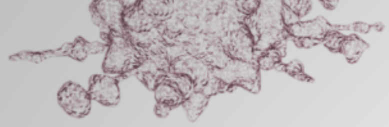
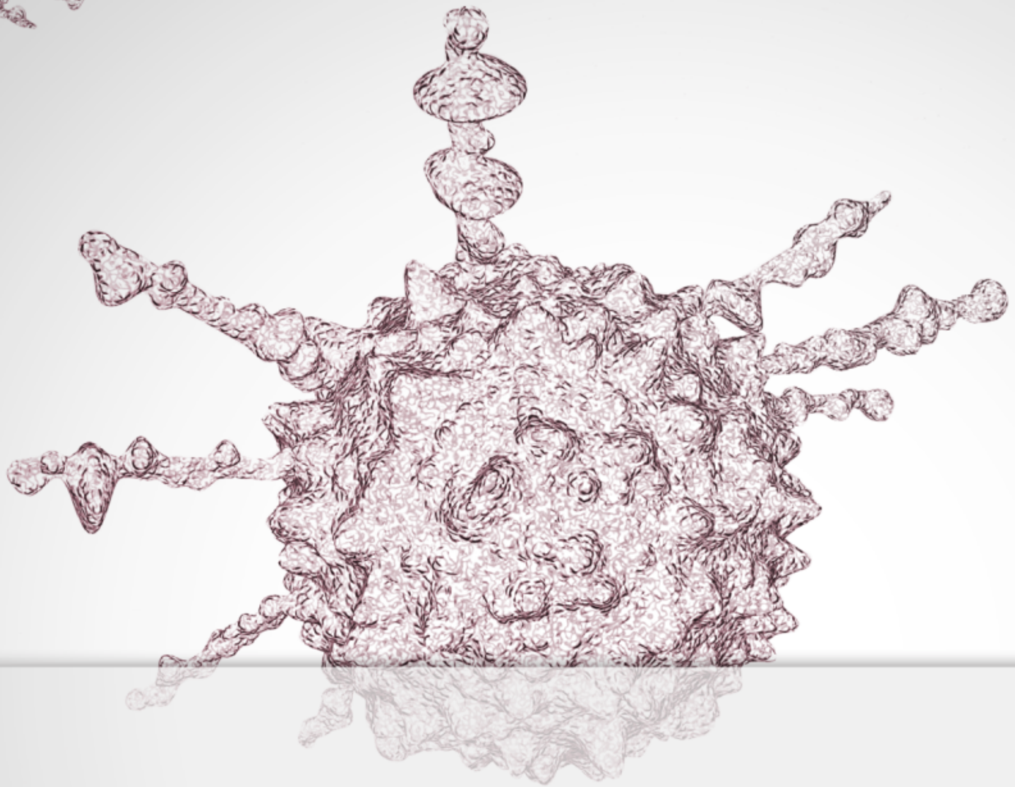
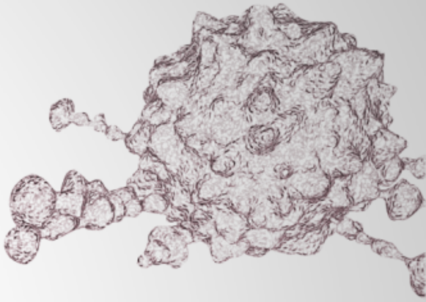






REVEAL THE TRUE BIOLOGY

Name, Date

Assessing and Troubleshooting Data Pre-Alignment Using stsPlots.2.0.R

PACIFIC BIOSCIENCES®

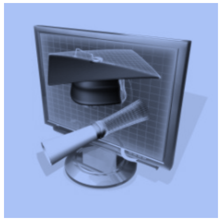
Learning Objectives

Target Audience

Scientists and Bioinformaticians:

- Familiar with bioinformatics concepts
- Interested in learning to evaluate and troubleshoot data from PacBio® *RS* runs.

Skills Learned



After the training, you will be able to:

- Use the sts.csv file to assess RS run quality.
- Distinguish among several possible causes of poor run quality.

Prerequisite

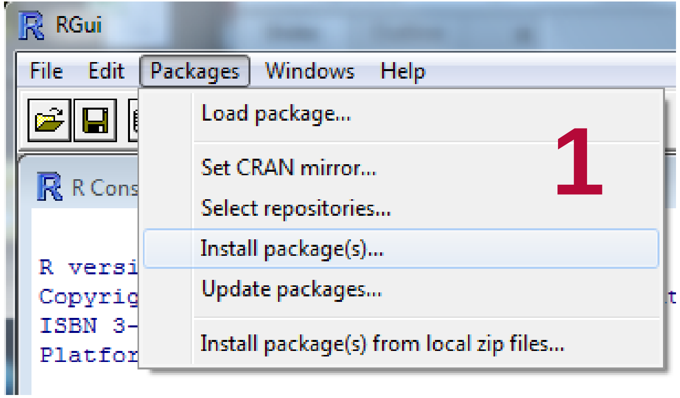
- SMRT Technology
- PacBio *RS* Workflow

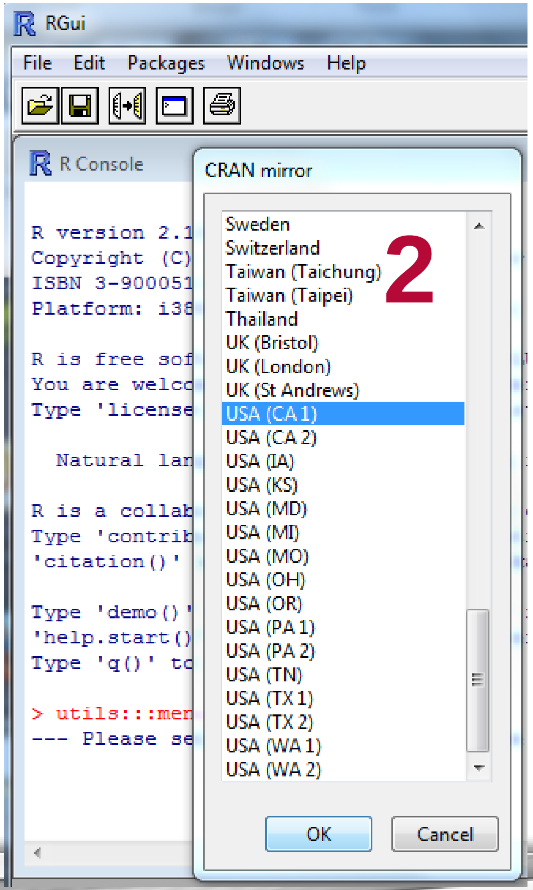
StsPlots Tool: Quickly Evaluate Data Quality After Primary Analysis Finishes.

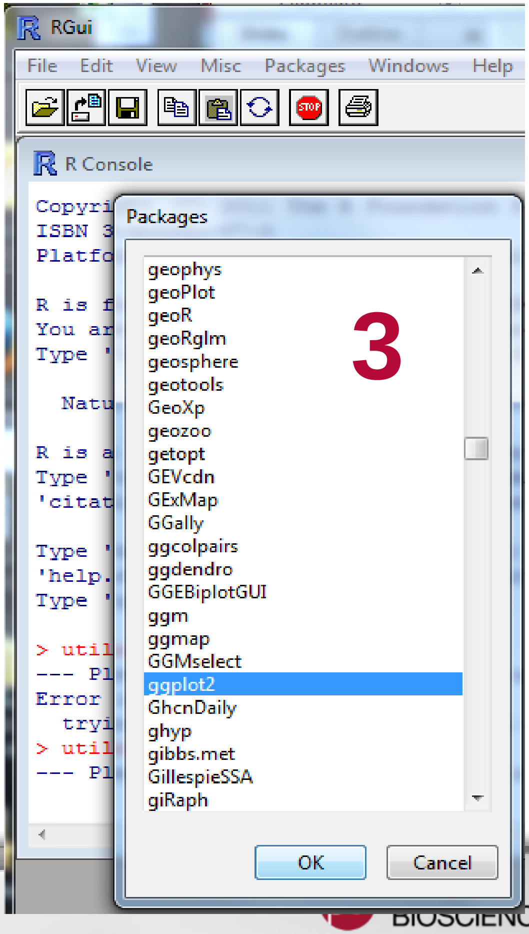
- Information from the sts.csv file, generated after primary analysis completes, is plotted using R. (R is free statistics and graphing software which runs on any platform.)
- The plots allow the user to assess chip loading, readlength, read score, SNR, and oxygen exclusion.
- No reference genome is required.
- No sequence information is accessed.
- The plots render very quickly, allowing for rapid assessment of run quality, making it especially useful for reviewing a loading titration experiment before launching a longer run.

Using StsPlots : Install the Required Software

- Download and install the most recent version of R on your laptop. (2.15.0)
  - http://www.r-project.org/
- Open R and install the packages required to run stsPlots.batch.R (ggplot2, plyr, and reshape2).







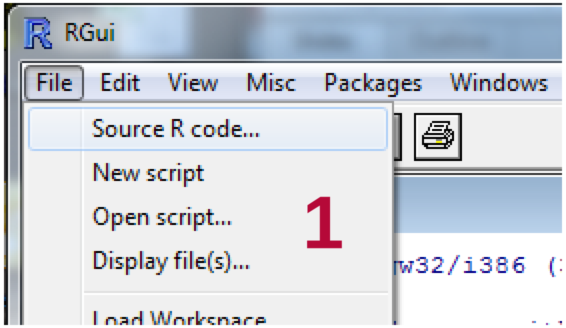
- Get stsPlots.2.0.R

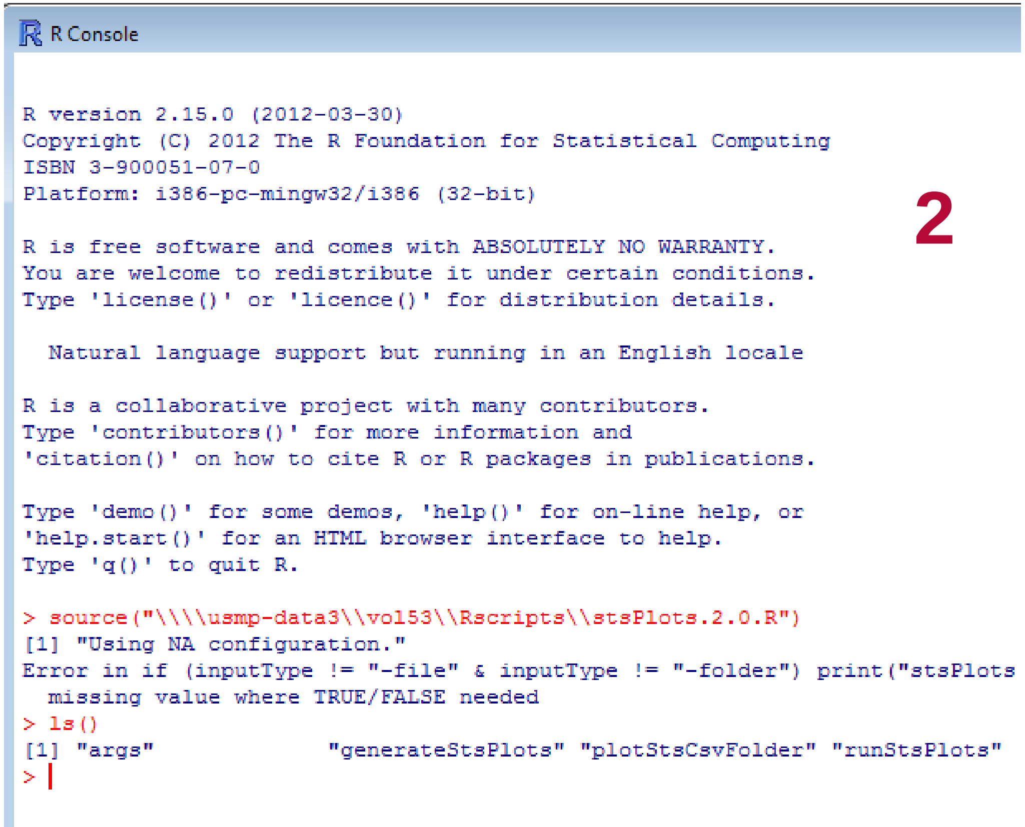
  and put it in a folder under

  My Documents.

Running the Script

- Collect all the sts.csv files to be processed in one folder, also under your My Documents folder.
- In R, source the stsPlots.2.0.R script. Type `ls()` to check that the functions in the script have loaded successfully into your workspace. The output should look like the image below.

