Next Generation Sequencing Technologies: Data Analysis and Applications

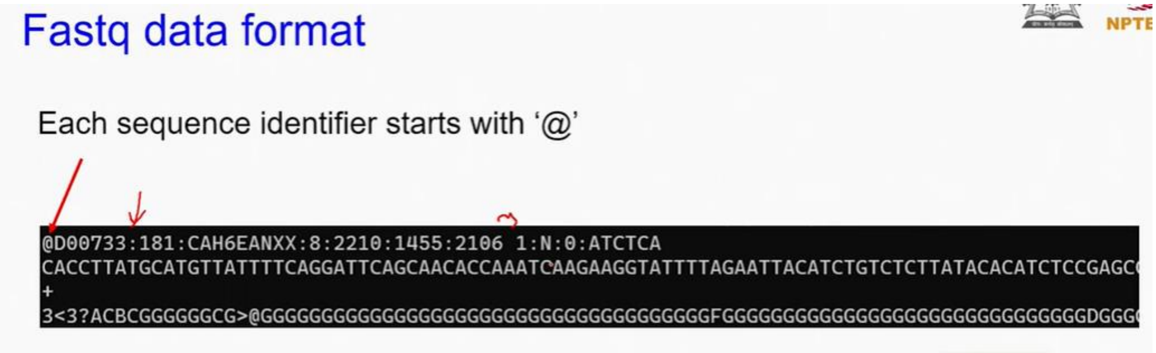
Data formats (Contd.)

Dr. Riddhiman Dhar, Department of Biotechnology

Indian Institute of Technology Kharagpur

Good day, everyone. Welcome to the course on generation sequencing technologies, data analysis, and applications. In the last class, we started a discussion on data formats. We discussed the FASTA format, then looked at the benefits and drawbacks of the format, and we also introduced the FASTQ format. So, to address the drawbacks of the FASTA format, this is the FASTQ format, which can combine sequence data along with quality scope. So, in today's class, we will be continuing our discussion on this FASTQ format and also looking at actual sequencing data.

So, these will be the concepts that we will be covering in this class. So, we will discuss the FASTQ format in a lot of detail; we will talk about ASCII encoding; we will talk about the FAST5 format; and these are the keywords you will come across: FASTQ, ASCII, and hierarchical data format, or HDF. So, as I mentioned earlier, data from the Illumina platform comes in FASTQ format, and we will take an example of this FASTQ format with Illumina data. So, this is the FASTQ format with quality score, or FASTQ.



Fastq data format

Each sequence identifier starts with '@'

```
@D00733:181:CAH6EANXX:8:2210:1455:2106 1:N:0:ATCTCA
CACCTTATGCATGTTATTTTCAGGATTCAGCAACACCAAATCAAGAAGGTATTTTAGAATTACATCTGTCTCTTATACACATCTCCGAGC
+
3<3?ACBCGGGGGGCG>@GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGFGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGDGGG
```

We have the header here, followed by the sequence data. There is a spacer plus, and then you have the quality score OK, and these are the reads 1, 2, and 3. if you have millions of reads, this will be there. You have these millions of sequences one after another in a single file. So, this could be quite a big file with all these sequences in place. Now, let us look at the header line. So, last time I just mentioned this is the header line; this contains some information, and each header line starts with this at the red sign, ok? But what about these other fields that are there? So, you will notice there are multiple fields separated by this colon, ok?

So, there are multiple fields there, and then there is a spacing here, followed by some information separated by a colon. So, we will go one by one and discuss whether this information is actually okay. So, the first information is OK. So, this is actually the sequence ID, right? So, what do we mean by this sequence ID? This is actually the Illumina machine number.

So, as I mentioned, the Illumina data comes in the FASTQ format. So, this is actually the machine ID, okay? So, in it is D 00733; this is the machine ID. Then you have the run number on that instrument for the experiment that is being carried out, right? So, in this case, it is the 181st experiment in this machine, ok?

