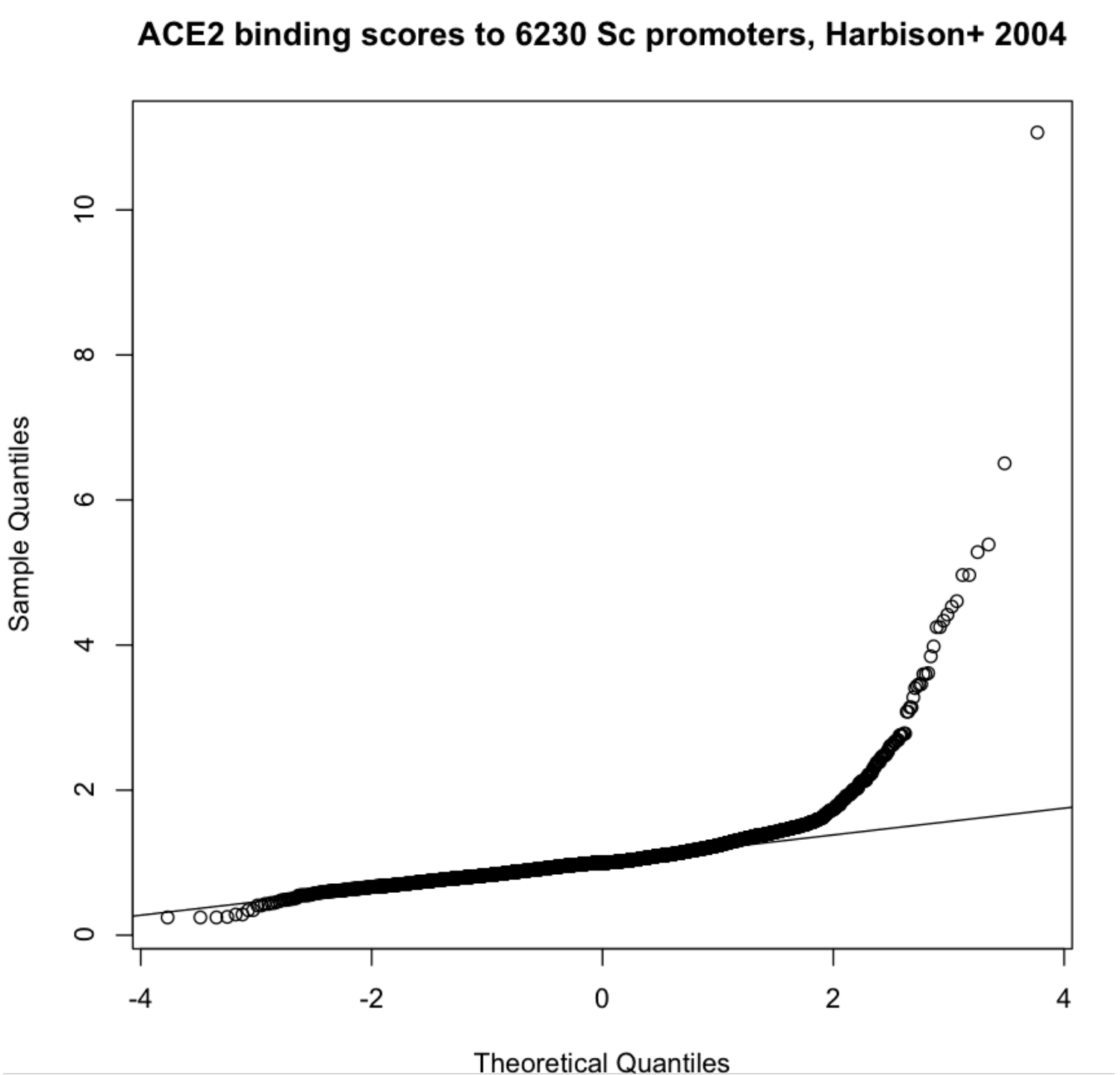
Visualization in statistical genomics

VJ Carey

CSAMA Brixen 2014


SPES ET PATIENTIA VINCVNT.
Ganz Schiesen gegeben den 25 April Anno 98. Von Johan Christoph Kirchberger. 3 der Schützen Meister.


ACE2 binding scores to 6230 Sc promoters, Harbison+ 2004
Sample Quantiles
0
2
4
6
8
10
-4
-2
0
2
4
Theoretical Quantiles

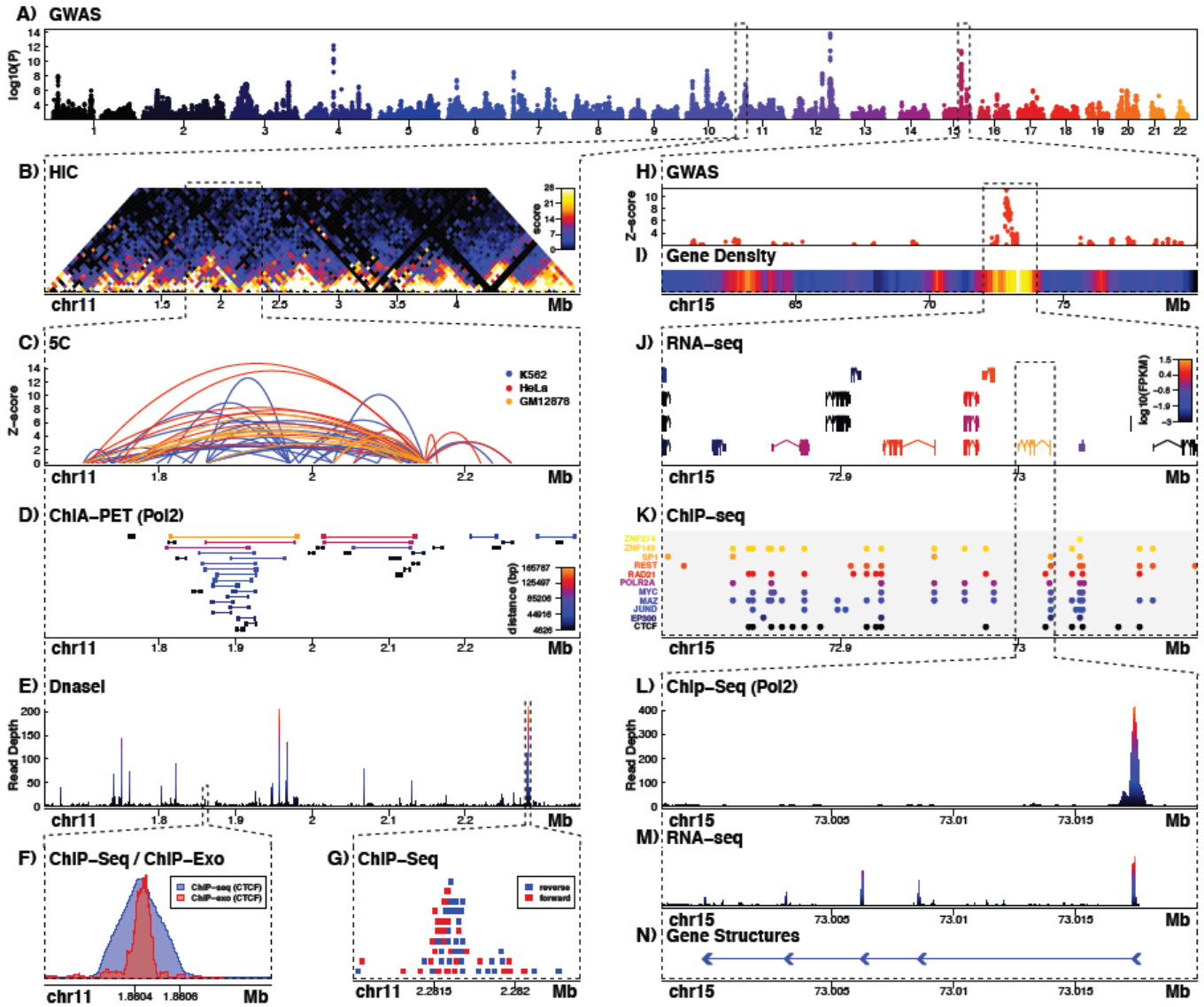
Open source software defining the model giving rise to the line

```
 qqline
unction (y, datax = FALSE, distribution = qnorm, probs = c(0.25,
   0.75), qtype = 7, ...)

   stopifnot(length(probs) == 2, is.function(distribution))
   y <- quantile(y, probs, names = FALSE, type = qtype, na.rm = TRUE)
   x <- distribution(probs)
   if (datax) {
       slope <- diff(x)/diff(y)
       int <- x[1L] - slope * y[1L]
   }
   else {
       slope <- diff(y)/diff(x)
       int <- y[1L] - slope * x[1L]
   }
   abline(int, slope, ...)

ytecode: 0x1283b1ff8>
```

Upshot

- A simple univariate visualization tool can have considerable flexibility
  - Choice of reference distribution
  - Quantiles to use to select a ‘nearby’ member of the reference family
- We thus avoid having different interfaces for related but nonidentical applications

Exemplar, very new: from the Sushi package:



Upshots

- Cascading, zooming scans of diverse related assays are evidently of interest for integrative analysis
- Query: can we do more than ‘gawk’ at such a busy display?
- Query: can we change focus to other regions, develop hints for where to go?

I will harp on the following notion: If you can make a display for a given gene (feature), try to program so that it can be made for ANY gene (feature)

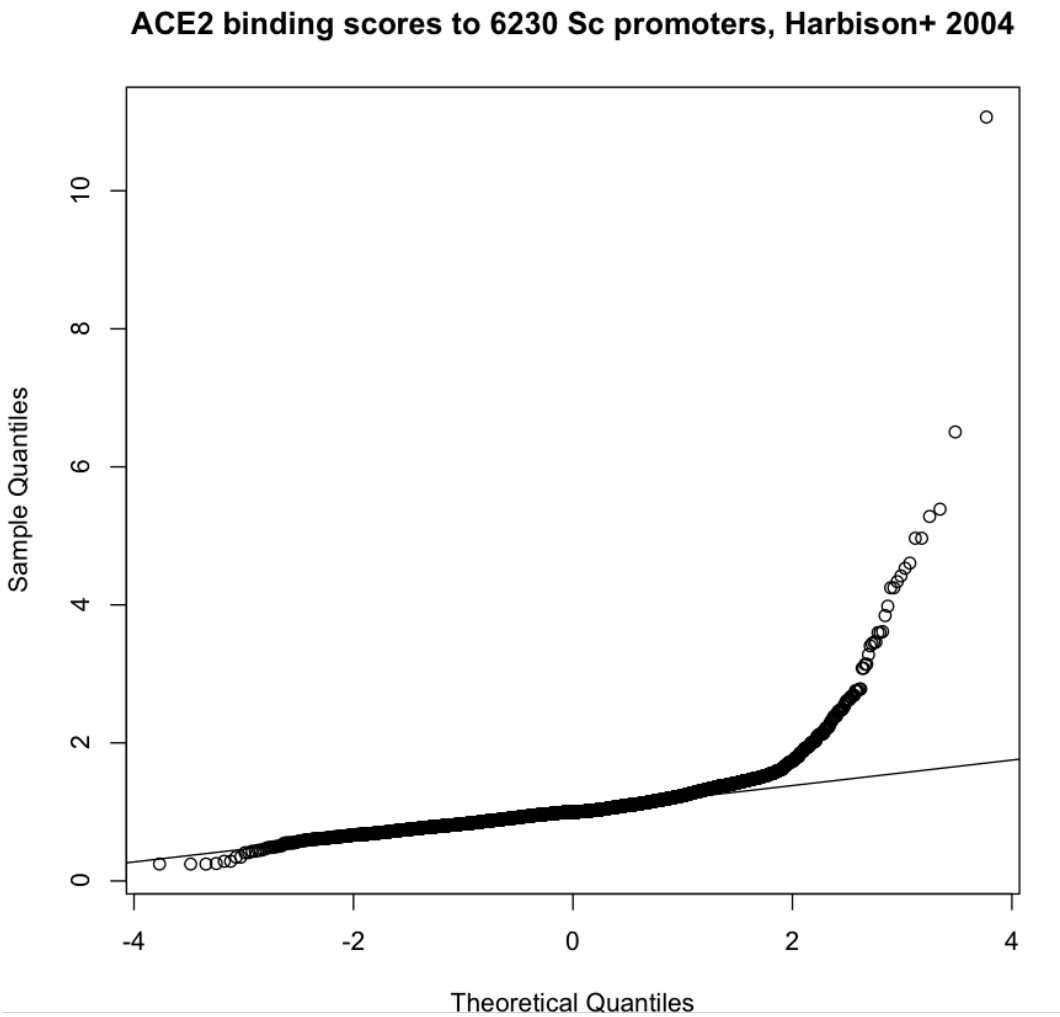
Overview

- Purposes of statistical and genomic visualization:
  - General sanity checks, rapid surveys
  - Criticism, refutation
  - Intimation of differences, relationships
- Well-executed visualizations may speed discovery or help to avoid dead ends
- **Interactive visualizations** are now quite feasible but must be well-designed

