Next Generation Sequencing Technologies: Data Analysis and Applications

Data QC and Trimming

Dr. Riddhiman Dhar, Department of Biotechnology

Indian Institute of Technology Kharagpur

Good day, everyone. Welcome to the course on Next Generation Sequencing Technologies, Data Analysis, and Applications. In this class, we will be talking about data QC and data streaming. Just to briefly recap, we have discussed data formats, followed by data quality control, or data QC. So, in the last few classes, we have talked about different aspects of data QC. We have talked about one tool that allows us to do this quality control.

So, we will continue that discussion today. We will complete all the quality control measurements that we have followed, and then we will talk about something called data streaming. So, this is a part of the pre-processing step and this is a very important step before we actually go into the actual analysis. So, these will be the keywords we will be seeing today: sequence-based control, quality streaming, and contamination.

So, this is the tool that we discussed in the last class. So, the tool is FASTQC. So, this is a free tool that we use to evaluate the next-generation sequencing data. So, this tool allows us to check quality only for FASTQ data sets, right? So, for example, Illumina data sets, or some other data sets that can be converted to FASTQ format, ok?

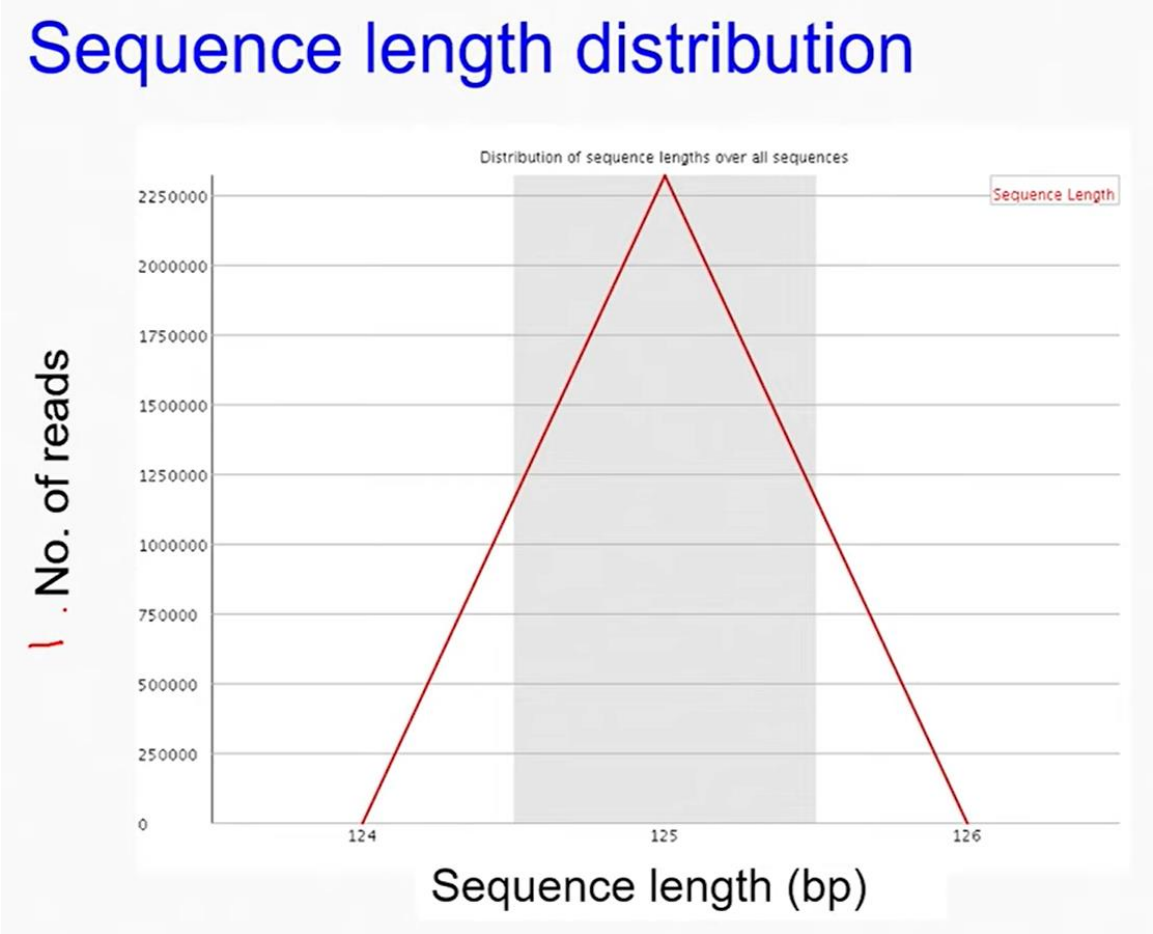
So, some of the controls we have talked about already are based on quality scores, right? So, this is something that we discussed in the last class. We talked about, for example, per read quality or per base quality, right, and tile quality—all those things we have discussed, and they rely on this quality score, right? So, this quality score comes up to the machine; it measures the signal-to-noise ratio and gives a measure of confidence in the base call. So, there is another aspect of quality control that we will be talking about today, and this is sequence-based quality control.

