

ENSAMBLE DE TRANSCRIPTOMA DE NOVO

M. En C. Verónica Jiménez Jacinto
veronica.jimenez@ibt.unam.mx

Enero 2025



UUSMB
UNIDAD UNIVERSITARIA DE
SECUENCIACIÓN MASIVA Y BIOINFORMÁTIC



Instituto Nacional
de Salud Pública



Objetivos a lograr

- * Explicar que es un transcripto de Novo.
- * Conocer y trabajar con una herramienta de reconstrucción de transcriptomas de novo, tanto para organismos procariontes y eucariontes.
- * Utilizarán varios parámetros que nos permitan ensamblar secuencias de datos tipo RNAseq de la manera más pertinente
- * Determinar la abundancia de cada uno de los transcritos reconstruidos y en los casos de organismos eucariontes, determinar la existencia de isoformas.



TEMARIO

1. Introducción a las metodologías de transcriptoma.
2. Aplicaciones
3. Flujo de trabajo
4. Linea de comando típica de Trinity
5. Monitoreo del progreso de trinity
6. Revisando la salida del ensamblado (Output Trinity Assembly)
7. Cuantificación de los transcritos
8. Análisis de Expresión Diferencial (ideamex)



1. Introducción al ensamblado de Novo de transcritos



El lenguaje genético

En el lenguaje genético, la organización de los genes (las palabras) y su regulación (el tiempo y la manera en que las frases son leídas o bien no leídas) determina los distintos tipos celulares que forman por ejemplo, el corazón, el riñón y el cerebro.

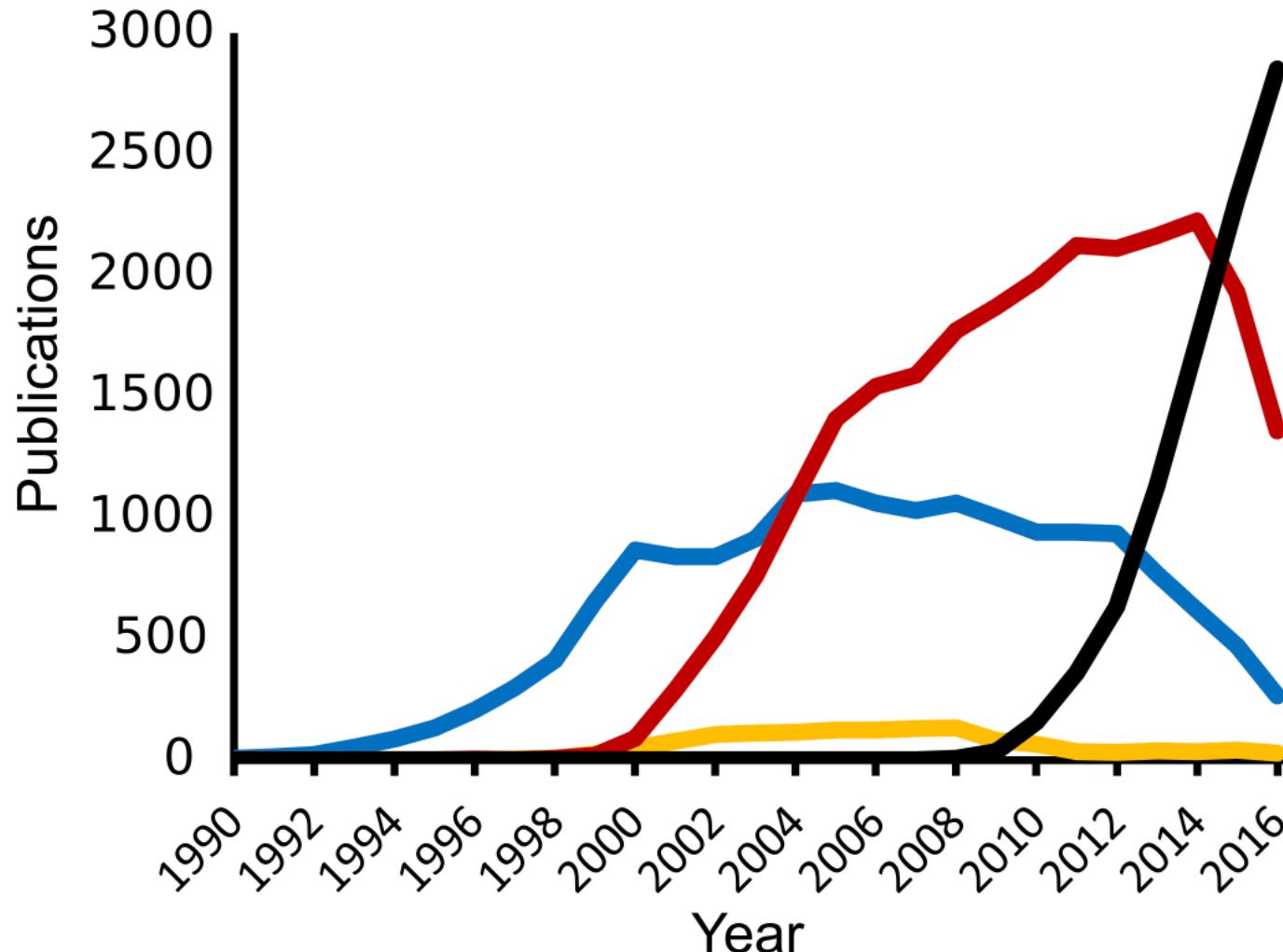
Al igual que las luces de un árbol de navidad, los genes de las células pueden estar encendidos o apagados alternativamente de acuerdo a un programa patrón. Algunos genes se apagan y encienden de manera intermitente, otros siempre están apagados, otros solo están encendidos por un corto periodo de tiempo.



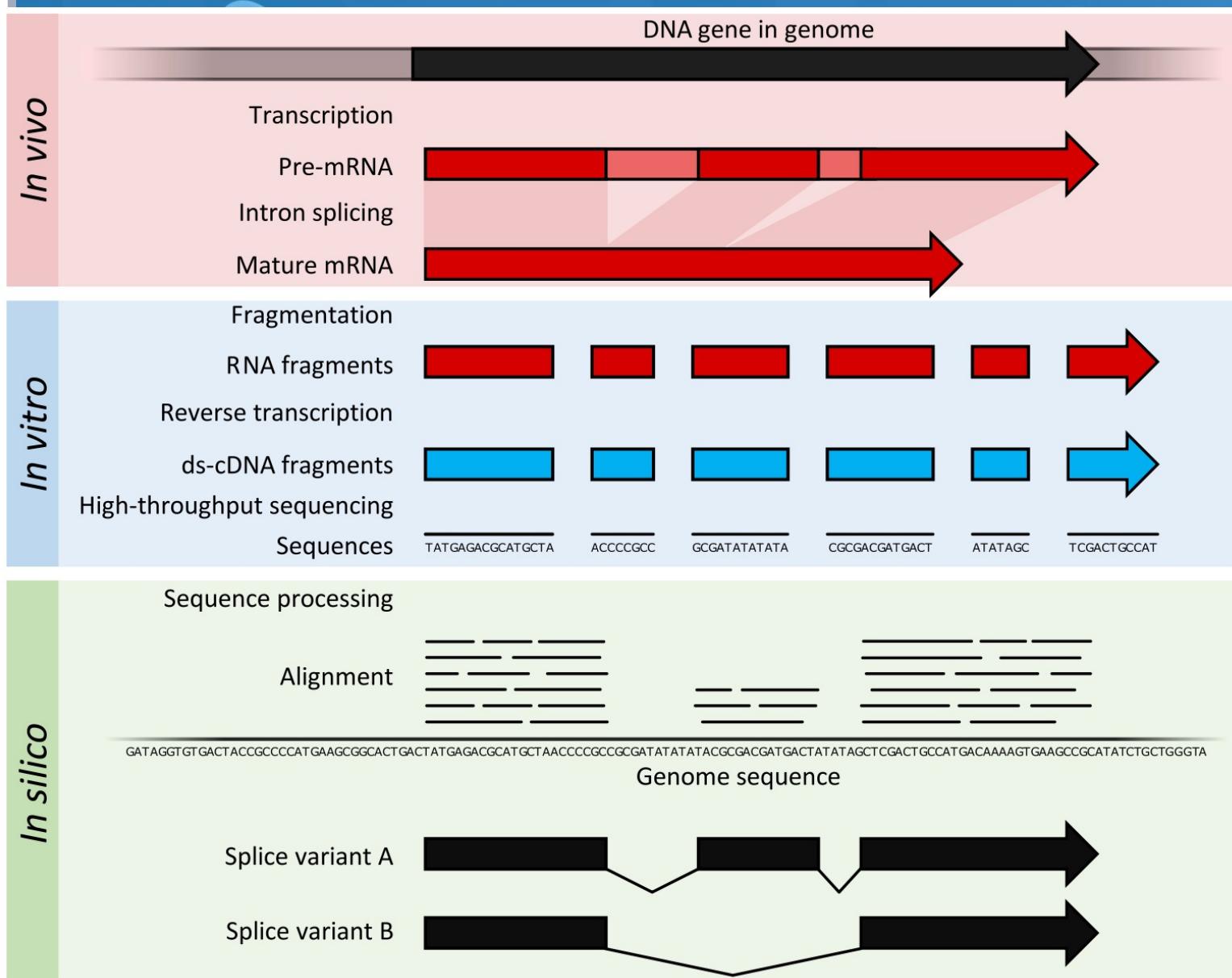


Las tecnologías transcriptómicas son las técnicas que se utilizan para estudiar el transcriptoma, la suma de todas las transcripciones de ARN. El contenido de información de un organismo se registra en el ADN de su genoma y se expresa a través de la transcripción. Aquí, el ARNm sirve como una molécula intermediaria transitoria en la red de información, mientras que los ARN no codificantes realizan diversas funciones adicionales. Un transcriptoma captura una instantánea en el tiempo de las transcripciones totales presentes en una célula.

