Scientific argumentation and software design

VJ Carey, Channing Lab, Harvard Medical School
BioC 2009

- Three case studies in cancer transcriptomics
- Containers
- Software reliability
- Scientific argumentation; Bioconductor's role/gaps

Possible take-home messages

- Bioc growth in software packages not matched by growth in experimental data packages
  - the concept is severely underappreciated
- Formally packaging data (for private development use) early on has various practical benefits
- A publication generated using a data package will satisfy many key reproducibility and maintainability requirements
- Packaging discipline can and should be adopted early in the analysis process
- also – EBImage allows you to take published figures and extract underlying numerical data, to reanalyze data that are numerically unavailable

Case study 1: Dressman JCO 2007

UMBER 5 · FEBRUARY 10 2007

CLINICAL ONCOLOGY

ORIGINAL REPORT

An Integrated Genomic-Based Approach to Individualized Treatment of Patients With Advanced-Stage Ovarian Cancer

Holly K. Dressman, Andrew Berchuck, Gina Chan, Jun Zhai, Andrea Bild, Robyn Sayer, Janiel Cragun, Jennifer Clarke, Regina S. Whitaker, LiHua Li, Jonathan Gray, Jeffrey Marks, Geoffrey S. Ginsburg, Anil Potti, Mike West, Joseph R. Nevins, and Johnathan M. Lancaster

ABSTRACT

Purpose

The purpose of this study was to develop an integrated genomic-based approach to personalized treatment of patients with advanced-stage ovarian cancer. We have used gene expression profiles to identify patients likely to be resistant to primary platinum-based chemotherapy and also to identify alternate targeted therapeutic options for patients with de novo platinum-resistant disease.

Patients and Methods

A gene expression model that predicts response to platinum-based therapy was developed using a training set of 83 advanced-stage serous ovarian cancers and tested on a 36-sample external validation set. In parallel, expression signatures that define the status of oncogenic signaling pathways were evaluated in 119 primary ovarian cancers and 12 ovarian cancer cell lines. In an effort to increase chemotherapy sensitivity, pathways shown to be activated in platinum-resistant cancers were subject to targeted therapy in ovarian cancer cell lines.