So, compared to Q-score-based control, here the FASTQC tool actually looks into the

sequence of the data, right? So, whether the data sequence actually makes sense So, maybe the quality score is very good, but nevertheless, you can have some sort of contaminant or some sort of DNA molecules that are not supposed to be there. So, they will not show up in the Q score-based control; they will appear only when you do the sequence-based QC, right? So, let us get started on that part.

Let us talk about sequence-based QC in much more detail. So, one of the first parameters that we check is called sequence length distribution, right? So, this is something that is applicable to all platforms that you work with. Here is an example with Illumina data, right? So, in the last class, we are seeing this example with Illumina data.

So, in this example, we are again seeing this Illumina data. Along the x axis, you have the sequence length, and along the y axis, you have the number of reads. So, what does it show? It shows the distribution or the number of reads that have a certain sequence length, ok, or read length. So, here, because this is Illumina data, all the reads are of the same length, right? So, which is the 125-base pair, right?



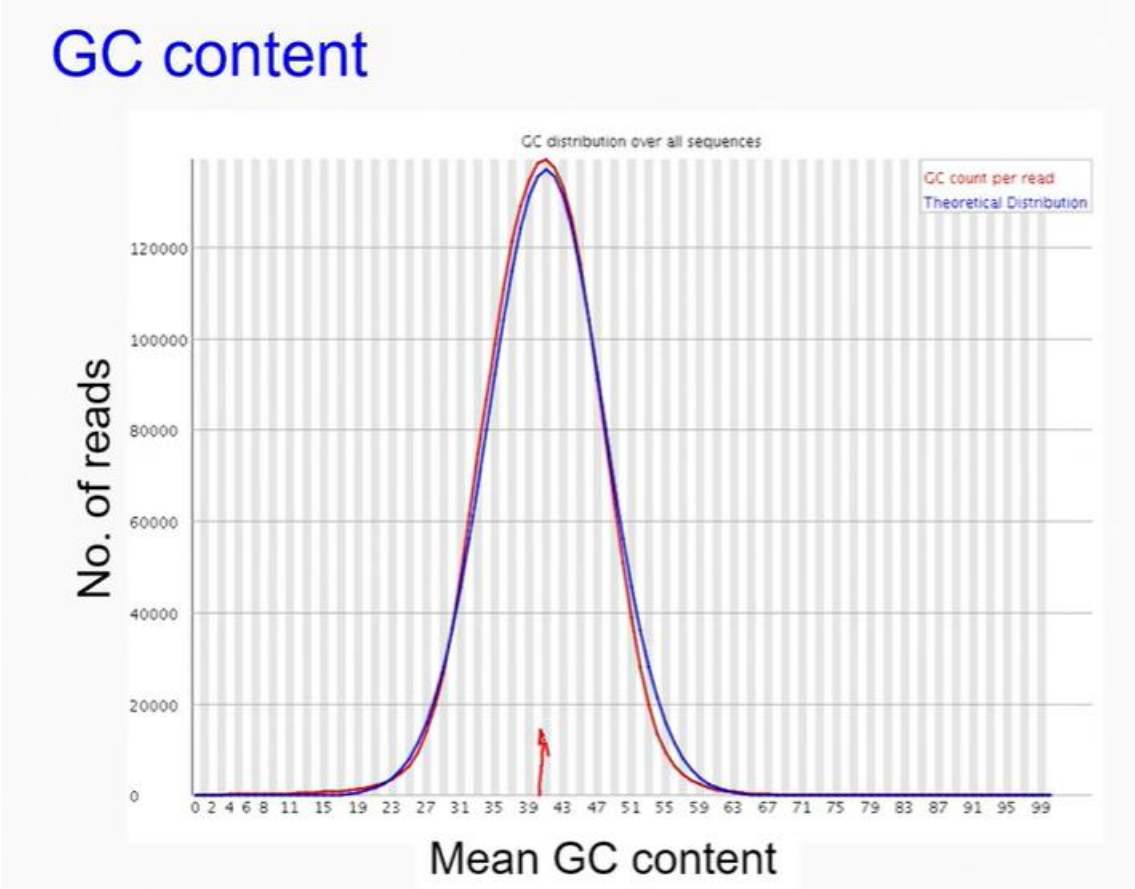
So, if you remember the principle Illumina will give you reads of the same length. So, that is what we see here: we have 125 base pairs. For other platforms, this will vary; you will not see this kind of distribution where all the pieces are of the same length; you can see different distributions of read length, right? So, what is the purpose of this? It is like checking that we have good-read data, right? We have at least a certain length of reads, right?