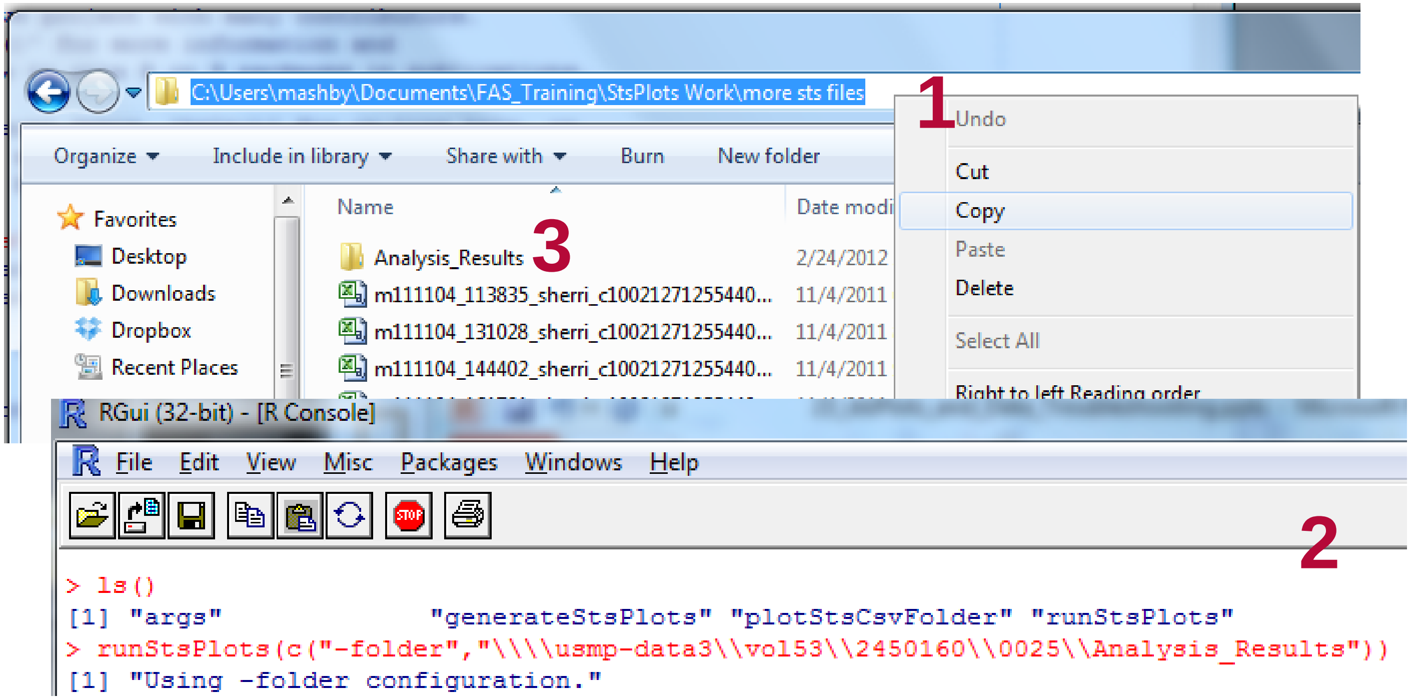




```
R version 2.15.0 (2012-03-30)
Copyright (C) 2012 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
Platform: i386-pc-mingw32/i386 (32-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> source("\\\\usmp-data3\\vol53\\Rscripts\\stsPlots.2.0.R")
[1] "Using NA configuration."
Error in if (inputType != "-file" & inputType != "-folder") print("stsPlots
  missing value where TRUE/FALSE needed
> ls()
[1] "args"             "generateStsPlots" "plotStsCsvFolder" "runStsPlots"
> 
```

Running the script on a folder of sts.csv files from the R interpreter

- Navigate to the folder with the sts.csv files. Click on the navigation bar to reveal the path and copy it.
- In R, use the runStsPlots function with the “-folder” flag to run the analysis for all sts.csv files a folder. Note that after pasting in the copied path, you have to enclose it in quotes and double all the backslashes.



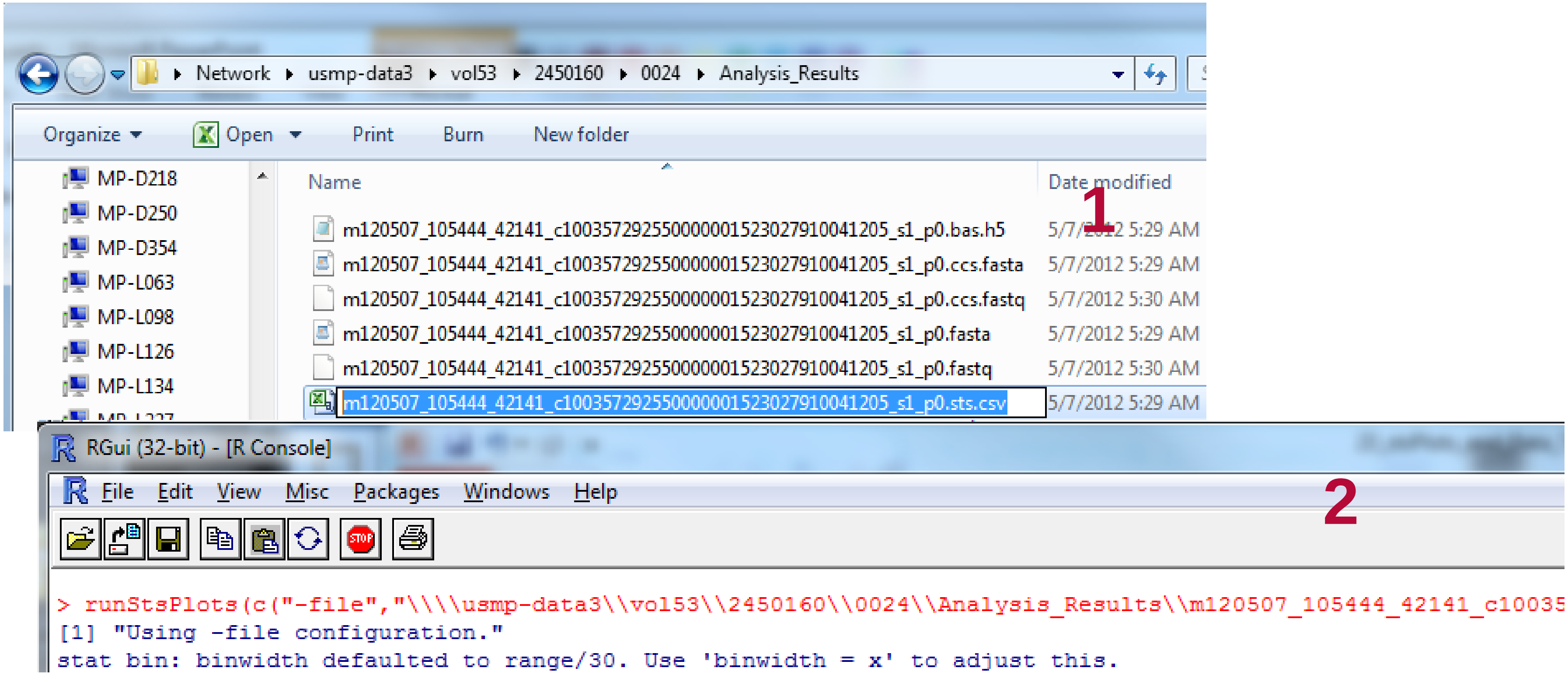
C:\Users\mashby\Documents\FAS_Training\StsPlots Work\more sts files

```
> ls()
[1] "args"              "generateStsPlots" "plotStsCsvFolder" "runStsPlots"
> runStsPlots(c("-folder","\\\\usmp-data3\\vol53\\2450160\\0025\\Analysis_Results"))
[1] "Using -folder configuration."
```

- Output pdfs will appear in a new subfolder named “Analysis_Results”.

Running the script on a single file inside the R interpreter

- In R, use the runStsPlots function with the “-file” flag.
- Navigate to the folder with the sts.csv file. Copy the path as before, enclosing it in quotes and doubling all the backslashes. Also copy the file name and paste that onto the path to complete the filename.

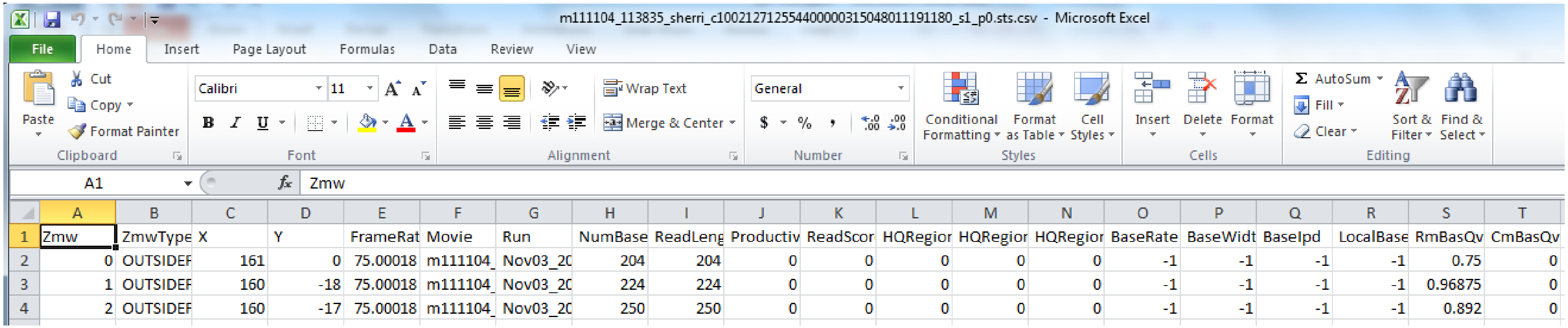


Name	Date modified
m120507_105444_42141_c100357292550000001523027910041205_s1_p0.bas.h5	5/7/2012 5:29 AM
m120507_105444_42141_c100357292550000001523027910041205_s1_p0.ccs.fasta	5/7/2012 5:29 AM
m120507_105444_42141_c100357292550000001523027910041205_s1_p0.ccs.fastq	5/7/2012 5:30 AM
m120507_105444_42141_c100357292550000001523027910041205_s1_p0.fasta	5/7/2012 5:29 AM
m120507_105444_42141_c100357292550000001523027910041205_s1_p0.fastq	5/7/2012 5:30 AM
m120507_105444_42141_c100357292550000001523027910041205_s1_p0.sts.csv	5/7/2012 5:29 AM

```
> runStsPlots(c("-file","\\\\usmp-data3\\vol53\\2450160\\0024\\Analysis_Results\\m120507_105444_42141_c10035
[1] "Using -file configuration."
stat bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
```

- The output pdf will appear the same folder.

For Additional Plotting, Open the sts.csv File in Excel

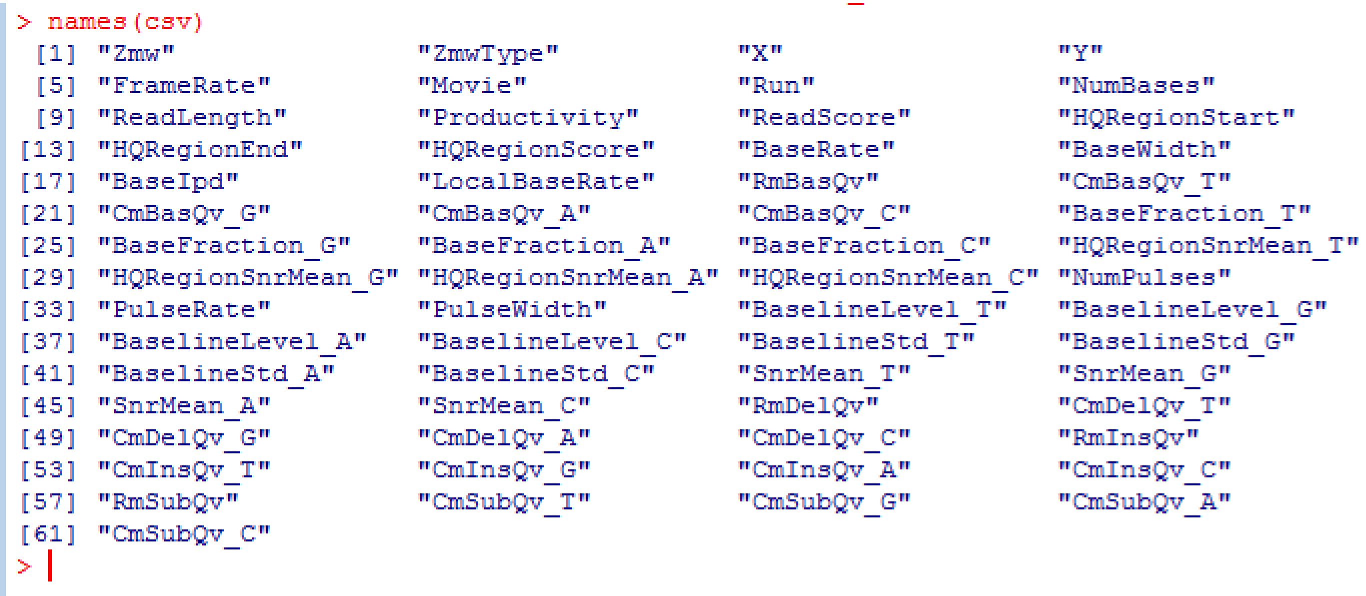


m111104_113835_sherri_c100212712554400000315048011191180_s1_p0.sts.csv - Microsoft Excel

