Writing our first Bioconductor package as members of the CDSB community

Joselyn Chávez, Carmina Barberena Jonas, Emiliano Sotelo

A recap of the Community of Bioinformatics Software Developers (CDSB in Spanish)

Founders

Leonardo Collado-Torres, PhD

Research Scientist

Genomics, R programming, Biostatistics, Teaching, Diversity

Alejandro Reyes, PhD

Genomic Data Scientist / Postdoc

Data Science, Genomics, R

Delfino García-Alonso

Laboratory Technician

Bioinformatics

Alejandra Medina Rivera, PhD

Investigator

Gene regulation, Bioinformatics

Heladia Salgado Osorio

Laboratory Technician

Bioinformatics, Teaching

Board

Leonardo Collado-Torres, PhD

Research Scientist

Genomics, R programming, Biostatistics, Teaching, Diversity

Alejandro Reyes, PhD

Genomic Data Scientist / Postdoc

Data Science, Genomics, R

Alejandra Medina Rivera, PhD

Investigator

Gene regulation, Bioinformatics

Heladia Salgado Osorio

Laboratory Technician

Bioinformatics, Teaching

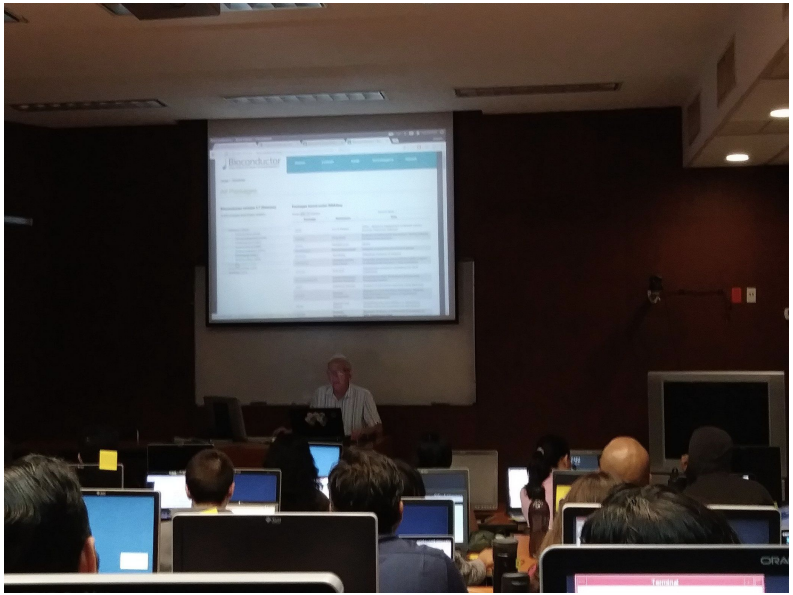
Joselyn Chavez, Ph.D. Candidate

Ph.D. Candidate

Bioinformatics, R programming, Bioconductor, Genetics

Events held by the CDSB

Workshop 2018: Latin American R/BioConductor Developers Workshop



Workshop 2019: How to Build and Create Tidy Tools

What is regutools?

RegulonDB

Transcriptional regulation and transcriptional networks in *E. coli*.

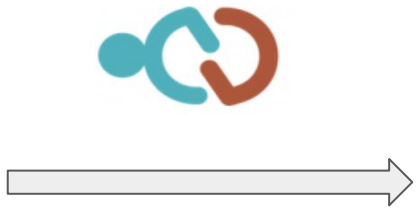
Bioconductor
OPEN SOURCE SOFTWARE FOR BIOINFORMATICS

How it started?

