1) edgeR stores data in object called DGEList

* Counts- integer counts. Put in raw, rather than transformed counts.
* Samples- info about the samples or libraries. Contains a column lib.size for the library size or sequencing depth for each sample. If not user-specified, edgeR will be computed from the sums of the counts. Must also have a “group” designation

2) Then, filter by CPM and “it is recommended to recalculate the library sizes of the DGE object after filtering though the difference is usually negligible.

3) edgeR does not generally need to adjust for gene length because gene length has the same relative influence on the read counts for each RNA sample. For this reason, normalization issues arise only to the extent that technical factors have sample-specific effects.” 🡪 in this case, we want to correct for species-specific gene length

4) Previously been noted that a small number of genes can be very high expressed in one sample but not in another. These genes can alter the ratio of more lowly expressed genes between sample 1 and sample 2. Therefore, we want to correct for the RNA composition effect.

* Normalize RNA composition with calcNormFactors
* calcNormFactors finds a set of scaling factors for the library sizes that minimize the log-fold changes between the samples for most genes. Default for computing scale factors uses TMM, a trimmed mean of M values
* Results in the effective library size

5) EDASeq and cqn packages estimate correction factors for GC content

Other possible ideas discussed:

A) Get TPM for each transcript so look at TPMs/gene (log normally distributed except high variance for lowly expressed genes) and then use voom to get the weights

B) Scale Counts\* = counts/[gene length / 10^3] 🡪 input to voom 🡪 output from voom will be the logCPM of the scaled counts, which is equivalent

Output from voom will be logCPM of scaled counts, which is how the RPKM is calculated.