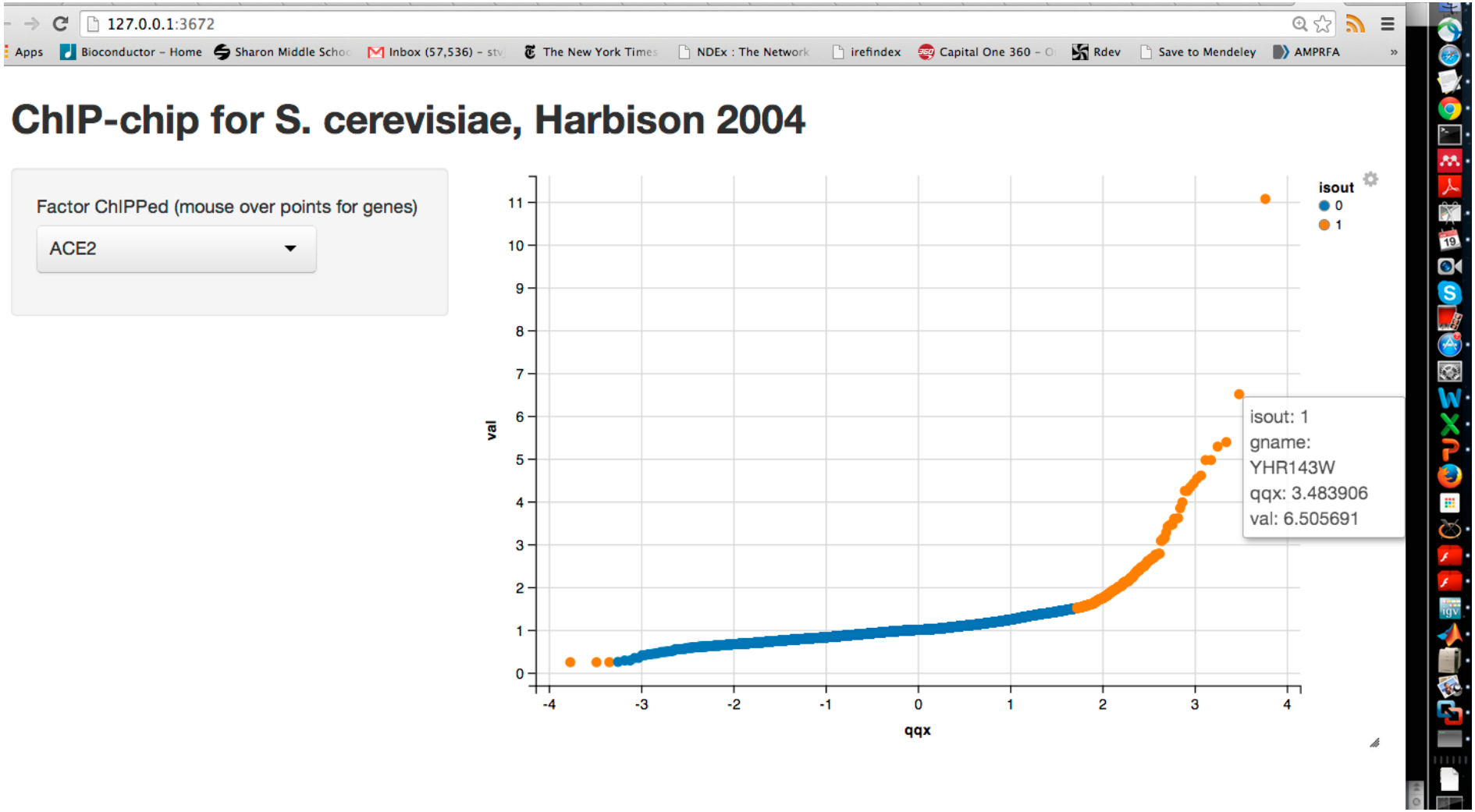
Three key concepts for all statistical analysis efforts

- Relation to population sampling or selection
  - What is the underlying source/process of data generation? Limitation of observation implies uncertainty of estimation/interpretation: Is this exposed?
- Model structure and assumptions
  - Has the model even been stated? Algebraic form? Structural and random elements?
- Model comparison and improvement
  - All models are false, so preferences emerge from comparisons

What sort of assumptions do we have to make to infer binding activity/inactivity of ACE2 to yeast promoters on the basis of these data?



Question for later: Is a static graphic on a single gene **ever** adequate or efficient **in genome-scale data analysis** ?



Multi-sample/multi-annotation survey

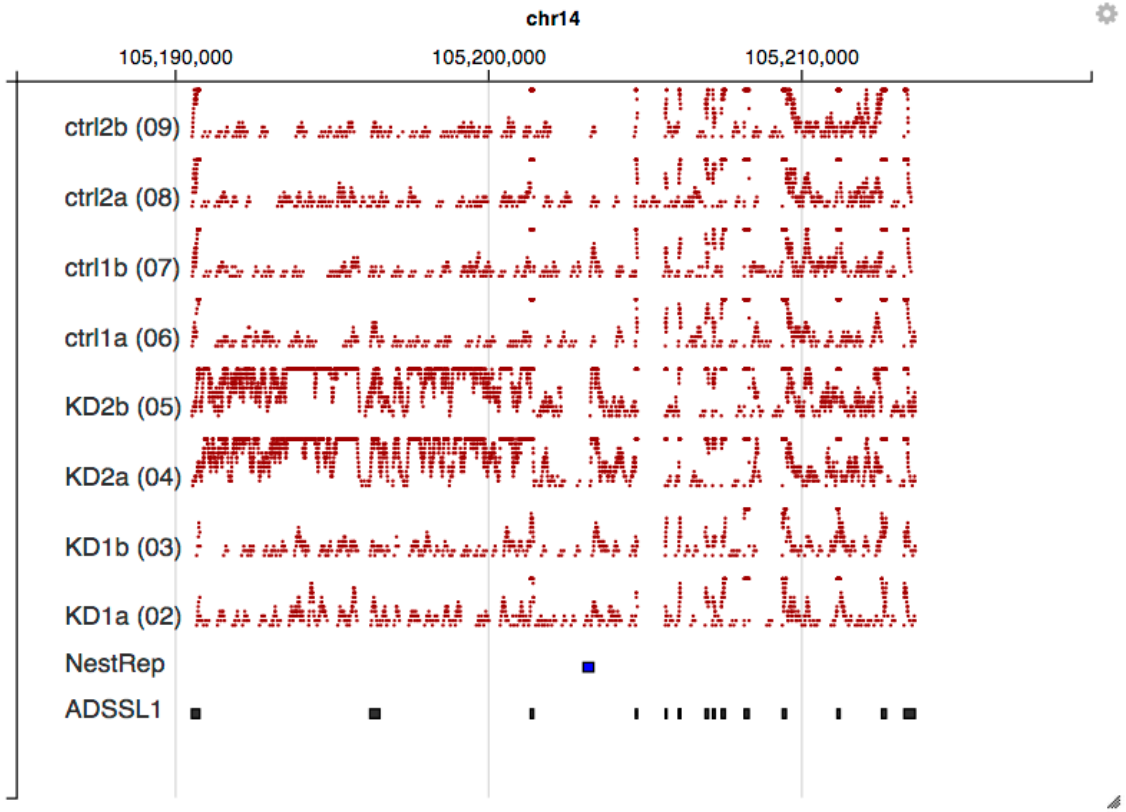
hnRNP C knockout RNA-seq data (Zarnack et al., Cell 2013)

Gene symbol for coverage display

ADSSL1

Truncate coverage height at

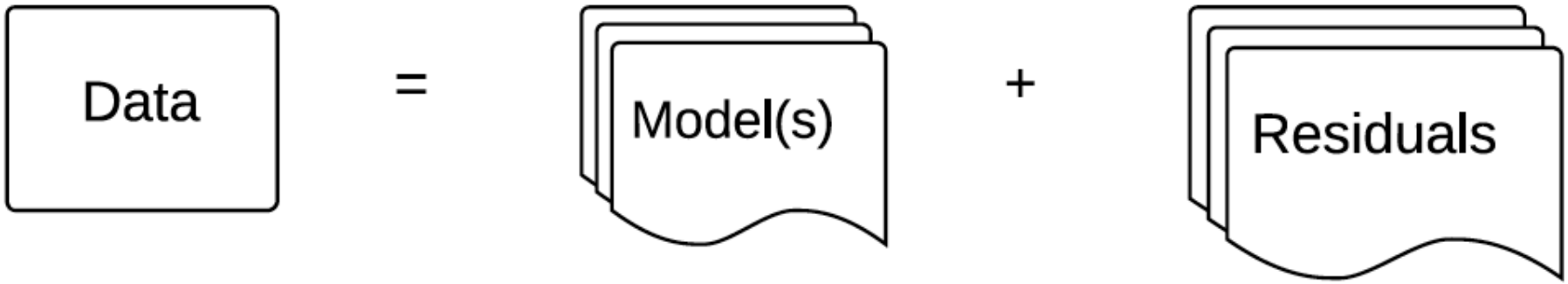
- 5
- 10
- 20
- 50

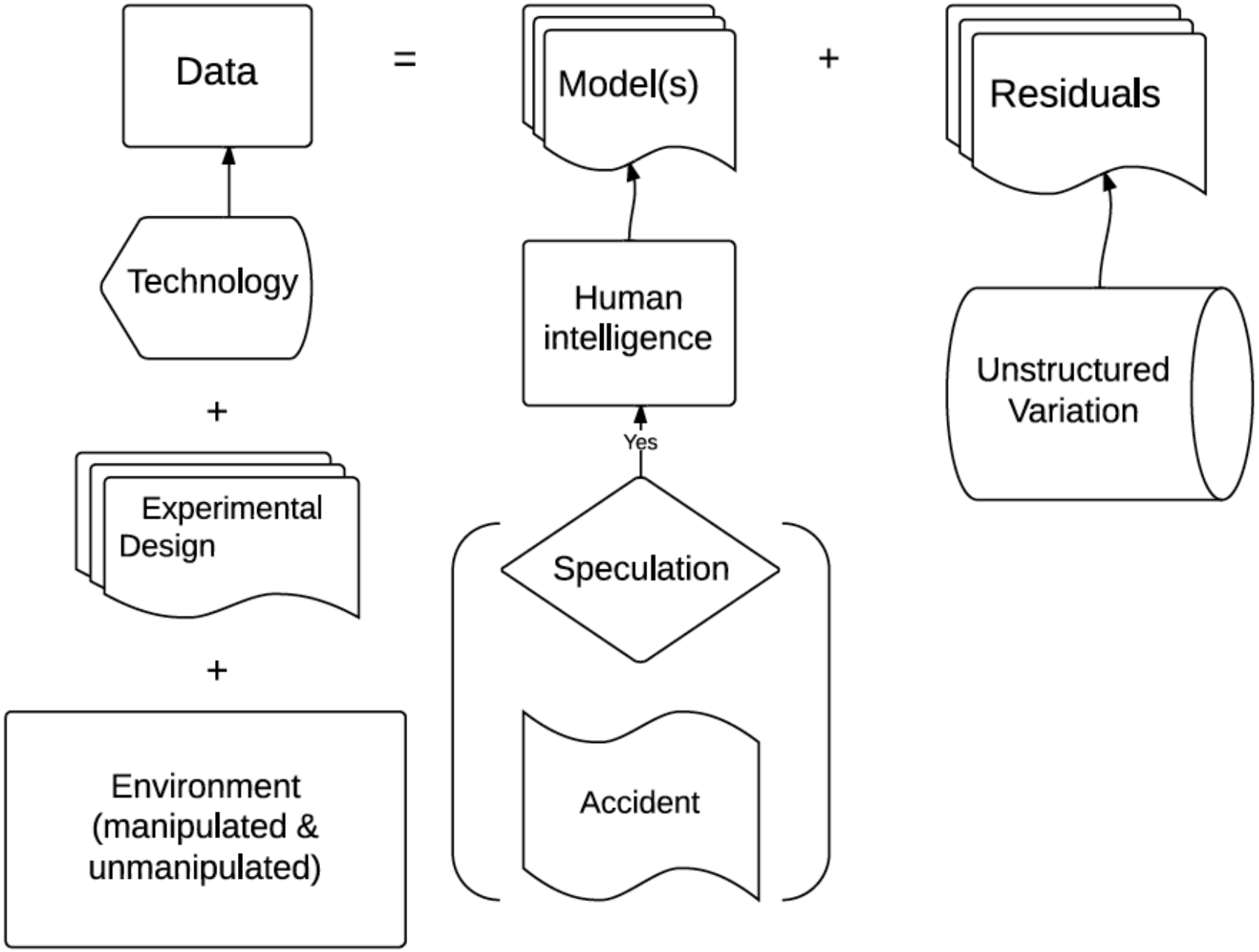


Basic strategy of this talk

- Visualization for genome-scale statistical analysis
  - Objects, processes, spaces
- Decomposing high-value graphical patterns
  - Grammatical elements
  - Rules of composition
  - Interoperable deployments
- Examples in R and Bioconductor
  - Unified solutions
  - Creating new solutions

Criticism, comparison, insights generally involve concepts of modeling …




Data
=
Model(s)
+
Residuals
Technology
Human intelligence
Unstructured Variation
+
Yes
Experimental Design
Speculation
+
Accident
Environment (manipulated & unmanipulated)

