Spring 2023

Modeling Sequencing Artifacts in Artificial Low Frequency Cancer Data

Hannele Padre
San Jose State University

Modeling Sequencing Artifacts in Artificial Low Frequency Cancer Data

A Project

Presented to

The Faculty of the Department of Computer Science

San José State University

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

by

Hannele Padre

May 2023

The Designated Project Committee Approves the Project Titled

Modeling Sequencing Artifacts in Artificial Low Frequency Cancer Data

by

Hannele Padre

APPROVED FOR THE DEPARTMENT OF COMPUTER SCIENCE

SAN JOSÉ STATE UNIVERSITY

May 2023

Dr. Wendy Lee	Department of Computer Science
Dr. William Andreopoulos	Department of Computer Science
Dr. Cleber Ouverney	Department of Mathematics

ABSTRACT

Modeling Sequencing Artifacts in Artificial Low Frequency Cancer Data

by Hannele Padre

The rapid advancement in technology for next-generation sequencing (NGS) continues to make NGS more affordable, and in turn there is a high influx of sequencing data. While NGS is relatively fast and efficient to previous sequencing technologies, there are a multitude of steps in the NGS workflow in which sequencing errors can be introduced. Such sequencing errors are known as artifacts, and if not careful, can be mistaken for true variants. It's especially important to distinguish artifacts in cancer biopsies, more specifically, liquid biopsies, a noninvasive method for sample collection. Somatic mutations occur at low frequencies, and a liquid biopsies adds another challenge for detection if not enough cancer calls are collected in the sample. Thus, the distinction between low frequency mutations and low frequency artifacts becomes more difficult. In this study, machine learning methods will be used to model sequencing artifacts in NGS cancer data. The Genome in a bottle (GIAB) genomes and BAMSurgeon will be used as "truth-sets" to distinguish true variants from low frequency sequencing artifacts.

ACKNOWLEDGMENTS

I want to thank my family and friends for their never ending support in completing this project. And most of all I would like to thank Dr. Wendy Lee and the members of the NGS Sequencing Artifacts group: David Zhou, Felix Mbuga, Kathy Lam, and Luc Tang.

TABLE OF CONTENTS

CHAPTER

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

1.1 Sequencing and Genomics

Over the last few decades, there have been numerous efforts in the study of genomics. Previously, genetics was observed in the early 1900's through Mendelian inheritance, but it wasn't until 1977 that Sanger Sequencing was introduced which allowed the first ever sequencing of genes [2, 3]. Since then, sequencing technologies rapidly advanced into what we know today, Next-generation Sequencing (NGS), and continues to improve [3]. Through sequencing the field of genomics was born and now we are able to study the human genome and obtain better insight between genes and disease etiology. Diagnosis is now possible for previously undiagnosable genetic diseases by sequencing specific genes and identifying genetic variants [2]. Over the last few years, sequencing has become both more affordable and robust. While it was once only done in laboratories and research settings, it has now extended into clinical settings, revolutionizing healthcare [2]. One of the most impacted fields is oncology, the study of cancer, where thanks to NGS, exists panel testing for cancer screening and hereditary cancer risk assessment [2, 4].

1.2 Cancer Genomics

The invention of sequencing is the primary reason it is now possible to study cancer genomics. S. Behjati and P.S. Tarpey [3] described cancer as a "disease of the genome" due to the fact cancer is caused by harmful mutations in the DNA. By studying the genome and genetic variations, it became possible to not only detect cancer driver gene variants and rare cancer variants, but also to characterize cancer variants to an accurate diagnosis [2]. Currently, approximately 500 out of 20,000 genes have been discovered to be related to cancer [5]. Genomic sequencing also provided a new method of treatment for cancer, as well as other diseases, called precision

medicine. In precision medicine, the patient's genomic data and other factors such as environment and lifestyle are all taken into consideration to curate a personalized treatment [5]. Sequencing has also been adapted for clinical use in cancer panel screening, in which a set of genes that are known to be related to cancer are screened for cancerous mutations [4, 5]. Panel testing wouldn't have been possible without the sequencing technologies that have developed over the last few decades, most notably, next-generation sequencings(NGS).

1.3 Next-Generation Sequencing

Next-generation sequencing (NGS), also known as massively parallel or deep sequencing, is today's most popular and widely used method of DNA sequencing. There are many platforms of NGS but each platform uses this similar technique: the sequencing of amplified short DNA fragments in parallel [3]. The parallelization NGS process is as follows: First, short DNA template strands are bound in distinct positions on a plate. Next, the reads undergo bridge amplification. Then, each strand is extended with nucleotides that have been modified to fluoresce based on nucleotide type, and to terminate extension. Because of the amplification step, the fluorescence is strong enough for the microscope to detect, and this wavelength is recorded and converted into a base pair reading. The modified base pairs are then converted to normal base pairs to allow for the addition of another modified base pair. This extension process is repeated until the strands are fully sequenced [4].

The development of NGS technology revolutionized the study of genomics by producing staggering amounts of sequencing data which was not possible previously with Sanger sequencing. While Sanger sequencing produces a single DNA sequence in one experiment, NGS is able to produce millions of unique reads. This is significantly faster and less labor intensive than Sanger sequencing [4]. Additionally, NGS requires

relatively smaller samples, and still maintains high sensitivity and high coverage [2, 3]. The advancements of NGS sequencing technology have also made NGS more and more affordable, and continues to do so, such that NGS is now being used in clinical practice [5].

Some applications of NGS include: sequencing of the whole genome (noncoding and coding regions, WGS), whole exome sequencing (coding regions only, WES), or sequencing of individual genes or regions or interests [3]. Through sequencing, genes, diseases, and genetic variations can be discovered.

1.4 Variants

Variants in sequencing data, also known as mutations, are defined as genomics differences from the human reference genome [3, 4]. Variants can cause synonymous or nonsynonymous mutations depending on the severity of change in the DNA. There are 2 types of variants based on cell type: germline and somatic. Germline variants are passed down genetically and happen before conception, while somatic variants occur after conception, during one's lifetime. Somatic mutations can happen spontaneously, during mistakes in DNA repair, or when DNA is exposed to stress, and while some may produce no negative effects to the individual, cancers are typically the result of an accumulation of somatic mutations [6].

1.4.1 Single Nucleotide Polymorphisms

There are three types of variants classified by the type of alteration it makes to the DNA: Single nucleotide variants (SNVs), indels, and structural variants (SVs). Single nucleotide variants (SNVs) alternate the DNA by substituting one nucleotide with another [4]. This is similar to single nucleotide polymorphisms (SNPs), however for a variant to be considered an SNP, the mutation must be present in at least 1% of the population [7]. Indel variants are short insertions or deletions of a set of

bases. Large insertions or deletions, duplications, inversion, or translocation are all examples of structural variants that cause major structural changes to the DNA [4]. SNVs/SNPs are the most commonly occurring type of variant in the human genome. Those that occur in genes associated with DNA repair, the cell cycle, metabolism, and immunity are typically susceptible in cancer and can therefore potentially be used as cancer biomarkers [8].

1.5 Artifacts

Artifacts are defined as errors in sequencing, typically caused by DNA damage during sample preparation or mistakes during sequencing and informatics analysis. Artifacts can happen at any point in the NGS workflow from "tissue processing, tissue storage, DNA isolation, fragmentation, probe hybridization, library amplification, sequencing, and informatics analysis" [9, 10]. For example, tissue biopsies are typically stored in formalin-fixed, paraffin-embedded (FFPE) tissue due to its preservation properties [11]. However, FFPE causes a variety of DNA damage such as fragmentation, formaldehyde-induced cross linking, and histone-DNA crosslinks [5, 11]. Artifacts from sequencing misreads are caused by regions in the DNA that are difficult for the sequencer to read such as homopolymers [11] or GC-rich or AT-rich regions [12]. According to source [4], mistakes occur about 0.1% of the time, with the exception of rare sites that are prone to error with about a 1% error rate. Sequencing technologies can curb some artifacts by utilizing read depth, coverage, and cross referencing variants to the reference genome; however even through these filters, sequencing artifacts may still appear in the output read data [4], and is even more difficult to filter in cancer data.

1.6 Artifacts in Cancer Data

Distinguishing between artifacts and cancer data is still an ongoing challenge in research [9]. The typical workflow for NGS cancer specimens involves sequencing normal tissue and cancer tissue and cross referencing the artifacts and variants. This way, it is possible to distinguish between low frequency artifacts from low frequency cancer variants and from constitutional variants [9]. However, this method is costly and it is not always possible to obtain both datasets, therefore most laboratories run tumor-only sequencing [9]. Because tumor mutations have extremely low frequency due to tumor cellularity, it can be hard to distinguish between extremely low frequency artifacts and extremely low frequency mutations even with high-depth sequencing.[4] While a larger depth increases the confidence of low frequency variant calls, at a certain point, read depth results in an increase of cost more than performance. Typically, a read depth of 50x is reliable [4, 9]. Another possible solution includes cross-referencing variant callers, although this method does not remove biases from variant callers.

Liquid biopsies are another method of sampling for cancer in which samples are taken in a liquid form (typically blood but may also include saliva etc). This method of sampling is becoming more popular since it's less invasive than surgically removing cancer tissue. However, cancer DNA is diluted in this form since it's mixed with billions of other cells, such as blood cells, which lowers the allele frequency further. Therefore, it's critical to distinguish low frequency artifacts from the allele frequency cancer mutations in liquid biopsies. [13]

1.7 Genome In A Bottle

Genome in a bottle (GIAB) is a private-public and academic consortium, created by the National Institute of Standard and Technology, whose goals are to characterize benchmark human genomes. The NIST developed reference materials, such as

reference genome bed files and variant calling format (VCF) files from the human genomes characterized by GIAB. The first 'genome in a bottle' and most widely used characterized genome was NA12878/HG001, a caucasian Female from Utah, and since then, additional genomes have been characterized in the Personal Genome Project. The goals of GIAB are to provide benchmarking in the field of biotechnology and health sciences and develop reference standards, method, and data. In this project, the sample NA12878 will act as our 'ground-truth'. [14, 15]

1.8 NCBI: SRA

The National Center for Biotechnology Information (NCBI) is a government funded division of the National Library of Medicine (NLM), located within the National Institutes of Health (NIH). Within NCBI is a collection of databases and literature relating to biology, health, and medicine. The sequence read archive (SRA) is an NGS data repository, curated by a collection of organizations: International Nucleotide Sequence Database Collaboration (INSDC), European Bioinformatics Institute (EBI), and the DNA Database of Japan (DDBJ). NGS data is first submitted to one of the organizations before added into the SRA database. [16, 17]

1.9 Current Study

Distinguishing between low frequency artifacts and low frequency somatic mutations is an ongoing challenge in sequencing and genomic oncology. Current solutions are either costly or limited to the abilities of variant calling tools. This study will explore machine learning to characterize artifacts in cancer data. To obtain cancer data with a "truth set", the tool BAMSurgeon will spike in low frequency SNVs with allele frequencies ranging between 0.01-0.3%. Supervised learning models will be trained with metadata from a collection of samples from different cell lines and sequencing platforms. Various supervised models and features will be tested for best

accuracy.

CHAPTER 2

Methods and Materials

2.1 Data

The data is collected from the NCBI SRA database [18]. All samples are NGS data from paired-end, whole exome sequencing on Illumina platforms. Refer to Table 1 for a list of all collected sample.

Table 1: Data Collected from the NCBI SRA Database

Sample	Cell Line	SRR #	Ref Genome	Seq Scope	Library Prep Kit	Library Kit Assembly	Seq Platform	Seq Unit	Seq Layout
NA24631	HG005	SRR14724459	hg38	WXS	TruSeq	hg38	Illumina	NovaSeq 6000	Paired-end
NA24385	HG002	SRR14724462	hg38	WXS	TruSeq	hg38	Illumina	NovaSeq 6000	Paired-end
NA24631	HG005	SRR14724469	hg38	WXS	IDT	hg38	Illumina	NovaSeq 6000	Paired-end
NA12878	HG001	SRR14724463	hg38	WXS	TruSeq	hg38	Illumina	NovaSeq 6000	Paired-end
NA12878	HG001	SRR14724473	hg38	WXS	IDT	hg38	Illumina	NovaSeq 6000	Paired-end
NA12878	HG001	SRR14724503	hg38	WXS	IDT	hg38	Illumina	HiSeq 4000	Paired-end
NA24631	HG005	SRR14724479	hg38	WXS	Agilent SureSelect v7	hg38	Illumina	NovaSeq 6000	Paired-end
NA24631	HG005	SRR14724489	hg38	WXS	TruSeq	hg39	Illumina	HiSeq 4000	Paired-end
NA24631	HG005	SRR14724499	hg38	WXS	IDT	hg38	Illumina	HiSeq 4000	Paired-end
NA24631	HG005	SRR14724508	hg38	WXS	Agilent SureSelect v7	hg38	Illumina	HiSeq 4000	Paired-end
NA24385	HG002	SRR14724472	hg38	WXS	IDT	hg38	Illumina	NovaSeq 6000	Paired-end
NA24385	HG002	SRR14724482	hg38	WXS	Agilent SureSelect v7	hg38	Illumina	NovaSeq 6000	Paired-end
NA24385	HG002	SRR14724492	hg38	WXS	TruSeq	hg38	Illumina	HiSeq 4000	Paired-end
NA24385	HG002	SRR14724502	hg38	WXS	IDT	hg38	Illumina	HiSeq 4000	Paired-end
NA24385	HG002	SRR14724512	hg38	WXS	Agilent SureSelect v7	hg38	Illumina	HiSeq 4000	Paired-end

2.2 Bioinformatics Workflow

The Bioinformatics workflow was implemented in Snakemake, a workflow management tool [19], and follows the workflow in Figure 1. Breifly, NGS data are retrieved by the NCBI SRA database and are analyzed by the pipeline, resulting with variant call format (VCF) files, and subsequently filtered into artifacts and variants. Each step of the workflow is explained in depth.

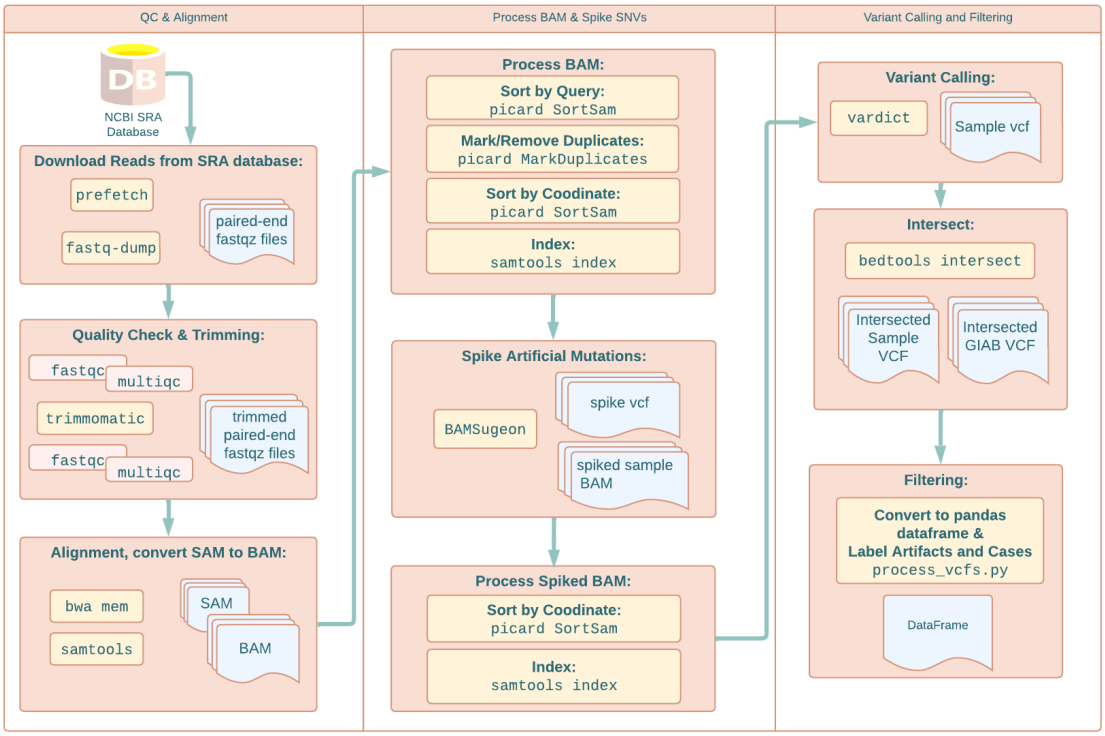

Figure 1: Bioinformatics Workflow to retrieve NGS data fom the NCBI SRA database for alignment, insert artificial somatic cancer mutations, variant calling, converting the resulting VCF file into a data frame, and filtering the variant calls into true variants and true artifacts.

This workflow consists of three main parts: QC and Alignment, Process BAM and Spike SNVs, and Variant Calling and Filtering. Under QC and Alignment, the reads are downloaded, checked for quality, trimmed, checked for quality post-trim, and aligned to the human reference genome. Process BAM and Spike SNVs is the part of the workflow in which artificial somatic mutations are added (or 'spiked') into the BAM file, and the resulting BAM file is prepped for variant calling. Lastly, Variant Calling and Filtering calls variants in the BAM files, resulting in VCF files, which are then converted into data frames and subsequently filtered into artifacts and variants.

2.2.1 Download SRA Reads: Prefetch and Fastq-dump

Prefetch and fastq dump are a part of the SRA toolkit, a collection of tools that is used to collect data from the SRA database. Prefetch retrieves the samples from the SRA database and fastq-dump downloads the FASTQ files. [20]

2.2.2 Quality Check: FastQC and MultiQC

FastQC is a tool to review the quality metrics of FASTQ files [21]. To view the metrics of multiple samples in one file, and MultiQC combines a collection of FastQC files into one readable file [22]. To assess the quality of our reads, one important metric is the phred score, which is a quality score that measures the base quality identified by the sequencer. [23]

2.2.3 Trimmomatic & QC

Trimmomatic was used to trim adapters and reads below a certain quality threshold [24]. The following trimming parameters are:

ILLUMINACLIP:{params.adapters} LEADING:30 TRAILING:30 MINLEN:70

Illumina Universal Adapters is in place of {params.adapters}.