Metodos transcriptomicos usados en el tiempo



RNA-seq



Fuente: <https://doi.org/10.1371/journal.pcbi.1005457.g004>

2. Aplicaciones



Aplicaciones

1. Diagnóstico y perfilado de enfermedades (con un análisis de Expresión diferencial)
2. Transcriptomas humanos y patógenos
3. Respuesta de un organismo al ambiente
4. Anotación de función genética
5. Análisis de RNA no codificante (Non codifying RNA)



¿cómo sabemos que un genoma esta secuenciado?

Resource List (A-Z)

All Resources

Chemicals & Bioassays

Data & Software

DNA & RNA

Domains & Structures

Genes & Expression

Genetics & Medicine

Genomes & Maps

Homology

Literature

Proteins

Sequence Analysis

Taxonomy

Training & Tutorials

Variation

All

Databases

Downloads

Submissions

Tools

How To

Databases

Assembly

A database providing information on the structure of assembled genomes, assembly names and other meta-data, statistical reports, and links to genomic sequence data.

BioProject (formerly Genome Project)

A collection of genomics, functional genomics, and genetics studies and links to their resulting datasets. This resource describes project scope, material, and objectives and provides a mechanism to retrieve datasets that are often difficult to find due to inconsistent annotation, multiple independent submissions, and the varied nature of diverse data types which are often stored in different databases.

Database of Genomic Structural Variation (dbVar)

The dbVar database has been developed to archive information associated with large scale genomic variation, including large insertions, deletions, translocations and inversions. In addition to archiving variation discovery, dbVar also stores associations of defined variants with phenotype information.

Genome

Contains sequence and map data from the whole genomes of over 1000 organisms. The genomes represent both completely sequenced organisms and those for which sequencing is in progress. All three main domains of life (bacteria, archaea, and eukaryota) are represented, as well as many viruses, phages, viroids, plasmids, and organelles.

Genome Reference Consortium (GRC)

The Genome Reference Consortium (GRC) maintains responsibility for the human and mouse reference genomes.

Results by database

Results found in 34 databases

Literature

Bookshelf	4,099
MeSH	277
NLM Catalog	516
PubMed	91,318
PubMed Central	126,647

Genes

Gene	50,696
GEO DataSets	10,825
GEO Profiles	165,553
HomoloGene	20
PopSet	524

Proteins

Conserved Domains	444
Identical Protein Groups	5,868,827
Protein	59,785,328
Protein Clusters	9,102
Sparcle	1,357
Structure	1,850

Genomes

Assembly	10,954
BioCollections	3
BioProject	4,129
BioSample	301,200
Genome	9
Nucleotide	3,020,537
Probe	1,719
SRA	300,427

Genetics

ClinVar	7
dbGaP	13
dbSNP	0
dbVar	0
GTR	1
MedGen	61
OMIM	111

Chemicals

BioSystems	23,913
PubChem BioAssay	6,605
PubChem Compound	12
PubChem Substance	1,387



Cuando existe un genoma de referencia

Salmonella enterica

Reference genome: **Salmonella enterica subsp. enterica serovar Typhimurium str. LT2**

Download sequences in FASTA format for [genome](#), [protein](#)

Download genome annotation in [GFF](#), [GenBank](#) or [tabular](#) format

BLAST against [Salmonella enterica genome](#), [protein](#)

All 5188 genomes for species:

Browse the [list](#)

Download sequence and annotation from [RefSeq](#) or [GenBank](#)

Display Settings: [Overview](#)

Send to:

[Organism Overview](#) ; [Genome Assembly and Annotation report \[5188\]](#) ; [Genome Tree report \[5188\]](#) ; [Genome Groups report \[33\]](#) ;

[Plasmid Annotation Report \[309\]](#)

ID:



Salmonella enterica

Causes enteric infections

Lineage: [Bacteria\[11938\]](#); [Proteobacteria\[3723\]](#); [Gammaproteobacteria\[1481\]](#); [Enterobacterales\[277\]](#); [Enterobacteriaceae\[129\]](#); [Salmonella\[3\]](#); [Salmonella enterica\[1\]](#)

Salmonella. This group of *Enterobacteriaceae* have pathogenic characteristics and are one of the most common causes of enteric infections (food poisoning) worldwide. They were named after the scientist Dr. Daniel Salmon who isolated the first organism, *Salmonella choleraesuis*, from the intestine of a pig. There are now two [More...](#)



Cuando no existe un genoma de referencia: nopal...

Opuntia X Search

NCBI Databases

Results found in 18 databases for: **Opuntia**

Literature	
Bookshelf	25
MeSH	1
NLM Catalog	5
PubMed	870
PubMed Central	1,849

Genes	
Gene	7
GEO DataSets	1,051
GEO Profiles	0
HomoloGene	0
PopSet	395
UniGene	0

Genetics	
ClinVar	0
dbGaP	0
dbSNP	0
dbVar	0
GTR	0
MedGen	1
OMIM	0

Proteins	
Conserved Domains	0
Identical Protein Groups	493
Protein	5,421
Protein Clusters	0
Sparcle	0
Structure	0

Genomes	
Assembly	0
BioCollections	0
BioProject	12
BioSample	127
Genome	0
Nucleotide	3,580
Probe	0
SRA	14
Taxonomy	1

Chemicals	
BioSystems	0
PubChem BioAssay	24
PubChem Compound	0
PubChem Substance	5



3. Flujo de trabajo

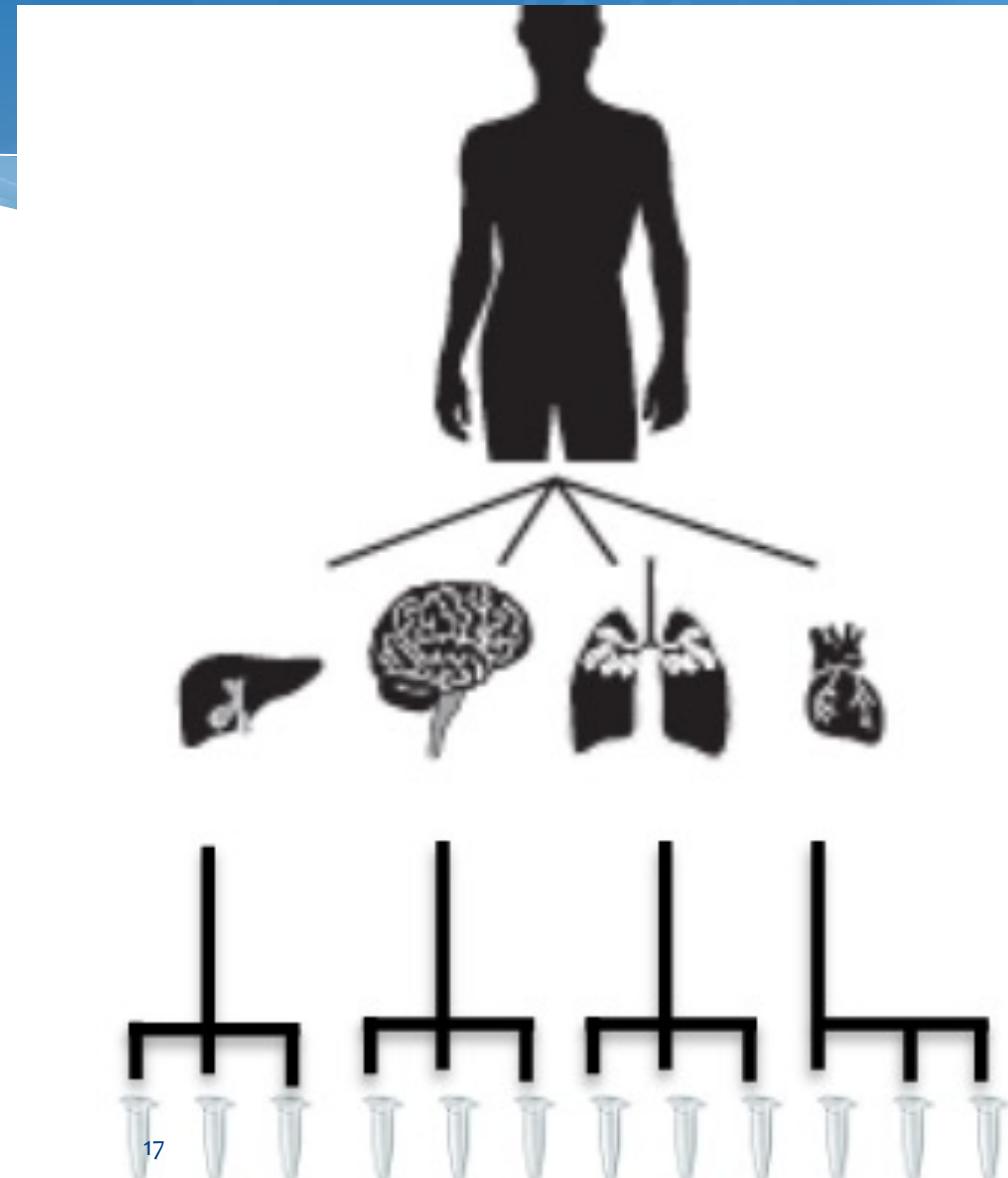
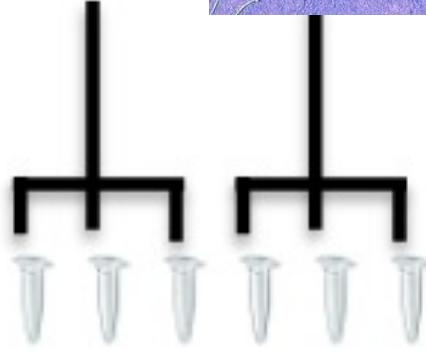
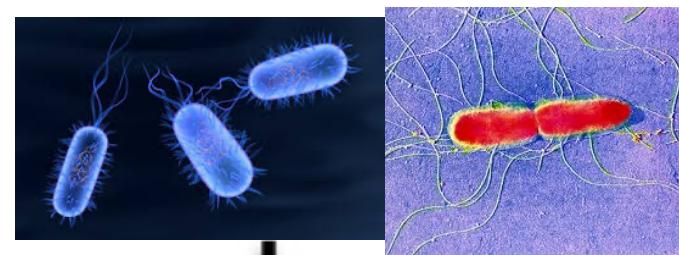