Licenciatura en
CIENCIAS
GENÓMICAS

What we had at this point

- Functions
- SQLite database



Building regutools as a package

- Functions improvement
- Documentation
- Vignette
- Tests
- Integrated workflow

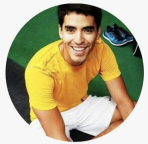
regutools team

Developers and Mentors

Leonardo Collado-Torres, PhD

Research Scientist

Genomics, R programming, Biostatistics, Teaching, Diversity

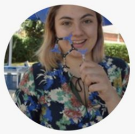


Alejandro Reyes, PhD

Genomic Data Scientist / Postdoc

Data Science, Genomics, R

Developer Alumni



Carmina Barberena-Jonas

Student Intern

Mexican Biobank, Bioinformatics, R programming, Bioconductor, Photography, Surrealist paintings

Jesus Emiliano Sotelo-Fonseca

MSc student

Plant Biotechnology, Bioinformatics, R programming, Bioconductor

Joselyn Chavez, Ph.D. Candidate

Ph.D. Candidate

Bioinformatics, R programming, Bioconductor, Genetics

Regulondb Maintainer

Heladia Salgado Osorio

Laboratory Technician

Bioinformatics, Teaching

What can you do with regutools?

- Connect to the RegulonDB database

```
regulondb_conn <- connect_database()
```

- Build a new object defined as a regulondb object

```
e_coli_regulondb <-
    regulondb(
        database_conn = regulondb_conn,
        organism = "E.coli",
        database_version = "1",
        genome_version = "1"
    )
```

What can you do with regutools?

- List datasets contained in the RegulonDB database

```
list_datasets(e_coli_regulondb)
#> [1] "DNA_OBJECTS"        "GENE"               "NETWORK"
#> [4] "OPERON"             "PROMOTER"           "REGULONDB_OBJECTS"
#> [7] "TF"                 "TU"
```

- List columns called attributes from the datasets

```
head(list_attributes(e_coli_regulondb, "GENE"), 8)
#> [1] "id"       "name"     "bnumber"  "gi"       "synonyms" "posleft"  "posri
ght"
#> [8] "strand"
```

What can you do with regutools?

- Retrieve and filter data

```
get_dataset(
    regulondb = e_coli_regulondb,
    dataset = "GENE",
    attributes = c("posleft", "posright", "strand", "name"),
    filters = list("name" = c("araC", "crp", "lacI"))
)
#> regulondb_result with 3 rows and 4 columns
#>     posleft  posright      strand        name
#>   <integer> <integer> <character> <character>
#> 1     70387     71265     forward        araC
#> 2   3486120   3486752     forward         crp
#> 3    366428    367510     reverse        lacI
```

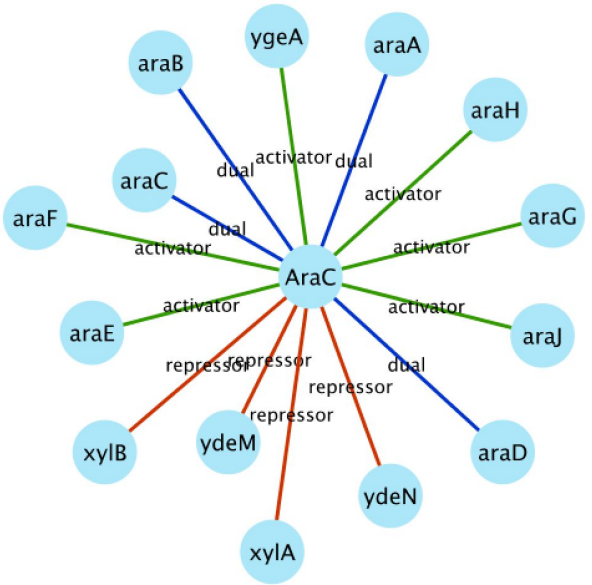
Importantly, the result of each function is by default a regulondb object which keeps the slots from the original object.

```
res <- get_dataset(
    regulondb = e_coli_regulondb,
    dataset = "GENE",
    attributes = c("posleft", "posright", "strand", "name"),
    filters = list("name" = c("araC", "crp", "lacI"))
)
slotNames(res)
#>  [1] "organism"         "genome_version"   "database_version" "dataset"
#>  [5] "rownames"         "nrows"            "listData"         "elementType"
#>  [9] "elementMetadata"  "metadata"
```

What can you do with regutools?

