Next Generation Sequencing Technologies: Data Analysis and Applications

Variant Annotation

Dr. Riddhiman Dhar, Department of Biotechnology

Indian Institute of Technology Kharagpur

Good day, everyone. Welcome to the course on Generation Sequencing Technology Data Analysis and Applications. In the last few classes, we have discussed variant calling, and we have seen hands-on demonstrations on how to do variant calling using SAM tools and BCF tools. We discussed the steps, and then we also discussed how to visualize these mappings or alignments and how to verify some of the observations and some of the variance that we observe in the data. And then we started discussing the VCF file format, where the output of this variant calling data is stored, and we specifically discussed the data lines. So, in this class, we will discuss the meta-information lines and finish the VCF files, and then we will talk about something called variant annotation. So, here is the agenda for this class.

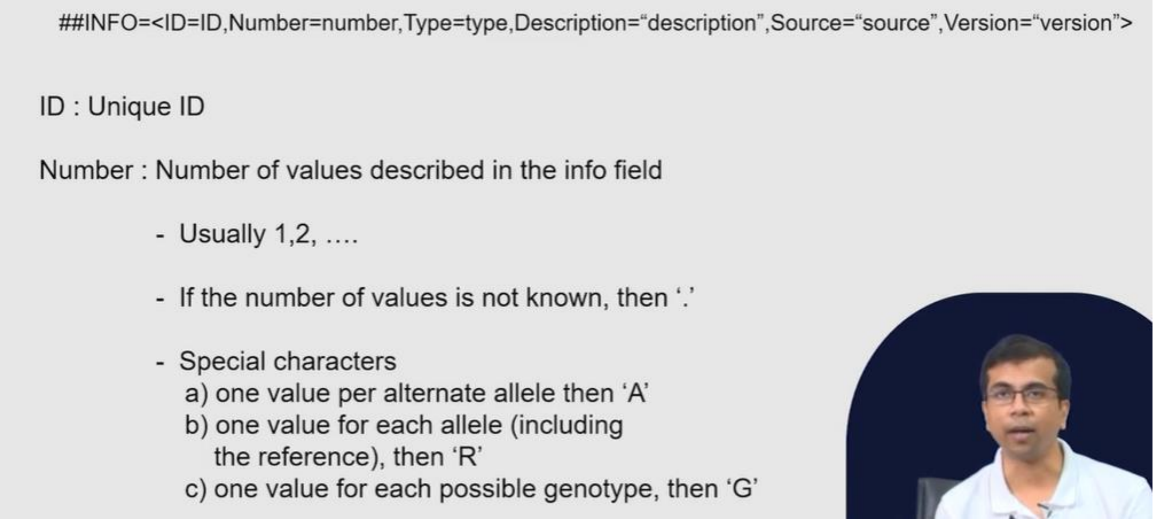
So, we will talk about VCF file meta-information, and then we will talk about variant annotation, what kind of data we need for variant annotation, what we mean by variant annotation, and then the tools that we can use to do variant annotation. So, let us start with the meta-information lines. We discussed the data lines, and we saw certain fields in the data lines that we did not understand completely, and I said those fields would be present in the meta information lines, ok? So, in meta information lines, the lines will start with the sign double hash.

We will see this in a moment when you look at the VCF file again carefully, and these lines will be written in key-to-value format, ok? So, we will see again what we mean by these key equals to value format, which means the key will have some name and then the value will be a certain value. The first line usually says that the file format equals the version number of the VCF we are working with because it is updated and there are minor modifications in the format. And in the next few lines, this will contain the first thing that will contain the source. So, the program that generated the files is okay.

So, in this case right, it should have the name of the program that we actually used. Then the next line will be reference, which is the reference genome. So, in this case, it should be the reference genome that we used; it will just take the file name, and then it will have something like contig assembly. So, this is actually identical to the SQ field from the SAM file, ok? So, it should have these chromosome numbers, or chromosome IDs in our data.

Meta information lines should contain information, filter and format entries. So, we will see what these fields are. We have seen this info filter and format in the data lines, and these should also be present in the meta information lines. So, let us talk about the info field. First, in the info field, we will have the IDs that will be used in the info column of the data lines. So, we will have some IDs that will contain some information, and the same IDs will be used in the data lines to convey some information about the variants. So, it will look something like this.

So, it will start with this double hash: info equals to id equals to id number equals to number type description source and version. So, this is how the info field will look. Now, the first part is mandatory, right? So, the number type and description should be there, and the source and version parts are optional; they might not be there for the info fields. So, the ID is a unique ID.



```
##INFO=<ID=ID,Number=number,Type=type,Description="description",Source="source",Version="version">
```

ID : Unique ID

Number : Number of values described in the info field

- Usually 1,2, ….
- If the number of values is not known, then '.'
- Special characters
  a) one value per alternate allele then 'A'
  b) one value for each allele (including the reference), then 'R'
  c) one value for each possible genotype, then 'G'

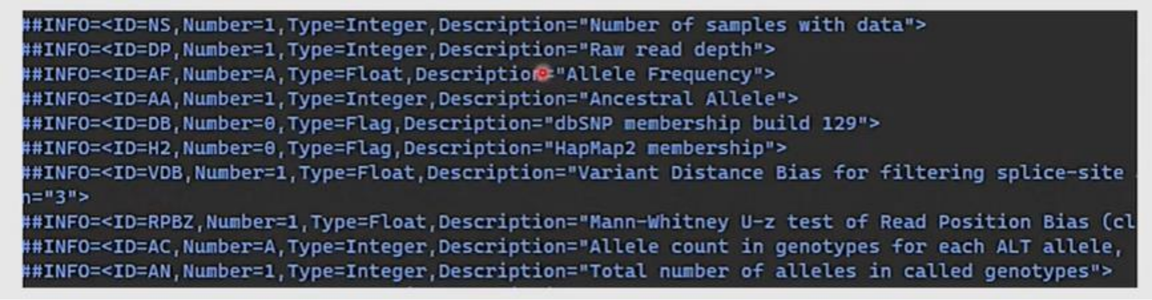
So, we will see some of these examples of what this ID would look like, and then you have the number, which describes the number of values for each ID that will be present in the info field. So, for each variant, whether you have one value or two values, etcetera, this will be given here, ok? So, again, this helps us understand how to parse this file. So, this usually this number is 1 2 right. So, 1 means we will expect 1 value for each variant. 2 means 2 values for each variant, etcetera, but in some cases the value may not be known, so the number of values that will be there may not be known.