The second measure that we will talk about is something called par-based content, ok? So, what does this measure? So n means this is an ambiguous base. The sequencer was not able to determine which base it was, right? So, it does not know whether it is A, G, C, or T. It knows that there is a base, but it is not able to call the base properly because the signal-to-noise ratio is not very good.

So, this is something that we want to minimize in our data, right? We do not want many n values in the sequence data, right? So, this is something we will see, right? So, what you have in the x axis, you have the position in read, right? This is in base pairs, and along the y axis, you have the percentage of n, right?

And in these data, what you see is a line hugging close to 0, right? So, that is a very good sign, which means we have very few N bases in the data, right? And this is a very good sign that the sequencing was generally good. You have very good data because there is not much N in your data, ok? Also, remember that this will also appear in the quality score-based QC, right?

So, you have because any position of N will have a low-quality score because the sequencer is not able to resolve this base properly. And this is something that will appear in both, right? If you are looking at sequence-based quality control or Q-score-based quality control, this will appear in both. The next measure is the GC content, right? So, you have studied what GC content is, right.



So, this is the percentage GC base in the genome of an organism, right? So, some organisms may have a high GC bias, and some organisms may have an AT bias, right? So, this is something that is there. So, what does this tool do? It gives you a distribution—the observed distribution of GC content in that case, ok? So, this is something you see here in red.

So, the red distribution is the observed distribution, right? So, along the x-axis, we have mean GC content, right? So, for this data, it is around 40 percent, right? You see this point here, right? So, it is around 40 percent.

Again, this can vary from one organism to another. And along the y-axis, you have the number of fleets, ok? Then you have the blue line, which is the theoretical distribution. So, this is a distribution that is actually generated by the tool based on the data, ok? Remember, the tool does not know which organism is being sequenced.

So, it does not have any idea about the expected GC distribution, right? It simply generates this theoretical distribution from what it sees in the data. Now, this looks fine. It means it shows that the theoretical distribution kind of matches the observed data. And most importantly, this is a normal distribution, right?

So, any deviation from the normal decision would suggest there is something wrong or something else has happened with the data set, right? Now, this distribution can shift left or right depending on the organism that we are working with. So, this is something that you can also check yourself, right? Once you know that my organism should have this percentage of GC content, you can check back and see whether you see that percentage here in this plot. This kind of feels like validation that you are actually getting the right organic data, right?

Sometimes, as we have discussed during library preparation, you can have contamination from other organisms. For example, you are working with primate genome samples, and maybe you got some human contamination as well, or maybe you are working with yeast samples and got the human genome. And if these two samples have different kinds of biases—right, GC biases—maybe they will show up here. Then the next measure is per base sequence content, ok? So, this is something that is again looking at position-wise GC and AT content, ok?

So, the earlier measure was overall GC content, right? This is par-based GC content, right? So, based on sequence content again, looking at each position, you have this percentage of AT GC, right? In four different colours, you can see the legend here, right? In four different colours, you have this percentage AT GC, and this kind of matches with what you have seen earlier, right?