A1 — Zmw

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	Zmw	ZmwType	X	Y	FrameRat	Movie	Run	NumBase	ReadLeng	Productiv	ReadScor	HQRegior	HQRegior	HQRegior	BaseRate	BaseWidt	BaseIpd	LocalBase	RmBasQv	CmBasQv
2	0	OUTSIDEF	161	0	75.00018	m111104_	Nov03_20	204	204	0	0	0	0	0	-1	-1	-1	-1	0.75	0
3	1	OUTSIDEF	160	-18	75.00018	m111104_	Nov03_20	224	224	0	0	0	0	0	-1	-1	-1	-1	0.96875	0
4	2	OUTSIDEF	160	-17	75.00018	m111104_	Nov03_20	250	250	0	0	0	0	0	-1	-1	-1	-1	0.892	0

- Here's all the information available in an sts.csv file (one row per ZMW):



```
> names(csv)
 [1] "Zmw"                "ZmwType"            "X"                  "Y"
 [5] "FrameRate"          "Movie"              "Run"                "NumBases"
 [9] "ReadLength"         "Productivity"       "ReadScore"          "HQRegionStart"
[13] "HQRegionEnd"        "HQRegionScore"      "BaseRate"           "BaseWidth"
[17] "BaseIpd"            "LocalBaseRate"      "RmBasQv"            "CmBasQv_T"
[21] "CmBasQv_G"          "CmBasQv_A"          "CmBasQv_C"          "BaseFraction_T"
[25] "BaseFraction_G"     "BaseFraction_A"     "BaseFraction_C"     "HQRegionSnrMean_T"
[29] "HQRegionSnrMean_G"  "HQRegionSnrMean_A"  "HQRegionSnrMean_C"  "NumPulses"
[33] "PulseRate"          "PulseWidth"         "BaselineLevel_T"    "BaselineLevel_G"
[37] "BaselineLevel_A"    "BaselineLevel_C"    "BaselineStd_T"      "BaselineStd_G"
[41] "BaselineStd_A"      "BaselineStd_C"      "SnrMean_T"          "SnrMean_G"
[45] "SnrMean_A"          "SnrMean_C"          "RmDelQv"            "CmDelQv_T"
[49] "CmDelQv_G"          "CmDelQv_A"          "CmDelQv_C"          "RmInsQv"
[53] "CmInsQv_T"          "CmInsQv_G"          "CmInsQv_A"          "CmInsQv_C"
[57] "RmSubQv"            "CmSubQv_T"          "CmSubQv_G"          "CmSubQv_A"
[61] "CmSubQv_C"
> 
```

Using R to Generate Additional Plots From the sts.csv File

- Alternately, if you are very motivated you can learn to use R's full analysis and graphing capabilities. Here are some links to get you started:
  - http://www.r-project.org/
  - http://www.cyclismo.org/tutorial/R/
  - http://www.ats.ucla.edu/stat/R/faq/
  - http://stat.ethz.ch/R-manual/R-devel/doc/manual/R-lang.html
  - http://had.co.nz/ggplot2/
  - http://svitsrv25.epfl.ch/R-doc/doc/html/search/SearchEngine.html

- Here are some handy R commands to get you started:
  - `df <- read.csv(“C:\\Path\\ToYour\\sts.csv”)`
    Reads your csv file into the workspace as a dataframe.
  - `names(df):` Lists the column names in your dataframe.
  - `levels(factor(df$columnName))`: Gives all unique values in the column
  - `table(df$columnName)`: Gives values in a column and #rows w/each
  - `dim(df)`: Gives nRows and nColumns
  - `head(df)`: Shows the first 6 lines of the dataframe
  - `ls()`: Lists all the objects in your workspace

Using stsPlots From the Command Line

The stsPlots.2.0.R code can be called from the command line using the “Rscript” executable included in both Windows and Linux R installations. Navigate to the folder containing the R script.

```
> “C:\Program Files\R\R-2.15.0\bin\Rscript.exe” --vanilla stsPlots.2.0.R -file
<path to sts.csv file>
```

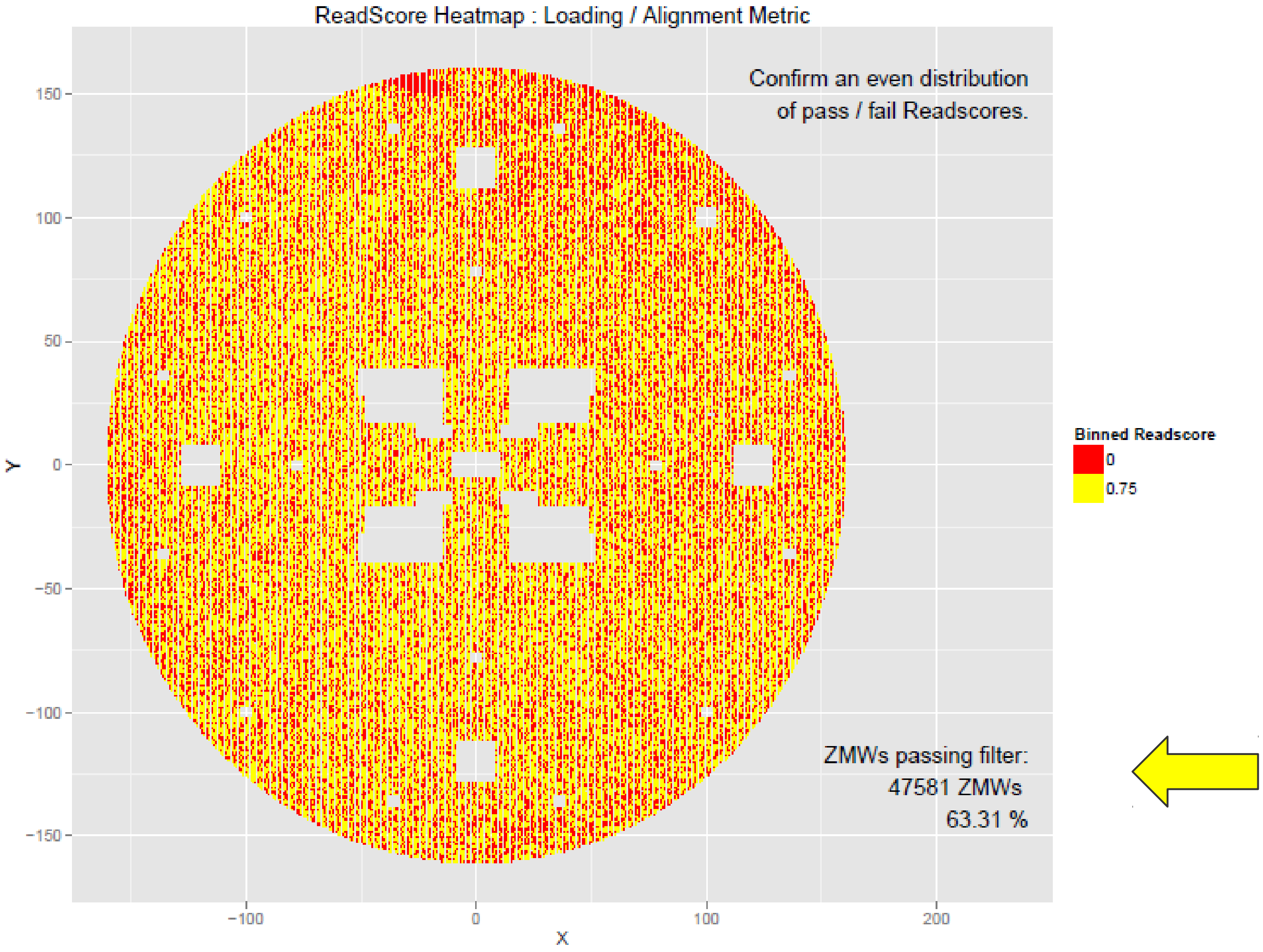
If sts.csv files are dispersed through your Linux based LIMS directory, this is an example of how you might analyze multiple sts.csv files:

```
$ echo /mnt/volume/run/cell/Analysis_Results/*s1_p0.sts.csv | xargs –n 1 Rscript
--vanilla <path to stsPlots.2.0.R> -file
```

The stsPlots.2.0.R script can analyze a folder of sts.csv’s like this:

```
> “C:\Program Files\R\R-2.15.0\bin\Rscript.exe” --vanilla stsPlots.2.0.R -folder
<path to sts.csv folder>
```

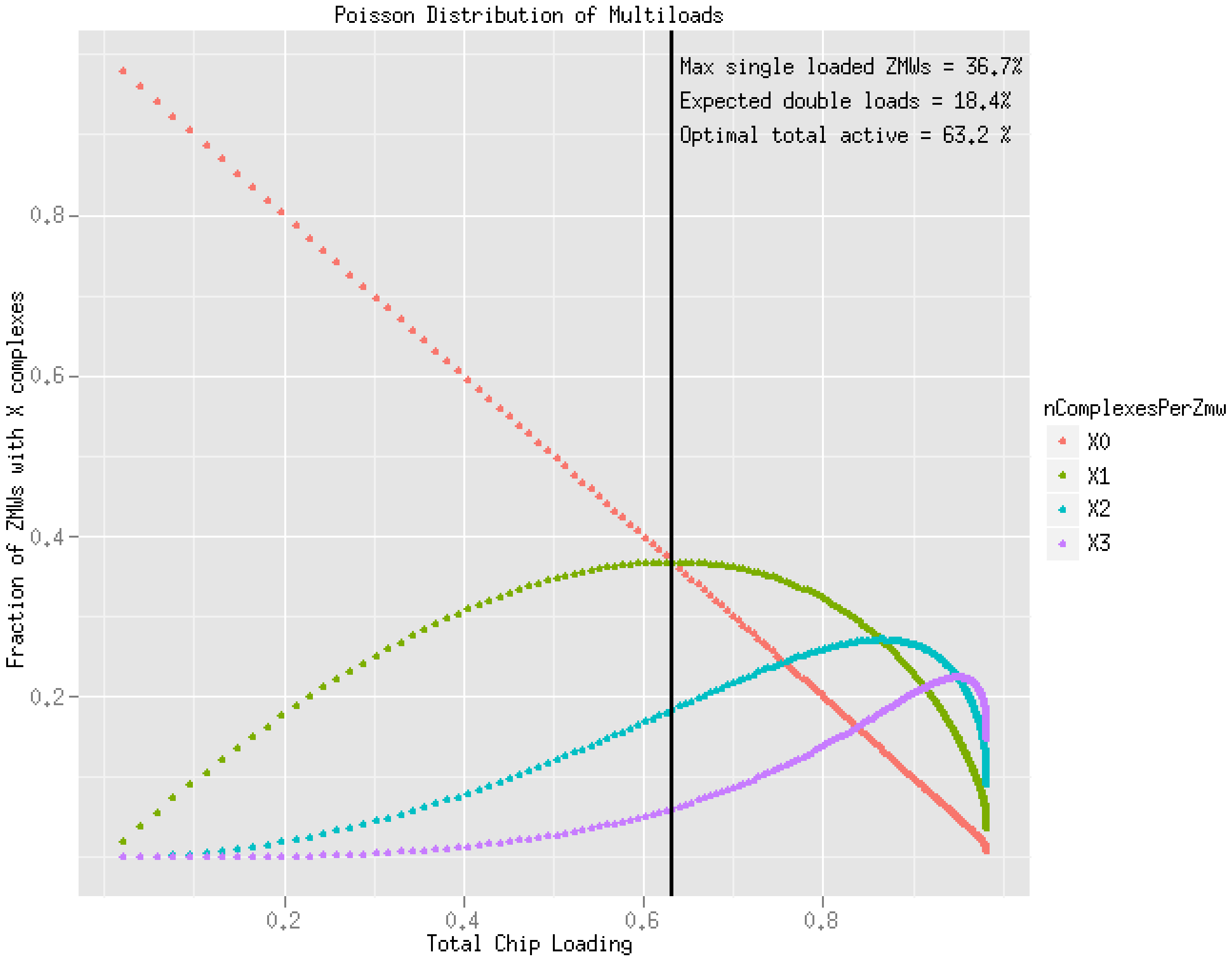
ReadScore Heatmap: Check Chip Alignment and Data Quality