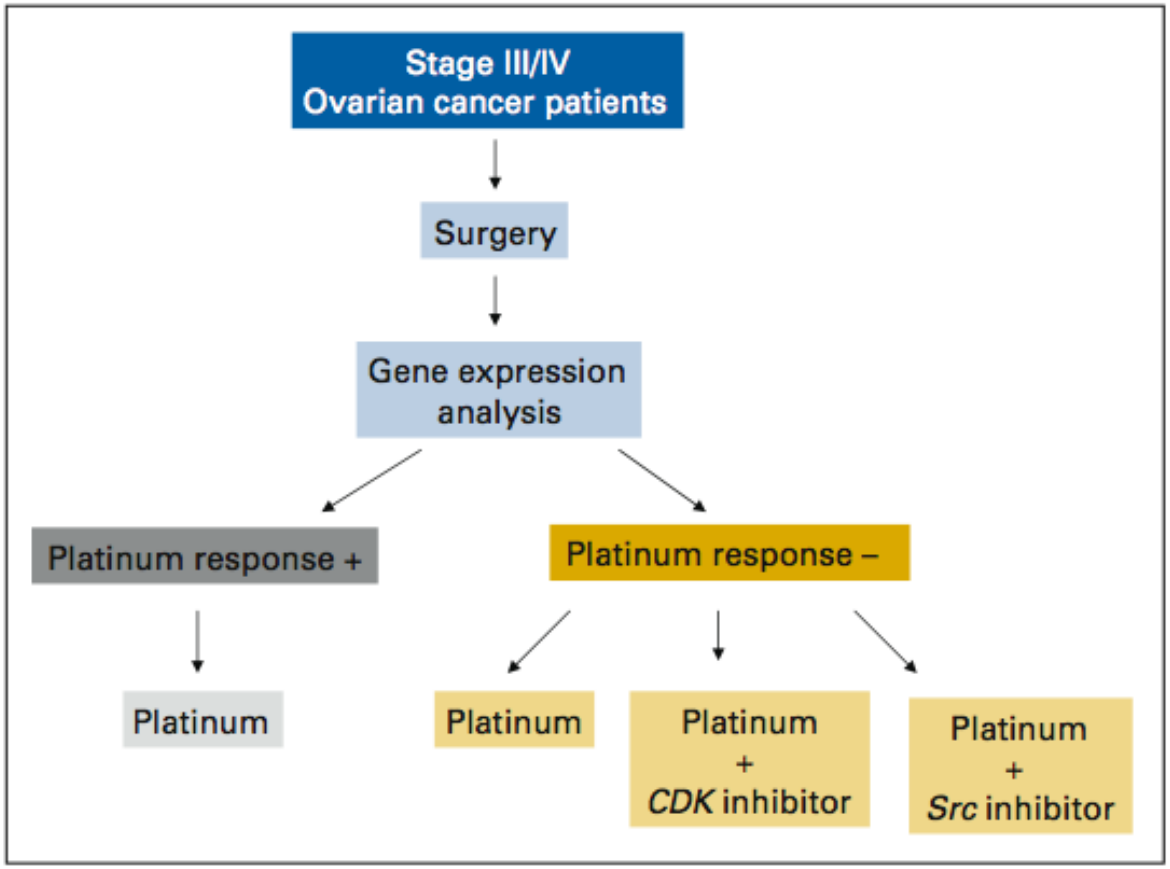
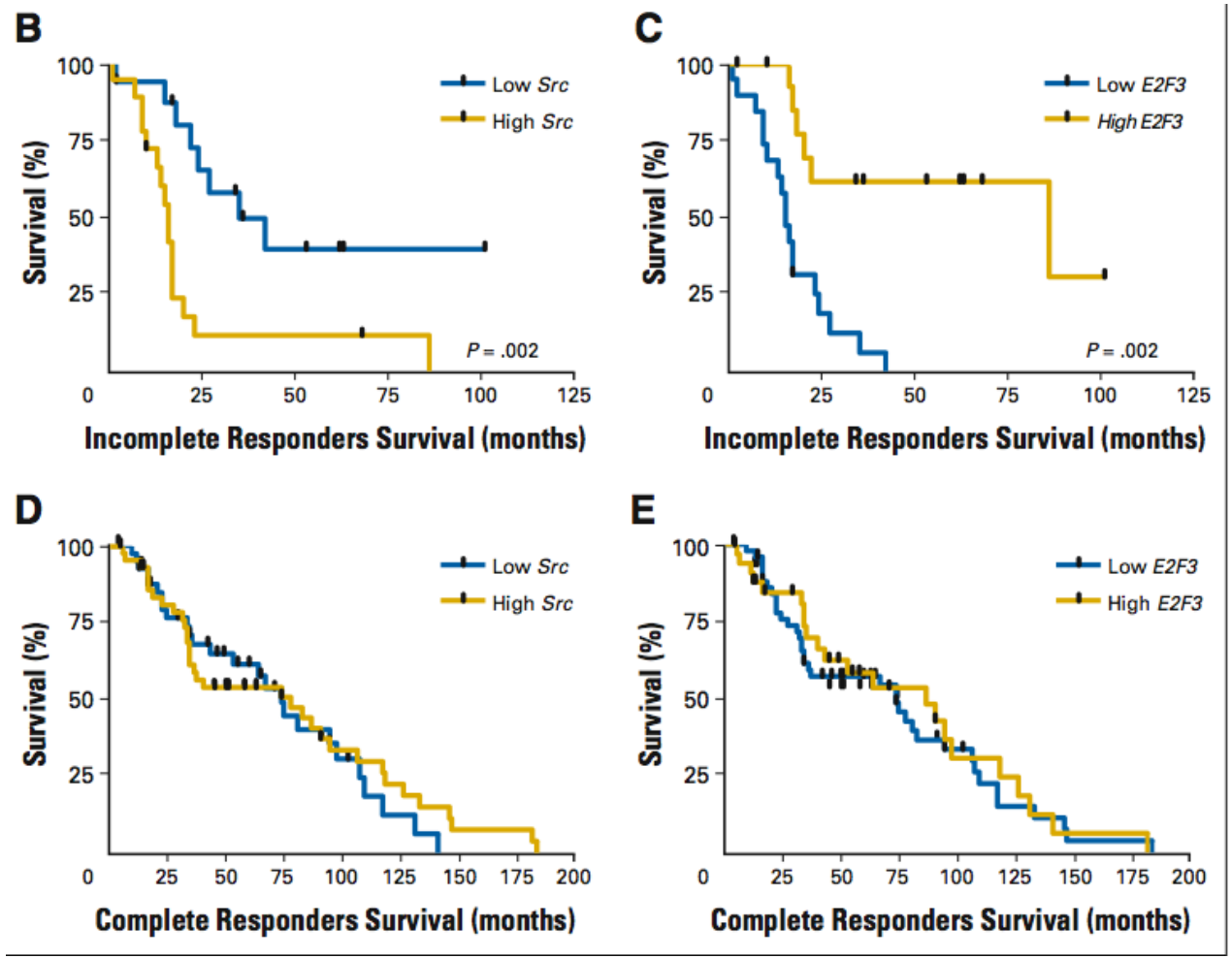
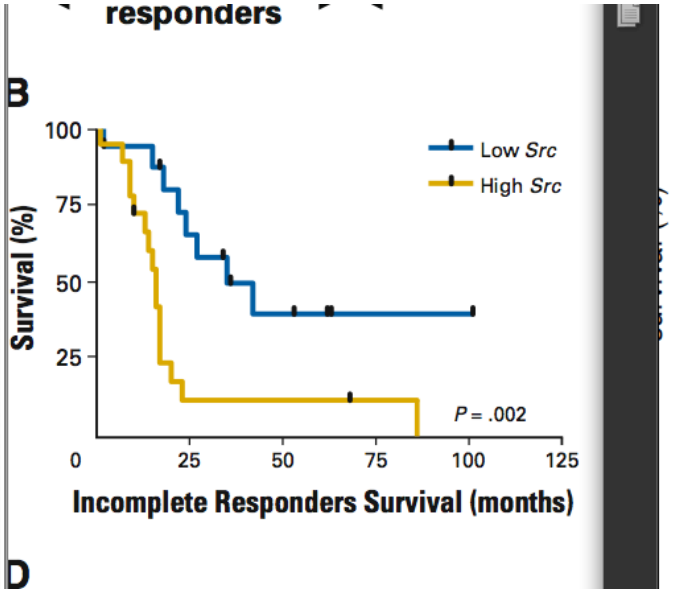


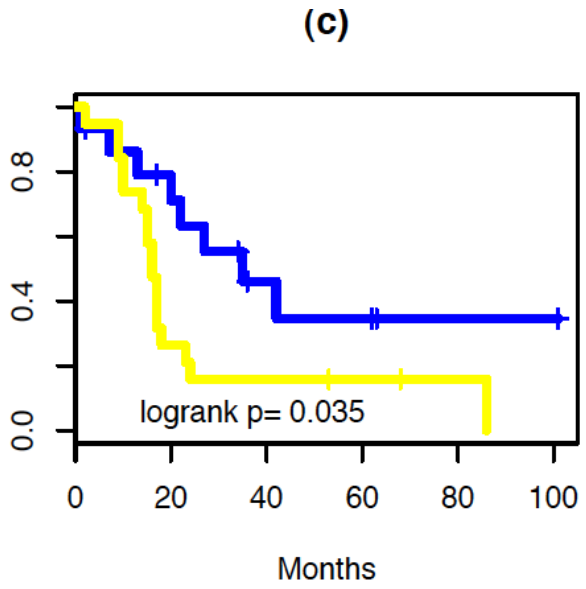
Fig 5. Potential application of platinum response and pathway prediction in the treatment of patients with ovarian cancer.