2.2.4 Alignment: bwa mem

The next step is alignment, in which reads are aligned to the human reference genome. The tool used is the Burrows-Wheeler-Aligner (BWA), bwa mem. The same reference genome, hg38 (Homo sapiens (human) genome assembly GRCh38), was used for all samples. The output of bwa mem is a Sequence Alignment Map (SAM) file. I used samtools view to convert it into a Binary Alignment Map (BAM) file. [25, 26]

2.2.5 Processing BAM files

The BAM files produced by the alignment step are then processed. First, the file is sorted by query by picard SortSam [27]. This is done to ensure unmapped mates and secondary/supplementary alignments are marked removed by picard MarkDuplicates [28]. Marking and remove duplicates is done to remove PCR products that are from the same template to avoid having redundant data that might affect the allele frequency estimate of the variant calls. The BAM files are sorted by coordinate by picard SortSam and indexed by samtools to prepare for downstream analysis. [27, 26]

2.2.6 Spike in SNVs: BAMSurgeon

BAMSurgeon is a tool that adds synthetic mutations into alignment data to simulate somatic mutations. Since cancer data with a "truth set" is difficult to obtain, BAMSurgeon was used to simulate somatic cancer mutations and provide a "truth set". BAMSurgeon has the capabilities to add SNVs, Indels, and SVs, however for the purpose of this project, only SNVs are spiked in the data [1]. The method that BAMSurgeon mutates our BAM files is displayed in Figure 2. The BAMfile is represented in (a), and in (b) the areas of interest are highlighted. BAMSurgeon mutates the reads overlapping each position (c), and the reads are re-mapped to the reference genome (d) [1].

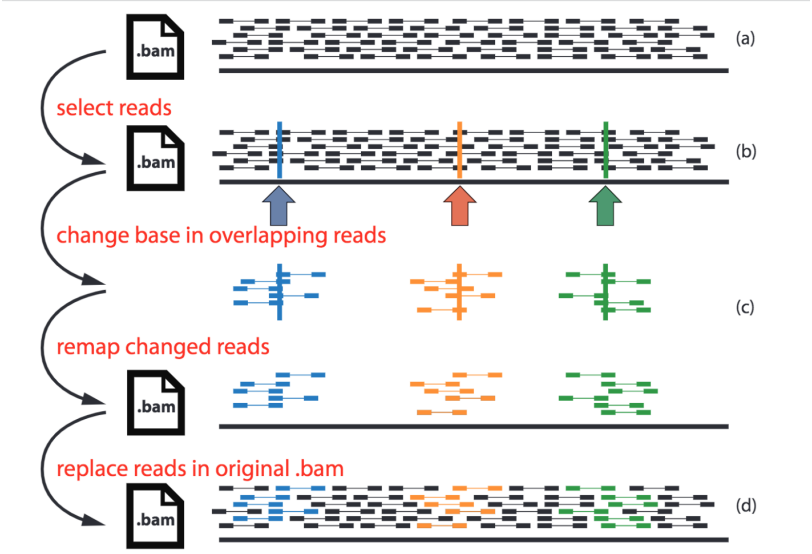

Figure 2: BAMSurgeon SNV Spike-in Method from [1]
This is the method in which BAMSurgeon spikes in SNV's into a BAMfile. (a) First, a region of interest is selected in the BAMfile. (b) BAMSurgeon selects a position to add the SNVs, here, this is indicated by the blue, orange, and green arrows. (c) The reads that overlap with this position are mutated and remapped. (d) The mutated and remapped reads are replaced into the original BAMfile.

BAMSurgeon's primary input is a BAM file, however there are optional parameters to specify the number of desired SNVs, the location, and the allele frequency [1]. The

desired number of SNVs for this project is 40,000 SNVs for each sample, with allele frequencies ranging between 0.01 to 0.3. This interval was chosen because it was observed in my data that the average AF of SNVs were approximately 0.2 and the goal is to simulate low frequency cancer variants. To give BAMSurgeon these specific directions, a VARFILE was created that gives BAMSurgeon regions and allele frequencies to add SNVs. To create the VARFILE for each sample, 40,000 rows were randomly selected from the corresponding exome bedfile and AF's chosen between 0.1 to 0.3 were selected for each row. Thus, the final VARFILE consisted of 3 columns: start position, end position, and allele frequency. It's important to note that bed files are 0-based, however BAMSurgeon takes in 1-based files for it's VARFILE parameter, therefore 1 was added to the start positions.

2.2.7 Variant Calling: VarDict

Variants were called using VarDict, a somatic variant caller [29], with a frequency of 0.01 to capture both variants and artifacts for analysis. The bed file from GIAB, corresponding to the sample cell line, is given during this step to narrow down our variant calling to these regions. The output of VarDict is a variant call file (VCF) that contains meta data information that will be used as features for machine learning analysis. The vardict variable descriptions are in the Appendix in Table .14.

2.2.8 Intersection

The intersection step is important to narrow down our dataset to the high confidence regions provided by GIAB, and the regions targeted by the exome sequencing kit. Without this step, it would increase the number of artifacts and skew our data. The variant calling tool VarDict already intersects our file with the sample's corresponding reference genome bed file. Bedtools intersect was used to intersect the sample VCF file with its corresponding sequencing exome kit bed file [30]. The end result is a

VCF file that only contains data with regions present in the GIAB and the exome kit. The GIAB vcf is also intersected with the exome sequencing kit bedfile. This is done such that comparisons between the sample and GIAB are limited to shared regions and true comparisons. A visual of how intersection narrows down the VCF files are presented in Figure 3 for the sample VCF and in Figure 4 for the GIAB VCF.

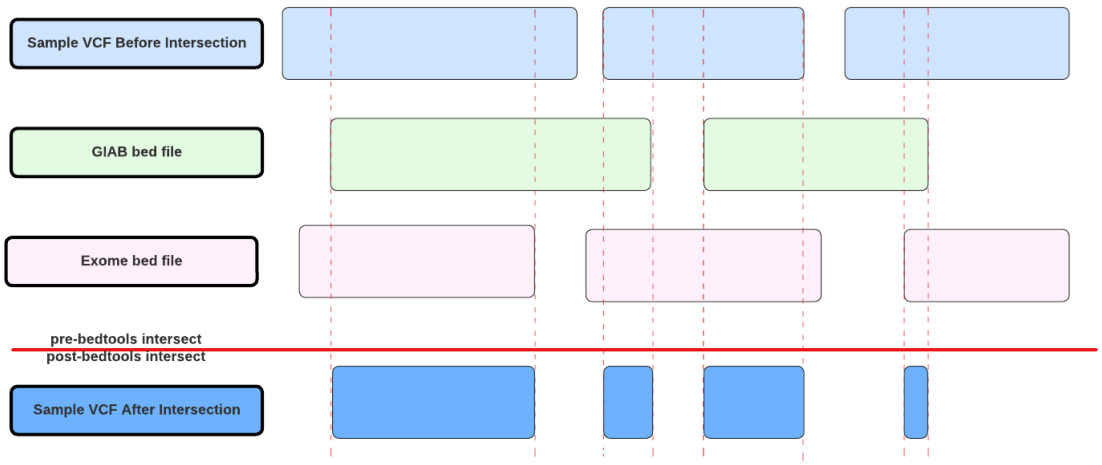

Figure 3: Sample VCF Intersection
The sample VCF (light blue) is intersected twice. First, the sample VCF (light blue) is intersected with the GIAB bed file (light green) to filter our sample down to regions that the GIAB vcf cover. Then, the sample VCF (light blue) is intersected with the exome bed file (red) to further filter our sample down to regions targeted by the exome sequencing kit. The final output is the sample VCF after intersections (dark blue).

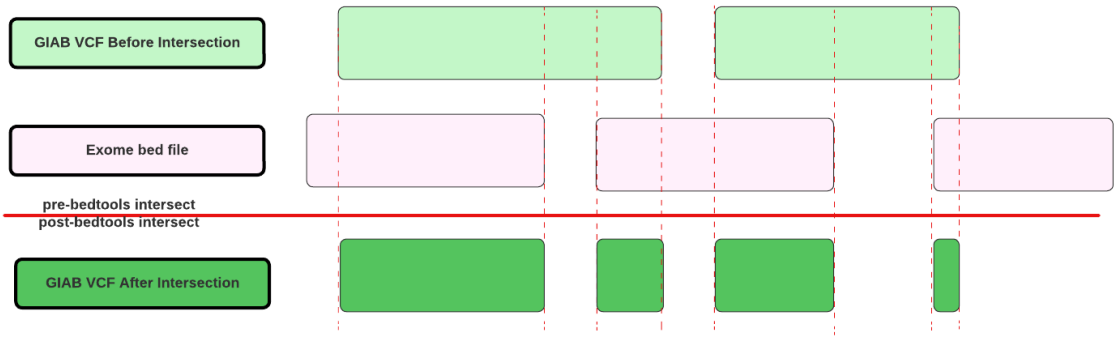

Figure 4: GIAB VCF Intersection
The GIAB vcf (light green) is intersected with the exome bed file (red) to filter the GIAB vcf down to regions covered by the exome bed file, provided by the exome sequencing kit.

2.3 Artifact and Variant Identification

Up to this point, three vcf files are produced, the intersected sample vcf and two ground truths: the vcf produced by BAMSurgeon and the intersected GIAB vcf. The last step in the Bioinformatis workflow is identifying the variant called as either artifacts and true variants. Variants and artifacts are defined in Table 2 and are split into 5 possible cases. The script process_vcfs.py converts the VCF files into pandas DataFrames follows the definitions of Table 2 to append 2 new columns 'ARTIFACT' and 'CASE'.

Case 3 occurs when no artifact/variant is identifited in a locus when GIAB called a variant. It is not tackled in this project since there is no available VCF data for this particular case. No reads- no vcf data. No call- no vcf data. The remainder of this project focuses solely on Case 1,2,4, and 5.

Table 2: Artifacts and Variant Identification

Case	**Description**	**ARTIFACT Assignment**
Case #1:	True Variant. The sample variant call matches the location and allele in GIAB.	False
Case #2:	Artifact. The sample variant call matches the location but differs in base call from GIAB.	True
Case #3:	Artifact. There is a variant call in GIAB but no variant call in the sample at the same location. There are 2 conditions in which case 3 can occur: 1. At the shared location, the sample's base call agrees with the reference genome, while GIAB does not, and call variant. 2. At the shared location, there are no reads in the sample.	True
Case #4:	Artifact. The sample calls a variant while GIAB does not.	True
Case #5:	BAMSurgeon Variant. BAMSurgeon spiked an SNV at this location.	False

2.4 Data Exploration

2.4.1 REF to ALT Heat Map

The variant of interest for this project are single nucleotide variants (SNVs), which are single base pair mutations/substitutions. There are two types of mutations, transitions (conversions between 2 purines (G,A) or conversions between 2 pyrimidines (C,T)), and transversions (conversions between purines(G,A) and pyrimidines(C,T)). Transition conversions occur most frequently according to D. M. Lyons and A. S. Lauring [31]. To visualize the transitions and transversions, a heat map was created from the data. To create the heat map, the dataframe is filtered to only contain the 'REF' and 'ALT' column and then pivoted. The heat map is then created using

seaborn [32]. The function is shown in Figure 5.

```
def ref_alt_heatmap(df):
    df_pivoted = (
        df[['REF','ALT']]
        .value_counts()
        .to_frame()
        .reset_index()
        .pivot('REF','ALT',0)
    )

    sns.heatmap(
        df_pivoted,
        annot=True,
        fmt='.2g',
        cmap='rocket_r'
    )
    plt.show()
```

Figure 5: Function to Create A Heat Map Given the VCF DataFrame.

2.5 Data Pre-processing

2.5.1 Feature Addition

The final output from the bioinformatic pipeline for this project are VCF files which are converted into python pandas DataFrames. The 'INFO' and 'FORMAT' columns are extracted from the VCF files and are used as additional columns in the DataFrame. Additional metadata obtained from the SRA webpage for each sample is added as additional features to the DataFrame: 'CELL_LINE', 'LIB_PREP_KIT', 'SEQ_PLATFORM', 'SEQ_UNIT'. Lastly, 'REF' and 'ALT' are combined into one column 'REF_TO_ALT'.

2.5.2 Feature Removal

The DataFrame is further prepared by removing features not needed for machine learning: 'DUPRATE', 'SVLEN', 'SVTYPE', 'LSEQ', 'RSEQ', 'REFBIAS', 'VARBIAS', 'RD', 'ALD', 'AD', 'SAMPLE', 'TYPE', 'CASE', 'CELL_LINE', 'SEQ_PLATFORM', 'ID', 'REF', 'ALT'. These columns are dropped for the following reasons: 1. Features that only had 1 value for all entries ('DUPRATE', 'TYPE', 'SEQ_PLATFORM'), 2. Features that had too many missing values ('SVLEN', 'SVTYPE'), 3. Categorical features that contained too many unique values ('LSEQ',

'RSEQ'), 4. Features that are redundant to other columns ('REFBIAS', 'VARBIAS', 'RD', 'ALD', 'AD','REF' and 'ALT'), 5. Features that identify the sample ('SAMPLE','CELL_LINE'), and 6. Features that directly identify the intended Artifact label ('CASE').

The DataFrame now contains the appropriate information for machine learning as displayed in Table 3. The descriptions for the features directly from the VCF meta data is described in Appendix in Table A.13.

Table 3: DataFrame Column Data Types for Machine Learning

Data Type	Column/Feature
Categorical	['REF_TO_ALT','LIB_PREP_KIT','CHROM','SEQ_UNIT', 'BIAS','GT','FILTER']
Numerical	['ADJAF', 'DP', 'AF', 'HIAF', 'HICNT', 'HICOV', 'MQ', 'MSI', 'MSILEN', 'NM', 'ODDRATIO', 'PMEAN', 'PSTD', 'QSTD', 'QUAL', 'SBF',' SHIFT3', 'SN', 'SPANPAIR','SPLITREAD', 'VD','POS']
Label	['ARTIFACT']

2.5.3 OneHotEncoder()

The categorical attributes of my data in Table 3 were one-hot encoded to create a binary representation in the form of a sparse matrix. Encoding categorical data is typically required to use for many scikit-learn estimators. [33]

2.5.4 StandardScalar()

The numerical attributes of my data in Table 3 were transformed using StandardScalar(), which changes the mean to 0 and scales the features to unit variance. This is done to normalize our data and decrease the distance between data points to increase the performance of the model. One caveat to using StandardScalar however is its sensitivity to outliers. [34]

2.6 Machine Learning Models

After pre-processing the data, the data is split into testing and training sets into 20% and 80%, respectively. The label column 'ARTIFACT' is separated from both sets. Since the data have categorical labels, the machine learning models to be used are supervised and classification models. To choose which classification models to train on, LazyPredict was used to quickly assess a total of 20 classification models. [35]

2.7 Assessing of ML Models Performance

Models were validated by 3-fold cross-validation (CV) and confusion matrices. K-fold cross validation produces a metric for the performance of the models by training the model in k-1 folds and performing validation on the kth fold. The average accuracy is a measure of the model's accuracy without bias from splitting the data set in training and testing sets.

Confusion matrices are typically used for classification problems [36], and produces 4 values as displayed in Table 6: TP for True Positives, FP for False Positives, FN for False Negatives, and TN for True Negatives. In the context of this project, true positives are true Artifacts that are correctly predicted by a model to be artifacts. False positives are True Variants that are mistakenly predicted as artifacts. False negatives are true artifacts that are mistakenly predicted to be variants. Lastly, true negatives are true variants that are correctly predicted to be variants.

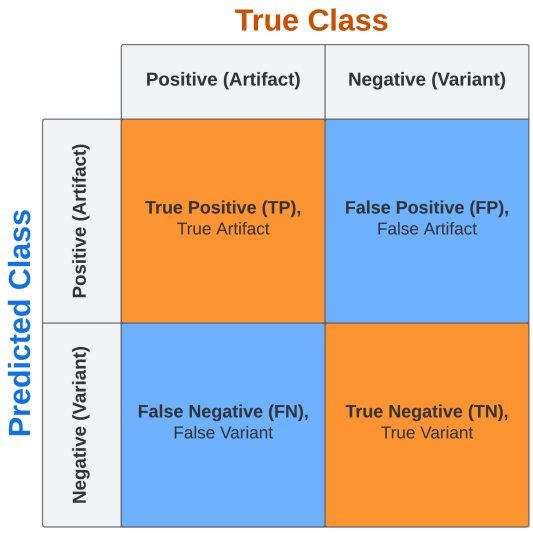

Figure 6: Confusion Matrix in the Context of Sequencing Artifacts

From the values of the confusion matrix, three more metrics are produced: Accuracy, Precision, and Recall [36]. Precision is a metric for how accurately a model can predict positive outcomes correctly and is calculated by the following:

$$Precision = \frac{TP}{(TP + FP)}$$

Recall, also known as sensitivity, is a measure of a models strength to predict positive outcomes, and is calculates by the following:

$$Recall/Sensitivity = \frac{TP}{(TP + FN)}$$

Lastly, accuracy is a measure of the model's ability to correctly identify both TP and TN, and is calculated by the following equation:

$$Accuracy = \frac{TP + TN}{(TP + FN + FP + TN)}$$

Additionally, a learning curve is created to evaluate the model and the optimal amount of training data. This is done by training over several data sets of different

sizes and evaluating both the training score and validating score. This is done using sklearn's learning curve. [37, 38, 39]

2.8 Experiments

2.8.1 Measuring Predictive Power

To measure the predictive power of the features from Figure 17, features were removed iteratively from most importance to least importance and trained a random forest model.

