**Running CrisprCountsAnalysis (CCA) and   
Understanding the Output**

**Running CCA**

1. Make sure Docker is installed on your computer.
   1. For Windows or Mac use Docker Desktop (<https://www.docker.com/get-started>) and for Linux, use the Linux Engine (<https://hub.docker.com/search?q=&type=edition&offering=community&operating_system=linux>).
2. Create an empty folder. For the purposes of example, let us say the folder is /Users/login/temp.
   1. WARNING: make sure that the folder path does not have spaces or special characters in it. Docker seems to have trouble with spaces.
3. Please make sure to read the “CrisprCountsAnalysis (CCA) Input Files Format” document.
4. If you have a read-counts file, skip to step 5. Otherwise copy the FASTQ files to the empty folder and make sure the FASTQ files end in .fastq and are named something simple and there is one FASTQ file per sample. Also, make sure the sequences are appropriately trimmed so they only contain the barcode related to the library. As an example, let’s say the files are named sample1.fastq, sample2.fastq, and sample3.fastq and the TKOv3 library was used. One may have more files, if one chooses. For this example, let us say the project id is EX1. The first time, one runs this, one will have to pull the Docker image by doing (all on one line)  
     
   docker pull tohsumirepare/cca  
     
   The above need not be done subsequently. Then do the below (all on one line)  
     
   docker run --rm -it -v "/Users/login/temp:/home:rw" tohsumirepare/cca convertCrisprFastqsToReadCountFile IN=sample1.fastq,sample2.fastq,sample3.fastq OUT=EX1.txt OUT\_GENES=EX1.genes.txt SUMMARY=EX1.summary.txt REF=/root/data/libraries/TKOv3.fasta  
     
   Please edit EX1.txt so that the sample names at the top conform to those described in the input file format document.
   1. Valid options are Brunello, TKOv3, and yusa.
      1. If a different library if used, then create a FASTA file with header and sequences of the form  
         >mylibrary\_gene\_exon1\_1  
         FASTA\_SEQUENCE  
         …   
         where mylibrary is the name of the library and gene is the name of the gene. The exon1 and 1 represent the exon number that the sgRNA is in and the 1 represents the first sgRNA in that gene. Then create a new copy of the CCA Docker file with the new library’s fasta file in /root/data/libraries.
      2. Note that the gene names have been updated to somewhat more modern names from the original gene names used in the libraries.
   2. CCA is library agnostic and the library file is only used to create the read-counts file. If the read-counts file is already created, the library file is never needed by CCA.
   3. The author has noticed some strange behavior at times of the Docker system. Thus, it is recommended that if not all files are generated to try restarting the Docker system (if it is not clear how, restart the computer instead and start Docker) if such things happen. Also, make sure to check the validity of your input files.
5. For example, suppose the read-counts file is named EX1.txt and the repmap file is EX1.repmap and suppose this is an isogenic screen. Copy them to /Users/login/temp. If you haven’t pulled the image from the Docker hub yet (in step 4) then do (all on one line)  
     
   docker pull tohsumirepare/cca  
     
   One need to do the above only once. Then, do the below (all on one line)  
     
   docker run --rm -it -v "/Users/login/temp:/home:rw" tohsumirepare/cca crisprCountsAnalysis COUNTS=EX1.txt REPMAP=EX1.repmap OUT\_MAX=EX1  
   1. If the screen is a chemogenomic screen, please add  
        
      CHEMO=true  
        
      to the above line, noting the requirement of having the control pair in the REPMAP file.
   2. Depending on the speed of the computer, this should take roughly 15 minutes.
   3. There are many parameters for CCA. Some of these are mentioned in another document.
   4. The program will create folder cca\_files in /Users/login/temp as well as files EX1.cca.tsv and EX1.qc.html. There are a number of files in cca\_files, but one needs to lookat only the aforementioned two.
   5. The author has noticed some strange behavior at times of the Docker system. Thus, it is recommended that if not all files are generated to try restarting the Docker system (if it is not clear how, restart the computer instead and start Docker) if such things happen. Also, make sure to check the validity of your input files.
6. Open the file EX1.qc.html using your web browser and optionally EX1.cca.tsv using Microsoft Excel or equivalent program.