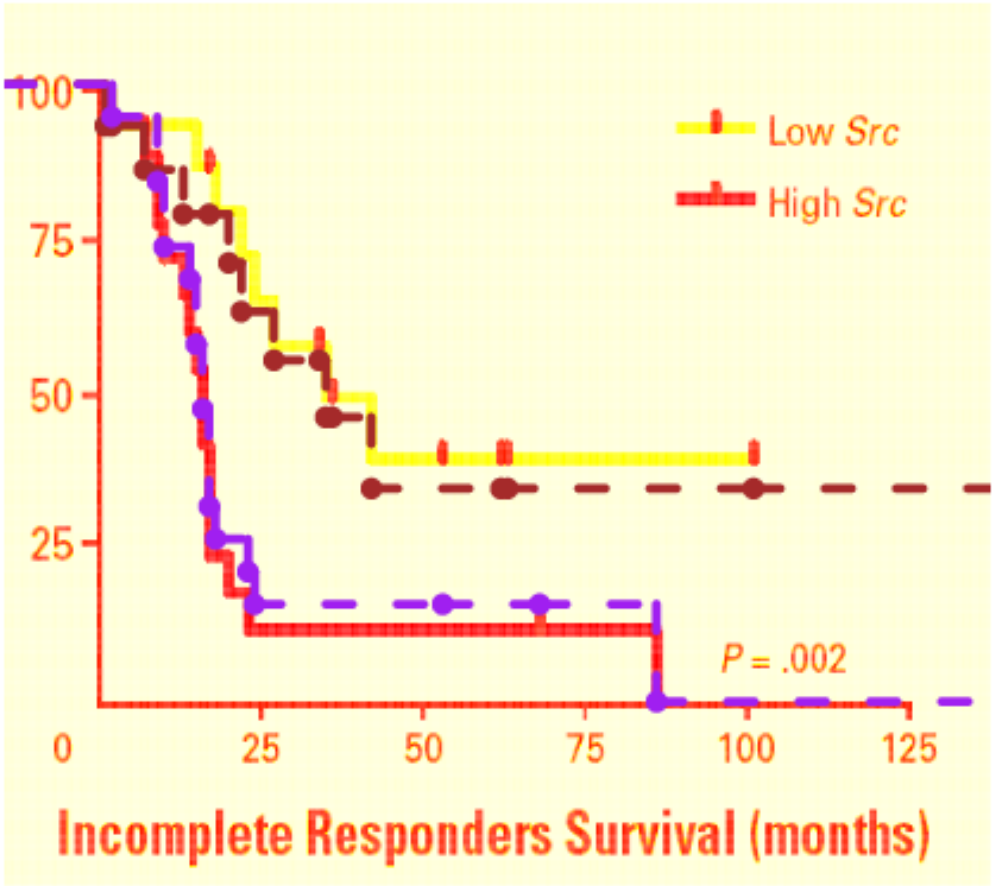

B
Survival (%)
100
75
50
25
0
25
50
75
100
125
Low Src
High Src
P = .002
Incomplete Responders Survival (months)
C
Survival (%)
Low E2F3
High E2F3
P = .002
Incomplete Responders Survival (months)
D
Survival (%)
Low Src
High Src
0
25
50
75
100
125
150
175
200
Complete Responders Survival (months)
E
Survival (%)
Low E2F3
High E2F3
Complete Responders Survival (months)

Irreproducibility following Baggerly et al. (letter, 7(!) vignettes)

- Much guesswork required to recompute based on supplemental data on Duke web site
- Sanity check – near reproduction of Src activation effect among platinum nonresponders

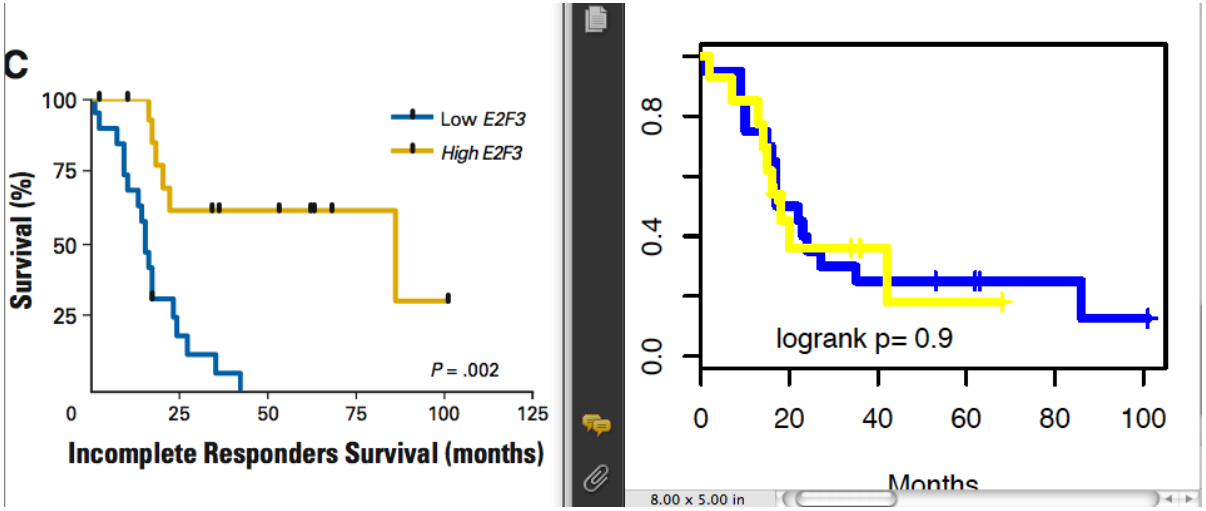





100
75
50
25
0
25
50
75
100
125
Low Src
High Src
P = .002
Incomplete Responders Survival (months)

Total non-reproducibility of asserted E2F3 effect

- Computations as with Src activation signature



Scientific interchange

- First inning:
  - Dressman et al: N=83 training, N=36 test samples, 12 cell lines, 1,727-gene predictive model, 2 heatmaps, 4 KM curves, 8 regressions, JCO paper, web site
  - Baggerly et al: 7 vignettes, over 100 supporting files and scripts, JCO letter (.6pp) with 7 major challenges to data and interpretation, web site
- Second inning:
  - Dressman et al: protest to the officials

    > While it is certainly important to present all information as accurately as possible, and we do regret the errors that were introduced when we generated several of the tables containing supplementary information, these errors do not affect the conclusions of

the study.