Flujo de Trabajo de RNA-seq

1. Extracción y secuenciación de RNaseq
2. Quality Control
3. Ensamblado de Novo (cuando no ha sido reportado el transcripto)
4. Alineamiento
5. Cuantificación
6. Análisis de expresión diferencial



Paso 0. Preparación de Muestras



Fuente: (Modificada) Nat Protoc. 2013 Aug; 8(8): 10.1038/nprot.2013.084.



UUSMB
UNIDAD UNIVERSITARIA DE
SECUENCIACIÓN MASIVA Y BIOINFORMÁTICA

Paso 1. Secuenciación



Sample1_rep1_R1.fastq Sample1_rep1_R2.fastq
Sample1_rep2_R1.fastq Sample1_rep2_R2.fastq
Sample1_rep3_R1.fastq Sample1_rep3_R2.fastq
Sample2_rep1_R1.fastq Sample2_rep1_R2.fastq
Sample2_rep2_R1.fastq Sample2_rep2_R2.fastq
Sample2_rep3_R1.fastq Sample2_rep3_R2.fastq
Sample3_rep1_R1.fastq Sample3_rep1_R2.fastq
Sample3_rep2_R1.fastq Sample3_rep2_R2.fastq
Sample3_rep3_R1.fastq Sample3_rep3_R2.fastq

more x.fastq

```
@HISEQ-MFG:495:C5WW4ACXX:2:1101:10005:5422
CGCCCTCCTACTGGTTGGACGTTCTGTTGCCTCAGCTAAGGCCGCGACTTCGGCTGCAGCC
+
BBDDDDDDCAABBDDDDDBADDDBDDDDDBDDDDDDDDDDDFHJJJIFIJJJJJJJJJJIDIG
@HISEQ-MFG:495:C5WW4ACXX:2:1101:10011:54197
GAAAAGTCTATTCGGTAAAATAGACAAAGATGTTGGGGATACCCAGAATTAGATCAGGA
+
@C@FFFABFFHBBHCGFHHIIJJIBHHIJJ9EF9EGHIDGGGJIJJIEIJGFGHEEGCGHGEIJG
@HISEQ-MFG:495:C5WW4ACXX:2:1101:10021:45670
TTCTCGGACAAGAGCTCCTTAAGCTTGCAGTACTCGGTTTCTCGCTGATCTGCCTC
+
DDB<BCCDDDDDDDDCCDDDDDDDDCEFFFDBFGFIJIIIGJJJJHHFJJJJJJJJIIIHJJ
...
...
```

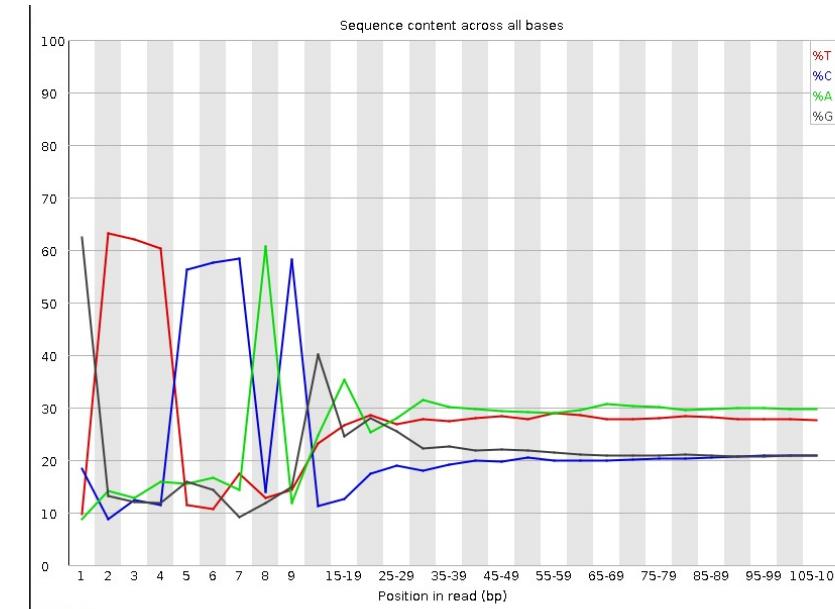
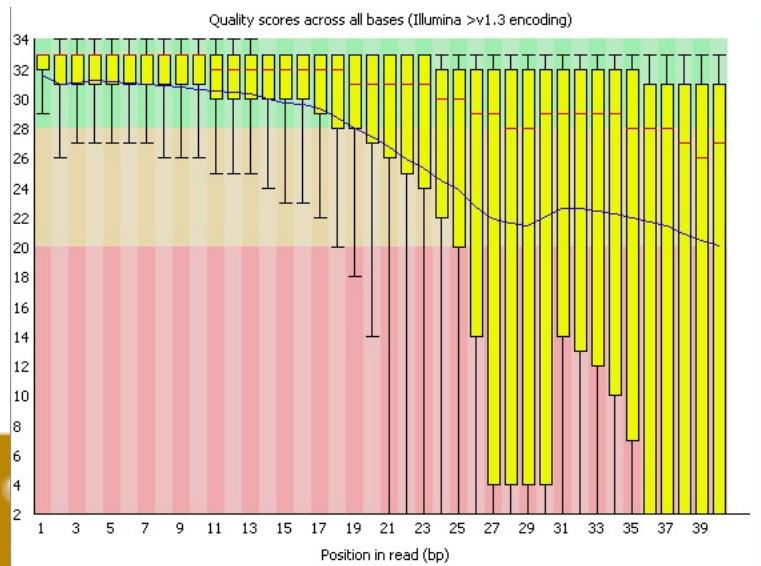
2. Quality control

1. Extracción y secuenciación de RNAseq
2. Quality Control
 - Fastqc
 - FaQCs
3. Ensamblado de Novo (cuando no ha sido reportado el transcripto)
4. Alineamiento
5. Cuantificación
6. Análisis de expresión diferencial



Paso 2. Quality control

Antes de hacer un proceso de ensamblado, es recomendable realizar un análisis de calidad, para identificar secuencias sobrerepresentadas, presencia de adaptadores o algún tipo de contaminación que pueda alterar los resultados.



Realiza el reporte de calidad de algún par de secuencias o todas

```
$ mkdir QualityReport  
$ fastqc -o QualityReport \  
/Share/Transc/Data/Fructanos_1_R1.fastq
```

Para todos:

```
$ fastqc -o QualityReport /Share/Transc/Data/*.fastq
```



Podemos llevarlas a otra máquina para verlas en un navegador...

```
$ scp -P 265 -r  
alumnoXX@bioinformatica.insp.mx:/home/alumno81/PT/QualityReport .
```



En un navegador abre cualquiera de los html

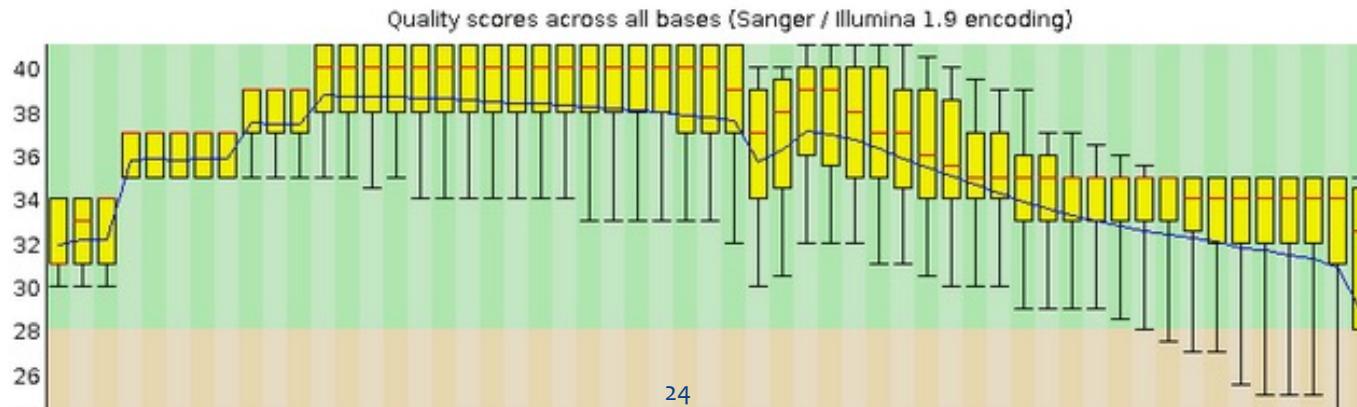


Basic Statistics

Measure	Value
Filename	Glicerol_3_R2.fastq
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	102820
Sequences flagged as poor quality	0
Sequence length	101
%GC	49



