test for chromosomal aberration in the fetus through non-invasive methods, thereby avoiding the 1 in 100 chance of causing a miscarriage. Since then, multiple methods have been developed to detect chromosomal abnormalities with increasing accuracy and decreasing costs. The current state-of-the-art WISECONDOR uses a within-sample reference set, which is then used to calculate the z-score on a sliding window to determine whether an aberration is present or not. Here, we introduce a deep learning approach to non-invasive prenatal testing in the form of a Long-Short Term Memory model, which takes a sequence of GC normalized read counts per bin on the genome and outputs the label healthy or aberrated per bin. To test the performance of both WISECONDOR and the newly proposed model, data is simulated, and multiple experiments are set up to test the influence of certain aspects of NIPT. When comparing the LSTM model to WISECONDOR, it was shown that the LSTM model is still too inconsistent in its performance. This is caused by its reliance on the initialization of the weights and its dependence on the training set's composition.

1 Introduction

In every pregnancy, a risk of the fetus being born with a chromosomal aberration is present. The most common chromosomal abnormality is aneuploidy, where there is either an extra chromosome - trisomy - or a missing chromosome - monosomy. The best-known form of aneuploidy is trisomy 21, also known as Down Syndrome, which occurs in 1 out of 700 live births in the United States(1). Two other more uncommon chromosomal abnormalities are trisomy 18, Edwards syndrome, and trisomy 13, Patau syndrome. These occur in 1 out of 5,000 and 1 out of 16,000 live births respectively(2; 3).

Obtaining information on the presence of a chromosomal aberration could help prepare the parents and caregivers, either through medication for the mother or fetus to help decrease the severity of the condition, or mentally by preparing them for the condition's effect or the chance of a stillbirth.

To detect such chromosomal aberrations in a fetus, broadly speaking, there are two tracks: invasive and non-invasive prenatal testing.

Invasive prenatal testing has been possible since the 1950s through either amniocentesis or chorionic villus sampling (CVS). Both procedures can determine the gender of the fetus and detect chromosomal abnormalities, though CVS can be performed at an earlier gestational age (4). Both methods have a high accuracy rate: amniocentesis has an accuracy of 99,4%, with a false positive rate of ~4% (5). CVS has an accuracy of ~99%, with a false positive rate of ~0,15%(6). Though these methods give definitive results with very high accuracy, there is a 1% chance of causing a miscarriage when taking the test (7; 8).

In non-invasive prenatal testing (NIPT), the fetus's DNA is obtained through the mother's blood plasma. In 1948 it was discovered that there are fragments of DNA circulating freely in the blood plasma: circulating free DNA (cfDNA)(9) and later in 1997, Lo et al. found that the circulating free DNA of pregnant women also contained DNA of the fetus, so-called cell-free fetal DNA (cffDNA)(10). Through Next Generation Sequencing, the fragments of cfDNA can be sequenced, giving an accurate representation of the mother and fetus's DNA, which can then be used to determine whether an aberration is present or not.

On average, the amount of fetal DNA in the cfDNA lies between 10% and 15%, but it can range between 3% and 30%(11). The amount of fetal DNA in the sample is called the fetal fraction. The fetal fraction plays a vital role in NIPT; if a sample does not contain enough fetal DNA, the test's conclusions may not be linked to the fetus. These results could lead to false negatives: i.e. the test concludes that no chromosomal aberrations are present, but the sample contained insufficient fetal DNA, meaning that the conclusion is drawn from the mother's DNA.

In 2008 Chiu et al.(12) developed a Z-scoring method that could detect trisomy 21 from the mother's blood plasma. This method was effective for both high and low coverage data. However, a drawback of this approach is that a set of healthy samples is required to compare to the target sample. To reduce the number of experimental con-founders, these healthy samples have to be re-sequenced each time, to ensure identical experimental conditions, which increases the testing cost.

To avoid this limitation, Straver et al.(13) created the current state-of-the-art method, WISECONDOR. WISECONDOR uses a within-sample reference set to determine for each bin whether it is aberrated or not. Furthermore, instead of only testing each bin individually for aberrations, a sliding window is used to find the aberration in its entirety.

In this thesis, a deep learning model is proposed as a novel method for NIPT. This model takes a sequence of GC normalized read counts per bin on the genome and outputs a label 0 (healthy) or 1 (aberrated) for each bin. This is achieved through a bidirectional LSTM model, consisting of a cell for each bin in the input.

This paper is organized as follows. In section 2.1 the available dataset and the procedure of simulating new datasets are explained, followed by a description of data preprocessing, which includes discretization of the aligned read counts into bins and correcting for GC content bias. Then, in section 2.2 the current state-of-the-art method WISECONDOR is explained in more detail. Section 2.3 introduces the newly proposed method, including the rest of the pipeline. In section 2.4 the experiments that have been conducted will be defined. Section 2.5 describes the metrics used to evaluate the model and evaluations of each of the experiments. In sections 3 and 4, the results of the experiments will be shown and discussed. Section 5 contains the conclusions that have been drawn and recommendations for future work.

2 Method

2.1 Data

As mentioned before, cfDNA from both the mother and the fetus can be found in the mother's blood plasma. Here the amount of fetal DNA is quite small. From this small fraction of fetal DNA, an accurate representation of the fetus's actual DNA must be created: it has to be sequenced. This DNA sequencing is done using next-generation sequencing (NGS): a process to determine the sequence of nucleotides A, C, G, and T in a sample. These sequences are then aligned to their respective location on the human genome. Once these reads are aligned, the preprocessing can start by counting the amount of reads mapping to each location and correcting them for the GC content. This will be explained in more details in sections 2.1.3 and 2.1.4.

2.1.1 Experimental Data

A set of experimental data from the VU Medical Center Amsterdam diagnostic centre is available for this thesis. It consists of 401 healthy samples and 183 aberrated samples. The labels for each of these samples have been obtained through other NIPT methods, which are not 100% accurate. Therefore we cannot definitively say that the labels are the ground truth. However, for this thesis, the assumption is made that the labels are correct. A thorough analysis of the data can be found in the supplementary.

2.1.2 Simulated Data

There are three reasons to utilize simulated data over the available experimental data. First of all, since we are building a machine learning model, having sufficient data is essential. If too little data is available or the data is incorrect, the model might not learn anything and will under-perform. Second, as mentioned before, the ground truth is not known for the available samples. By simulating the data, we can be sure of the ground truth. Last, by simulating the data, the influence of multiple of the samples' characteristics on the model's performance can be tested—for example, decreasing the fetal fraction by 1%.

The simulation process consists of five steps: *variation*, *replacement*, *simulation*, *alignment*, and *preparation*. In *variation*, a variant is introduced in a given chromosome. This variant can either be a duplication or a deletion. A set of N's surrounds the variant to attach to the rest of the chromosome. Next, in *replacement*, the aberrated chromosome is placed within the hg19 human reference genome. Seqan's Mason is then used to *simulate* the data (14). Taking a maternal and a fetal reference sample as input Mason will simulate a new sample with a user-defines number of reads. Hg19 is used as the maternal reference, and the newly aberrated sample is used as the fetal reference. In *alignment*, the created reads are then aligned to the reference genome. For the alignment, Burrows-Wheeler Alignment (BWA) was used (15). BWA is used because it efficiently aligns small sequencing reads against a large reference genome (such as the human genome) while allowing gaps and mismatches. Last, in *preparation*, the aligned reads are counted per bin and GC corrected to prepare for the model.

Datasets for six experiments are created using this simulation process. In each of these experiments, the influence of a different NIPT factor is analyzed. These datasets are summarized below, and the Experiment plan is attached in the supplementary for a more detailed overview.

The main goal of Experiment 1 is to obtain an overall performance of the model to benchmark against WISECONDOR. For this experiment, 1000 aberrated and 1000 healthy samples are simulated. Each sample has 20 million reads and a fetal fraction of 10%. The aberrated samples contain one duplication per sample, located on either chromosome 13, 18, 19, or 21 with a size of 60% to 100% of the chromosome length. In Experiment 2, the focus lies on the size of the aberration. They are duplications located on chromosome 21 with a size ranging from 1Mb to 49Mb with steps of 1Mb. For Experiment 3, the samples have a coverage ranging between 0.05x and 1.0x. This results in samples containing between 5 and 85 million reads per sample with steps of 10 million. For Experiment 4, the fetal fraction of the samples ranges between 1 and 20%. Experiment 5 tests the influence of the bin size, for which the samples of Experiment 1 can be re-used. In Experiment 6, samples containing a deletion instead of a duplication are added to the dataset of Experiment 1 to analyze the influence of a different variant type on the model's performance.

2.1.3 Read count per bin

After aligning the reads to their corresponding location, the number of reads can be counted to obtain the read depth or read coverage for each locus or area. Most commonly the chromosome or genome is divided into bins of a size N. So a chromosome of 1M base pairs can be divided into ten bins of size 100k. Next, for each of these bins, the number of reads mapping inside that bin can be counted to obtain the read count per bin.

2.1.4 GC correction

A challenge that occurs when using NGS is the GC bias. GC bias is the dependency between read coverage and the amount of GC content. When a fragment contains many 'G' or 'C' nucleotides, this fragment will be easier to sequence and will therefore be amplified more. This means that a GC-rich area will have more reads than an AT-rich area, even though that might not be the case in the original DNA sequence(16; 17). A well-known method to correct for this bias is Locally Weighted Scatterplot Smoothing.

Locally Weighted Scatterplot Smoothing Locally Weighted Scatterplot Smoothing (LOWESS) is a non-parametric method that creates a smooth line through data points to show the relationship between variables and helps the user spot trends. For GC content it can show where the GC-rich and GC-poor areas are, which can then be used in correcting the read counts.

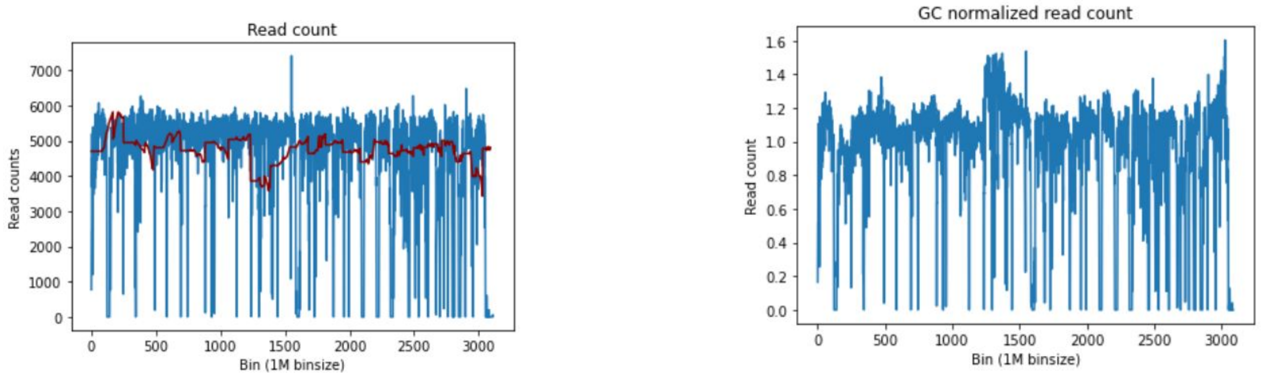
Similar to how the number of reads mapped to an individual bin is counted, the amount of 'G' and 'C' nucleotides is counted in that same bin on each chromosome using the reference genome. These GC counts are then plotted against their bins, and from this plot, a LOWESS fit can be obtained as depicted by the red line in Figure 1a. The LOWESS fit is obtained as follows. The smooth y value is found by taking the $N{*}frac$ closest points to the real y value of the target bin and calculating the mean. Here, *frac* is the proportion of the chromosome that should be considered when determining the smooth value, a larger *frac* leads to a smoother line. By calculating the smooth y value for each bin, a smooth line is found through each chromosome, indicating the LOWESS fit on each bin's GC content. The read counts can then be GC-normalized by dividing the read counts by their corresponding LOWESS value:

$$\text{GC-normalized read count} = \text{RC}_i/\text{GC}_i$$

Where RC_i is the read count for bin i and GC_i is the G and C nucleotide counts for bin i. Figure 1b shows the GC normalized read counts for the sample in Figure 1a.

2.2 WISECONDOR

A drawback of previous NIPT methods was the need for healthy samples to which the target sample is compared. To keep the experimental influences as low as possible, the healthy samples have to be re-sequenced



(a) Read count per bin of size 1 million base pairs. The number of G and C's are counted per bin and used to obtain the LOWESS fit (red line).

(b) Read count normalized for the GC bias.

Figure 1: Read counts are corrected for the GC bias. The number of G and C nucleotides within each bin are counted, from which the LOWESS fit can be obtained (indicated by the red line in (a)). The read counts per bin are corrected by dividing the read count by the LOWESS fit. Through this process, the GC normalized read count as depicted in (b) is obtained.

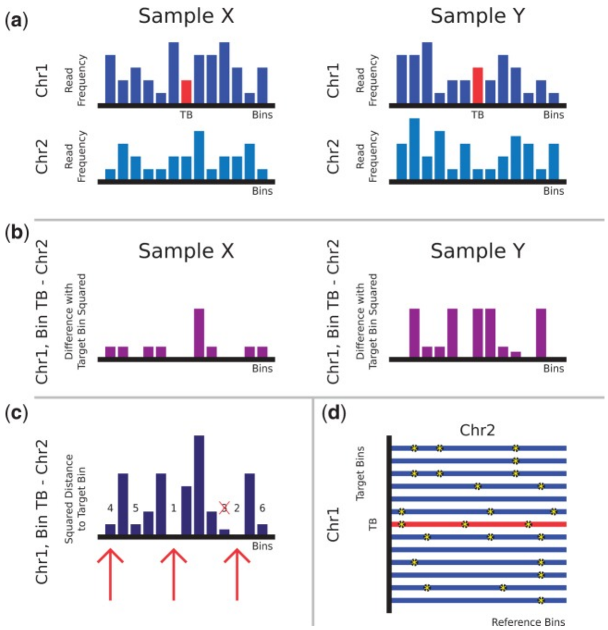


Figure 2: Finding within-sample reference bins by WISECONDOR. (a) Shows an area on chromosomes 1 and 2 for two normal (diploid) samples X and Y. The red bar is the target bin (TB) for which a set of reference bins is to be determined. (b) Squared differences between target bin TB and each of the bins on chromosome 2 for both samples. (c) Summation of the squared differences between target bin TB and each of the bins on chromosome 2 over both samples. Numbers show the similarity ranking of the bins with respect to target bin TB. Red arrows indicate the bins chosen for target bin TB to be included in the set of reference bins. (d) Stars on each row illustrate selected reference bins on chromosome 2 for every bin of chromosome 1.

for each new target sample, increasing the costs. To overcome this drawback, Straver et al. created WISECONDOR: a WIthin SamplE COpy Number aberration DetectOR(13).

WISECONDOR uses a Z-score to determine whether an area is aberrated or not. Instead of using separate healthy samples to compare to, WISECONDOR determines a within-sample reference set for each target bin.

A schematic representation of how the within reference bins are determined can be seen in Figure 2. The genome is divided into bins of size B and the GC normalized read count is determined (Figure 2a). The Euclidean Squared Distance between the target bin to every bin located on the other chromosomes is calculated (Figure 2b) and the bins with the smallest distance to the target bin are selected. If selected bins are neighbouring, only the bin with the smallest distance is kept(Figure 2c).

Once the within reference set is found, z-scoring is applied on a sliding window. The Stouffer's z-score is used on a window of bins:

$$z_i^w = \frac{\sum_{k=i-v}^{i+v} z_k}{\sqrt{2 \cdot (v+1)}}$$

where z_i^w is the sliding window z-score for bin i when considering v bins on either side of bin i and z_k is the z-score of bin k individually. A bin is considered aberrated when the absolute value of the sliding window is larger or equal to 3.

To detect chromosomal aneuploidy, the user defines a threshold for the ratio of bins that need to be aberrated on a chromosome for aneuploidy to be present.

2.2.1 Results

In their paper, WISECONDOR was tested on 56 samples. This test set contained:

- Eight samples with trisomy 21
- Two samples with trisomy 13
- Two samples with trisomy 18
- Two samples with trisomy 22
- Four sample with subchromosomal variants

For aneuploidy detection, 0.5 of the bins had to be aberrated for aneuploidy to be present. With this threshold, all 14 cases of trisomy were identified.

Of the four cases used to test subchromosomal classification by WISECONDOR, three cases were correctly identified by the sliding window method of WISECONDOR. WISECONDOR could not detect the fourth case. This was most likely because of the combination of very low coverage, low fetal fraction, and mosaism.

Though WISECONDOR caught most cases, there were also some false positives. These false positives were relatively small, however, with the largest one being 13Mb.

2.3 LSTM

The goal is to determine for each bin whether it is aberrated. Three factors that could signal a possible aberration are:

- *The read count for the target bin*: a high read count could indicate a duplication whereas a low read count could result from a deletion on this bin. To determine whether this read count is high or low, a reference value is needed.
- *The read count of the previous bin*: the read count of the target bin can be compared to the read count of the previous bin. If an aberration starts at the target bin, the difference between the two bins will be significant.
- *Information on other previous bins*: if an aberration spans multiple bins, comparing them to each other might not reveal this aberration. For example, when a trisomy is present, the read counts within one chromosome will be roughly similar, but comparing them to a bin on the previous chromosome may show an increase in read counts.

One of the most suitable models to incorporate all three factors is a Recurrent Neural Network: a Long-Short Term Memory Model. In the remainder of this section, the proposed model architecture and the rest of the pipeline will be explained.

2.3.1 Preprocessing

Each sample's data is saved in a numpy file containing the GC corrected read count per bin, where each bin consists of 10.000 base pairs (bp). This data must be scaled to the desired window by adding up the read counts within the window. By design, this window has to be a factor of 10.000. The class -healthy or aberrated- of this new window, is decided by which of the two classes is assigned to the majority of the 10.000bp bins within the window.

Of the binned read counts per sample, the bins belonging to the X and Y chromosome are removed for two reasons. First of all, because the read count of chromosome X and Y is heavily dependent on both the fetal fraction and the gender of the fetus, the read count of the X and Y chromosome can vary wildly, regardless of aberrations. Secondly, WISECONDOR, the method against which the model will be benchmarked, omits the gender-specific chromosomes. By removing them here as well, we can straightforwardly compare both methods.

Next, the read counts are masked for unmappable regions. Reads are aligned to a location if that alignment is unique. The human genome has repetitious regions, for such regions no unique alignment can be found, and therefore no reads will be mapped there: i.e. they are unmappable (18). Unmappable regions are identified by determining the read count per base pair for 100 samples. If in over half the samples no reads were mapped to a base pair, that base pair is categorized as unmappable.

Lastly, the class imbalance in the data has to be addressed. Though there is an equal amount of healthy and aberrated samples in most simulated data sets, there is an imbalance of labels within a single sample. For aberrated samples, only a fraction of the labelled bins has label 1. For example, using a bin size of 1 million, a sample with trisomy on chromosome 21 has 49 bins with label 1 and 2833 bins with label 0. This means only $49/2882 = 1,7\%$ of the bins is labelled 1. To compensate for this imbalance, sample weights are calculated and assigned to each bin in each sample. These weights are calculated by solving $\sum^{\#0} w_0 = \sum^{\#1} w_1$, where w_0 is the weight for bins labelled 0 and w_1 is the weight for bins labelled 1.

2.3.2 Model

This thesis aims to create a deep learning model that takes the sequence of read counts per bin as input and outputs whether each bin has been aberrated. For this goal, an LSTM is chosen.

A Long-Short Term Memory Model is a version of a Recurrent Neural Network designed to solve the lack of long term memory in existing RNNs. Its architecture consists of a cell and three 'gates' that regulate the information within the LSTM unit. The three gates are known as; the input gate, the output gate, and the forget gate. A schematic representation of three LSTM cells and their gates can be seen in Figure 3.