A focus on these errors as presented by Baggerly et al is misleading since it suggests they are a contributing factor in the supposed lack of reproducibility, which is not the case.

Most importantly, the claim that they cannot reproduce the results of the study, when in fact they did not even try to do so, is an egregious flaw in their commentary. To reproduce means to repeat, using the same methods of analysis as reported.

It does not mean to attempt to achieve the same goal of the study but with different methods.

- Bottom of the second inning: Texas bull-pen exhausted after seven vignettes; Duke pinch-hitters writing ARRA grants – rain delay

Summary of first case study

- Transcriptome-wide studies are complex – this is well-known
- Supplementary data are symbolically used to emulate wet lab open protocols
  - in this case there were/are errors making the data literally useless for reproducibility (unless forensic methods were adopted)
  - workflow leading to published artifacts (KM curves, heatmaps, regressions) not explicitly available
- The authors introduce an important obligation on primary authors to facilitate reproducibility:

  > To reproduce means to repeat, using the same methods of analysis as reported.

- If you purport to do reproducible research, you must facilitate independent *repetition*
- To satisfy this condition you must publish the data, the software, and the conditions of use and report extraction

Case study 2: Michiels random validation method

Prediction of cancer outcome with microarrays: a multiple random validation strategy

Stefan Michiels, Serge Koscielny, Catherine Hill

Summary

Background General studies of microarray gene-expression profiling have been undertaken to predict cancer outcome. Knowledge of this gene-expression profile or molecular signature should improve treatment of patients by allowing treatment to be tailored to the severity of the disease. We reanalysed data from the seven largest published studies that have attempted to predict prognosis of cancer patients on the basis of DNA microarray analysis.

Methods The standard strategy is to identify a molecular signature (ie, the subset of genes most differentially expressed in patients with different outcomes) in a training set of patients and to estimate the proportion of misclassifications with this signature on an independent validation set of patients. We expanded this strategy (based on unique training and validation sets) by using multiple random sets, to study the stability of the molecular signature and the proportion of misclassifications.

Findings The list of genes identified as predictors of prognosis was highly unstable; molecular signatures strongly depended on the selection of patients in the training sets. For all but one study, the proportion misclassified decreased as the number of patients in the training set increased. Because of inadequate validation, our chosen studies published overoptimistic results compared with those from our own analyses. Five of the seven studies did not classify patients better than chance.

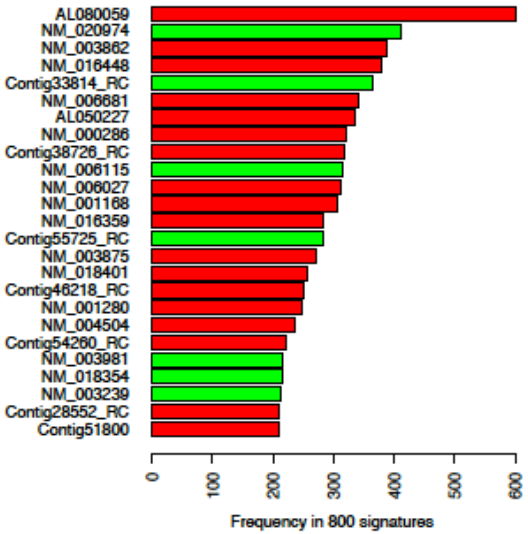
Interpretation The prognostic value of published microarray results in cancer studies should be considered with caution. We advocate the use of validation by repeated random sampling.

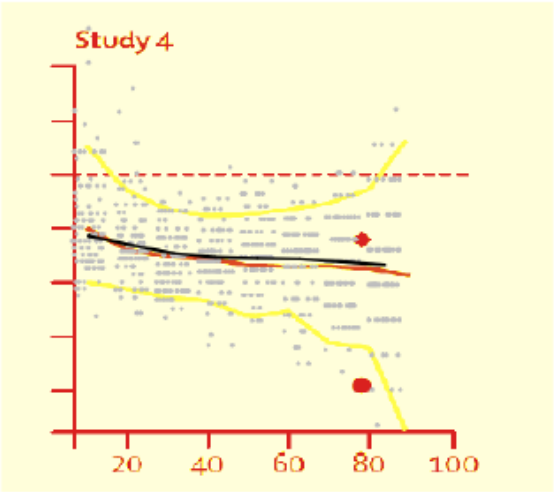
Strong impact

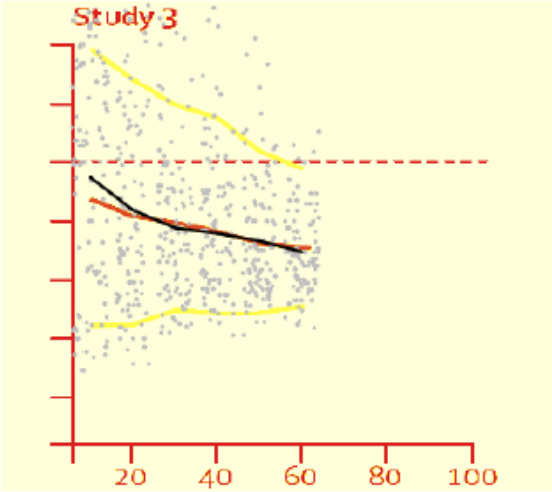
- Findings
  - Estimated signatures (test/train) are unstable
  - For a given dataset, estimated misclassification rates vary according to training set size
  - Five of seven major studies do not classify patients better than chance