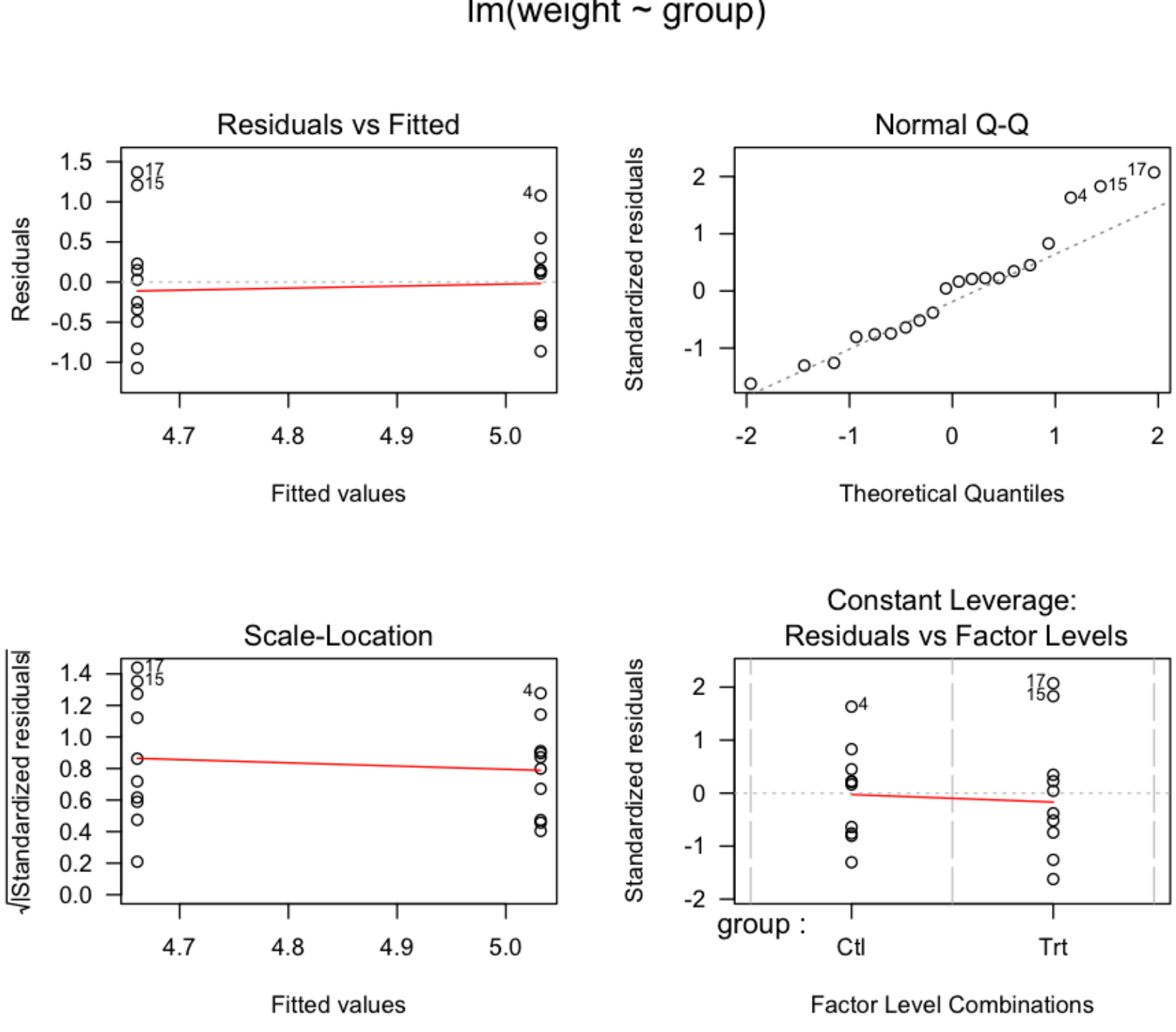
A grammar of statistical modeling

- Data = model + residuals
- All of the data are decomposed in this schema
- The “+” is arithmetic
- Revise the model if
  - Residuals show “interesting structure”
  - Model is implausible or uninterpretable
  - New data or model structures become available

An ambition of S (a precursor of R)

- `example(lm) # creates lm.D9`
  `# "lm" instance`
- `par(mfrow=c(2,2))`
- `plot(lm.D9)`

The analytical artifact (lm.D9) and the related instance of visualization generic plot() are jointly designed to bring out what an analyst needs when evaluating linear model fits


lm(weight ~ group)
Residuals vs Fitted
Residuals
Fitted values
Normal Q-Q
Standardized residuals
Theoretical Quantiles
Scale-Location
√|Standardized residuals|
Fitted values
Constant Leverage:
Residuals vs Factor Levels
Standardized residuals
group :
Ctl
Trt
Factor Level Combinations

Grammar of statistical graphics

- Proposed specifically by Leland Wilkinson
- Formalized in R by Hadley Wickham
- Key components: specify visualization with choices of
  - Coordinate system (cartesian, transformation, geographical map projections)
  - Statistic (aggregation of data through averaging, binning, smoothing)
  - "Geom" – glyph or other symbol that locates statistics in the selected coordinate system
  - "Aesthetic" – binding of data variables to graph variables

Simon et al., Cell 2001: Serial regulation of TxReg in SacCer

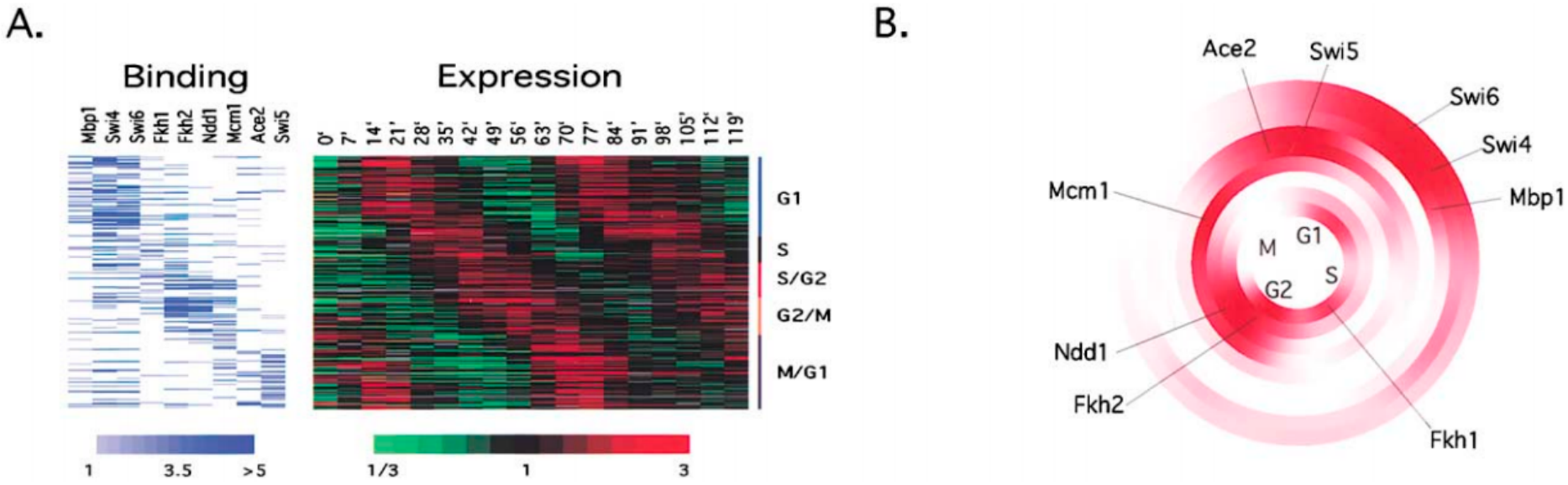
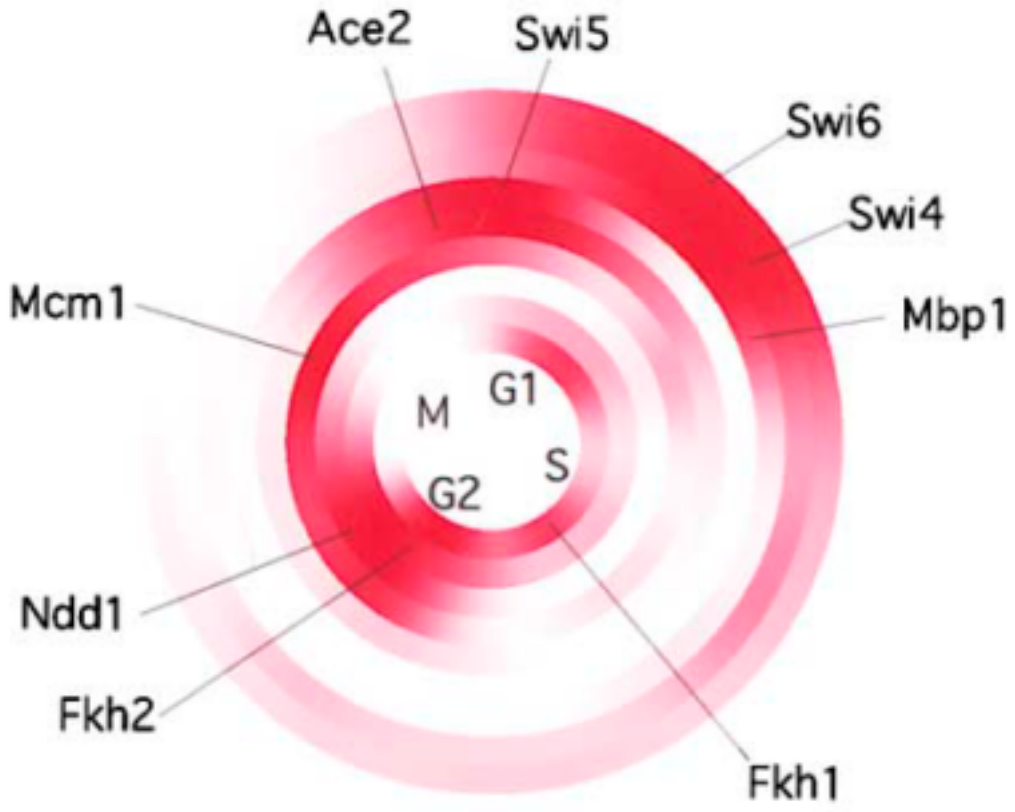


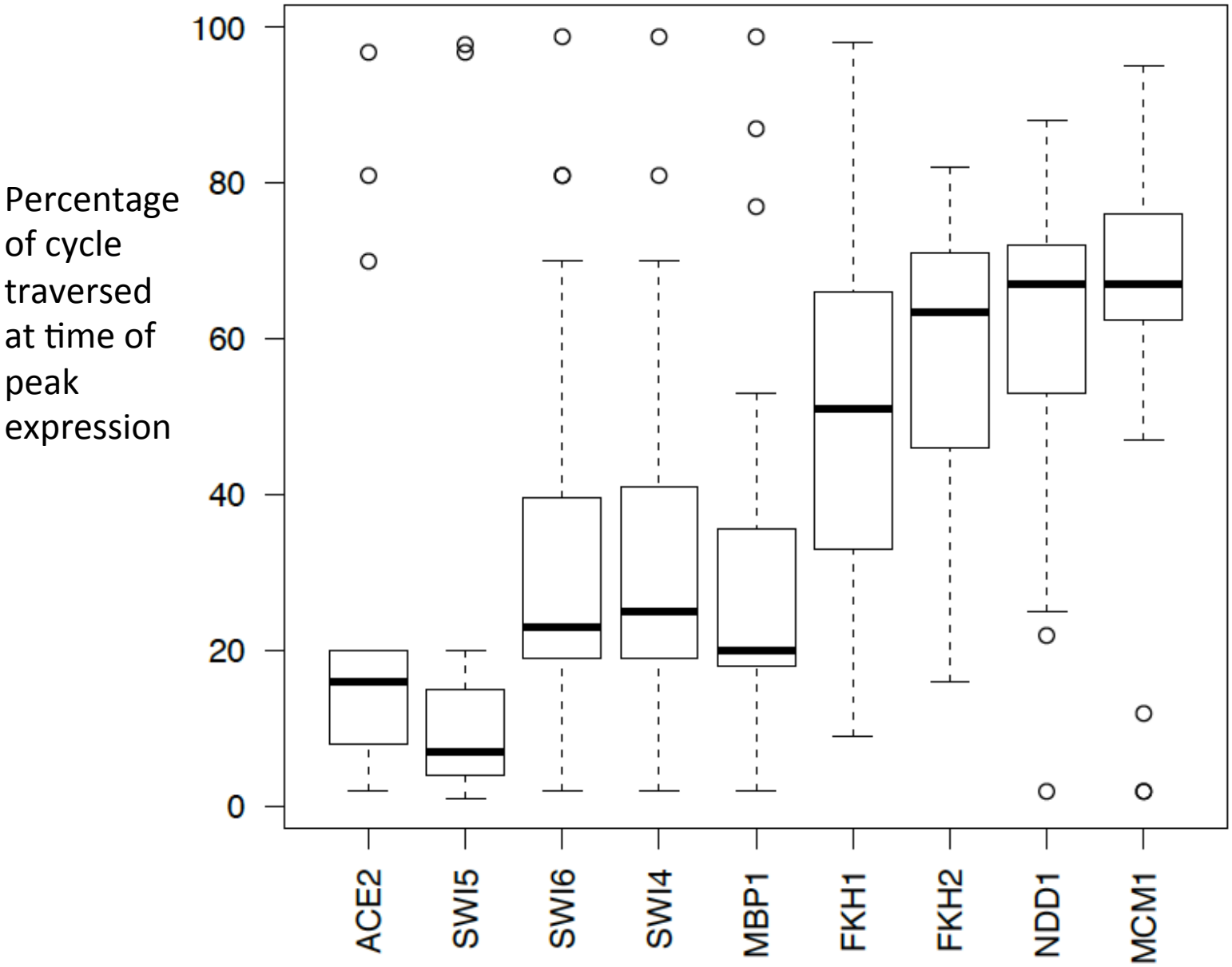
Figure 2. Genome-wide Location of the Nine Cell Cycle Transcription Factors

(A) 213 of the 800 cell cycle genes whose promoter regions were bound by a myc-tagged version of at least one of the nine cell cycle transcription factors ($p < 0.001$) are represented as horizontal lines. The weight-averaged binding ratios are displayed using a blue and white color scheme (genes with p value < 0.001 are displayed in blue). The expression ratios of an α factor synchronization time course from Spellman et al. (1998) are displayed using a red (induced) and green (repressed) color scheme.