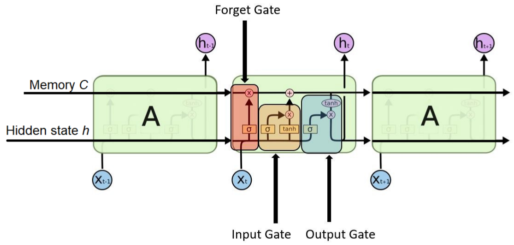


Figure 3: A schematic representation of three cells of an LSTM. The top line indicates the long-term memory through the cells and the bottom line indicates each previous cell's hidden state. In red, the Forget Gate operations are shown, in yellow the operations of the Input Gate and in blue the operations of the Output Gate. (19)

The sequence of GC normalized read counts per bin is the model's input, which maps each of the bins to either 0 for healthy or 1 for aberrated. Within the LSTM cell, the forget gate decides what information from previous bins should be kept in the long-term memory and what information currently being kept can be forgotten. This updated long term memory is then passed on to the input gate. This gate first decides what information in the long term memory should be updated and then creates a vector of new candidate values, combines and adds them to the long term memory. Last, the read count of the current bin, the hidden state of the previous bin, and the long-term memory are used as input for the output gate, which decides the current target bin's state. This state is used as the output for this bin and is passed along as short term memory for the next bin.

For the LSTM layer(s) a Bidirectional LSTM layer is used. A Bidirectional Layer creates two copies of the layer: one copy where the input sequence is used in the order as-is and one copy where the input sequence is reversed. The results from both layers are then concatenated and used as output for the layer. This allows the model to decide based on the bins that came earlier and bins that come after the target bin. Each LSTM layer consists of a cell per bin on the genome. The optimal amount of LSTM layers in the model has been empirically set using the dataset used for Experiment 1. First, a set of possible values for the number of layers was chosen. Next, for each value, the model was trained 100 times, and the performance on the training and test set were determined. Results were compared, and the optimal value was chosen. The results of this test can be found in Supplementary Figure S1. The tests showed that one LSTM layer works best. This LSTM layer used the ReLU activation function (20).

A decision has been made to make the LSTM stateful, meaning that the end value of the previous iteration is used as the initial value for each iteration. This allows for the model to start where it left off at the previous iteration.

The LSTM layer's output is then used as input for a Time Distributed Dense layer, which serves as the output layer. Here the Time Distributed layer allows the model to use the same Dense layer with the same trained weights on each 'time step' in the input. In this case, each 'time step' represents one bin on the genome. For the output layer, the Sigmoid activation function assigns a label between 0 and 1 to each bin.

Since this is a binary classification problem between 0 and 1, binary cross-entropy is chosen as the loss function. The model is optimized using Adam (21).

As described earlier, hyperparameter tuning was performed to determine the optimal number of epochs and dropout rate. The results can be found in Supplementary sections S2 and S3. The tests determined that the model performed best with a dropout rate of 0. Therefore, no dropout layer is added to the model. 10.000 epochs has shown to be the optimal value. It has been shown that when the number of epochs during training is increased, the performance on the training set increases, but the performance on the test set decreases, this signals that the model is overfitting on the training data.

2.3.3 Post-process

For each bin in the input sequence, the model predicts the probability of that bin belonging in class 1. If this probability is higher than some cutoff point x, the class label 1 is given to that bin, otherwise class label 0 is assigned. As described before, hyperparameter tuning is done to determine the optimal cutoff point. The result of this can be found in Supplementary Figure S4. It shows that though a lower cutoff point has a higher sensitivity, it decreases in precision. The cutoff point of 0.5 is chosen as the optimal parameter.

Once all bins have acquired their labels, the prediction can be fine-tuned. The user can define a threshold for the minimal number of bins an aberration should span to be called a true aberration. Through this threshold calls made on small peaks in the read count can be filtered out.

2.4 Experiments

To test the performance and the limits of the LSTM model, multiple experiments were set up, each validating a different aspect of the model and its influence on the NIPT model performance.

2.4.1 Experiment 1

The first experiment focuses on the overall performance of the model. The dataset is divided into a training and a test set, where 20% of the data will be in the test set and the other 80% in the training set. As mentioned in Section 2.1.2, the data consists of both healthy and aberrated samples, where the aberrated samples contain aberrations of different sizes on chromosome 13, 18, 19, and 21. The test set is created to have an equal number of samples of each 'group'. Meaning that of the test set, 20% of the samples consist of healthy samples, 20% of the samples has an aberration on chromosome 13, and so forth. Within the group of samples with an aberration on each chromosome, there are an equal number of samples with aberration size x as there are of size y. Likewise, the training set has an equal number of samples from each group.

To decrease the influence of specific files on both models' performance, the data splitting process is repeated once to create two pairs of training and tests sets that consist of the same data distribution but different files in the test and training sets. The model's performances on both datasets are used to compare the two algorithms.

2.4.2 Experiment 2 - 6

Each of the other experiments focuses on tuning a different characteristic of NIPT:

- The size of the aberration
- The coverage of the sample
- the fetal fraction
- The bin size
- The variant in the sample

For each of these experiments, the training and test sets are generated as follows. The test set will consist of 20% of the data, and the other 80% is added to the training set. The test set contains an equal amount of samples for each of the values of the target characteristic:

- Experiment 2 *aberration size*: there are 1000 healthy samples and 1000 aberrated samples, with aberrations varying from 1Mb to 49Mb. Out of each aberration size, 20 samples are available. Five of these are added to the test set, while the other 15 are added to the training set. This adds up to 100 aberrated samples in the test set. One hundred healthy samples are added randomly to maintain the desired distribution.
- Experiment 3 *coverage*: there are 1000 samples, both aberrated and healthy. Of the coverages between 5M and 85M read per sample, 120 samples are available per coverage, of which 11 aberrated and 11 healthy samples per coverage are added to the test set.
- Experiment 4 *fetal fraction*: the fetal fractions range from 1% to 20%. There are 100 samples per fetal fraction, both healthy and aberrated. Per fetal fraction, ten aberrated samples and ten healthy samples are added to the test set.
- Experiment 5 *bin size*: the training and test sets of Experiment 1 are re-used.

- Experiment 6 *variant*: for both deletions and duplications, there are 250 samples with varying aberration sizes. Of each aberration size of the 20 aberration sizes, three samples are added, adding up to 60 samples per variant. This is paired with 60 healthy samples to create the test set.

These steps are repeated for every experiment to avoid data skew based on specific files. By having an equal number of samples for each value, the results can be compared directly.

The performance of a model relies heavily on the initialization of the weights. Therefore, the performance of two separately trained models can vary greatly. To account for this variation in our experiments, every model is trained on the same dataset 100 times to capture these models' range of performance. This will be discussed in Section 4.

2.5 Evaluation

2.5.1 Metrics

As mentioned before, the labels in the training data are heavily imbalanced. Most data sets have an equal amount of healthy and aberrated samples, but the labels are assigned per bin, not per sample. This means that aberrated samples have a combination of 0 and 1 labels. For example, using the data of experiment 1 and a bin size of 1 Mb, there are 5.764.000 bins, of which 5.694.490 bins are healthy and 69.510 are aberrated. If a model predicts only the larger class, healthy, the confusion matrix would result in:

	Classified positive	Classified negative
Positive	TP 0	FN 69.510
Negative	FP 0	TN 5.694.490

Table 2: Confusion matrix if the model only predicts healthy for each bin

This results in an accuracy of $\frac{0+5.694.490}{0+5.694.490+0+69.510} = 98,8\%$ even though it predicted the larger class for all bins. This shows that accuracy is unfit as a metric for an imbalanced dataset. Therefore the model will be assessed through other metrics.

Instead of accuracy, the following metrics have been chosen to be used to determine the performance of the model (22): Sensitivity, Specificity, Precision, Youden's index, and Matthew's Correlation Coefficient. These are calculated as shown in Table 1.

Youden's index is the difference between the true positive rate and the false positive rate. It gives equal importance to the positive and negative classes regardless of the size of each class. Looking at the first example again where everything is classified as negative, Youden's Index equates to:

$$\gamma = 0 - (1 - 1) = 0$$

Matthew's Correlation Coefficient (23) is a correlation coefficient between the observed and predicted when dealing with a binary classification problem. It maps the true and false positives and negatives between -1 and 1, where -1 means that the prediction is completely different from the true label, 0 means the prediction is equal to a random prediction and 1 represents a perfect prediction. For the example of experiment 1 MCC equals:

$$MCC = \frac{0 \cdot 5.694.490 - 0 \cdot 69.510}{\sqrt{(0+0)(0+69.510)(5.694.490+0)(5.694.490+69.510)}} = \frac{0}{0} \longrightarrow 0$$

Of the two metrics, Youden's index and MCC, MCC will be the most important. Though Youden's index uses the true positive and false positive **rates** instead of directly calculating the amount of correctly classified base pairs as is done when calculating accuracy, this still does not take the imbalance between the classes enough into account. When using specificity, the number of false positives only influences the true negative rate. With this dataset, the number of true negatives will often be around 100x as large as the number of the other classes, which leads to a high specificity regardless of the number of false positives. This class imbalance does not influence the MCC since it uses all classes in one calculation.

2.5.2 Experiments

For the experiments, the model is trained, and the true and false positives and negatives predicted for the test set for each of the varying characteristics are counted from which the metrics can be calculated. As mentioned in 2.5.1 MCC is the most important metric.

For experiment 1, the overall performance of the model is tested first. True and false positives and negatives of each sample are added together, and the Youden's index

Metric	Formula	Performance
Sensitivity (Recall)	$\frac{TP}{TP+FN}$	Sensitivity = 0 poor, Sensitivity = 1 good
Specificity	$\frac{TN}{TN+FP}$	Specificity = 0 poor, Specificity = 1 good
Precision	$\frac{TP}{TP+FP}$	Precision = 0 poor, Precision = 1 good
Youden's index	γ = sensitivity - (1 - specificity)	γ = -1 poor, γ = 1 good
Matthew's Correlation Coefficient	$\frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$	MCC = -1 worst, MCC = 0 equal to random prediction, MCC = 1 perfect

Table 1: Metrics for performance assessment

and MCC can be calculated. The highest MCC value will be taken as the overall performance of the model. Besides the overall performance, the performance per aberrated chromosome will be determined as well. The samples are divided into four groups, each containing files where all aberrations are on the same chromosome. From this, the model's ability to detect aberrations on each of the chromosomes can be analyzed separately. For each of these chromosome groups, the MCC will be calculated again. First, the individual best score will be determined as the highest MCC value reached by any of the trained models. This individual score will then be compared to the MCC score for each chromosome reached by the model that did the best overall chromosomes.

For experiments 2-6, the datasets each contain an equal number of samples for each target value. The MCC value for each is calculated and compared. Two values are taken to compare to the results of WISECONDOR: the individual best and the overall best. The individual best is the highest MCC value for that target value reached by any trained model, regardless of how it did on the other target values. The overall best model is the model with the highest overall MCC score. So for experiment 2 the individual highest MCC value for files with an aberration of size 30Mb might be 0.8 by model *a*, but model *a* did not do well for other aberration sizes. Model **b** however had an MCC score of 0.7 on aberration size 30Mb, but did better on the other aberrations sizes, giving the overall performance of the model a higher score.

WISECONDOR is run on the same test sets for each experiment. The within-sample reference set is created using the negative samples from experiment 1. This reference set is then used in all experiments to calculate the Z-score.

3 Results

In this section, the results of the experiments will be shown. For each experiment, the figures for the MCC value, sensitivity (recall), and precision are shown in this section. The other metrics' figures can be found in Supplementary Figures S5 - S10.

3.1 Experiment 1

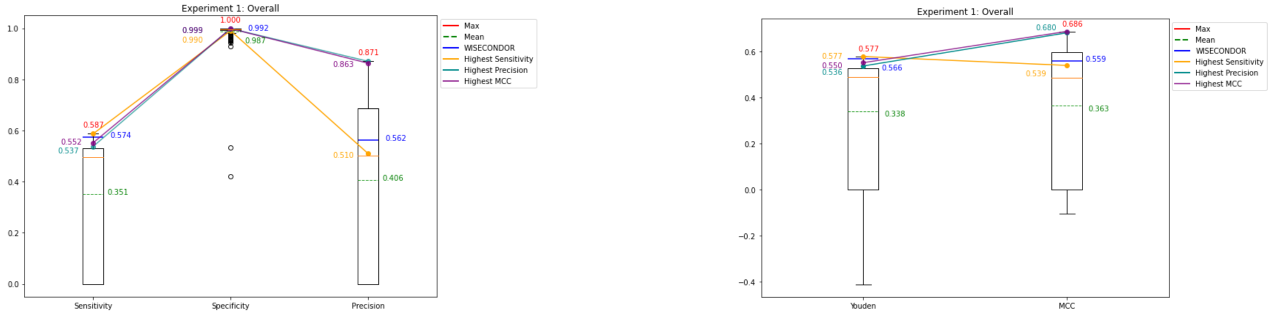
In Figure 4 a boxplot of the sensitivity, specificity, and precision and a boxplot of the Youden's index and MCC score show the results of experiment 1. Here the red, green, and blue indicate the highest and average value of the LSTM, and the performance of WISECONDOR, respectively. The orange, cyan, and purple lines depict the performance of the model with the highest achieved sensitivity, the highest achieved precision, and the model with the highest MCC value: the best scoring model overall.

Looking at Figure 4a, we see that WISECONDOR and the LSTM score similarly for sensitivity and specificity. However, this sensitivity is still relatively low. WISECONDOR, the best LSTM model, and the individual best sensitivity all lie around 0.57. This means that little over half the aberrations are found. Combining this with the precision, we see that WISECONDOR not only misses almost half the aberrations, it also predicts just as many false positives. The LSTM has higher precision, both the individual best and the overall best model scoring around 0.87, which means that the LSTM also finds only half of the aberrations, but far fewer false positives.

In Figure 4b sensitivity, specificity, and precision are combined in the metrics Youden's index and MCC. This Figureshows that the overall performance of the LSTM and WISECONDOR on this dataset is 0.686 and 0.559, respectively.

As indicated in section 2.4.1, the samples are split into four groups where each group contains the samples where the true aberration is on chromosome 13, 18, 19, and 21 respectively. In Figure 5 the sensitivity, precision, and MCC value results per chromosome can be seen. The red numbers indicate the individual best score per chromosome, green the average, blue WISECONDOR, and purple the overall best LSTM model. Starting with the sensitivity, we can see that WISECONDOR and the LSTM follow the same trend. Chromosomes 18 and 19 are the easiest to detect, followed by chromosome 13, with chromosome 21 trailing far behind. For chromosome 21, the best working model only calls 38% of the aberrations present. WISECONDOR and the best overall model detect only 13 and 8%, respectively. Notably, for all chromosomes there exists a model that finds more true positives than WISECONDOR. However, when looking for one model that can accurately detect aberrations on all chromosomes, WISECONDOR does outperform the best LSTM model on all chromosomes except 18. When we combine this with their precision scores, we see that even though WISECONDOR finds more true positives than the best LSTM model, it also finds more false positives. In the case of chromosome 21, WISECONDOR even finds more false positives than true positives. From this, we can conclude that though the LSTM may not detect many aberrations on some chromosomes, the aberrations it does call are accurate.

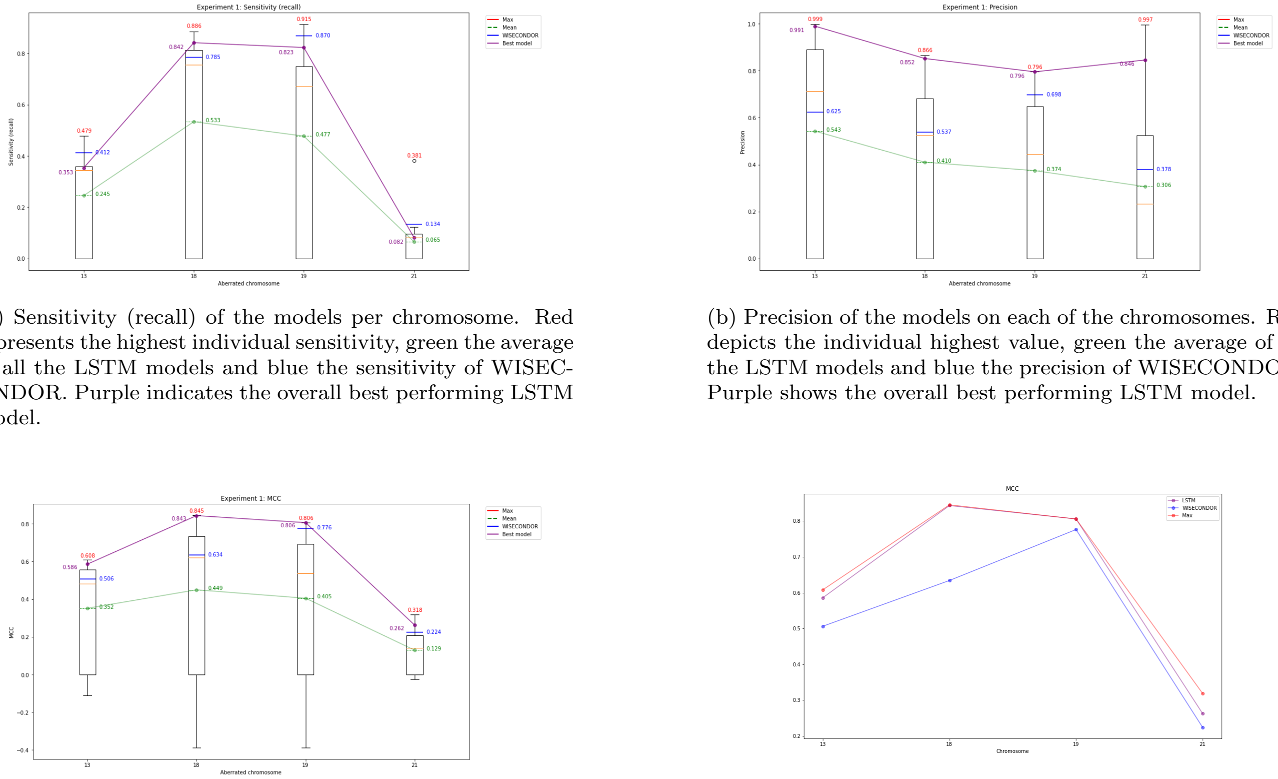
In Figures 5c and 5d the Youden's index and MCC value of WISECONDOR and the LSTM are compared to each other. Per chromosome, the LSTM again does better than WISECONDOR. Noticeably, the performance of both the LSTM and WISECONDOR is drastically lower for chromosome 21 compared to chromosomes 13, 18, and 19. This will be discussed in further detail in section 4.



(a) Sensitivity, specificity, and precision of the LSTM and WISECONDOR in Experiment 1.

(b) Youden's index and MCC score of the LSTM and WISECONDOR in Experiment 1.

Figure 4: (a) depicts the sensitivity, specificity, and precision of experiment 1 and (b) shows the Youden's index and MCC value for experiment 1. In both figures red and green indicate the highest and average values from the LSTM respectively, blue indicates WISECONDOR's performance, orange shows the model with the highest sensitivity, cyan the model with the highest precision, and purple the model with the highest MCC score (the best model).