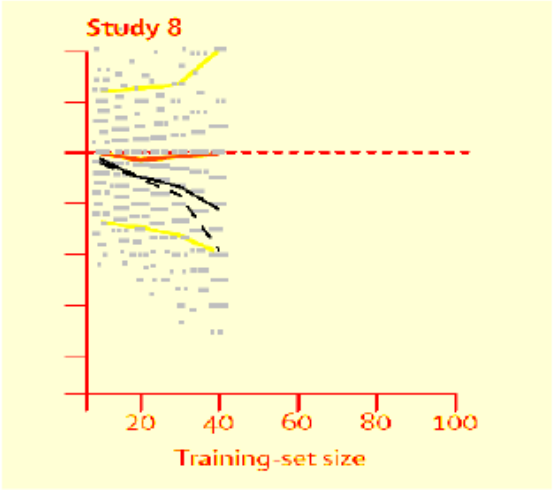
  Michiels paper has, as of April 29 2009, been cited 270 times (ISI Web of Science), with citations repeating concerns about "well-documented" signature instability, divergent results, and, in one case, indicating that microarray-based findings are "not robust to the mildest of perturbations" (Ramasamy and Mondry, 2008).

Two confirmations and a disconfirmation – black – my estimates of MC(T)









Summary on Michiels

- no explicit effort to foster reproducibility (no datasets, software, scripts)
- to explore the work we must solve problems of
  - data acquisition
  - algorithm implementation (fewer than 20 lines of reasonably generic R)
  - results juxtaposition (EBImage capture of TIFF extract of statistical graphics)
- What went wrong with Pomeroy? No clue.

Case study 3: Ben-Porath and 'stemness' of aggressive breast cancer tumors

An embryonic stem cell–like gene expression signature in poorly differentiated aggressive human tumors

Ittai Ben-Porath[1,2,5], Matthew W Thomson[3], Vincent J Carey[4], Ruping Ge[1], George W Bell[1], Aviv Regev[3] & Robert A Weinberg[1,2]

Cancer cells possess traits reminiscent of those ascribed to normal stem cells. It is unclear, however, whether these phenotypic similarities reflect the activity of common molecular pathways. Here, we analyze the enrichment patterns of gene sets associated with embryonic stem (ES) cell identity in the expression profiles of various human tumor types. We find that histologically poorly differentiated tumors show preferential overexpression of genes normally enriched in ES cells, combined with preferential repression of Polycomb-regulated genes. Moreover, activation targets of Nanog, Oct4, Sox2 and c-Myc are more frequently overexpressed in poorly differentiated tumors than in well-differentiated tumors. In breast cancers, this ES-like signature is associated with high-grade estrogen receptor (ER)-negative tumors, often of the basal-like subtype, and with poor clinical outcome. The ES signature is also present in poorly differentiated glioblastomas and bladder carcinomas. We identify a subset of ES cell-associated transcription regulators that are highly expressed in poorly differentiated tumors. Our results reveal a previously unknown link between genes associated with ES cell identity and the histopathological traits of tumors and support the possibility that these genes contribute to stem cell–like phenotypes shown by many tumors.

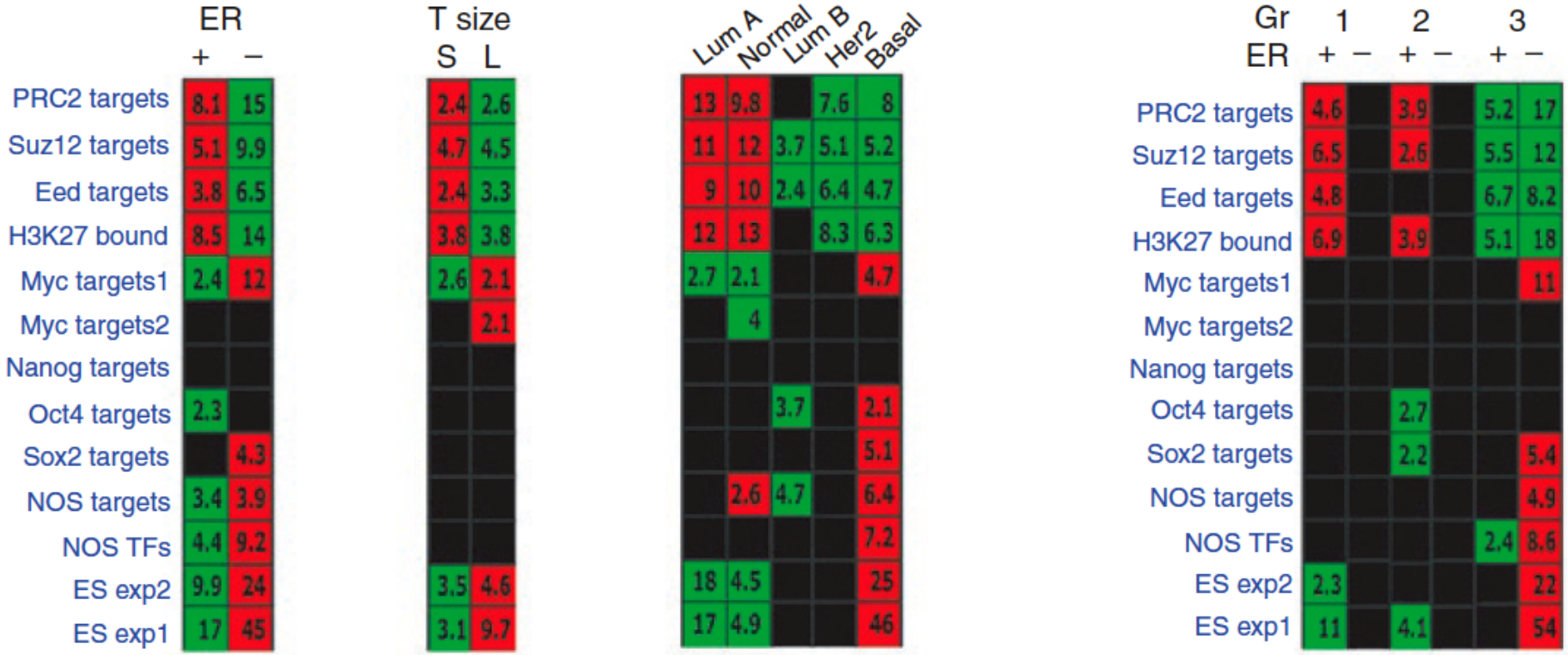
stem or progenitor cells or, alternatively, that cancer cells can undergo progressive de-differentiation during their development[1–3]. Additionally, some have proposed that cancer stem cells—a subpopulation of cancer cells possessing tumor-initiating capability—are derived from normal stem cells[1,4]. Although certain regulators of stem cell function have been implicated in cancer pathogenesis[2], a broad description of the activity of stem cell–associated regulatory networks in tumors is lacking.