Then you have the flow cell ID, right? So, hopefully, you remember the concept of flow cells, right? All the sequencing in Illumina is done on flow cells. So, this is the flow cell ID, and every flow cell has a unique ID. Then you have the lane number in the flow cell; if you remember the figure, you have multiple lanes right, and this is the lane number in that flow cell. The next three fields are actually showing the positions of this read in the flow cell.

So, we have the tile number 2210 here; this is the tile number in the flow cell. Then you have the x coordinate of the cluster in the tile right where the cluster is present, and this is the y coordinate of the cluster in the tile. So, you remember the cluster generation process, right? So, the x-y coordinate gives you the cluster location on the flow cell. So, why is it important? Why do you actually store this information? So, in general, if the data is good, we do not have to worry about it, but if there is some sort of issue, then this information becomes useful, ok?

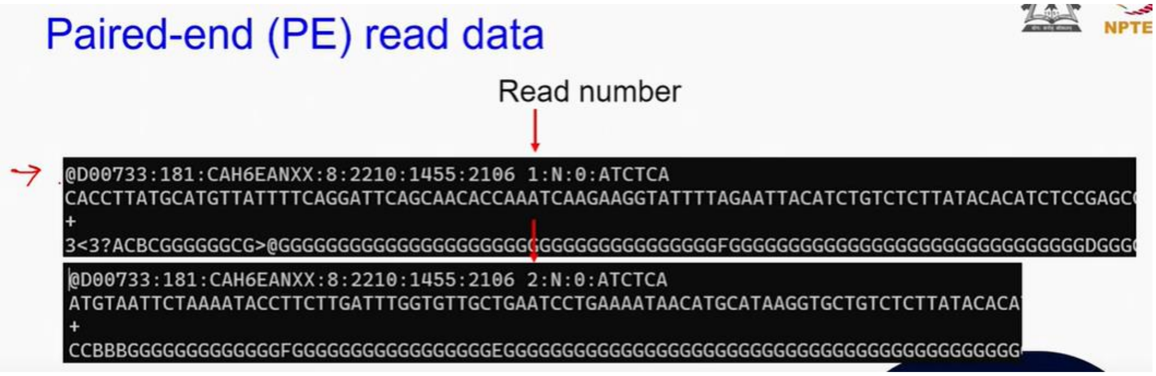
So, we will see this right when you look into the data quality check or quality control, ok?

So, this gives you the last three numbers; these are the positions in the flow cell. Now, you have this other part, right? So, the first part is about the machine, the location in the flow cell, the flow cell ID, etcetera. The other part is actually information about the read, ok?

So, this one is actually the read number. Now this one can mean two things ok? So, if it is one, it could be coming from single-end sequencing data. So, we talked about this; we are sequencing DNA fragments from one end only. So, this is where we have single-end sequencing data, and the read number will be given as 1.

But in the case of paired-end sequencing, where we are sequencing from both ends, reading number 1 would mean this is the read 1 right-only pair of like one of the pairs, ok, and then this means that for paired-end data, you can have reading number 2 as well, ok. So, in paired-end data, you can do paired-end sequencing on the Illumina platform, right? So, if you have two files, you get two FASTQ files, which will have exactly identical sequence identifiers, but they will have different read numbers. So, in this field that I just mentioned, the rest of the things will be exactly the same because they are coming from the same flow cell, same location, same time, etcetera, the same machine, but only the read number will be different. So, let us look at this paired-end data to see how it would be different, ok?

So, here is an example. I am just showing part of the read data. So, that is why the length is different; otherwise, the length will be identical. So, here is the first of the pair, right? So, sometimes we call them mates, too. So, this is mate 1, mate 2, ok?