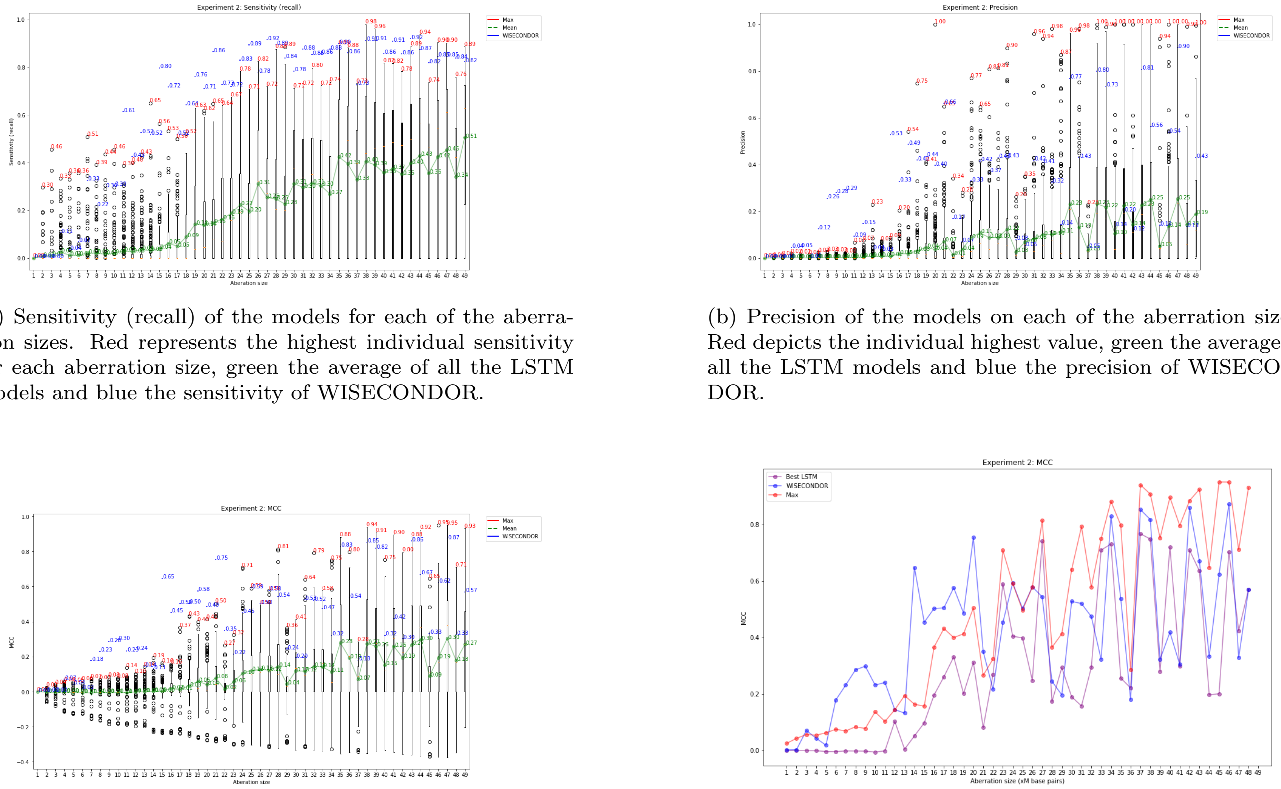
(a) Sensitivity (recall) of the models per chromosome. Red represents the highest individual sensitivity, green the average of all the LSTM models and blue the sensitivity of WISECONDOR. Purple indicates the overall best performing LSTM model.

(b) Precision of the models on each of the chromosomes. Red depicts the individual highest value, green the average of all the LSTM models and blue the precision of WISECONDOR. Purple shows the overall best performing LSTM model.

(c) The MCC values for each chromosome in Experiment 1. Red indicates the highest individual value, green the mean LSTM value, blue indicates WISECONDOR, and purple indicates the overall best performing LSTM model.

(d) The MCC value of Experiment 1 per chromosome of the best LSTM model (purple), WISECONDOR (blue) and the individual best LSTM model per chromosome (red)

Figure 5: Results for Experiment 1 per chromosome. The x-axis shows aberrated chromosome. The y-axis depicts (a) the sensitivity (recall), (b) the precision, (c) the MCC value, and (d) also the MCC value. The red numbers and lines represent the best individual model per chromosome, green the average of the LSTM models, blue depicts WISECONDOR's performance and purple the best overall performing LSTM model.



(a) Sensitivity (recall) of the models for each of the aberration sizes. Red represents the highest individual sensitivity for each aberration size, green the average of all the LSTM models and blue the sensitivity of WISECONDOR.

(b) Precision of the models on each of the aberration sizes. Red depicts the individual highest value, green the average of all the LSTM models and blue the precision of WISECONDOR.

(c) The distribution of the MCC values achieved by the LSTM models. Red shows the individual highest value for each aberration size, green the average of the LSTM models an blue WISECONDOR's score.

(d) MCC value for each of the aberration sizes of the overall best performing model (purple), WISECONDOR (blue) and the individual best LSTM model (red)

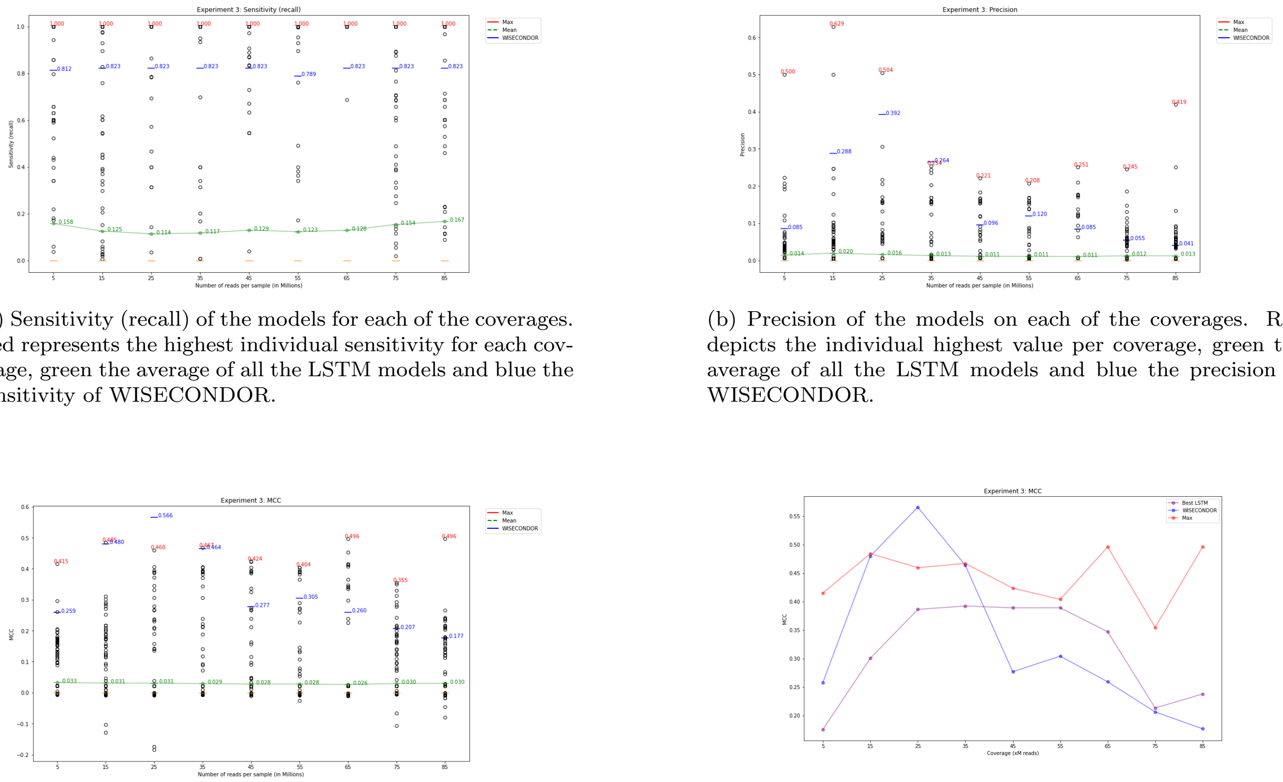
Figure 6: Results for Experiment 2: Aberration size. The x-axis shows the aberration size in millions of base pairs. The y-axis depicts (a) the sensitivity (recall) for the various aberration size, (b) the precision, (c) the MCC value and (d) also the MCC value. The red numbers and lines represent the best individual model for each aberration size, green the average of the LSTM models, blue depicts WISECONDOR's performance and purple the best overall performing LSTM model.

3.2 Experiment 2: Aberration size

In Figure 6 and Supplementary Figure S6 the results of experiment 2 can be seen. In this experiment, the aberration size ranges from 1Mb to 49Mb (trisomy) on chromosome 21. First looking at the sensitivity for each aberration size in Figure 6a, it can be seen that the sensitivity increases as the aberration size grows. This seems logical since a bigger aberration should be easier to detect. The LSTM model has a higher sensitivity than WISECONDOR for the smaller aberration sizes, but slightly worse for the larger aberrations. Precision (Figure 6b) also increases as the aberration size increases for the LSTM model. WISECONDOR's precision is higher for the smaller aberrations sizes, but when the aberration sizes increase, precision is inconsistent and varies wildly for aberration sizes only 1 Mb apart in size. This is reflected in the MCC score in figures 6c and 6d as well. Here WISECONDOR does better for aberrations sizes smaller than 21Mb. From 21Mb onward, the individual best for each aberration size is better. However, looking at the best model overall aberration sizes, it rarely performs better than WISECONDOR.

3.3 Experiment 3: Coverage

In Figure 7 and Supplementary Figure S7 the results for experiment 3 for various coverages is shown. Here the number of reads per sample is indicated on the x-axis, where 5M reads corresponds to a coverage of 0.05x and 85M reads to 1x coverage. In Figure 7a and 7b the sensitivity and precision are shown. We see that the sensitivity is very consistent for both the highest individual value, the average, and WISECONDOR. The highest individual value is 1.0 for all coverages, which means that it found all aberration present on all samples, which would be optimal. However, when we look at the precision, even if the highest individual sensitivity belongs to the same model as the highest precision, the model predicts the same amount of false positives along with the true positives. WISECONDOR follows that same trend, the sensitivity is very consistent around 0.82, but its precision is 0.39 at most. These two scores are combined in the MCC value, shown in figures 7c and 7d. These figures show the distribution of MCC values achieved by the LSTM, and the MCC value for the highest individual LSTM model per coverage, the overall best LSTM model, and WISECONDOR. Here we see that WISECONDOR does best for a coverage of 25M reads per sample with an



(a) Sensitivity (recall) of the models for each of the coverages. Red represents the highest individual sensitivity for each coverage, green the average of all the LSTM models and blue the sensitivity of WISECONDOR.

(b) Precision of the models on each of the coverages. Red depicts the individual highest value per coverage, green the average of all the LSTM models and blue the precision of WISECONDOR.

(c) The distribution of the MCC values achieved by the LSTM models. Red shows the individual highest value for each coverage, green the average of the LSTM models an blue shows WISECONDOR's score.

(d) MCC value for each of the coverages of the overall best performing model (purple), WISECONDOR (blue) and the individual best LSTM model (red)

Figure 7: Results for Experiment 3: Coverage. The x-axis shows the various coverages noted in the number of reads per sample. The y-axis depicts (a) the sensitivity (recall) for the various coverages, (b) the precision, (c) the MCC value and (d) also the MCC value. The red numbers and lines represent the best individual model for each coverage, green the average of the LSTM models, blue depicts WISECONDOR's performance and purple the best overall performing LSTM model.

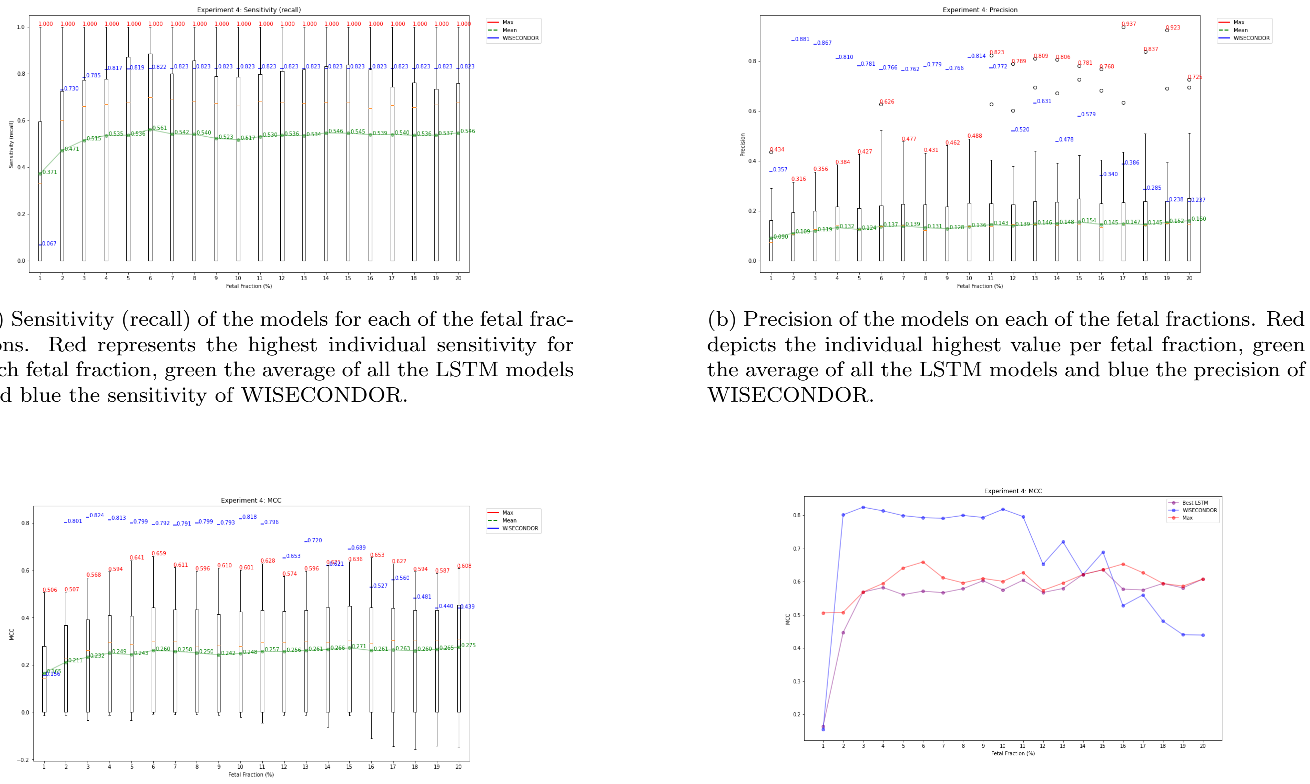
MCC value of 0.566 versus 0.460 of the best LSTM. For the higher coverages, the LSTM outperforms WISECONDOR. WISECONDOR performing best at 25M reads per sample might be because the reference set used in calculating the z-score was determined using the dataset from experiment 1. In this dataset, all samples have a coverage of 20M reads. This will be discussed in section 4.

3.4 Experiment 4: Fetal Fraction

In Figure 8 and Supplementary Figure S8 the results for experiment 4 can be seen. Figure 8a shows the sensitivity (recall) of the LSTM models and WISECONDOR. The sensitivity of both models is quite consistent. WISECONDOR's sensitivity lies above 0.73 and mostly around 0.82 for fetal fractions above 1%. The highest sensitivity for each fetal fraction individually is 1.0 for all fetal fractions, so for each fetal fraction, a model exists that can find all aberrations. Figure 8b shows that for samples with a fetal fraction of 11% or higher, a model can find the aberration with precision around 0.8 or higher. However, these could very well be two different models. WISECONDOR's precision is high for lower fetal fractions and drops drastically when the fetal fraction increases. A large influence on this is that the within-sample reference set is created for a dataset where all samples have a fetal fraction of 10%. This will be discussed further in section 4. If we look at the MCC value in figures 8c and 8d we see that the MCC value of WISECONDOR follows the same line as its precision. It does better at the lower fetal fractions above 2% but gradually worsens with an increasing fetal fraction. For the LSTM we can see that the highest sensitivity and the highest precision are not from the same model. The MCC value of the LSTM, both individually highest and overall best, is consistent for all fetal fraction above 2%.

3.5 Experiment 5: Bin size

The results of the LSTM model in experiment 5 can be seen in Figure 9, Figure 10, and Supplementary Figure S9. In Figure 9a the sensitivity, specificity, and precision of the model are depicted for each bin size: 2Mb, 1.5Mb, 1Mb, 750kb, 500kb, and 250kb. The orange boxplots show the sensitivity for each bin size, the black boxplots show the specificity and the cyan boxplots show the precision. The red and green numbers indicate the maximum and mean values, respectively. The purple lines and numbers depict the model with the overall highest



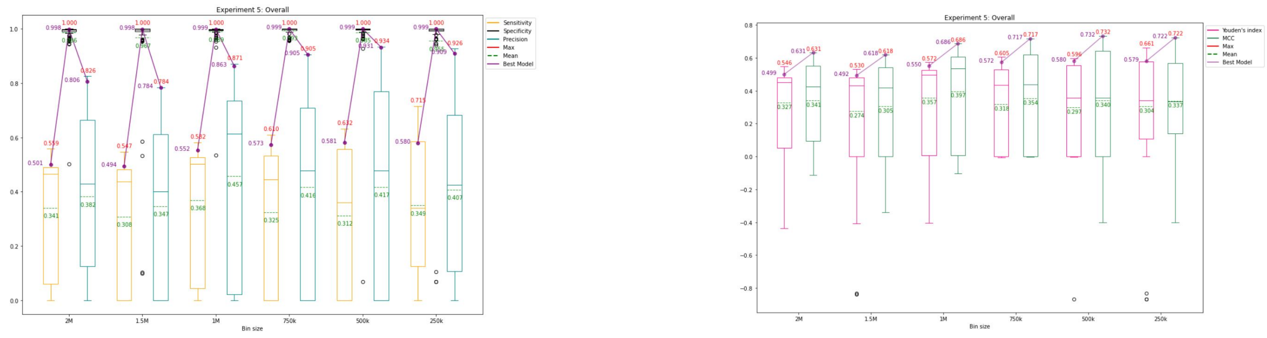
(a) Sensitivity (recall) of the models for each of the fetal fractions. Red represents the highest individual sensitivity for each fetal fraction, green the average of all the LSTM models and blue the sensitivity of WISECONDOR.

(b) Precision of the models on each of the fetal fractions. Red depicts the individual highest value per fetal fraction, green the average of all the LSTM models and blue the precision of WISECONDOR.

(c) The distribution of the MCC values achieved by the LSTM models. Red shows the individual highest value for each fetal fraction, green the average of the LSTM models an blue WISECONDOR's score.

(d) MCC value for each of the fetal fractions of the overall best performing model (purple), WISECONDOR (blue) and the individual best LSTM model (red)

Figure 8: Results for Experiment 4: Fetal Fraction. The x-axis shows the various fetal fractions in percents. The y-axis depicts (a) the sensitivity (recall) for the various fetal fractions, (b) the precision, (c) the MCC value and (d) also the MCC value. The red numbers and lines represent the best individual model for each fetal fraction, green the average of the LSTM models, blue depicts WISECONDOR's performance and purple the best overall performing LSTM model.



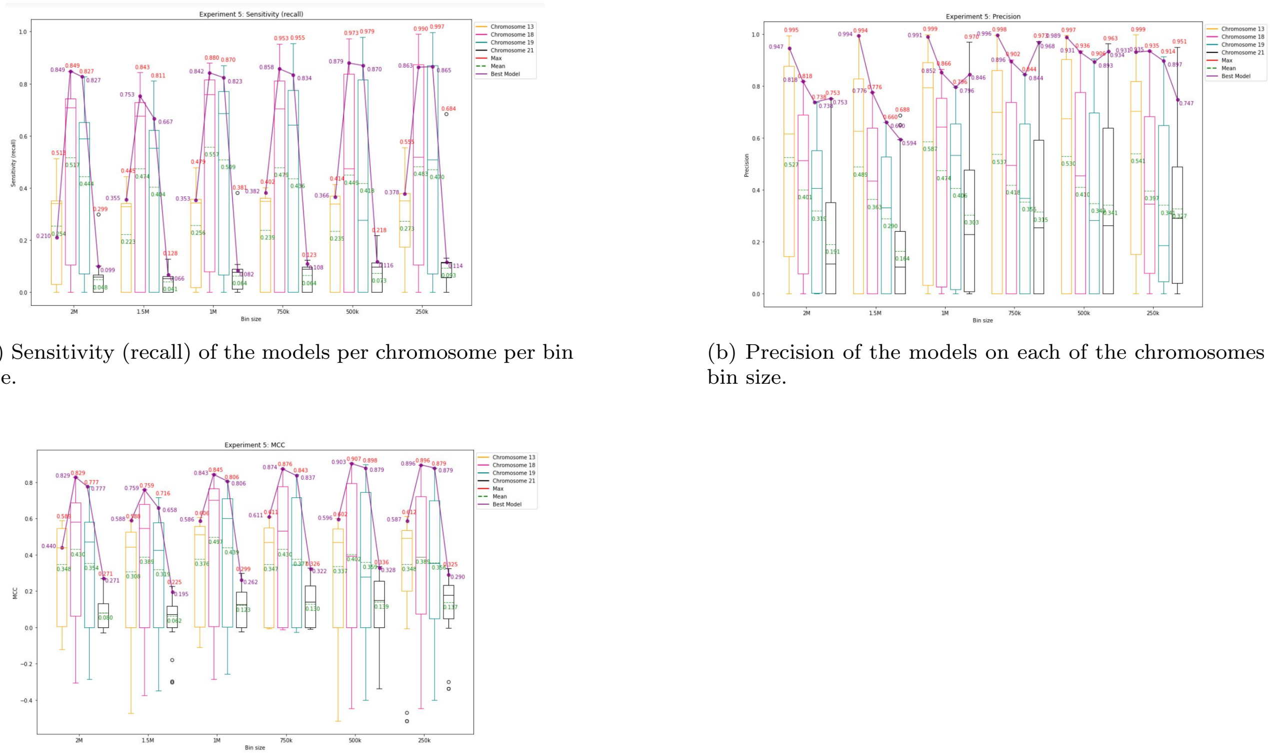
(a) Sensitivity, specificity, and precision of the LSTM in Experiment 5.