The differentiation level (or grade) of human tumors is assessed routinely in the clinic, with poorly differentiated tumors generally having the worst prognoses. However, this classification is based on histopathological criteria, and the underlying molecular pathways controlling tumor differentiation are poorly described. Moreover, it is not known whether a lack of histological differentiation markers in tumor cells reflects the possession of stem cell–like traits. A number of oncogenes are known to interfere with normal cell differentiation, *myc* being a notable example[5,6], and such oncogenes could also affect tumor cell differentiation. The recent demonstration that adult fibroblasts can be reprogrammed into pluripotent ES-like cells[7,8] raises the possibility that the combined expression of stem cell–associated factors and specific oncogenes could also induce a nondifferentiated state in cancer cells. In fact, ectopic expression of Oct4, a central determinant of ES cell identity, is sufficient to induce tumor growth in the adult mouse[9], and Polycomb complex components central to stem cell

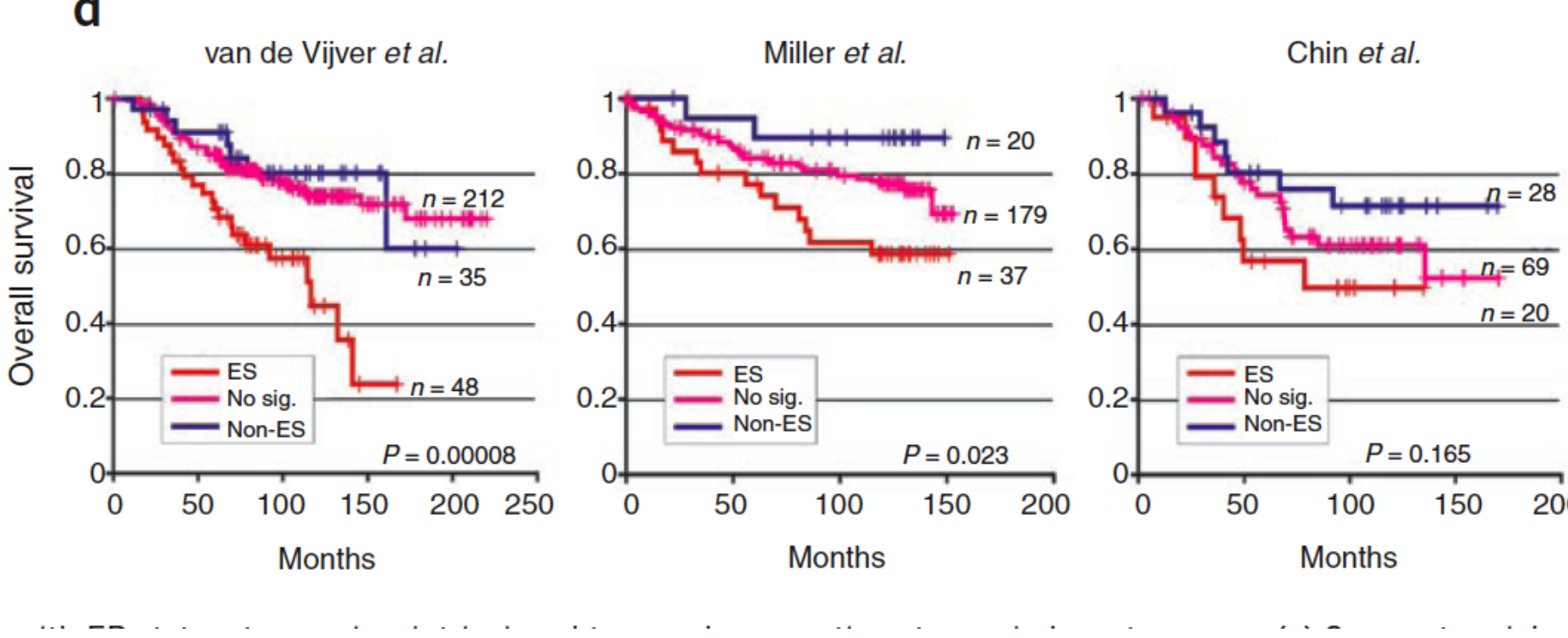
Data acquisition and translation

- Six studies: NEJM, JNCI, PNAS, Cancer Cell, Lancet, Clinical Cancer Research
- Some overlapping cases identified and removed; harmonize phenotypic labeling
- Translate reporters in use to Entrez Gene ids
- Form knowledge-based gene sets

Observation: Specific clinical classes associated with high relative expression of ES and allied gene sets



Metaanalytic $p < 0.0001$ for ES+ effect on survival



Reproducibility considerations

- All processed expression data and clinical annotations are on line
- Genomica settings?
- Inheritance of nonreproducibility from foundational studies?
- Cost to make one of the survival figures independently reproducible? Probably low, but yet to be attempted...
- What are the benefits? More restful sleep, easier extension by other researchers, introduction of versioning

Summary of case studies

- Commitments to concrete reproducibility of computational analysis of microarrays are generally weak or nonexistent
- Data-sharing obligations may be symbolically met without verification
- Strong positive and negative claims move into literature and medicine regardless of demonstrable unreliability
- Incentives for supporting independent repetition of analyses are weak
  - can be difficult
  - maybe no one will ever use it