Per base sequence quality



RNA-seq análisis

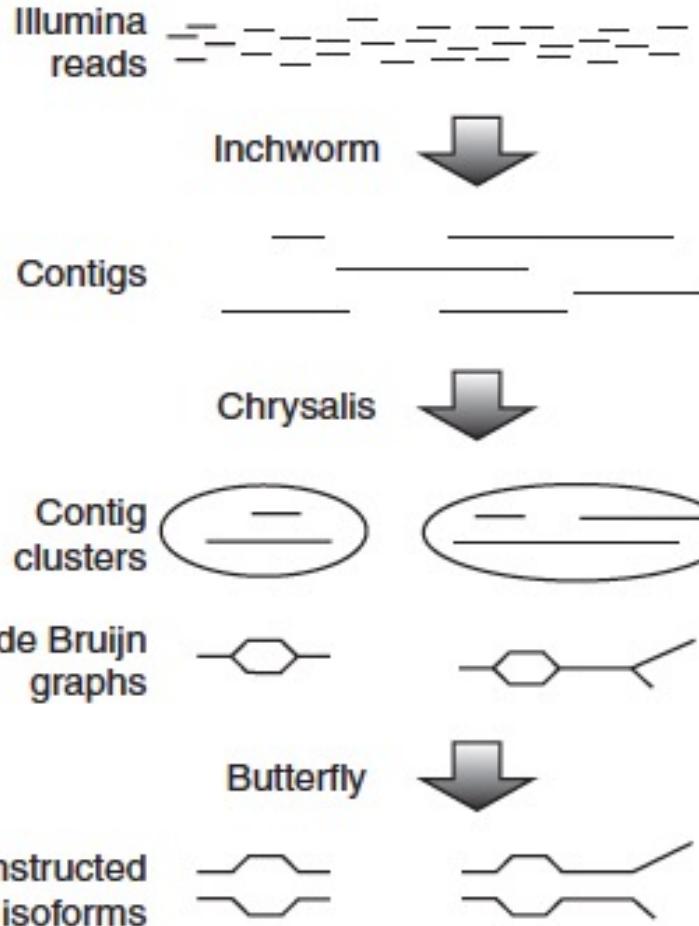
1. Extracción y secuenciación de RNaseq
2. Quality Control
3. Ensamblado de Novo (cuando no ha sido reportado el transcripto)
4. Alineamiento
5. Cuantificación
6. Análisis de expresión diferencial



Ensambladores

Software	liberado	Ultima actualización	Rcursos usados	Numero de referencias	Fortalezas y debilidades
Velvet-Oases [6]	2008	2018	Pesado	1708	Ensamblador original, ahora muy superado por otros
SOAPdenovo-trans [7]	2011	2015	Moderado	1004	De los primeros ensambladores genomicos, actualizado para hacer ensamble de transcriptomas
Trans-ABySS [8]	2010	2016	Moderado	669	Trabaja con lecturas cortas y genomas grandes, tiene opciones para trabajar en parelelo
Trinity [9]	2011	2023	Moderado	17310	Trabaja con lecturas cortas y genomas grandes, Computo intensivo
miraEST [10]	1999	2016	Moderado	1200	Trabaja con ssecuencias repetitivas, entrada de datos hibrido, amplio rango de plataformas permitido
Newbler [11]	2004	2012	Pesado	10597 (incluye genomas)	Especializado en Roche 454, Manejo del error de homo-polimeros
CLC [12]	2008	2014	Liviano	337	Interface gráfica, datos hibridos. Es de pagoa
BinPacker	2024	2016	Liviano	Sin publicación	
RNA-Bloom[13]	2020	2023	Moderado	43	Rapido y con manejo eficiente de memoria. Trabaja con RNAseq, single Cell, y lcrNAseq
rnaSPAdes[14]	2020	2024	Moderado	1054	Tiene protocolos para ensamblar bacterias, metagenomas, plasmidos , transcritos y cluster de genes biosinteticos

Trinity



Indication of computing resources



Single high-memory,
multicore server

Massively parallel on computing grid



4. Línea de comando típica de trinity



```

Trinity
#####
#
# [-----] | [-----] \ | [-----] | [-----] \ | [-----] | [-----] | [-----]
#           D ) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
#           / \ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
#           . \ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
#           [-----] | [-----] | [-----] | [-----] | [-----] | [-----]
#           ~ , | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
#
#####
# Required:
#   --seqType <string>      :type of reads: ( fa, or fq )
#   --max_memory <string>    :suggested max memory to use by Trinity where limiting can
be
#           enabled. (jellyfish, sorting, etc) provided in Gb of RAM, ie. '--max_memory
10G'
#
# If paired reads:
#   --left  <string>  :left reads, one or more file names (separated by commas,no
spaces)
#   --right <string>  :right reads, one or more file names (separated by commas,no
spaces)
#
# Or, if unpaired reads:
#   --single <string>   :single reads, one or more file names, comma-delimited
#####

```

Línea de comando típica de trinity

Un típico comando de Trinity para ensamblar datos de RNAseq strand no específico, puede lucir como el siguiente:

```
$ Trinity --seqType fq --max_memory 50G \
--left reads_1.fq.gz --right reads_2.fq.gz \
--CPU 6
```

```
$ Trinity --help
```



Línea de comando típica de trinity

Cuando se realiza un análisis de expresión diferencial es muy común tener más de un archivo. Uno por tejido o por tratamiento y con varias réplicas, en ese caso, es posible invocar a Trinity con más de un archivo fastq de entrada:

```
$ Trinity --seqType fq --max_memory 50G \
--left condA_1.fq.gz,condB_1.fq.gz,condC_1.fq.gz
--right \
condA_2.fq.gz,condB_2.fq.gz,condC_2.fq.gz \
--CPU 6
```

Nuevamente, la extensión gz, indica que los archivos están comprimidos.



Línea de comando típica de trinity

Trinity acepta un archivo describiendo las características de las muestras a analizar (opción “**--samples_file file**”)

```
$ cat sample
condA      condA      condA_1.fq.gz      condA_2.fq.gz
condB      condB      condB_1.fq.gz      condB_2.fq.gz
condC      condC      condC_1.fq.gz      condC_2.fq.gz
```



Preparando los datos

1. Genere una carpeta llamada PracticaTrinity
\$ mkdir PracticaTrinity
2. Entre en la carpeta:
\$ cd PracticaTrinity
3. Genere las ligas a los archivos fastq de la carpeta
/tmp/Transc/Data/*.*
\$ ln -s /tmp/Transc/Data/*.* .
4. Verifique que se generaron bien las ligas
\$ ls -ltr



Línea de ejecución típica

Realizamos el ensamblado con todos las replicas disponibles (6 en total, 12 archivos fastq), en segundo plano. Usamos 2G de memoria y un solo cpu:

```
$ Trinity --seqType fq --max_memory 2G --CPU 1 \
--left \
Fructanos_1_R1.fastq,Fructanos_2_R1.fastq,Fructanos_3_R1.fastq,Glicerol_1_R1.fastq,Glicerol_2_R1.fastq,Glicerol_3_R1.fastq \
--right \
Fructanos_1_R2.fastq,Fructanos_2_R2.fastq,Fructanos_3_R2.fastq,Glicerol_1_R2.fastq,Glicerol_2_R2.fastq,Glicerol_3_R2.fastq \
--output trinity_output1 2>&1 > run_all1.log &
```

Línea de ejecución típica

Otra opción, es usar el archivo descriptor:
Archivo_Muestras.txt

```
$ Trinity --seqType fq --max_memory 2G --CPU 1 \
--samples_file Archivo_Muestras.txt \
--output trinity_output > run_all.log 2>&1 &
```



Práctica 1

Consideré muestras en 2 condiciones (Fructuano y Glicerol), con 3 replicas cada una, en archivos en formato fastq, pareados, que se encuentran en el subdirectorio:

/tmp/Transc/Data/

para llevar a cabo el ensamblado de novo, usando trinity,

1. Conectarse al servidor bioinformatica.insp.mx
2. Si aun no lo ha hecho, haga la preparación de los datos con lo indicado en la diapositiva 33
3. Calcule cuantas Secuencias participaran en el ensamblado, para determinar la cantidad de GB
4. Nos preparemos para llevar a cabo el ensamblado de Novo usando trinity . Ejecute cualquiera de las versiones mostradas en la diapositiva 34 o 35



5. Monitoreando el Progreso de Trinity



Monitoreo de la ejecución de Trinity

Trinity puede fácilmente tomar varios días por lo que es muy útil tener la capacidad de monitorear el proceso y el estado (Inchworm, Chrysalis, Butterfly) . Hay algunas maneras generales de hacerlo:

1. Revisar como proceso de Trinity y verificar cuanta memoria esa consumo:

top

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
32853	vjimenez	20	0	1286212	20520	3228	S	179.0	0.0	1:14.71	bowtie-align-s
32854	vjimenez	20	0	125540	35904	2404	S	56.3	0.0	0:25.80	samtools
32855	vjimenez	20	0	746340	729944	2208	S	4.6	0.6	0:02.40	samtools
8	root	20	0	0	0	0	S	0.3	0.0	8:27.28	rcu_sched
68	root	20	0	0	0	0	S	0.3	0.0	2:21.11	rcuos/8
35229	vjimenez	20	0	5373648	339968	14852	S	119.0	0.3	0:11.30	java
35387	vjimenez	20	0	5373648	38648	14560	S	18.2	0.0	0:00.55	java



Monitoreando la ejecución de Trinity

2. Enviando el proceso a segundo plano. Hay que asegurarse de capturar la salida estándar ‘stdout’ y los errores ‘stderr’ mientras se corre el proceso de Trinity. Usando bash, esto se hace de la siguiente manera:

```
Trinity ... opts ... 2>&1 > run_all.log &  
tail -f run_all.log
```



archivo de salida
Estándar output
background



Tipo de libreria

1. Con el parámetro `-SS_lib_type` podemos especificar el tipo de librería con el que se prepararon las muestras



Read 1 Read 2

F	→	-
R	←	-
FR	→	←
RF	←	→ ⁴⁰



7. Revisando la salida del ensamblado



Al terminar...