(b) Youden's index and MCC score of the LSTM in Experiment 5.

Figure 9: (a) depicts the sensitivity, specificity, and precision of Experiment 5 and (b) shows the Youden's index and MCC value for Experiment 5. In both figures, red and green indicate the highest and average values from the LSTM respectively, and purple indicates the overall highest performing model. In (a) the orange boxplot shows the sensitivity of the model for each bin size, the black boxplot shows the specificity and the cyan boxplot shows the precision. In (b) the pink boxplot shows the Youden's index and the green boxplot the MCC values.

MCC value, which is the best performing model. It can be seen that as the bin size decreases, the sensitivity and precision increase. The models using a bin size between 2Mb and 1Mb have a sensitivity around 0.56 and a precision around 0.82. The models using a bin size between 750kb and 250kb achieve a sensitivity above 0.6 and a precision over 0.9.

Figure 9b depicts the Youden's index and MCC values of the models for each bin size. The pink boxplot shows the Youden's index and the green boxplot the MCC value in this figure. The red, green, and purple numbers indicate the maximum, mean, and overall best performing model. It can be seen that the MCC value increases as the bin size decreases, indicating that the model per-



(a) Sensitivity (recall) of the models per chromosome per bin size.

(b) Precision of the models on each of the chromosomes per bin size.

(c) The MCC values for each chromosome for each bin size in Experiment 5.

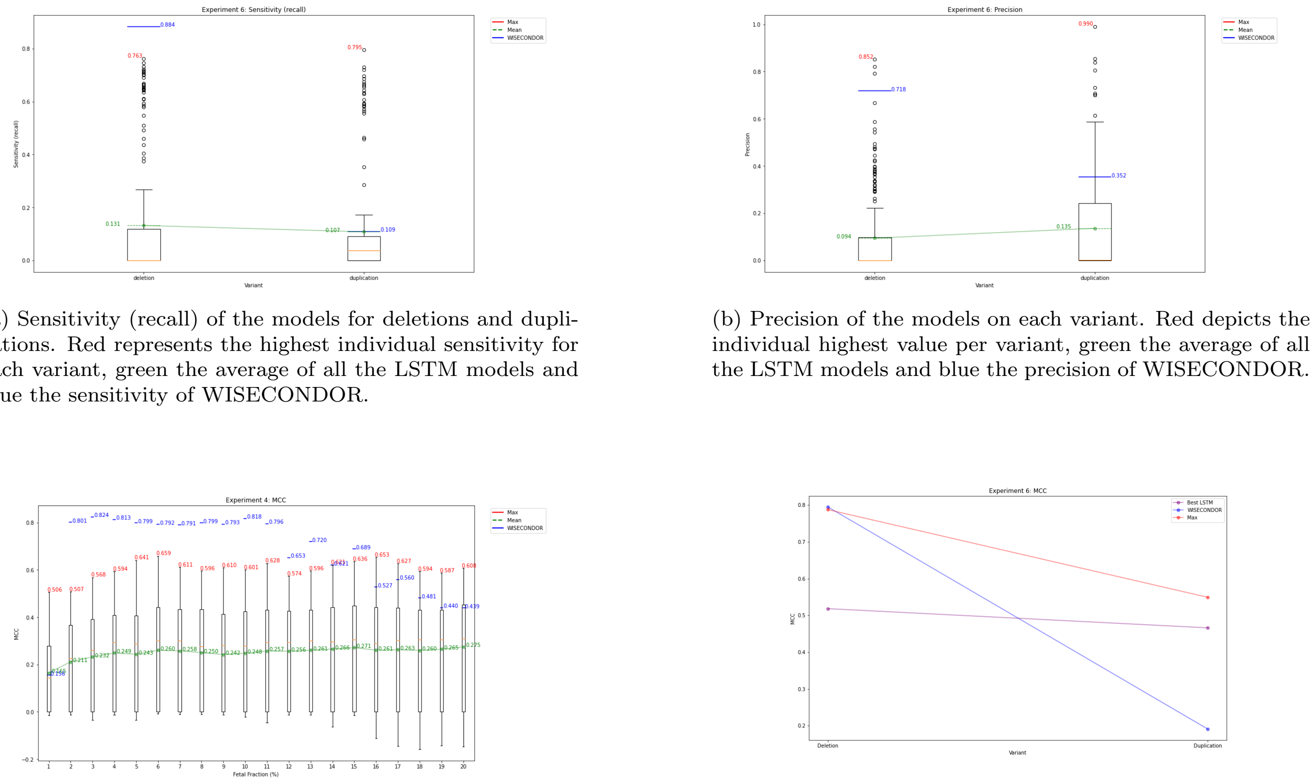
Figure 10: Results for Experiment 5: Bin size. The x-axis shows the various bin sizes and per bin size it shows each of the chromosome 13, 18, 19, and 21. Chromosome 13 is shown by the orange boxplots, chromosome 18 is shown by the pink boxplots, chromosome 19 is shown by the cyan boxplots, and chromosome 21 is shown by the black boxplots. The y-axis depicts (a) the sensitivity (recall) for the various bin sizes and chromosomes, (b) the precision, and (c) the MCC value. The red numbers and lines represent the best individual model for each bin size for each chromosome, green the average of the LSTM models, and purple the best overall performing LSTM model.

forms better as the bin size decreases, reaching at most an MCC value of 0.732 for a bin size of 500kb.

In Figure 10 the sensitivity, precision and MCC value for each bin size are shown per chromosome. In this figure, the orange boxplot depicts chromosome 13, the pink boxplot depicts chromosome 18, the cyan boxplot depicts chromosome 19, and the black boxplot depicts chromosome 21. The maximum, mean, and best overall model are shown in red, green, and purple, respectively. Figure 10a shows that the sensitivity (slightly) increases as the bin size decreases. For chromosome 13 it increases from 0.513 for a bin size of 2Mb to 0.553 for a bin size of 250kb. For chromosome 18 and 19, it increases from 0.849 and 0.827 to 0.990 and 0.997, respectively, and for chromosome 21 it increases from 0.299 to 0.684. However, the highest values for the sensitivity on chromosome 21 do not stem from the best performing model. If we look for a model that does well on both chromosome 21 and the other chromosomes, sensitivity only increases from 0.099 to 0.114. A sensitivity of 0.114 means that the model detects 10% of the aberrations on chromosome 21 at most. In Figure 10b the precision for each chromosome for each bin size is shown. It can be seen that the precision for chromosome 13 is stable with the highest value above 0.994 for each bin size. For chromosomes 18 and 19, the precision increases slightly from 0.818 an 0.738 to 0.935 and 0.914, respectively. The precision for chromosome 21 increases as well, rising above 0.95 for bin sizes of 1Mb and smaller. However, as we have seen in Figure 10a, even if the highest precision belongs to the same model as the highest sensitivity, the models only find 20% of the aberrations at most. In Figure 10c the MCC value for each chromosome for each bin size can be seen. A (slight) increase in MCC value can be seen for chromosome 13, 18, and 19, the latter two reaching an MCC value of 0.896 and 0.879 respectively for a bin size of 250kb. However, for chromosome 21 it can be seen that the highest MCC value stagnates, increasing only from 0.271 for a bin size of 2Mb to 0.325 for a bin size of 250kb.

3.6 Experiment 6: Variant

Figure 11 and Supplementary Figure S10 show the results for experiment 6. In this experiment, the difference in performance between samples with a duplication versus samples with a deletion is tested. First looking at the sensitivity and precision in Figures 11a and 11b, the performance of the LSTM is very similar for both variants. The maximum sensitivity lies at 0.763 and 0.795 for deletions and duplications, respectively, while the maximum precisions lie at 0.852 and 0.990. For WISECONDOR, we see very different performances for each variant. For



(a) Sensitivity (recall) of the models for deletions and duplications. Red represents the highest individual sensitivity for each variant, green the average of all the LSTM models and blue the sensitivity of WISECONDOR.

(b) Precision of the models on each variant. Red depicts the individual highest value per variant, green the average of all the LSTM models and blue the precision of WISECONDOR.

(c) The distribution of the MCC values achieved by the LSTM models. Red shows the individual highest value for each variant, green the average of the LSTM models an blue WISECONDOR's score.

(d) MCC value for each of the variants of the overall best performing model (purple), WISECONDOR (blue) and the individual best LSTM model (red)

Figure 11: Results for Experiment 6: Variants. The x-axis shows the variant type: deletion or duplication. The y-axis depicts (a) the sensitivity (recall) for the two variants, (b) the precision, (c) the MCC value and (d) also the MCC value. The red numbers and lines represent the best individual model for both variants, green the average of the LSTM models, blue depicts WISECONDOR's performance and purple the best overall performing LSTM model.

samples with a deletion, WISECONDOR has a sensitivity of 0.884 and a precision of 0.718. For samples with a duplication, on the other hand, the sensitivity is 0.109 and the precision 0.352. This is shown in the MCC value in figures 11c and 11d as well. For deletion, WISECONDOR has an MCC value of 0.795, but for duplication, it is only 0.191. The LSTM model equally has a better score for deletion than duplication, 0.788 versus 0.549. These results will be discussed further in the next section.

4 Discussion

In this section, we will analyze the results seen in the previous chapter.

A few things have to be noted before we analyze the results. First of all, the unmappable regions were not taking into account during the simulation of the data. While simulating an aberration, the aberration size was given as input for which a start base pair location was chosen at random. This means that the true aberration might be smaller or split into multiple aberrations by an unmappable region. Though these unmappable regions were taken into account when calculating the performance per base pair, decreasing the size or splitting the aberration in multiple smaller aberrations might make it harder to detect.

Furthermore, both WISECONDOR and the LSTM use bins to split up the genome and determine a label for each bin instead of each base pair. The simulated aberrations do not adhere to these bins. If an aberration spans half of a bin, the methods have two choices: add the bin to the aberration or label it as healthy. Half of the bin will either be added to the false positives or the false negatives, respectively.

Last, before the experiments were done, the within-sample reference set for WISECONDOR had to be created. This was done on the negative samples in the dataset for experiment 1, meaning that the within-sample reference set was created on data with a coverage of 20M reads per sample and a fetal fraction of 10%. Therefore WISECONDOR will likely perform better in experiments 3 and 4 for coverages and fetal fractions closer to this value.

Experiment 1 showed that the overall best performing LSTM works better than WISECONDOR. Though both methods had a sensitivity around 0.57, finding only slightly more than half the aberrations, the LSTM model had a higher precision, indicating that of the aberrations found 86% is correct as opposed to the 56% of WISEC-

ONDOR. When split out over the chromosomes, it became apparent that both methods do well on chromosomes 13, 18, and 19, but considerably worse on chromosome 21. A sensitivity of 0.082 and 0.134 for the LSTM and WISECONDOR, respectively, means that both methods find less than 15% of the true aberrations. In addition to finding only a small amount of the true aberrations, WISECONDOR also found a considerable amount of false positives. The LSTM was relatively precise with a precision of 0.846, though this model found only 8% of the true aberrations.

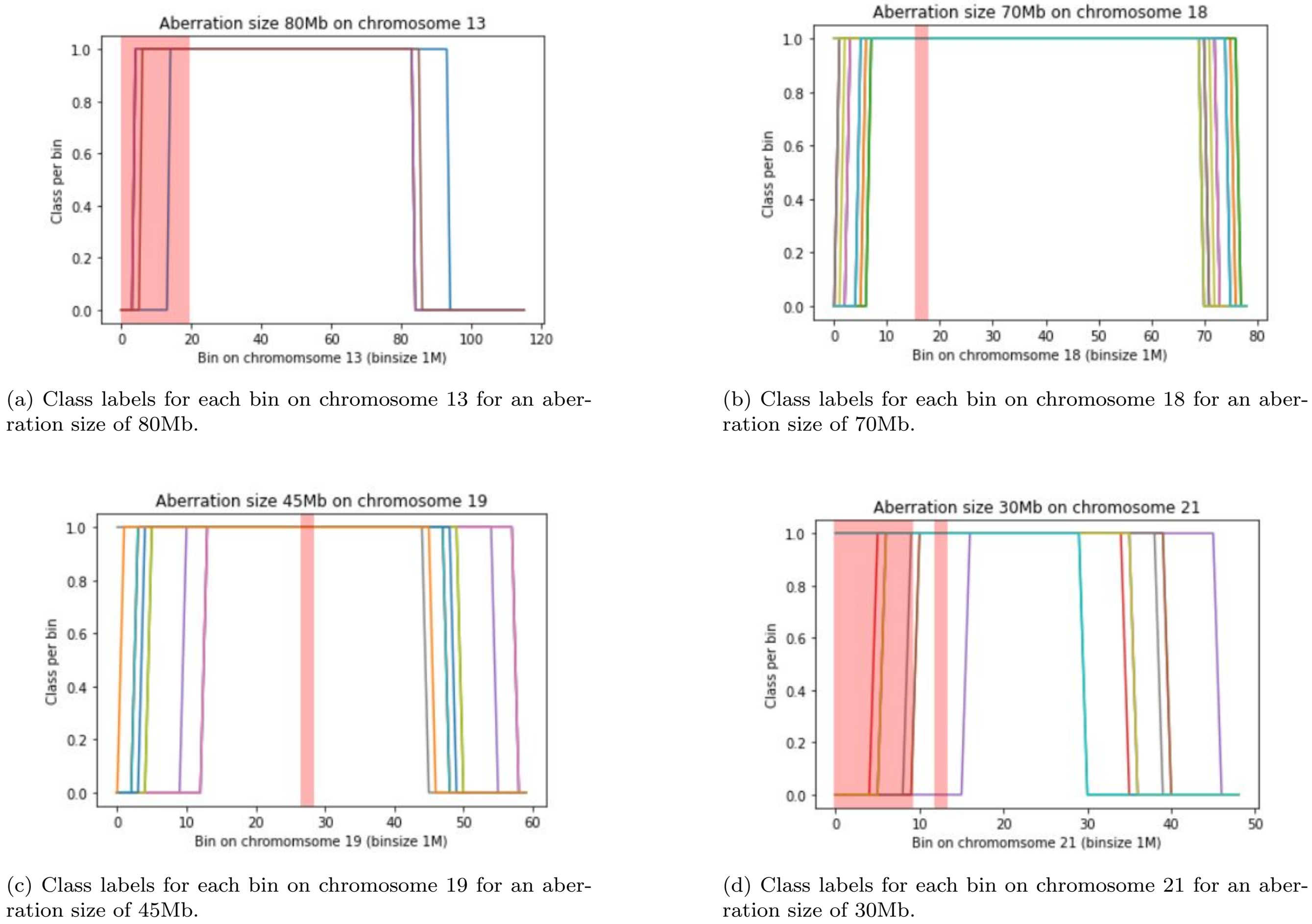
One of the reasons that both methods do worse on chromosome 21 than on other chromosomes might be that chromosome 21 is one of the smallest chromosomes and has the smallest amount of reads mapped to it. Each sample in this dataset has 20M reads, but these reads are mapped proportionately to the chromosomes' size. With chromosome 13 being over twice as big as chromosome 21, twice as many reads will be mapped there. Having more data to learn from, detecting aberrations on the larger chromosomes will be more comfortable.

Another reason could be the unmappable regions. As can be seen in table 3 almost 27% of chromosome 21 is unmappable. Using a bin size of 1Mb, the unmappable regions could span up to 12 of the 49 bins.

	Unmappable	Chromosome Length	U\L
Chr13	19,55M	115,17M	0,170
Chr18	3,38M	78,077M	0,0439
Chr19	3,28M	59,129M	0,0554
Chr21	12,94M	48,130M	0,269

Table 3: Size of the unmappable regions, the length of the chromosome and the ratio of unmappable regions versus length of the chromosome for chromosomes 13, 18, 19, and 21

In Figure 12, the unmappable regions are depicted in red for each of the chromosomes 13, 18, 19, and 21. Since the unmappable regions were not taken into account when simulating new data, the chance is high that a part of the aberration is located in the unmappable region. In Figure 12, it can be seen that for both chromosome 13 and 21, the unmappable region is located at the beginning of the chromosome, whereas it is located in the middle of the chromosome for chromosomes 18 and 19. Though the unmappable regions on chromosome 21 relatively span the largest area, the aberrations on chromosome 18 and 19 are split into two smaller aberrations



(a) Class labels for each bin on chromosome 13 for an aberration size of 80Mb.

(b) Class labels for each bin on chromosome 18 for an aberration size of 70Mb.

(c) Class labels for each bin on chromosome 19 for an aberration size of 45Mb.

(d) Class labels for each bin on chromosome 21 for an aberration size of 30Mb.

Figure 12: The class labels per bin for an aberration size of 80, 70, 45, and 30Mb for chromosome 13, 18, 19, and 21 respectively. Class 0 indicates a healthy bin and class 1 indicates an aberration bin. The unmappable regions are depicted in red.

by the unmappable region, making them harder to detect. Consequently, the unmappable regions do not seem to cause a drop in performance when comparing chromosome 21 to the other chromosomes.

Experiment 2 showed that both methods worked better when detecting larger aberrations. The LSTM model has an MCC value of over 80% for aberrations of size 35Mb or higher, with a few exceptions. For trisomy detection, the LSTM has an MCC score of 93%. WISECONDOR scores lower on the larger aberration sizes, with its MCC values varying wildly between aberration sizes that are only 1Mb apart. WISECONDOR does better for the smaller aberrations though, reaching an MCC value above 0.2 at aberration size 8Mb, while the LSTM does not reach that value until 17Mb.

However, the LSTM model only outperforms WISECONDOR when for each aberration size a different model is used. If we look for one model that can accurately predict aberrations of all sizes, Figure 6d shows that WISECONDOR outperforms the best overall performing model on almost all aberrations sizes. Again we see that the performance of both methods varies wildly for aberration sizes only 1Mb apart. Especially for the larger aberrations sizes, the performance of both methods seems to follow the same trend. This could indicate that the data causes this.

From the experimental data analysis in the experiment plan (in the Supplementary), we know that the smallest aberration in the data set is 30.25Mb. Looking only at both methods' performance on aberrations of size 30Mb and larger, we obtain the best LSTM model depicted in Figure 13. Here the LSTM model performs better than WISECONDOR, especially on the larger aberrations. It still has the sudden drops for a few aberrations sizes, so this does seem to point at it being caused by the data.

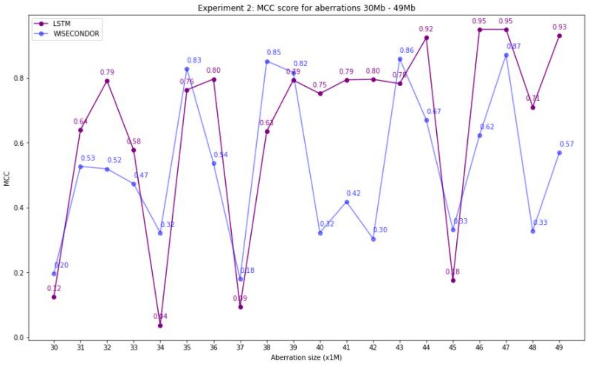


Figure 13: MCC score of the best performing LSTM model on aberration sizes 30Mb or larger depicted against WISECONDOR. Purple indicates the best LSTM model and blue shows WISECONDOR.