So, we have this dot sign. In addition, sometimes we can have special characters instead of numbers or dots. So, these special situations are here, right? So, in some cases, we might have one value per alternate allele, and then we will use the letter a for this number. So, you will see that this number equals the capital A. So, then, it means we will see one value per alternate allele.

Now, this may vary right because if you have three alternate alleles, you would expect three numbers; if you have two alternate alleles, then you will see two numbers. So, that is how a is actually more relevant than specifying the exact number because we do not know the number of alleles for each variant. So, the number of alleles can vary for each variant. You can have a second situation where one value for each allele, including the reference, will be given, and then we will use the letter r for a number equal to r. You can also have one value for each possible genotype, and then we will also use the letter G. So, these are the possibilities that we have for the number field.

What about then? So, type actually describes the type of value that we can expect. So, this could be an integer, right? So, in the case of 1 2 3 right, this will be an integer value, or it could be a string, or it could be a float. Right, if you are talking about frequencies, right, fractions, right? So, that will be float, or it could be flag, ok? So, we will see some examples, and then it will be clear right away what we mean by these different types, and then finally, you will have a description right away that will tell us what this ID specifies and what this ID means.

So, it helps us interpret the data. So, here is an example, ok? So, we can see this for example, the first line info ID equals to n, and then you have a number equal to 1. So, we expect one value for this ID, then you have type integer; this should be an integer value, and this description means number of samples with data. So, this is something how this format will be defined that this info will be defined ok.

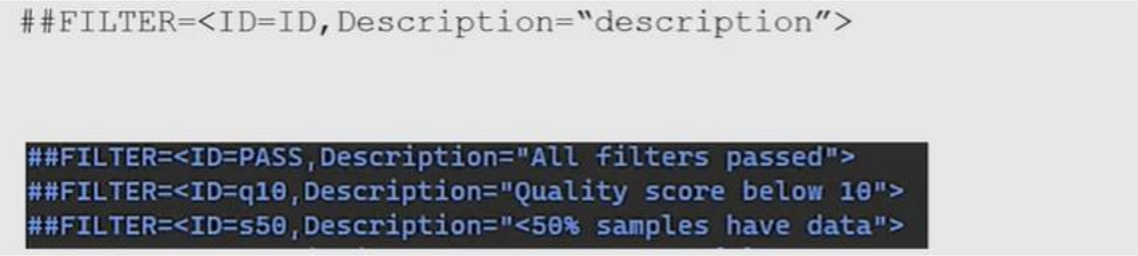


```
##INFO=<ID=NS,Number=1,Type=Integer,Description="Number of samples with data">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Raw read depth">
##INFO=<ID=AF,Number=A,Type=Float,Description="Allele Frequency">
##INFO=<ID=AA,Number=1,Type=Integer,Description="Ancestral Allele">
##INFO=<ID=DB,Number=0,Type=Flag,Description="dbSNP membership build 129">
##INFO=<ID=H2,Number=0,Type=Flag,Description="HapMap2 membership">
##INFO=<ID=VDB,Number=1,Type=Float,Description="Variant Distance Bias for filtering splice-site
n="3">
##INFO=<ID=RPBZ,Number=1,Type=Float,Description="Mann-Whitney U-z test of Read Position Bias (cl
##INFO=<ID=AC,Number=A,Type=Integer,Description="Allele count in genotypes for each ALT allele,
##INFO=<ID=AN,Number=1,Type=Integer,Description="Total number of alleles in called genotypes">
```

Next, you have to let us say, for example, that id equals to dp, right number equals to 1, type equals to integer right, and then the description is drawn in depth. Here is one example: as you can see, this number equals to A in capital A. So, this is one value per allele, right? So, the type that will be given will be a float, and this description is allele frequency, right? So, as you just mentioned, if you are specifying frequencies, then we probably use this type of float value.

So, this is the type that would be float. You also see this type of flag right, and then number is 0 right because you do not expect a number, but it is only a flag that will be used, and here the description is some membership in a certain database, whether this variant is present in this database or not. If this flag is present, the variant is present in the database; if it is not, the variant is not there. So, this is how the flag will be used, ok, and similarly, you can see that a lot of these values and a lot of these ids can be there, and again, you now understand how they will be specified, ok, and these ids will be there in the info field of the data lines. So, we have seen the data lines, right? So, you can now connect this information to the data lines and see what these values actually mean for each variant.

Then you have the filter field, as you have mentioned right? So, this shows how the data is being filtered, and this is how it would look: filter equals to ID equals to ID, and some description is okay. So, what you have here are some examples from the files, right from an actual VCF file. So, you can have a filter id equal to the pass description; all filters pass. So, we have just mentioned this.



```
##FILTER=<ID=ID,Description=“description”>
```

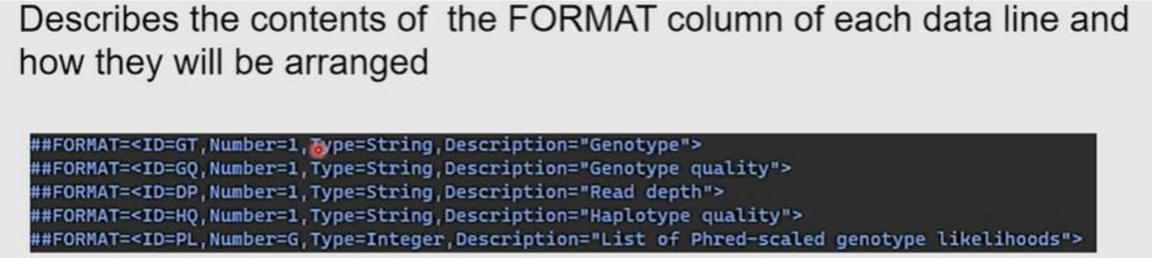
```
##FILTER=<ID=PASS,Description="All filters passed">
##FILTER=<ID=q10,Description="Quality score below 10">
##FILTER=<ID=s50,Description="<50% samples have data">
```

So, if the variant passes all filters, then you will just use this pass in the filter field, but then we also describe this id as equal to Q 10 right quality scored below 10 right. So, this is actually given in the description. You can see a quality score below 10, so we use this Q-10. So, if a variant does not pass this filter, we will just mention Q 10, right? So, we had one example where the quality score was below 10. So, we use the filter Q10.