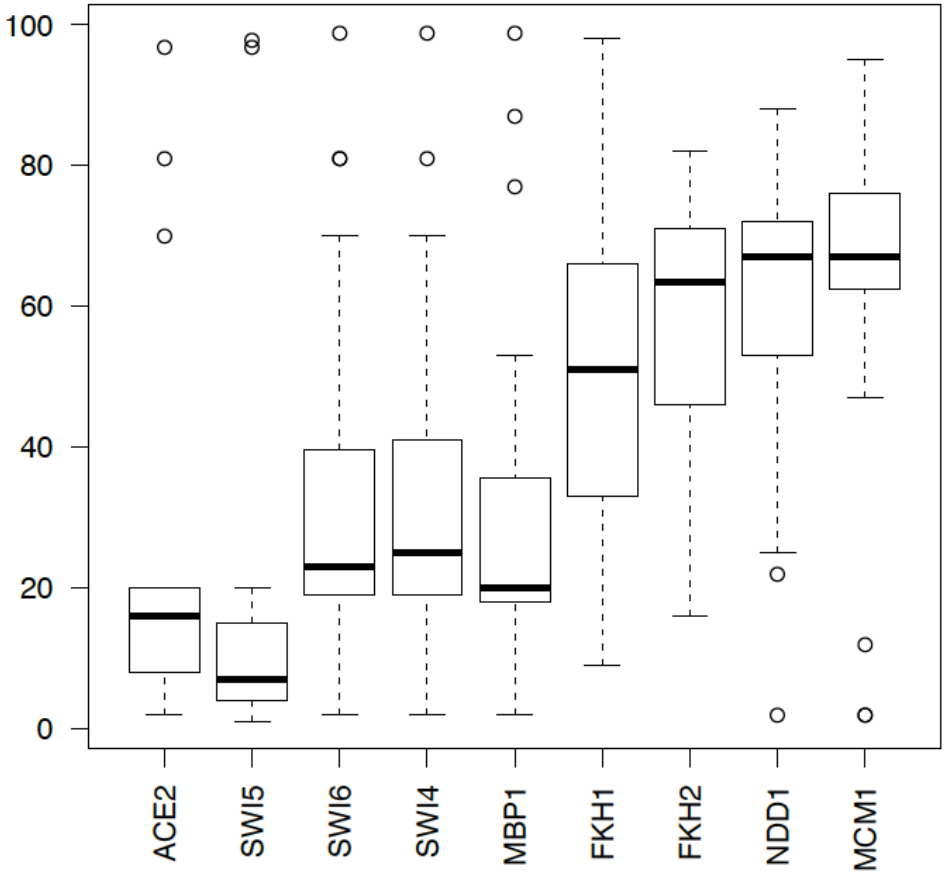
(B) The circle represents a smoothed distribution of the transcription timing (phase) of the 800 cell cycle genes (Spellman et al., 1998). The intensity of the red color, normalized by the maximum intensity value for each factor, represents the fraction of genes expressed at that point that are bound by a specific activator. The similarity in the distribution of color for specific factors (with Swi4, [illegible]) shows that these factors bind to genes that are expressed during the same time frame.


Ace2
Swi5
Swi6
Swi4
Mbp1
Mcm1
M
G1
S
G2
Ndd1
Fkh2
Fkh1

Lab activity: use ChIP-chip data from Harbison+ 2004 to sort genes, binding intensities, and expression peak timings to obtain:



There is a conceptual problem with this display: what is it?



Some comments

- Cartesian coordinates not ideal for periodic processes
  - Comment on ‘high outliers’ in the boxplots
- Network relationships of central concern, also not immediately handled
- Data = Model + Residual? How can influential visualizations present/emphasize
  - Links to data?
  - Assessment of uncertainty and opportunities for enhancement?

Using ggplot2

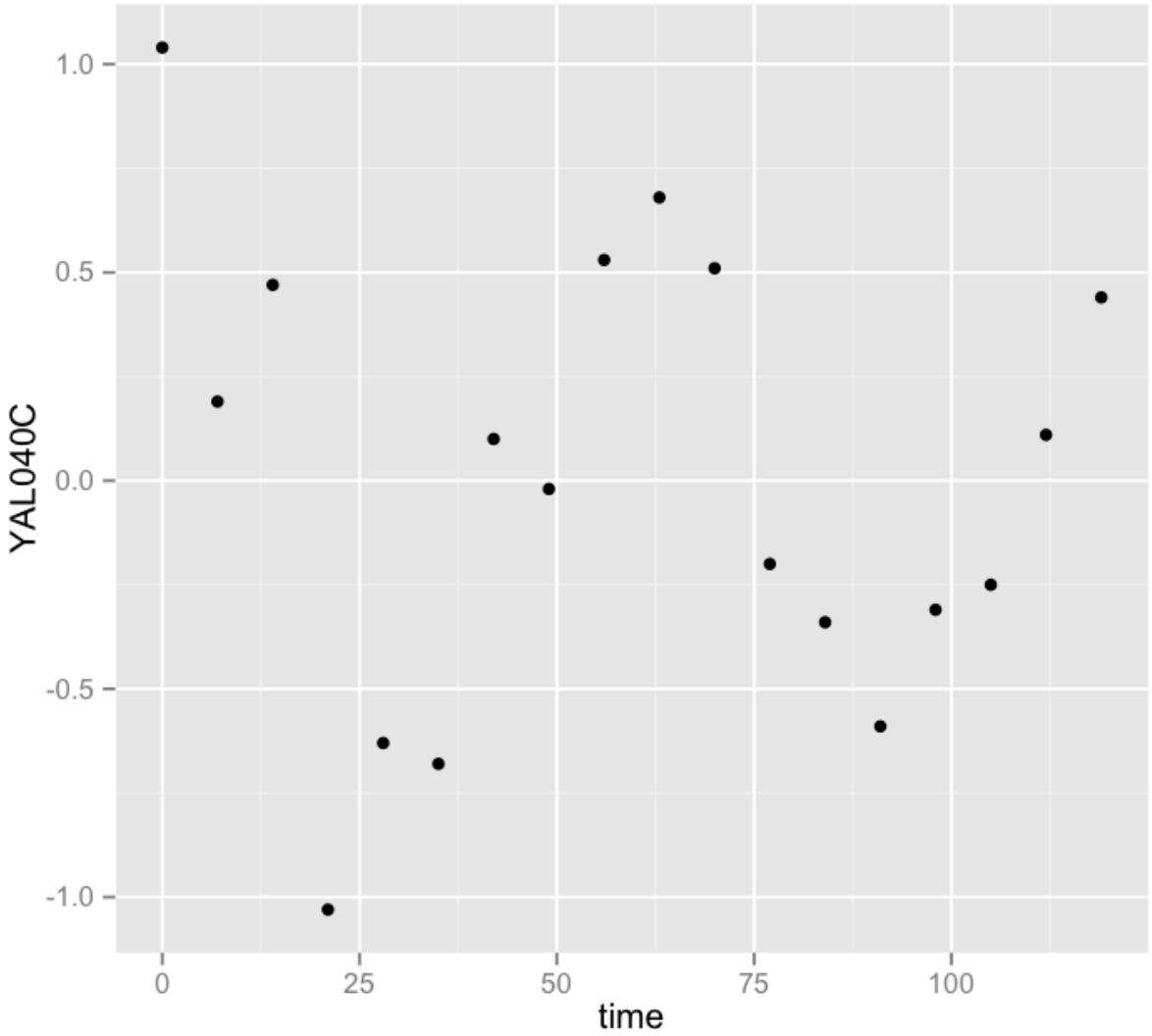
- Obtain a data frame with candidates for x, y

```
> mydf[1:4,]
          time YAL040C     orf   scatime
alpha_0      0    1.04 YAL040C   0.00000
alpha_7      7    0.19 YAL040C  38.18182
alpha_14    14    0.47 YAL040C  76.36364
alpha_21    21   -1.03 YAL040C 114.54545
```

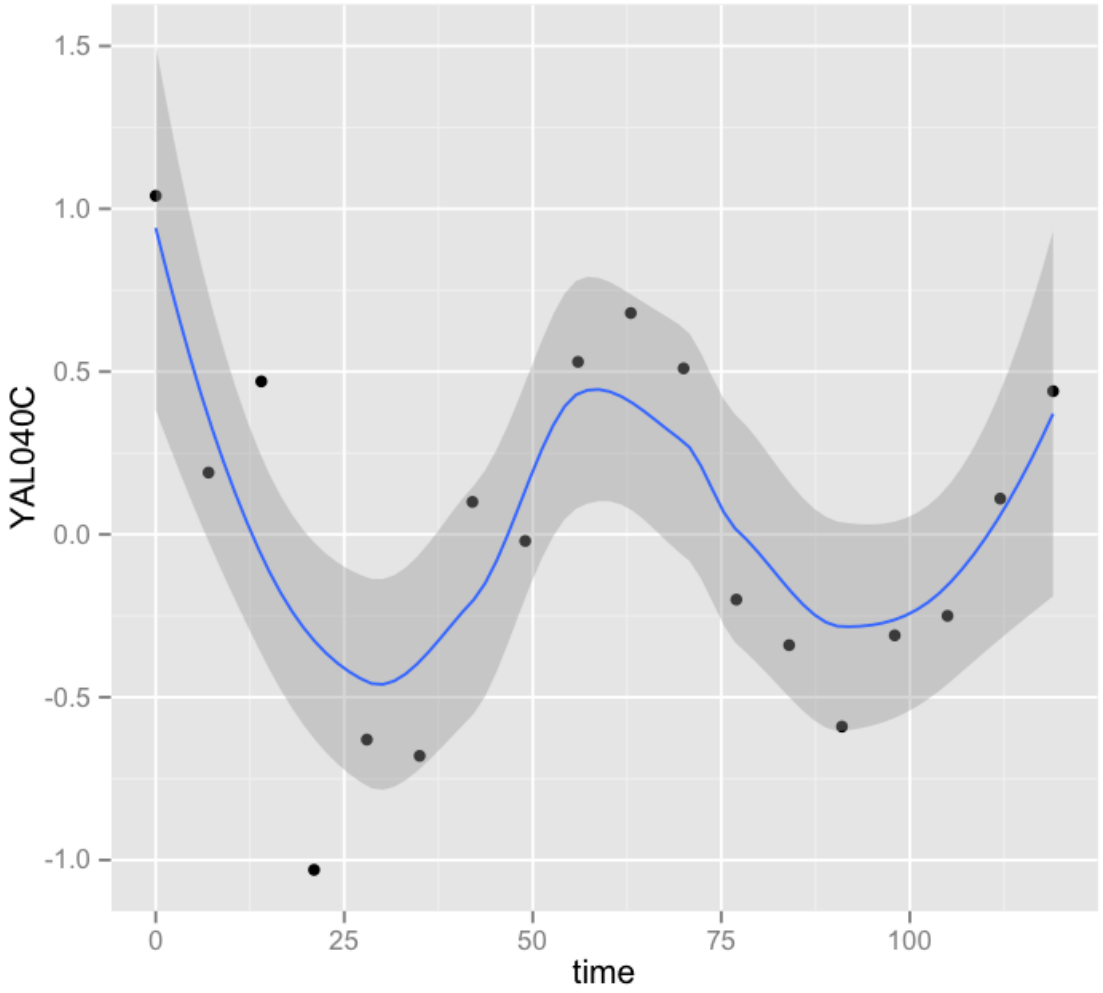
- Construct a ggplot instance

```
> ggd = ggplot(mydf, aes(x=time,y=YAL040C))
> summary(ggd)
data: time, YAL040C, orf, scatime [18x4]
mapping:  x = time, y = YAL040C
faceting: facet_null()
```

ggd + geom_point()