2.8.2 AF 0.01 to 0.03 BAMSurgeon Spike-ins

AF was the strongest feature in the work done by Y. Leung [40], and also remains a strong feature in my model training. By using BAMSurgeon, AF can be manipulated to create an even more challenging data set by spiking in extremely low allele frequencies. BAMSurgeon was used to spike in 40,000 SNV's with AF's between 0.01 to 0.03 into two samples, DRR189730 and DRR189732. A small set of samples is done to briefly check whether the performance of the models are affected.

2.8.3 Removing Case 1 True Variants from Data Set

The AF gap between germline variants (Case #1) and artifacts (Case #2 and #4) makes these cases fairly distinguishable. The challenge, however, is in distinguishing between low AF cancer variants (in this case, our Case 5 BAMSurgeon spike-ins) from low AF artifacts. To decrease the power of AF as a feature and study the importance of other features, models were trained on the dataset after the removal of germline variants (Case #1).

2.8.4 Sub-Sampling Equal Amounts of BAMSurgeon Variants and Artifacts

To balance the data set and further challenge allele frequency as a predictive feature, equal amounts of BAMSurgeon somatic variants and Artifacts were sampled within the range allele frequency range.

CHAPTER 3

Results

3.1 Raw Data FastQC

Figure 7 displays the mean quality score per base pair for all samples. Overall, most FASTQ reads are within the green area, indicating these base pair positions are of passing quality. However, early base pair positions for a few of the samples dip in the yellow. Figure 8 displays the quality score per sequence. Peaks in the green area indicate most of the reads are of high quality. Lastly, Figure 9 measure the adapter content of each sample. Almost all samples curve up into the yellow or red regions indicating that adaptor content is present.

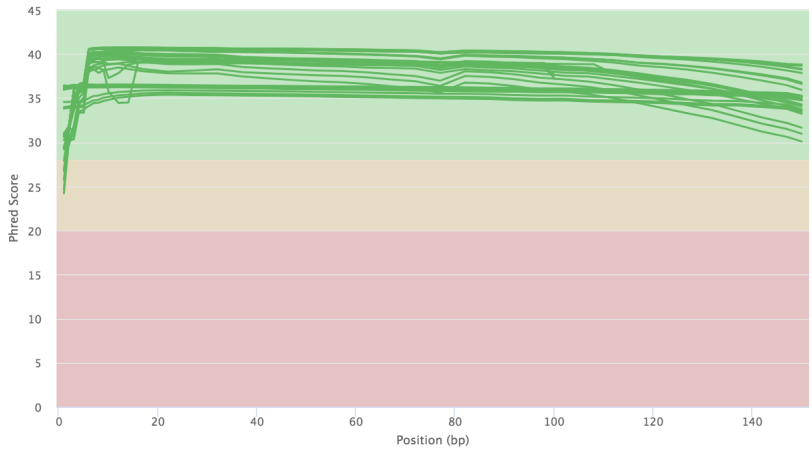

Figure 7: Raw Mean Per Base Quality Scores
This figure displays the mean per base quality score, phred score, of all samples used in this project. This graph was created with MultiQC, which combines the FastQC files of all the samples to be displayed in one format.

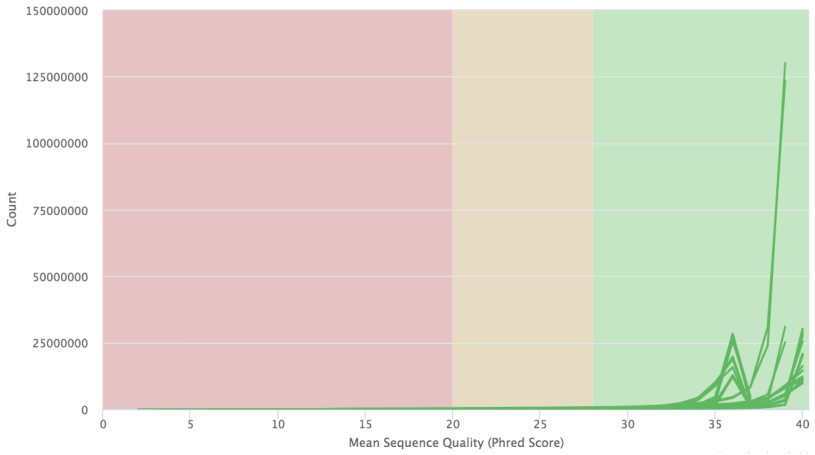

Figure 8: Raw Per Sequence Quality Scores
This figure displays the quality score, phred score, per sequence. Green indicates high quality. This graph was created with MultiQC, which combines the FastQC files of all the samples to be displayed in one format.

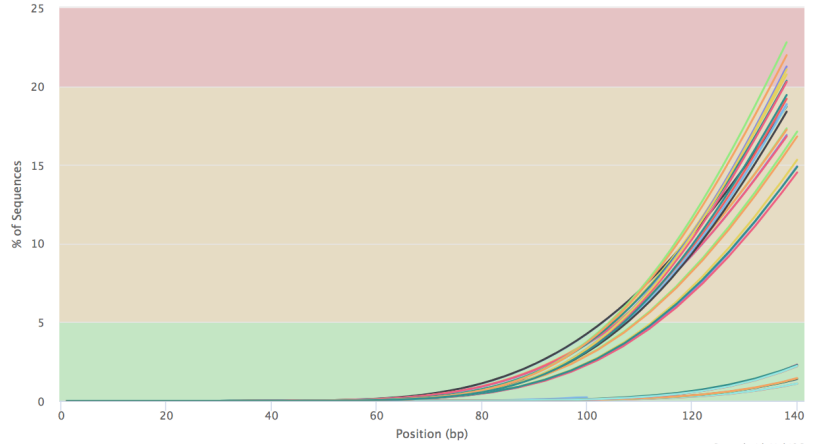

Figure 9: Raw Adapter Content
This figure displays whether raw adaptor content is present in the sample. Each line corresponds to a sample and any increase in the y-axis indicates adaptors are present. This graph was created with MultiQC, which combines the FastQC files of all the samples to be displayed in one format.

3.2 Trimmomatic & QC

After trimmomatic, we see overall improvement in the mean quality scores per base pair in Figure 10 and per sequence quality score in Figure 11. However, some samples still contain adapters, most are trimmed, as seen in Figure 12. This occurred

due to giving trimmomatic the incorrect path to the adapters for the first few samples.

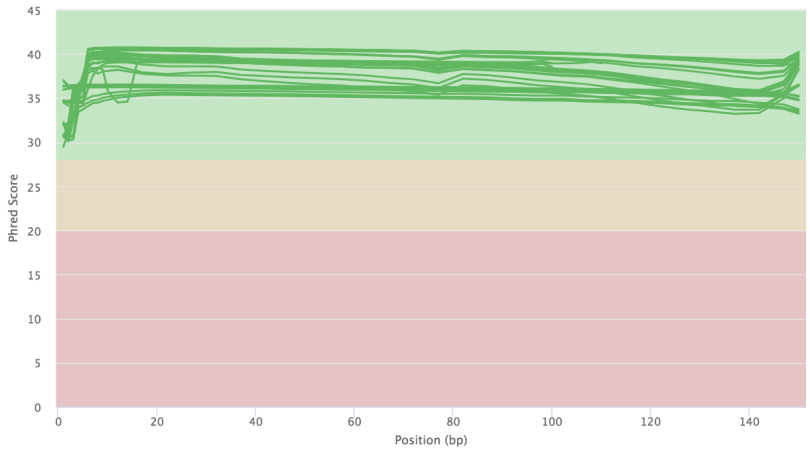

Figure 10: Post-Trimmed Mean Per Base Quality Scores
This figure displays the mean per base quality score, phred score, of all samples used in this project. This graph was created with MultiQC, which combines the FastQC files of all the samples to be displayed in one format.

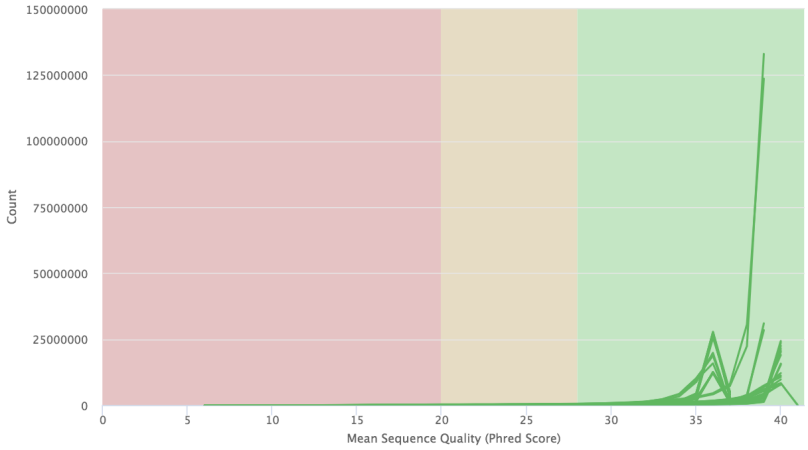

Figure 11: Post-Trimmed Per Sequence Quality Scores
This figure displays the quality score, phred score, per sequence. Green indicates high quality. This graph was created with MultiQC, which combines the FastQC files of all the samples to be displayed in one format.

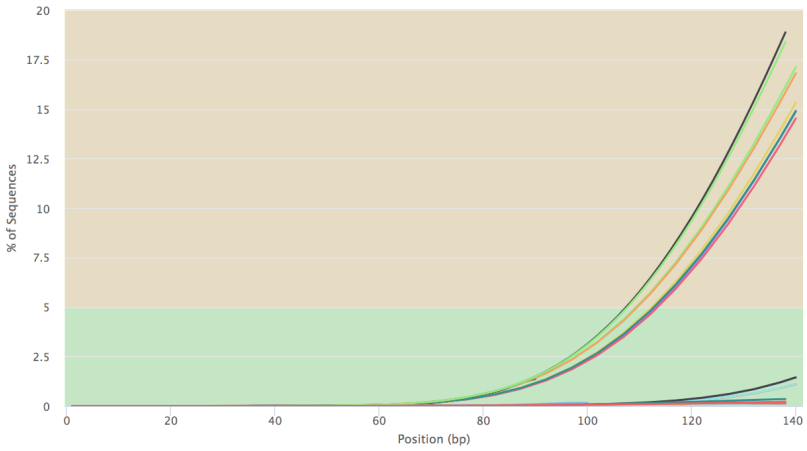

Figure 12: Post-Trimmed Adapter Content
This figure displays whether raw adaptor content is present in the sample. Each line corresponds to a sample and any increase in the y-axis indicates adaptors are present. This graph was created with MultiQC, which combines the FastQC files of all the samples to be displayed in one format.

3.3 Spike in SNVs: BAMSurgeon

The resulting amount of BAMSurgeon variants are displayed in the AF histogram in Figure 13. The sample true variants and atrifacts are included in the histogram as well. The contains a histogram of the allele frequencies for Variants, Artifacts, and BAMSurgeon added variants.

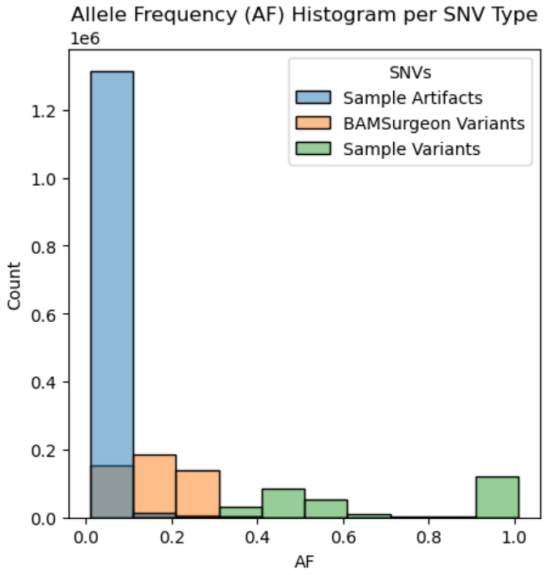

Figure 13: Allele Frequency (AF) Histogram per SNV Type
A histogram of the allele frequencies for Variants, Artifacts, and BAMSurgeon added variants.

3.4 Artifacts and Variant Exploration

The total number of cases per sample is displayed in Table 4 and in the histogram in Fig 14. The number of BAMSurgeon Spikes (Case #5) is expected to be 40,000, however not all spikes made it into the final VCF file.

Table 4: Case Counts per Sample

Each variant is classified into either Case 1 sample Variant, Case 2 Artifact, Case 4 Artifact, or Case 5 BAMsurgeon Variant. The total of each case per sample is recorded in this table.

Sample	**Case #1**	**Case #2**	**Case #4**	**Case #5**	**Total SNVs**
SRR14724459	23876	3	115765	30489	170133
SRR14724463	24329	6	120805	31414	176554
SRR14724462	28912	9	136732	34111	199764
SRR14724469	16010	1	63168	31030	110209
SRR14724473	16134	0	58990	32079	107203
SRR14724503	16123	1	82342	31659	130125
SRR14724479	17893	3	73093	31003	121992
SRR14724489	23656	6	103721	30010	157393
SRR14724499	15941	1	65284	29773	110999
SRR14724508	17735	2	74384	29841	121962
SRR14724472	19258	4	72730	34451	126443
SRR14724482	21334	7	74741	34711	130793
SRR14724492	28668	13	121855	33748	184284
SRR14724502	19170	3	80129	33551	132853
SRR14724512	21177	9	87347	33381	141914

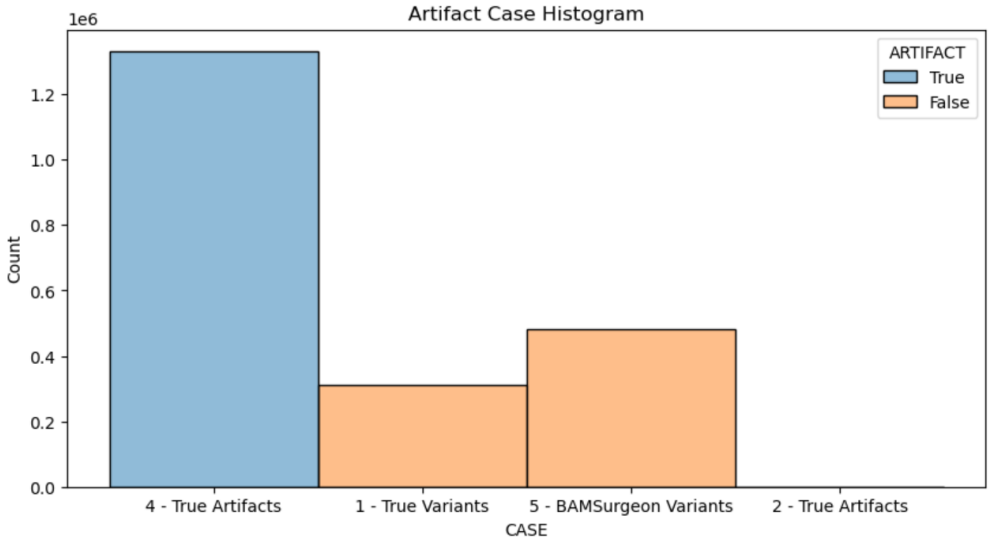

Figure 14: Artifact Case Histogram
The total amount of artifacts, true variants, and BAMSurgeon variants among all samples are displayed in this histogram.

Transition conversions occur more frequently [31], as observed in the heat map in Figure 15, where the largest amount of mutations are from G to A, followed by C to T. The conversion from C to T is also a common artifact caused by the deamination of cytosine (C) for FFPE stored tissue [5].

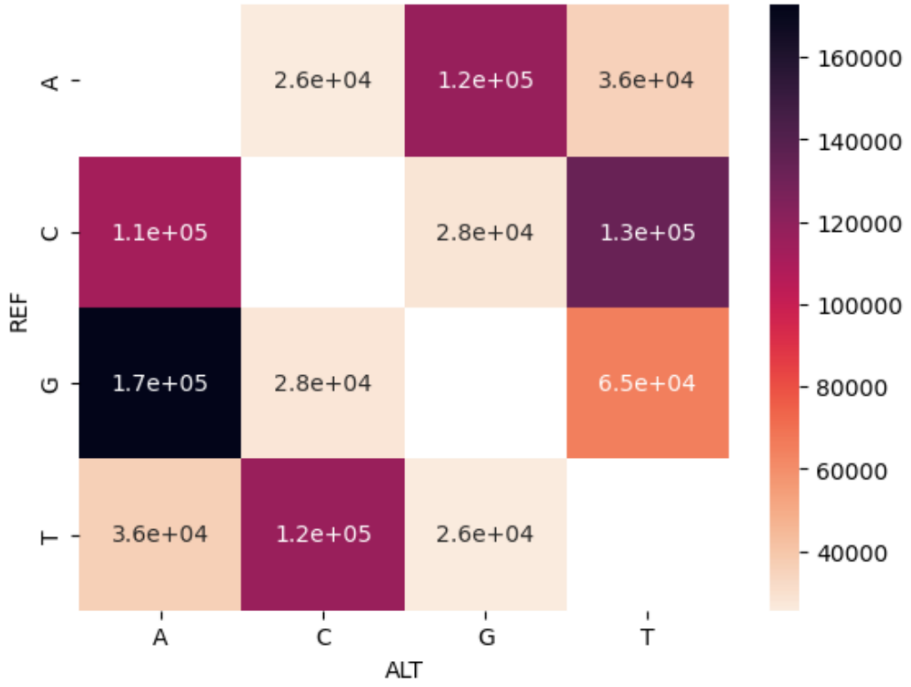

Figure 15: REF to ALT Heatmap
This heatmap displays the frequency of base changes from REF (reference call) to ALT (alternative call).