- Extract and visualize regulatory networks

```
get_gene_regulators(e_coli_regulondb, c("araC", "fis", "crp"))
#> regulondb_result with 9 rows and 3 columns
#>         genes  regulators      effect
#>   <character> <character> <character>
#> 1         crp         Fis           -
#> 2         fis         Fis           -
#> 3        araC         CRP           +
#> 4         crp         CRP         +/-
#> 5         fis         CRP         +/-
#> 6        araC        AraC         +/-
#> 7         crp         Cra           +
#> 8        araC        XylR           -
#> 9         fis         IHF           +
```



Cytoscape

What can you do with regutools?

- Search binding sites and retrieve them in multiple formats.

```
get_binding_sites(e_coli_regulondb, transcription_factor = "AraC")
#> GRanges object with 15 ranges and 1 metadata column:
#>                         seqnames          ranges strand |
#>                            <Rle>       <IRanges>  <Rle> |
#>   ECK120015742-araB-araC      chr     70110-70126      + |
#>   ECK120012328-araB-araC      chr     70131-70147      + |
#>   ECK120012320-araB-araC      chr     70184-70200      - |
#>   ECK120012323-araB-araC      chr     70205-70221      - |
#>   ECK120012603-araB-araC      chr     70342-70358      - |
#>                      ...      ...             ...    ... .
#>        ECK120012333-araF      chr 1986396-1986412      - |
#>        ECK120012915-araE      chr 2982244-2982260      - |
#>        ECK120012913-araE      chr 2982265-2982281      - |
#>        ECK125108641-xylA      chr 3730824-3730840      - |
#>        ECK125108643-xylA      chr 3730847-3730863      - |
#>                                                  sequence
#>                                                 <character>
#>   ECK120015742-araB-araC ataaaaagcgTCAGGTAGGATCCGCTAatcttatgga
#>   ECK120012328-araB-araC ccgctaatctTATGGATAAAAATGCTAtggcatagca
#>   ECK120012320-araB-araC tctataatcaCGGCAGAAAAGTCCACAttgattattt
#>   ECK120012323-araB-araC caaaaacgcgTAACAAAAGTGTCTATAatcacggcag
#>   ECK120012603-araB-araC attcagagaaGAAACCAATTGTCCATAttgcatcaga
#>                      ...                                 ...
#>        ECK120012333-araF ccaaagacaaCAAGGATTTCCAGGCTAatcttatgga
#>        ECK120012915-araE tccatatttaTGCTGTTTCCGACCTGAcacctgcgtg
#>        ECK120012913-araE cgacatgtcgCAGCAATTTAATCCATAtttatgctgt
#>        ECK125108641-xylA taacataattGAGCAACTGAAAGGGAGtgcccaatat
#>        ECK125108643-xylA attatctcaaTAGCAGTGTGAAATAACataattgagc
#>   -------
```

```
get_binding_sites(e_coli_regulondb,
    transcription_factor = "AraC",
    output_format = "Biostrings")
#>   A DNAStringSet instance of length 15
#>      width seq                                               names
#>  [1]    37 ATAAAAAGCGTCAGGTAGGATCCGCTAATCTTATGGA            ECK120015742-ara
B...
#>  [2]    37 CCGCTAATCTTATGGATAAAAATGCTATGGCATAGCA            ECK120012328-ara
B...
#>  [3]    37 TCTATAATCACGGCAGAAAAGTCCACATTGATTATTT            ECK120012320-ara
B...
#>  [4]    37 CAAAAACGCGTAACAAAAGTGTCTATAATCACGGCAG            ECK120012323-ara
B...
#>  [5]    37 ATTCAGAGAAGAAACCAATTGTCCATATTGCATCAGA            ECK120012603-ara
B...
#>  ...   ... ...
#> [11]    37 CCAAAGACAACAAGGATTTCCAGGCTAATCTTATGGA            ECK120012333-ara
F
#> [12]    37 TCCATATTTATGCTGTTTCCGACCTGACACCTGCGTG            ECK120012915-ara
E
#> [13]    37 CGACATGTCGCAGCAATTTAATCCATATTTATGCTGT            ECK120012913-ara
E
#> [14]    37 TAACATAATTGAGCAACTGAAAGGGAGTGCCCAATAT            ECK125108641-xyl
A
#> [15]    37 ATTATCTCAATAGCAGTGTGAAATAACATAATTGAGC            ECK125108643-xyl
A
```