What we see for each position is that the GC content is around 40 percent, right? So, G is about 20 percent, C is also about 20 percent, and so GC content is about 40 percent and AT content is about 60 percent, right? So, in the earlier distribution, we also saw that this GC content was about 40 percent, ok? So, the initial part maybe you want to see, right? In this initial part, you see like a lot of fluctuations, right? And as we have discussed, right, when we are discussing the quality control data format, etcetera, we have discussed, right, at the beginning of the sequencing on the Illumina platform, you have this low-quality score, because this is where the sequencer is kind of identifying the cluster, setting the thresholds for signals, etcetera.

So, in this part, this kind of fluctuates a lot, but this means that we do not have to worry about too much. The rest of the data looks fine to us. So, there could be different types of duplicate sequences, and we will have a look now at the different types that you can get, especially on the Illumina platform. So, one is optical duplicates, where a single cluster is mistaken for two clusters by the detector. So, this is something that is kind of a problem with the detector, because instead of identifying a single cluster, it identifies them as two clusters.

So, you get the same sequence data for these reads. Then you can have clustering duplicates where the same molecule goes and occupies two adjacent clusters during the cluster generation process, right? So, again, we do not have control over exactly which molecule will go where in this cluster generation process. Now, this might happen, right? You can have the same molecule going to two adjacent clusters, and they will again give rise to the same read sequence.

Then you have the PCR duplicates. So, this is present on all types of platforms where you need to do amplification, right? This actually arises because some molecules are preferentially amplified, and then these molecules go and occupy different cluster locations. And finally, sister duplicates. So, these are actually a class of duplicates where complementary strands of the same molecule are found in independent clusters. Now, you might ask whether these are actually duplicates.

So, some programs or tools will mark these as duplicates, whereas others will not mark them as duplicates because they are complementary strands, which means we are sequencing from two different reactions. So, what is the problem with these duplicates? So, duplicates mean they are coming from the same molecule, right? They are not independent molecules that have been sequenced. They have the same molecule sequence multiple times. Now, what would be the problem? So, one of the first problems is that we have problems with the detection of single or multinuclear variants, ok?

So, we will discuss this in a lot more detail when you talk about this single nuclei variant detection, etcetera, right? So, the problem starts is that if you have a read that has a mutation compared to the reference sequence, right, and if that read is duplicated, present in multiple copies, right, you might see, you might say, ok. This mutation appears to be very dominant across my DNA library, right? In reality, this could be something else, like this mutation, which may be present at very low frequency in the library but may appear as a very frequent mutation in the library.

So, this is something we will discuss. So, this can happen, and this can also cause problems in the detection of the actual SNVs or MNVs. And as I mentioned, you also have problems with calculating the frequencies of these variants, the actual frequencies, because duplicates might alter or, in some cases, inflate the frequency. It might also cause problems in copy number variation analysis. Again, in terms of copy number variation, we are looking at whether there is any change in the copy number of a gene or a segment of the genome, but if you have too many duplicates for that region, they might appear as amplified or with an increased copy number, right? So, this is something we do not want; we want to avoid this kind of situation, ok?