Containers – main families

- Metadata: databases, tables, tracks; annotation networks
- Data: tables, objects, packages
- Software: functions, methods under OOP discipline, packages
- Workflows culminating in arguments/manuscripts – container concept not well-adapted to these?
- Web services: assume good container designs for all the constituents above
- Bioconductor premise: support meaningful work on commodity hardware without assuming WWW connectivity – compact computational environment, counter to apparently prevailing view that a web site can suffice for supporting reproducibility

Containers – main properties

- self-describing, self-documenting
- contents have guaranteed structure/datatypes
- API – formal specs on feasible interrogations and range of replies
- can be programmatically validated

Containers – examples in Bioconductor

- metadata: AnnDbBimap (used for org.Hs.eg.db, GO.db), GeneSet (GSEABase), data.frame (SNPlocs.Hsapiens), PDInfoPkgSeed (pd.genomewidesnp.6) – and the associated packages themselves
- data: eSet, ExpressionSet, exCGHset, snp.matrix, smlSet, methylumiSet, graph
- software: packages, task views
- workflow culminating in argument: Sweave vignette

methylumi container examples –GoldenGate

```
> mldat

Object Information:
MethyLumiSet (storageMode: environment)
assayData: 1536 features, 10 samples
  element names: Avg_NBEADS, BEAD_STDERR, betas, methylated, pvals,
phenoData
  sampleNames: M_1, M_2, ..., F_10  (10 total)
  varLabels and varMetadata description:
    sampleID: sampleID
    SampleLabel: SampleLabel
    Sample: Sample
    Gender: Gender
featureData
  featureNames: AATK_E63_R, AATK_P519_R, ..., ZP3_P220_F  (1536 tota
  fvarLabels and fvarMetadata description:
    TargetID: NA
    ProbeID: NA
```

```
    ...: ...
    PRODUCT: NA
    (17 total)
experimentData: use 'experimentData(object)'
Annotation:
Major Operation History:
             submitted            finished
1 2009-07-28 00:33:29 2009-07-28 00:33:30

1 methylumiR(filename = system.file("extdata/exampledata.samples.txt
```

```
> fData(mldat)[100:101, ]
                   TargetID ProbeID SEARCH_KEY      PROBE_ID      GID
BCL2L2_E172_F BCL2L2_E172_F    5384     BCL2L2 BCL2L2_E172_F 14574571
BCL2L2_P280_F BCL2L2_P280_F    4235     BCL2L2 BCL2L2_P280_F 14574571
                ACCESSION SYMBOL GENE_ID CHROMOSOME REFSEQ CPG_COORDINATE
BCL2L2_E172_F NM_004050.2 BCL2L2     599         14   36.1       22846038
BCL2L2_P280_F NM_004050.2 BCL2L2     599         14   36.1       22845586
              DIST_TO_TSS CPG_ISLAND
BCL2L2_E172_F         172          N
BCL2L2_P280_F        -280          N
                                                          INPUT_SEQUENCE
BCL2L2_E172_F                TTGGGCTGCACTAGGGGGAACCGGGAATAGAGATGGTGTCGG
BCL2L2_P280_F CTGGAAAAGTTCAACAAGTGCATGGAACATCGGAAACCTCCTGAAAATGCTAAATT
                           SYNONYM
BCL2L2_E172_F BCLW, BCL-W, KIAA0271
BCL2L2_P280_F BCLW, BCL-W, KIAA0271

BCL2L2_E172_F apoptosis regulator BCL-W; go_component: membrane; go_component:
BCL2L2_P280_F apoptosis regulator BCL-W; go_component: membrane; go_component:
                            PRODUCT
BCL2L2_E172_F BCL2-like 2 protein
```

BCL2L2_P280_F BCL2-like 2 protein

Containers and sanity checks – do Illumina's input CPG addresses agree with hg18?

```
> c14 = Hsapiens$chr14

> matchPattern(as.character(fData(mldat)["BCL2L2_E172_F",
                                          "INPUT_SEQUENCE"]), c14)
  Views on a 106368585-letter DNAString subject
subject: NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN...NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
views:
       start      end width
[1] 22846017 22846058    42 [TTGGGCTGCACTAGGGGGAACCGGGAATAGAGATGGTGTCGG]

> matchPattern(as.character(fData(mldat)["BCL2L2_P280_F",
                                          "INPUT_SEQUENCE"]), c14)
  Views on a 106368585-letter DNAString subject
subject: NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN...NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
views:
       start      end width
[1] 22845556 22845611    56 [CTGGAAAAGTTCAACAAGTGCATG...AAACCTCCTGAAAATGCTAAAT

> GGpdict = PDict(substr(as.character(fData(mldat)[,"INPUT_SEQUENCE"]),1,41))
> GGpdict
```

```
TB_PDict object of length 1536 and width 41 (preprocessing algo="ACtree2")
> sum(countPDict(GGpdict, c14))
[1] 37
> sum(fData(mldat)[,"CHROMOSOME"]==14)
[1] 37
```

```
> getValidity(getClass(class(mldat)))

function (object)
{
    msg <- Biobase:::validMsg(NULL, Biobase:::isValidVersion(object,
        "eSet"))
    msg <- Biobase:::validMsg(msg, assayDataValidMembers(assayData(object),
        c("betas")))
    if (is.null(msg))
        TRUE
    else msg
}
<environment: namespace:methylumi>
```

Foils to the container discipline

- Common student request after learning about ExpressionSets: These are very nice but how do I get the data out to disk so I can run my perl programs on them?
- MLInterfaces: explicitly addresses holistic use of containers but ignored by colleagues in favor of matrix exports to functions
- formulae:

  `y ~ f(x1, x2, ...) | g(z1, z2, ...)`

  powerful idiom, can bind gene, gene sets, SNPs, addresses, sequences, graphs to elements as desired – not often used; instead, take data out of the container as vectors and put into ordinary formulae
- interface contracts: e.g., predict(), resid() should be im-

plemented wherever suitable

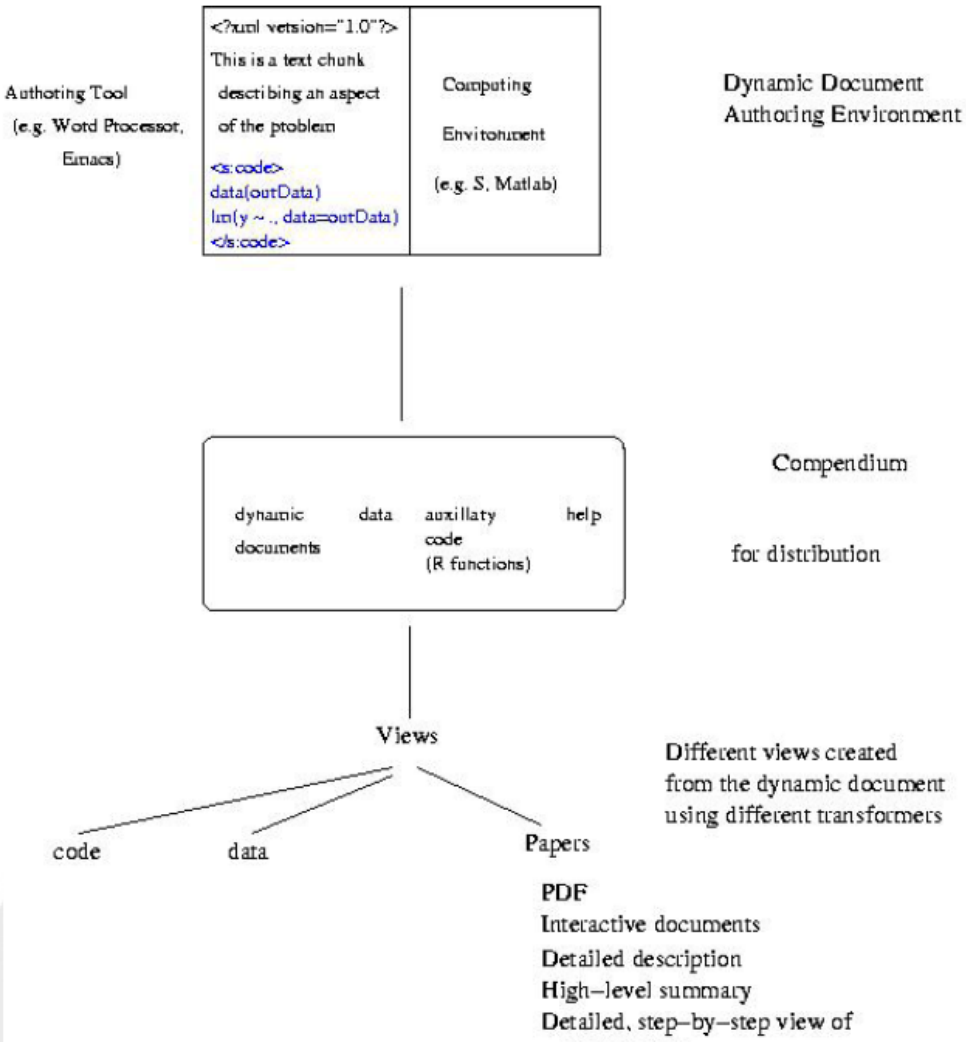
Software reliability and reliability of scientific argument

- software reliability measures
  - face validity (usually not realistic)
  - unit testing results (important, biased)
  - recovery of truth in applications where truth is known (good, rare in bioinformatics)
  - interoperability, success of integrated applications (good, self-consistent behavior established, truth very rarely known)
- Disconnect: reliable software and reliable scientific argument
  - Case study 1: each analytic module may have been sound, but processes similar to the admitted mislabeling of records seem to have wrecked the reproducibility of the research
  - Case study 2: computations are simple, but an important application seems to be mistaken; 270+ citations, no audit
  - Case study 3: curves/p-value implications; integrative analysis may inherit errors in previous work of others

– The established vulnerabilities are not in the “killer (methodological) applications” – those get reasonable testing

What would a solution look like?

- Every step of the workflow needs to be amenable to testing and verification
- A 'compendium' in the sense of Gentleman and Temple Lang (SAGMB 2005) combines all needed data and software – a general protocol with emphasis on dynamic content, entity that can be interrogated for provenance of any data reference, and modified to alter any computation
- An R package with functions and scripts specifying computational basis for every data reference in the manuscript
- an equivalent construction based on some other language


Authoring Tool
(e.g. Word Processor, Emacs)
<?xml version="1.0"?>
This is a text chunk describing an aspect of the problem
<s:code>
data(outData)
lm(y ~ ., data=outData)
</s:code>
Computing Environment
(e.g. S, Matlab)
Dynamic Document Authoring Environment
dynamic documents
data
auxillary code (R functions)
help
Compendium
for distribution
Views
code
data
Papers
Different views created from the dynamic document using different transformers
PDF
Interactive documents
Detailed description
High–level summary
Detailed, step–by–step view of

	R	Perl	Python
Document Format	XML or Sweave	XML or POD	XML
Skeleton	`package.skeleton`	`h2xs`	
Distribution unit	R package	Perl module	Python module
Distribution Mechanism	Repository tools	CPAN	Vaults of Parnassus
Installation	`R CMD INSTALL`	`perl Makefile.PL make install`	`python setup.py install`
Test Command	`R CMD check`	`make test`	`python unittest.py file`
Test Tools	`tools` package	`Test` module	`PyUnit`

Table 1: The tools in the different languages R, Perl, Python available to author, create, manage and distribute a compendium.

Conclusions

- Independent reproducibility is a basic requirement
- Reproducing errors is of no interest, but errors will occur; ergo versioning is essential
- (R) Packaging discipline is an aid to reproducibility
- Package construction/maintenance should begin early in the analysis cycle – preferably at data capture/QA activities
- Versioning, self-describing character, portability secured by any package that is reasonably faithful to WRE guidelines
- Interoperability can be used to avoid challenges due to data volume (SQLite, web service access, caching of intermediate data)
- Each investigator needs to commit to the discipline for their own sake – benefits arguably exceed costs every step of the way
- bioc should take the lead by publicizing exemplars in the hardest areas (data packages with vignettes that recover/disconfirm accepted findings)