Similarly, we have this filter ID 50. So, this description is 50 percent less than 50 percent of the samples that have data. Again, you can set these different filters, and for different types of analysis, etcetera, you can have different types of filters set, and all those will be described in the meta-information lines. So, as you can see, this vcf file can vary from one analysis to another depending on the types of filters and the information that is given. So, that is why you need to carefully look into the meta-information line to interpret the data lines.

So, to understand more about the variants ok. So, then you have the format field, as we have seen right here, and this format field in the meta information line describes the contents of the format column of the data lines. So, for each variant it will mention ok what are the ids that are there for format and how the data will be arranged right how they will be arranged this is also given. Again, an example from a vcf file. You see this again: it starts with a double hash. You have format id equal to gt number equals to 1 type equals to string description genotype. Again, we are following the same process. If you have some ID that is given, then you have the number, whether you can expect a number or it can be something you can see.

So, each individual value for a genotype is right. So, you will have this, and then type is an integer description list of red scale genotype likelihoods, ok? So, again, this will describe what you are, what you are, and what you are describing with these IDs. So, this description is very important to understand the data. So, then you also have an alternative allele field, right?

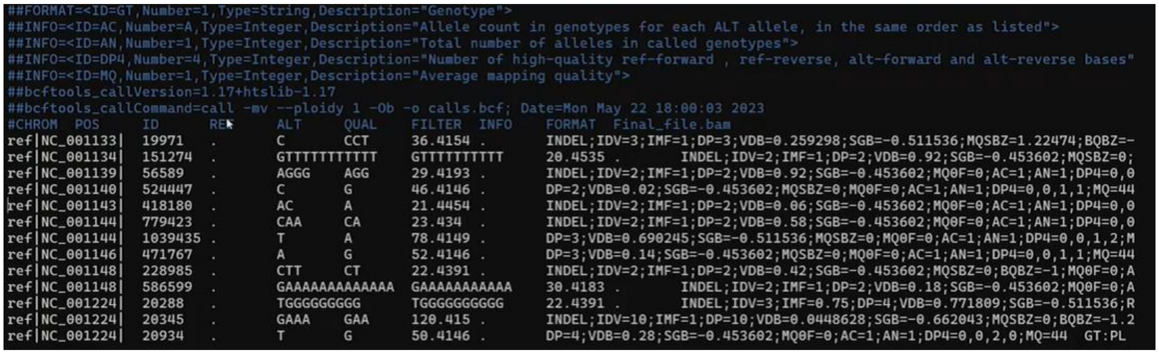


Describes the contents of the FORMAT column of each data line and how they will be arranged

```
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=String,Description="Genotype quality">
##FORMAT=<ID=DP,Number=1,Type=String,Description="Read depth">
##FORMAT=<ID=HQ,Number=1,Type=String,Description="Haplotype quality">
##FORMAT=<ID=PL,Number=G,Type=Integer,Description="List of Phred-scaled genotype likelihoods">
```

So, this describes alternative alleles for structural variants; this is especially used. So, alt id equals to type and description. Now, for svs, the type, this id will indicate the type of structural variant that you are looking at, and this is of any colon-separated list, but the first type must be one of the following: del for deletion, ins for insertion, dup for duplication, inv for inversion, and cnv for copy number variation. You also have different subtypes and can define more complex SVs, but we are not going into detail; there are specific ways to do that with the vcf file. So, you can see that a VCF file is a very information-rich file, and you have to understand what you are actually representing or what is present in these files.

So, we need to go through all of this information. So, coming back to the data lines, now we have understood the meta-information lines. So, we can now look at these data lines, and we can see what this actually looks like. So, you have this chromosome position right? So, again, not chromosome number, some ID position, then you have the ID right there, some number.



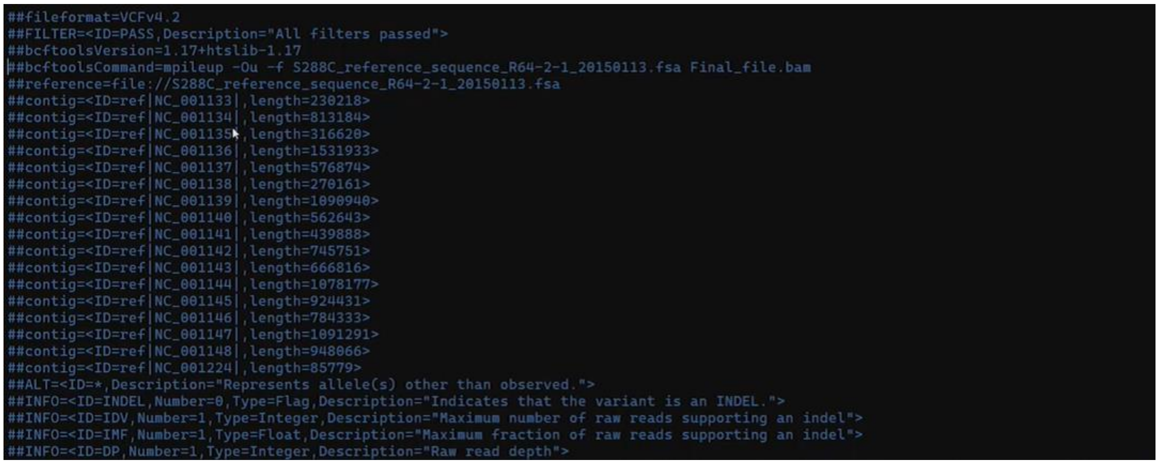
```
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##INFO=<ID=AC,Number=A,Type=Integer,Description="Allele count in genotypes for each ALT allele, in the same order as listed">
##INFO=<ID=AN,Number=1,Type=Integer,Description="Total number of alleles in called genotypes">
##INFO=<ID=DP4,Number=4,Type=Integer,Description="Number of high-quality ref-forward , ref-reverse, alt-forward and alt-reverse bases"
##INFO=<ID=MQ,Number=1,Type=Integer,Description="Average mapping quality">
##bcftools_callVersion=1.17+htslib-1.17
##bcftools_callCommand=call -mv --ploidy 1 -Ob -o calls.bcf; Date=Mon May 22 18:00:03 2023
#CHROM  POS      ID      REF     ALT     QUAL    FILTER  INFO    FORMAT  Final_file.bam
ref|NC_001133|  19971   .       C       CCT     36.4154 .       INDEL;IDV=3;IMF=1;DP=3;VDB=0.259298;SGB=-0.511536;MQSBZ=1.22474;BQBZ=-
ref|NC_001134|  151274  .       GTTTTTTTTTTT    GTTTTTTTTTT     20.4535 .       INDEL;IDV=2;IMF=1;DP=2;VDB=0.92;SGB=-0.453602;MQSBZ=0;
ref|NC_001139|  56589   .       AGGG    AGG     29.4193 .       INDEL;IDV=2;IMF=1;DP=2;VDB=0.92;SGB=-0.453602;MQ0F=0;AC=1;AN=1;DP4=0,0
ref|NC_001140|  524447  .       C       G       46.4146 .       DP=2;VDB=0.02;SGB=-0.453602;MQSBZ=0;MQ0F=0;AC=1;AN=1;DP4=0,0,1,1;MQ=44
ref|NC_001143|  418180  .       AC      A       21.4454 .       INDEL;IDV=2;IMF=1;DP=2;VDB=0.06;SGB=-0.453602;MQ0F=0;AC=1;AN=1;DP4=0,0
ref|NC_001144|  779423  .       CAA     CA      23.434  .       INDEL;IDV=2;IMF=1;DP=2;VDB=0.58;SGB=-0.453602;MQ0F=0;AC=1;AN=1;DP4=0,0
ref|NC_001144|  1039435 .       T       A       78.4149 .       DP=3;VDB=0.690245;SGB=-0.511536;MQSBZ=0;MQ0F=0;AC=1;AN=1;DP4=0,0,1,2;M
ref|NC_001146|  471767  .       A       G       52.4146 .       DP=3;VDB=0.14;SGB=-0.453602;MQSBZ=0;MQ0F=0;AC=1;AN=1;DP4=0,0,1,1;MQ=44
ref|NC_001148|  228985  .       CTT     CT      22.4391 .       INDEL;IDV=2;IMF=1;DP=2;VDB=0.42;SGB=-0.453602;MQSBZ=0;BQBZ=-1;MQ0F=0;A
ref|NC_001148|  586599  .       GAAAAAAAAAAAAA  GAAAAAAAAAAA    30.4183 .       INDEL;IDV=2;IMF=1;DP=2;VDB=0.18;SGB=-0.453602;MQ0F=0;A
ref|NC_001224|  20288   .       TGGGGGGGGG      TGGGGGGGGGG     22.4391 .       INDEL;IDV=3;IMF=0.75;DP=4;VDB=0.771809;SGB=-0.511536;R
ref|NC_001224|  20345   .       GAAA    GAA     120.415 .       INDEL;IDV=10;IMF=1;DP=10;VDB=0.0448628;SGB=-0.662043;MQSBZ=0;BQBZ=-1.2
ref|NC_001224|  20934   .       T       G       50.4146 .       DP=4;VDB=0.28;SGB=-0.453602;MQ0F=0;AC=1;AN=1;DP4=0,0,2,0;MQ=44  GT:PL
```