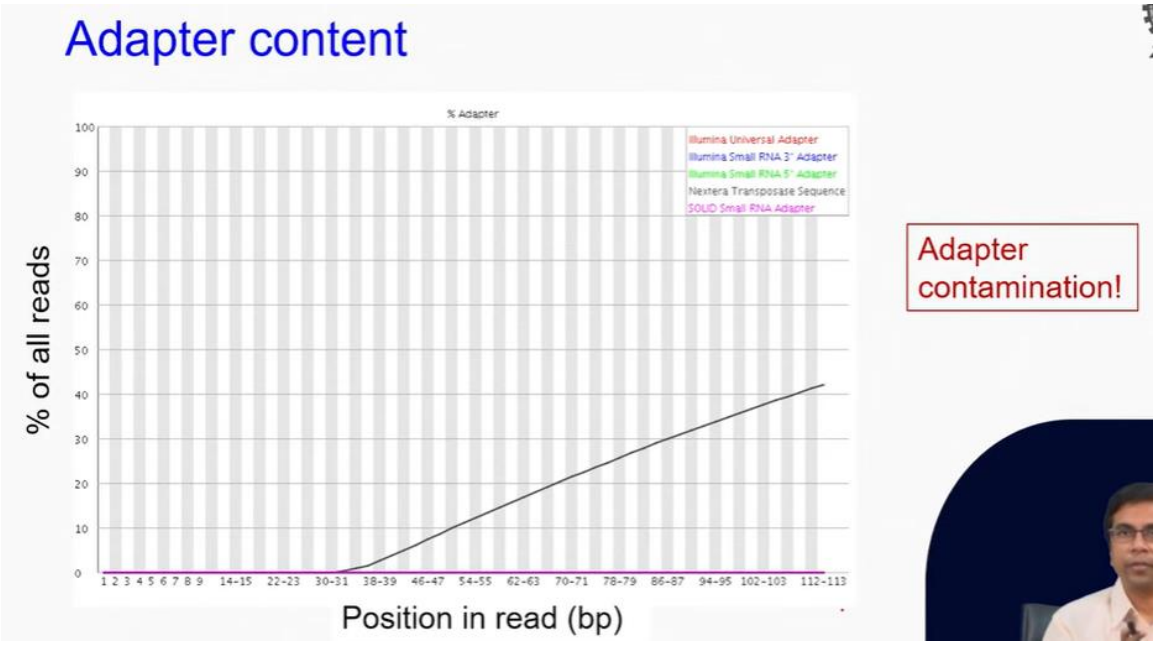
So, we will discuss, like, how we actually go about it and what could be the actual errors when we have this duplicate data, right? So, one of the things we do, right? So, coming back to this duplicate sequence and duplicated sequences, this is something that we have to worry about, and we have to pre-process the data if we actually can do that, right? So, before we actually go into the downstream analysis, So, it is also important to notice that this is not really that important for RNA sick data analysis when you are looking at your transcriptome data analysis because in transcriptome analysis, it is expected that if you are sequencing at a very high depth, you will get duplicates, right? Because in transcriptomics you are looking at only a limited part of the genome, only a small part of the genome, right?

So, these regions are kind of limited and can get multiple copies of the same fragment, ok? So, in RNA seq, we are usually not doing this duplicate analysis or duplicating, ok? Another problem that might occur, and this is something that this tool will look at, is something called overrepresented sequences. So, in the NGS lab, you can imagine that we are working with big genomes where you have fragmented this DNA into small fragments, and they are likely to be very diverse, and you should not see a single rich sequence at very high frequency and occurring multiple times in the library, right? This kind of problem might occur, right, this program will identify whether there are represented sequences, and if they are present, this might mean that there is something relevant biologically. Right, then there are some important biological phenomena that have happened that have enriched certain parts of the genome or certain parts of certain genes, etcetera.

Secondly, it could mean there is contamination; right there could be large-scale contamination from other fragments in your sample. So, you may have to be careful when you are processing the sample, and number c, maybe the library preparation was not good enough; probably many of the genomic fragments were lost during library preparation. So, the library is not as diverse as expected, ok? So, these are the things that you have to worry about. The third point, of course, is something that will require library preparation and sequencing again.

The first problem is something interesting and relevant; the second problem we can deal with through computational tools, ok? So, we will go forward and see how we can do that. The final point that this tool will look at is something called adapter content. So, you remember these adapters that are added at the end of DNA fragments while we are doing sequencing, right? So, these adapters could be different, and usually, they are different.

What you see here is a plot that is generated by this tool: along the x-axis, you have position in read again a certain length, and along the y-axis, you have percent of all reads. This program will check whether it finds some adapter sequences that match certain adapter sequences in this data. And here in this data set, you can see there is some adapter content or adapter contamination, ok? This black line actually goes up, and as we go towards the end of the read, the percentage of reads showing this adapter sequence actually goes up quite high, right? About 40 percent of the reads actually have adapter contamination.