Paired-end (PE) read data

```
@D00733:181:CAH6EANXX:8:2210:1455:2106 1:N:0:ATCTCA
CACCTTATGCATGTTATTTTCAGGATTCAGCAACACCAAATCAAGAAGGTATTTTAGAATTACATCTGTCTCTTATACACATCTCCGAGC
+
3<3?ACBCGGGGGGCG>@GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGFGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGDGGG
```

```
@D00733:181:CAH6EANXX:8:2210:1455:2106 2:N:0:ATCTCA
ATGTAATTCTAAAATACCTTCTTGATTTGGTGTTGCTGAATCCTGAAAATAACATGCATAAGGTGCTGTCTCTTATACACA
+
CCBBBGGGGGGGGGGGGGGFGGGGGGGGGGGGGGGGGEGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
```

So, this is the first of the pairs where you see the header line OK and the read number is given as 1 OK, and then you have the second one, which comes from a separate file OK. Remember, you will get to see two files, one of which will say write some name, read 1 dot FASTQ, and the other will say write some name read 2 dot FASTQ. So, read 1 read 2 FASTQ; this will also be reflected in the read number here. This read number will be reflected in the file name as well as in the header line. So, here is the header line number. For this one, the first read you see is shown as 1; for the second one, it is 2 read 2 right.

So, this is what tells us right: we are dealing with paired end data. We have two files; we have these numbers, and we are dealing with paired end data. The next field is this, which shows whether the read is filtered or not. So, when you are running some experiments in the instrument, the instrument signal processing unit, right, the software that processes signals and calls the bases, can do some filtering. It has some sort of filtering, and if the read is filtered, it has some issues, which probably means it is not filtered. So, it is actually of good quality as per the software, whatever the thresholds, etcetera. The next part is actually the control number.

So, this is something that is showing whether there are some control experiments that were run in this experiment, right? In some cases, you want to do some control for some tests, but in this case, there is no control experiment. The last bit here actually shows the sample number, or usually the index sequence. So, you remember that this index sequence is there when you add those adapters? So, during adapter ligation, you can have this index sequencing right, and this index sequence allows you to add multiple samples to one experiment.

So, what we call sample multiplexing is okay. So, this gives you the index sequence, okay? So, just to summarise again in one single table, right? So, if this is the header line, you have this information there. So, you have this instrument ID right here, DE 00733, and then you have this run number on that instrument flow cell ID, the location here, right this lane number tile number x coordinate y coordinate.

Then you have the read number, and we have discussed for single and paired ends what this number would be. They will be different whether the read is filtered or not, whether you have control experiments, and finally, the sample number, which is usually the index sequence. So, the next line is the actual DNA sequence. This is the sequence that we are interested in and we have the quality score OK, and in between I have the plus sign, which is the spacing. So, sometimes you will see this followed by this plus sign, and you will again see part of the header. Now, focus on the quality score, right? So, this is not a number; sometimes you see these numbers like 3 or 3, but most of them are like some letter like a, c, b f g etcetera, or some symbols, for example, a question mark or a greater than sign, etcetera.

So, how are these quality scores represented here? So, one thing that you should notice is that the quality scores are right. So, this quality score b here is for this base A, and this quality score at the rate is for this base T. So, you have this one-to-one correspondence, right? So, you can just simply read the quality score down here, ok?

So, this is something that is very convenient in fasta format. Now, the question is: how are these quality scores represented? So, what do these numbers mean? Can we extract this information and correlate it with the probability of error that we discussed in the last class? So, using probability of error and accuracy calculations, can we do that? So, it turns out these quality scores are ASCII-encoded.

So, what is this ASCII encoding? So, these numbers are represented by some sort of symbol or letter. This is the encoding that we use. And similarly, we see we have these 4 columns right. So, for numbers from 0 to 127, you have these corresponding characters, ok?

So, these characters can represent these numbers, and this can help save space when you are representing these quality scores. Now, one of the things you probably notice is that only few numbers would be usable for us, and this is actually after 33. Before 33 we have again the same problem; we are using 2 letters, etcetera, and then we run into the same problem with 2 digits. Only after 33 right can we use these encodings, OK, and up to 126.

So, this is the range in which you can run.