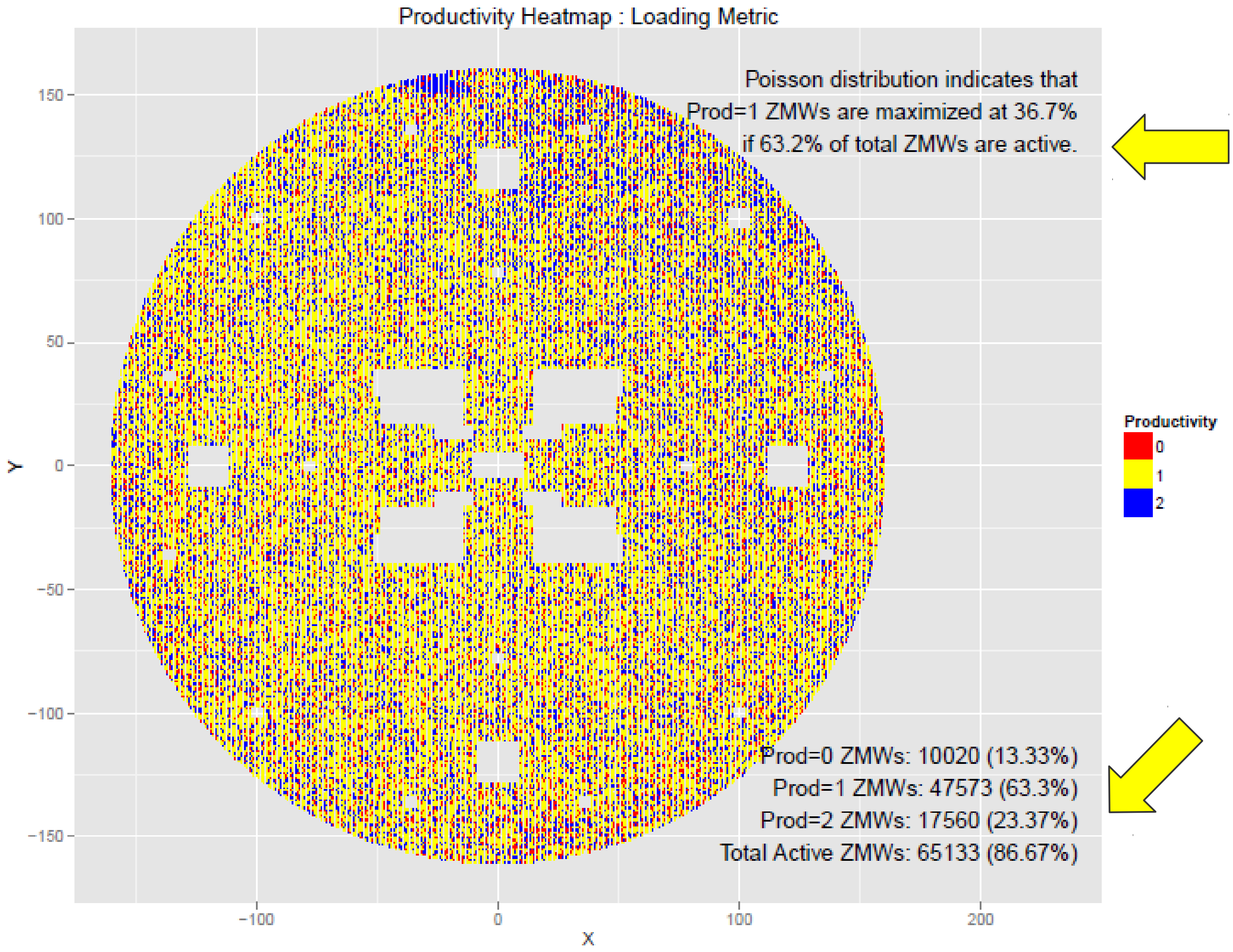
Confirm an even distribution of pass / fail Readscores.

ZMWs passing filter:
47581 ZMWs
63.31 %

Assuming a Poisson Distribution of Loading (Caveat Emptor) Singly Loaded ZMWs are Maximized at 63% Total Active ZMWs



Productivity Heatmap: Check Chip Alignment and Loading

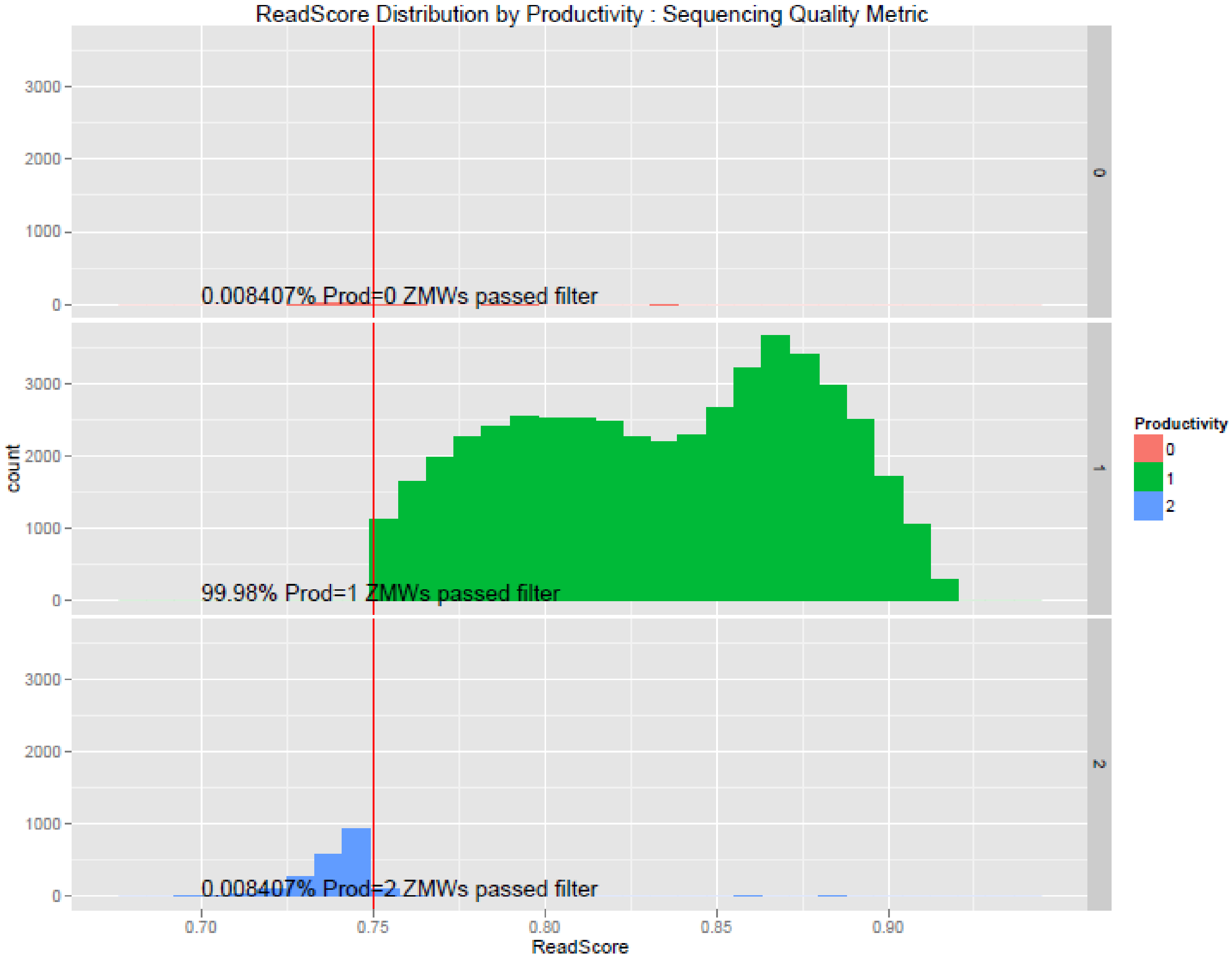


Prod=1 means the ZMW has an HQ region with > half of all bases and a Readscore > 0.75

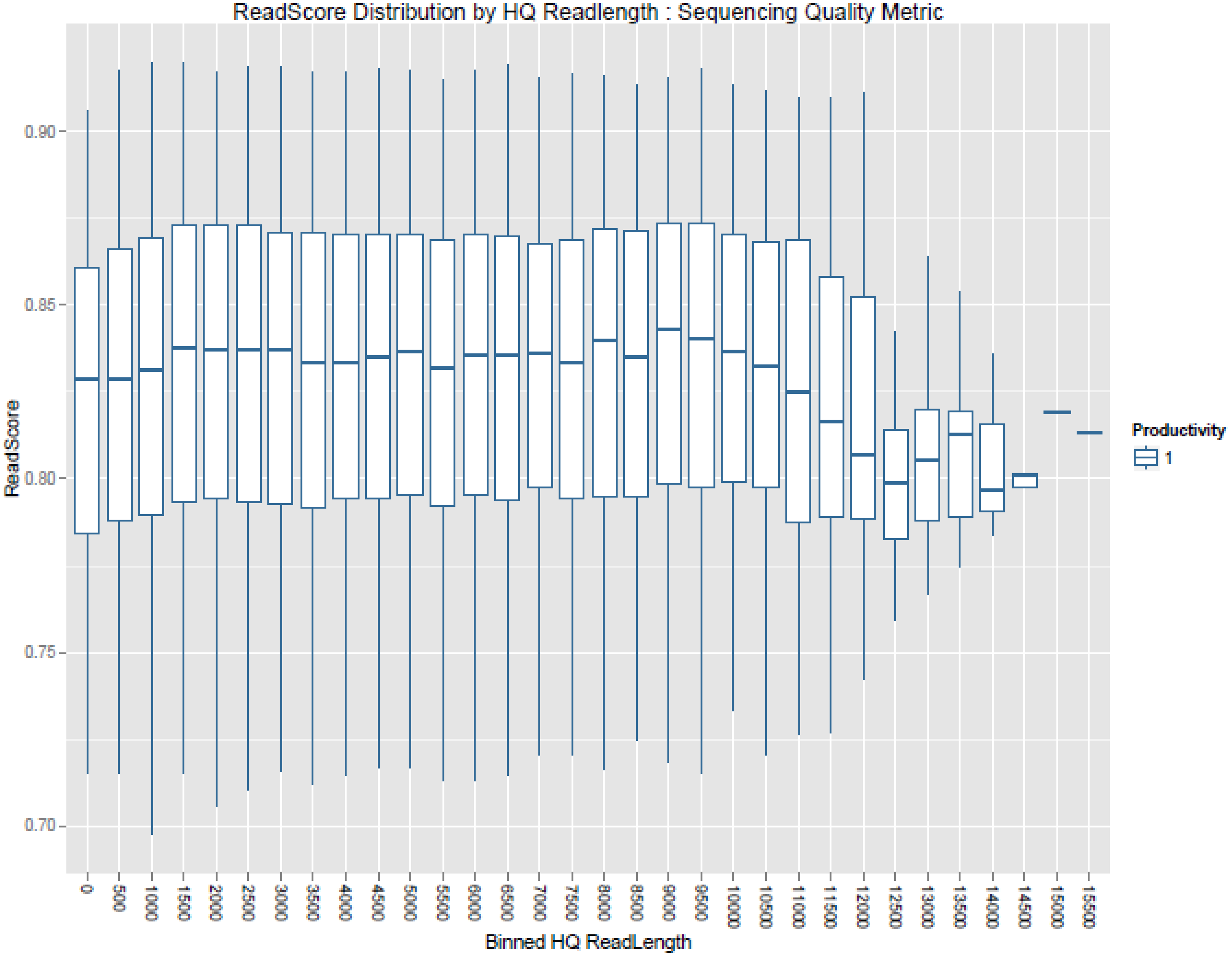
Note that Prod=2 **does not** mean doubly loaded!

It means not P=0 and not P=1.