Es posible que al termino de la ejecución de un proceso de trinity, se vean asi:
succeeded(88) 100% completed.

Number of Commands: 85

succeeded(85) 100% completed.

All commands completed successfully. :-)

** Harvesting all assembled transcripts into a single multi-fasta
file...

.

.

.

```
#####
Trinity assemblies are written to
/home/vjimenez/PracticaTrinity/trinity_output/Trinity.fasta
#####
```

Revisando la salida de trinity

Cuando Trinity termina de ejecutarse, se crea un archivo **Trinity.fasta** como archivo de salida en el subdirectorio de salida **Trinity_out_dir/** (o en el directorio de salida que se haya especificado con el opción **--output**)

Numero de Acceso Gen Isoformas

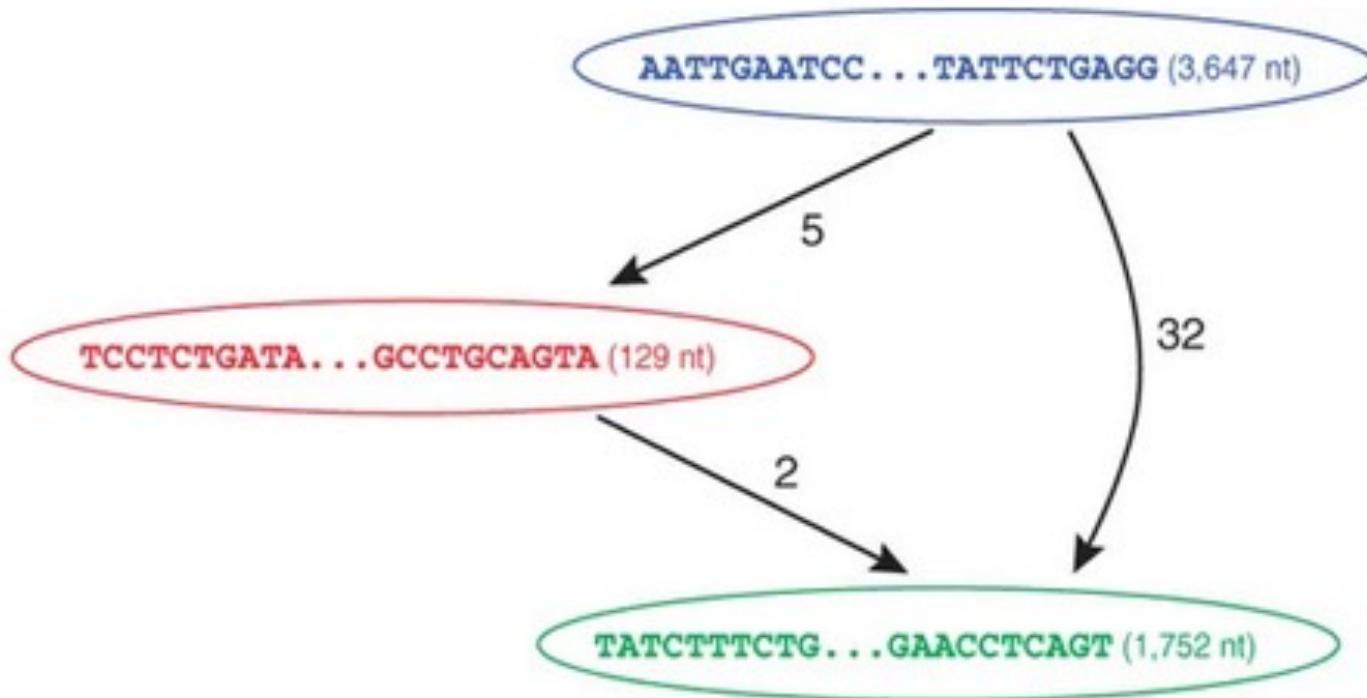
```
>TRINITY_DN53_c0_g1_i1 len=352 path=[0:0-351]
GGGTATTTATAAATGTACATATGATATAAATTGAAGTATAGCGTTCAAGCATTACCTTAAT
TCGTGCTTAGATTTATCATACCAGTATCCTCTGAGGGAACGACAATCCATTGCTAAATAGTT
CAGCATGTCCATCAGGTACTAGATTGATAATATCGTTGAAGTAAAATCCAATGATGATAA
CCAAGTAGGCCAGAATCAAATCGAACCTTGAAGTAGTTACTCTCCCTCCGAATGGACGT
GACAATAGAGCAGCATGCTACAATCCACGCCAAGTATCACACTCGGGAATAAGAGAACAA
ATACGTAGTCTTGAGGTTCTCCCAAGTGCTCGGT
```

Revisando la salida de Trinity

2 isoformas

```
>TRINITY_DN32_c0_g1_i2 len=1108 path=[0:0-69 2:70-1107]
AGATCTATCACTAATCCGATGCAAATCTTTGTTCTCACTGCTGTGAGATTCTAATGAATTAACTTACT
TTCTCTGGGAAGATGCAAAGTCGTTGGCAGTCTGAATATTACCCCTAACGCCATCAACTGCCACCTAACGA
GCTCAGACTCAGCGGGAGAAATCTTAGGGAGAATATTCGTGACCTTTCGGCACCATAGGTCAAAGGTAA
TGGGTAAGGCGAAGTAATCGATACCAAGCTCCTGCGAACCTCTCACCTCCAGGAAGACCCTCTAGG
.../...
>TRINITY_DN32_c0_g1_i1 len=1102 path=[1:0-63 2:64-1101]
ATCATATCAAGAAAAGCGGGTTGACAATAATAGGGTATCATCAATGCTCACTTACCGCTTCTCT
GGGAAGATGCAAAGTCGTTGGCAGTCTGAATATTACCCCTAACGCCATCAACTGCCACCTAACGAGCTCAG
ACTCAGCGGGAGAAATCTTAGGGAGAATATTCGTGACCTTTCGGCACCATAGGTCAAAGGTAAATGGGTA
AGGCGAAGTAATCGATACCAAGCTCCTGCGAACCTCTCACCTCCAGGAAGACCCTCTAGGTGAACA
.../...
```

Isoformas



Isoform A



Isoform B



Fuente: Nat Protoc. 2013 Aug; 8(8): 10.1038/nprot.2013.084.

Published online 2013 Jul 11. doi: [10.1038/nprot.2013.084](https://doi.org/10.1038/nprot.2013.084)

Revisando la salida de trinity

Una vez que el ensamblado esta completo, se puede conocer que tan “bueno” es, y seguramente comparar la calidad del ensamblado con ensamblados similares de distintos ensambladores ó distintas corridas con distintos parámetros.



Caracterizando la calidad de un ensamblado

1. Examinar la representación del ensamblado de las lecturas de RNASeq. Idealmente, al menos el 80% de los datos de entrada estarán representados en el transcriptoma ensamblado. El resto de las lecturas no ensambladas corresponden a transcriptomas con baja expresión, con cobertura insuficiente para ser ensamblados o de muy baja calidad o lecturas aberrantes.
2. Examinar la representación de los genes reconstruidos de longitud completa codificantes para proteínas, buscando los transcritos ensamblados a través de bases de datos de secuencias de proteínas conocidas

Caracterizando la calidad de un ensamblado

3. Calcular el E90N50 de los contigs. El valor del contig N50 esta basado sobre el conjunto de los transcritos que representan 90% de total de la expresión.



Calcular el N50 de los contigs

1. Sobre la base de las longitudes de los contigs ensamblados del transcriptoma, podemos calcular la longitud estadística Nx convencional, tal que al menos x% de los nucleótidos de transcripción reunidos se encuentran en contigs que son al menos de la longitud Nx. El método tradicional está calculando N50, de tal manera que al menos la mitad de todas las bases montadas están en contigs de transcripción de al menos el valor de longitud N50.