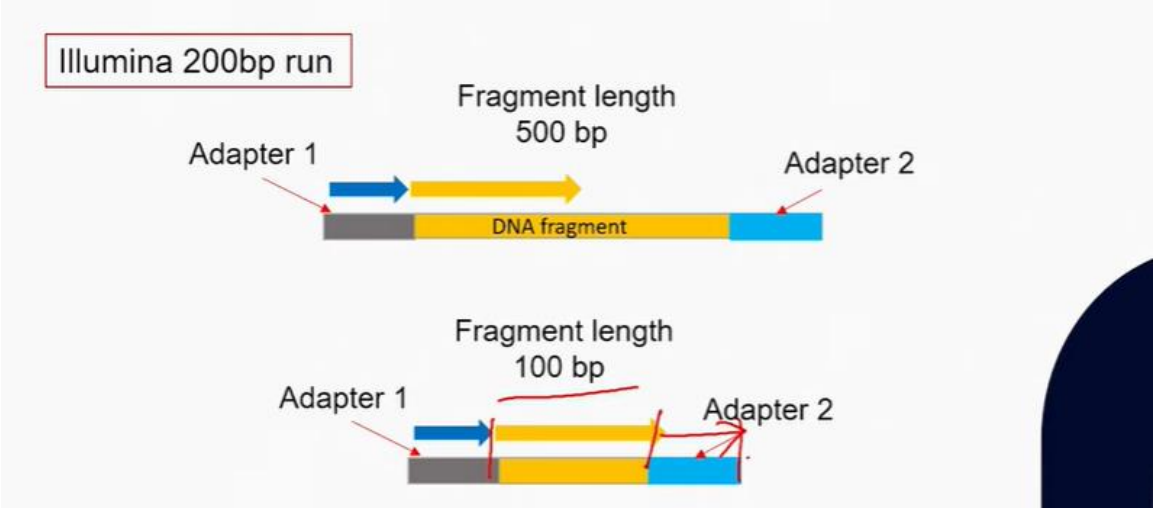
This is something that is a problem; it also shows you the type of adapter that is actually contaminating the data, right? So, in this case, the black line means this is a next-era adapter that was used for this library preparation, and the other adapter sequence is not present, right? You can see this one in purple; the solid small is an adapter, right? So, this is not present, but the tool could find this next-era adapter in the data, ok? So, this is again a very important observation, and it means we need to actually pre-process data.

Now, the question that you have is: how does this adapter contamination happen? They are not supposed to be there in the read data, right? We should get the sequence of the DNA fragment or the genome fragment, right? So, one of the things that I will mention is that this adapter contamination depends on the length of DNA fragments that are present in the library. So, let us take this example, and then we will probably understand how we get this adapter contamination.

So, let us imagine we are doing Illumina to 200 base pair runs, ok? And we have prepared this genomic fragment and some fragments with these adapters, right? Some fragments are, let us say, 500 base pairs. So, here is the DNA fragment, 500 base pair length, and you have adapter 1 on one side, adapter 2 on one side, and right on the other side, and you are sequencing, let us say from one side, ok. Now, once you sequence, you will sequence 200

base pairs from this side.

So, this is fine, right? You will get this sequence within this DNA fragment. Now, imagine the scenario when this DNA fragment is actually small, ok? So, remember when you are preparing this library, when you are fragmenting the DNA, we have very little control over the actual size. Unless we are doing enzymatic digestion, we cannot really accurately control the fragment that will be generated. So, this is something that has been optimised with different trials and errors, ok? So, if your fragment length is small, if it is 100 base pairs only, we are sequencing 200 base pairs, let us say, right?



So, what will happen? When you start sequencing from this side, it will sequence this 100, and then it will go into adapter 2, right on the right side. So, this adapter sequence will come as part of your read sequence, ok? Because the fragment length is small, you end up sequencing the adapters, right? So, this can happen both in single-end sequencing and paired-end sequencing, right? So, if you are sequencing from the other end also, you will end up getting the sequence of this adapter along with this yellow genomic fragment, right?

So, this will give you this adapter continuation, right? So, what is the impact of adapter contamination? Why are we worried about this? So, it actually can impact that in this data analysis, and as you will see, the mapping quality suffers because the adapter is a unique sequence and there is no match with the genomic DNA. So, the program that will be used for aligning this reads back to the reference genome, right? The mapping problem here might be that the program might not be able to find this adapter sequence in the genome

data, right? So, if it does not find it, then it might get confused. Right, what is happening is that there is no alignment, there is no mapping, and it might actually end up discarding the read from analysis, ok? So, there might be some issues with the mapping, ok? So, this is something that is a very important point, because mapping is actually the first step for a lot of downstream analysis.

So, this is something you want to do very well. So, these are some of the quality controls that we have talked about in FASTQC, but there are, of course, other quality control tools. For example, you have FASTP, you have HT stream, etcetera, and again, they have their own ways of doing things; they report different measures, but if you are interested, you can, of course, go and check these sites where you can actually find all the details about these tools. The next step, once you have done the quality control, is to do the quality evaluation. So, you have to, once these statistics are generated, kind of look at all these statistics and make decisions about how you should proceed with the data analysis.