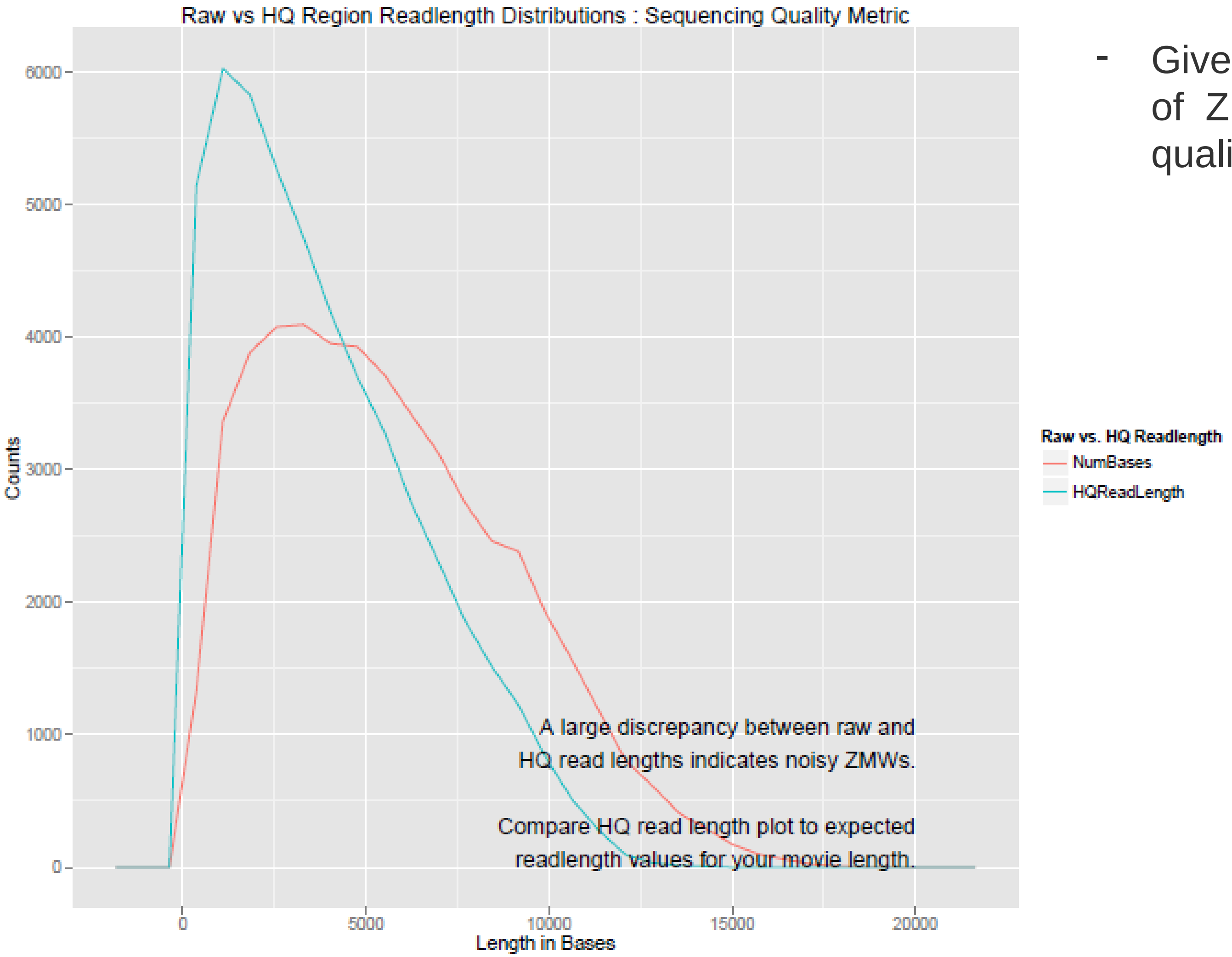
ReadScore Distribution: Check the Read Quality for Each Productivity Segment (Usually Only P=1 Passes Filtering)



ReadScore Distribution by HQ Readlength: Read Quality Should Be Stable Across All Readlength Bins



Raw vs HQ Readlength: Many Reads May Pass Filtering, But How Much are Those Reads Shortened by Filtering?

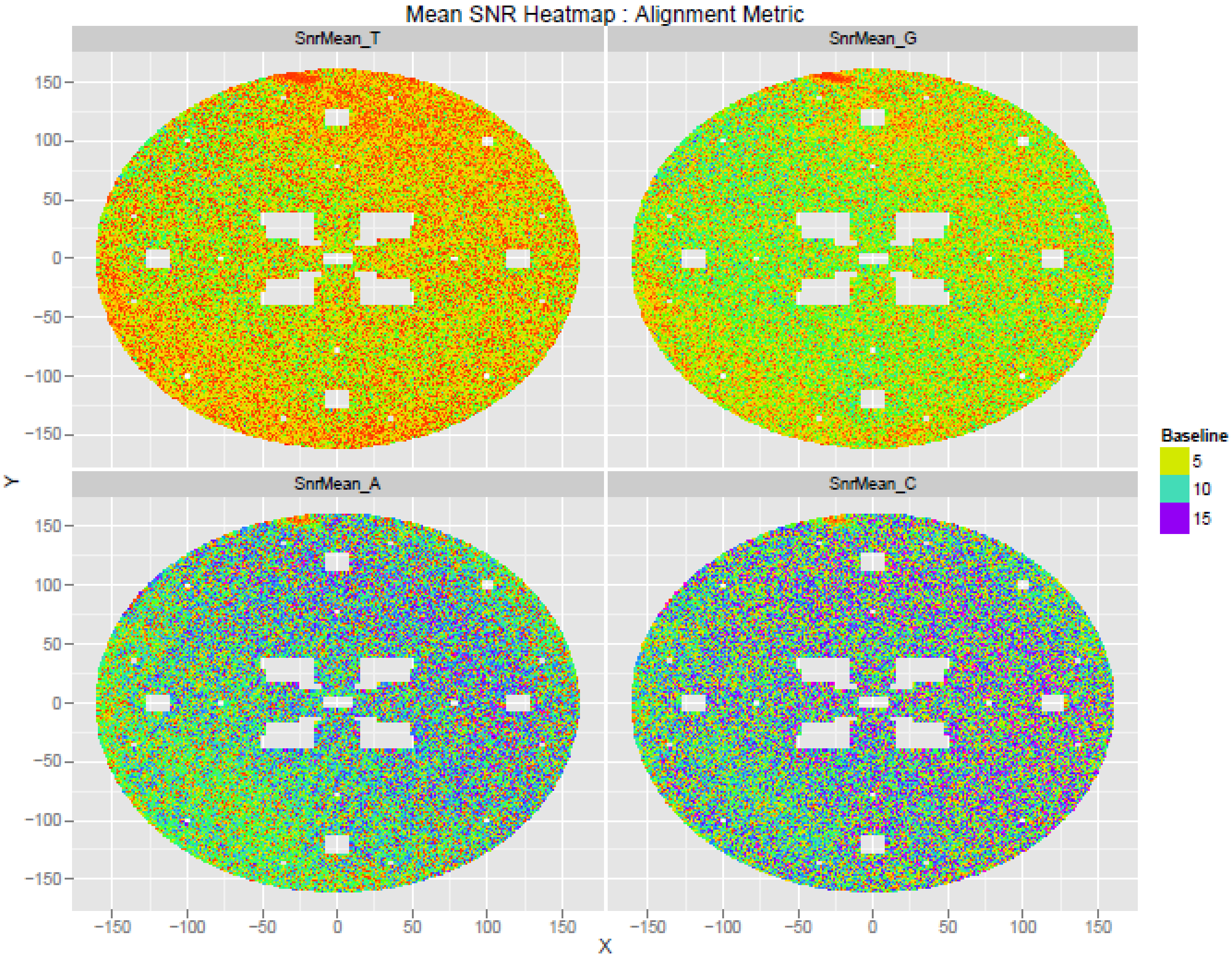


A large discrepancy between raw and HQ read lengths indicates noisy ZMWs.

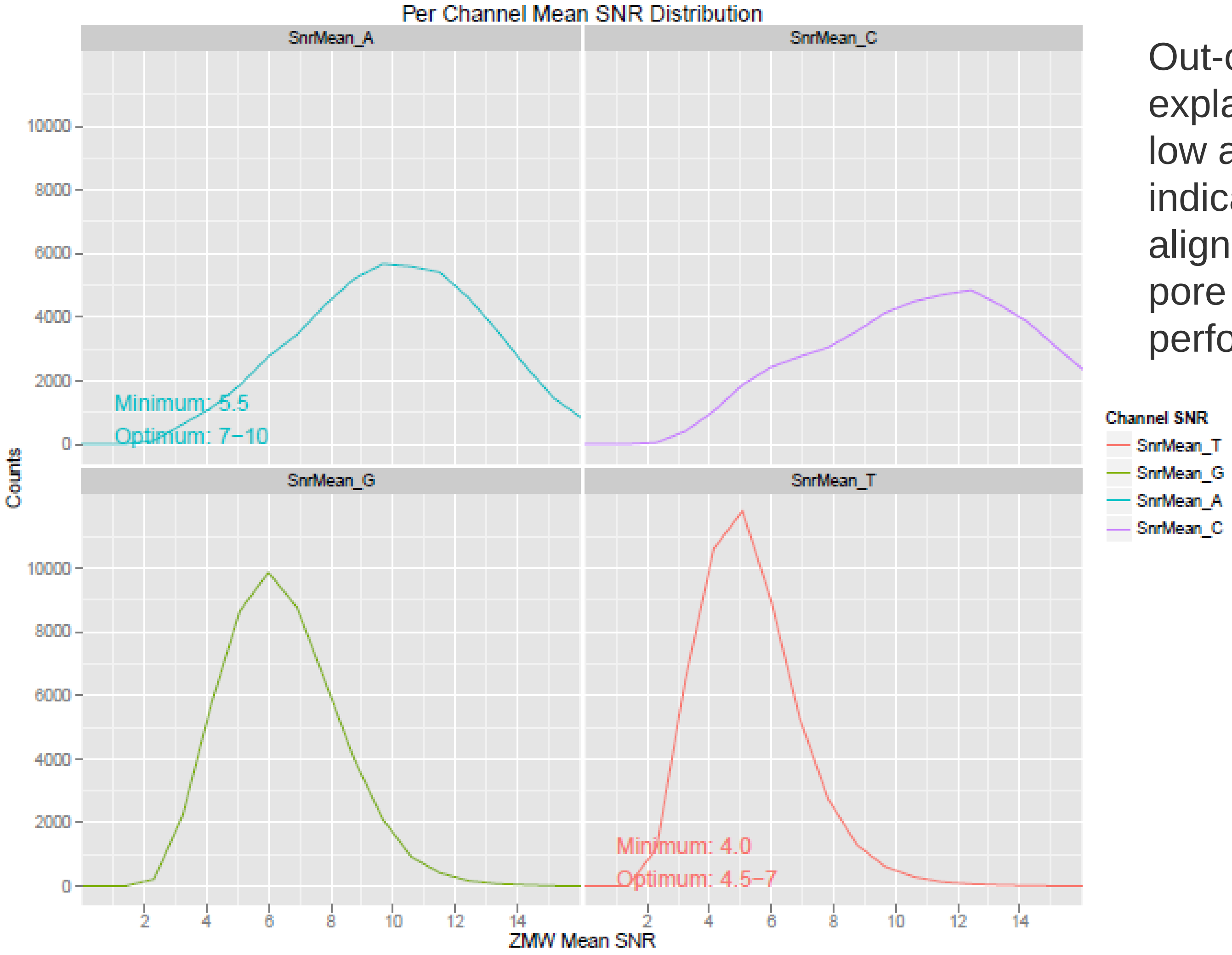
Compare HQ read length plot to expected readlength values for your movie length.