So, this actually gives us an ID for some database, right? This variant is present in that database. Then you have the reference, alternate allele, quality, and filter, right? Again, now you better understand what this filter actually means: pass means all filters passed; q

10 means this filter was not cleared by this variant, and you can see why. Then, in the information, we now know what this means: dp means, a means, or f means right. So, all this information we have described in the meta-information line is right.

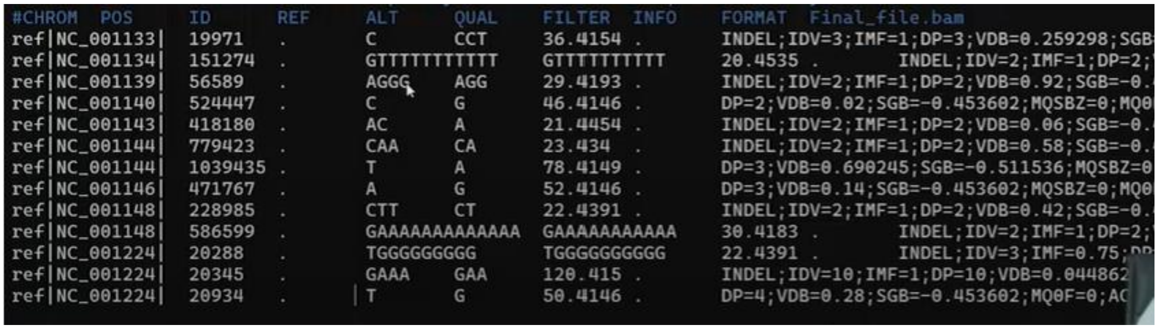
So, we can now connect this, and we can understand what it means. For example, you can see this DB means this variant is present in a certain database. So, this is a flag that is given right, and then, of course, you have a lot more such values after this, and of course, you have to then go back to the meta information and see how to interpret this data. Then you have the format; you can see certain parts here. So, GT and GQ are again defined in the format fields, and with this again, you can go back and see and connect with the meta information lines, and then you will understand what these data lines contain.

So, hopefully, this is how we actually read this file by combining the meta information line and the data lines. This is where the file is open, right? These are the data lines that you have already looked at, and this is the header line. We have this chromosome position ID, etcetera. And on top of that, if you go up now, I did not show you the meta information part, but on top of that, you have the meta information part now, ok? And the first line is file format equals to vcf v4.2. You can see that then you have the filter right id equals to pass.



```
##fileformat=VCFv4.2
##FILTER=<ID=PASS,Description="All filters passed">
##bcftoolsVersion=1.17+htslib-1.17
##bcftoolsCommand=mpileup -Ou -f S288C_reference_sequence_R64-2-1_20150113.fsa Final_file.bam
##reference=file://S288C_reference_sequence_R64-2-1_20150113.fsa
##contig=<ID=ref|NC_001133|,length=230218>
##contig=<ID=ref|NC_001134|,length=813184>
##contig=<ID=ref|NC_001135|,length=316620>
##contig=<ID=ref|NC_001136|,length=1531933>
##contig=<ID=ref|NC_001137|,length=576874>
##contig=<ID=ref|NC_001138|,length=270161>
##contig=<ID=ref|NC_001139|,length=1090940>
##contig=<ID=ref|NC_001140|,length=562643>
##contig=<ID=ref|NC_001141|,length=439888>
##contig=<ID=ref|NC_001142|,length=745751>
##contig=<ID=ref|NC_001143|,length=666816>
##contig=<ID=ref|NC_001144|,length=1078177>
##contig=<ID=ref|NC_001145|,length=924431>
##contig=<ID=ref|NC_001146|,length=784333>
##contig=<ID=ref|NC_001147|,length=1091291>
##contig=<ID=ref|NC_001148|,length=948066>
##contig=<ID=ref|NC_001224|,length=85779>
##ALT=<ID=*,Description="Represents allele(s) other than observed.">
##INFO=<ID=INDEL,Number=0,Type=Flag,Description="Indicates that the variant is an INDEL.">
##INFO=<ID=IDV,Number=1,Type=Integer,Description="Maximum number of raw reads supporting an indel">
##INFO=<ID=IMF,Number=1,Type=Float,Description="Maximum fraction of raw reads supporting an indel">
##INFO=<ID=DP,Number=1,Type=Integer,Description="Raw read depth">
```