3.5 LazyPredict

LazyPredicts runs the default hyperarameters of 20 models and produces: Accuracy Score, Balanced Accuracy, ROC AUC, F1 Score, and Time Taken. [35] The output of Lazy predict is displayed in Figure 16. RandomForestClassifier scored the highest accuracy of 0.99 and balanced accuracy 0.98. This is followed by ExtraTreesClassifier, BaggingClassifier, AdaBoostClassifier, DecisionTreesClassifier, and ExtraTreeClassifier. The top 6 models produced by LazyPredict are all ensemble learning algorithms, which are typically preferred for medium to large data sets [41]. Since RandomForestClassifier scored the highest, the feature exploration is done with this classifier.

Model	Accuracy	Balanced Accuracy	ROC AUC	F1 Score	Time Taken
RandomForestClassifier	0.99	0.98	None	0.99	383.72
ExtraTreesClassifier	0.99	0.98	None	0.99	414.25
BaggingClassifier	0.98	0.98	None	0.98	444.30
AdaBoostClassifier	0.98	0.98	None	0.98	391.48
DecisionTreeClassifier	0.98	0.98	None	0.98	98.33
ExtraTreeClassifier	0.97	0.97	None	0.97	22.98
SGDClassifier	0.97	0.96	None	0.97	20.46
LogisticRegression	0.96	0.96	None	0.96	29.79
LinearSVC	0.96	0.95	None	0.96	935.20
Perceptron	0.94	0.94	None	0.94	22.65
PassiveAggressiveClassifier	0.94	0.93	None	0.94	20.06
KNeighborsClassifier	0.92	0.91	None	0.92	286.89
LinearDiscriminantAnalysis	0.93	0.90	None	0.93	39.05
RidgeClassifier	0.93	0.90	None	0.93	21.90
RidgeClassifierCV	0.93	0.90	None	0.93	30.53
BernoulliNB	0.91	0.88	None	0.91	23.58
NearestCentroid	0.87	0.83	None	0.86	16.89
GaussianNB	0.51	0.61	None	0.45	19.29
QuadraticDiscriminantAnalysis	0.47	0.58	None	0.39	23.10
DummyClassifier	0.63	0.50	None	0.48	15.99

Figure 16: LazyPredict Model Evaluation
LazyPredict trains the default of several models and procuded an output listing the models from best to worst performing, ranked by accuracy. Balanced accuracy, F1 Score, and Time Taken is also included. This method allows several models to be evaluated in a shorter amount of time.

3.6 Feature Exploration

3.6.1 Feature Importance

Two methods were used to explore feature importance. Since Random Forest Classification was the top model from LazyPredict, this is the model that was used. The first method created by Leung [40], Random Forest Classification is trained for 1000 iterations, and each iteration is a different and random combination of 3 features. The accuracy scores are recorded for each iteration along with the 3 features used for training. The accuracy scores are averaged over each feature, producing the Figure 17. From Figure 17, 'SN' has the most predictive power, followed by 'HICNT', 'AF', 'HIAF', and 'VD'.

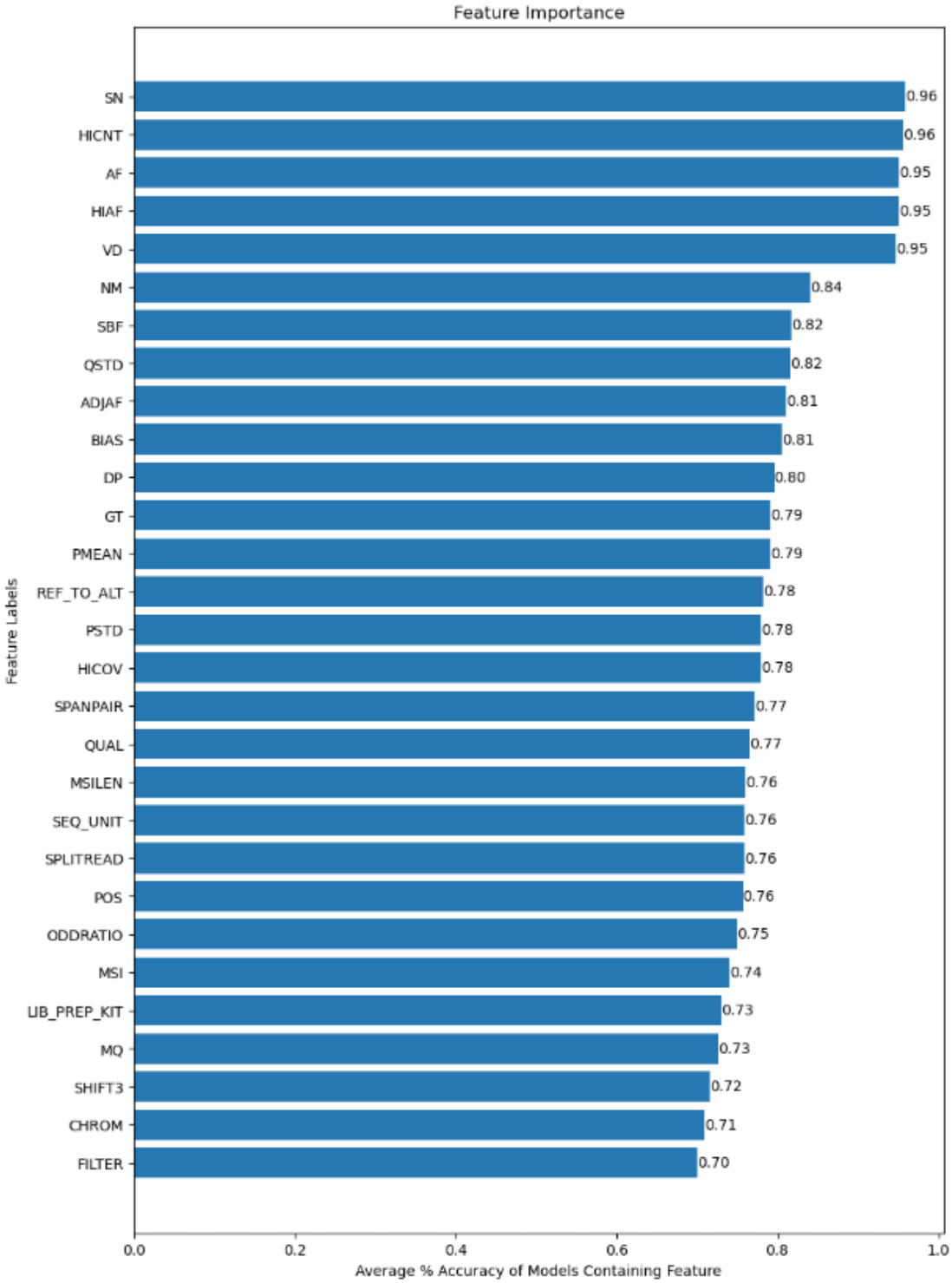

Figure 17: RandomForestClassifier Feature Importance
Feature importance is a measure of how important a feature is for the model's performance. It was calculated by training 1000 iterations of the Random Forest Classifier with a combination of 3 random features for each iteration. The average accuracy of each feature is calculated by averaging the score of the models that included the feature.

In addition to Leung's method for measuring predictive power per feature, sklearn's contains the attribute feature_importance_ for the Random Forest Classifier that measures impurity-based feature importance [42, 43]. Feature importance is measured by how well a feature predicts the target, or in other words, how much a feature is used in each tree. This metric is computed as the (normalized) total reduction of the criterion brought by that feature [44]. The top important features in Figure 18 made by feature_importance_ are similar to those in Figure 17, however the in Figure 18 the top feature is 'VD' followed by 'SN'. In both cases, 'SN', 'VD', 'HIAF', 'HICNT' and 'AF' are important features. However, it is important to note that impurity-based feature importance contains biases toward high cardinality features (features with many unique values) [43].

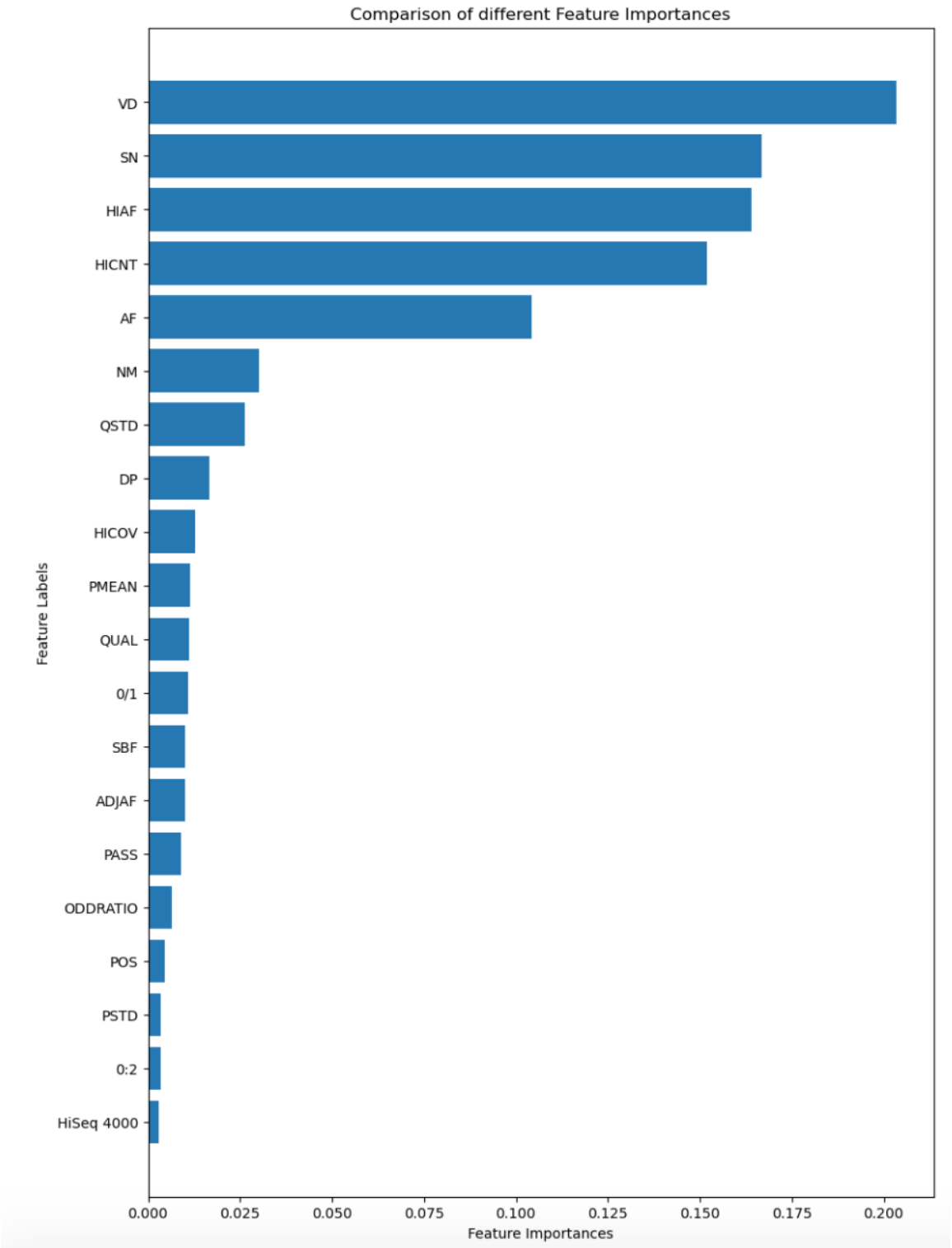

Figure 18: RandomForestClassifier sklearn tree.feature_importance_
Feature importance is a measure of how important a feature is for the model's performance.

A correlogram of the top 5 features is displayed in Figure 19 to view the relationship and histograms among the features.

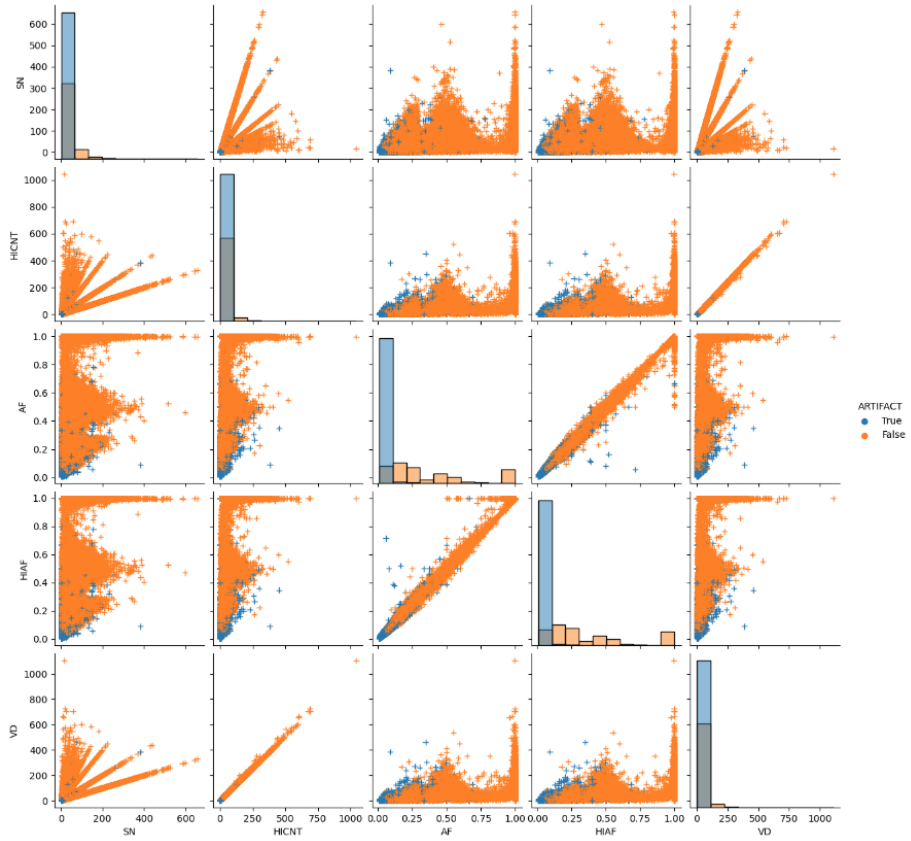

Figure 19: Artifacts Correlogram
This figure displays a correlogram, a series of histograms, for the top five features from the feature importance in 17. The correlation between the different features is displayed in this figure, as well as the histograms for each feature.

3.7 Models

3.7.1 RandomForestClassifier

The random forest classifier is an ensemble learning algorithm that utilizes multiple randomly generated decision trees and bootstrap replicas, in other words, sub-sampling of the data set with replacement when training. RFC reduces the over-fitting of the decision tree algorithm and in turn improves the predictive accuracy of the model [41, 45].

The average 3-fold cross validation accuracy score of the Random Forest Classifier is 97.8%. With the test set, the accuracy of the random forest classifier is 97.8%, the precision is 99.1%, the recall is 97.3%, and the F1 score is 98.2%. The confusion matrix in Figure 20 and Table 5 reflects these values.

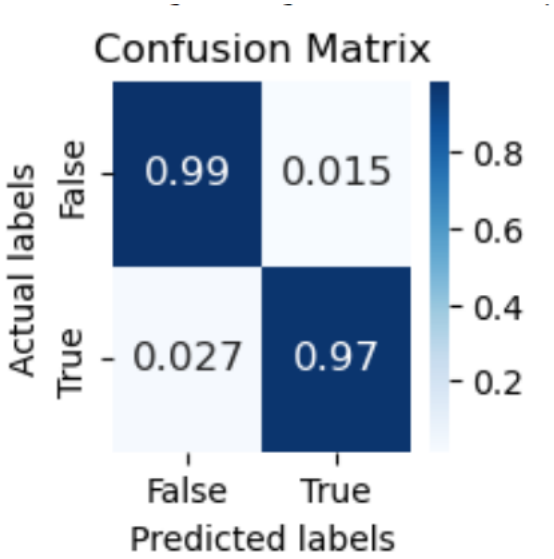

Figure 20: Random Forest Classifier Confusion Matrix
This figure displays the true positives (TP, the percentage of correctly identified artifacts), true negatives (TN, the percentage of correctly identified true variants), false positives (FP, the percentage of falsey identified artifacts), and false negatives (FN, the percentage of falsely identified variants).

Table 5: Random Forest Classifier Results

Metric	Score
Cross-Validation	97.8% 97.8% 97.8%
Accuracy	97.8%
Precision	99.1%
Recall/Sensitivity	97.3%
F1	98.2%

Two learning curves were created for this model and is displayed in Figure 21

and Figure 22. Both the training and validation scores remain high and steady with each sample size. Overall, the model performs well with the given data, even with less training data.

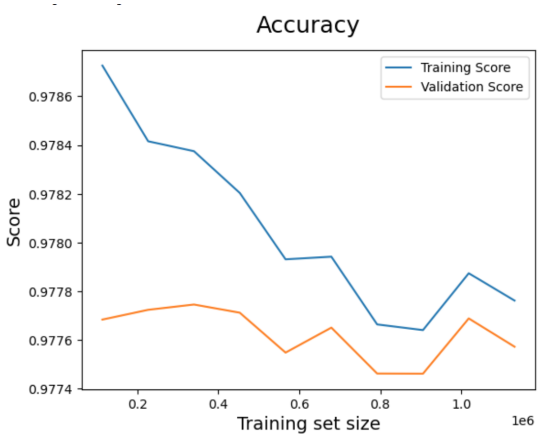

Figure 21: Random Forest Learning Curve for Accuracy Score
The Accuracy training scores and validation scores of the random forest model are calculated over several training sizes.

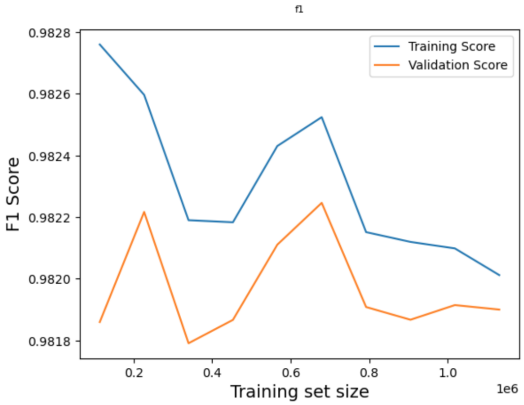

Figure 22: Random Forest Learning Curve for F1 Score
The F1 training scores and validation scores of the random forest model are calculated over several training sizes.