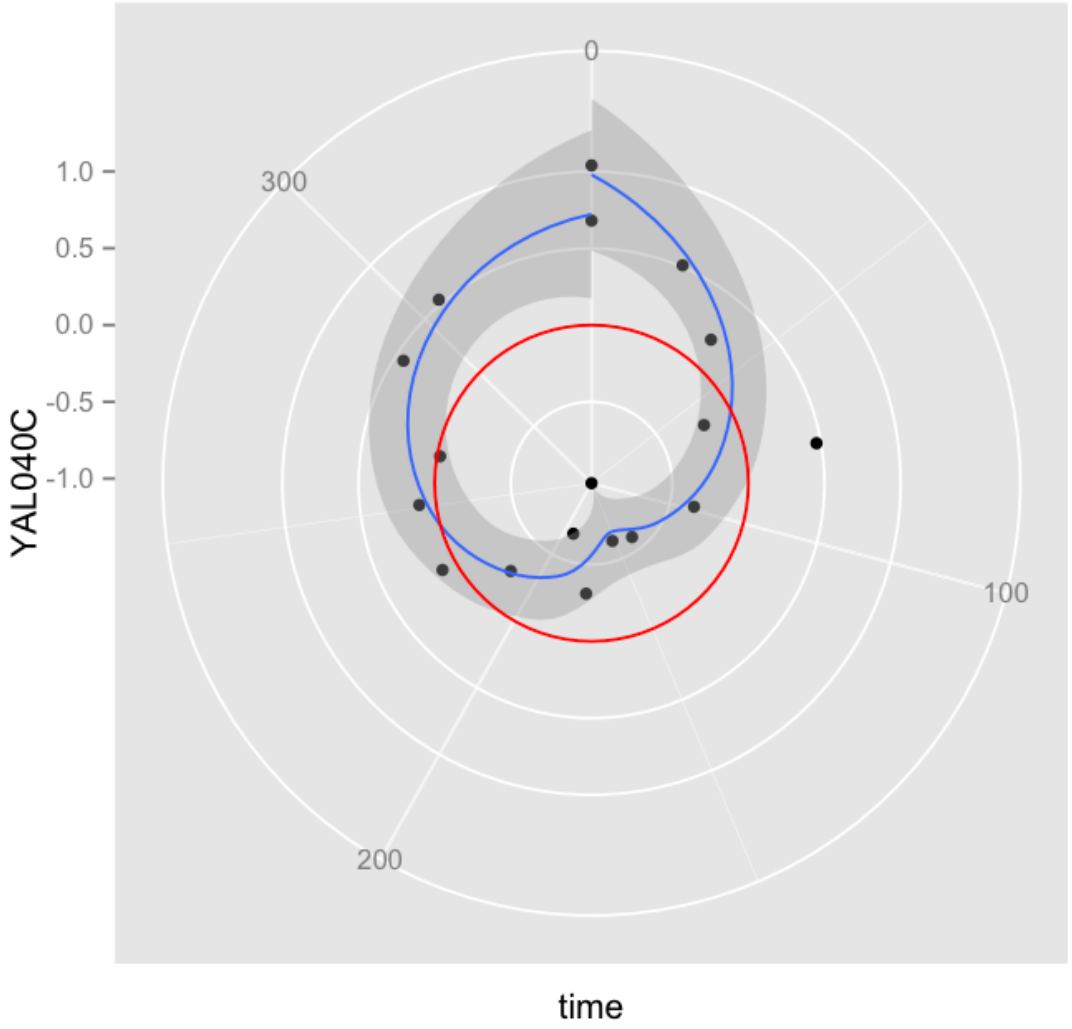


```
ggd + geom_point() +
    stat_smooth()
```



```
> ggd + geom_point(aes(x=scatime)) +
    stat_smooth(aes(x=scatime)) + coord_polar() +
    stat_abline(slope=0, intercept=0, colour="red")
```

NB. The xlab is wrong, time has been converted to degrees for period 64 min



Upshots

- *ggplot2* is capable of interesting high-level visualizations
- Fitting our visualization ambitions into the formal grammar and expanding the grammar to handle new general visuals are worthwhile activities
- For genomics: *ggbio*

2012 Genome Biology

SOFTWARE Open Access

ggbio: an R package for extending the grammar of graphics for genomic data

Tengfei Yin[1], Dianne Cook[2] and Michael Lawrence[3*]

Abstract

We introduce ggbio, a new methodology to visualize and explore genomics annotations and high-throughput data. The plots provide detailed views of genomic regions, summary views of sequence alignments and splicing patterns, and genome-wide overviews with karyogram, circular and grand linear layouts. The methods leverage the statistical functionality available in R, the grammar of graphics and the data handling capabilities of the Bioconductor project. The plots are specified within a modular framework that enables users to construct plots in a systematic way, and are generated directly from Bioconductor data structures. The ggbio R package is available at http://www.bioconductor.org/packages/2.11/bioc/html/ggbio.html.

Rationale

Data are typically plotted along with annotations with

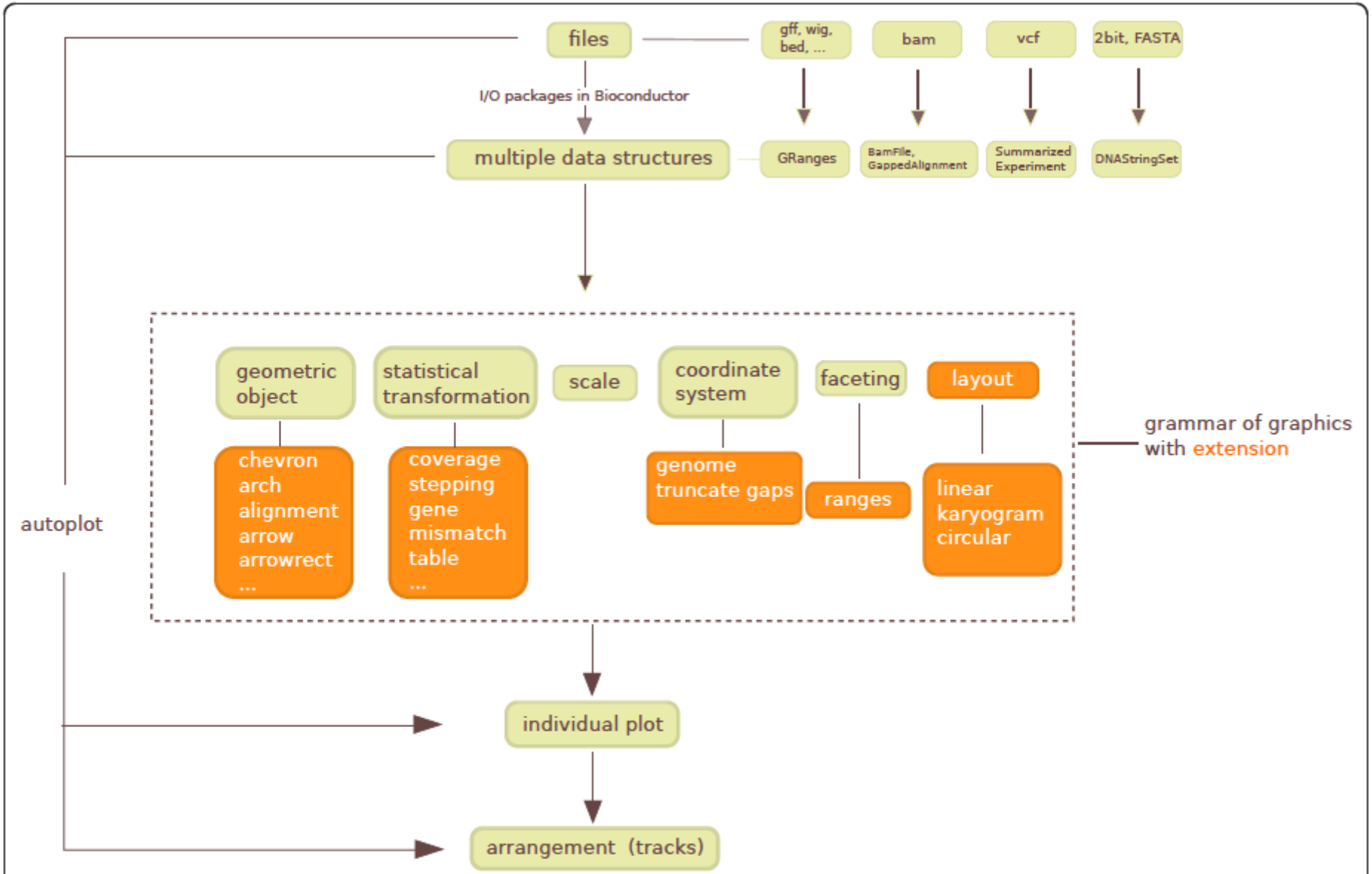


Figure 9 Diagram of the ggbio framework for processing sequence data. It starts with a mapping from different file types to different objects or data structure in R, using Bioconductor tools, followed by general and extended grammar of graphics mapping of data elements to graphical components. The final stage arranges the graphics in a designed layout to show annotation tracks or multiple data sets. Orange boxes and dark brown arrows indicate the extensions provided by ggbio.

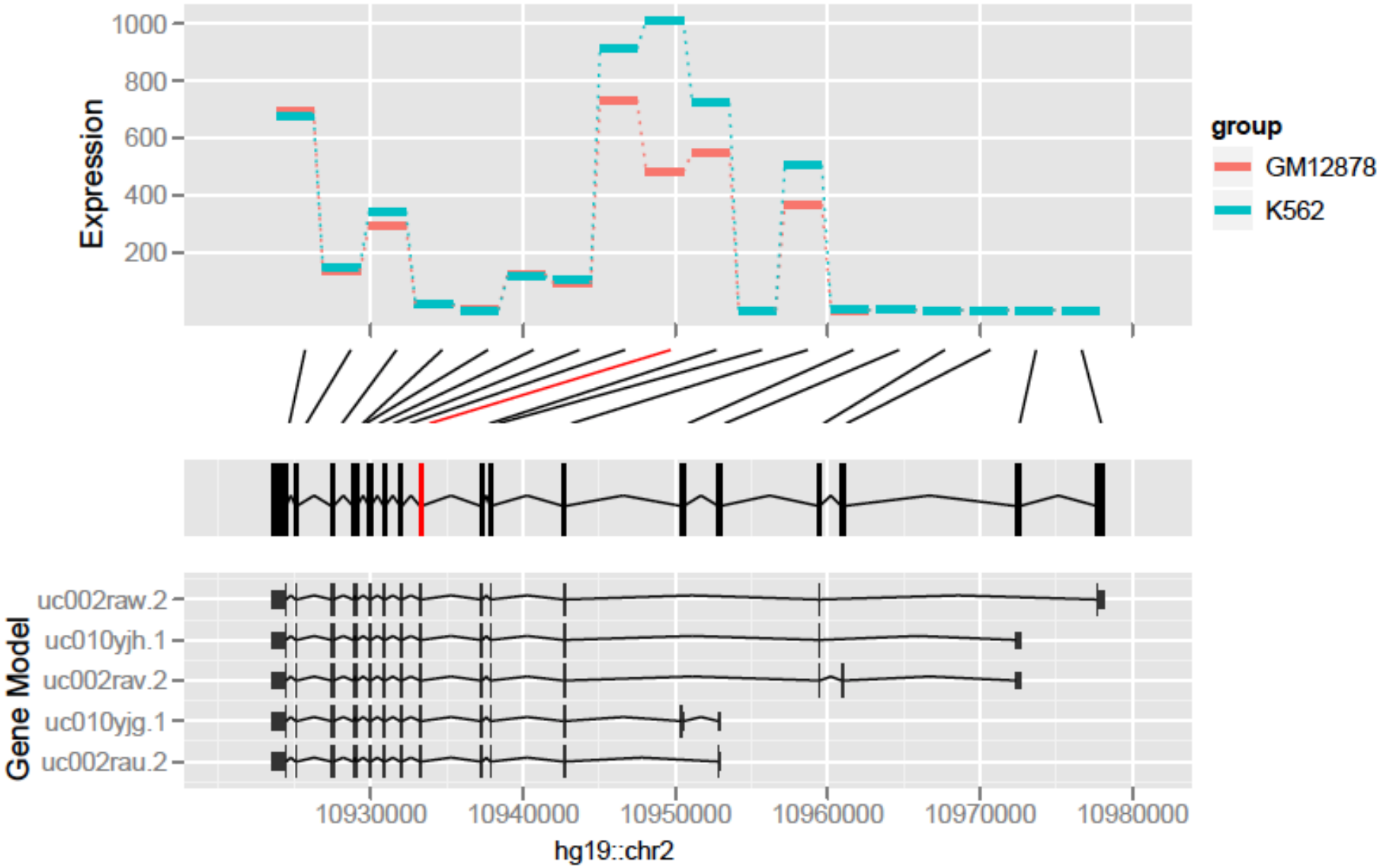


Figure 7 Edge-linked interval to data view. Edge-linked interval to data view for the expression of the exons of gene *PDIA6*. The top track shows the expression level for each of the exons, and the color indicates the sample (GM12878 or K562). The second track shows the links between the even-spaced expression track and the exons track, below. The package DEXseq, which produces a similar graphic, computes differential expression and significance, and significance is indicated by coloring the connecting lines red. The track at the bottom shows the annotated transcripts.