These drops could occur because part of the aberration is located in the unmappable regions, decreasing the aberration size. Suppose for a specific aberration size, a large part of the aberration is located in the unmappable regions, and for an aberration size of 1Mb larger, the aberration is located mainly outside the unmappable regions. In that case, the aberration size of the former will be smaller. However, when we look at the aberrations' sizes, this does not seem to be the case. In Figure 14, the aberration size when the unmappable region is removed is plotted against the overall aberration size. The red vertical lines show aberration sizes where a sudden drop in performance was perceived. Though for some of the red lines, the average aberration size without the unmappable regions does show a slight drop, this does not seem to be why both methods perform this inconsistent for aberrations only 1Mb apart.

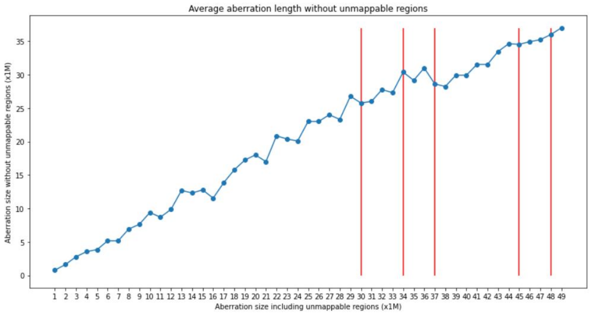
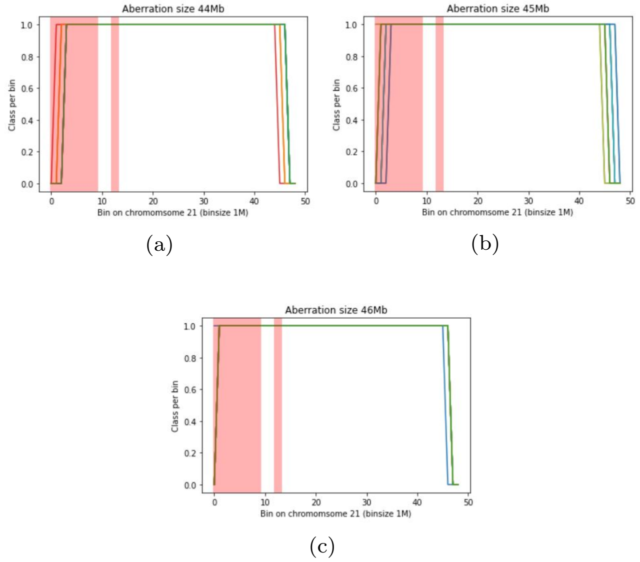


Figure 14: Plot of the average aberration size when the unmappable regions are removed from the length of the aberration. The blue line depicts the aberration length when the unmappable regions are removed (x1Mb) and the red vertical lines are five values where a sudden drop in performance occurred (30, 34, 37, 45, and 48Mb

Besides the size of the aberration, once the unmappable region is removed, it could be caused by the aberration's actual location on the chromosome in relation to the unmappable region. As mentioned before, if an aberration is split in two by an unmappable region, it will be harder to detect both aberrations. However, when looking at the aberrations' location on chromosome 21, there is no large difference between aberrations sizes only 1Mb apart. For example, a large sudden drop in performance occurs for aberrations of size 45Mb. In Figure 15, the class for each bin on chromosome 21 can be seen for the samples with an aberration size of 44Mb, 45Mb, and 46Mb. In the Figure, the class for each bin (0 for healthy and 1 for aberrated) is depicted on the y-axis, the bins on chromosome 21 on the x-axis and the red zones indicate the unmappable regions. All three plots do not deviate from each other largely.

Overall, when only focusing on detecting larger aberrations, the LSTM model outperforms WISECONDOR on most aberration sizes. For smaller aberrations, the LSTM is too unreliable. However, both methods show sudden drops in performance for aberrations only 1Mb apart for which no reason has been found.

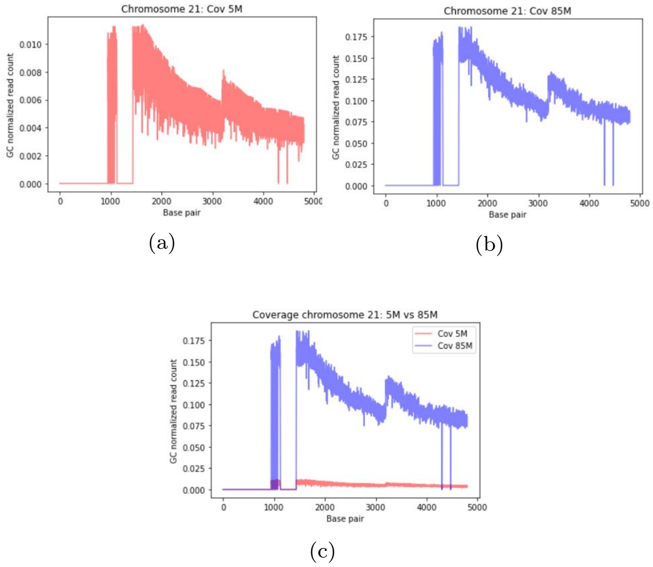
In experiment 3, we saw that both WISECONDOR and the LSTM did not perform well (0.566 and 0.496 at most for any coverage, respectively). The coverage has a considerable influence on the performance of a method. This may be because even though the read count's form remains similar, the read count itself might be as much as 10x as large. In Figure 16 the GC normalized read counts is shown for a sample with 5M reads (a) and a



(a) (b)

(c)

Figure 15: Plot of the class label for each bin on chromosome 21 for aberration sizes 44, 45, and 46Mb. Class 0 indicates a healthy bin and class 1 indicates an aberration bin. The unmappable regions are depicted in red.



(a) (b)

(c)

Figure 16: GC normalized read counts for a sample with (a) 5M reads, (b) 85M reads and (c) 5M versus 85M reads

sample with 85M reads (b). Both read counts have a similar form, but as can be seen in (c) the sample with 85M reads has a read count that is 10x higher.

This shows that samples whose coverages are far apart are quite diverse when compared to each other. Therefore, intuitively, a model that does well for coverage x will do better for coverages close to x than it does for coverages more deviating from x. To test this, a plot can be seen in Figure 17 where the MCC values of the best model for each coverage are shown.

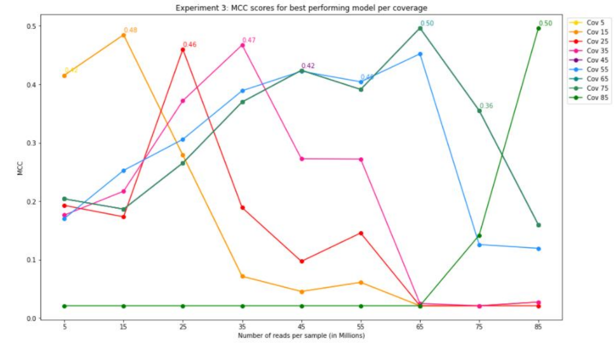


Figure 17: MCC value of the best performing model for each coverage. The highest value for each coverage is shown.

For each coverage, the highest value is shown in the colour of the model it belongs to. For each of these models, we see that the more the coverage deviates from the best performing coverage, the lower the performance becomes. For WISECONDOR, we saw the same. In Figure 7, we saw that WISECONDOR performed best at a coverage of 25M reads per sample, which is the coverage from which the reference set is made. The performance of WISECONDOR decreases as the coverage deviates more from 25M reads per sample.

It became apparent that the LSTM model is not capable of generalizing. If the coverages in the training data deviate too much, the model only learns to predict for a small range of coverages, performing worse for coverages outside this range.

We can deduce that the LSTM model should be trained on a set of samples within a specific range of coverages. That trained model can then be used to detect aberrations on samples within that range of coverages. If a lab would decide to sequence more or fewer samples than the range of samples the model is trained on, the model should be retrained on samples with comparable coverages. This means that the model should be re-trained if a new sequencing protocol becomes available, especially if new types of data are used. The LSTM does not seem to work adequately enough for a training set where the coverages are too widely spread.

Experiment 4 showed the influence of the fetal fraction on the performance of the methods. Compared to the coverage, the fetal fraction has a smaller influence on the read counts. In Figure 18, a plot of a sample with 1%, 10%, and 20% fetal fraction can be seen for a bin size of 1M. The difference in read count is relatively small. One could argue that this means that if samples with varying fetal fractions are tested, the performance will not deviate much between the samples.

In Figure 8 this seems to be the case. The performance of the LSTM on a set of varying fetal fraction remains between 0.51 and 0.66. Even for the best overall performing LSTM, the MCC value for each fetal fraction lies close together, except for a fetal fraction of 1%. For WISECONDOR, we see that it performs best for fetal fraction between 2 and 11 %. This is mainly because the reference set was created using samples with a fetal fraction of 10%. Figure 18 shows that the read count for a

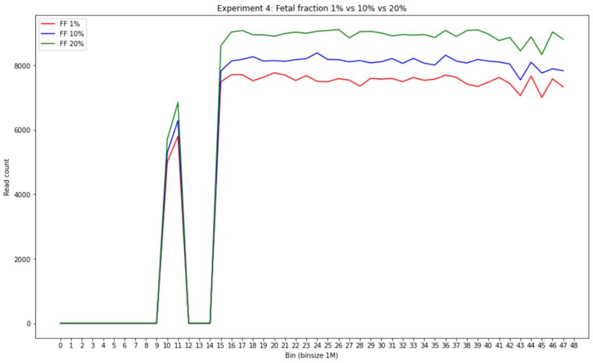


Figure 18: Read counts for a sample with 1% fetal fraction (red), 10% fetal fraction (blue) and 20% fetal fraction (green)

sample with a fetal fraction of 20% lies farther from the read count of a sample with 10% fetal fraction than a sample with 1% fetal fraction. Therefore it makes sense that WISECONDOR does better for the fetal fraction lower than 10% than for those higher than 10%.

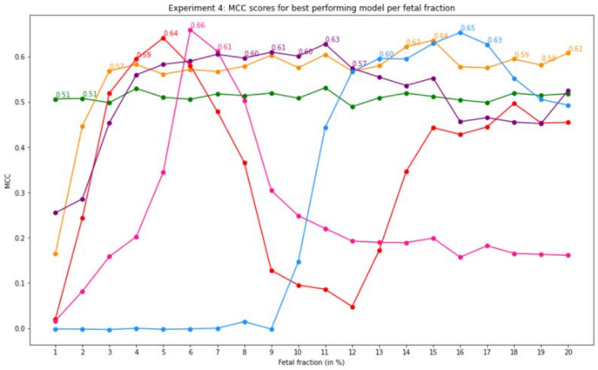


Figure 19: MCC values of the model that performed best per fetal fraction. The color of the MCC value indicates to which model it belongs.

For each of the fetal fractions, the model with the highest MCC score is plotted in Figure 19. In this Figure, the highest MCC value for each fetal fraction is shown in the colour of the model it belongs to. It can be seen that the highest MCC scores for each of the 20 fetal fractions belong to only six models. Noticeably, most of the highest values belong to the same model if the fetal fractions lie close to each other. This shows that if a model does well for a certain fetal fraction, it also does well for fetal fractions close to it. The experiment has shown that though the LSTM model's performance could be higher, the samples' fetal fraction does not significantly influence its performance as long as they do not deviate too much from each other.

In experiment 5, the influence of the bin size on the performance was measured. Decreasing the bin size had a positive effect on the performance on chromosomes 18 and 19. As the bin size decreased, the sensitivity, precision, and MCC value each increased. As mentioned before, the LSTM model uses bins to determine whether an aberration is present. This means that if an aberration spans only a part of the bin, the model will either label it as aberrated or healthy, which adds false positive or false negative base pairs to the performance, respectively. With a smaller bin size, the number of base pairs in the bin that are called wrong is smaller, leading to a higher precision, which results in a higher MCC value. For chromosome 13, the metrics do not deviate much between a bin size of 2Mb and 250kb. Its highest MCC value remains between 0.588 and 0.612. This is mostly because of the sensitivity remaining low for each bin size. The highest sensitivity it reaches is 0.555, detecting little over half the aberrations present at most. For chromosome 21, the MCC value increases only slightly. This is due to the precision increasing slightly, which, as explained before, could be because the aberrations do not adhere to bins, while the LSTM does. The sensitivity remains very low, detecting 9,3% of the aberrations present on average. Only one model detected over half the aberrations present with a sensitivity of 0.68, but this model has a precision of 0.079, meaning that it also found many false positives. We can see that the LSTM model performing worse for chromosome 21 is not due to the bins' size. Overall a smaller bin size does increase the performance of the model. However, even with a smaller bin size, the model still does not perform adequately for chromosomes 13 and 21.

In experiment 6, the difference in detecting duplications and deletions was tested. Both WISECONDOR and the LSTM model were better at detecting deletions. Intuitively it is easier to notice if (a part of) a chromosome is missing as opposed to noticing if the number of reads for (a part of) a chromosome is higher than usual. Since read counts in itself can vary depending on many factors such as the coverage or fetal fraction, detecting a rise in read counts might be more challenging.

Interestingly, the LSTM model did better at detecting duplications on chromosome 21 in this experiment than in experiment 1. Though the precision decreased, the sensitivity increased. This shows that the model detected more true aberrations, but also added more false positives. The model might have been able to find more aberrations because, in this experiment, the aberrations only occurred on chromosome 21, allowing the model to focus on this chromosome.

In Figure 20, the MCC values for the best model for each variant are shown. The pink line and number show the model that performed best when detecting deletions and the green line and number show the same for duplications. From this plot, we can see that the model only learns to detect deletions **or** duplications. If the score for one variant is high, the score for the other is zero or lower.

This leads us to believe that if the LSTM model is trained solely on either duplication or deletions on chromosome 21, the performance will be higher than when combined. Since evaluation is fast, multiple models could be trained, each on a different variant, and the results of

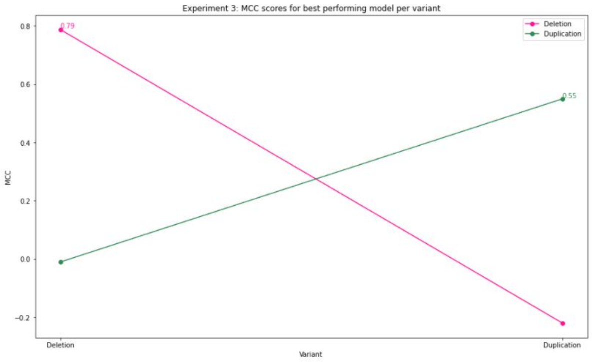


Figure 20: MCC values of the model that performed best per variant. Pink depicts the best model for detecting deletions and green shows the best model for detecting duplications.

each model on a new test sample could be combined to obtain the end detection for multiple variants.

5 Conclusion

Though the LSTM shows potential for detecting fetal chromosomal aberrations, the current model is too inconsistent. The models range from an MCC value of -0.374 to 0.95, while WISECONDOR is more robust in its detection.

One of the reasons the model is very inconsistent is its reliance on the initialization. If the initialization of the weights is not right, the model will do one of two things. It learns nothing and predicts the larger class for each bin, leading to an MCC value of 0. Alternatively, it learns wrong information, predicting random aberrations that are not true, leading to an MCC value below zero, indicating that its prediction is worse than a random prediction. One way to improve the initialization is to train the initial state as a parameter. This can be done by either not resetting the states after the training phase and using this as initial states for the test set, or by first detecting aberrations on a sample from the training set and using those states as initial states for the test set.

Another reason the model often does not perform well is its dependence on the dataset. As we saw by comparing experiments 1 and 6, the model performs differently when the dataset consists of other samples. From this, we can see that the LSTM has difficulty with generalization. When trained on one particular coverage, the model has difficulty detecting aberrations on a sample with much smaller or larger coverage. The same goes for samples containing different variants. When the model is trained on one variant, it detects barely any other variants present. An optimal training set for the current model would have the following characteristics. The sample should contain aberrations on all chromosomes with a size of at least half the chromosome size. The current model does not detect the aberrations of a smaller size consistently enough. The aberration on the chromosome should be of one variant. To improve this, perhaps two LSTM models can be combined, each specializing in a different variant, and combining both models' output to obtain the overall output. The sample's fetal fraction should be $\tilde{3}$% or higher, though a higher fetal fraction is preferred. The coverages of the sample should be no more than 10M reads apart. The same goes for the sample that will be tested. If the lab decides to sequence more or fewer samples than the range on which the model is trained, the model should be re-trained on comparable coverages. The model should be trained on a small bin size. This thesis's smallest bin size is 250kb, which performed the best of all tested bin sizes. In this thesis, no smaller bin sizes due to the time restraint when training the model, since a smaller bin size leads to a longer training time. A recommendation would be to test even smaller bin sizes when more time is available. The dataset might also be improved by taking the unmappable regions into account while simulating data.

Though the model often does not perform well, some trained models do detect aberrations quite accurately, and we saw in most experiments that although WISECONDOR has a higher sensitivity, the LSTM often has higher precision. So even though the LSTM finds fewer aberrations than WISECONDOR, the calls it does make are more precise.

In this thesis, we focused on creating an LSTM model to predict chromosomal aberrations. For this purpose, other many-to-many classification methods could be looked into as well. A Conditional Random Field (24) might be useful for this purpose. It can take a sequence as input, each with its label, and instead of predicting the label for each bin separately, it uses the dependencies between the predictions. It uses the input sequence in a single exponential model to determine the joint probability of the entire sequence of output labels. This model could be tested on its own or in combination with an LSTM model, as seen in (25).

Currently determining the fetal fraction of a sample is very hard, especially for female pregnancies. A method that can determine the fetal fraction for both male and female pregnancies is SeqFF(26), which uses both an elastic net (Enet)(27) and Second Weighted Rank Selection Criterion (WRSC)(28) to determine the fetal fraction from the read count on the gender chromosome. Ensuring that the fetal fraction is high enough is very important in NIPT, as mentioned before. Therefore, a method that could detect chromosomal abnormalities and determine the fetal fraction would be beneficial. Multi-task learning could be used to achieve this goal, where both tasks are solved simultaneously using the information from each other.

References

[1] Centers for Disease Control and Prevention. Facts about Down Syndrome — CDC, 2018.

[2] GHR. Trisomy 18 - Genetics Home Reference, 2016.

[3] GHR. Trisomy 13 - Genetics Home Reference, 2013.

[4] Laura M. Carlson and Neeta L. Vora. Prenatal Diagnosis: Screening and Diagnostic Tools, 6 2017.

[5] Zarko Alfirevic, Faris Mujezinovic, and Karin Sundberg. Amniocentesis and chorionic villus sampling for prenatal diagnosis. In *Cochrane Database of Systematic Reviews*, number 3, page CD003252. John Wiley & Sons, Ltd, 7 2003.

[6] Johanne M Hahnemann and Lars O Vejerslev. Accuracy of cytogenetic findings on chorionic villus sampling (CVS) - Diagnostic consequences of CVS mosaicism and non-mosaic discrepancy in centres contributing to eucromic 1986-1992. *Prenatal Diagnosis*, 17[9]:801–820, 9 1997.

[7] R. Akolekar, J. Beta, et al. Procedure-related risk of miscarriage following amniocentesis and chorionic villus sampling: A systematic review and meta-analysis. *Ultrasound in Obstetrics and Gynecology*, 45[1]:16–26, 1 2015.

[8] Amniocentesis Test: Risks, Benefits, Accuracy, and More.

[9] P Mandel and P Metais. Les acides nucléiques du plasma sanguin chez l'homme. *Comptes rendus des seances de la Societe de biologie et de ses filiales*, 142[3-4]:241–3, 2 1948.

[10] Y. M. Dennis Lo, Noemi Corbetta, et al. Presence of fetal DNA in maternal plasma and serum. *Lancet*, 350[9076]:485–487, 8 1997.

[11] Jacob A. Canick, Glenn E. Palomaki, et al. The impact of maternal plasma DNA fetal fraction on next generation sequencing tests for common fetal aneuploidies. *Prenatal Diagnosis*, 33[7]:667–674, 7 2013.

[12] Rossa W.K. Chiu, K. C.Allen Chan, et al. Noninvasive prenatal diagnosis of fetal chromosomal aneuploidy by massively parallel genomic sequencing of DNA in maternal plasma. *Proceedings of the National Academy of Sciences of the United States of America*, 105[51]:20458–20463, 12 2008.