Calculando N50:

Primero, definamos la variable TRINITY_HOME:

```
$  
TRINITY_HOME=/usr/local/miniconda/envs/tools_bioinfo/opt/trinity-2.5.1/
```

Luego verificamos que fue correcta la asignación:

```
echo $TRINITY_HOME
```

Calculando N50:

```
$ $TRINITY_HOME/util/TrinityStats.pl trinity_output/Trinity.fasta
#####
## Counts of transcripts, etc.
#####
Total trinity 'genes':    109
Total trinity transcripts: 113
Percent GC: 45.43

#####
Stats based on ALL transcript contigs:
#####

Contig N10: 3366
Contig N20: 2133
Contig N30: 1888
Contig N40: 1748
Contig N50: 1434

Median contig length: 753
Average contig: 1006.62
Total assembled bases: 113748
```

Calculando N50:

```
#####
## Stats based on ONLY LONGEST ISOFORM per 'GENE':
#####
```

Contig N10: 3366

Contig N20: 2238

Contig N30: 1888

Contig N40: 1748

Contig N50: 1456

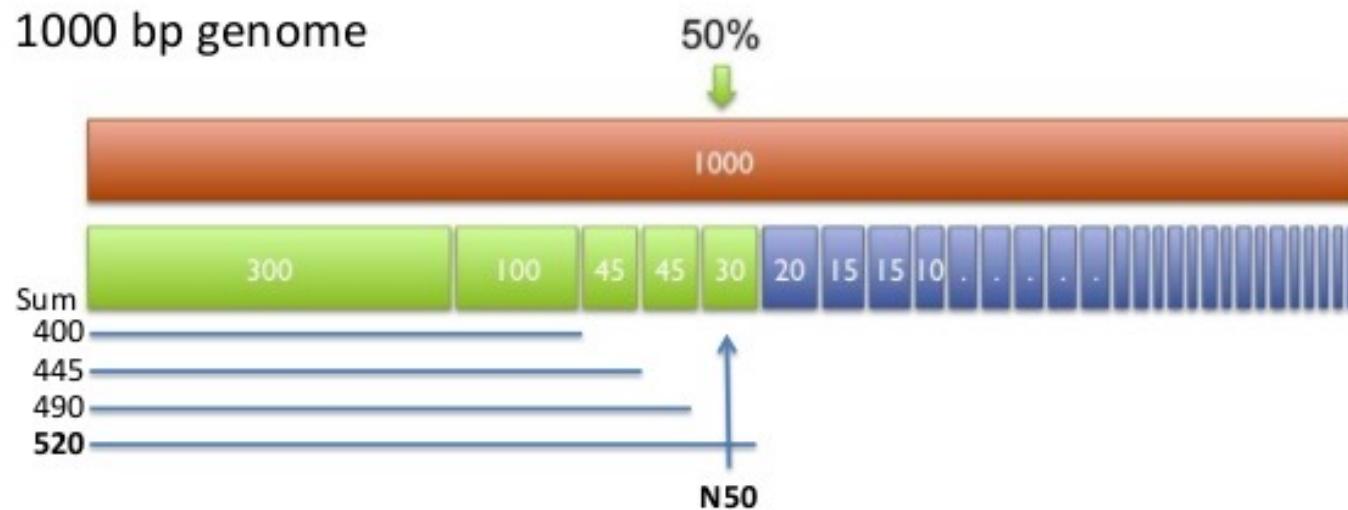
Median contig length: 753

Average contig: 1006.35

Total assembled bases: 109692

N50

50% of the genome is in contigs as large as the N50 value



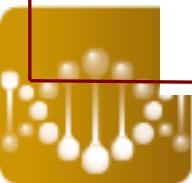
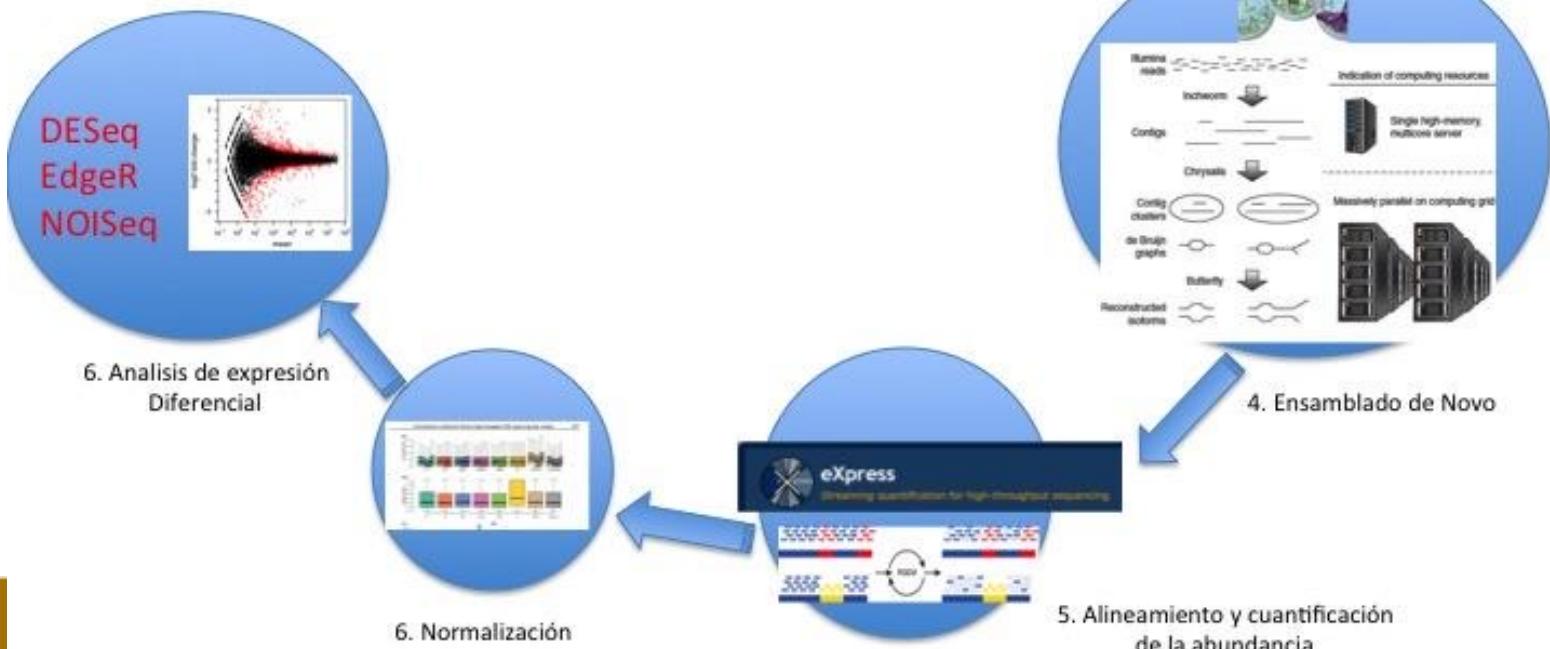
Analizar los transcritos

1. ¿Cómo podríamos ver si algunos de los transcritos obtenidos son partes de proteínas referenciadas?
2. ¿Qué datos necesita?