3.7.2 ExtraTreesClassifier

The extra trees classifier is another ensemble learning algorithm. Unlike random forest, extra trees classifier does not utilize bootstrap replicas but rather utilizes the

entire sample. Also, extra trees classifier chooses cut points at random when splitting nodes, while random forest and decision tree chooses the optimal split [46, 47, 41].

The average 3-fold cross validation accuracy score of this model was 91.5%. The test set accuracy is 91.5%, the precision is 89.4%, the recall is 98.0%, and the F1 is 93.5%. The confusion matrix is displayed in Figure 23 and the values are found in Table 6

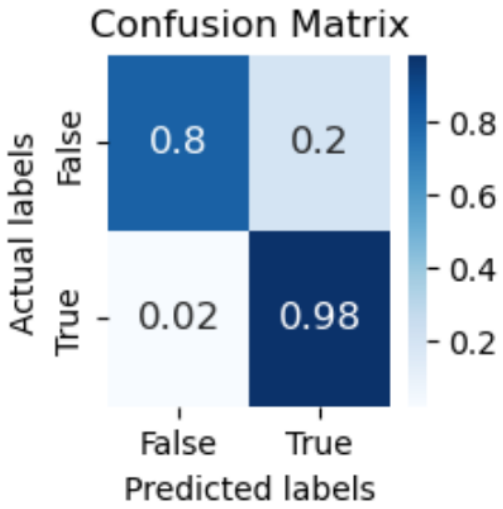

Figure 23: Extra Trees Classifier Confusion Matrix
This figure displays the true positives (TP, the percentage of correctly identified artifacts), true negatives (TN, the percentage of correctly identified true variants), false positives (FP, the percentage of falsey identified artifacts), and false negatives (FN, the percentage of falsely identified variants).

Table 6: Extra Trees Classifier Results

Metric	Score
Cross-Validation	92.5%
	92.3%
	92.0%
Accuracy	91.5%
Precision	89.4%
Recall/Sensitivity	98.0%
F1	93.5%

Two learning curves for accuracy and F1 score were created for this model and is displayed in Figure 24 and Figure 25. Both the training and validation scores in both plots converge and within scores around 90%. Over an increase of sample size, the scores intially dip but then steadily increase.

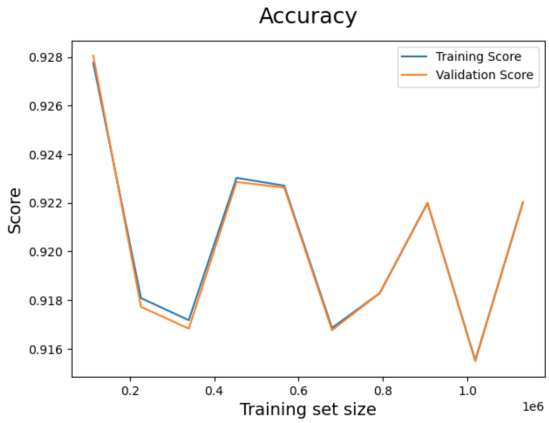

Figure 24: Extra Trees Classifier Learning Curve for Accuracy Score
The Accuracy training scores and validation scores of the random forest model are calculated over several training sizes.

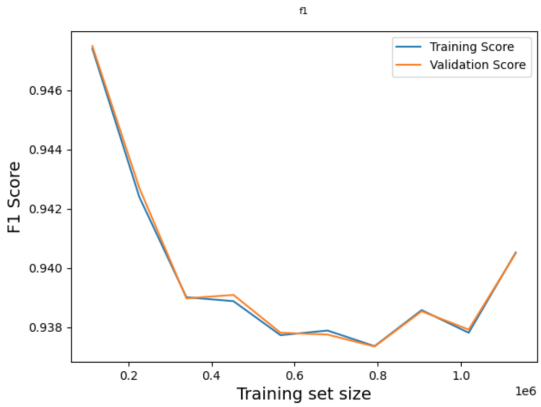

Figure 25: Extra Trees Classifier Learning Curve for F1 Score
The F1 training scores and validation scores of the random forest model are calculated over several training sizes.

3.7.3 BaggingClassifier

The bagging classifier is an ensemble learning algorithm that performs a final prediction based on the aggregate of predictions performed on random subsets [48].

The average 3-fold cross validation accuracy score is 97.6%. The test set accuracy is 97.6%, the precision is 99.0%, the recall is 97.1% and the F1 is 98.0%. The confusion matrix is displayed in Figure 26 and the model scores are in Table 7.

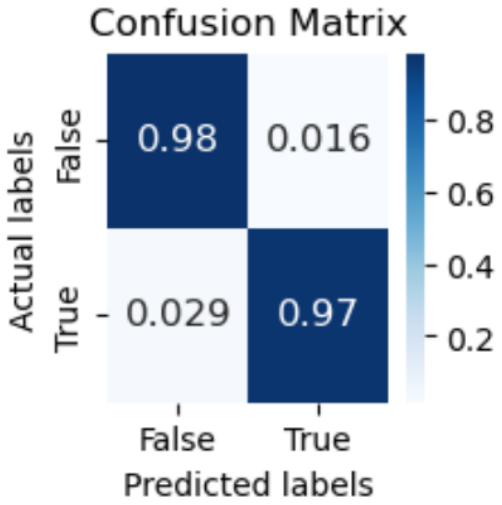

Figure 26: Bagging Classifier Confusion Matrix
This figure displays the true positives (TP, the percentage of correctly identified artifacts), true negatives (TN, the percentage of correctly identified true variants), false positives (FP, the percentage of falsey identified artifacts), and false negatives (FN, the percentage of falsely identified variants).

Table 7: Bagging Classifier Results

Metric	Score
Cross-Validation	97.6% 97.6% 97.6%
Accuracy	97.6%
Precision	99.0%
Recall/Sensitivity	97.1%
F1	98.0%

Two learning curves for accuracy and F1 score were created for this model and is displayed in Figure 27 and Figure 28. Both the training and validation scores in both plots converge as the training set size increases.

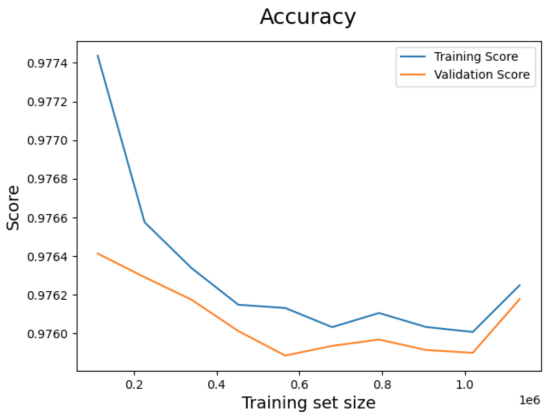

Figure 27: Bagging Classifier Learning Curve for Accuracy Score
The Accuracy training scores and validation scores of the random forest model are calculated over several training sizes.

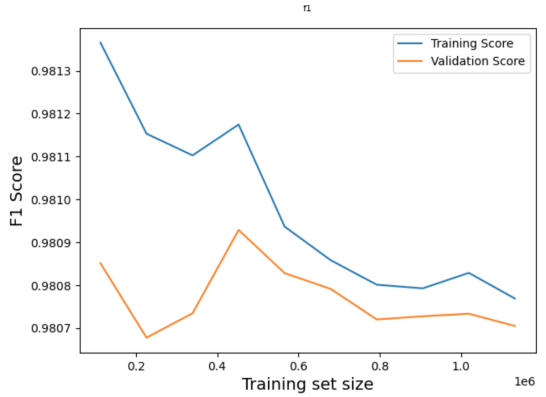

Figure 28: Bagging Classifier Learning Curve for F1 Score
The F1 training scores and validation scores of the random forest model are calculated over several training sizes.

3.7.4 DecisionTreesClassifier

Decision trees are a supervised learning algorithm which makes predictions based on decision rules inferred from data features [49, 50].

The average 3-fold cross validation accuracy is 97.6%. The test set accuracy, precision, recall, and F1 are 97.6%, 99.1%,97.1%, and 98.1% respectively and displayed in Table 8. The confusion matrix is in Figure 29.

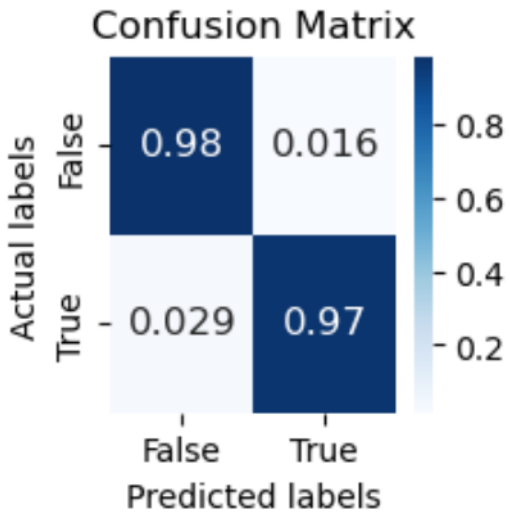

Figure 29: Decition Trees Classifier Confusion Matrix

Table 8: Decision Trees Classifier Results

Metric	Score
Cross-Validation	97.5% 97.6% 97.6%
Accuracy	97.6%
Precision	99.1%
Recall/Sensitivity	97.1%
F1	98.1%

Two learning curves for accuracy and F1 score were created for this model and is displayed in Figure 30 and Figure 31. Both the training and validation scores in both plots converge as the training set size increases.

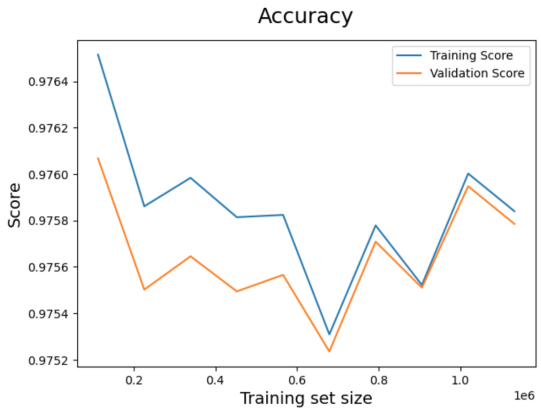

Figure 30: Decision Trees Classifier Learning Curve for Accuracy Score
The Accuracy training scores and validation scores of the random forest model are calculated over several training sizes.

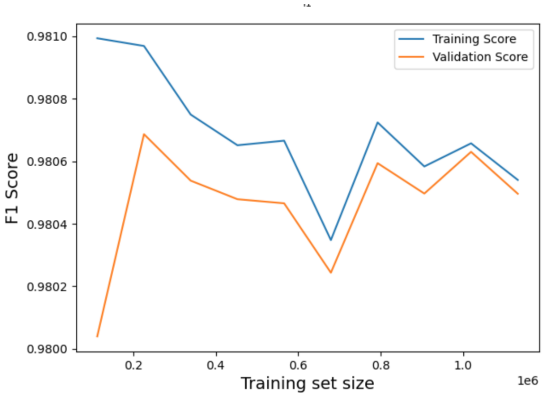

Figure 31: Decision Trees Classifier Learning Curve for F1 Score
The F1 training scores and validation scores of the random forest model are calculated over several training sizes.

3.7.5 LogisticRegression

Logistic regression is a linear classification algorithm [51]. The accuracy, precision, recall, and F1 scores are 96.8%, 96.9%, 98.0%, and 97.5%, and displayed in Table 9. The confusion matrix is in Figure 32.

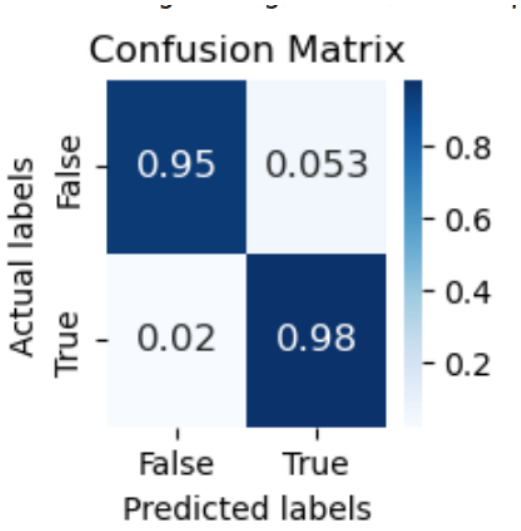

Figure 32: Logistic Regression Classifier Confusion Matrix
This figure displays the true positives (TP, the percentage of correctly identified artifacts), true negatives (TN, the percentage of correctly identified true variants), false positives (FP, the percentage of falsey identified artifacts), and false negatives (FN, the percentage of falsely identified variants).

Table 9: Logistic Regression Classifier Results

Metric	Score
Cross-Validation	96.8% 96.8% 96.8%
Accuracy	96.8%
Precision	96.9%
Recall/Sensitivity	98.0%
F1	97.5%

Two learning curves for accuracy and F1 score were created for this model and is displayed in Figure 33 and Figure 34. In both plots, the accuracy and F1 scores increase as training size increases, and the training score curve and validation score curve converge.

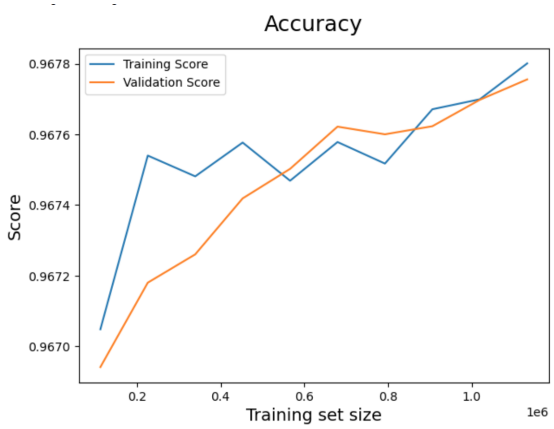

Figure 33: Logistic Regression Learning Curve for Accuracy Score
The Accuracy training scores and validation scores of the random forest model are calculated over several training sizes.

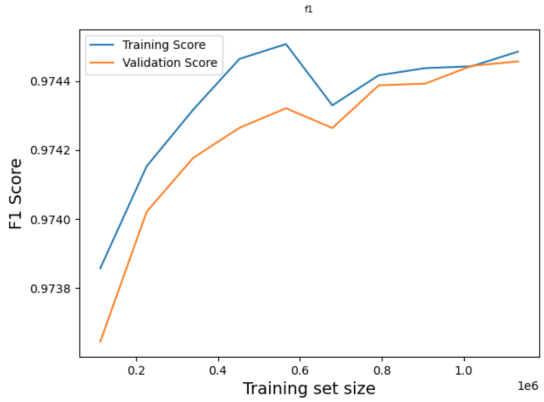

Figure 34: Logistic Regression Learning Curve for F1 Score
The F1 training scores and validation scores of the random forest model are calculated over several training sizes.

3.8 Summary and ROC Curve and Precision-Recall Curve

The Receiver Operating Characteristic (ROC) Curve is a graph of the true positive rate (TPR) vs the false positive rate (FPR) and displays the overall performance of the model. The Area Under the Curve (AUC) score is a measure of the model's ability to discriminate between 2 classes, and summarizes the ROC Curve. The models are compared in Figure 35, and almost all models have a high AUC score indicating they are able to distinguish between artifacts and variants with high accuracy [52]. The

Precision-Recall Curve displays the tradeoff between precision (False positive rate), and recall (False negative rate). High precision indicates a low false positive rate and high recall indicates low false negative rate [52]. The models are compared in Figure 36, and again, all models performed well.

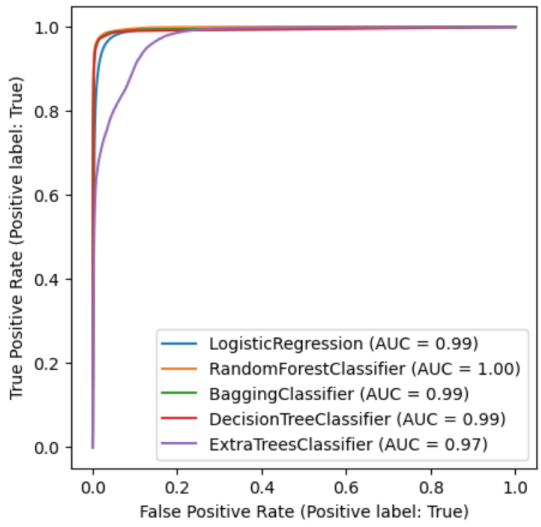

Figure 35: ROC Curve Comparison for ML Classifers on Test Set
The Reciver Operating Characterisic (ROC) Curve graphs true positive rate (TPR) vs. false positive rate (FPR) and displays the overall performance of the model.

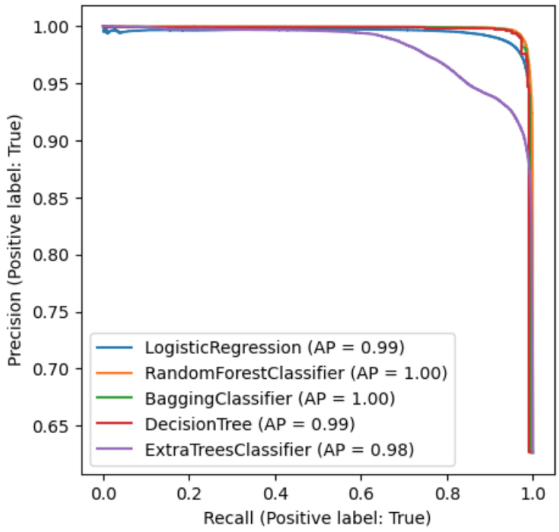

Figure 36: Precision-Recall Curve for ML Classifiers on Test Set
The precision-recall curve displays the tradeoff between precision and recall.