So, this will mean all filters passed, and then you have the VCF Tools version. So, this is the program that generated it; it also mentions the command that was used to generate this file, the reference file name that we used right here, as I mentioned the contig names or the ids of the chromosomes. So, these are also given, and the lengths are also given. Then you have the alternate alleles, right? So, these are the alternate alleles that are present, then you have the info part right here, you have this info part here, and here you have id equals to indel number equals to 0 type is flag whether it is an indel or not.

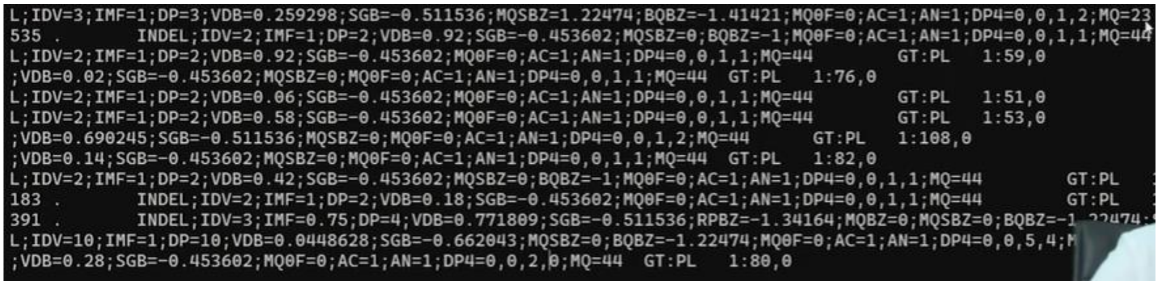


```
#CHROM  POS       ID       REF       ALT             QUAL          FILTER  INFO      FORMAT  Final_file.bam
ref|NC_001133|  19971    .         C           CCT           36.4154 .        INDEL;IDV=3;IMF=1;DP=3;VDB=0.259298;SGB
ref|NC_001134|  151274   .         GTTTTTTTTTTT   GTTTTTTTTTT    20.4535 .        INDEL;IDV=2;IMF=1;DP=2;
ref|NC_001139|  56589    .         AGGG        AGG           29.4193 .        INDEL;IDV=2;IMF=1;DP=2;VDB=0.92;SGB=-0.
ref|NC_001140|  524447   .         C           G             46.4146 .        DP=2;VDB=0.02;SGB=-0.453602;MQSBZ=0;MQ0
ref|NC_001143|  418180   .         AC          A             21.4454 .        INDEL;IDV=2;IMF=1;DP=2;VDB=0.06;SGB=-0.
ref|NC_001144|  779423   .         CAA         CA            23.434  .        INDEL;IDV=2;IMF=1;DP=2;VDB=0.58;SGB=-0.
ref|NC_001144|  1039435  .         T           A             78.4149 .        DP=3;VDB=0.690245;SGB=-0.511536;MQSBZ=0
ref|NC_001146|  471767   .         A           G             52.4146 .        DP=3;VDB=0.14;SGB=-0.453602;MQSBZ=0;MQ0
ref|NC_001148|  228985   .         CTT         CT            22.4391 .        INDEL;IDV=2;IMF=1;DP=2;VDB=0.42;SGB=-0.
ref|NC_001148|  586599   .         GAAAAAAAAAAAAA  GAAAAAAAAAAA   30.4183 .        INDEL;IDV=2;IMF=1;DP=2;
ref|NC_001224|  20288    .         TGGGGGGGGG    TGGGGGGGGGG    22.4391 .        INDEL;IDV=3;IMF=0.75;
ref|NC_001224|  20345    .         GAAA        GAA           120.415 .        INDEL;IDV=10;IMF=1;DP=10;VDB=0.044862
ref|NC_001224|  20934    .         T           G             50.4146 .        DP=4;VDB=0.28;SGB=-0.453602;MQ0F=0;AC
```

So, there is not a this is not a number it is a flag right. So, if it is an indel, there will be a flag indel flag right. You will just mention indel as you have seen right, as you can probably see in some of the data lines visible here that you can see this flag indel right. If this flag is present, it means the variant is an indel, but if this flag is not present, then the variant is not an indel and is a SNP. As you can see here, this is an SNP. So, you have a lot of these now defined, right? We took some examples, but as you can see in our file, there are a lot more info fields defined for you. So, for example, idv is the maximum number of rods supporting an indel, right? How many of the reads actually show this indel in the data right?

You have have IMF maximum fraction of rod, which is that you are supporting an indel right; this is converting to a fraction of the data; then you have this DP, which is the rod in depth; then you have a lot of other measures as well. So, you can have some sort of test for base quality bias, mapping quality bias, position bias, etcetera. So, what they are actually looking at is whether the variant that has been called has been generated because of some biases in our data. There might be a lot of biases.