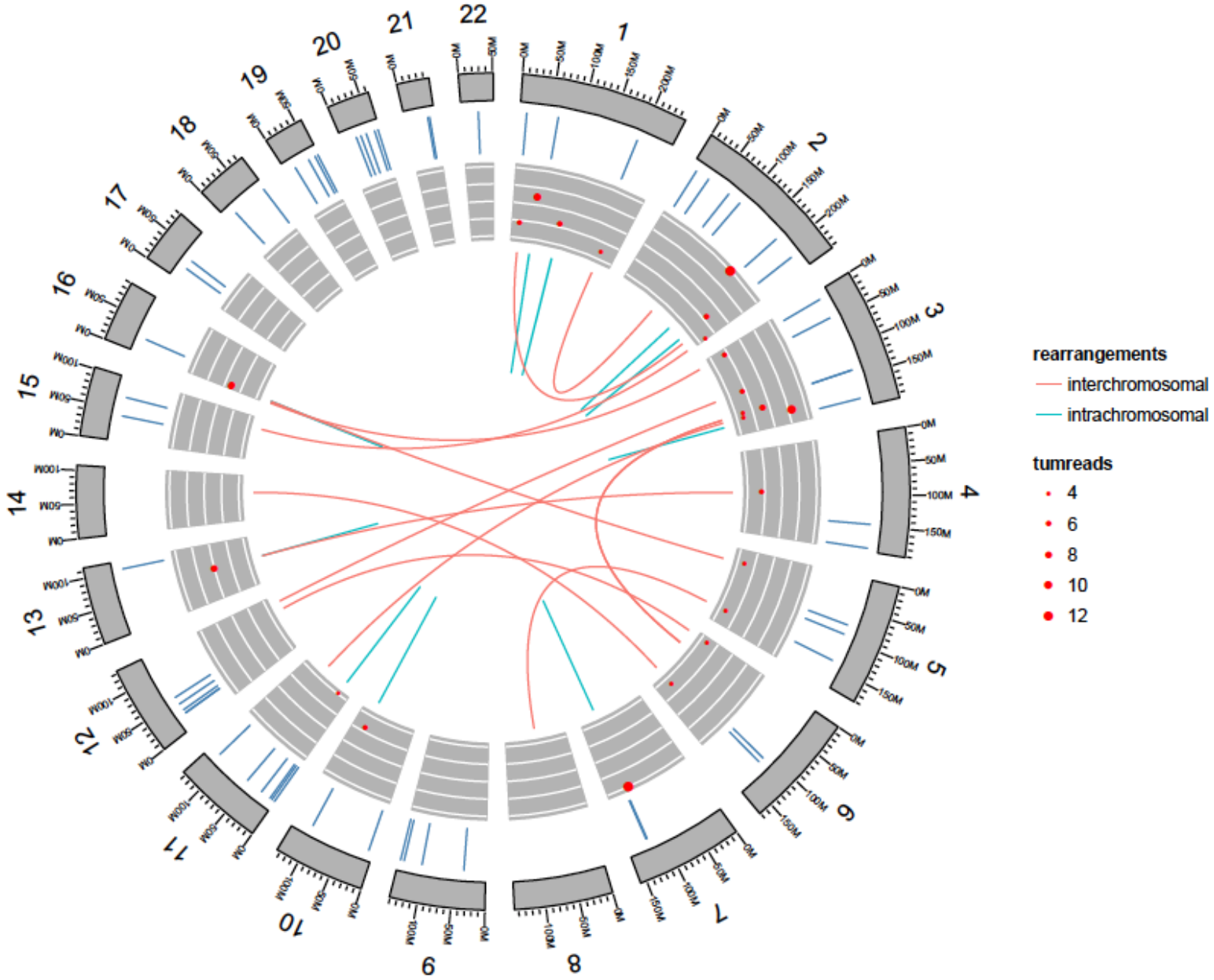


Figure 5 Single sample circular view. DNA structural rearrangements and somatic mutation in a single colorectal tumor sample (CRC-1). The outer ring shows the ideogram of the human autosomes, labeled with chromosome numbers and scales. The segments represent the missense somatic mutations. The point tracks show score and support for rearrangement. The size of the points indicates the number of supporting read pairs in the tumor and the y value indicates the score for each rearrangement. The links represent the rearrangements, where intrachromosomal events are colored green and interchromosomal events are colored orange.

Upshots

- ggbio extends ggplot2 grammar elements
- Autonomous selection of detailed transformations and coordinatization
- Helps with the exploration of upstream high-volume quantities
- Supporting statistical inference?
  - Featurewise: Group significant vs. nonsignificant
  - Residuals? Perhaps compute them separately and reapply visualization functions

Approaches to visualization with NGS data

- ggbio – extended grammar of statistical graphics
- Gviz – track-oriented object sequences
- epivizr – genome browser with statistical enhancements
- “built-in minimal genome browser”, based on shiny (see h5vc package)
- And more …

An embarrassment of riches, learn them all…

Bioconductor version 2.14 (Release)

Autocomplete biocViews search:

- ResearchField (148)
- StatisticalMethod (206)
- Technology (509)
- WorkflowStep (405)
  - Alignment (3)
  - Annotation (63)
  - BatchEffect (1)
  - ExperimentalDesign (1)
  - MultipleComparison (59)
  - Normalization (2)
  - Pathways (56)
  - Preprocessing (121)
  - QualityControl (81)
  - ReportWriting (20)
  - Visualization (180)
- AnnotationData (867)
- ExperimentData (202)

Packages found under Visualization:

Show All entries Search table:

Package	Maintainer	Title
AffyExpress	Xuejun Arthur Li	Affymetrix Quality Assessment and Analysis Tool
ampliQueso	Michal Okoniewski	Analysis of amplicon enrichment panels
annmap	Tim Yates	Genome annotation and visualisation package pertaining to Affymetrix arrays and NGS analysis.
aroma.light	Henrik Bengtsson	Light-weight methods for normalization and visualization of microarray data using only basic R data types
arrayQuality	Agnes Paquet	Assessing array quality on spotted arrays
ArrayTools	Arthur Li	geneChip Analysis Package
Basic4Cseq	Carolin Walter	Basic4Cseq: an R/Bioconductor package for analyzing 4C-seq data
BiGGR	Anand K. Gavai, Hannes Hettling	Constraint based modeling in R using metabolic reconstruction databases.
bioassayR	Tyler Backman	R library for Bioactivity analysis
biocGraph	Florian Hahne	Graph examples and use cases in Bioinformatics
BioMVCClass	Elizabeth Whalen	Model-View-Controller (MVC) Classes That Use Biobase
biomvRCNS	Yang Du	Copy Number study and Segmentation for multivariate biological data
BioNet	Marcus Dittrich	Routines for the functional analysis of biological networks

And 167 more ….

Exemplar: Zarnack et al., Cell 2013

Direct Competition between hnRNP C and U2AF65 Protects the Transcriptome from the Exonization of *Alu* Elements

Kathi Zarnack,[1,8] Julian König,[2,8] Mojca Tajnik,[2,3] Iñigo Martincorena,[1] Sebastian Eustermann,[2] Isabelle Stévant,[1] Alejandro Reyes,[4] Simon Anders,[4] Nicholas M. Luscombe,[1,5,6,7,*] and Jernej Ule[2,*]

[1]European Molecular Biology Laboratory (EMBL) European Bioinformatics Institute, Wellcome Trust Genome Campus, Hinxton, Cambridge CB10 1SD, UK

[2]MRC Laboratory of Molecular Biology, Hills Road, Cambridge CB2 0QH, UK

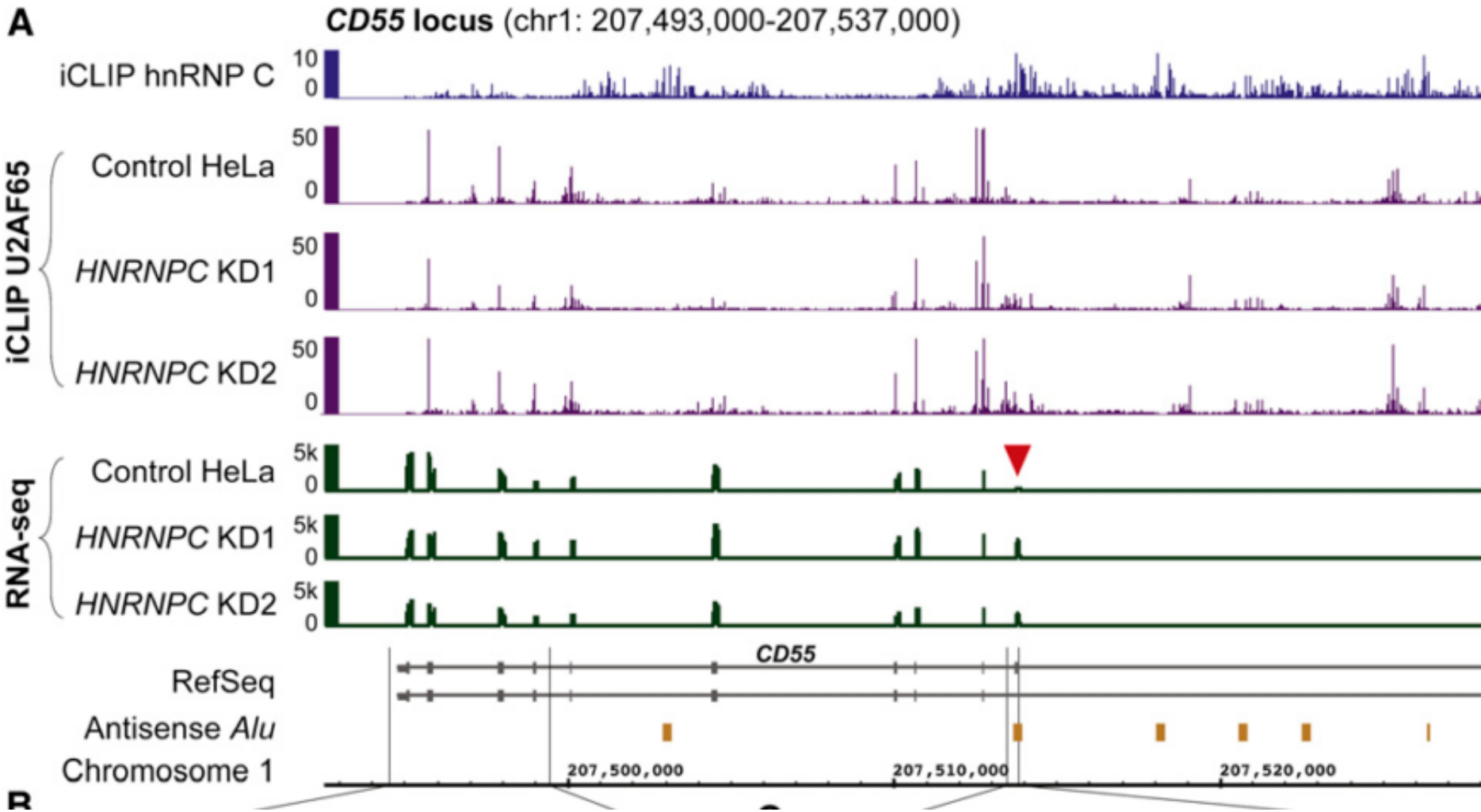
[3]Faculty of Medicine, University of Ljubljana, Vrazov trg 2, SI-1104 Ljubljana, Slovenia

[4]EMBL, Genome Biology Unit, Meyerhofstraße 1, 69117 Heidelberg, Germany

[5]UCL Genetics Institute, Department of Genetics, Environment and Evolution, University College London, Gower Street, London WC1E 6BT, UK

[6]Cancer Research UK London Research Institute, 44 Lincoln's Inn Fields, London WC2A 3LY, UK

[7]Okinawa Institute for Science and Technology Graduate University, 1919-1 Tancha, Onna-son, Kunigami-gun, Okinawa 904-0495, Japan