Now, the problem is that the quality score comes from 0 to 40. We said it can go higher than 40, but usually in NGS datasets, the most likely maximum is 40. So, how do we represent that? So, what we know is that Illumina uses something called PHRED plus 33, encoding OK, but it earlier used PHRED plus 64 encoding OK. So, what is this PHRED plus 33 encoding? So, the idea is very simple, right? So, if the PHRED score of the quality score that comes out of the machine is 25, you just add 33 to that.

So, here is 25 plus 33, which is 58, and simply look at the ASCII encoding for 58, which is a colon. The score range that is available, as I mentioned, is between 33 and 126. So, if your quality score from the machine is 0, your PHRED plus 33 encoding value would be So, the PHRED plus 33 value would be 0 plus 33, which will be 33. So, encoding would be this question mark, sorry, exclamation mark, right?

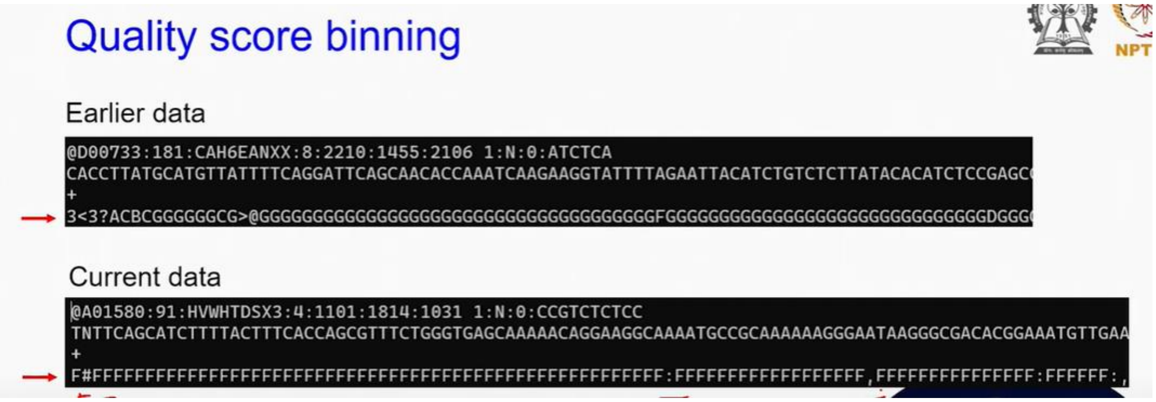
Similarly, you can calculate for each quality score what would be the PHRED plus 33 value, and then you can use the corresponding encoding. On the other hand, if you see the encoding right, for example, you saw this, for example, this capital G OK in many of these data sets. So, for many reasons, it was G, right? You had this encoding G. So, G now corresponds to 71 right? Now, if you want to calculate the actual quality score, this will be 71.

So, the actual quality score would be 71 minus 33 in PHRED plus 33 encoding right. So, this number you can calculate now, and based on it, you can calculate the probability of error and the accuracy. So, as we discussed earlier, So, now we have a way to do the encoding or get the quality score from the encoded value. Earlier, Illumina used PHRED Plus 64 encoding.

So, in that case, the range was different, right? So, for the PHRED score, if you take the same example of a PHRED score of 25 in PHRED plus 64 encoding, you will get 24 plus 64. So, this will come out to 89 right, and then SK encoding for 89 would be capital Y OK, and the score range that was available in PHRED plus 64 was 64 to 126 right? So, if you have a quality score of 0, your PHRED plus 64 value would be 64, right? So, that was the lowest value that was available, ok?

So, now we understand how this encoding works, and we can figure out what the actual Q value is if we know the encoding. Now, there is something else that is also added along with this encoding, ok? So, what Illumina does now is do something called quality score binning. So, what is this quality score binning? So, it simply says, instead of having every possible quality score, why not just give a median value or some sort of representative value for each bin of each range of quality score? So, the entire quality score can be divided into, let us say, 8 classes here.

So, in this example, it is 8 classes, or 8 bins, right from 2 to 9, then 10 to 19, etcetera, and the final one is greater than 40, and for each of these bins, why not just give one quality score value? So, every base that falls within this range will be given a quality score that is representative of that bin. So, for example, we can take this example here. Let us say any base that has this quality score between 35 and 39. The quality score will be given 37. So, all bases will be given a quality score of 37. So, there is some sort of change, and this actually helps in saving storage space, it turns out.