So, there could be some errors in calling. So, this is what is being tested by this test, ok? So, that is what these different IDs are mentioning, and you will also mention which one is better and which one is actually good for you, and what you can do is use this information to set up filters. If you are interested, you can say we do not want this kind of bias in our data in the final variant call. So, that is why we will set out certain filters based on these biases that have been defined in this file. So, this is the info part, then you have the format part, and there are only two formats, right?



```
L;IDV=3;IMF=1;DP=3;VDB=0.259298;SGB=-0.511536;MQSBZ=1.22474;BQBZ=-1.41421;MQ0F=0;AC=1;AN=1;DP4=0,0,1,2;MQ=23
535 .       INDEL;IDV=2;IMF=1;DP=2;VDB=0.92;SGB=-0.453602;MQSBZ=0;BQBZ=-1;MQ0F=0;AC=1;AN=1;DP4=0,0,1,1;MQ=44
L;IDV=2;IMF=1;DP=2;VDB=0.92;SGB=-0.453602;MQ0F=0;AC=1;AN=1;DP4=0,0,1,1;MQ=44        GT:PL   1:59,0
;VDB=0.02;SGB=-0.453602;MQSBZ=0;MQ0F=0;AC=1;AN=1;DP4=0,0,1,1;MQ=44  GT:PL   1:76,0
L;IDV=2;IMF=1;DP=2;VDB=0.06;SGB=-0.453602;MQ0F=0;AC=1;AN=1;DP4=0,0,1,1;MQ=44        GT:PL   1:51,0
L;IDV=2;IMF=1;DP=2;VDB=0.58;SGB=-0.453602;MQ0F=0;AC=1;AN=1;DP4=0,0,1,1;MQ=44        GT:PL   1:53,0
;VDB=0.690245;SGB=-0.511536;MQSBZ=0;MQ0F=0;AC=1;AN=1;DP4=0,0,1,2;MQ=44      GT:PL   1:108,0
;VDB=0.14;SGB=-0.453602;MQSBZ=0;MQ0F=0;AC=1;AN=1;DP4=0,0,1,1;MQ=44  GT:PL   1:82,0
L;IDV=2;IMF=1;DP=2;VDB=0.42;SGB=-0.453602;MQSBZ=0;BQBZ=-1;MQ0F=0;AC=1;AN=1;DP4=0,0,1,1;MQ=44        GT:PL
183 .       INDEL;IDV=2;IMF=1;DP=2;VDB=0.18;SGB=-0.453602;MQ0F=0;AC=1;AN=1;DP4=0,0,1,1;MQ=44        GT:PL
391 .       INDEL;IDV=3;IMF=0.75;DP=4;VDB=0.771809;SGB=-0.511536;RPBZ=-1.34164;MQBZ=0;MQSBZ=0;BQBZ=-1.22474;
L;IDV=10;IMF=1;DP=10;VDB=0.0448628;SGB=-0.662043;MQSBZ=0;BQBZ=-1.22474;MQ0F=0;AC=1;AN=1;DP4=0,0,5,4;M
;VDB=0.28;SGB=-0.453602;MQ0F=0;AC=1;AN=1;DP4=0,0,2,0;MQ=44  GT:PL   1:80,0
```

So, you can see this. So, one is the list of weight scale genotype likelihoods, and then you have this genotype only as GT, ok? So, here, you will not have this haplotype quality because we are dealing with a haploid genome. So, it is not a diploid genome. So, you do not have to do phasing, etcetera, alright? So, then you also have some other ones, right? You can see some other info fields you have: ID equals to AC, ID equals to AM, right? Again, there would be descriptions saying what this is doing, and again, according to that, you have, for example, number A.

So, you have a value for each alternate allele, right? So, this is something that is mentioned again here, and then for the other ones, you can see the numbers. So, 1/4. So, for every variant, you will have 4 values or 1 value, etcetera. So, this is defined here, and then finally, it ends with the vcf2 call version again and the final call command that was used for vcf2s.

Now, let us look at this right side, ok? So, this is the first part of the data lines that we have discussed. So, here is the ID; we do not have any mapping tools or any specific database. So, the ID field is empty, right? We have not set out filters right except for this q 20. So, the filter field is also empty, and the info here is populated with all these numbers, as you can see, and then you have the format, and then you have only one file here. So, this file name, the sample name, is given as the final file name.

So, if you have multiple ones, those will be listed one after another, ok? So, let us see if you can see this right for this. So, first, you have these information fields, and for each of them, you have these values given; you can see them here. So, these values are given, and again, this follows the meta-information line. So, the meta information line says this SGB stands for what it stands for, and it should have one value, right?

So, that should be floating or something, and that will be followed here, ok? And similarly, after that, you have this format part. So, this format part specifies how this would be defined for the samples. So, this is the order, and again, this has been defined in the meta information line, and the information is for this final file, dot bam. So, imagine if we had multiple ones. Let us say final file 1 dot bam, final file 2 dot bam, etcetera. Those will be repeated one after another, and those will be present in the same format one after another.

So, now, I hope it is clear right how you can combine this meta information line and then that with the data line to understand the full VCF file and what is actually represented in this file. So, this is done. So, the next step is to actually go for variant annotation. So, we will go back to the presentation, and we will talk about variant annotation. So, we take the vcf files, and we have seen this meta information line in the data lines, and we can set up certain filtering schemes.

If we say we do not want certain biases in our data in our final calls, then we can set up certain filtering based on these fields that we have just described. There are many fields, and maybe some of them could be useful for certain analyses. Once we have done this filtering, we can do something called variant annotation. So, what is this variant annotation? We will discuss this in detail now, and why is this useful? So, for any variant call, you have to interpret right what this variant actually means and whether there is any significance of this variant in your data. So, what we are doing by annotation is measuring the likely impact of a variant on function, maybe gene function, cellular function, or cellular phenotype.

So, we can probably map it to certain genes, etcetera, or we can or are doing known associations with phenotypes, disorders, or diseases. So, whether this variant has been seen in other individuals and has been associated with certain diseases, for example, genetic diseases, is also important. So, this kind of annotation is something we can do. So, there could be many possibilities. So, a single nucleotide polymorphism of the variant could be in the promoter region of a gene, and it can thus regulate gene expression.

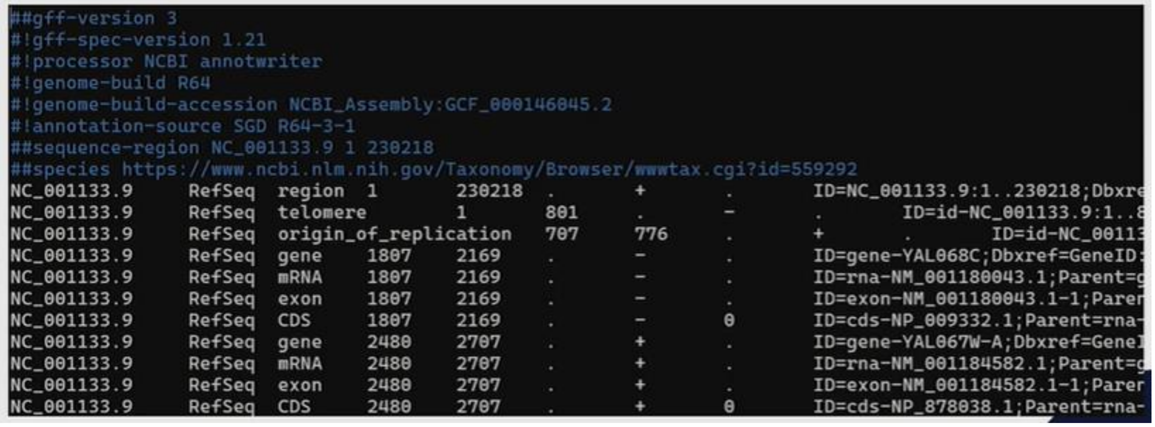
So, you can say it can affect the binding of transcription factors and it can impact the expression of the downstream gene, or the same variant can be synonymous, non-synonymous, or nonsense. So, synonymous means there are changes in the coding DNA sequence but no change in the amino acid sequence. It could be non-symbolic, which means there are changes in the coding DNA sequence leading to changes in the amino acid sequence. So, these variants are likely to have functional impacts, and on top of that, you can have nonsense mutations, too. So, this nonsense mutation refers to the introduction of a premature stop codon in the gene, and this can lead to a loss of function.