So, the tool FASTQC will not do anything for you, right? It just shows, ok, there might be problems here, there might be problems there, or this might be, ok, etcetera. Now, you have to make this decision yourself. Right, what are you going to do? Okay, and the first step in this is the data preprocessing. So, preprocessing actually consists of two steps. So, the first one is duplicate removal. So, as we have seen, if there might be duplicate sequences, and if the duplicate table is very high, you want to get rid of these duplicates probably first, ok?

So, this is something that we will do if it is required. For example, if you are looking into single-nucleotide variance or doing CNV analysis, we want to get rid of duplicates. The second step is something called data trimming, ok? So, we will discuss what data trimming is in a moment. So, for duplicate removal tools, we have two different tools I will mention, but you can use other tools. There are multiple tools available, but we will focus on maybe one or two tools in this course.

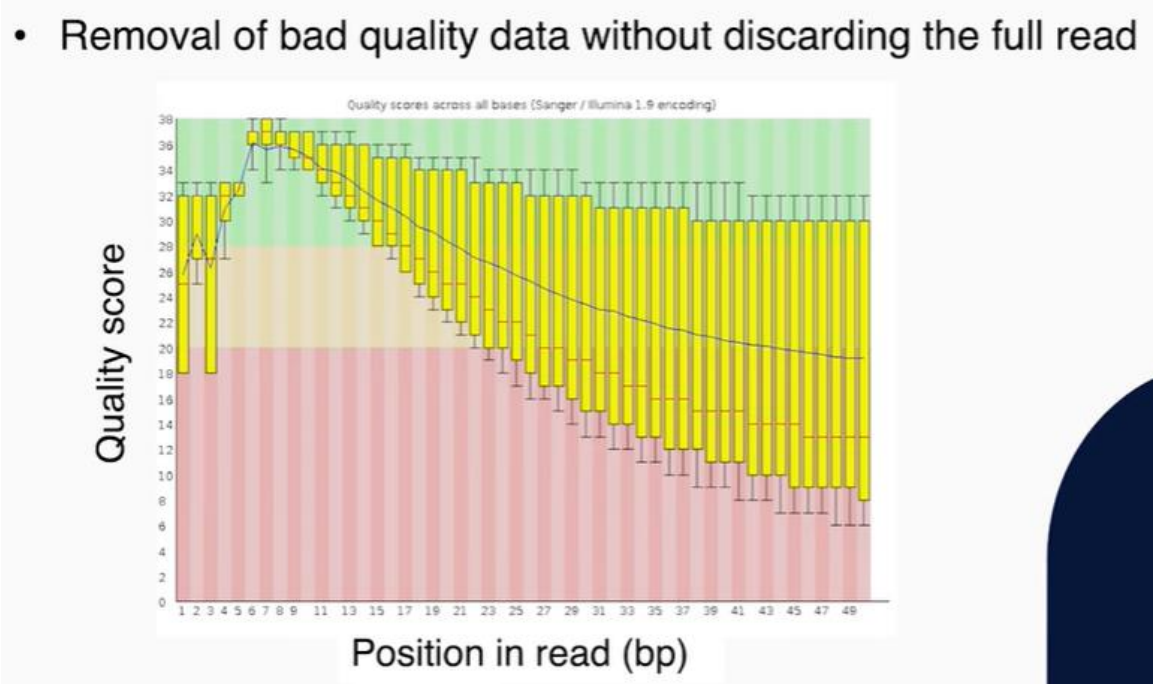
So, one tool is Picard, right? So, again, I am giving the link so that you can go and check; you can read about the tools, how they work, etcetera. All these details are available, and

there is a function in this tool. So, if you are familiar with Linux commands, you will find this function, mark duplicates, in this Picard tool. If not, do not worry we will discuss how you run these Linux command lines and how you run all these tools. So, you have these options: mark duplicates or remove duplicates. So, you can remove these duplicate sequences, or you can use the command marked duplicates minus minus remove sequencing duplicate, ok?

So, they have similar functions. The other tool that you can use is called Samtools. So, here again, you have the link, and again, it has some function called the mark duplicates function. So, if you want to remove duplicates, the mark duplicates function will just mark the duplicates, right? If you want to remove them, you have to use this command called mark duplicates minus minus R, ok? Again, command line tools, how to write this, all commands, etcetera, we will discuss hands-on.