**Non-parametric Z-score**

In the plots and the CCA table, there will be mention of the non-parametric Z-score. This is not explained in the paper but explained here. Please refer to the article

M. Colic, et al., Identifying chemogenetic interactions from CIRSPR screens with drugZ, Genome Medicine 11, No 52 (2019). <https://genomemedicine.biomedcentral.com/articles/10.1186/s13073-019-0665-3>

and Github source code

<https://github.com/hart-lab/drugz>

for the original drugZ algorithm. We have reimplemented the algorithm but using non-parametric methods (median instead of average, interquartile range instead of standard deviation, etc.) instead of assuming a Gaussian distribution. It gives similar results to drugZ, but is likely less prone to outliers. We call this the non-parametric Z-score. Note that in our implementation, we have the most positive score being equivalent to having a positive score in CCA. In drugZ, the most negative score is equivalent to having a positive score in CCA.

Some notes about the non-parametric Z-score:

* While CCA takes into consideration the depletion of the test samples for a given gene, the non-parametric Z-score does not and is only measure the difference in depletion between the test and control samples for a given gene. This may be useful for chemogenomic screens.
* It is less used and thus less developed and less well understood than CCA. Please use with caution.

**Understanding the HTML Output**

* Total reads per sample on raw data and distribution
  + The first plot just gives the number of raw reads per sample. The red horizontal line is roughly 200X coverage on a TKOv3 library. All samples should be above that.
    - The red line may need to be elsewhere for other libraries.
  + The second plot gives the distribution of number of reads per guide for each sample. The horizontal line is 30 reads per guide. The vast majority of the distribution should be well above the red line.
* Distribution of essential and nonessential sgRNAs
  + The first plot is the distribution of essential genes with the red line being 30 reads.
    - It is all right if a good portion of the distribution falls below this line. This means the essential genes are being depleted, as it should be.
    - If almost all of the distribution falls below this line, that is NOT good, as it means the timepoint is likely too long and most essential genes are depleted.
  + The second plot is the distribution of nonessential genes. The vast majority of the distribution should be well above the red line.
* Clustering of raw read counts in linear space
  + This is a heat map of your samples and the Pearson correlation.
    - The Pearson correlation was used since here we do want to see abnormalities that may be driven by outliers.
  + The samples should cluster logically.
    - T0s should cluster together.
    - Control samples should cluster together. Similarly, test sample should cluster together as well.
      * Sometimes, replicates will cluster instead of by test vs control. This may be OK, as we have found that top hits are often robust even under such circumstances that the technical replicates are different.
* Analysis of normed read counts and foldchange
  + This should nominally be an empty section, but internally CCA checks the correlation between T0 samples and if any are < 0.9 a warning message is printed in this section.
* Depletion distributions
  + Reminder: the depletion is defined as 1 – foldchange.
  + The first plot is the distribution of depletion of essential genes.
    - What one should see is that the distribution is very on the right. That is, most of the essential genes when knocked out kill the cell.
    - **The plots of the control will not look the same as test since we are capping the foldchange of the control at 1 by default.**
  + The second plot is the distribution of depletion of nonessential genes.
    - Generally, the distribution should peak at zero for the control and may peak around zero (but not too far off one side) with the test.
  + The third plot is the distribution of depletion of the top 300 genes.
    - The test (blue) distribution should be more biased toward the right than the control (red) distribution.
      * This will show that the test hits should be depleted more so than the same genes in the control.
    - The control distribution should be similar to the distribution of the control in the nonessential genes.
* Comparative plots of control median depletions among various groups
  + If the control’s foldchange is capped at 1, then most of the distributions should be bunched at 0, except for the essential genes that should be more bunched to the right.
* Comparative plots of test median depletions among various groups
  + Here, one should see the essential genes’ peak being the rightmost followed by the top 300 hits’ peak. The nonessential and other distributions should be roughly overlapping. It is important that the essential genes’ peak is a little more to the right than the top 300’s peak since otherwise that means the final timepoint may be either too soon (essential genes did not kill) or too late (many knockouts killing the cell).
* Cutoff stratas
  + We show the stratification of the top 200 or so hits.
    - There is the stratification by CCA score and by the non-parametric Z-score.
  + The rightmost points are the top hits.
    - The rightmost set is called Jenks class 4.
    - The next rightmost set is called Jenks class 3.
  + Points on a vertical line belong to the set to the right of the line.
* Bar plots of the top 25 hits
  + **These are the most important plots!** They will show if your genes of interest are indeed true hits.
    - By default, the control foldchanges are always cut to 1.
    - The test bars (blue) should be significantly lower than the control bars (red).
      * It is OK if one of the sgRNAs do not show such difference. That may be because of the design of the sgRNAs.
        + Most sgRNAs should show a difference.
      * If one does not see significant differences, there may not be many (if any) true hits in the screen.
  + One can make bar plots of other genes as well. To do so:
    - Create a file whose first line is the word Gene. Subsequent lines are gene symbols such as BRCA1. There can be only one gene per line.
      * Recall the note regarding newlines in the CCA file format documentation. It applies to this input file as well.
      * For this example, let the file be called genes.txt.
      * If possible, rank the genes from the most important coming first. The rank of the gene within genes.txt can be part of the output file name.
    - If one ran CCA as mentioned earlier in this document, the foldchange file should be /Users/login/temp/cca\_files/EX1.correl.foldchange.tsv.
    - In /Users/login/temp, run the below (all on one line)  
        