3.9 Experiments

3.9.1 Measuring Predictive Power

The predictive power of each feature displayed in Table 10. From Table 10, a steady decline in accuracy is observed for every removal or a feature. The precision appears to switch between dropping and rising and the recall drops significantly as features are removed.

Table 10: Feature Predictive Power with Random Forest Classification

Features	Accuracy	Precision	Recall
All Features	96.9%	99.2%	96.4%
Previous Row Minus 'SN'	97.3%	98.7%	97.1%
Previous Row Minus 'HICNT'	96.9%	97.2%	97.7%
Previous Row Minus 'AF'	96.9%	97.5%	97.6%
Previous Row Minus 'HIAF'	96.1%	98.3%	95.3%
Previous Row Minus 'VD'	86.9%	90.8%	89.8%
Previous Row Minus 'NM'	86.3%	88.5%	91.9%
Previous Row Minus 'SBF'	82.2%	87.5%	86.2%
Previous Row Minus 'QSTD'	80.8%	85.0%	85.5%
Previous Row Minus 'ADJAF'	78.1%	83.1%	83.9%
Previous Row Minus 'BIAS'	77.9%	83.3%	83.8%
Previous Row Minus 'DP'	77.7%	83.1%	87.2%
Previous Row Minus 'GT'	71.6%	92.1%	66.6%
Previous Row Minus 'PMEAN'	69.5%	90.3%	65.4%
Previous Row Minus 'REF_TO_ALT'	65.0%	85.6%	67.4%
Previous Row Minus 'PSTD'	65.3%	84.1%	62.3%
Previous Row Minus 'HICOV'	60.0%	89.3%	44.9%
Previous Row Minus 'SPANPAIR'	61.7%	89.3%	44.4%
Previous Row Minus 'QUAL'	53.4%	78.0%	36.8%
Previous Row Minus 'MSILEN'	53.5%	73.6%	50.2%
Previous Row Minus 'SEQ_UNIT'	52.3%	73.6%	50.3%
Previous Row Minus 'SPLITREAD'	51.4%	74.1%	48.8%
Previous Row Minus 'POS'	51.2%	74.5%	47.8%
Previous Row Minus 'ODDRATIO'	48.1%	96.8%	17.9%
Previous Row Minus 'MSI'	48.1%	96.8%	17.9%
Previous Row Minus 'LIB_PREP_KIT'	48.1%	96.8%	17.9%
Previous Row Minus 'MQ'	48.1%	96.8%	17.9%
Previous Row Minus 'SHIFT3'	48.1%	96.8%	17.9%

3.9.2 AF 0.01 to 0.03 BAMSurgeon Spike-ins

The histogram of SNVs in Figure 37, BAMSurgeon variants overlap significantly with the sample true artifacts.

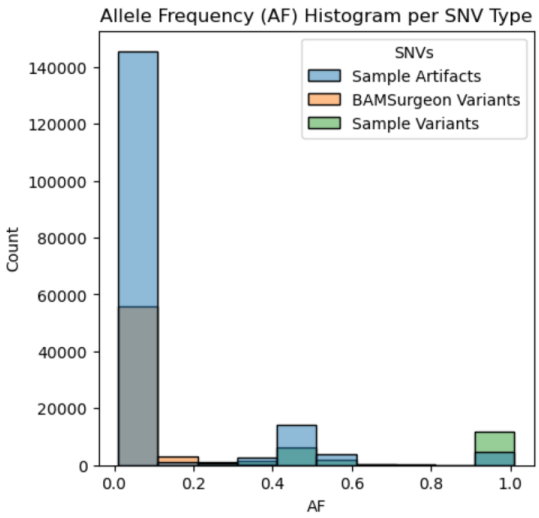

Figure 37: Allele Frequency (AF) Histogram per SNV Type

A Random Forest Classifier is then train on this data, and the accuracy is lower than that of previous models. The accuracy, precision, and recall are 91.7, 92.7, 95.4, respectively and the confusion matrix and scores are in Figure 38 and Table 11.

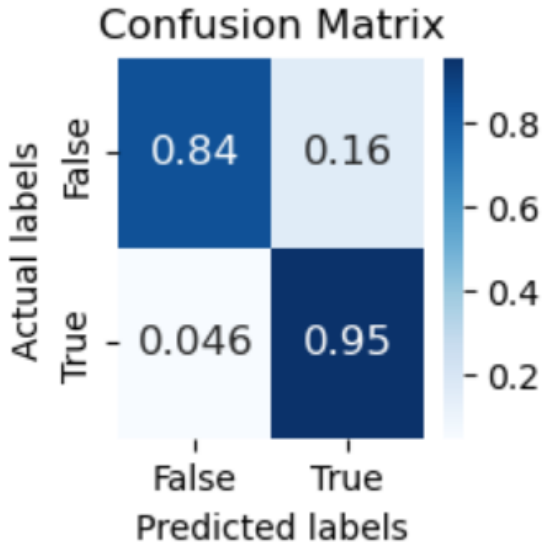

Figure 38: Random Forest Classifier Confusion Matrix
This figure displays the true positives (TP, the percentage of correctly identified artifacts), true negatives (TN, the percentage of correctly identified true variants), false positives (FP, the percentage of falsey identified artifacts), and false negatives (FN, the percentage of falsely identified variants).

Table 11: Random Forest Classifier Results

Metric	Score
Cross-Validation	91.8% 91.6% 91.8%
Accuracy	91.7%
Precision	92.9%
Recall/Sensitivity	95.4%

The ROC Curve and Precision-Recall Curve (Figure 39) for the training sets have outstanding scores of 1.0, however when these curves (Figure 40) are created on the test set, the scores are slightly lower: 0.98 for the AUC score and 0.99 for the AP score.

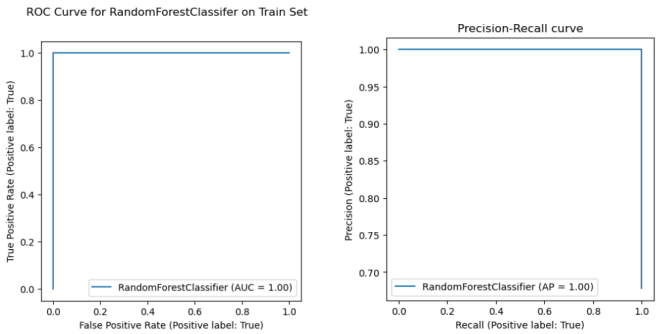

Figure 39: Random Forest Classifier ROC and Precision-Recall Curve for Train Set

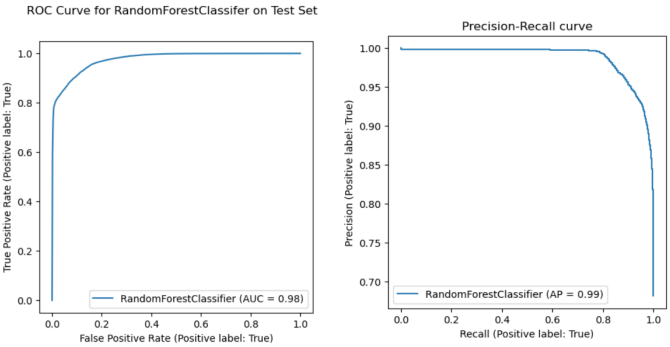

Figure 40: Random Forest Classifier ROC and Precision-Recall Curve for Train Set

Additionally, upon looking at the feature importance in Figure 41 utilizing sklearn.featre_importance_ [42] , the top features are similar to what is observed previously.

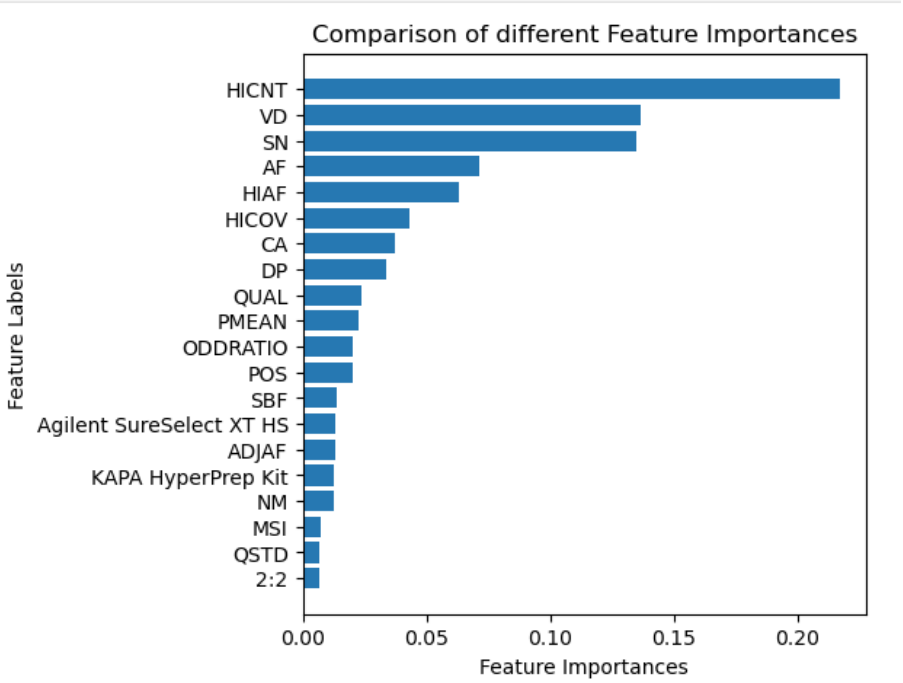

Figure 41: Random Forest Classifier Feature Importance

3.9.3 Removing Germline Variants from Data Set

Germline variants are removed and the remaining data consists of Artifacts and BAMSurgeon somtic variants. The histogram of allele frequency counts are divided by color for variant type and are in Figure 42. There is more overlap in allele frequency between variants and artifacts.

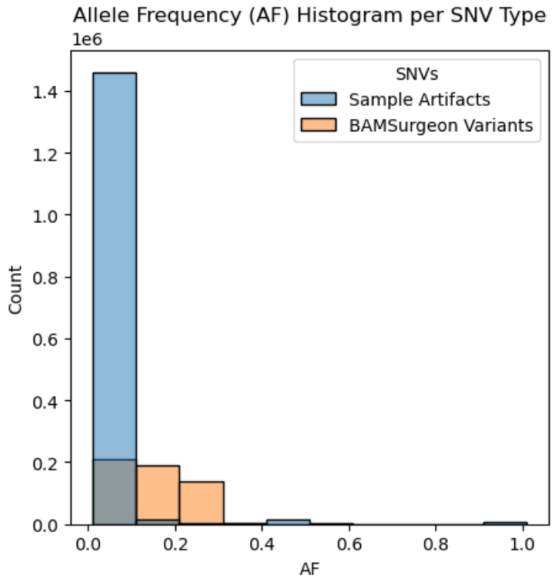

Figure 42: Allele Frequency (AF) Histogram per SNV Type

The model exhibited high performance in distinguishing between the BAMSurgeon variant spike-ins and sample artifacts. A 5-fold cross validation was performed and plotted in Figure 43, and random forest classifier performed the highest with accuracy scores hovering around 0.96. This is followed by bagging classifier, decision trees classifier, logistic regression, and extra trees classifier. The ROC-AUC Curve is in Figure 44 and the precision-recall curve is in Figure 45. Random Forest Classifier has a perfect AUC score, indicating that its identifying true artifacts perfectly. The other models have high AUC scores as well. Random Forest Classifier also has a perfect precision-recall score.

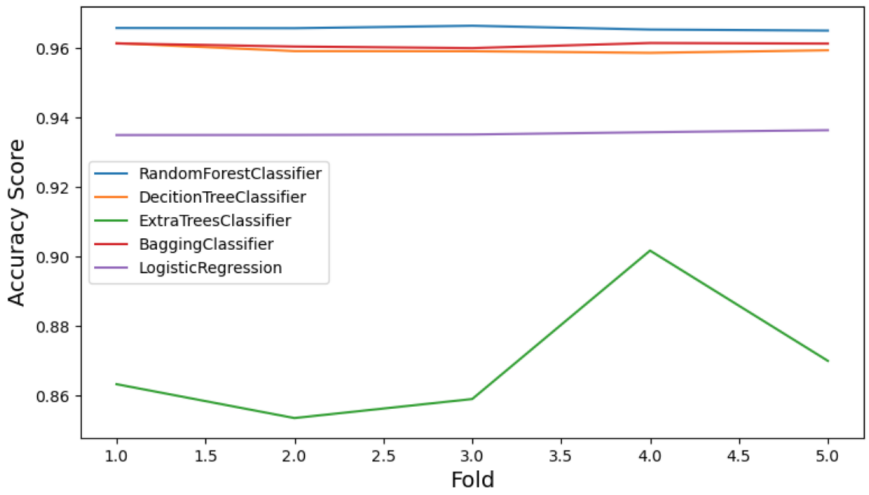

Figure 43: 5-Fold Cross Validation for Random Forest Classifier

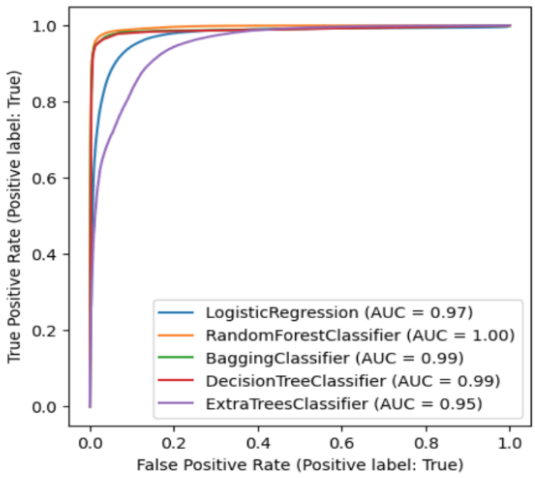

Figure 44: ROC-AUC Curve Comparison of Classification Models

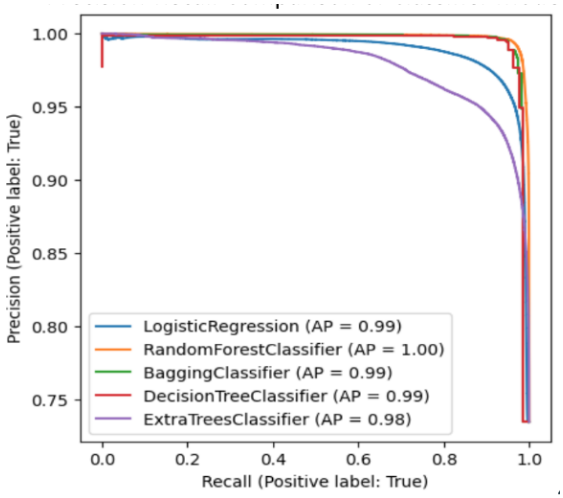

Figure 45: Precision-Recall Comparison of Classification Models

Since random forest classifier performed the strongest, the hyper-parameters of this moderl were tweaked. Several hyper-parameters were tested: n_estimators, min_samples_leaf, min_samples_split, and max_depth. To ensure the model was not over-fitting, training set and testing set AUC was plotted over a range of values for max_depth, and plotted in Figure 46. The point at which the training set AUC and testing AUC separate is where over-fitting occurs [53].

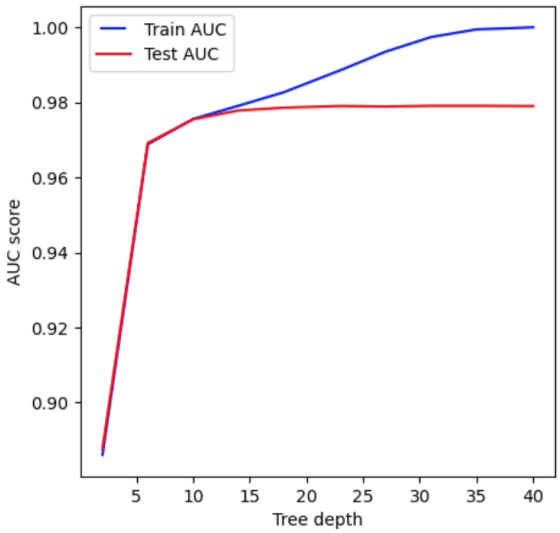

Figure 46: AUC vs. Tree Depth
The AUC (Area Under the Curve) plotted over a range of values for tree depth.

The best parameters were found to be n_estimators = 80, min_samples_leaf = 1, min_samples_split = 5, and max_depth = 11.

The accuracy, precision, recall, and F1 scores are 98.3, 98.9, 98.9, and 98.0% respectively, and the confusion matrix and scores are in Figure 47 and Table 12.

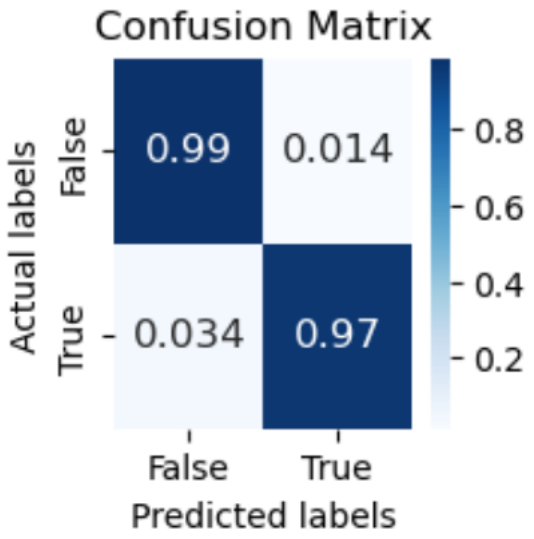

Figure 47: Random Forest Classifier Confusion Matrix
This figure displays the true positives (TP, the percentage of correctly identified artifacts), true negatives (TN, the percentage of correctly identified true variants), false positives (FP, the percentage of falsey identified artifacts), and false negatives (FN, the percentage of falsely identified variants).