Now, coming to the next step, which is the data trimming, ok, and this is what we would have to do, right? So, remember, duplicate removal we may or may not do depending on the level of duplication and the kind of analysis that you are doing. If you are analysing RNA seq data, we may not worry about duplicate removal, etcetera. But if we have some sort of adapter contamination, etcetera, we will have to do trimming, ok? This is a step that is required. Now, there are two types of trimming that you can do: quality trimming and adapter trimming.

So, what is quality trimming? So, this is the removal of bad-quality data without discarding the full data. So, we have discussed this kind of situation, right? So, imagine this situation where we are looking at this quality score par base quality score, x-axis you have the position in read, y-axis you have the quality score, and what you see towards the end, maybe beyond this point or so, the quality score drops significantly, right? So, you may not want to go ahead with reading this data, right? So, what are the options? One is, of course, let us discard this data for which you have this kind of low quality score, and this might end up kind of removing a lot of data that you have generated.



The other option is that maybe we can do something called quality trimming, right? Maybe we do not use this part of the book, right? Let us not use this right part, right? Let us trim this part of the read; only use the left part where the quality score is good, right? So, this is a good solution because we are not completely throwing out the read; you are actually working with good data that is available, right?

So, this is quality trimming. The other option is to remove contaminant sequences, such as adapter sequences. So, we have seen this adapter contamination scenario, where we have this adapter on the right-hand side towards the end of the sequence, and again we want to remove this adapter. So, again, what will this tool do that the trimming tools will do? They will identify, ok, there is this adapter sequence, and I should remove this adapter, right? So, this is where this trimming is important. So, in the example that we have taken, we need to do adapter trimming; we need to remove this adapter sequence, right?

And the importance of trimming is that we get an overall improvement in data quality, right? So, if you check your score, etcetera, sequence-based QCs, you will see this improvement without much loss of good quality data that you have, ok? As usual, there are many tools available. Again, we will use certain tools in our hands-on class. So, you can

go and check again. This one tool is called trimometric, right? And it can remove adapters; it can do this leading low-quality based removal; it can also remove certain bases in between the reads, right inside the read if they are below a certain quality score; and it can also remove reads that are shorter than 36 bases.

So, there is another tool, which is called the bbduk, right? So, again, it can do this quality trimming adapter trimming, and these tools use different algorithms for different ways of trimming, right? And we will use some of these tools in our hands. And finally, I just want to give you some links; if you are interested, just go and check. Again, people use all sorts of tools. You can pick anything and see which one works best for you. They are slightly different from here; they have the same function, but they are slightly different from each other in how they implement these functions.

And you have to apply it to your data and see which one actually works the best, right? So, these are again tools, right? So, for example, we have SQL, which trims based on quality score or type, which does trim based only on adapter sequences. So, these are the references that I have mentioned in this class. So, to conclude, we have looked into two types of quality control, right? So, we have looked into quality score QC, and we have also now looked at sequence-based QC and taken them together, right?

They give us an overall idea, but not the quality of the data. So, both of these QCs are critical for evaluating the overall quality of the data that we have generated, right? And this is a very important first step, right? So, you want to look at the data, and you want to do all sorts of pre-processing that is necessary for your data, right? We have also seen that duplicate reads can have a detrimental effect on some of the downstream analyses. Again, this is something you need to decide: whether you want to do duplicate removal and whether it is actually going to affect your results, depending on the level of duplication, etcetera.

Then we have seen that adapter contamination happens when the fragment size is small, right? So, I have given you a schematic of how this works and how it happens, right? And

this is actually a very common problem with many DNA sequencing projects. You will see this adapter contamination, and hence this downstream processing, right? So, the pre-processing, where we remove duplicates and also remove adapter contamination, is required for correct inferences.

So, these steps are really critical for all downstream analysis, whether you are going for genome assembly, identification of mutants, transcriptomics, etcetera. So, for any of these downstream analyses, you probably want to do the first step, which is the QC, and then, if required, you want to do the pre-processing. And there are two steps in pre-processing that we talked about duplicate removal and adapter contamination. So, once you have done this, you are now good to go with the downstream analysis, and you can be sure that whatever inferences come out of these data analyses will be robust and accurate, ok? So, with that, I will stop. Thank you.