Quality score binning

Earlier data

```
@D00733:181:CAH6EANXX:8:2210:1455:2106 1:N:0:ATCTCA
CACCTTATGCATGTTATTTTCAGGATTCAGCAACACCAAATCAAGAAGGTATTTTAGAATTACATCTGTCTCTTATACACATCTCCGAGCC
+
3<3?ACBCGGGGGGCG>@GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGFGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGDGGGG
```

Current data

```
@A01580:91:HVWHTDSX3:4:1101:1814:1031 1:N:0:CCGTCTCTCC
TNTTCAGCATCTTTTACTTTCACCAGCGTTTCTGGGTGAGCAAAAACAGGAAGGCAAAATGCCGCAAAAAAGGGAATAAGGGCGACACGGAAATGTTGAA
+
F#FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF:FFFFFFFFFFFFFFFFF,FFFFFFFFFFFFFF:FFFFFF:,
```

So, again, without going into too much detail, you can see how it will work. You can look at Illumina's data, etcetera, and see how this helps in storage space. So, because of this quality score binning right in the earlier data, you will see these kinds of numbers: question mark A, C, B C etcetera, all sorts of letters, etcetera. But in newer data, right in current data, if you look at this quality score encoding, you will see only a limited number of letters, not all possible letters that are out there. So, for example, in here you see, for example, this hash if you see a colon and a comma ok, and these 4 letters you see in this quality score because of this quality score binning ok. So, most of the bases will appear similar, and when they fall into one range, they will be given the same value.

So, that is for the Illumina data format. What about the Pacific Biosciences SMRT sequencing data format? Do they use the same FASTQ format? So, the answer is no, they have different data formats, and we will discuss right now how this looks. So, in this SMRT sequencing, it reported the data in HDF5 format earlier. So, we will talk about what this HDF 5 is in a moment, and the base call data was present in HDF5 format. But currently they use something called BAM format, and we will again talk about what this BAM format is when talking about the mapping problem, etcetera. So, BAM format is a machine-readable format that is a binary format, and this BAM format can be converted to FASTQ format.

So, there are tools that will allow us to convert this BAM format to the FASTQ format. So, why are we interested in getting this into the FASTQ format? First, most of the tools were developed for the FASTQ format. So, we have mentioned that most of the sequencing today is done on the Illumina platform. So, most of the tools that are out there—a lot of the tools out there—are designed for handling FASTQ datasets. So, this is something that will help. If we can convert this BAM format to FASTQ format, you can use those tools, ok?

So, there are two tools just mentioned here: one is BAM Tools, and the other is Pacific Bioscience's own tool, which is called the bam2fastx. You can download this data from their webpage; the web link is given. Now, what you will probably see if you are working with the SMRT sequencing data and you have sequences recently right in all current pack by sequences, like if the systems that we have shown right in the last few classes back are sequel I, sequel II, and sequel IIe systems. By default, the quality scores will be set to 0, and the SK encoding for all of them will be an exclamation mark. So, as you have seen in

Plate Plus 30D format, SK encoding is an exclamation mark, right? So, if you download the data and see the quality score of all bases, this will be given as an exclamation mark.

So, this is something we see, and apparently quality score data cannot be computed reliably for raw data from SMRT, and they are not relevant. Based on what that bio has been suggesting, it is actually better to look at the sequencing data itself rather than the quality score. Now, moving on to the ion torrent data format, So, what does the ion torrent data look like? So, the raw data in the ion torrent is stored in those files, ok? So, ion torrent raw data means you have the current data on the pH change, right? So, you have this pH change that will induce more current, and that signal data is stored in these files, and from that, you get the sequence data in either BAM format, FASTQ format, or VCF format.