Table 12: Random Forest Classifier Results

Metric	Score
Cross-Validation	97.2% 97.2% 97.2%
Accuracy	97.4%
Precision	99.5%
Recall/Sensitivity	96.6%
F1	98.0%

Two learning curve was created for accuracy (Figure 48) and F1 (Figure 49) scores to evaluate the model training over varying training sizes. The validation and training scores for both plots converge, with a difference in 0.0005 for accuracy and 0.0003 for F1.

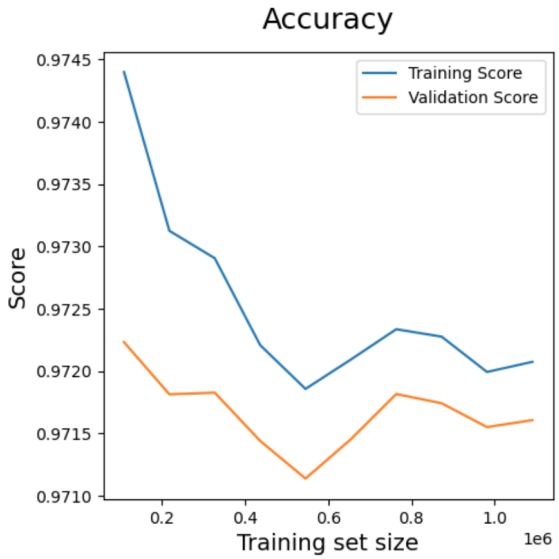

Figure 48: Random Forest Classifier Accuracy Learning Curve, Trained With Data Without Germline Variants
The training scores and validation scores of the random forest model are calculated over several training sizes.

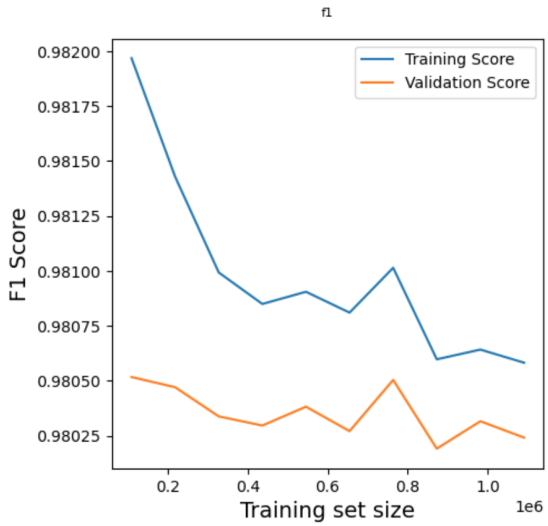

Figure 49: Random Forest Classifier F1 Learning Curve, Trained With Data Without Germline Variants
The training scores and validation scores of the random forest model are calculated over several training sizes.

Additionally, upon looking at the feature importance (Figure 50, the top 5 features

are similar to that of the previous experiment: 'HICNT', 'SN', 'VD', 'HIAF', 'AF'.

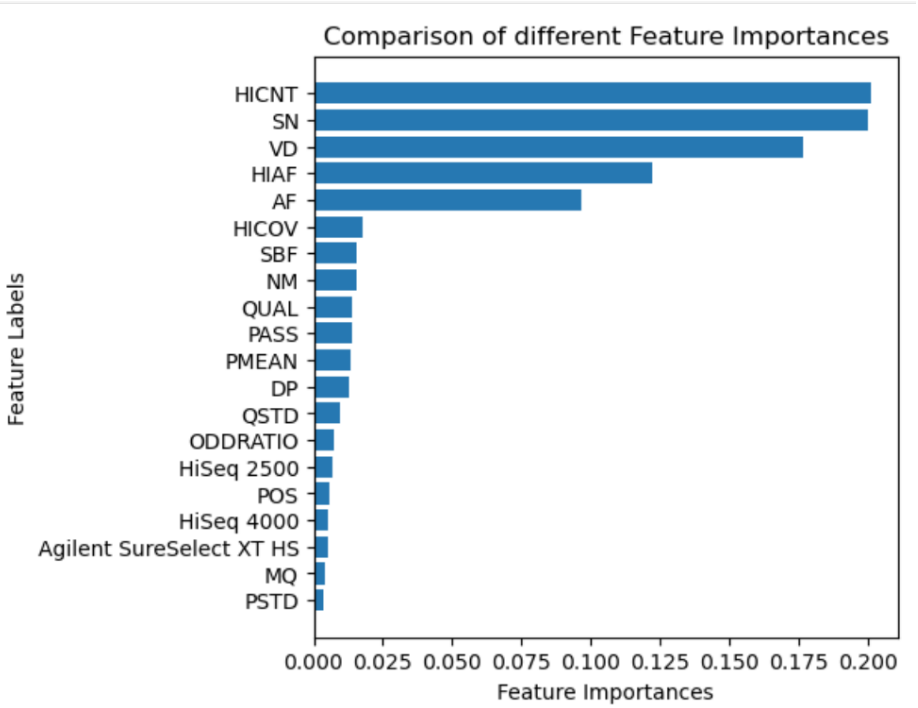

Figure 50: Random Forest Classifier Feature Importance

3.9.4 Sub-Sampling Equal Amounts of BAMSurgeon Variants and Artifacts

Equal amounts of BAMSurgeon somatic variants and artifacts were subsampled, and the histrogram over allele frequency is displayed in Figure 51.

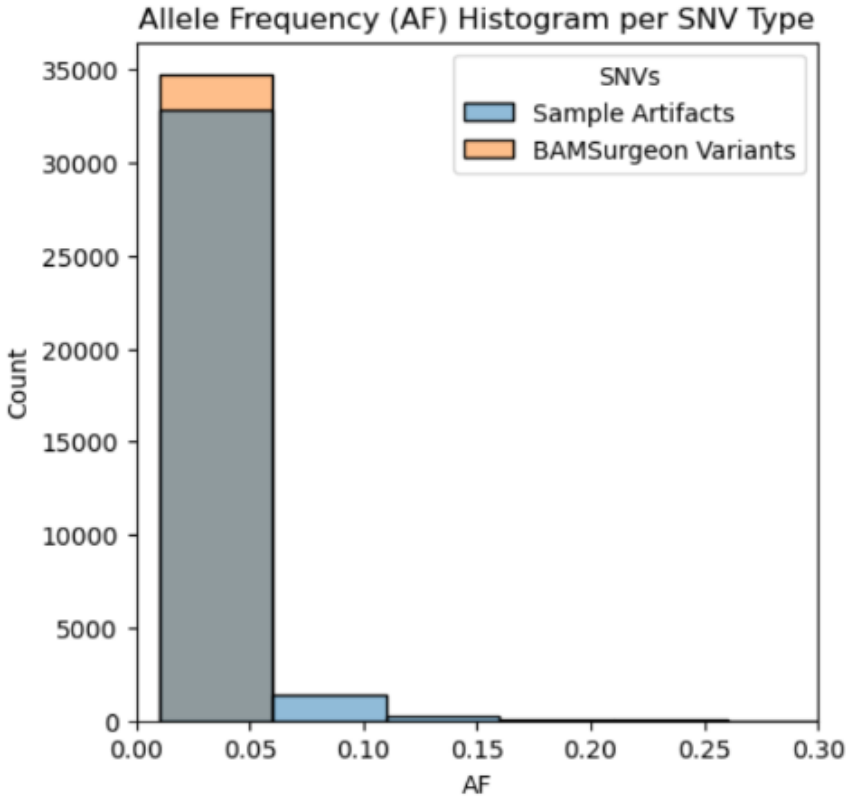

Figure 51: Allele Frequency (AF) Histogram per SNV Type

5-fold cross validation was performed and plotted in Figure 52. Random forest, decision trees, and bagging classifier hover around a 0.95 accuracy score. This is followed by Logistic regression with a cross validation accuracy of around 0.92, and extra trees classifier around 0.82. The ROC-AUC curve and precision-recall curve are displayed in Figure 53 and Figure 54. Random forest classifier and bagging classifier have the highest AUC scores of 0.98, and random forest classifier performs highest in the precision-recall curve.

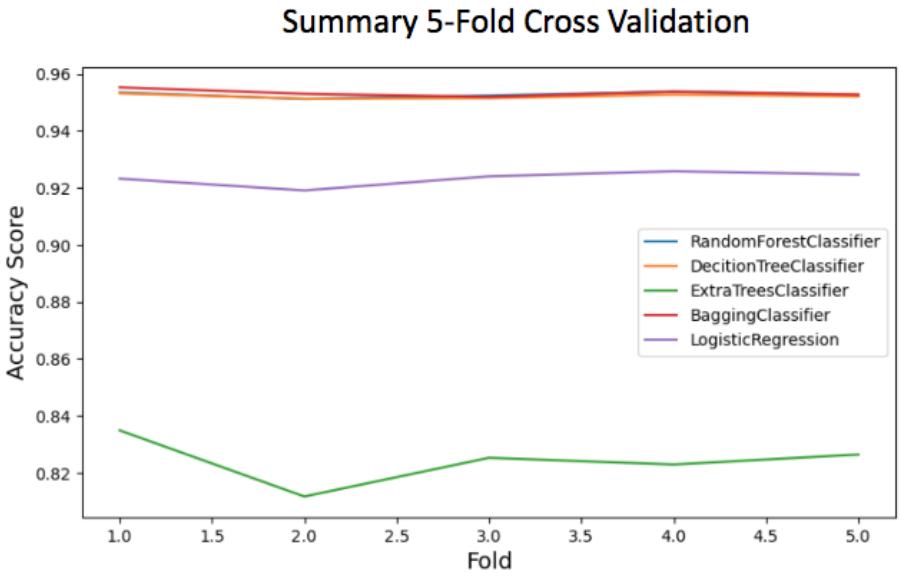

Figure 52: 5-Fold Cross Validation for Random Forest Classifier

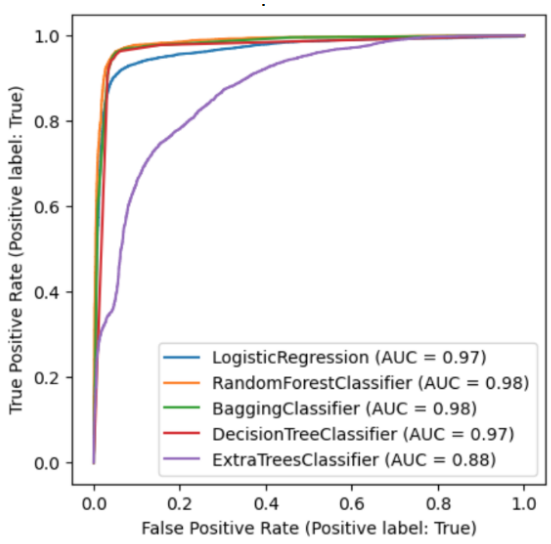

Figure 53: ROC-AUC Curve Comparison of Classification Models

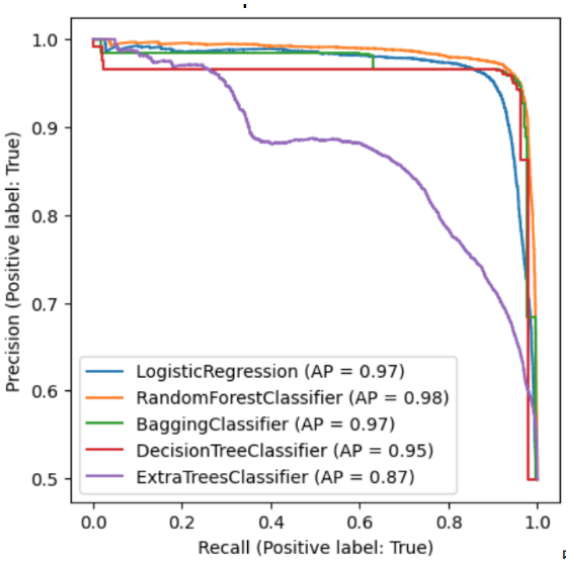

Figure 54: Precision-Recall Comparison of Classification Models

Since random forest classifier had the highest overall performance in comparison to the other models, the hyper-parameters were tuned and the top hyper-parameters were found to be: n_estimators = 80, min_samples_leaf = 1, min_samples_split = 5, and max_depth = 8.

The average cross-validation accuracy of this model was 95.1%. The test set accuracy, precision, recall, and F1 score are 95.6%, 96.0%, 95.3%, and 95.6% respectively. The confusion matrix is displayed in Figure 55 and the metric values are displayed in Table 13

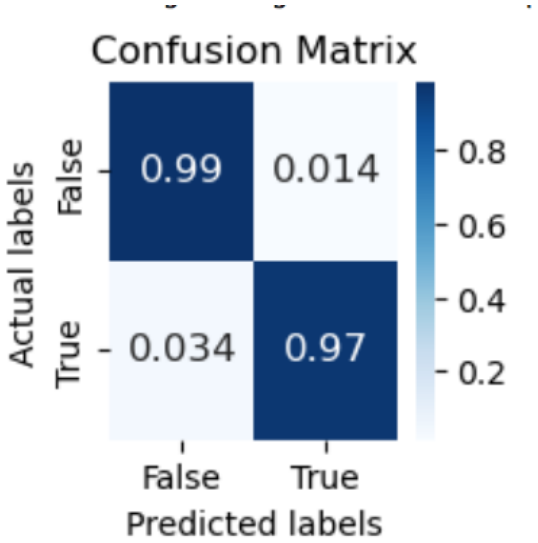

Figure 55: Random Forest Classifier Confusion Matrix
This figure displays the true positives (TP, the percentage of correctly identified artifacts), true negatives (TN, the percentage of correctly identified true variants), false positives (FP, the percentage of falsey identified artifacts), and false negatives (FN, the percentage of falsely identified variants).

Table 13: Random Forest Classifier Results

Metric	Score
Cross-Validation	95.1% 95.1% 95.2%
Accuracy	95.6%
Precision	96.0%
Recall/Sensitivity	95.3%
F1	95.6%

Two learning curves were created for accuracy(Figure 56) and F1(Figure 57) scores. The training score and validation curves in each plot converge as the training set size increases.

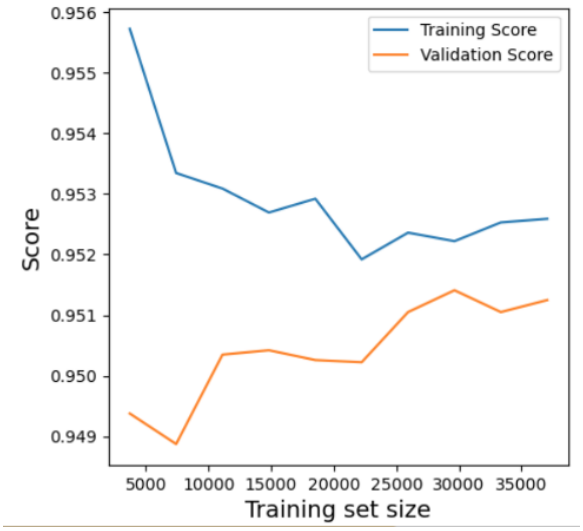

Figure 56: Random Forest Classifier Accuracy Learning Curve
The training scores and validation scores of the random forest model are calculated over several training sizes.

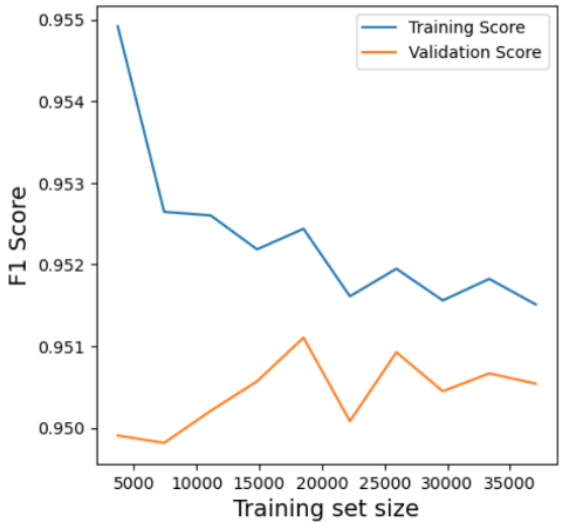

Figure 57: Random Forest Classifier F1 Learning Curve
The training scores and validation scores of the random forest model are calculated over several training sizes.

The feature importance was also explored, and displayed in Figure 58, are the top features for this experiment. As seen in other experiments, 'HICNT', 'VD', 'SN', 'AF', and 'HIAF' are still strong features, however 'AF' does decrease in importance.

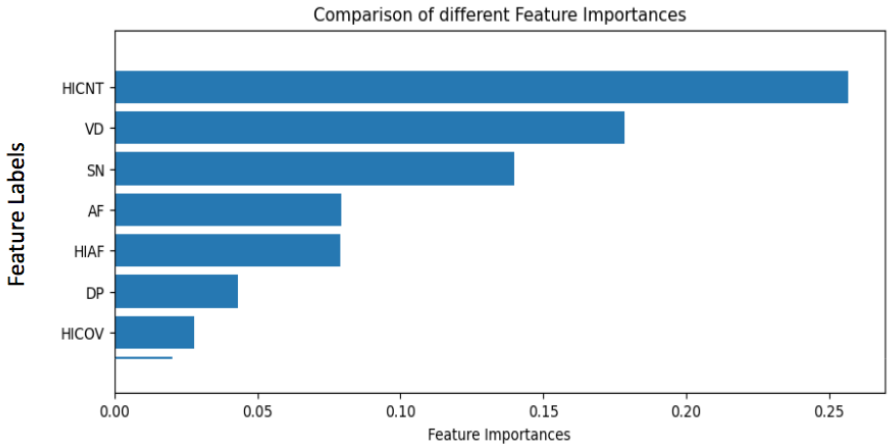

Figure 58: Random Forest Classifier Feature Importance

Since the top 5 features for all experiments were 'HICNT', 'VD', 'SN', 'AF', 'HIAF', a correlogram was created (Figure 59 to display the histogram count of each feature as well as the correlation between each feature. The histogram in each feature shows significant overlap between artifacts and BAMSugeon variants, however in the plots that display the correlation between each feature, the artifacts and variants are visually distinguishable.