- Gives a sense of how much of ZMW activity is not high quality sequencing

Mean SNR Heatmap: Look for Instrument Issues and Chip Alignment Quality

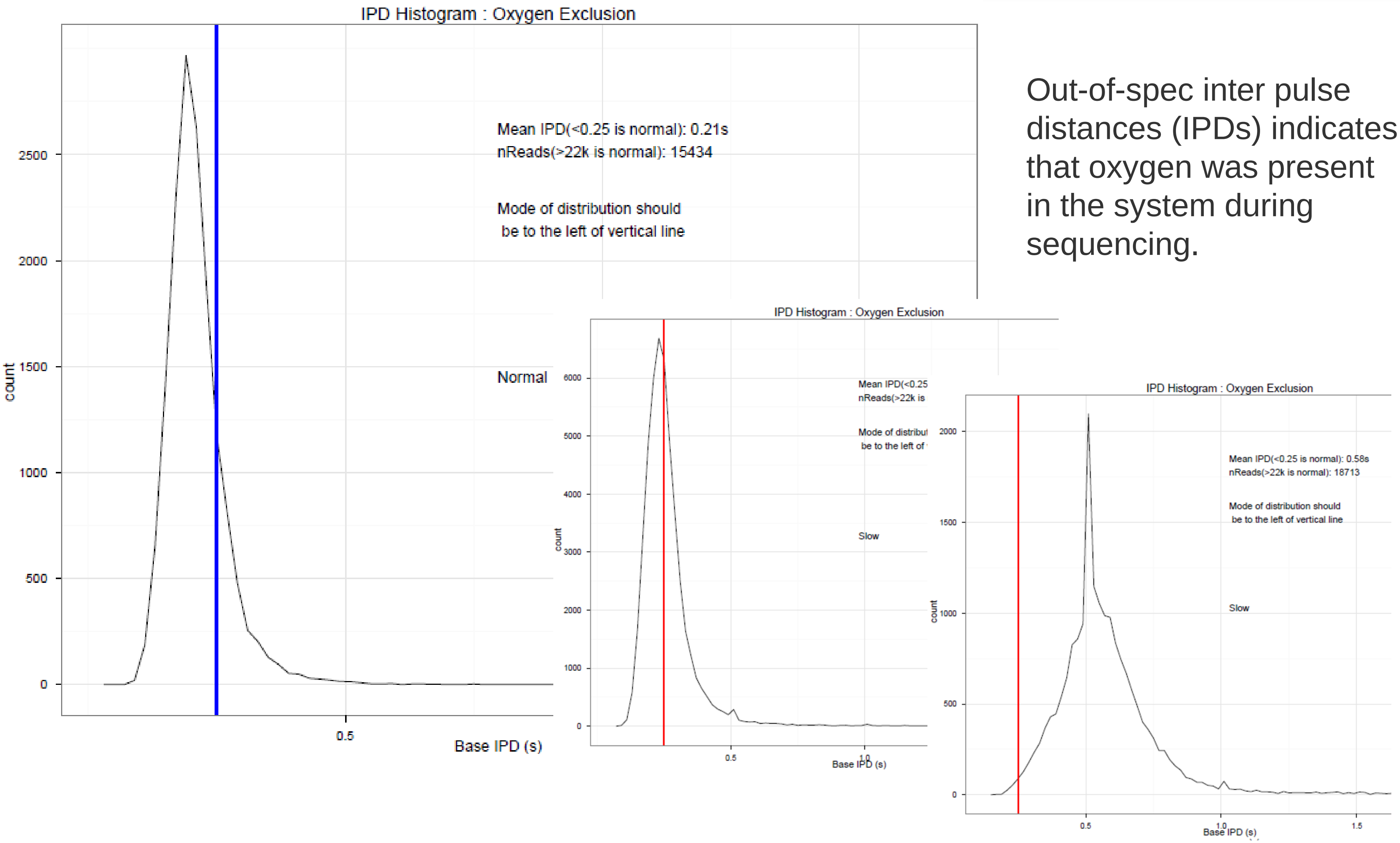


Mean per ZMW SNR Distribution, Including Performance Guidelines From SIV



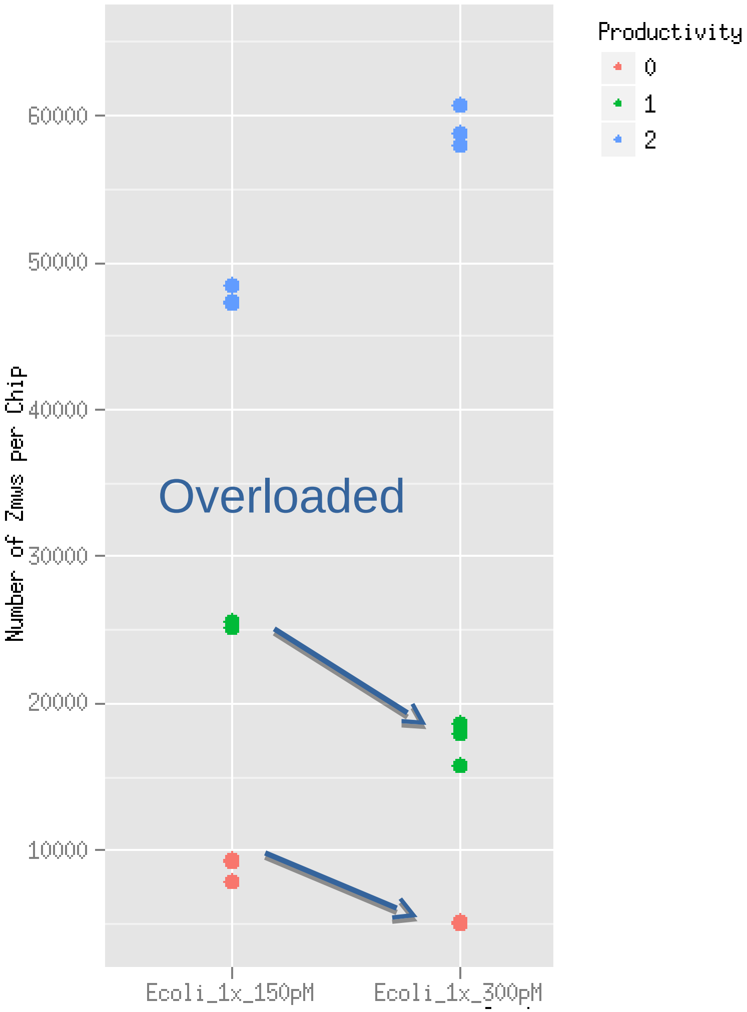
Out-of-spec SNR can explain short readlengths or low accuracy, and can indicate poor chip alignment, out of spec ZMW pore size, or laser performance problems.

IPD Histogram: Oxygen Exclusion



Out-of-spec inter pulse distances (IPDs) indicates that oxygen was present in the system during sequencing.

Distinguishing Between Similar Phenotypes with stsPlots



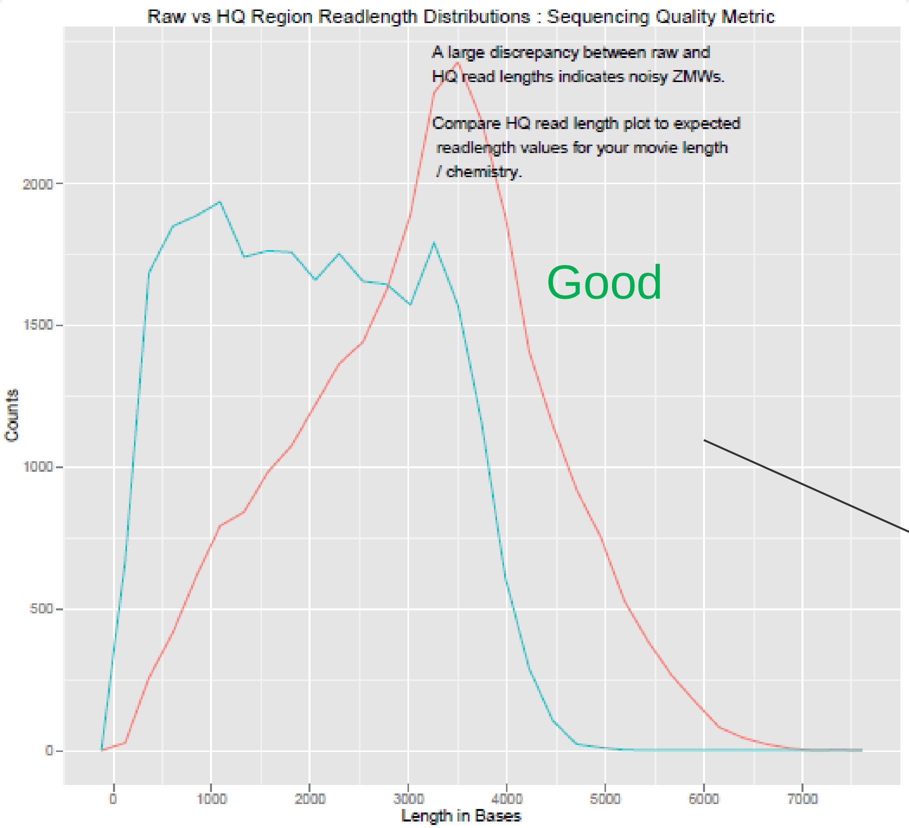
The stsPlots tool can be used to distinguish between different sample prep problems.

High prod=2 does not necessarily mean ‘overloading’

- ZMWs are prod=2 if they have pulse activity and the base rate is significantly higher than for high quality sequencing.
- Free polymerases bound to the bottom of a ZMW generate reads that are binned as prod=2, as they sample dye-linked nucleotides during sequencing.
- When using productivity values to troubleshoot, always focus on the prod=1 and prod=0 counts as well as which is higher.

At left, overloading (300 pM) does increase the number of prod=2 ZMWs, but it is better to focus on low number of prod=1 and prod=0 ZMWs, and the fact that there are more prod=1 than prod=0 ZMWs.

Optimal vs Overloaded Chips (C2 Chemistry)

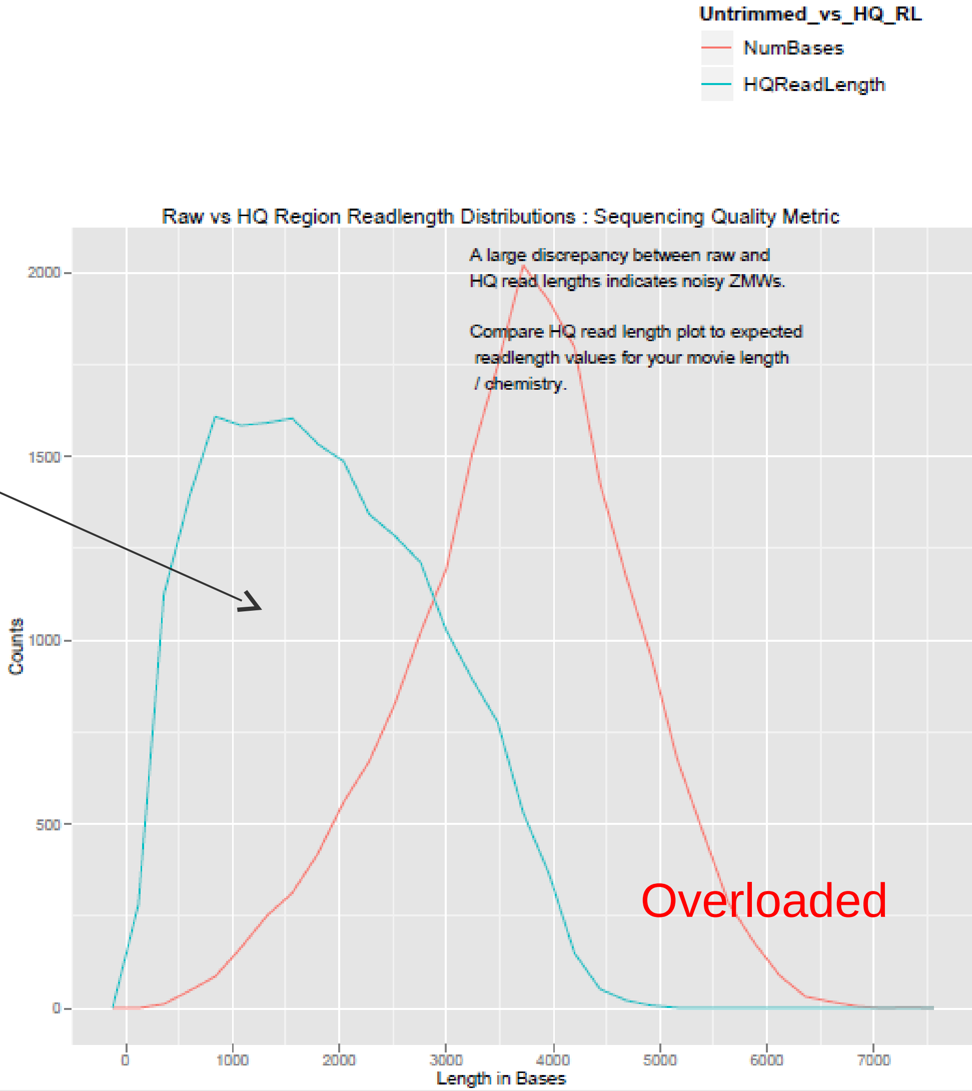


Raw vs HQ Region Readlength Distributions : Sequencing Quality Metric

A large discrepancy between raw and HQ read lengths indicates noisy ZMWs.

Compare HQ read length plot to expected readlength values for your movie length / chemistry.

The discrepancy between raw and HQ readlengths increases as more of the activity in each ZMW is filtered out due to (in this case) overloading.