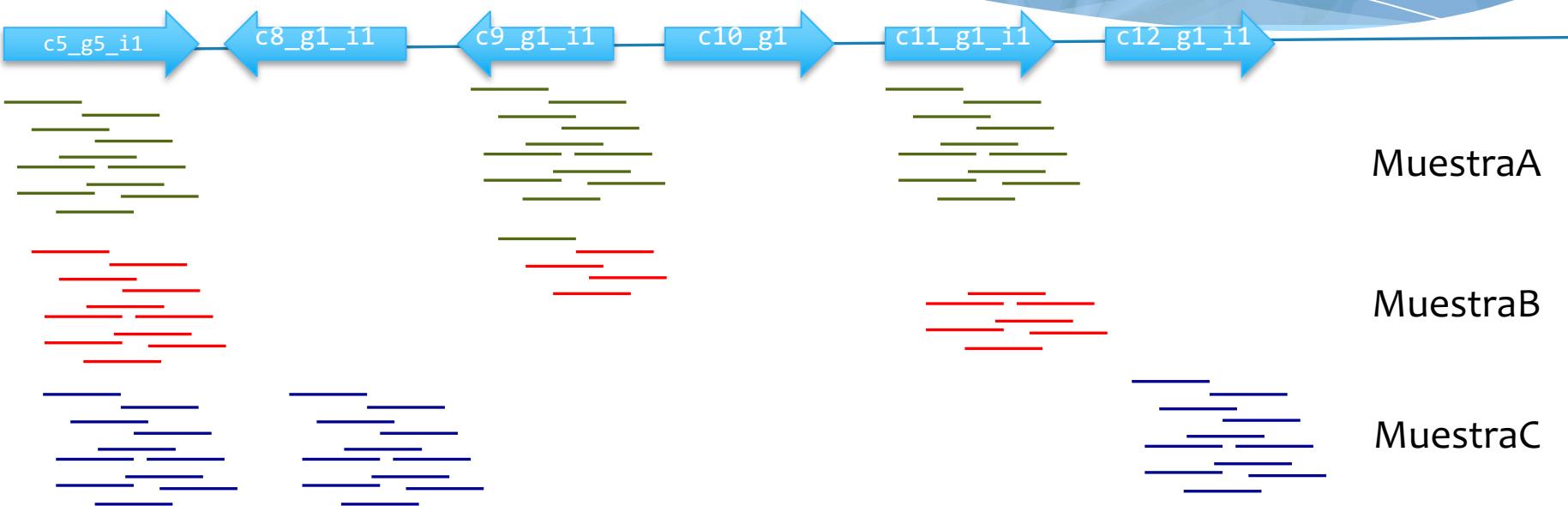


8. Cuantificación de los transcritos

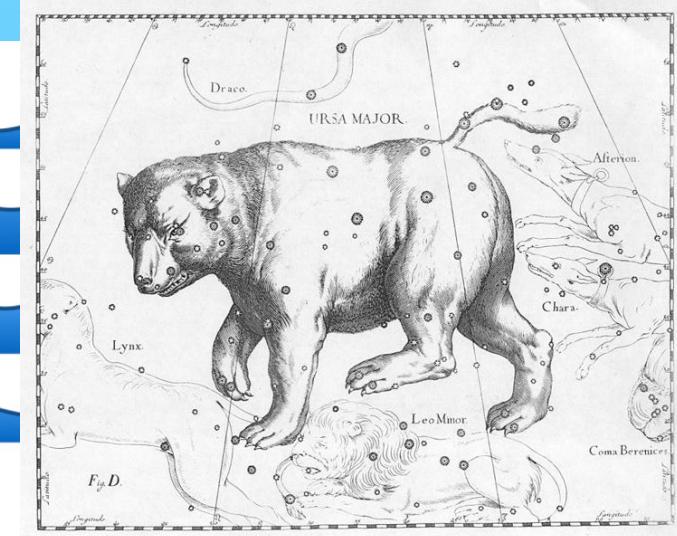
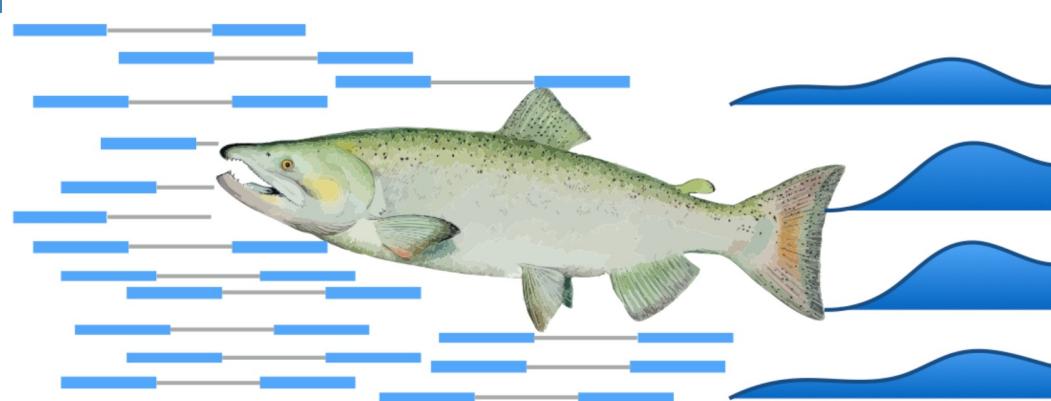




Cuantificación por transcripto

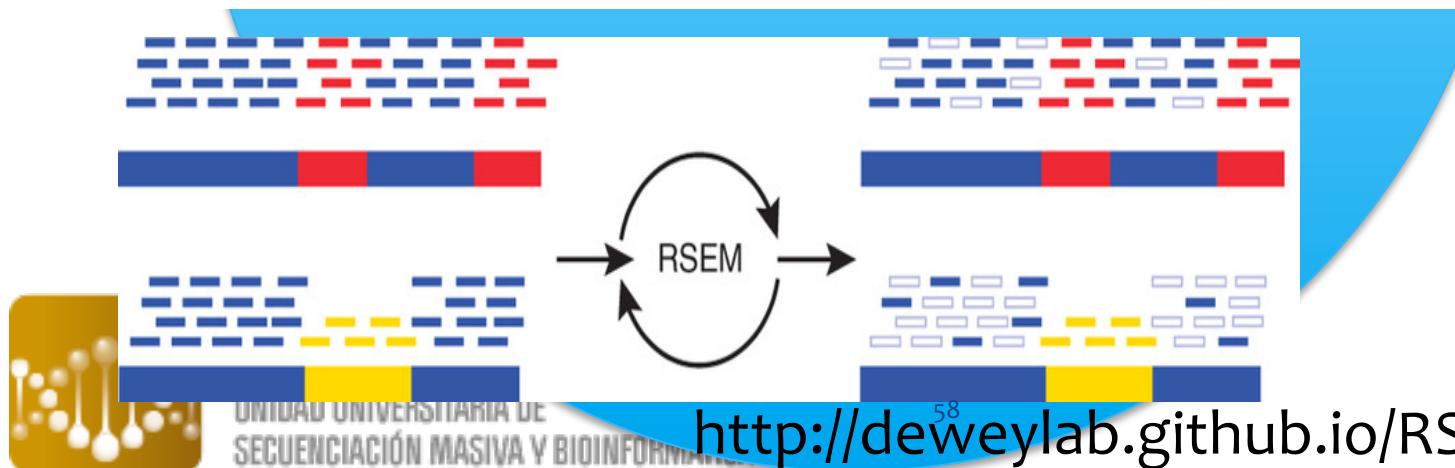


métodos basados en la cuantificación del alineamiento



Salmon —Don't count . . . quantify!

<https://pachterlab.github.io/kallisto>



Trinity soporta RSEM, eXpress y kallisto y salmon.

Trinity soporta RSEM y kallisto. y salmon

```
# --est_method <string>          abundance estimation method.  
#                                         alignment_based: RSEM|eXpress  
#                                         alignment_free: kallisto|salmon  
#  
# --output_dir <string>          write all files to output directory  
#  
#  
# if alignment_based est_method:  
#   --aln_method <string>          bowtie|bowtie2|(path to bam file)  
#                                         (note: RSEM requires bowtie)  
#                                         (if you already have a bam file,  
#                                         you can use it here instead of rerunning bowtie)
```



Preparamos el genoma de referencia

1. Preparamos el genoma de referencia para los alineamientos y estimación de la abundancia:

```
$TRINITY_HOME/util/align_and_estimate_abundance.pl \
--transcripts trinity_output/Trinity.fasta \
--est_method RSEM --aln_method bowtie2 \
--trinity_mode --prep_reference
```



Ejecución de la alineación y contabilización

2. Luego corremos el alineamiento y la estimación de la abundancia para cada uno de los tratamientos, para cuantificar la replica:

```
$TRINITY_HOME/util/align_and_estimate_abundance.pl \
--transcripts trinity_sample/Trinity.fasta \
--seqType fq \
--left Fructanos_1_R1.fastq \
--right Fructanos_1_R2.fastq --est_method RSEM \
--aln_method bowtie2 --trinity_mode \
--output_dir Fruct_rep1
```

Ejecución de la estimación de abundancias

Lo podemos hacer también con todas las muestras de una vez
En este caso, creara un directorio por replica.

```
$ $TRINITY_HOME/util/align_and_estimate_abundance.pl \
--transcripts trinity_output/Trinity.fasta \
--seqType fq \
--samples_file Archivo_Muestras.txt \
--est_method RSEM \
--aln_method bowtie2 --trinity_mode
```



- 
1. Además, de el número de lecturas mapeadas al transcripto, se calcula una medida normalizada considerando la longitud del transcripto y otra normalización considerando el tamaño de la librería.
 2. Las métricas de expresión normalizadas son reportadas como “*fragments per kilobase transcript length per million fragments mapped*” (FPKM)
 3. o '*transcripts per million transcripts*' (TPM).



Salida de RSEM

1. La aplicación de RSEM genera dos archivos de salidas principales conteniendo la información de la estimación de las abundancias:
2. **RSEM.isoforms.results**: conteo de lecturas por transcripto
3. **RSEM.genes.results**: conteo de lecturas por gene



Revisando la salida

3. Revisemos al final el archivo de conteos de genes o isoformas:

```
$ head Fructanos_rep1/RSEM.*results
==> Fructanos_rep1/RSEM.genes.results <==
gene_id transcript_id(s)    length effective_lengthexpected_countTPMFPKM
TRINITY_DN0_c0_g1  TRINITY_DN0_c0_g1_i1,TRINITY_DN0_c0_g1_i2 520.00368.64394.003862.955457.39
TRINITY_DN10_c0_g1 TRINITY_DN10_c0_g1_i1   2235.002083.64674.001169.141651.71
TRINITY_DN11_c0_g1 TRINITY_DN11_c0_g1_i1   2368.002216.64218.00355.46502.18

==> Fructanos_rep1/RSEM.isoforms.results <==
transcript_id gene_id length effective_lengthexpected_countTPMFPKMIsoPct
TRINITY_DN0_c0_g1_i1  TRINITY_DN0_c0_g1  520 368.64394.003862.955457.39100.00
TRINITY_DN0_c0_g1_i2  TRINITY_DN0_c0_g1  496 344.640.000.000.000.00
TRINITY_DN10_c0_g1_i1 TRINITY_DN10_c0_g1 22352083.64674.001169.141651.71100.00
```



Construyendo las matrices de Abundancias

1. Usando las estimaciones de abundancias para cada una de las muestras se construye una matriz de conteos y una matriz de valores de expresión normalizados, usando el siguiente script:



```

$ $TRINITY_HOME/util/abundance_estimates_to_matrix.pl

#####
#
# Usage:
#   $TRINITY_HOME/util/abundance_estimates_to_matrix.pl \
#     --est_method <method>  sample1.results sample2.results ...
# Required:
#
#   --est_method <string>      RSEM|eXpress|kallisto
#                               (needs to know what format to expect)
#
# Options:
#
#   --cross_sample_norm <string>      TMM|UpperQuartile|none
#                                       (default: TMM)
#
#   --name_sample_by_basedir          name sample column by
#                                       dirname instead of filename
#
#   --basedir_index <int>             default(-2)
#
#   --out_prefix <string>            default: 'matrix'

```



Construcción de la matriz de abundancias

```
$ $TRINITY_HOME/util/abundance_estimates_to_matrix.pl \
--est_method RSEM --name_sample_by_basedir\
--gene_trans_map
trinity_output/Trinity.fasta.gene_trans_map \
Fructanos_rep1/RSEM.isoforms.results \
Fructanos_rep2/RSEM.isoforms.results \
Fructanos_rep3/RSEM.isoforms.results \
Glicerol_rep1/RSEM.isoforms.results \
Glicerol_rep2/RSEM.isoforms.results \
Glicerol_rep3/RSEM.isoforms.results
```



Práctica 2.