So, you can think of the many different other types of annotations, right? We have just discussed two types of annotations that can have regulatory functions. So, a variant that we have found might regulate or might actually change the expression pattern of a gene. So, it could be very important there or it could be changing the function of a gene, right? It can lead to loss of function or maybe change the function in certain ways, but you can also have other types of variants. For example, it can affect, let's say, the folding of a protein. You can think of splicing events as well.

You can have variants that change splicing events for a gene; all those things can be there. So, these will be done through this kind of annotation, ok? Now, how do you do this annotation? Ok. So, for doing this annotation, we need something called GFF or GTF files, ok?

So, GFF and GTF files So, GFF stands for general feature format, and GTF stands for general transfer format which is actually the same as GFF version 2. So, we will discuss only the GFF format, and that should be enough for doing this variant annotation, ok? So, let us talk about the GFF file, what it contains, and what kind of information you have. It is actually quite simple compared to other ones that we have discussed, the VCF or the SAM, which are much more elaborate. So, GFF is much easier to interpret.

So, here are some examples you can see. So, here you have on top you again have some headers again you have this hash or double hash right and it tells us like which GFF file we are looking at which is the species right the genome build where what is the GFF version etcetera right and where did we get this data from what is the assembly version of this genome etcetera So, here we have this information, OK, and then you can see these numbers and IDs on the left and right. So, these N-C numbers are the IDs of the chromosomes, then you have the source, then you have some other information, ok? So, we will discuss what these fields are actually in a moment, ok? So, each line is a feature in the genome, and this is actually represented by nine fields.



```
##gff-version 3
#!gff-spec-version 1.21
#!processor NCBI annotwriter
#!genome-build R64
#!genome-build-accession NCBI_Assembly:GCF_000146045.2
#!annotation-source SGD R64-3-1
##sequence-region NC_001133.9 1 230218
##species https://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi?id=559292
NC_001133.9     RefSeq  region  1       230218  .       +       .       ID=NC_001133.9:1..230218;Dbxre
NC_001133.9     RefSeq  telomere        1       801     .       -       .       ID=id-NC_001133.9:1..8
NC_001133.9     RefSeq  origin_of_replication   707     776     .       +       .       ID=id-NC_0011
NC_001133.9     RefSeq  gene    1807    2169    .       -       .       ID=gene-YAL068C;Dbxref=GeneID:
NC_001133.9     RefSeq  mRNA    1807    2169    .       -       .       ID=rna-NM_001180043.1;Parent=g
NC_001133.9     RefSeq  exon    1807    2169    .       -       .       ID=exon-NM_001180043.1-1;Parer
NC_001133.9     RefSeq  CDS     1807    2169    .       -       0       ID=cds-NP_009332.1;Parent=rna-
NC_001133.9     RefSeq  gene    2480    2707    .       +       .       ID=gene-YAL067W-A;Dbxref=GeneI
NC_001133.9     RefSeq  mRNA    2480    2707    .       +       .       ID=rna-NM_001184582.1;Parent=g
NC_001133.9     RefSeq  exon    2480    2707    .       +       .       ID=exon-NM_001184582.1-1;Parer
NC_001133.9     RefSeq  CDS     2480    2707    .       +       0       ID=cds-NP_878038.1;Parent=rna-
```

So, these nine fields are tab-separated, and the empty fields are denoted by a dot, as we have seen for other files as well. So, if you do not have information, just put this dot. So, what are these file fields? So, if you want to annotate your variance using this GFF filing, you can probably write a code that will do this for you. Some of the very simple annotations can be done by yourself, ok? So, what are these file fields that you have? So, the first one is called the sequence name, which is usually the name of the chromosome, the scaffold, or a valid identifier.

So, in our case, the data in the example that I have shown you is an identifier from a database, right? So, this is again pointing to a chromosome, right? So, instead of that identifier, you can directly name the chromosome there. Then the second field is a source, right? So, this is the name of the program that generated that feature, or the source database, right?

So, in this case, as you can see in the example, it is a RefSeq database from which we obtain the GFF file. The third one, which is very important, is the feature itself. So, what is this feature that we are actually talking about? Is it a gene, is it an exon, or is it a variation? Right, what type of feature are we looking at? So, we will look at the example again and probably understand it better. The fourth is the start position ok, where does this feature start in the genome, and this follows the one base positioning system ok.

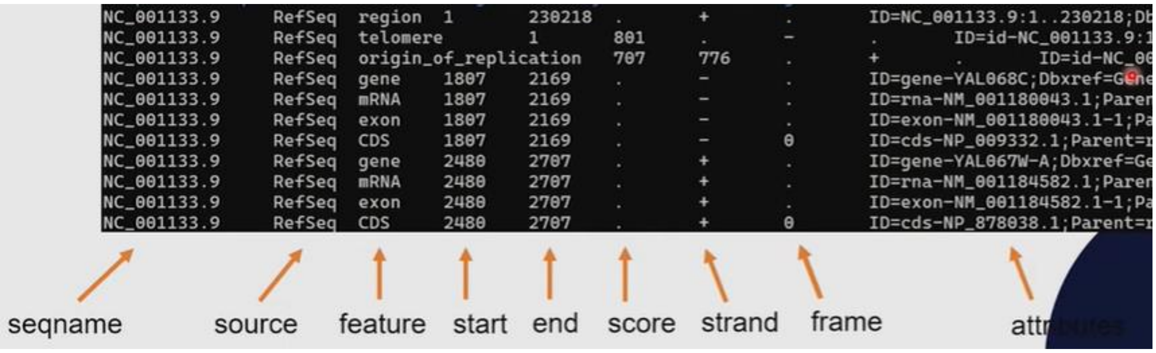
So, we talked about the one-based and zero-based positioning systems. GFF follows the one-base rule. So, the starting base is number one. Then you have the fifth field, which is the end position where the feature ends right. So, if it is a gene, what is the location, whether it is from 100 to 150, 200 to 700, etcetera? So, this is what this start and end position will give you, right? Again, this follows the one-base positioning system.