So, do not be overwhelmed by all these formats; we will discuss this later on. For example, BAM will discuss the human-readable version of BAM, and we also discuss the VCF format briefly when we talk about the mutation data analysis. So, if we can get the data in FASTQ, we can use all the tools that are available for Illumina data analysis, and here in the FASTQ format, the quality score follows with plus 33 encoding. So, again, you know the actual quality score, you know the actual error you can calculate it yourself, and then you have the nanopore sequencing data format right. So, here the data comes in something called the FAST5 format.

So, this is a special form of HDF5 format, ok? And how does it look? So, before that, we can go a little bit into the HDF5 data format, ok? So, as you probably have seen, in many cases we have this HDF5 format. In the case of SMRT, there is some earlier-used HDF5, and here also we are seeing this FAST5, which is a variant of HDF5, which is a simplified version of HDF5. So, the name HDF5 comes from the hierarchical data format. So, HDF, and this is version 5, I think also have the earlier versions HDF4, etcetera, and this data format is used to store very large, complex data sets.

So, it is not just for storing this next-generation sequencing data; it can also be used for storing other different types of data sets. And the strength of this data set is that you have the metadata included in the data set. So, you have the data set, and you can have as many data sets as you like, and for each of the data sets, you can also have the metadata saying what kind of data it is, or maybe the condition under which it was collected, etcetera. So, it is a nested data format, which means you have groups OK, and this contains data sets or other groups. You can have data sets, which are actual measurements of data, and you have attributes for groups and data sets OK.

So, are these terms actually okay? So, you can think of these nested data in terms of the file system in Windows, ok? So, if you have this file system, you have this directory right under it. You can have a file, or you can have a subdirectory. Under the subdirectory, you can have, let's say, 2 files and another 2 subdirectories. So, similarly, this is what that system might be. So, you have these groups, right? You have one group, and you can have one data set under that group and another group under that group subgroup.

So, this can propagate, and in each group, you can have multiple data sets as you wish, and you can classify them according to the experiment the data collection method, etcetera. The data sets are the actual data, right? So, for comparison with the Windows file system, these are the files, ok? So, these are the actual data set measurements, ok, and then attributes are like features, ok. So, for each of these data sets or groups, you can add features saying this data set was collected for this condition this data set was collected for this condition, etcetera.

For groups of data sets, when you define groups for these groups, you can say that this group of data was collected from this location, perhaps right? So, that kind of thing is okay. So, as you can see, you can store a very complex data set along with the metadata in here, and this is a data format that is widely used. FAST 5 is a specific structure of this HDF 5 data.

So, it is a simplified version; it is not as complex. So, what you have is something called raw data and analysis data in FAST 5. So, what is this raw data in FAST 5? So, if you go back to the nanopore sequencing principle, the nanopore sequencer uses current to determine the base sequence. We have these nanopores through which the DNA molecule passes, and as the DNA molecule passes in these terms, depending on the base compositions or the KMRs that pass through the nanopore, it gives a specific pattern that can be detected. So, the raw data is the values of these currents in pico amperes. These are very small currents and this raw data is stored at different time points.

And then you have the analysis data which is the base called data, right? So, we apply different algorithms, or the artificial neural networks that we talked about, to the raw data to detect the base sequence of the base KMRs. So, this analysis data contains the base call data. So, this is something that is now standard, right? So, we have the raw data in there, and the base call data is also available. Now, why is this raw data important? So, for many applications, this raw data is useful because it contains real-time data on how that current is changing in real time, and that data can be utilised for specific applications.