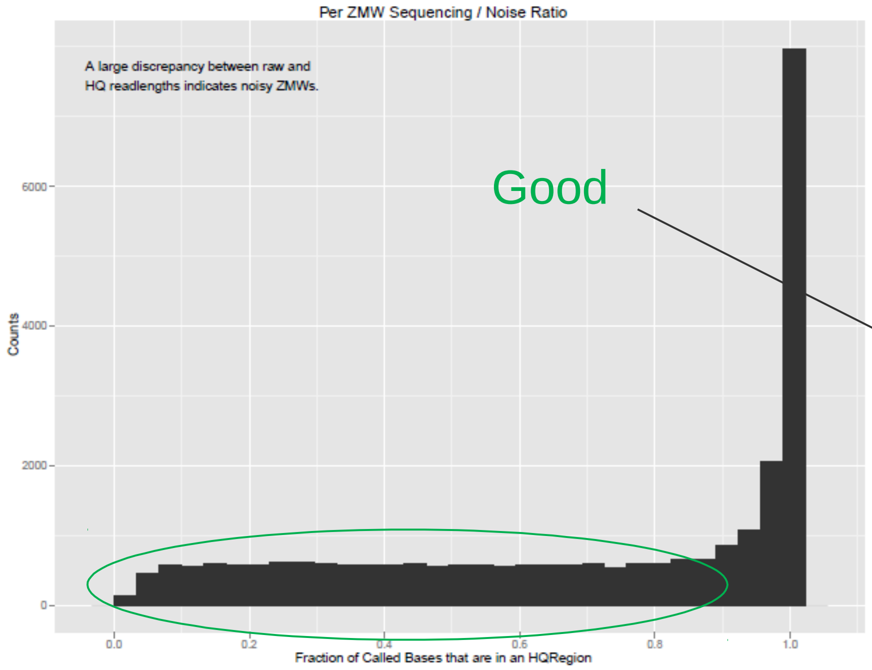


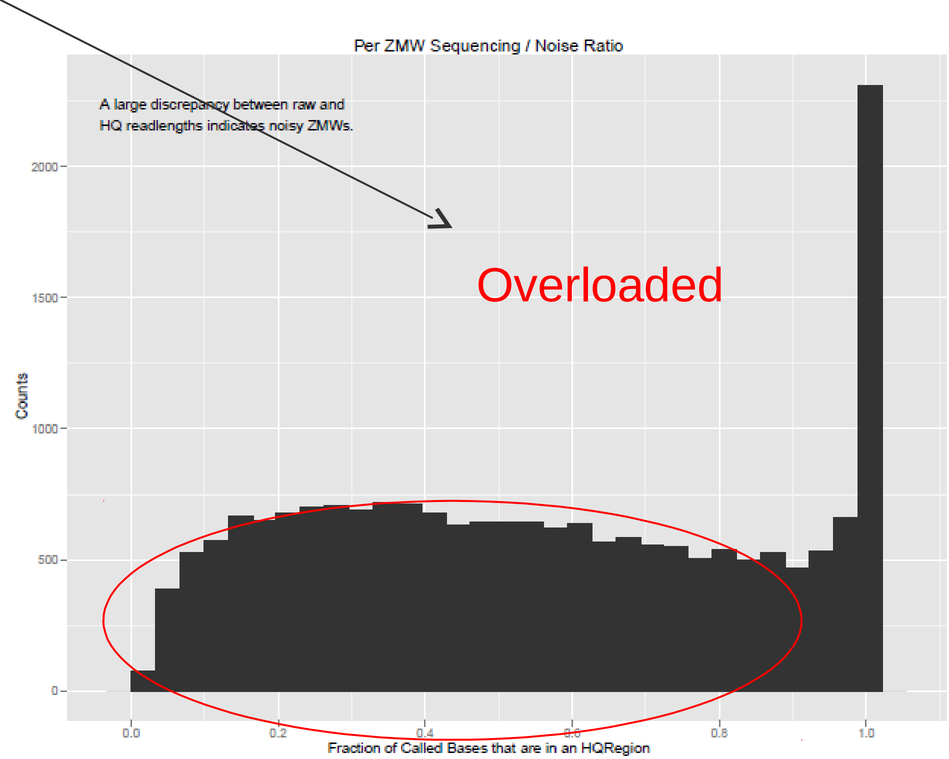
Raw vs HQ Region Readlength Distributions : Sequencing Quality Metric

A large discrepancy between raw and HQ read lengths indicates noisy ZMWs.

Compare HQ read length plot to expected readlength values for your movie length / chemistry.

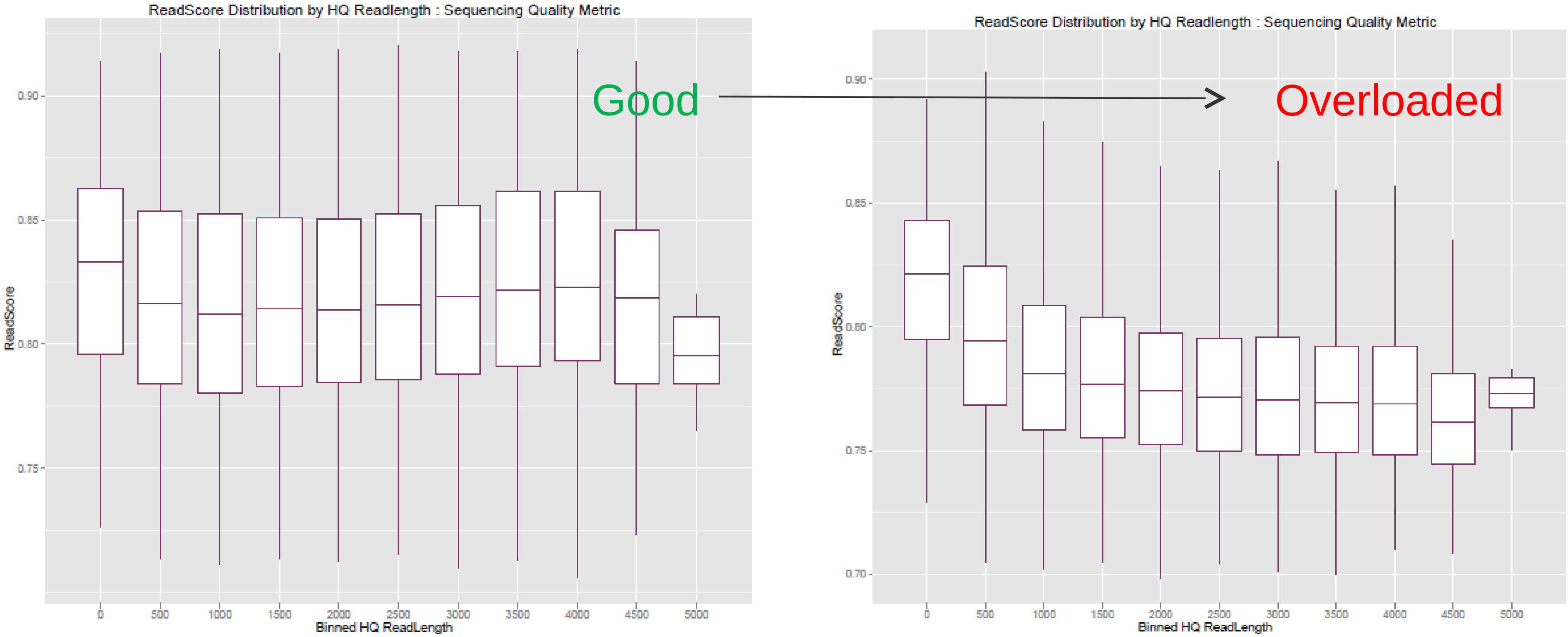
Optimal vs Overloaded Chips (C2 Chemistry)





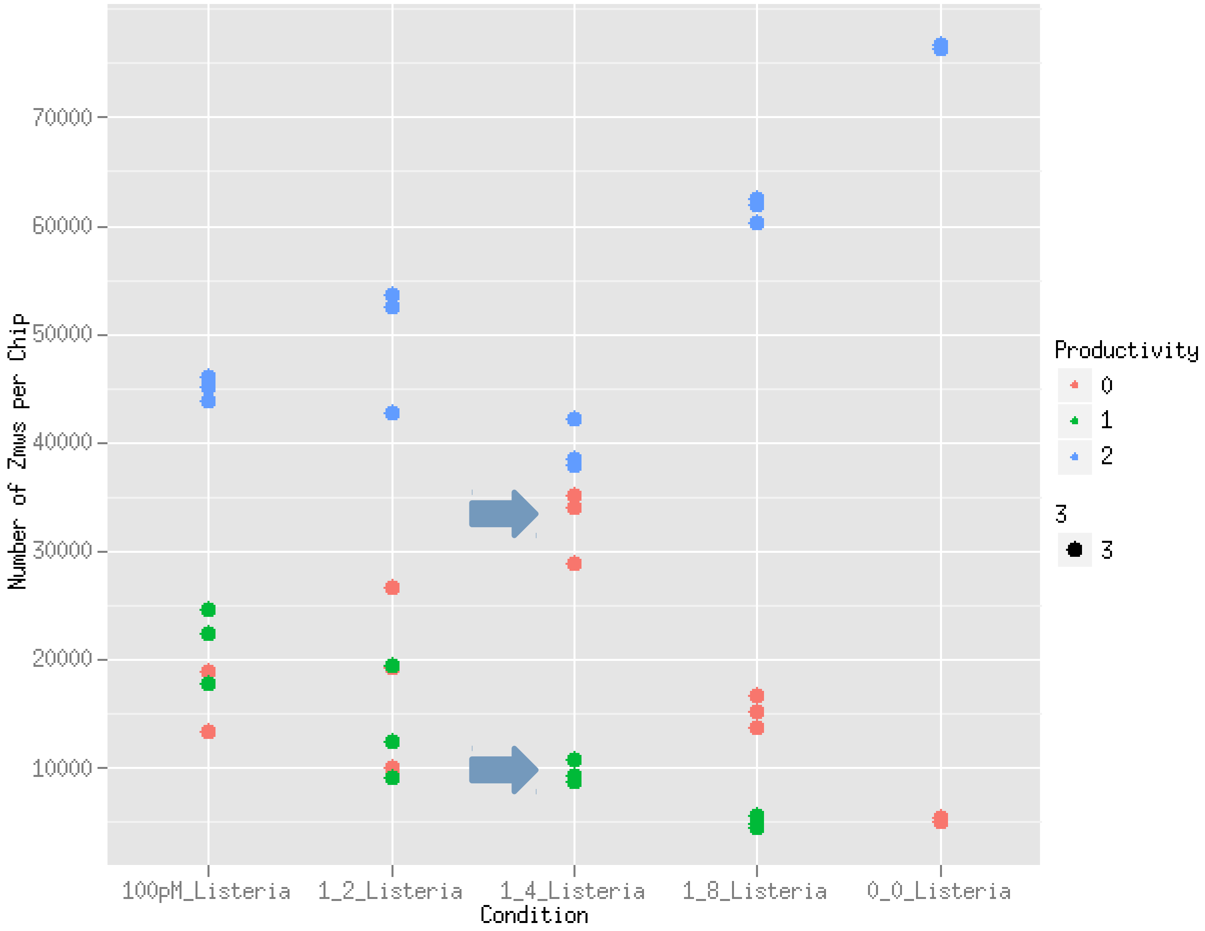
More and more ZMWs have a low proportion of called bases located inside HQ regions due to (in this case) overloading.

Optimal vs Overloaded Chips (C2 Chemistry)



Readscore drops for overloaded chips, but is often higher in the shortest readlength bins because greatly shortened HQ regions have captured parts of the trace where only one polymerase was active.