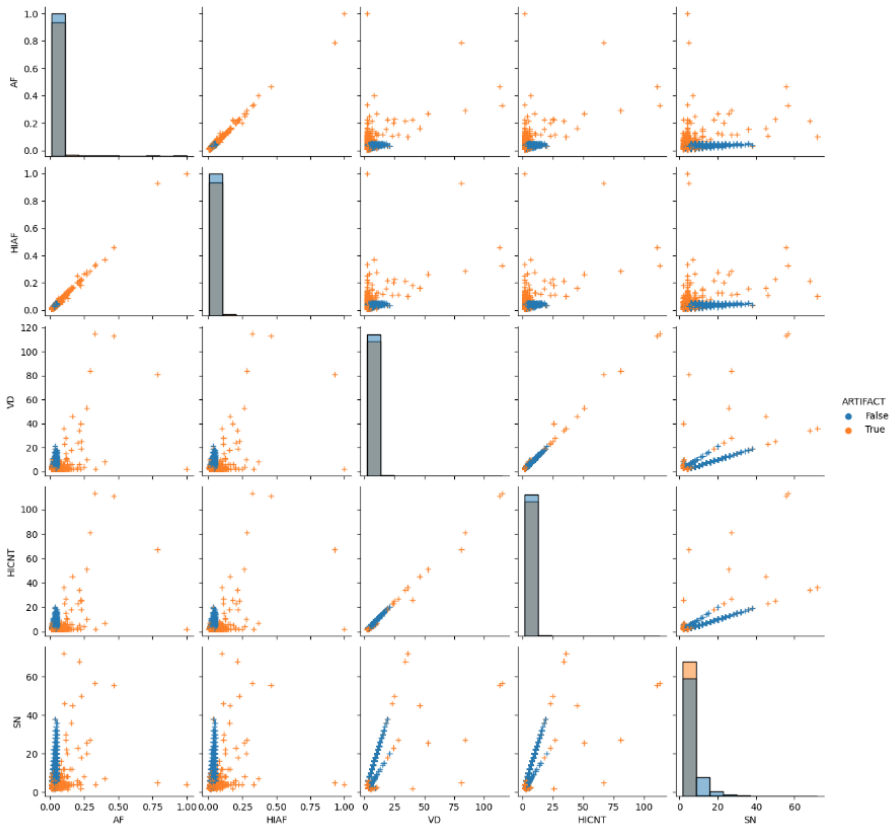

Figure 59: Feature Correlogram
A feature correlogram displays the histogram counts per feature and the correlation between each feature.

CHAPTER 4

Discussion

4.1 Summary and Conclusions

The strongest models for this data set are the ensemble algorithms, as seen from the LazyPredict output in Figure 16, with the top model being Random Forest Classification. Overall, most models have high accuracy scores of at least 0.90 with the highest scoring model, Random forest Classifier, to be 0.99.

Random forest is then used to explore the features of most importance. All the features appear to contribute to training the model efficiently, even the feature with the lowest importance provides an average accuracy score of 0.70 when included for training. The top 5 features that produced accuracy scores of at least 90% are 'SN','HICNT','AF','HIAF', and 'VD', refer to Figure 17. These features are a measure of the variant and read depth and overall quality, and the descriptions are found in the Appendix in Table A.13. The top feature 'SN' is the variant's signal to noise. From the histogram in Figure 19, true artifacts appear to have genrally lower 'SN' values in comparison to variants.

After evaluating LazyPredict and feature importance, Several models were trained: Random Forest Classifier, Extra Trees Classifier, Bagging Classifier, AdaBoost Classifier, Decision Trees Classifier, SGD Classifier, and Logistic Regression. Random forest performed the best as expected, with an accuracy score of 98.6% averaged across 3-fold validation, with a precision of 98.9% and 98.8%. An AUC-ROC Curve and Precision-Recall Curve were created over the train set among all the models to compare the learning abilities. Overall, all models performed exceptionally well.

A few experiments were they conducted. First, the predicted power of each feature was explored. From each feature removal, a steady decline in accuracy is observed, with the largest drop (-10%) at the removal of 'VD', also known as variant

depth.

In another experiment the allele frequency of BAMSurgeon spikes was decreased from the threshold of 0.01 - 0.3 to 0.001 - 0.03. This is done to provide more challenging data for the model to challenge, since 'AF' is an important feature in modeling and a feature that can be manipulated by BAMSurgeon. Upon decreasing allele frequency of spikes in BAMSurgeon, the performance of the Random Forest Classifier decreases slightly and over-fitting is observed.

The third experiment, Case # 1 True Variants (germline variants) were removed from the data set. From the 5-fold cross validation, ROC-AUC, and precision-recall curves that were produced; random forest classifier performed the strongest, and was hyper-tuned. The a accuracy, precision, recall, and F1 score of random forest after hyper-tuning are 97.4%, 99.5%, 96.6% and 98.0% respectively. Upon creating learning curves, as more training data was added, the less over-fitting occurs and the more the validation and training scores converge.

In the last experiment, equal amounts of BAMSurgeon variants and artifacts were sampled for training and testing supervised classification models. A 5-fold cross validation was plotted for 5 classification models. The accuracy score of random forest, decision trees, and bagging classifier hover around 95%, followed by logistic regression at 92% and extra trees classifier at 82%. Overall, the performance of these models in this experiment do not perform as strong as previous experiments. However, an accuracy score of 95% is still very high.

For all experiments, the top 5 feature importance for the random forest classification model were 'HICNT', 'VD', 'SN', 'AF', 'HIAF'. The histogram displays that despite the overlaps that BAMSurgeon low-frequency variants and low frequency artifacts have in each feature, 2 of these features are plotted against each other, there is a visual distinction between the BAMSurgeon variants and artifacts.

4.2 Future Research

BAMSurgeon proved to be a successful tool in spiking in low frequency somatic mutations, however more methods to add somatic mutations could be explored that can target to change not only allele frequency, but other factors as well. The models created in the experiments can also be tested on completely new datasets, or perhaps on actual NGS cancer data sets. Lastly, further exploration can be done on the top 5 features, such as training models with only the top 5 features or a deeper dive into the features for actual cancer somatic variants.

LIST OF REFERENCES

[1] A. Ewing, *BAMSurgeon: Methods for spike-in mutations on BAM files.*

[2] J. Prokop, T. May, K. Strong, S. Bilinovich, C. Bupp, S. Rajasekaran, E. Worthey, and J. Lazar, ''Genome sequencing in the clinic: the past, present, and future of genomic medicine,'' *Physiol Genomics*, aug 2018. [Online]. Available: https://doi.org/10.1152/physiolgenomics.00046.2018

[3] S. Behjati and P. Tarpey, ''What is next generation sequencing?'' *Arch Dis Child Educ Pract Ed.*, vol. 98, dec 2013. [Online]. Available: https://ep.bmj.com/content/98/6/236

[4] D. Muzzey, E. Evans, and C. Lieber, ''Understanding the basics of ngs: From mechanism to variant calling,'' *Curr Genet Med Rep.*, vol. 3, p. 158–165, sept 2015. [Online]. Available: https://link.springer.com/article/10.1007/s40142-015-0076-8

[5] M. Nagahashi, Y. Shimada, H. Ichikawa, H. Kameyama, K. Takabe, S. Okuda, and T. Wakai, ''Next generation sequencing-based gene panel tests for the management of solid tumors,'' *Cancer Science*, vol. 110, pp. 6--15, oct 2018. [Online]. Available: https://doi.org/10.1111/cas.13837

[6] B. Miles and P. Tadi, ''Genetics, somatic mutation,'' apr 2022. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK557896/

[7] ''Various types of variant: what is genomic variation?'' sept 2016. [Online]. Available: https://www.genomicseducation.hee.nhs.uk/blog/various-types-of-variant-what-is-genomic-variation/

[8] N. Deng, H. Zhou, H. Fan, and Y. Yuan, ''Single nucleotide polymorphisms and cancer susceptibility,'' *Oncotarget*, vol. 8, no. 66, pp. 110 635--110 649, 2017. [Online]. Available: https://www.oncotarget.com/article/22372/

[9] D. Koboldt, ''Best practices for variant calling in clinical sequencing,'' *Genome Medicine*, oct 2020. [Online]. Available: https://genomemedicine.biomedcentral.com/articles/10.1186/s13073-020-00791-w#citeas

[10] N. Tanaka and et al, ''Sequencing artifacts derived from a library preparation method using enzymatic fragmentation,'' *PLoS One*, jan 2020. [Online]. Available: https://doi.org/10.1371/journal.pone.0227427

[11] H. Do and A. Dobrovic, ''Sequence artifacts in dna from formalin-fixed tissues: causes and strategies for minimization,'' *Clin Chem.*, jan 2015. [Online]. Available: https://doi.org/10.1373/clinchem.2014.223040

[12] G. Mediratta, E. Ke, M. Aziz, D. Liarakos, M. Tong, and E. Stites, ''Cancer gene mutation frequencies for the u.s. population,'' *Nat Commun*, oct 2021. [Online]. Available: https://doi.org/10.1038/s41467-021-26213-y

[13] F.Castro-Giner, S.Gkountela, C.Donato, I.Alborelli, L. Quagliata, C. K. Y. Ng, S. Piscuoglio, and N. Aceto, ''Cancer diagnosis using a liquid biopsy: Challenges and expectations,'' may 2018. [Online]. Available: https://doi.org/10.3390/diagnostics8020031

[14] J. Zook and et al., ''Extensive sequencing of seven human genomes to characterize benchmark reference materials,'' *Sci. Data*, june 2016. [Online]. Available: https://doi.org/10.1038/sdata.2016.25

[15] NIST, ''Genome in a bottle,'' (Accessed on 03/28/2023). [Online]. Available: https://www.nist.gov/programs-projects/genome-bottle

[16] NIH, ''About the sequence read archive (sra),'' (Accessed on 03/28/2023). [Online]. Available: https://datascience.nih.gov/data-ecosystem/sra

[17] E. Sayers and et al., ''Database resources of the national center for biotechnology information,'' *Nucleic Acids Res.*, jan 2022. [Online]. Available: https://doi.org/10.1093/nar/gkab1112

[18] NIH, ''Download sra sequences from entrez search results,'' (Accessed on 03/28/2023). [Online]. Available: https://www.ncbi.nlm.nih.gov/sra/docs/sradownload/

[19] F. MMölder and et al., ''Sustainable data analysis with snakemake,'' *F1000Research 10,33*, jan 2021. [Online]. Available: https://doi.org/10.12688/f1000research.29032.1

[20] N. SRA, ''Ncbi sra tools,'' 2022. [Online]. Available: https://github.com/ncbi/sra-tools

[21] S. Andrews, ''Fastqc.'' [Online]. Available: https://www.bioinformatics.babraham.ac.uk/projects/fastqc/

[22] P. Ewels, M. Magnusson, S. Lundin, and M. Käller, ''Multiqc: Summarize analysis results for multiple tools and samples in a single report,'' 2016. [Online]. Available: https:https://academic.oup.com/bioinformatics/article/32/19/3047/2196507

[23] K.Goswami and N.Sanan-Mishra, ''Chapter 7 - rna-seq for revealing the function of the transcriptome,'' 2022. [Online]. Available: https://doi.org/10.1016/B978-0-323-89775-4.00002-X

[24] A. B. et al., ''Trimmomatic: A flexible trimmer for illumina sequence data,'' *Bioinformatics*, vol. 30, no. 15, p. 2114–2120, 2014.

[25] H. L. et al., ‘‘Fast and accurate short read alignment with burrows-wheeler transform.’’

[26] H. L. et al., ‘‘samtools.’’ [Online]. Available: http://www.htslib.org/doc/samtools.html

[27] GATK, ‘‘Sortsam (picard).’’ [Online]. Available: https://gatk.broadinstitute.org/hc/en-us/articles/360036510732-SortSam-Picard-

[28] GATK, ‘‘Markduplicates (picard).’’ [Online]. Available: https://gatk.broadinstitute.org/hc/en-us/articles/360037052812-MarkDuplicates-Picard-

[29] Z. L. et al., ‘‘Vardict: a novel and versatile variant caller for next-generation sequencing in cancer research,’’ pp. 1--11, 2016.

[30] A. R. Q. et al., ‘‘Bedtools: a flexible suite of utilities for comparing genomic features,’’ p. 841–842, 2010.

[31] D. M. Lyons and A. S. Lauring, ‘‘Evidence for the selective basis of transition-to-transversion substitution bias in two rna viruses,’’ dec 2017. [Online]. Available: 10.1093/molbev/msx251

[32] Seaborn, ‘‘seaborn.heatmap.’’ [Online]. Available: https://seaborn.pydata.org/generated/seaborn.heatmap.html

[33] Scikit-Learn, ‘‘sklearn.preprocessing.onehotencoder.’’ [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.OneHotEncoder.html#sklearn.preprocessing.OneHotEncoder

[34] Scikit-Learn, ‘‘sklearn.preprocessing.standardscaler.’’ [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

[35] LazyPredict, ‘‘Welcome to lazy predict's documentation!’’ [Online]. Available: https://lazypredict.readthedocs.io/en/latest/index.html

[36] A. Kulkarni, D. Chong, and F. A. Batarseh, ‘‘5 - foundations of data imbalance and solutions for a data democracy,’’ pp. 83--106, 2020. [Online]. Available: https://doi.org/10.1016/B978-0-12-818366-3.00005-8

[37] scikitlearn, ‘‘sklearn.model_selection.learning_curve.’’ [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.learning_curve.html

[38] F. Mohr and J. N. van Rijn, ‘‘Learning curves for decision making in supervised machine learning -- a survey,’’ *Machine Learning*, jan 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2201.12150

[39] scikitlearn, ‘‘3.4. validation curves: plotting scores to evaluate models.’’ [Online]. Available: https://scikit-learn.org/stable/modules/learning_curve.html#learning-curve

[40] Y.Leung, ‘‘Modeling sequencing artifacts for next generation sequencing.’’ [Online]. Available: https://scholarworks.sjsu.edu/cgi/viewcontent.cgi?article=2092&context=etd_projects

[41] M. Schonlau and R. Y. Zou, ‘‘The random forest algorithm for statistical learning,’’ mar 2020. [Online]. Available: https://doi.org/10.1177/1536867X20909

[42] scikitlearn, ‘‘Feature importances with a forest of trees.’’ [Online]. Available: https://scikit-learn.org/stable/auto_examples/ensemble/plot_forest_importances.html#feature-importance-based-on-mean-decrease-in-impurity

[43] scikitlearn, ‘‘Permutation importance vs random forest feature importance (mdi).’’ [Online]. Available: https://scikit-learn.org/stable/auto_examples/inspection/plot_permutation_importance.html

[44] scikitlearn, ‘‘Feature importance.’’ [Online]. Available: https://inria.github.io/scikit-learn-mooc/python_scripts/dev_features_importance.html

[45] scikitlearn, ‘‘sklearn.ensemble.randomforestclassifier.’’ [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[46] scikitlearn, ‘‘sklearn.ensemble.extratreesclassifier.’’ [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.ExtraTreesClassifier.html

[47] P. Geurts, D. Ernst, and L. Wehenkel, ‘‘Extremely randomized trees,’’ pp. 3--42, mar 2006. [Online]. Available: https://link.springer.com/article/10.1007/s10994-006-6226-1

[48] scikitlearn, ‘‘sklearn.ensemble.baggingclassifier.’’ [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.BaggingClassifier.html

[49] scikitlearn, ‘‘1.10. decision trees.’’ [Online]. Available: https://scikit-learn.org/stable/modules/tree.html#tree

[50] scikitlearn, ‘‘sklearn.tree.decisiontreeclassifier.’’ [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html#sklearn.tree.DecisionTreeClassifier

[51] scikitlearn, ‘‘sklearn.linear_model.logisticregression.’’ [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

[52] J.Davis and M.Goadrich, ‘‘The relationship between precision-recall and roc curves.’’ [Online]. Available: https://www.biostat.wisc.edu/~page/rocpr.pdf

[53] M. B. Fraj, ‘‘In depth: Parameter tuning for random forest,’’ *Medium*, dec 2017. [Online]. Available: https://medium.com/all-things-ai/in-depth-parameter-tuning-for-random-forest-d67bb7e920d

Table .14: VCF Meta Information Table

VCF Meta Information/Feature	Description
SAMPLE	Sample name (with white space translated to underscores)
TYPE	Variant Type: SNV Insertion Deletion Complex
DP	Total Depth
END	Chr End Position
VD	Variant Depth
AF	Allele Frequency
BIAS	Strand Bias Info
REFBIAS	Reference depth by strand
VARBIAS	Variant depth by strand
PMEAN	Mean position in reads
PSTD	Position STD in reads
QUAL	Mean quality score in reads
QSTD	Quality score STD in eads
SBF	Strand Bias Fisher p-value
OODRATIO	Strand Bias Odds Ratio
MQ	Mean mapping quality
SN	Signal to noise
HIAF	Allele frequency using only high quality reads
ADJAF	Adjusted AF for indels due to local realignment
SHIFT3	No. of bases to be shifted to 3 prime for deletions due to alternative alignment
MSI	MicroSatellite >1 indicates MSI
MSILEN	MicroSatellite unit length in bp
NM	Mean mismatches in reads
LSEQ	5' flanking seq
RSEQ	3' flanking seq
HICNT	High quality variant reads
HICOV	High quality total reads
SPLITREAD	No. of split reads supporting SV
SPANPAIR	No. of pairs supporting SV
SVTYPE	SV type: INV, DUP, DEL, INS, FUS
SVLEN	The length of SV in bp
DUPRATE	Duplicate rate in fraction
LongMSI	The somatic variants is flaked by long A/T (>=14)
AMPBIAS	Indicate the variant has amplicon bias.
GT	Genotype
DP	Total Depth
VD	Variant Depth
ALD	Variant foward, reverse reads
AD	Allelic depths for the ref and alt allele in the other listed
RD	Reference forward, reverse reads