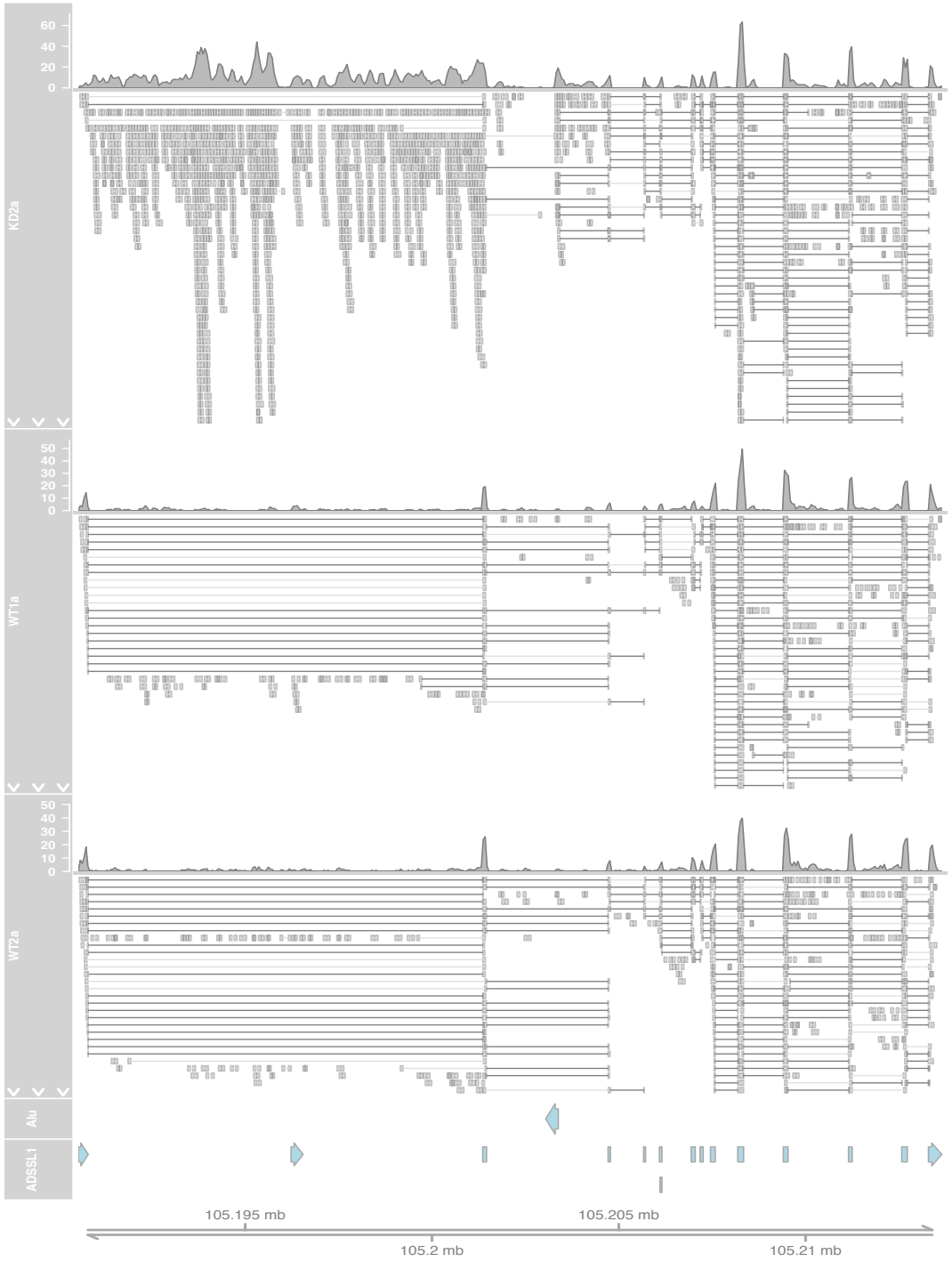


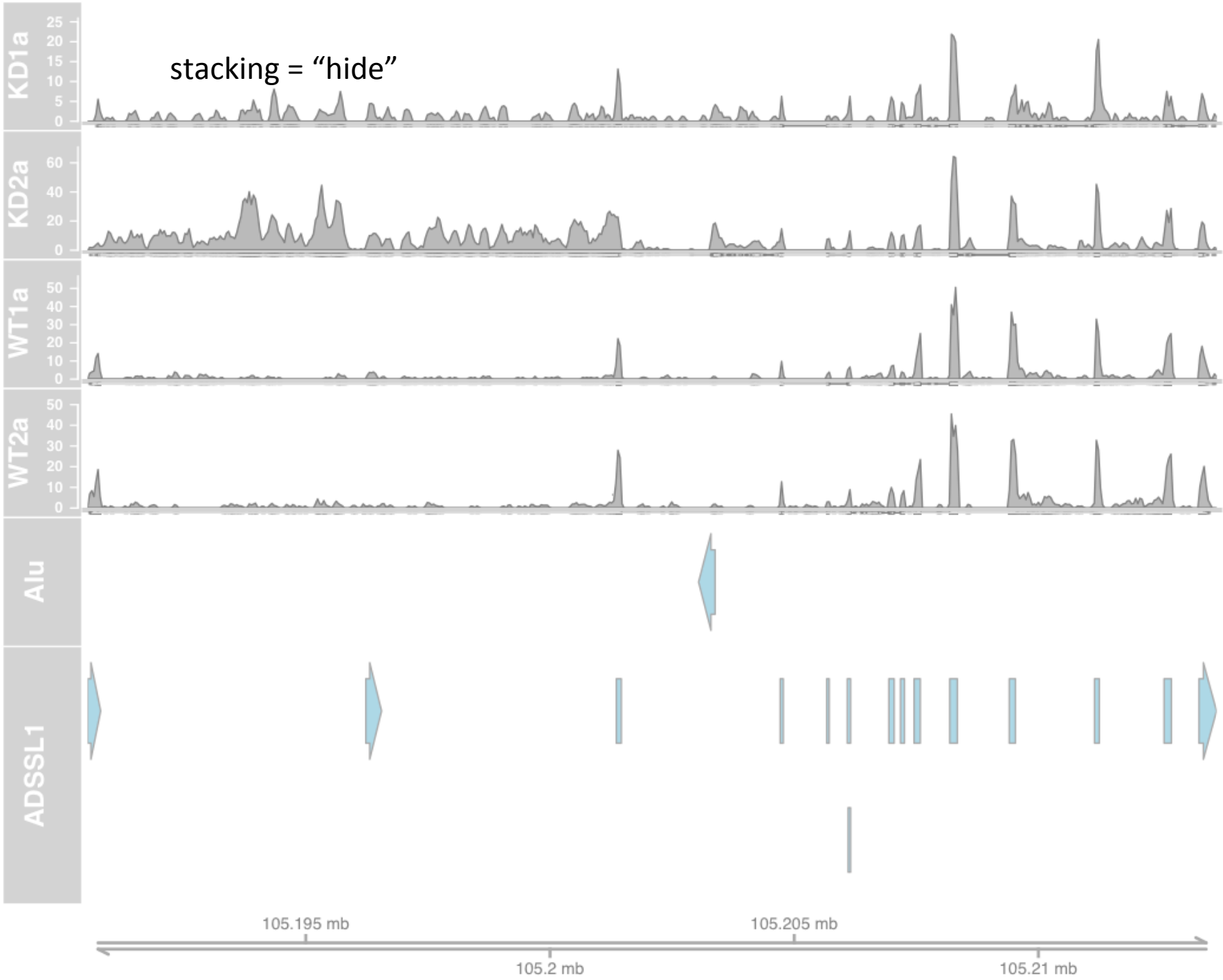
iCLIP reads show U2AF65 present when HNRNPC knocked out; RNA-seq shows exonization in vicinity of Alu … note scales

Table S3B: Validation of *Alu* exon that are more than 2-fold upregulated in the RNA-seq data using RT-PCR.

Gene symbol	Gene description	Exon coordinates	Strand	RNA-seq p value KD1	RNA-seq p value KD2
Validated					
ADSSL1	adenylosuccinate synthase like 1	chr14:105203269-105203425	+	7.59E-02	2.66E-06
BLM	Bloom syndrome, RecQ helicase-like	chr15:91267265-91267374	+	6.85E-02	3.28E-03
CDK5RAP2	CDK5 regulatory subunit associated protein 2	chr9:123333181-123333269	-	3.10E-03	9.93E-02
CGRRF1	cell growth regulator with ring finger domain 1	chr14:54994502-54994601	+	1.31E-01	2.90E-03
CSPP1	centrosome and spindle pole associated protein 1	chr8:67987041-67987161	+	5.19E-02	2.25E-04
DAP3	death associated protein 3	chr1:155678824-155678918	+	9.42E-02	1.01E-02
DDX23	DEAD (Asp-Glu-Ala-Asp) box polypeptide 23	chr12:49233258-49233334	-	6.65E-02	4.31E-03
EDC3	enhancer of mRNA decapping 3 homolog (S. cerevisiae)	chr15:74972143-74972252	-	1.17E-02	1.01E-02
IL6R	interleukin 6 receptor	chr1:154409911-154410069	+	1.13E-01	8.17E-03
NPLOC4	nuclear protein localization 4 homolog (S. cerevisiae)	chr17:79530947-795310054	-	2.83E-02	1.11E-01
NUP133	nucleoporin 133kDa	chr1:229601516-229601634	-	*	*
NUP160	nucleoporin 160kDa	chr11:47848627-47848746	-	1.62E-02	7.48E-02
PCNX	pecanex homolog (Drosophila)	chr14:71520031-71520129	+	5.18E-02	2.04E-02
RPS15A	ribosomal protein S15a	chr16:18797615-18797658	-	6.82E-03	9.17E-02
YARS2	tyrosyl-tRNA synthetase 2, mitochondrial	chr12:32905908-32906083	-	1.05E-02	7.75E-01
ZNF791	zinc finger protein 791	chr19:12723040-12723130	+	9.11E-04	1.71E-01
Not validated					
HAUS2	HAUS augmin-like complex, subunit 2	chr15:42852980-42853068	+	2.02E-01	6.26E-03
KIAA0101	KIAA0101	chr15:64671411-64671508	-	4.17E-03	6.75E-01
STAG1	stromal antigen 1	chr3:136111495-136111604	-	2.08E-02	6.31E-02
NUP98	nucleoporin 98kDa	chr11:3718950-3719043	-	3.39E-02	1.14E-01

Gviz squish presentation of all reads in ADSSL1



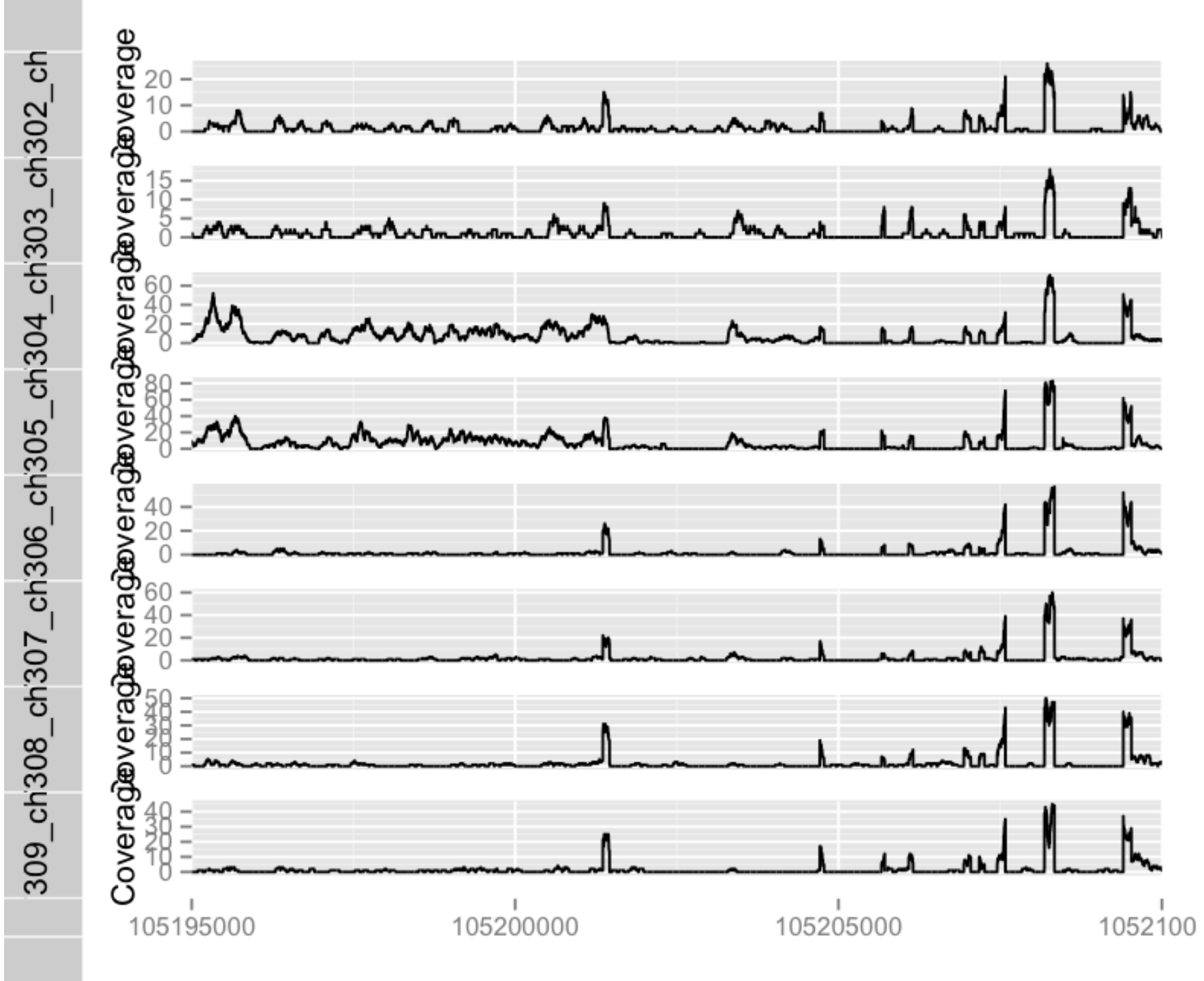

KD1a
25
20
15
10
5
0
stacking = “hide”
KD2a
60
40
20
0
WT1a
50
40
30
20
10
0
WT2a
50
40
30
20
10
0
Alu
ADSSL1
105.195 mb
105.2 mb
105.205 mb
105.21 mb

Gviz workflow for display

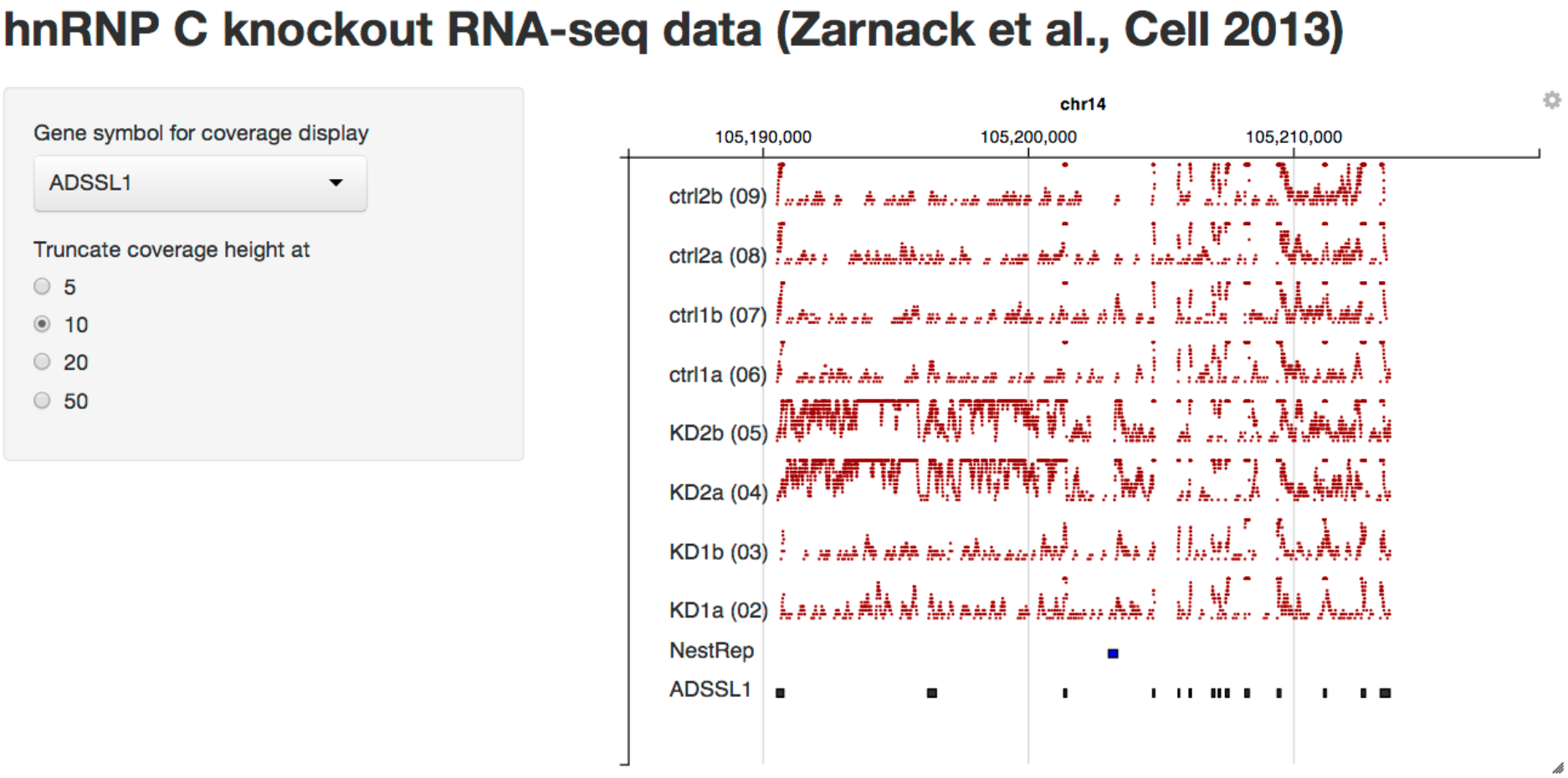
- BAM files, also need sample annotation
- AlignmentsTrack generated directly from BAM
- AnnotationTrack for gene model derived from TxDb instance
- AnnotationTrack for Alu addresses, derived from nestedRepeats track in AnnotationHub
- plotTracks call orders displays

ggbio workflow

- autoplot(fileList(bfv), which=gma, names=n) +ylim(0,60) + xlim(105.195e6,105.210e6)