Another Common Phenotype: Excess Free Polymerase

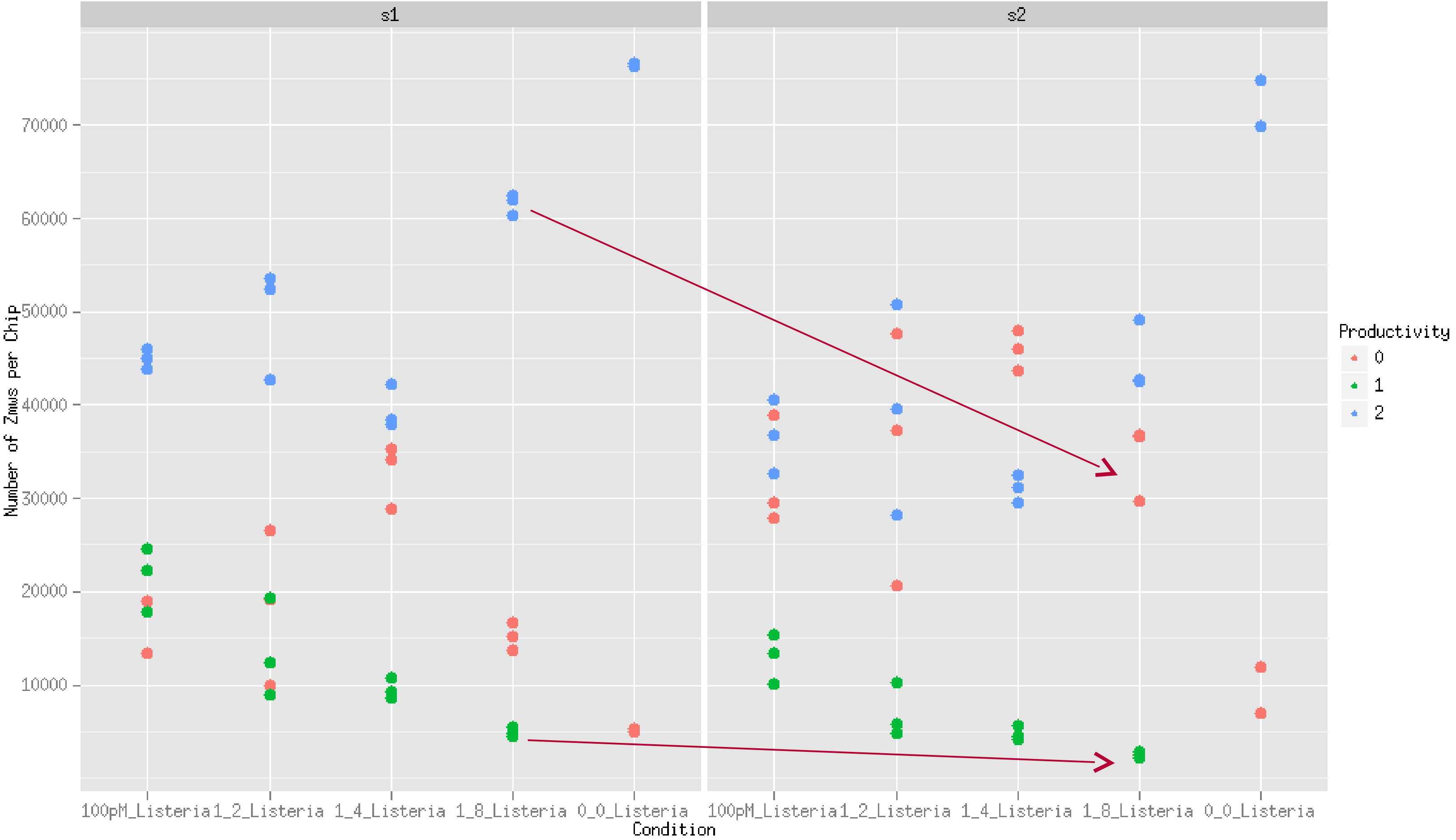


A series of binding tubes were made with the same amount of polymerase but decreasing amounts of SMRTbell, all the way down to no DNA. (6 kb library)

As a result, the samples have increasing amounts of free DNA polymerase from left to right.

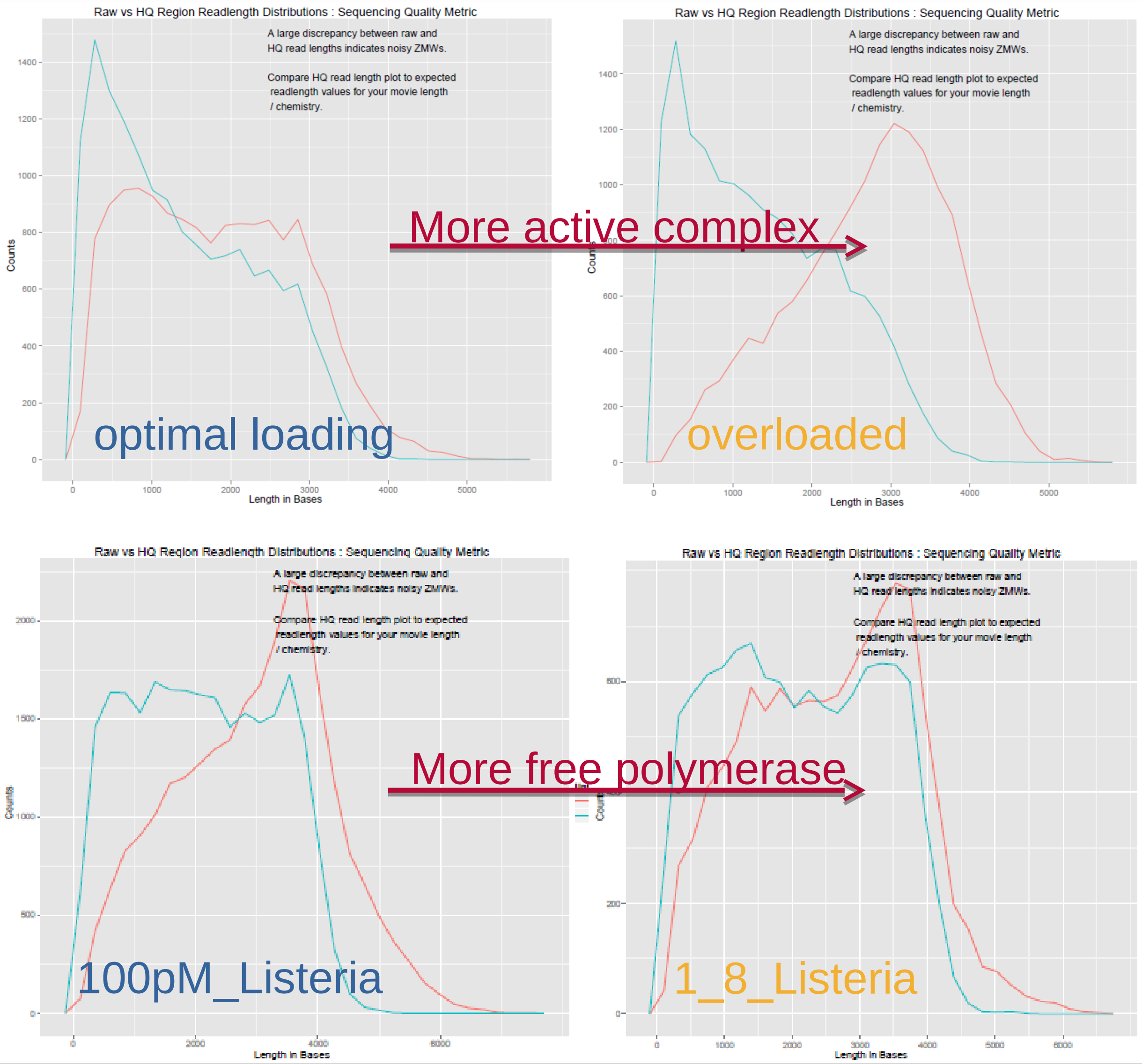
Just like overloading, free polymerase gives a high prod=2 phenotype. It is more useful to note that excessive free polymerase leads to very few prod=1 ZMWs, and that the prod=0 count matches or exceeds the prod=1 count.

Comparison of Set1 and Set2 with Excess Free Polymerase



In the second set, prod=1 ZMWs do not increase, as they would with overloading: when the free polymerases die out, these wells turn from prod=2 to prod=0, not to prod=1. Unlike with true overloading, there is no active complex in these prod=2 ZMWs.

The Phenotypes Can be Distinguished Using Additional Plots

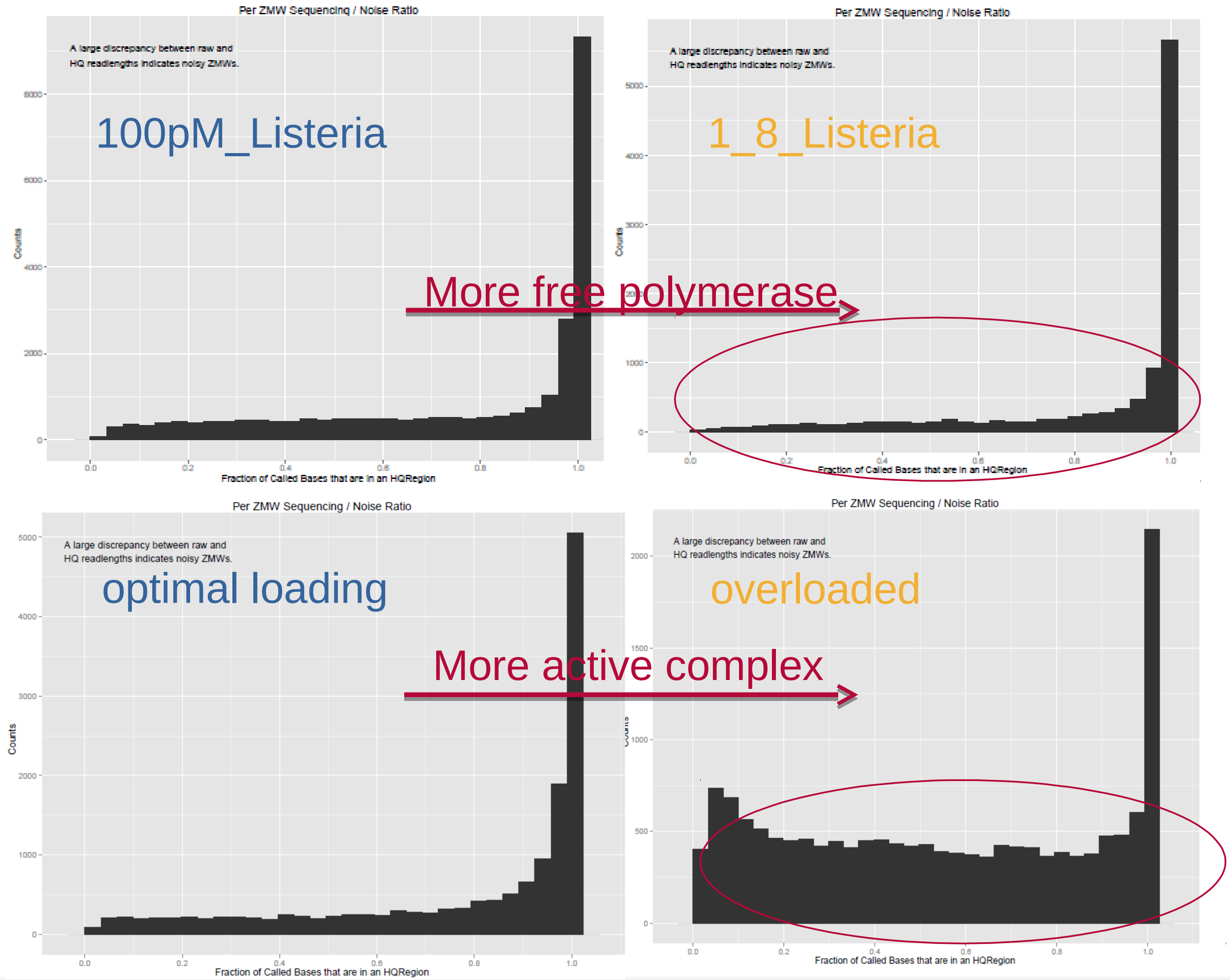


Overloading with active complex leads to the trimming of reads to exclude low quality regions; excess free polymerase does not.

Free polymerase creates random single pulses along the whole trace as the enzyme samples free analogs.

Multiple sequencing complexes result in continuous, excludable sections of low quality, which end when one complex stops either from photodamage, dissociation, or pausing.

A Related Plot Also Helps Distinguish the Two Phenotypes



100pM_Listeria

1_8_Listeria

More free polymerase

optimal loading

overloaded

More active complex

Overloading with active complex leads to a lower proportion of called bases inside HQ regions; excess free polymerase does not.

Note that overloading can be fixed by reducing the concentration of complex on the plate;

Excess free polymerase cannot!

Phenotypes and Possible Root Causes

- High Prod=2 and low Prod=1
- Prod=0 <= Prod=1
- Large discrepancy in RL distribution between Called Bases and HQRegion
- Significant proportion of called bases are outside of HQ region

- Overloading
  - Reduce the concentration of complex on the plate.

- High Prod=2 and low Prod=1
- Prod=0 >= Prod=1
- Little shift in RL distribution between Called Bases and HQRegion
- Nearly all called bases are in HQRegion

- Under quantitation of DNA leads to too much polymerase in the binding step
  - Make a new binding tube.

Questions About or Problems with stsPlots

- Email Customer Support
- Send
  - The exact command you used to execute stsPlots.batch.R (copy/paste or screen shot preferred)
  - The error message output onto your screen
  - The sts.csv file that failed

Summary of Key Points

- Key Points
  - The stsPlots tool allows for rapid assessment of RS run quality.
  - The generated plots allow the user to diagnose:
    - sample overloading
    - excess free polymerase in the sample
    - poor oxygen exclusion during the run
    - poor chip alignment

- Where to Find More Information
  - The most up to date stsPlots scripts can be downloaded from GitHub.
  - https://github.com/PacificBiosciences