Things we learned

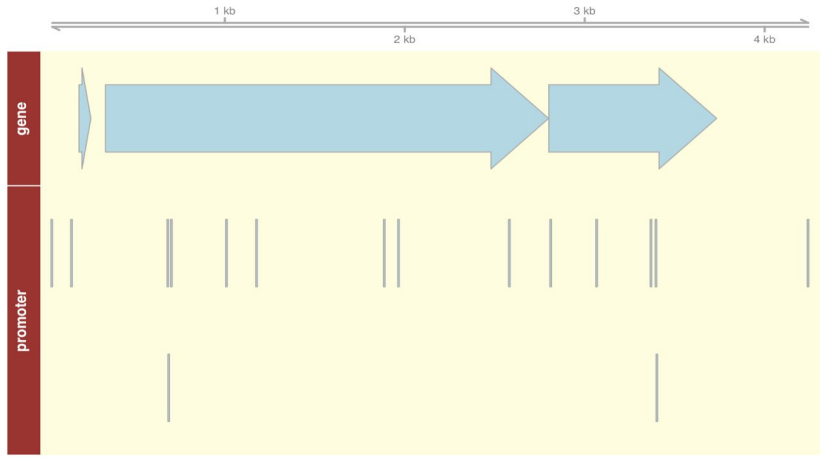
Joselyn:

- Modifying parameters into a function implies to run and sometimes update tests.
- Implementing Travis CI App (Thanks to Leo) makes a big difference to test code.
- It is better to write separate functions when we expect VERY different outputs.

That was how these two functions born

```
get_dna_objects(e_coli_regulondb, grange , elements = c("gene", "promoter"))
#> GRanges object with 19 ranges and 4 metadata columns:
#>        seqnames    ranges strand |          id        type
#>           <Rle> <IRanges>  <Rle> |  <character> <character>
#>    [1]   E.coli  337-2799      + | ECK120000987        gene
#>    [2]   E.coli 2801-3733      + | ECK120000988        gene
#>    [3]   E.coli   190-255      + | ECK120001251        gene
#>    [4]   E.coli       148      + | ECK120010236    promoter
#>    [5]   E.coli        38      + | ECK125230824    promoter
```

```
plot_dna_objects(e_coli_regulondb, grange, elements = c("gene", "promoter"))
```



Integration with Gviz

Things we learned

Emiliano:

- Working on a coding project collaboratively using github, slack.
- Using R developer tools: devtools::test_coverage() makes writing unit tests a game.

Carmina:

- Writing the code it's an important part of development but it's not all!

The experience of submitting regutools to Bioconductor

We used guidelines to know important facts about the submitting process like:

- There is a developers mail list.
- How to create a SSH key to Github.

But, the experience and guide from Leonardo and Alejandro was crucial to perform the submission process and understand build reports.

Feedback during review process

Some fixes:

- Keep just one maintainer.
- Remove the .Rproj file.
- Add the NEWS file.
- Adjust lines length and indentation.

Good comments:

R

- Nicely done! Well written code.
- Try line wrapping certain functions to avoid the 80 chars per line NOTE on the build machine.

vignette

- Good!

Thanks a lot Nitesh!

Current status of regutools

Almost done but dealing with a Warning in the R CMD check

```
Status: OK

 WARNING: R CMD check exceeded 20 min requirement
```

Final thoughts

The development process has been very rewarding as a collaborative and learning experience.

We hope regutools will be a very useful tool for projects related with microbiological studies.