309_ch308_ch307_ch306_ch305_ch304_ch303_ch302_ch
Coverage
20
10
0
15
10
5
0
60
40
20
0
80
60
40
20
0
40
20
0
60
40
20
0
50
40
30
20
10
0
40
30
20
10
0
105195000
105200000
105205000
1052100

An interactive RNA-seq browser using ggvis+shiny – see viz14dyn::zarnBrowse()



Key data elements: BAM files
Key annotation elements: Nested repeats from AnnotationHub, gene models from TxDb
Infrastructure: ggvis+shiny+ …

```
> sessionInfo()
R version 3.1.0 (2014-04-10)
Platform: x86_64-apple-darwin10.8.0 (64-bit)

locale:
[1] C

attached base packages:
[1] parallel  stats     graphics  grDevices datasets  utils     tools     methods   base

other attached packages:
 [1] GenomicFiles_1.1.10   BiocParallel_0.7.2    rtracklayer_1.25.11   Rsamtools_1.17.27     Biostrings_2.33.10
 [6] XVector_0.5.6         GenomicRanges_1.17.18 org.Hs.eg.db_2.14.0   RSQLite_0.11.4        DBI_0.2-7
[11] AnnotationDbi_1.27.8  GenomeInfoDb_1.1.8    IRanges_1.99.15       S4Vectors_0.0.8       Biobase_2.25.0
[16] BiocGenerics_0.11.2   viz14dyn_0.0.3        viz14_0.0.11          shiny_0.10.0          ggvis_0.2.0.99
[21] weaver_1.31.0         codetools_0.2-8       digest_0.6.4          BiocInstaller_1.15.5

loaded via a namespace (and not attached):
 [1] BBmisc_1.6                BSgenome_1.33.8           BatchJobs_1.2             Formula_1.1-1
 [5] GGally_0.4.6              GenomicAlignments_1.1.14  GenomicFeatures_1.17.10   Gviz_1.9.7
 [9] Hmisc_3.14-4              MASS_7.3-33               OrganismDbi_1.7.2         R.methodsS3_1.6.1
[13] RBGL_1.41.0               RColorBrewer_1.0-5        RCurl_1.95-4.1            RJSONIO_1.2-0.2
[17] Rcpp_0.11.2               VariantAnnotation_1.11.11 XML_3.98-1.1              assertthat_0.1
[21] biomaRt_2.21.0            biovizBase_1.13.7         bitops_1.0-6              brew_1.0-6
[25] caTools_1.17              cluster_1.15.2            colorspace_1.2-4          dichromat_2.0-0
[29] dplyr_0.2                 fail_1.2                  foreach_1.4.2             ggbio_1.13.7
[33] ggplot2_1.0.0             graph_1.43.0              grid_3.1.0                gridExtra_0.9.1
[37] gtable_0.1.2              htmltools_0.2.4           httpuv_1.3.0              iterators_1.0.7
[41] lattice_0.20-29           latticeExtra_0.6-26       magrittr_1.0.1            matrixStats_0.10.0
[45] munsell_0.4.2             parody_1.23.0             plyr_1.8.1                proto_0.3-10
[49] reshape_0.8.5             reshape2_1.4              scales_0.2.4              sendmailR_1.1-2
[53] splines_3.1.0             stats4_3.1.0              stringr_0.6.2             survival_2.37-7
[57] xtable_1.7-3              zlibbioc_1.11.1
```

From Lawrence and Morgan 2014 (to appear), the MVC pattern for a genomic visualization

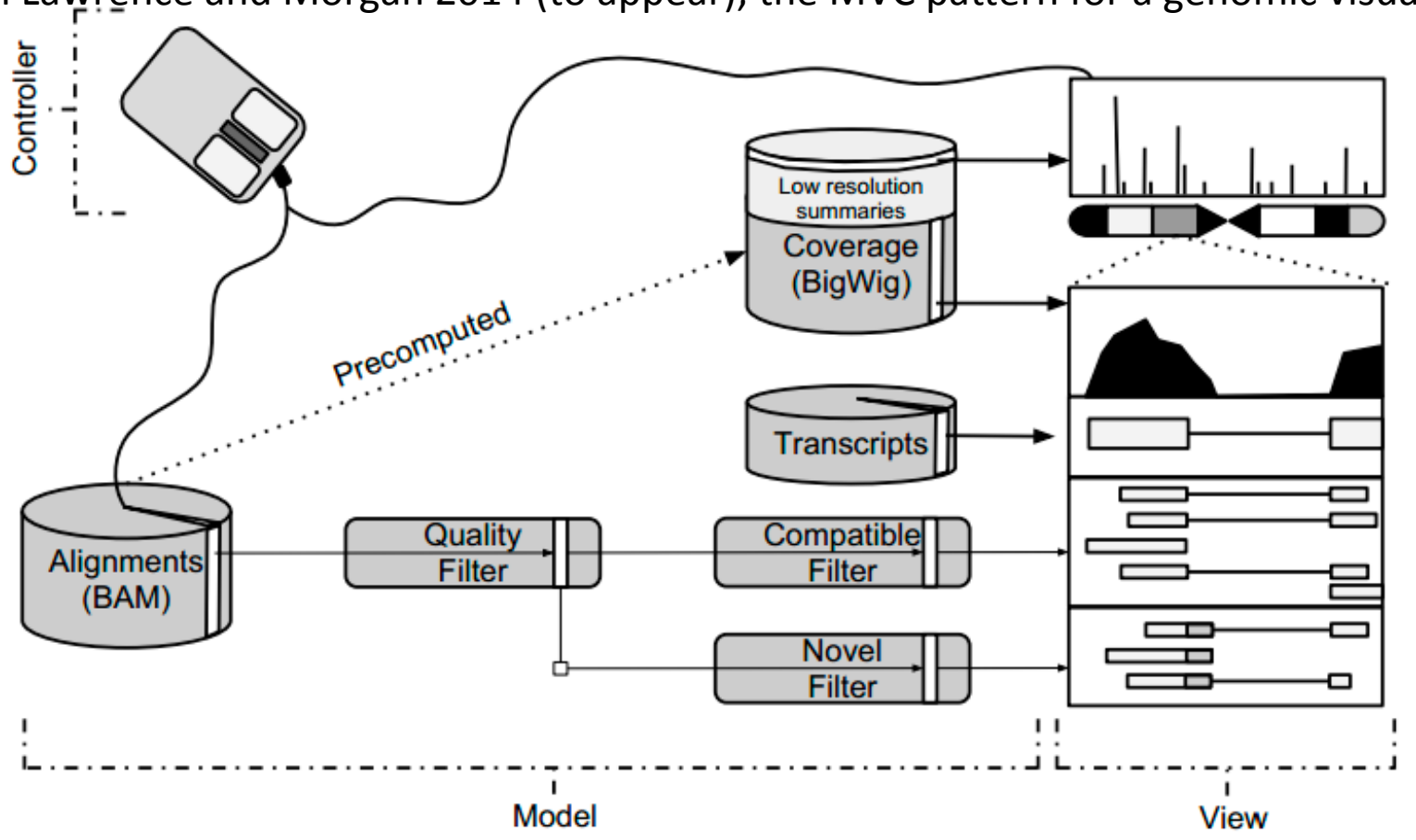
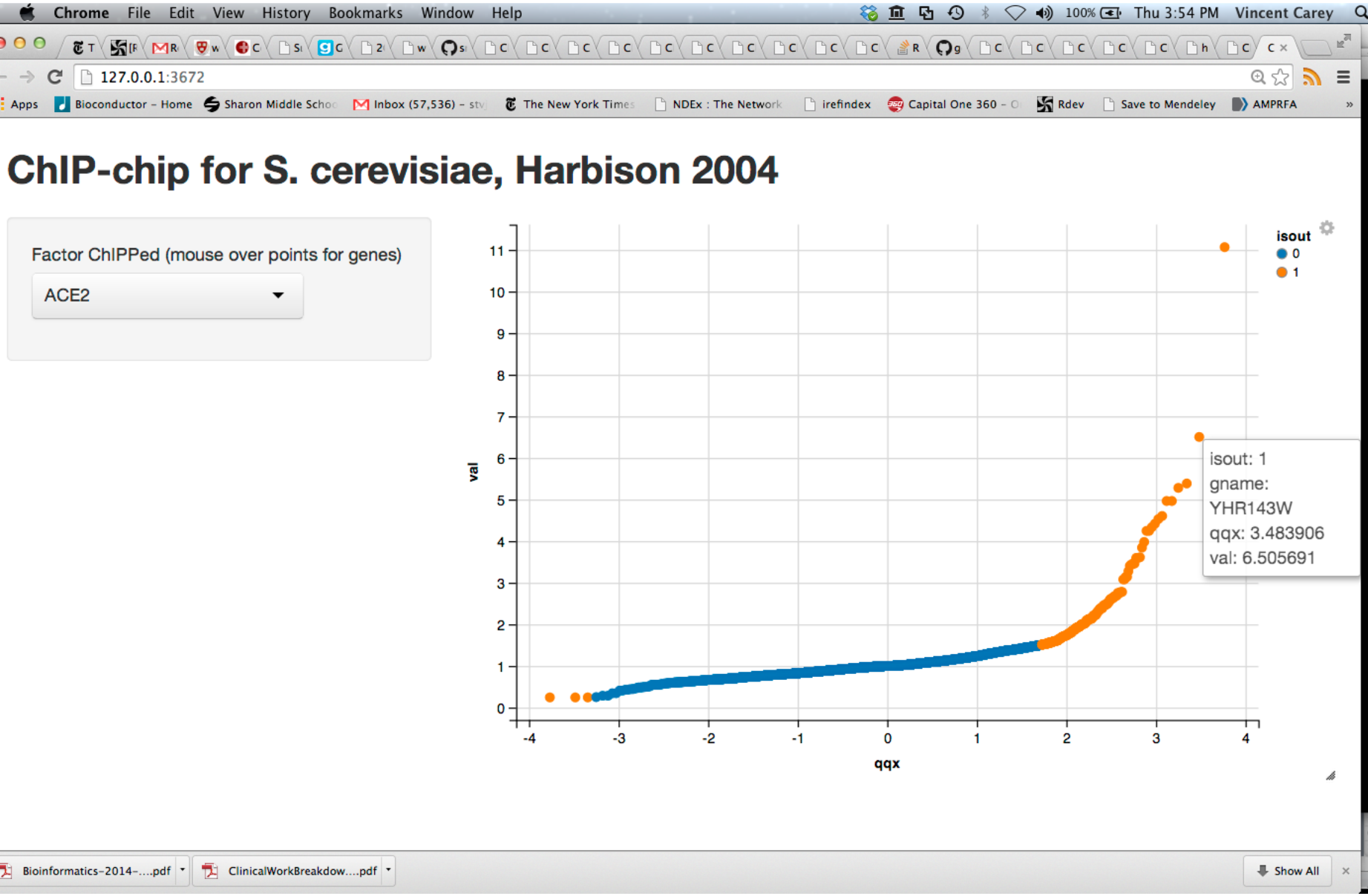


FIG 3. *An application of the model-view-controller pattern and pre-computed summaries to genomic visualization. Coverage is displayed at two levels of resolution (whole chromosome and the current zoom) after efficient extraction from the multi-resolution BigWig file. The BAM file holding the read alignments is abstracted by a multi-stage data model, consisti source, a dynamic read quality filter, and two filters that effectively split the alignm*

Function harbQQtool() in viz14dyn package



Controls can be added to appraise (is a Gaussian 'background' plausible?) and alter the model (how is an outlier boundary drawn?)

Summary

- Many packages offer plot and plot* methods for derived data structures
  - Think about 'reflectance' when choosing/designing data structures, to simplify effort in rendering
- Grammar of graphics concepts effectively modularize the analytical and aesthetic processes in visualization development
- ggvis + shiny is a potent combination, allowing us to get past static displays, to broaden and deepen our engagement with the data – if you are developing data-analytic methods, learning these will likely pay off