      docker run --rm -it -v "/Users/login/temp:/home:rw" cca barplotGenesFromCcaFoldchange GENES=genes.txt REPMAP=EX1.repmap FOLDCHANGE=cca\_files/EX1.correl.foldchange.tsv OUT\_HEAD=EX1\_GENE\_PLOTS ADD\_RANK=true  
        
      which will produce files EX1\_GENE\_PLOTS.1.gene1.png, EX1\_GENE\_PLOTS.2.gene2.png, etc., where gene1 is the first gene in genes.txt and gene2 is the second gene in genes.txt. With ADD\_RANK=false, the file names are EX1\_GENE\_PLOTS.gene1.png, EX1\_GENE\_PLOTS.gene2.png, etc.
* CCA top hits QC
  + This table is a simplified version of the file EX1.cca.tsv.
  + It only contains the hits of Jenks class 3 and 4 of both CCA and the non-parametric Z-score.
  + If a top hit is essential or non-essential, it may still be a top hit, depending on the context (cell line) of the screen. However, one should proceed with caution.
  + The Positive Control column only appears if there is a positive control for that condition in /root/data/positive\_controls/{condition}/positive\_controls.txt.
    - Only BRCA1 and BRCA2 are included.
    - One may create a similar Docker file adding positive controls for other conditions as needed.
  + If the top hit is a tumor suppressor gene, it may still be a top hit, but again proceed with caution.
  + The rank and Jenks class for both the top CCA and top non-parametric Z-score genes are given. The best gene is with rank = 1 and the best Jenks class is 4.
  + The column labeled “Less than 20 percent of essential genes with a lower killing measure” is Y if the test median depletion depletes less than the bottom 20% essential gene ranked by the amount of depletion. If this is Y what it means is that the knockout of this gene will not result in a large amount of depletion/killing. This may mean that validation in the lab may result in minimal synthetic lethal difference between the test and control.
  + The “Has variance > …” columns is simply giving the amount of variance of sgRNA depletion between samples for either test or control for that gene. If it is large, the hit may still be valid, but it may indicate one or more sgRNAs did not cut well or have much of an effect.
  + The Ensembl paralog column is important to note since if there is a paralog, it is possible that the sgRNA may have cut either the paralog or cut in more than one place, thus yielding a false positive. Proceed with caution.
  + The biological function of the genes are provided for convenience.

**Understanding the CCA TSV table**

Most of the columns of the TSV table have been discussed in the HTML output section. Below are notes on the columns specific to the TSV table.

* The test and control median depletion is given for each gene. This give the magnitude of depletion of test relative to control.
* The Score and Rank columns is the CCA score and rank based on the score.
* The Negative non-parametric Z-score is the Z score described in the previous section of this document. It is called Negative because unlike drugZ, we multiplied the score by -1 so that the top most hits have a larger positive value. A ranking of genes based on this score follows in the next column.
* The Jenks class for CCA and the non-parametric Z-score follows.

**Final notes**

There are a number of files in the cca\_files folders. The PNG files used in the HTML output are located therein. However, it is advised not to look at the TSV files are many columns are not documented publicly and were meant only for development and diagnostic use.

The source code for CCA is in the Docker file. Please look in /root/src/Projects/Repare for the source code specific to the CCA package. Note that the other directories in /root/src not named Projects are from the MolBioLib library (<https://sourceforge.net/projects/molbiolib/>).