1. Genere la matriz de abundancias a partir de los archivos de conteos de los 3 Fructanos y los 3 gliceroles. Recuerde que fueron generados con el metodo RSEM.



Práctica 3.

1. Utilizamos el método **kallisto** para realizar el mismo estudio de abundancia y obtener la matriz final
2. Para la matriz final, se utilizan los archivos “**abundance.tsv**”
3. ¿Notan una diferencia en velocidad, resultados?
4. Podrían realizar ahora el estudio de E90N50

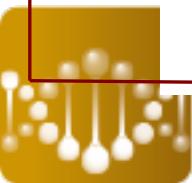
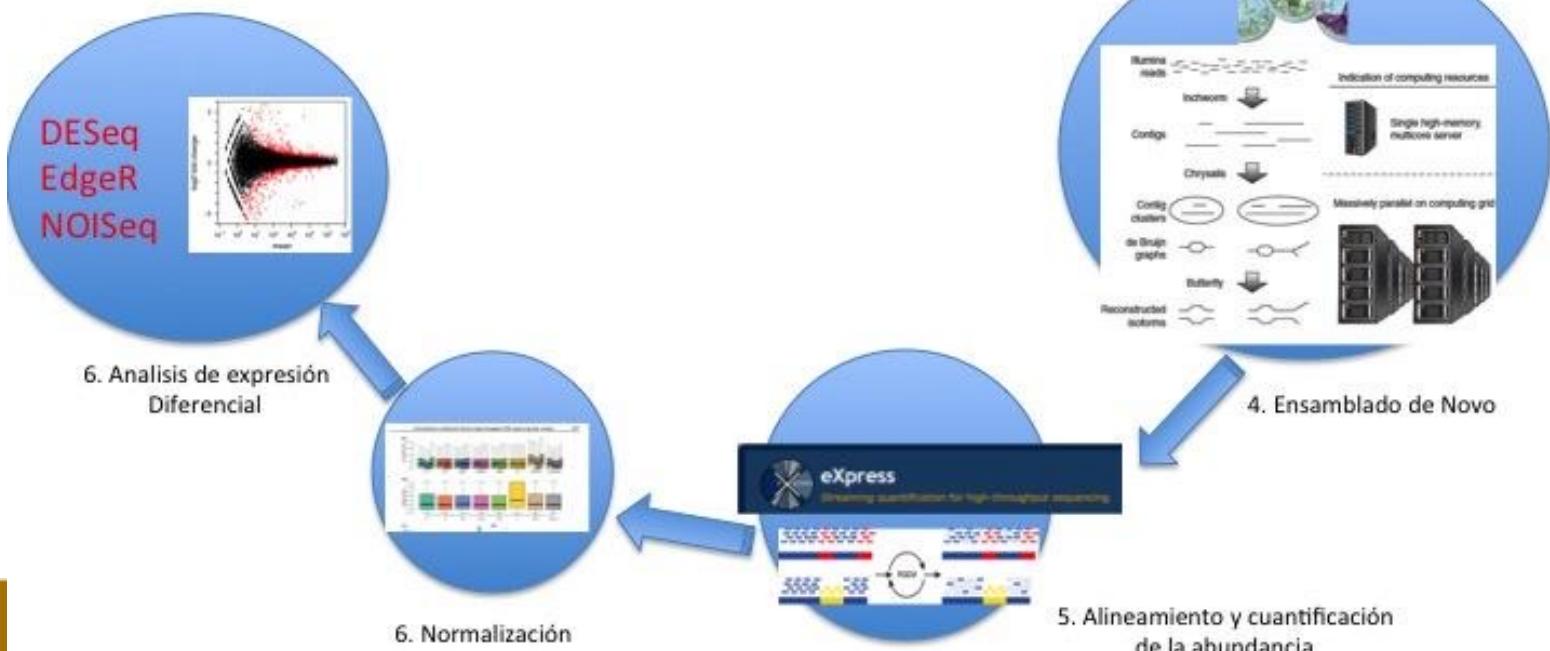


Estadístico ExN50

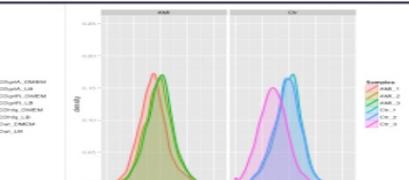
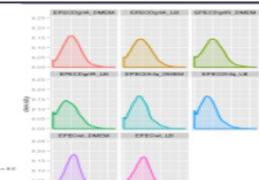
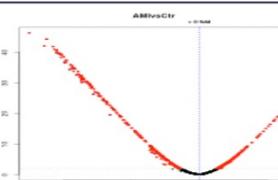
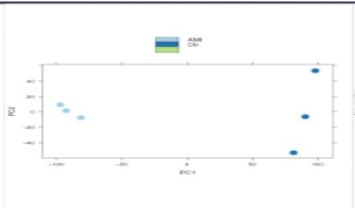
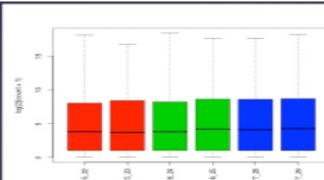
1. Una alternativa al estadístico Contig Nx que podría ser considerado más apropiado para los datos de ensamblaje de transcriptoma es el estadístico ExN50. Aquí, N50 se calcula igual que el anterior pero se limita a los mejores transcripciones más altamente expresados que representan x% del total de los datos de expresión normalizados. Esto requiere que se haya realizado la estimación de la abundancia de transcripción primero:

```
$ $TRINITY_HOME/util/misc/contig_ExN50_statistic.pl \
RSEM.isoform.TMM(EXPR.matrix trinity_output/Trinity.fasta | tee
ExN50.stats
```





<http://www.uusmb.unam.mx/ideamex/>



Integrated Diferencial Expression Analysis

[User's Guide](#)[Spanish](#)[Exit](#)

Welcome to IDEAmex

IDEAMEX *Integrated Differential Expression Analysis MultiExperiment* (<http://www.uusmb.unam.mx/ideamex/>) is a WEB server to perform differential expression analyses of RNA-Seq by four different Bioconductor packages, it also integrates the information outcomes. The IDEAMEX pipeline needs a raw count table for as many desired replicates and conditions, allowing the user to select which conditions will be compared, instead of doing all-vs-all comparisons. The whole process consists of three steps 1) Data Analysis: that allows a preliminary analysis for quality control based on the data distribution per sample, using different types of graphs; 2) Differential expression: performs the differential expression analysis using the bioconductor packages: edgeR[1], limma[2], DESeq2[3] and NOISeq[4], and generates reports for each method; 3) Result integration: the integrated results are reported using Venn diagrams and text lists. Our server allows an easy and friendly visualization of results, providing an easy interaction during the analysis process, as well as error tracking and debugging by providing output log files.

Request Analysis

Date: 07/01/25

email:

[Do you want to register as a user?](#)

File:

Seleccionar archivo Sin archivos seleccionados

[RNA-Seq example file1](#)

[RNA-Seq example file2 \(batch effect\)](#)

To download any of these files right-click on the link and select "save link as..."

Citation

Jiménez-Jacinto V, Sanchez-Flores A and Vega-Alvarado L (2019) Integrative Differential Expression Analysis for Multiple EXperiments (IDEAMEX): A Web Server Tool for Integrated RNA-Seq Data Analysis. Front. Genet. Vol 11, pg 279.
<https://doi.org/10.3389/faene.2019.00279>



Referencias

1. Lowe R, Shirley N, Bleackley M, Dolan S, Shafee T (2017) Transcriptomics technologies. PLoS Comput Bio 13(5): e1005457. <https://doi.org/10.1371/journal.pcbi.1005457>
2. FastQC: a quality control tool for high throughput sequence data. [Internet]. Babraham Institute [cited 2017 Apr 27]. <http://www.bioinformatics.babraham.ac.uk/projects/fastqc/>.
3. Lo CC & Chain PS. Rapid evaluation and quality control of next generation sequencing data with FaQCs. BMC Bioinformatics. 2014 15:366. <https://doi.org/10.1186/s12859-014-0366-2> PMID: 25408143
4. Nat Protoc. 2013 Aug; 8(8): 10.1038/nprot.2013.084. Published online 2013 Jul 11. doi: [10.1038/nprot.2013.084](https://doi.org/10.1038/nprot.2013.084)
5. Venket Raghavan, Louis Kraft, Fantin Mesny, Linda Rigerte, A simple guide to *de novo* transcriptome assembly and annotation, *Briefings in Bioinformatics*, Volume 23, Issue 2, March 2022, bbab563, <https://doi.org/10.1093/bib/bbab563>



A C I
T G G



0 1 1
1 0 0
0 1 0
0 1 1
1 0 0
1 1 1
0 0 0
1 1 1

A los Dres Tina y Ulises
Gracias!!

veronica.jimenez@ibt.unam.mx



UUSMB
UNIDAD UNIVERSITARIA DE
SECUENCIACIÓN MASIVA Y BIOINFORMÁTICA