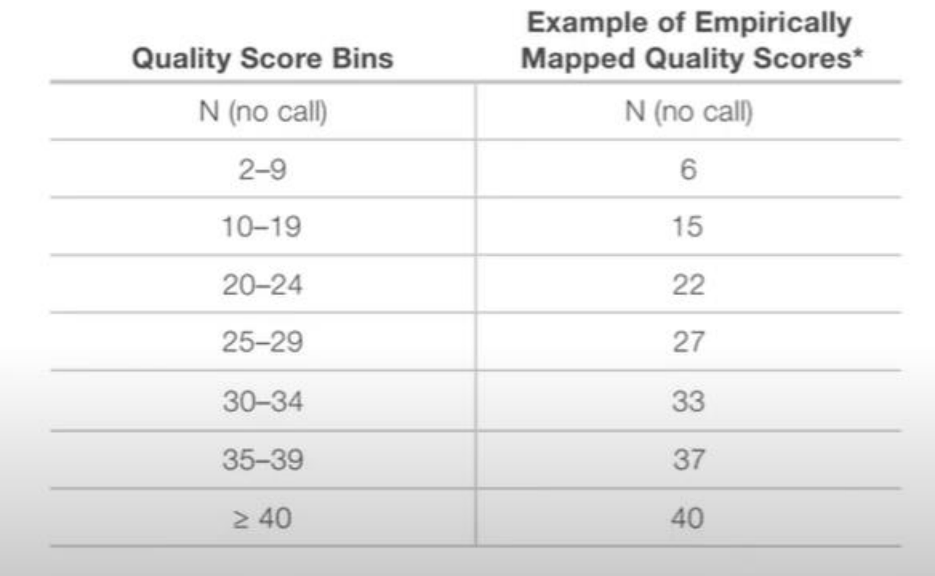
So, these are the references that we have used just to conclude. So, we have talked about the FASTQ data format in detail. So, this is a data format that stores sequence and quality scores in one place. So, that is a very efficient way of storing data because it helps in parsing the files very easily; we do not have to go back and forth between one and two files. So, between this sequence file and the quality file, So, as was the case in the case of 454 right, you had to really struggle between these two files when you were writing your own programme.

It was time-consuming it took a lot of time, and especially when you are dealing with a huge amount of data and maybe you are dealing with multiple such files, this could become troublesome. So, this is something that needs to be taken care of, right? So, that was taken care of by this FASTQ data format, where you had the sequence and the quality score in one place. Then we also saw the quality score in one place, right? The quality score is also ASCII encoded, which again helps in saving storage space, and we also saw the quality score binning in Illumina data, which also saves storage space. So, earlier Illumina data used all sorts of ASCII encoding, right, all possible ASCII encodings for all possible quality values, but now it kind of bins the data into a certain number of bins, a specific number of bins, which could be 8 or maybe less, and then it assigns a representative value for the quality score in that bin.

So, this also helps in simplifying the data analysis, and it also saves on storage space. The current ASCII encoding that is used by Illumina follows PHRED plus 33 format; earlier data used PHRED plus 64 format. So, this is something you need to be careful about when you are analyzing Illumina data. This will come into play when we are looking into the data analysis path, ok? So, if it is PHRED plus 33 you need to then it is ok you need to be aware of that, but if it is PHRED plus 64 again you need to be aware of with that you are following PHRED plus 64 right otherwise you will kind of overestimate the quality ok.

So, this is something that will be mentioned when you download data from the archives. When you download earlier data sets, this quality encoding will be mentioned along with the data set, whether it is following PHRED Plus 64 or PHRED Plus 33. So, specifically, it will mention when it is PHRED plus 64. And finally, we saw that nanopore data is stored in FAST5 format, which is a variant of HDF5. So, where you have the raw data, which is the current data, and this raw data can be useful in case you have some applications where you want to look at the real-time events that are happening right now, So, this is something that is there in the first format, and then you also have the analysis data, which is the actual sequencing data.

So, this sequencing data will give you the actual sequence, and from the raw data, you can also extract the quality score because, given the current etcetera, the base call algorithm can give you the quality score. Now, one of the things that probably we have not discussed right now is how you process this HDF5 data, etcetera. We will talk about that in the data analysis part. For FASTQ, we can read them quite easily. You can open them with an



Quality Score Bins	Example of Empirically Mapped Quality Scores*
N (no call)	N (no call)
2–9	6
10–19	15
20–24	22
25–29	27
30–34	33
35–39	37
≥ 40	40

editor, and you can read one after another. Right, one read after another. For FAST5, we have not discussed it; we can discuss it later. With that, I will close. Thank you.