The next one is score-right. So, this is a value that indicates how confident the source is about the annotation. So, whether it is a gene or an exon, this confidence will be reflected in the score in the sixth field. So, if the value is not available there is no confidence value; it is just represented by a dot. Then you have the final one, right?

So, it is the strand field, which is defined as the plus or minus. So, you probably understand whether the feature is present in the plus strand or the negative strand. The eighth one is the frame, okay? So, this is actually important if your feature is present in the coding sequence, right? So, it could be 0, 1, or 2.

So, 0 means that the first best of the features is the first base of a codon. So, I think you can probably now relate whether this is likely a synonymous mutation or a non-synonymous one, etcetera. Those will be those you can find out if you know this value, right? So, 0 means the first base of the feature is the first base of a codon, 1 means the second base of the feature is the first base of a codon, and so on. And the final field is the attribute OK. So, this is a semicolon-separated list of tag-value pairs; it contains additional information about each feature.

So, it will have some kind of description, ok? So, let us go back to this example again, right? So, here is your identifier. This is the chromosome ID, then you have the source, then you have the feature name, whether it is a gene, mRNA, exon, or coding sequence CDS. This is given here, or in here, you can also see the telomere. So, where is the Telomere region? So, the positions are given, and then you have the start and end positions, right? So, for this gene, of course, the gene mRNA and exon follow the same start and end because there is no multi-exon gene.



```
NC_001133.9    RefSeq  region  1       230218  .       +       .       ID=NC_001133.9:1..230218;Db
NC_001133.9    RefSeq  telomere        1       801     .       -       .       ID=id-NC_001133.9:1
NC_001133.9    RefSeq  origin_of_replication   707     776     .       +       .       ID=id-NC_00
NC_001133.9    RefSeq  gene    1807    2169    .       -       .       ID=gene-YAL068C;Dbxref=Gene
NC_001133.9    RefSeq  mRNA    1807    2169    .       -       .       ID=rna-NM_001180043.1;Paren
NC_001133.9    RefSeq  exon    1807    2169    .       -       .       ID=exon-NM_001180043.1-1;Pa
NC_001133.9    RefSeq  CDS     1807    2169    .       -       0       ID=cds-NP_009332.1;Parent=r
NC_001133.9    RefSeq  gene    2480    2707    .       +       .       ID=gene-YAL067W-A;Dbxref=Ge
NC_001133.9    RefSeq  mRNA    2480    2707    .       +       .       ID=rna-NM_001184582.1;Paren
NC_001133.9    RefSeq  exon    2480    2707    .       +       .       ID=exon-NM_001184582.1-1;Pa
NC_001133.9    RefSeq  CDS     2480    2707    .       +       0       ID=cds-NP_878038.1;Parent=r
```

So, this is not a multi-exon gene, right? So, then you have the dot. So, this is okay. So, start and end, and then you have the score right; this is confidence if we remember correctly. Then you have the strand ok, this is the strand plus-minus and then you have this feature, whether it is 0 1 2 right, whether it starts with the codon right, etcetera, right, that information is given. So, it is again not given for many of these features, but for some, such as the coding sequence CDS, this feature is given right 0 and this is the starting position of the codon right. Then you have some additional attributes, as you can see again some extra information.

So, here is the gene name. So, this gene name is given, or in the case of CDS, this is the CDS ID again. This is the identifier that is given, and you can also see a lot more information. So, if you go to the original GFF file, you will find a lot more information about these things. So, just to again highlight, this is the sequence name, this is the source, this is the feature start-end score, this is the strand and this is the frame right (0, 1, 2 and then you have the attributes here, ok? Now, the second type of notation that you can do is associate these variants with phenotypes, disorders, or diseases.

So, how we do that this is very simple actually. So, people have collected or researchers have collected this data over many years from many patients, etcetera, from many studies, and they have put it into the databases. And what we need to do is simply analyze whether the variant that we have found is present in these databases of variants. And this is very important, especially for human data. If you are working with human variants or human samples, you want to check this. And there are several databases; I will just mention some of them. Here is one, which is a very common one.

So, it is called DBSNP, right? So, it contains a lot of these human signal nuclear variants, microsatellite information, etcetera, and small indels. So, you can check if you have this variant data in your case, whether this is already reported in DBSNP, whether it is a common variant in humans, or whether there is some clinical association. So, this information will also be present, okay? Then you have something called the OMIM database, which is the online Mendelian inheritance in men.

So, here again, this is an online catalog of human genes and genetic disorders. So, again, looking at some of the variants that have been associated with genetic disorders or diseases, you can, and you also have more recent databases coming up. So, for example, this is one database where you have a lot of this comprehensive variant variation annotation. So, it is not just this clinical annotation, etcetera; it also contains information about regulatory information, such as whether this variant is associated with certain gene regulation changes or whether there is some sort of change in transcription factor binding. All those information can now be combined together and put into the database.

So, just a few words about the variant annotation tools, ok? So, here again are some common ones; of course, there are many more out there. The first one is called ANOVAIR. So, you can go and check in at the link. So, you can have a gene-based annotation or region-based annotation, looking at whether it affects certain variant effects at the function of certain genes or whether they are present in certain regions, which might change gene regulation, etcetera.

You can also have another tool called VarAFT. So, this is a variant annotation and filtration tool. So, this actually helps with annotation using DBSNP or OMIM if you are working with human data. It also associates your variants with gene ontology and certain biological pathways. We will talk about gene ontology in later classes in much more detail and see what this actually contains. So, it mostly looks at the functions, etcetera, or involvement of genes in biological processes.

So, these kinds of tools are now available, which you can utilize for your variant annotation. So, here are the references that we have utilized for this class. So, just to summarize, we have talked about VCF files and the meta information lines, and it kind of gives us a complete picture of the data lines. What kind of information are we getting from the data lines? Then we talked about the variant annotation part. So, we are looking at the possible impact of variants or gene functions and phenotypes, as well as trying to associate them with disorders and diseases if they are already known.

And we talked about some of the variant annotation databases that can be used and will be useful if you are especially working with human data, and we also talked about some variant annotation tools that can be utilized. Thank you.