[13] Roy Straver, Erik A Sistermans, et al. WISECONDOR: Detection of fetal aberrations from shallow sequencing maternal plasma based on a within-sample comparison scheme. *Nucleic Acids Research*, 42[5]:e31, 3 2014.

[14] Manuel Holtgrewe. Mason-A Read Simulator for Second Generation Sequencing Data FACHBEREICH MATHEMATIK UND INFORMATIK SERIE B • INFORMATIK. Technical report, 2010.

[15] Heng Li and Richard Durbin. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25[14]:1754–1760, 7 2009.

[16] Yen Chun Chen, Tsunglin Liu, et al. Effects of GC Bias in Next-Generation-Sequencing Data on De Novo Genome Assembly. *PLoS ONE*, 8[4], 4 2013.

[17] Yuval Benjamini and Terence P. Speed. Summarizing and correcting the GC content bias in high-throughput sequencing. *Nucleic Acids Research*, 40[10], 5 2012.

[18] Wentian Li and Jan Freudenberg. Mappability and read length. *Frontiers in Genetics*, 5[NOV]:1–1, 2014.

[19] Recurrent Neural Networks and LSTM explained — by purnasai gudikandula — Medium.

[20] Abien M Fred Agarap. Deep Learning using Rectified Linear Units (ReLU). Technical report.

[21] Diederik P Kingma and Jimmy Lei Ba. ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION. Technical report.

[22] Josephine S Akosa. Predictive Accuracy: A Misleading Performance Measure for Highly Imbalanced Data. Technical report.

[23] Davide Chicco and Giuseppe Jurman. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, 21[1]:6, 1 2020.

[24] John Lafferty, Andrew Mccallum, et al. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data Part of the Numerical Analysis and Scientific Computing Commons Recommended Citation "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data" Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. Technical report, 2001.

[25] Zhiheng Huang, Baidu Research, et al. Bidirectional LSTM-CRF Models for Sequence Tagging. Technical report.

[26] Sung K. Kim, Gregory Hannum, et al. Determination of fetal DNA fraction from the plasma of pregnant women using sequence read counts. *Prenatal Diagnosis*, 35[8]:810–815, 8 2015.

[27] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. Technical Report 2, 2005.

[28] Alan Julian Izenman. Reduced-rank regression for the multivariate linear model. *Journal of Multivariate Analysis*, 5[2]:248–264, 1975.

Supplemental Materials

The supplementary consists of the following files and figures:

Experiment plan

Noor van Ruyven

1 Data Simulation Experiment

Overall there are three reasons for simulating data for this project. First of all, to train a deep learning model, (a lot of) data is needed. If there is not enough data available, a deep learning model may not be able to learn anything and will therefore not perform well. Second, the ground truth of the data is not known. The data is labeled, but these labels come from WISECONDOR [1]. Even though WISECONDOR has high accuracy, it is not 100% certain that the labels they output are true. So by simulating data yourself, the ground truth will be known. Last, by simulating data, experiments can be set up to test the limits of the model. Experiments can be done to determine for which value of, for example, the coverage the model does not work adequately anymore.

The current dataset consists of 584 samples, of which only 183 are positive for chromosomal abnormality. These samples consist of reads, aligned to the human genome. For each of these samples, the mapped reads can be counted (per bin), which can then be used as input for a model.

For simulation, Mason [2] will be used, which takes two reference genomes as input and simulates a new sample from them. Here one reference will be hg19 for the maternal reference and the other will be hg19 where one contig has been replaced by an aberrated contig, to represent a fetal reference. Afterward these new samples will be aligned to a reference genome, after which they are ready to be pre-processed and then used as input for a model.

In this experiment plan, the available data will be analyzed and the experiments and their data will be defined.

2 Available data

Multiple parameters can vary during data simulation. To obtain a general feel of what the simulated data should look like, the currently available data is analyzed. This dataset consists of 401 negative samples and 183 positive samples. Their labels originate from calls made by WISECONDOR. These labels are assumed to be true.

The following parameters will be analyzed: the number of reads per sample, the number of aberrations per sample and their size, the location of the aberrations on the chromosome, and which variants are present.

2.1 Number of reads per sample

For both positive and negative samples the number of reads are counted and a boxplot is created as can be seen in figure 1. The average for both positive and negative samples lies around 20M reads per sample.

2.2 Number of aberrations per sample

For the positive samples, the number of aberrations within a single sample is counted. In Figure 2 it can be seen that samples with one aberration occur most often with a considerable amount. Of the 183 positive samples, 79% contains one aberration, 17,5% contains two aberrations, and 3,5% contained three or more.

2.3 Size of aberrations

As can be seen in figure 3, the size of the aberrations mostly varies between 40 and 60Mb. The smallest aberration in the dataset has a size of 30.25Mb and the largest a size of 180.75Mb.

When this is split up per chromosome (Figure 4) it can be seen that the aberration size differs for each

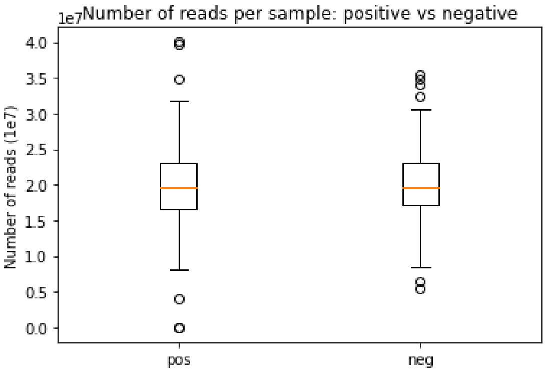


Figure 1: Number of reads per sample

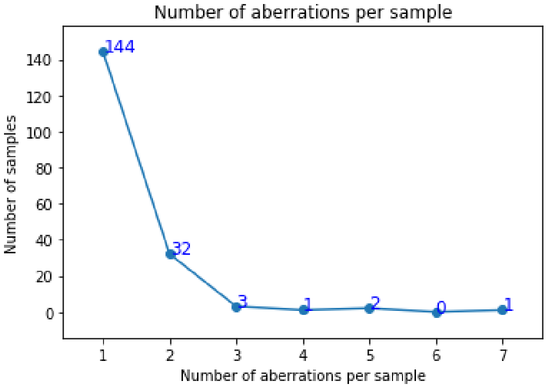


Figure 2: Number of aberrations per sample

chromosome. The figure also shows that for some chromosomes no sample with an aberration located on said chromosome exists in this dataset (4, 15, X, Y). For these chromosomes, no conclusion can be drawn solely from this dataset.

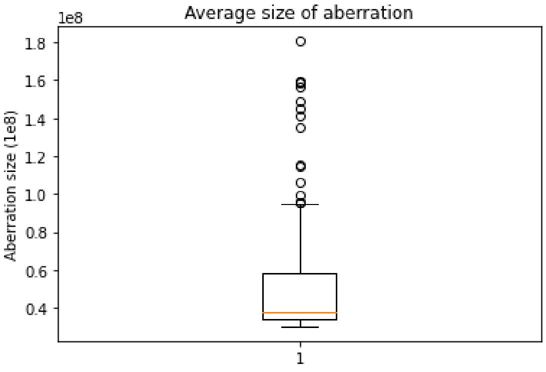


Figure 3: Boxplot of the average aberration size

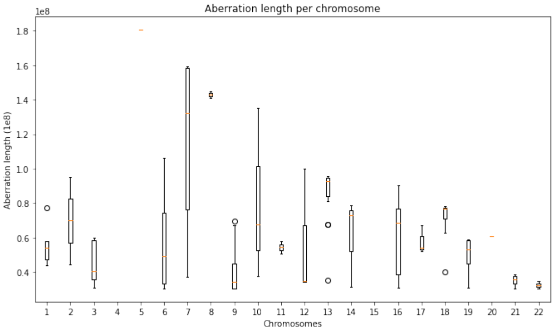


Figure 4: Boxplot of the average aberration size per chromosome

2.4 Location of aberrations

In figure 5 the distribution of aberrations over the chromosomes can be seen. As noted before, for some chromosomes the dataset does not contain a sample with an aberration located there. These chromosomes have been removed from the x-axis.

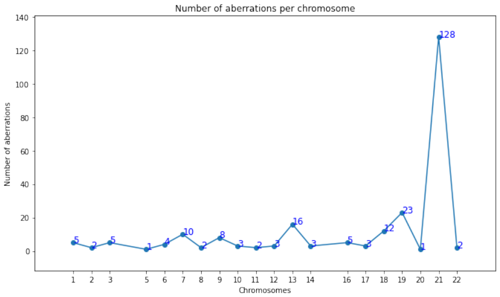
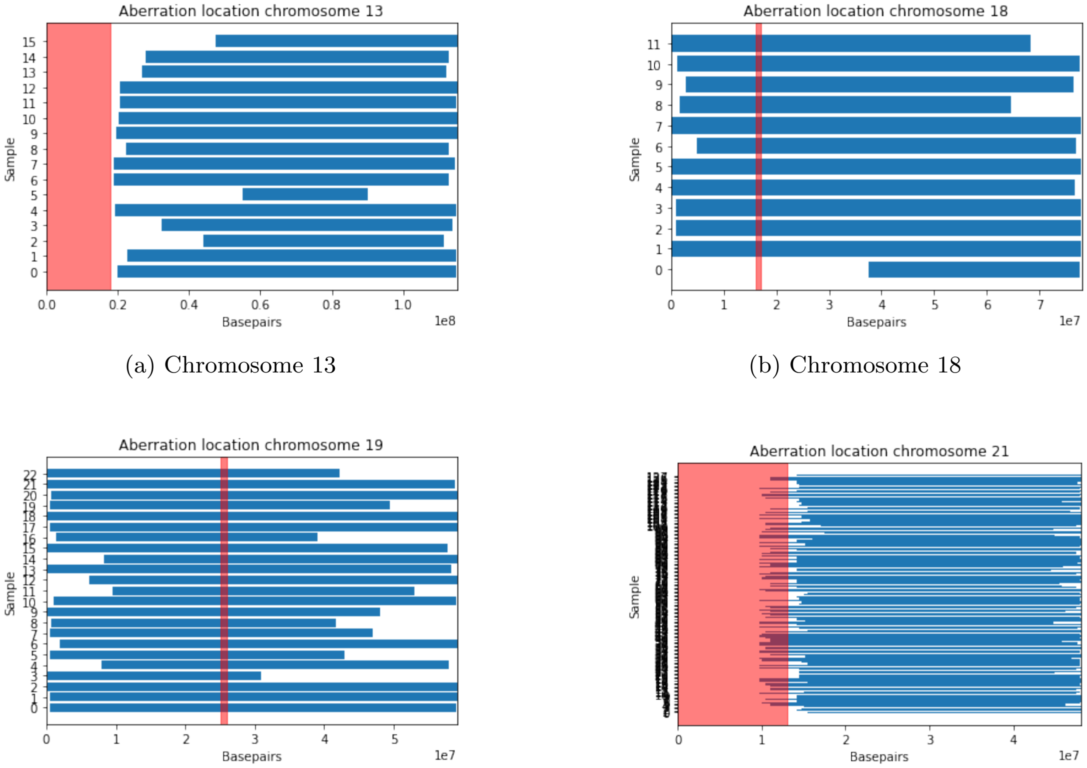


Figure 5: Number of aberrations per chromosome

Most aberrations occur on chromosome 21, followed by chromosomes 19, 13, and 18.

For each of these chromosomes, the location is analyzed, whether they occur only on a specific part of the chromosome or uniformly over all base pairs. For each chromosome, a plot is created showing the span of each aberration. The large unmappable regions (telomeres and centromeres) are marked in red. It can be seen that some of the aberrated regions include the unmappable region and some do not. This has to do with the parameters set by WISECONDOR.
In figure 6 the plots for chromosomes 13, 18, 19, and 21 (the chromosomes with the most samples in the dataset) have been added.



(a) Chromosome 13

(b) Chromosome 18

(c) Chromosome 19

(d) Chromosome 21

Figure 6: Aberration location for chromosomes 13, 18, 19 and 21

The figures show that the aberrations span most of the chromosome and do not adhere to a specific part.

2.5 Variant types

Of the 183 positive samples, not all present variants are known. It is known that 127 of the 183 samples (70%) are trisomies. Of the other 30% the variants are unknown.

3 Experiment 1: Basis

From the available data the following dataset characteristics for the basis experiment have been decided on: Each sample will consist of 20M reads, since this is the average amount of reads for both positive and negative data. Half of the samples will contain zero aberrations (healthy), the other half will contain a single aberration. This is divided evenly to maintain a balanced dataset. For this experiment samples with more than one aberration are not included, since this occurs far less often than one aberration.
Within this experiment, aberrations will only be simulated on four chromosomes: chromosomes 13, 18, 19, and 21. Trisomy 21, 18, and 13 are the most common forms of aneuploidy, therefore aberrations on these chromosomes occur most often. Chromosome 19 has been added to analyze if there is a difference in

performance between the most common aneuploidy chromosomes and one that is not commonly aberrated. Here chromosome 19 has been chosen because even though it is not one of the three most well-known chromosomes for trisomy, it still occurs relatively often (Figure 5).
For these four chromosomes the lower limit of the aberration size is chosen as the lower limit from Figure 7 and the full chromosome length is chosen as the upper limit for the aberration size:

- Chromosome 13: 80Mb - 116Mb
- Chromosome 18: 70Mb - 79Mb
- Chromosome 19: 45Mb - 60Mb
- Chromosome 21: 30Mb - 49Mb

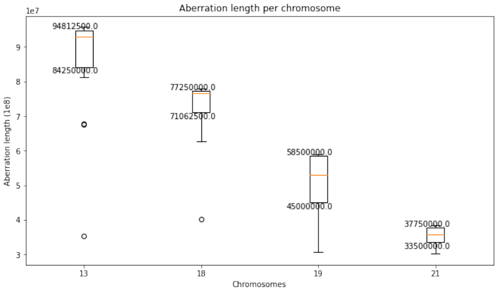


Figure 7: Aberration size for chromosomes 13, 18, 19 and 21

For simplicity, only one variant of aberration will be used in this experiment: duplications. Duplication has been chosen over deletion mainly because trisomy occurs more often than monosomy.
In real data, samples have an average fetal fraction of 10%. This value can vary quite a lot and depends on multiple factors, among which the presence of a trisomy. For this experiment, a fetal fraction equal to the normal average is used: 10%.
To decide on the amount of samples to simulate, the time and memory needed for each simulation is first evaluated.
Simulating and pre-processing the data consists of five steps. First, a variant has to be introduced into a healthy contig (*var*). Second, this contig has to be inserted into a healthy reference instead of the healthy contig (*rep*). Third, from the aberrated reference and a healthy (maternal) reference, a new sample has to be simulated (*sim*). Fourth, this new sample has to be aligned to the hg19 reference genome (*aln*). And last, the reads on the aligned file have to be counted per bin and pre-processed after which they can be used as input for the model (*prep*).
Table 1 shows average time and memory usage for each of the steps.

Table 1: Time (T) and Memory (M) usage

chr	*var*		*rep*		*sim*		*aln*		*prep*	
	T	M	T	M	T	M	T	M	T	M
13	10s	200Mb	80s	3.3Gb	1h	5.8Gb x2	2h	2Gb	1.5h	14.2Mb
18	10s	150Mb	80s	3.25Gb	1h	5.8Gb x2	1.5h	2Gb	1.5h	14.2Mb
19	7s	120Mb	80s	3.2Gb	1h	5.8Gb x2	1.5h	2Gb	1.5h	14.2Mb
21	6s	100Mb	80s	3.2Gb	1h	5.8Gb x2	1.5h	2Gb	1.5h	14.2Mb

From Table 1 it can be seen that steps *rep* and *var* take at most 10 seconds and 80 seconds respectively. *Sim*, *aln* and *prep* take 1, roughly 1.5 and 1.5 hours per sample respectively. So collectively, simulating a single sample takes 4 to 4.5 hours. However, steps *sim*, *aln* and *prep* can be done in parallel for multiple samples at the same time.

The memory needed for each sample is quite a lot. However, the output file from each step only has to be used as input for the next step, after which it can be removed. So only the output files from *prep* have to be kept, which is 4.84 Mb per sample for a bin size of 10kb. By only keeping the pre-processed bin counts for a small bin size a lot of memory can be saved and the read count per bin can easily be upscaled to a larger bin size.

2000 samples will be simulated, where half will be healthy and half will be aberrated. That means that for each of the four chromosome there will be 250 samples where the aberration is located on that chromosome.

4 Experiment 2: Aberration size

First, the influence of the aberration size on the performance of the model will be tested. For this experiment a data set will be simulated with the following characteristics:

The data set will consist of samples with 20M reads per sample and this dataset can consist of all aberrated samples, since the healthy samples from Experiment 1 can be re-used. The samples will contain one duplication, located on chromosome 21. The fetal fraction is equal to 10%.

The aberrations will range from the full length of the chromosome ($\sim$ 49Mb) to a size of 1Mb with steps of 1Mb. A bin size of 1M will be used. This will also test whether the model can detect aberrations of only 1 bin.

For this experiment 1000 samples will be simulated, so the data set will be balanced when combined with the 1000 healthy samples from Experiment 1. This means that there will be about 20 samples per aberration size between 49M and 1M base pairs.

5 Experiment 3: Coverage

Next, the influence of the coverage of the samples on the performance of the model will be tested. In the basis experiment the coverage is:

$$\begin{aligned}\text{Coverage} &= \text{(read count * read length) / total genome size}\\ &= (20.000.000 * 36)/3.000.000.000\\ &= 0.24\end{aligned}$$

Samples will be simulated for both a higher and lower coverage. The data set will consist of samples where half has no aberration and half has one aberration on chromosome 21. This aberration will span the entire length of the chromosome and the variant is a duplication (trisomy). A bin size of 1M is used and the fetal fraction is equal to 10%.

To get an idea of the amount of reads per sample needed for certain coverage values, the number of reads for a coverage of 0.05 and 1.0 are as follows:

$$\begin{aligned}\text{read count} &= \frac{\text{coverage} * 3.000.000.000}{36}\\ \text{read count}_{\text{min}} &= \frac{0.05 * 3.000.000.000}{36} \approx 4.166.666\\ \text{read count}_{\text{max}} &= \frac{1.0 * 3.000.000.000}{36} \approx 83.333.333\end{aligned}$$

Based on these, the number of reads per sample will range from 5M to 85M with steps of 10M. This results in nine different coverages to be tested.

For each coverage 60 samples with trisomy 21 and 60 samples that are healthy will be simulated. This roughly adds up to simulating 1000 samples.

6 Experiment 4: Fetal Fraction

On average a sample contains about 10% fetal fraction, which is also the fetal fraction chosen for the basis experiment. For this experiment the samples will consists of 20M reads per sample containing zero or one duplication of the full length of chromosome 21 (trisomy 21) divided into bins of size 1M bp. The fetal fraction will range from 1 to 20% with a step of 1%. For each fetal fraction 50 samples with trisomy and 50 healthy samples are created.

7 Experiment 5: Bin Size

For this experiment no new data has to be simulated, the data from Experiment 1 can be reused. After simulation and alignment, the number of reads are counted per bin of size 10.000 and saved. To test the influence of the bin size on the performance the saved read count for bin size 10.000 can be upscaled to the desired bin size. Here the label will be 1 (aberrated) if over half the bin is aberrated and 0 (healthy) otherwise. The bin sizes that will be tested are: $[250.000, 500.000, 750.000, 1.000.000, 1.500.000, 2.000.000]$. Here 250.000 is the smallest size because of the time restraint.

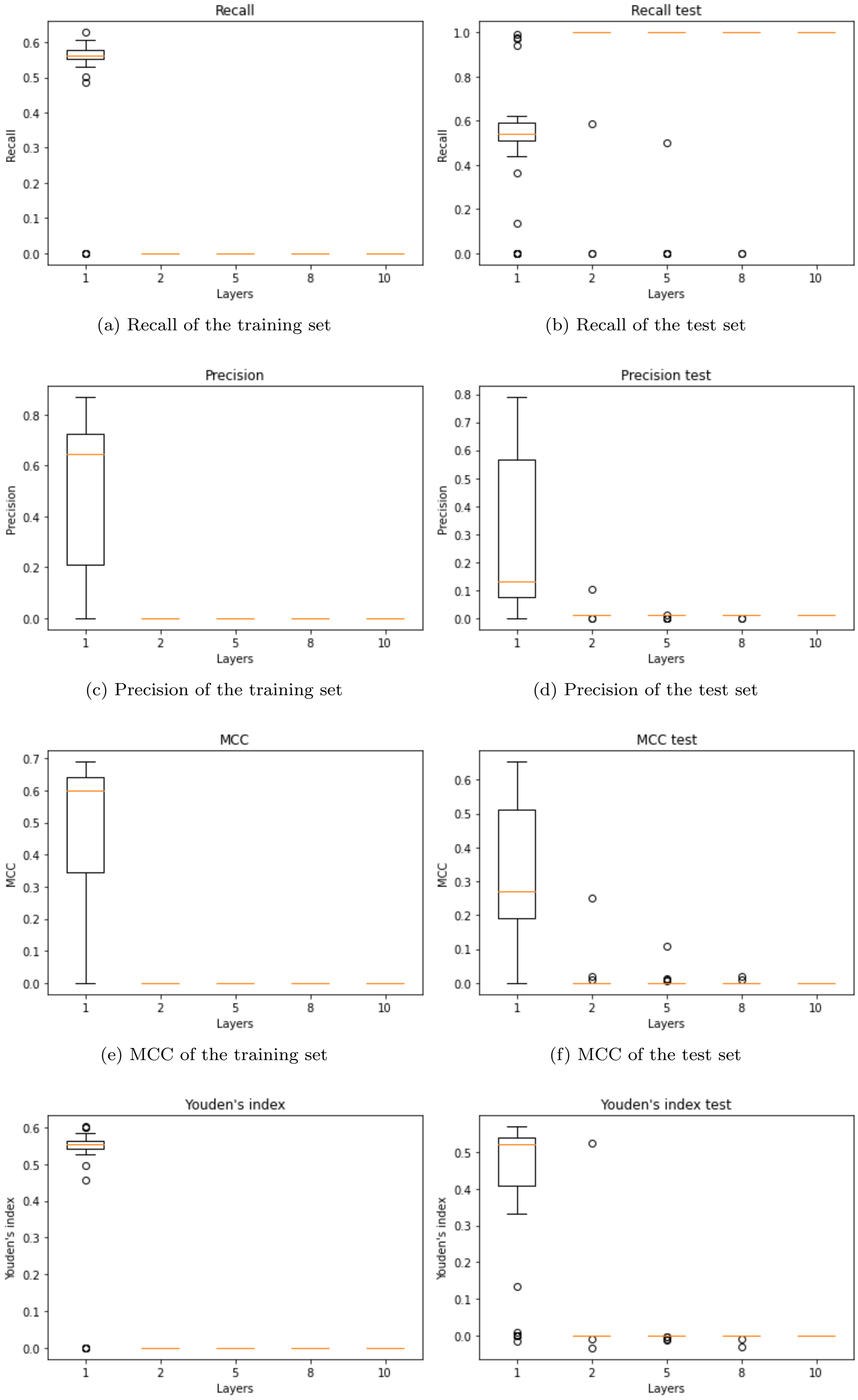
8 Experiment 6: Variant type

For the last experiment samples with deletions are compared to the samples with duplications and the healthy samples. For this experiment the samples will contain 20M reads per sample, containing zero or one aberration and a fetal fraction of 10%. The samples from Experiment 1 can be reused for the duplication and the healthy samples. The samples with a deletion will be simulated following the characteristics of Experiment 1, except that the aberration will only be located on chromosome 21. Here 250 samples will be created containing a deletion to add to the 250 sample containing a duplication on chromosome 21 in Experiment 1 and 250 healthy samples from Experiment 1, creating a data set of 750 samples.

References

[1] Straver R, Sistermans EA, Holstege H, Visser A, Oudejans CB, Reinders MJ. (2013) *WISECONDOR: detection of fetal aberrations from shallow sequencing maternal plasma based on a within-sample comparison scheme.* Nucleic Acids Res. 2014;42(5):e31. doi:10.1093/nar/gkt992

[2] Holtgrewe, M. (2010) *Mason – a read simulator for second generation sequencing data.* Technical Report TR-B-10-06, Institut für Mathematik und Informatik, Freie Universität Berlin.



(a) Recall of the training set

(b) Recall of the test set

(c) Precision of the training set

(d) Precision of the test set

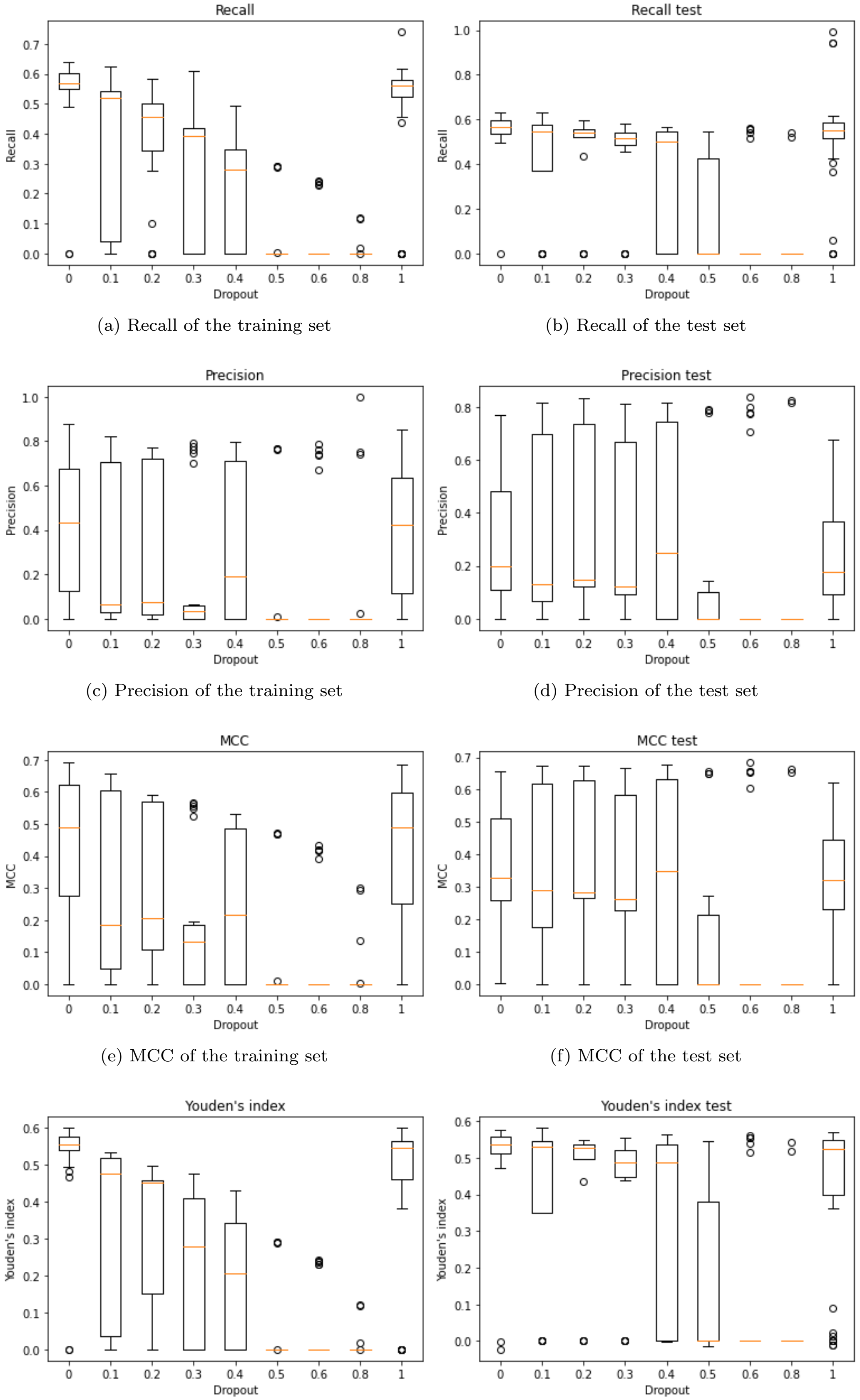
(e) MCC of the training set

(f) MCC of the test set

(g) Youden's index of the training set

(h) Youden's index of the test set

Figure S1: Results of parameter testing for the number of layers in the LSTM model. The left column shows the results of each metrics for the last epoch during training. The right column shows the metrics obtained by predicting for the test set after training.



(a) Recall of the training set

(b) Recall of the test set

(c) Precision of the training set

(d) Precision of the test set

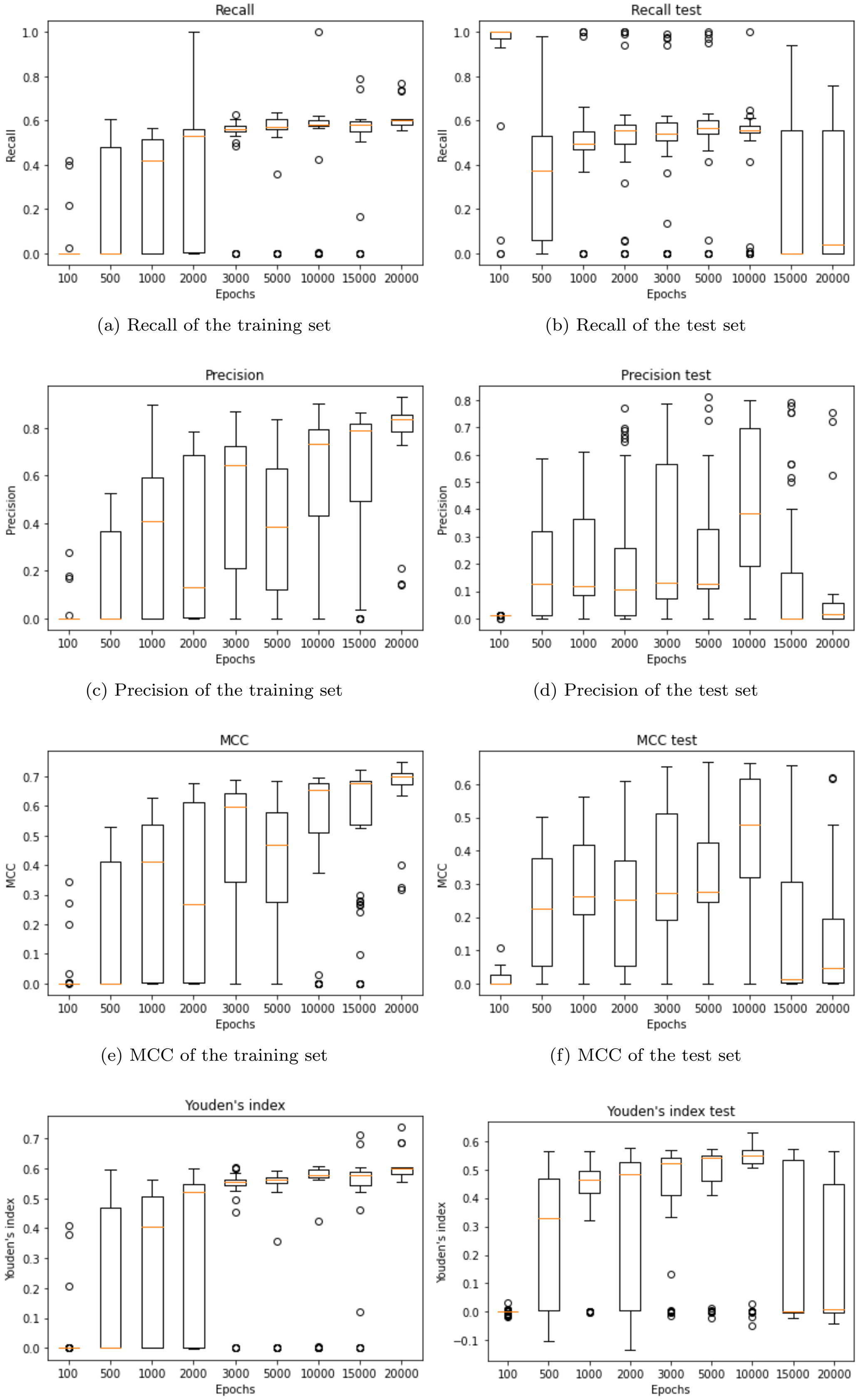
(e) MCC of the training set

(f) MCC of the test set

(g) Youden's index of the training set

(h) Youden's index of the test set

Figure S2: Results of parameter testing for dropout rate of the LSTM model. The left column shows the results of each metrics for the last epoch during training. The right column shows the metrics obtained by predicting for the test set after training.



(a) Recall of the training set

(b) Recall of the test set

(c) Precision of the training set

(d) Precision of the test set

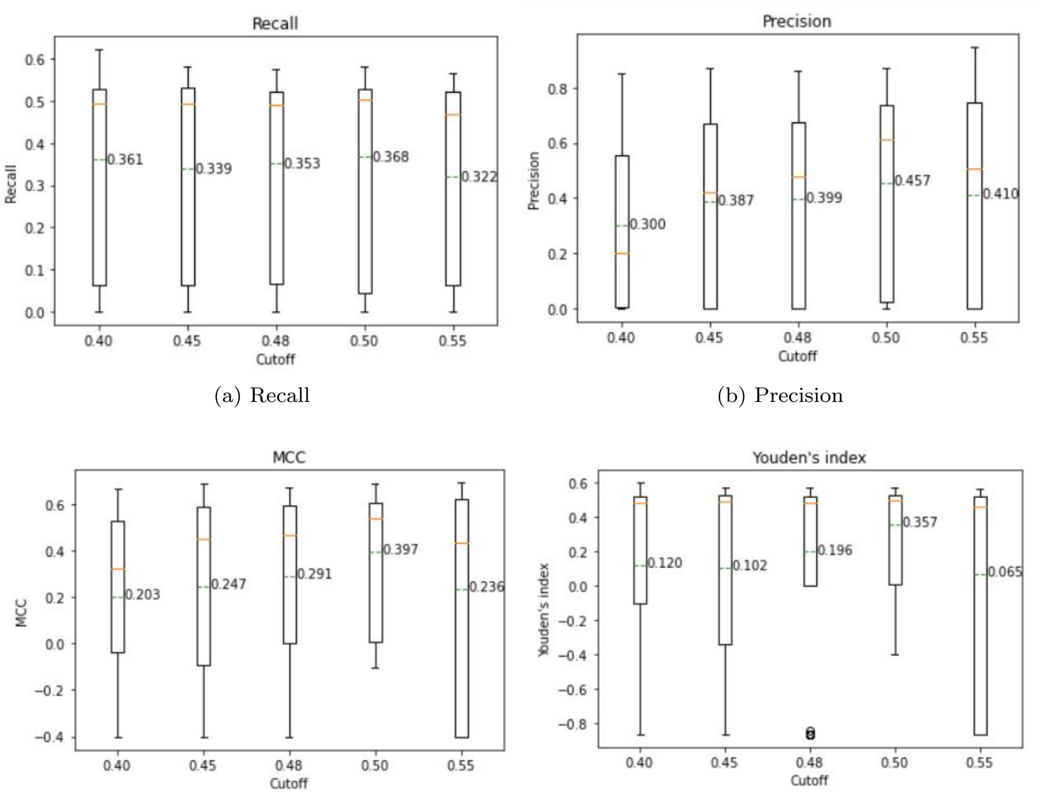
(e) MCC of the training set

(f) MCC of the test set

(g) Youden's index of the training set

(h) Youden's index of the test set

Figure S3: Results of parameter testing for the number of epochs during training. The left column shows the results of each metrics for the last epoch during training. The right column shows the metrics obtained by predicting for the test set after training.



(a) Recall

(b) Precision

(c) MCC

(d) Youden's index

Figure S4: Results of parameter testing for the cutoff between the healthy and aberrated class label.

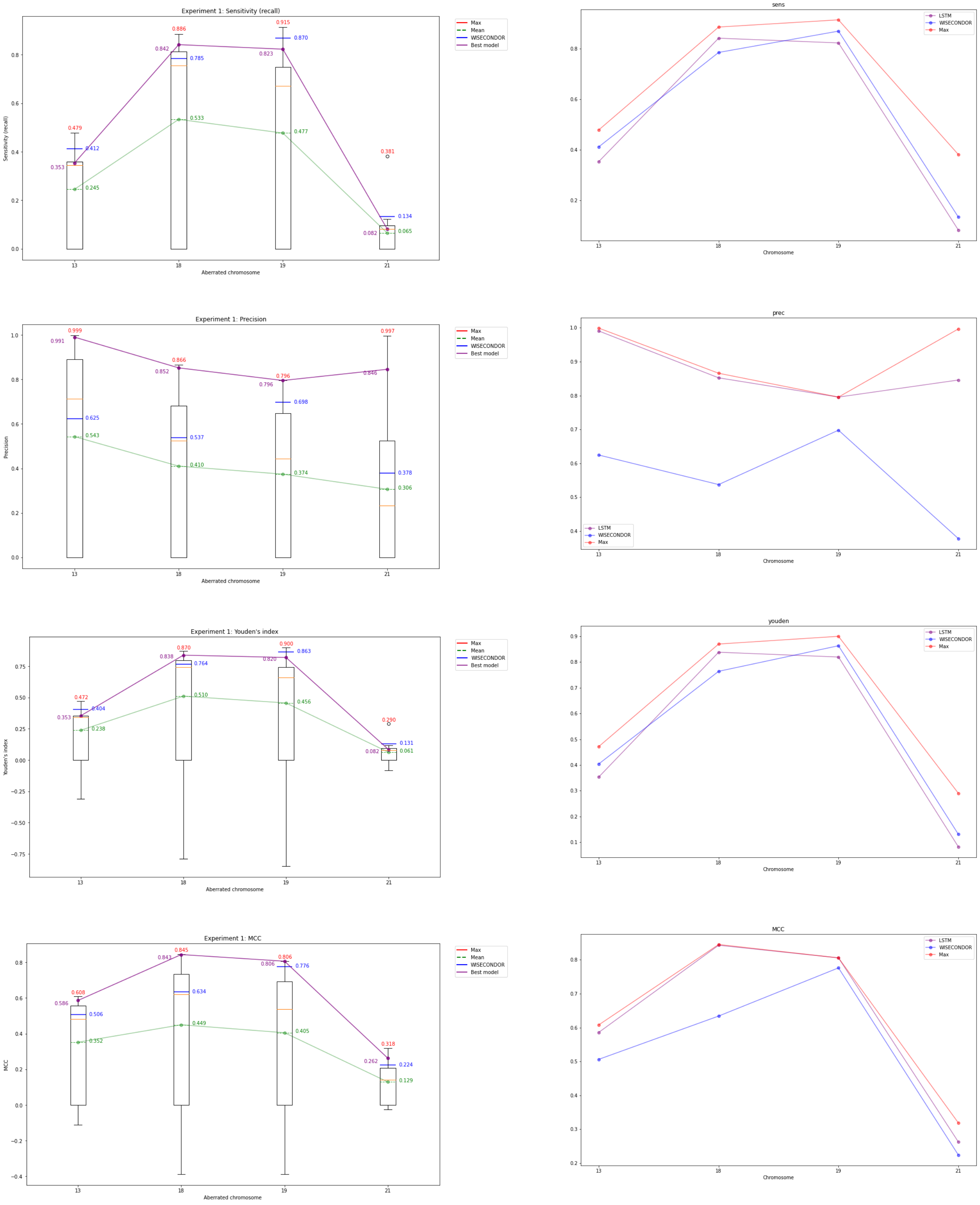


Figure S5: Results Experiment 1. In the left column the boxplot for each metric is depicted per chromosome. In the right column only the highest values are shown in a line plot. The red line and numbers indicate the individual highest value for the metric, the green line and numbers indicate the mean value, blue indicates WISECONDOR, and purple indicates the overall best performing model.

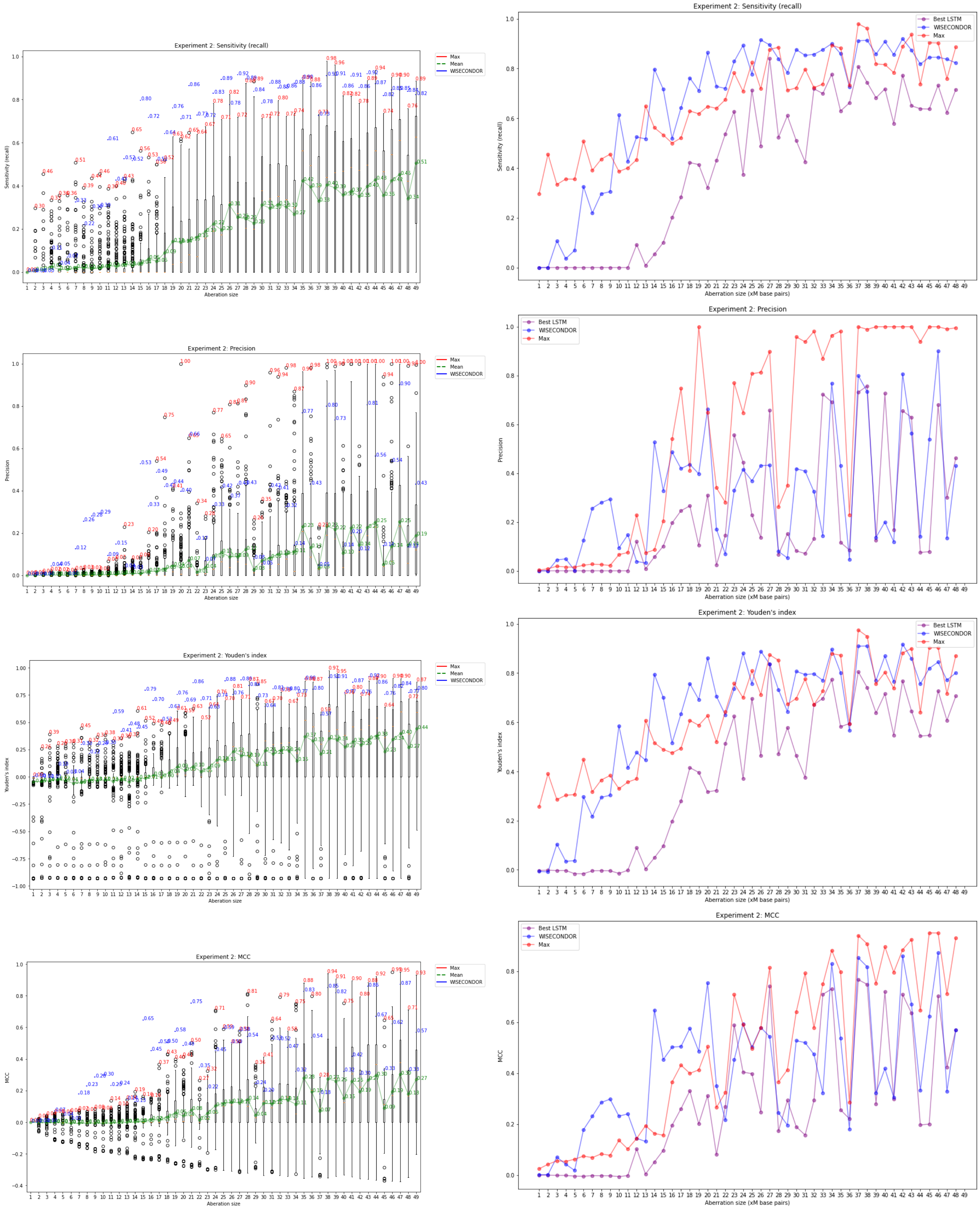


Figure S6: Results Experiment 2: Aberration size. In the left column the boxplot for each metric is depicted per aberration size. In the right column only the highest values are shown in a line plot. The red line and numbers indicate the individual highest value for the metric, the green line and numbers indicate the mean value, blue indicates WISECONDOR, and purple indicates the overall best performing model.

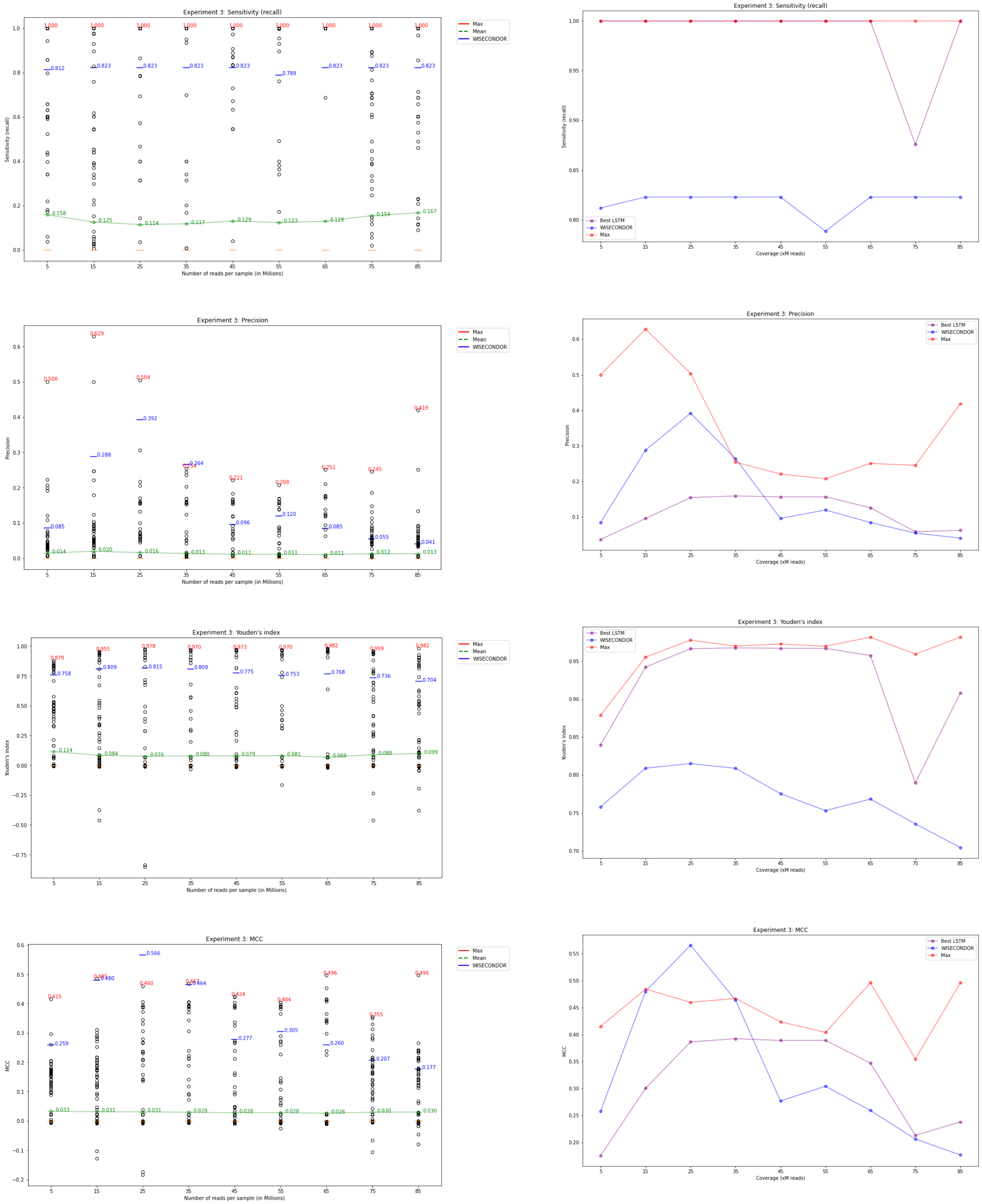


Figure S7: Results Experiment 3: Coverage. In the left column the boxplot for each metric is depicted per coverage. In the right column only the highest values are shown in a line plot. The red line and numbers indicate the individual highest value for the metric, the green line and numbers indicate the mean value, blue indicates WISECONDOR, and purple indicates the overall best performing model.

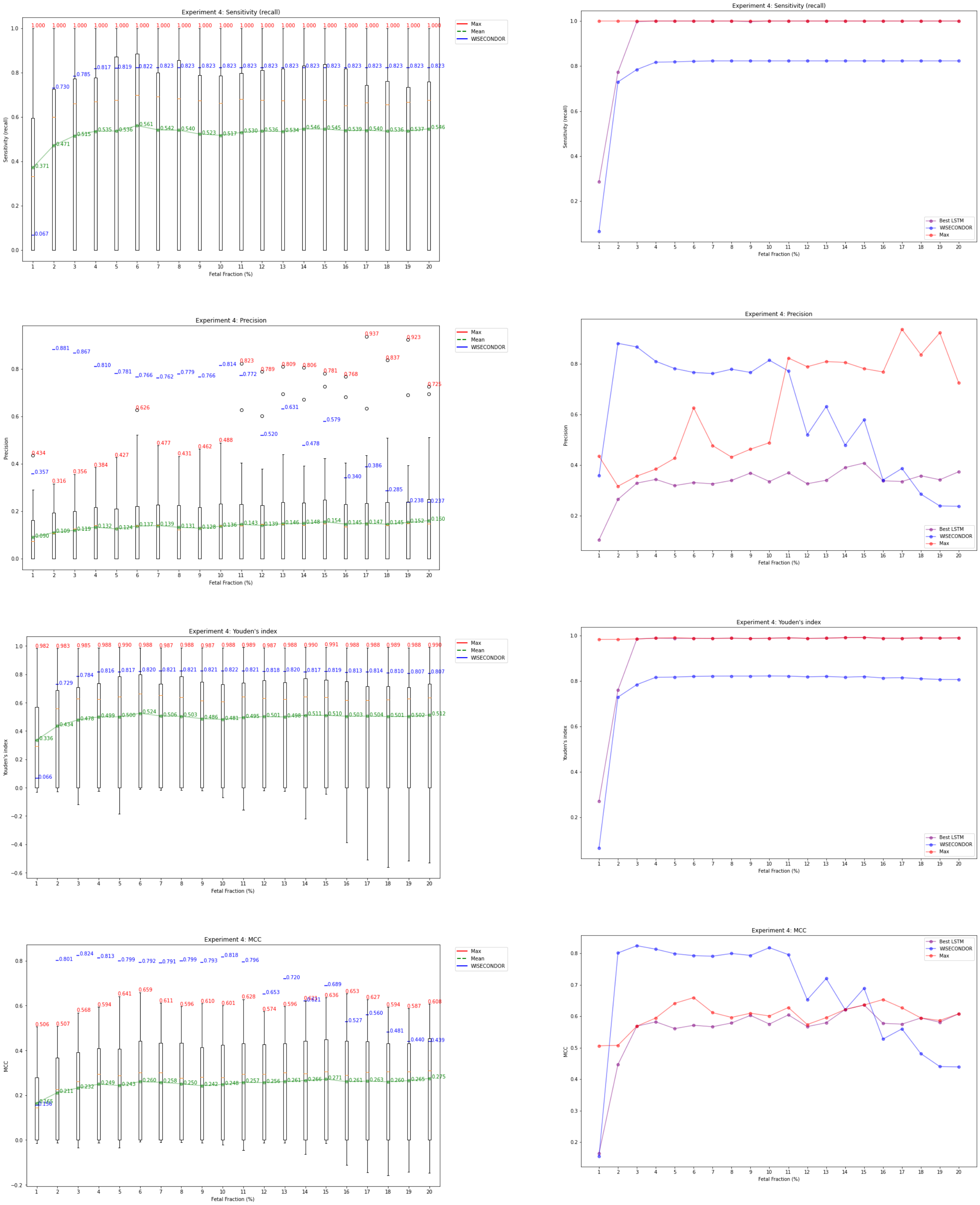


Figure S8: Results Experiment 4: Fetal Fraction. In the left column the boxplot for each metric is depicted per fetal fraction. In the right column only the highest values are shown in a line plot. The red line and numbers indicate the individual highest value for the metric, the green line and numbers indicate the mean value, blue indicates WISECONDOR, and purple indicates the overall best performing model.

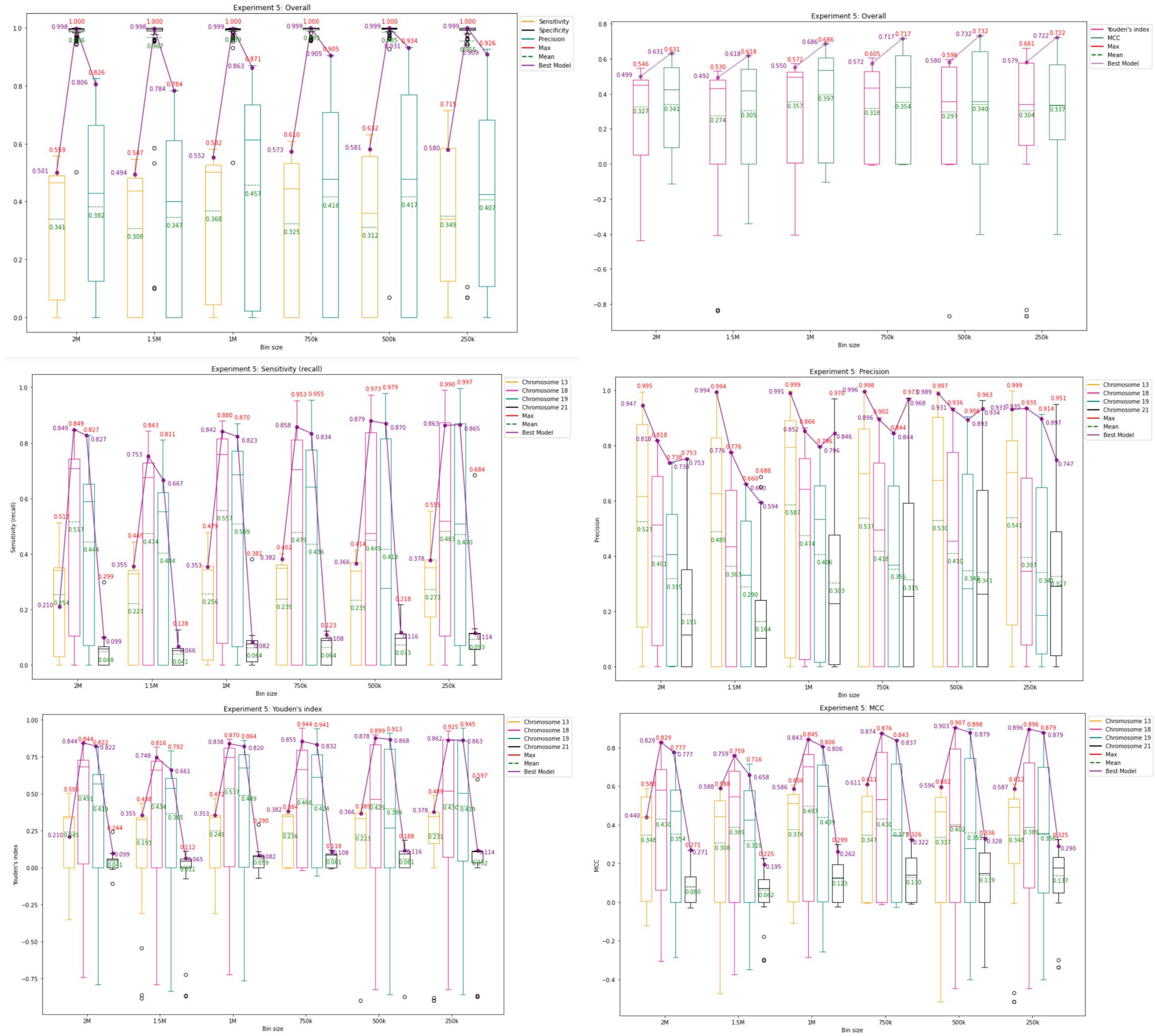


Figure S9: Results Experiment 5: Bin size. The first row shows the performance over all chromosomes. The left figure shows the sensitivity in orange, the specificity in black, and the precision in cyan. The right figure shows the Youden's index in pink and the MCC value in green. The second and third row show the sensitivity, precision, Youden's index, and MCC value split out per chromosome. The orange boxplot shows chromosome 13, the pink boxplot shows chromosome 18, the cyan boxplot shows chromosome 19, and the black boxplot shows chromosome 21. In all figures the red indicates the highest individual value per metric, green indicates the mean, and purple the overall best performing model.

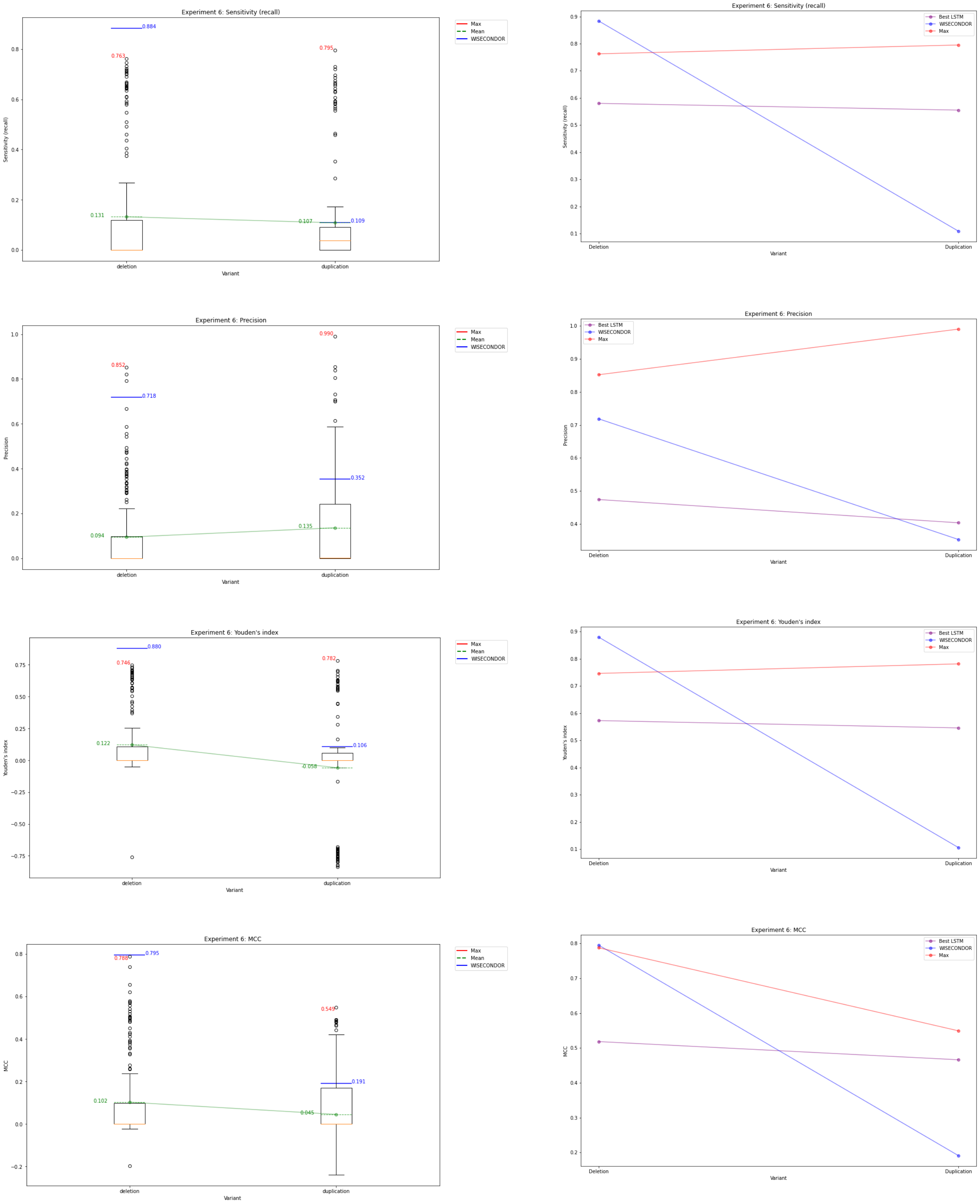


Figure S10: Results Experiment 6: Variant. In the left column the boxplot for each metric is depicted per variant. In the right column only the highest values are shown in a line plot. The red line and numbers indicate the individual highest value for the metric, the green line and numbers indicate the mean value, blue indicates WISECONDOR, and purple indicates the overall best performing model.